

Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung

zur Erlangung des akademischen Grades eines
DOKTORS DER NATURWISSENSCHAFTEN (Dr. rer. nat.)
der Fakultät Elektrotechnik, Informatik und Mathematik
der Universität Paderborn

genehmigte
DISSERTATION

von
M.Sc. Daniel Eckertz (geb. Nickchen)
aus Arnsberg

Tag des Kolloquiums: 15. November 2024
Referent: Prof. Dr.-Ing. Roman Dumitrescu
Korreferent: Prof. Dr.-Ing. Dr. phil. Dr. rer. soc. Carsten Röcker

Vorwort

Die vorliegende Dissertation entstand während meiner Tätigkeit als wissenschaftlicher Mitarbeiter und Gruppenleiter am Fraunhofer-Institut für Entwurfstechnik Mechatronik IEM. Sie ist das Ergebnis meiner wissenschaftlichen Arbeit im Rahmen von Forschungs- und Industrieprojekten.

Mein herzlicher Dank geht an Herrn Prof. Dr.-Ing. Roman Dumitrescu für die stets fordernde und förderliche Zusammenarbeit. Besonders danke ich ihm für das große Vertrauen, das er mir und meiner Arbeit auch in herausfordernden Zeiten entgegengebracht hat. Du und das von dir geschaffene Umfeld haben meine fachliche und persönliche Weiterentwicklung nachhaltig geprägt!

Herrn Prof. Dr.-Ing. Dr. phil. Dr. rer. soc. Carsten Röcker vom Institut für industrielle Informationstechnik (inIT) der TH OWL danke ich für die Übernahme des Korreferats.

Besonders bedanken möchte ich mich zudem bei Dr.-Ing. Harald Anacker, der mich meine gesamte bisherige Zeit am IEM fachlich und persönlich begleitet und gefördert hat. Danke für Deine Unterstützung und Dein Vertrauen.

Allen aktuellen und ehemaligen Arbeitskolleginnen und -kollegen danke ich für die kollegiale Zusammenarbeit und das inspirierende und motivierende Arbeitsklima. Darüber hinaus danke ich allen Studierenden, die mich durch ihre Abschlussarbeiten und studentischen Hilfstätigkeiten unterstützt haben.

Mein größter Dank gilt meiner Familie. Vielen Dank an meine Eltern, die mich in allen Lebenssituationen stets unterstützen und mir den Weg zur Promotion ermöglicht haben. Danke an meinen Bruder, der mir die Promotion vorgemacht hat. Und natürlich vielen Dank an meine Frau, die mich stets unterstützt und motiviert hat. Vielen Dank für deine Geduld und dein Verständnis.

Paderborn, im November 2024

Daniel Eckertz

Liste der veröffentlichten Teilergebnisse

- [EAD21] ECKERTZ, D.; ANACKER, H.; DUMITRESCU, R.: Augmented Reality-based Product Validation to Support Collaborative Engineering of Complex Technical Systems. International Conference on Remote Engineering and Virtual Instrumentation (REV). Online. 2021
- [EAD22a] ECKERTZ, D.; ANACKER, H.; DUMITRESCU, R.: Knowledge-based Interactive Configuration Tool for Industrial Augmented Reality Systems. International Conference on Information and Computer Technologies (ICICT). New York. 2022
- [EAD22b] ECKERTZ, D.; ANACKER, H.; DUMITRESCU, R.: Systematics for the individual assessment of augmented reality potentials to support product validation: How product and manufacturing design enable sustainable companies and societies. Proceedings of NordDesign 2022, 16th - 18th August 2022, The Design Society, 2022
- [EAD23] ECKERTZ, D.; ANACKER, H.; DUMITRESCU, R.: Modular Toolbox for Low-Code Development of Individual Augmented Reality Applications in Unity: International Conference on Remote Engineering and Virtual Instrumentation (REV). Thessaloniki, Griechenland, 2023
- [EBA+19] ECKERTZ, D.; BERSSENBRÜGGE, J.; ANACKER, H.; DUMITRESCU, R.: Work in Progress: Enhancing Collaboration using Augmented Reality Design Reviews for Product Validation on the Example of Additive Manufacturing. International Conference on Remote Engineering and Virtual Instrumentation (REV). Bangalore. 2019
- [EMA+21] ECKERTZ, D.; MÖLLER, M.; ANACKER, H.; DUMITRESCU, R.: Digital Knowledge Base for Industrial Augmented Reality Systems Based on Semantic Technologies. International Conference on Information and Computer Technologies (ICICT). Hawaii. 2021
- [DSE+20] DYCK, F.; STÖCKLEIN, J.; ECKERTZ, D.; DUMITRESCU, R.: Mixed Mock-Up - Systematic Development of an Interactive Augmented Reality System for Assembly Planning. International Conference on Human Computer Interaction (HCI). Kopenhagen. 2020
- [RBD+18] RÖLTGEN, D.; BANSMANN, M.; DUMITRESCU, R.; ECKERTZ, D.; WORTMANN, F.: Datenbrillen als Arbeitshilfe in der Industrie 4.0 – Bewertung von Anwendungsszenarien von Augmented Reality auf Basis einer Technologie-Roadmap. Symposium für Vorausschau und Technologieplanung. Berlin. 2018

Zusammenfassung

Die zunehmende Individualisierung von Produkten führt in Kombination mit dem Zeit- und Kostendruck auf globalen Märkten zu neuen Herausforderungen an die Produktentwicklung. Für eine erfolgreiche Produktentwicklung ist eine frühe und kontinuierliche Validierung mit den Kunden von enormer Bedeutung. Hier bietet die Visualisierungs-technologie Augmented Reality (AR) große Potentiale zur Effizienzsteigerung. Produkte können mit AR früh erlebt und erprobt und auch standortübergreifend validiert werden. Die technologische Komplexität von AR und der gleichzeitige Mangel an methodischer Unterstützung hindern viele Unternehmen jedoch an der Nutzung der Potentiale.

Zur *Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung* wird daher ein *Instrumentarium* erarbeitet. Das Instrumentarium unterstützt zunächst die Identifikation individueller Potentiale von AR als Validierungswerkzeug. Anschließend wird eine bedarfsgerechte Konfiguration des einzusetzenden AR-Systems ermöglicht. Darauf aufbauend werden Hilfsmittel zur aufwandsarmen Entwicklung individueller AR-Validierungssoftware bereitgestellt. Ein übergeordnetes Vorgehensmodell bildet den systematischen Rahmen und leitet die Planung und Vorbereitung an.

Die Anwendung und Evaluierung des Instrumentariums erfolgt exemplarisch anhand eines Regalbediengeräts, das individuell für einen Kunden entwickelt wird.

Summary

The increasing individualization of products in combination with the time and cost pressure on global markets leads to new challenges for product development. For successful product development, an early and continuous validation with the customers is of enormous importance. Here, the visualization technology Augmented Reality (AR) offers great potential for increasing efficiency. Products can be experienced and tested at an early stage with AR and be validated across locations. However, the technological complexity of AR and the accompanying lack of methodological support prevent many companies from exploiting the potentials.

A *toolset* is therefore developed for the *planning and preparation of AR-based validation in product development*. The toolset first supports the identification of individual potentials of AR as a validation tool. Subsequently, a demand-oriented configuration of the AR system to be used is enabled. Based on this, support tools are provided for the low-effort development of individual AR validation software. A high-level procedure model provides the systematic frame and guides the planning and preparation.

The application and evaluation of the toolset are carried out exemplarily based on a shelving control unit, which is developed individually for a customer.

Inhaltsverzeichnis Seite

1	Einleitung	1
1.1	Problematik	1
1.2	Zielsetzung	3
1.3	Vorgehensweise	4
2	Forschungsdesign	7
2.1	Design Research	7
2.2	Design Research Methodology	10
2.3	Einordnung und Vorgehen der vorliegenden Arbeit	11
3	Problemanalyse	15
3.1	Begriffsdefinitionen und Einordnung der Arbeit	15
3.2	Wandel der Produktentwicklung	16
3.2.1	Advanced Systems – Die Marktleistung von morgen	16
3.2.2	Individualisierung und Nutzerorientierung	19
3.2.3	Herausforderungen der heutigen Produktentwicklung	24
3.3	Validierung	27
3.3.1	Validierung im Kontext der Produktentwicklung	27
3.3.2	Ansätze zur Produktvalidierung	31
3.3.3	Kunden-integrierte Validierung	37
3.4	Augmented Reality	42
3.4.1	Interaktive Erweiterung der Realität	42
3.4.2	Augmented Reality im Kontext Validierung	48
3.4.3	Komplexität und Herausforderungen der Technologie	53
3.5	Problemabgrenzung	58
3.6	Anforderungen an die Arbeit	63
4	Stand der Technik	65
4.1	Systematische Planung AR-basierter Validierung	65
4.1.1	Das House of Quality nach KLEIN	65
4.1.2	Leitfragen-basierte Augmented Reality-Potential-Bewertung nach PALMARINI ET AL	68
4.1.3	Nutzen-Aufwands-Bewertung von Augmented Reality-Potentialen nach RÖLTGEN	69
4.1.4	Agile Practices Impact Model nach DIEBOLD UND ZEHLER	72

4.1.5 Konfiguration AR-basierter Validierungsumgebungen nach REINEMANN	73
4.2 Wissensbasierte AR-System-Konfiguration	76
4.2.1 Repräsentation und Verarbeitung von Wissen	77
4.2.1.1 Modelle und Modellierungssprachen	77
4.2.1.2 Semantische Technologien	80
4.2.2 Konfiguration technischer Systeme	85
4.2.2.1 AR-System-Klassifikation nach FELLMANN ET AL.	85
4.2.2.2 Bewertung von AR-Systemen nach RÖLTGEN.....	86
4.2.2.3 Konfiguration von Fahrsimulatoren nach HASSAN.....	89
4.2.2.4 Checklisten-basierte Evaluierung von AR-Displays nach PAELKE ET AL.....	92
4.2.2.5 AR-System-Konfiguration nach SCHILLING	95
4.2.2.6 Wissensbasierte Produktionssystemkonzipierung nach BAUER.....	97
4.3 Aufwandsarme Entwicklung von AR-Validierungssoftware	99
4.3.1 Entwicklung von AR-Anwendungen	99
4.3.1.1 AR-Entwicklungsprozess	99
4.3.1.2 Software Development Kits (SDK).....	103
4.3.2 Ansätze zur aufwandsarmen AR-Entwicklung	106
4.3.2.1 OpenXR	107
4.3.2.2 Forschungsprojekt AcRoSS.....	108
4.3.2.3 No-Code AR-Editoren	110
4.3.3 AR-Anwendungssoftware im Kontext Validierung	112
4.3.3.1 Aufbau AR-basierter Validierungsumgebungen nach REINEMANN	113
4.3.3.2 Bestehende Lösungen im Kontext AR-basierter Produktvalidierung	114
4.3.3.3 Ansätze für Feedback in AR	117
4.4 Bewertung und Handlungsbedarf	119
5 Instrumentarium zur Planung und Vorbereitung AR-basierter Validierung ..	123
5.1 Das Instrumentarium im Überblick	123
5.2 Systematik zur Nutzen-Aufwands-Analyse von AR-Potentialen	124
5.2.1 Die Systematik im Überblick	125
5.2.2 AR-Potential-Bewertungsgraph	128
5.2.2.1 Generische Definitionen und Bewertungen	128
5.2.2.2 Anwender-Bewertungen und deren Auswertung	135
5.2.3 Interaktives Hilfsmittel	137
5.2.3.1 Generalisierung der Anforderungen	139
5.2.3.2 Bestimmung der notwendigen Aufwände	140

5.2.3.3	Auswahl vielversprechender Anforderungen	140
5.2.3.4	Auswertung und Entscheidungsfindung	141
5.3	Wissensbasierte AR-System-Konfiguration	145
5.3.1	AR-System-Wissensbasis.....	146
5.3.1.1	Relevantes Wissen	147
5.3.1.2	Modellierung des Wissens	151
5.3.1.3	Maschinenlesbare Wissensrepräsentation	153
5.3.2	Interaktives Konfigurationswerkzeug für AR-Systeme	159
5.3.2.1	Software-Struktur.....	159
5.3.2.2	Interaktive AR-System-Konfiguration.....	160
5.4	Werkzeugkoffer zur aufwandsarmen Entwicklung individueller AR-Validierungsanwendungen	165
5.4.1	Das Konzept des Werkzeugkoffers im Überblick	167
5.4.2	Leitfaden zur Anleitung der Entwicklung.....	169
5.4.3	ARVKit - Augmented Reality-Validierungs-Framework.....	172
5.4.3.1	Templates	172
5.4.3.2	Modulare Funktionsbausteine.....	175
5.4.3.3	Hilfsmittel	184
5.5	Vorgehensmodell	187
6	Anwendung und Bewertung des Instrumentariums	189
6.1	Exemplarischer Anwendungsfall: Regal-Bediengerät.....	189
6.2	Anwendung des Instrumentariums	191
6.2.1	Phase 1 – Analyse zu validierender Produktanforderungen	192
6.2.2	Phase 2 – Entscheidung bzgl. AR als Validierungswerkzeug...	194
6.2.3	Phase 3 – AR-System-Konfiguration	195
6.2.4	Phase 4 – Entwicklung der AR-Validierungsanwendung	200
6.3	Bewertung des Instrumentariums	208
7	Zusammenfassung und Ausblick.....	213
8	Abkürzungsverzeichnis.....	217
9	Literaturverzeichnis.....	219

Anhang

A1	Ergebnisse der Unternehmensbefragung	A-1
----	--	-----

A2	Ergänzungen zum Stand der Technik.....	A-7
A2.1	Systematische Planung AR-basierter Validierung	A-7
A2.1.1	Hauptmerkmale eines Produkts nach PAHL ET AL.	A-7
A2.1.2	Wiedergabetreue von Prototypen	A-9
A2.2	Wissensbasierte AR-System-Konfiguration	A-10
A2.2.1	Die Struktur wissensbasierter Systeme	A-10
A2.2.2	Systematische Wissensmodell-Erstellung nach KOHN.....	A-12
A2.2.3	Technische Details zur Web Ontology Language (OWL)	A-14
A2.2.4	Technische und funktionale Aspekte eines AR-Systems.....	A-16
A2.2.5	AR-Endgeräte	A-18
A2.2.6	AR-Anwendungs- und Erfahrungswissen	A-20
A2.2.7	Einflussfaktoren auf AR-Systeme aus der Einsatzumgebung	A-21
A2.2.8	Das Ontologie-Modellierungswerkzeug Protégé.....	A-24
A2.3	Aufwandsarme Entwicklung von AR-Validierungssoftware	A-26
A2.3.1	Entwicklungsumgebung Unity	A-26
A2.3.2	Datenvorbereitung und -integration	A-29
A2.3.3	Ergänzungen zu AR-SDKs	A-31
A2.3.4	Studie zu AR-basierter Produktvalidierung	A-33
A2.3.5	Usability-Untersuchungen von PAES UND IRIZARRY.....	A-34
A3	Ergänzungen zum Instrumentarium.....	A-37
A4	Ergänzungen zur Anwendung und Bewertung	A-40

1 Einleitung

Die vorliegende Arbeit entstand im Rahmen der anwendungsorientierten Forschung am *Fraunhofer-Institut für Entwurfstechnik Mechatronik IEM*. Kern der Arbeit ist der Einsatz von Augmented Reality (AR) im Kontext der Produktvalidierung. Erste Ergebnisse dazu resultieren aus dem vom *Bundesministerium für Wirtschaft und Energie (BMWi)* geförderten Verbundprojekt „*DigiKAM – Digitales Kollaborationsnetzwerk zur Erschließung von Additive Manufacturing (AM)*“. Ziel des Projekts war eine digitale Kollaborationsplattform zur Erschließung von Additive Manufacturing (AM). Die Plattform vernetzt AM-Technologie-Experten¹ und AM-Kunden miteinander und unterstützt sie bei der gemeinsamen Entwicklung von AM-Bauteilen. Zur Unterstützung der Kollaboration wurde der Einsatz von AR zur effizienten Validierung der individuell entwickelten Bauteile durch den Kunden erforscht. Die Ergebnisse aus *DigiKAM* wurden im Rahmen weiterer Forschungs- und Industrieprojekte am Fraunhofer IEM vertieft und zur Anwendung in weiteren Branchen generalisiert. Damit ordnet sich die vorliegende Arbeit im Themenfeld Advanced Systems Engineering ein und beschreibt ein *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*.

1.1 Problematik

Die aktuellen Trends Digitalisierung, Globalisierung und Nachhaltigkeit haben einen starken Einfluss auf die zukünftigen Marktleistungen und deren Entstehungsprozesse. Es vollzieht sich ein Wandel von früheren Mechanik-zentrierten Systemen über mechatronische Systeme hin zu intelligenten, cyber-physischen Systemen. Diese sogenannten Advanced Systems lassen sich durch vier wesentliche Eigenschaften charakterisieren: Autonomie, Dynamische Vernetzung, soziotechnische Interaktion und Produkt-Service-Verzahnung [DAG+21]. Digitale Werkzeuge und Methoden haben bereits dazu geführt, dass Produkte häufig individuell für Kunden entwickelt werden [AR11, S. 15, Vog13]. Im Kontext der Advanced Systems wird die Individualisierung aus der Perspektive der Kunden und Nutzer weiter zunehmen [DAG+21]. Durch das Zusammenspiel der verschiedenen Disziplinen vereinen Advanced Systems dabei vielfältige für die Kunden relevante Produktmerkmale wie Gestalt, Optik, Funktionalität, Verhalten, Kinematik, Dynamik und Ergonomie. Der beschriebene **Wandel der Produktentwicklung** resultiert in einer steigenden Anzahl an Produktvarianten und immer kürzeren Produktlebenszyklen [GDE+18, S. 382, Ing20-ol]. So führt die kundenwahrnehmbare Individualisierung in Kombination mit dem Zeit- und Kostendruck auf globalen Märkten zu neuen Herausfor-

¹ Die Inhalte der vorliegenden Arbeit beziehen sich in gleichem Maße sowohl auf Frauen als auf Männer. Aus Gründen der besseren Lesbarkeit wird jedoch die männliche Form (Entwickler, Kunde etc.) für alle Personenbezeichnungen gewählt. Die weibliche Form wird dabei stets mitgedacht.

derungen an die Produktentwicklung [DAG+21]. Dabei sind unter anderem die Charakteristika Entwicklungszeit, Entwicklungskosten und Produktqualität ausschlaggebend [UE12, S. 2f.]. Als Erfolgsfaktoren gelten zum einen eine hohe Flexibilität der Unternehmen und zum anderen eine Veränderung der Unternehmenskultur. Unternehmen müssen ihre Entwicklungszyklen weiter verkürzen und Produkte bei gleicher oder gesteigerter Produktqualität schneller an den Markt bringen [DAG+21]. Ein möglicher Ansatz, mit dem dies gelingen kann, ist eine frühe und kontinuierliche Validierung mit den Kunden.

Validierung entspricht dem Abgleich der sich aus den Kundenwünschen und -anforderungen ergebenden Zielformulierungen und Lastenheftforderungen und den im Laufe der Produktentwicklung entstehenden Lösungen [Kli16]. Bei der Validierung wird entsprechend überprüft, ob das richtige Produkt entwickelt wird [ABK+16]. Regelmäßige Validierungsaktivitäten ermöglichen eine flexible und effiziente Reaktion auf sich ändernde Kundenanforderungen und die Berücksichtigung von Kundenfeedback in nachfolgenden Entwicklungsiterationen [GDE+18, S. 384, Wer20-ol, ARF+19, S. 429, SCK+17]. So kann der Kunden- und Anbieter-Nutzen gezielt gesteigert, Kundenwünsche und -anforderungen besser erfüllt und die Produktqualität letztendlich gesteigert werden [AHM+19, SCK+17]. Durch die steigende Komplexität der Systeme kommen klassische Validierungsansätze allerdings schnell an ihre Grenzen [Kli16]. Eine bisher übliche Entwicklung mit mehreren Prototypen und vorangehenden Kleinserien zur kontinuierlichen Fehlerreduktion wird für Unternehmen immer schwerer [Bög18-ol]. Physische Prototypen können oft erst spät in der Entwicklung hergestellt werden und sind zudem nicht zeit- und kosteneffizient [ARF+19, S. 429]. Digitale Validierungswerzeuge wie Simulationen und X-in-the-loop-Ansätze ermöglichen eine Validierung einzelner Produktmerkmale anhand digitaler Prototypen [GDE+18, S. 383]. Durch den Kunden zu validierende, meist subjektiv zu bewertende Produktaspekte werden durch diese Ansätze jedoch nicht abgedeckt. Durch die zunehmend menschzentrierte und multidisziplinäre Gestaltung der Marktleistungen sind diese Aspekte zunehmend relevant. Sie erfordern eine frühe und kontinuierliche Integration des Kunden in die Validierung. Hierzu mangelt es bisher allerdings an geeigneten Validierungsmethoden [DAG+21]. Unternehmen sehen daher speziellen Bedarf an der konsequenten Berücksichtigung des Menschen als Entwickler, Käufer und Nutzer der Marktleistungen. Demgegenüber stehen eine zunehmende geografische Trennung und flexible Arbeitsplätze und -zeiten, die die notwendige Zusammenarbeit erschweren [Bun15, S. 64, GDE+18, S. 384, DAG+21]. Hier bietet die Visualisierungs- und Kommunikationstechnologie Augmented Reality innovative Möglichkeiten.

Augmented Reality (AR) ermöglicht die interaktive und immersive Visualisierung digitaler Inhalte eingebettet in realen Umgebungen [TAB19-ol]. Digitale Daten und Modelle können frei im Raum oder präzise und maßstabsgetreu am zukünftigen Einsatzort in der Realität überlagert und betrachtet werden. Der Betrachter kann mit den dargestellten Inhalten interagieren und sie manipulieren. Im Kontext der Produktvalidierung ergeben sich enorme Potentiale durch AR [Gar20-ol, ARF+19, S. 430]. Dabei steht nicht die Validierung eines Systems als Ganzes im Fokus, sondern vielmehr die Absicherung einzelner

Produktmerkmale [PH17]. AR führt grundsätzlich zu einem besseren Verständnis der Entwicklung beim Kunden [ME19, LNO17]. So können Kunden dem Entwickler aussagekräftiges Feedback geben, das dieser wiederum in die Entwicklung einfließen lassen kann [AKN+19, RFH+19, ARF+19, S. 429]. Neben der reinen Visualisierung ermöglicht AR zudem eine sehr effiziente Vernetzung räumlich verteilter Akteure [VDC19-ol]. So mit sind mittels AR effiziente und aussagekräftige Validierungsaktivitäten auch über Distanzen hinweg möglich [PH17]. Allerdings stellt die technologische Komplexität von AR Unternehmen vor Herausforderungen [ME19, EM20]. Speziell im Kontext der Produktvalidierung fällt es Unternehmen aufgrund mangelnder methodischer Unterstützung schwer, die Potentiale von AR zu erkennen und zu nutzen [ARF+19, S. 429]. Neben der frühzeitigen Planung der Validierung stellt die bedarfsgerechte Entwicklung technischer AR-Lösungen die größte Herausforderung für Unternehmen bei der Erschließung der Potentiale von AR für die Validierung dar. Grundsätzlich mangelt es vielen Unternehmen an Knowhow und Ressourcen sowie einer systematischen Vorgehensweise, um AR bedarfsgerecht als Validierungswerkzeug zu nutzen und die genannten Potentiale zu erschließen [ARF+19].

Zusammenfassend kann gesagt werden, dass der Einsatz von AR als Validierungswerkzeug zu einer Effizienzsteigerung in der Produktentwicklung bei gleichzeitig verbesserter Produktqualität entsprechend der Kundenanforderungen führen kann. Wie eine Unternehmensbefragung ergeben hat, setzen viele Unternehmen bereits digitale Produktdaten zur standortübergreifenden Validierung mit Kunden ein (Anhang A1). Daher ändert sich die grundsätzliche Vorgehensweise durch den Einsatz von AR nicht. Allerdings sind Unternehmen aufgrund fehlender Ressourcen und mangelnder Expertise meist nicht in der Lage, den Einsatz von AR als Validierungswerkzeug systematisch zu planen und entsprechende technische AR-Lösungen bedarfsgerecht zu entwickeln. Es besteht daher der Bedarf an Leitfäden und Hilfsmitteln, die Unternehmen bei der eigenständigen Planung und Vorbereitung von AR-Lösungen als Validierungswerkzeug unterstützen.

1.2 Zielsetzung

Ziel der vorliegenden Arbeit ist ein *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*. Für die Validierung zuständige Akteure eines Unternehmens sollen das Instrumentarium in Entwicklungsprojekten einsetzen können, um den Einsatz von AR systematisch zu planen und technische AR-Lösungen als Validierungswerkzeug bedarfsgerecht zu konfigurieren und zu entwickeln.

Das Instrumentarium soll sich aus vier Bestandteilen zusammensetzen. Erster Bestandteil ist eine **Systematik zur Planung AR-basierter Validierung**. Die Systematik soll – ausgehend von individuellen Kundenanforderungen und Rahmenbedingungen – die Identifikation individueller Potentiale für den Einsatz von AR als Validierungswerkzeug ermöglichen. Zudem unterstützt die Systematik die Definition grundlegender technischer

und funktionaler Anforderungen an die AR-Lösung. Der zweite Bestandteil ist ein **Hilfsmittel zur wissensbasierten AR-System-Konfiguration**. Das Hilfsmittel ermöglicht die bedarfsgerechte Auswahl des AR-Geräts sowie die Definition notwendiger Softwarefunktionalitäten. Darauf aufbauend ist der dritte Bestandteil ein **Werkzeugkoffer zur aufwandsarmen Entwicklung individueller AR-Validierungssoftware**. Ein **übergeordnetes Vorgehensmodell** als vierter Bestandteil leitet den Einsatz der Hilfsmittel aus den anderen Bestandteilen an und ermöglicht somit die eigenständige Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung. Als Resultat liegt dem Unternehmen eine bedarfsgerechte AR-Lösung als Validierungswerkzeug vor. Die AR-Lösung kann im Rahmen des Entwicklungsprojekts eingesetzt werden und ermöglicht eine effiziente und aussagekräftige Validierung mit den Kunden.

Die Anwendbarkeit des Instrumentariums soll an einem praxisnahen Anwendungsbeispiel aus einem Forschungsprojekt nachgewiesen werden.

1.3 Vorgehensweise

In **Kapitel 2** werden zunächst das dieser Arbeit zugrundeliegende Forschungsdesign sowie das Vorgehen zur Entwicklung des angestrebten Instrumentariums beschrieben.

In **Kapitel 3** wird die Problemanalyse konkretisiert. Dazu werden zunächst relevante Begriffe definiert und anschließend das Forschungsfeld der Arbeit abgesteckt. Ausgangssituation der Problemanalyse ist der Wandel der Produktentwicklung. In diesem Kontext werden heutige Marktleistungen sowie deren Entwicklung analysiert. Die für eine erfolgreiche Entwicklung zunehmend an Bedeutung gewinnende Validierung wird in einem eigenen Kapitel näher betrachtet. Dabei werden bestehende Validierungsansätze sowie aktuelle Herausforderungen untersucht. Da Augmented Reality (AR) einen wesentlichen Schwerpunkt dieser Arbeit darstellt, wird die Technologie sowie mit ihr verbundene Potentiale und Herausforderungen im Kontext der Validierung analysiert. In einer Problemabgrenzung werden die dieser Arbeit zugrundeliegende Problematik zusammengefasst und abschließend Anforderungen an das angestrebte Instrumentarium abgeleitet.

Gegenstand von **Kapitel 4** ist eine Analyse des Stands der Technik. Zunächst werden Arbeiten untersucht, die sich mit der Klassifikation von Produktanforderungen sowie der systematischen Identifikation von Potentialen befassen. Zudem werden Ansätze analysiert, die eine wissensbasierte Konfiguration von AR-Systemen unterstützen. Dazu zählen Möglichkeiten der digitalen Dokumentation und Verarbeitung von Wissen zur Konfiguration technischer Systeme. Im Kontext der Entwicklung von AR-Softwareanwendungen werden Ansätze zur aufwandsarmen Programmierung und zur Reduzierung der Entwicklungsaufwände betrachtet. Den Abschluss des Kapitels bildet eine Bewertung der untersuchten Ansätze anhand der Anforderungen aus Kapitel 3.6. Die Bewertung belegt den bestehenden Bedarf am angestrebten Instrumentarium.

Den Kern der Arbeit bildet das **Kapitel 5**. In diesem wird das Instrumentarium zur Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung vorgestellt. Zunächst wird ein Überblick über das Instrumentarium gegeben und der Zusammenhang der einzelnen Bestandteile erläutert. Anschließend werden die vier Bestandteile des Instrumentariums im Detail beschrieben. Den Abschluss bildet das übergeordnete Vorgehensmodell, das die Planung und Vorbereitung systematisch anleitet.

Die Evaluierung des Instrumentariums erfolgt in **Kapitel 6**. Anhand eines praxisnahen Anwendungsfalls aus dem Forschungsprojekt DigiKAM wird das Instrumentarium exemplarisch angewendet. So können die Erfüllung der Anforderungen sowie der Nutzen des Instrumentariums evaluiert werden.

In **Kapitel 7** erfolgt eine Zusammenfassung der Arbeit. Zudem wird ein Ausblick auf zukünftige Arbeiten und mögliche Weiterentwicklungen des Instrumentariums gegeben.

Der **Anhang** umfasst ergänzende Informationen zu einzelnen Kapiteln dieser Arbeit.

2 Forschungsdesign

In diesem Kapitel wird das Forschungsdesign der vorliegenden Arbeit beschrieben. Das Forschungsdesign stellt die Grundlage der Entwicklung des angestrebten Instrumentariums dar und gewährleistet wissenschaftlich fundierte Ergebnisse. Zunächst wird in Kapitel 2.1 die Design Research als wissenschaftlicher Rahmen des Forschungsdesigns vorgestellt. Anschließend wird in Kapitel 2.2 die Design Research Methodology als domänenübergreifender Forschungsdesignansatz beschrieben. Darauf aufbauend erfolgt in Kapitel 2.3 die Beschreibung des Vorgehens sowie die Einordnung dieser Arbeit in die Design Research.

2.1 Design Research

Produkte und Produktionssysteme und somit übergreifend die Produktentstehung werden immer interdisziplinärer und komplexer [KWH13, S. 6]. In diesem Kontext sind kreative und innovative Problemlösungen notwendig. Diese Problemlösungen entstehen im Rahmen des **Designs**. Laut BLESSING UND CHAKRABARTI umfasst Design alle Aktivitäten, um ein Produkt zu erstellen [BC09]. Grundlage des Designs können ein Bedarf, eine Produktidee oder auch eine Technologie sein. Ziel des Designs ist die Entwicklung aller zur Realisierung eines Produkts und zur Erfüllung der wahrgenommenen Bedürfnisse des Nutzers und anderer Interessengruppen erforderlichen Artefakte. Ein Produkt ist dabei nicht auf ein physisches Produkt beschränkt. Die Entwicklung der Artefakte kann zudem von Einzelpersonen, Unternehmen oder Gemeinschaften und basierend auf verschiedenen Produktentwicklungsverfahren erfolgen. Die Resultate des Designs können technischer oder auch nichttechnischer Natur sein sowie physische und virtuelle Artefakte umfassen. Neben Wissen über die Ziele und den kompletten Lebenszyklus des Produkts ist Wissen über den Designprozess von besonderer Bedeutung. Der Designprozess beschreibt, wie man effektiv und effizient bei der Entwicklung der notwendigen Artefakte vorgeht. Designer müssen für die Gestaltung auf Wissen aus unterschiedlichen Domänen sowie auf Methoden und Werkzeuge zurückgreifen, um die Anwendung dieses Wissens zu unterstützen. Ist dieses Wissen nicht oder nur teilweise vorhanden, können Designer sich entweder auf Annahmen verlassen oder Forschung betreiben, um dieses Wissen zu generieren [BC09].

Diese Forschung erfolgt im Rahmen der **Design Research**, die sich mit der Generierung von Wissen über sowie für das Design befasst [Hor01]. Es gibt keine einheitliche Definition der Design Research sowie der Ziele und Vorgehensweisen. Oftmals wird als Ziel von Design Research die Verbesserung von Design in der Praxis definiert. Mit diesem Ziel wurde eine Vielzahl an Unterstützungs möglichkeiten entwickelt, die jedoch oft lediglich auf konzeptioneller Ebene blieben. BLESSING UND CHAKRABARTI sind der Meinung, dass die Entwicklung von Unterstützungsmaßnahmen zur Verbesserung des De-

signs effizienter und effektiver ist, wenn das Design besser verstanden wird. Ihre Definition von Design Research umfasst daher sowohl die Entwicklung eines besseren Verständnisses von Design sowie die Entwicklung von unterstützenden Hilfsmitteln für das Design. Nur durch eine Kombination dieser Aspekte können die beiden wesentlichen Ziele der Design Research erreicht werden [BC09]:

- Formulierung und Validierung von Modellen, Methoden, Werkzeugen und Theorien für das Design.
- Entwicklung und Validierung von Unterstützungsmaßnahmen auf der Grundlage der Modelle, Methoden, Werkzeuge und Theorien, um die Designpraxis effektiver und effizienter zu machen.

Eine **Strukturierung und Klassifikation von Forschungsbeiträgen** im Rahmen der Design Research kann anhand der Reife der Lösung in Kombination mit der Reife der Anwendungsdomäne erfolgen (Abbildung 2-1). Daraus ergeben sich vier Arten von Forschungsbeiträgen. *Routine-Designs* leisten keinen wesentlichen Wissensbeitrag, da lediglich bekannte Lösungen auf bekannte Probleme angewandt werden. Bei einer *Verbesserung* werden neue Lösungen für bekannte Probleme entwickelt. Eine *Zweckentfremdung* dagegen entspricht der Übertragung bekannter Lösungen auf neue Probleme. Werden neue Lösungen für neue Probleme erarbeitet, spricht man von einer *Erfindung* [GH13].



Abbildung 2-1: *Klassifikation von Forschungsbeiträgen nach GREGOR UND HEVNER [GH13], Darstellung in Anlehnung an [Jop21]*

GREGOR UND HEVNER definieren drei Level zur **Bewertung der Forschungsbeiträge** [GH13]. Den geringsten Wert hat eine praktische Umsetzung eines Design-Artefakts

durch Anwendung beispielsweise eines Modells oder einer Methode. Dies kann beispielsweise ein Software-Produkt als Resultat oder die Anwendung eines Design-Prozesses sein. Das zweite Level entspricht der Entwicklung einer Methode, eines Modells oder einer Design-Richtlinie. Übergreifende, umfangreiche Designtheorien zu speziellen Phänomenen werden Level 3 zugeordnet. Mit aufsteigendem Level steigt der Grad der Abstraktion, der Vollständigkeit sowie der Reife des Wissens.

In der Design Research sind drei wesentliche **Herausforderungen** zu adressieren. Zunächst existiert eine Vielzahl an Forschungsbeiträgen. Daraus resultiert ein fehlender *Überblick über die Forschungslandschaft*. Darunter fällt auch ein Mangel an eindeutigen Begriffsdefinitionen sowie eine fehlende Verifikation und Validierung von existierenden Ansätzen. Aufgrund eines oft fehlenden Verständnisses der Wissenschaftler über die industriellen Bedarfe und die Praxis, werden viele wissenschaftliche Ansätze nicht in der Praxis angewendet. Es mangelt somit an der *Ergebnisverwertung*. Durch die Anwendung ungeeigneter Vorgehensweisen sowie die schwierige Überprüfung der Güte von wissenschaftlichen Arbeiten herrscht zudem ein *Mangel an wissenschaftlicher Sorgfalt* [BC09].

Zur Adressierung dieser Herausforderungen beschreibt HEVNER drei **Design Research Zyklen**, die das Verständnis, die Durchführung sowie die Auswertung von Design Research-Projekten unterstützen (Abbildung 2-2). Das Umfeld umfasst die Aspekte Mensch, Organisation und Technik sowie deren Ziele, Aufgaben, Probleme und Potentiale. Im Rahmen des *Relevanz-Zyklus* werden die Anforderungen an die Forschung aus dem Umfeld analysiert und erarbeitet. Dies erfolgt beispielsweise durch Feldversuche. Im anschließenden *Design-Zyklus* erfolgen die iterative Erstellung und Evaluation der Design-Artefakte und somit die Erarbeitung des Forschungsbeitrags. Im *Strenge-Zyklus* werden zum einen die Grundlagen der Entwicklung als Wissensbasis erarbeitet und zum anderen die erarbeiteten Artefakte und das gewonnene Wissen wissenschaftlich aufbereitet und zur Ergänzung der Wissensbasis des jeweiligen Forschungsfelds bereitgestellt. Nach HEVNER sind diese drei Zyklen in allen wissenschaftlichen Forschungsaktivitäten explizit zu durchlaufen [Hev07].

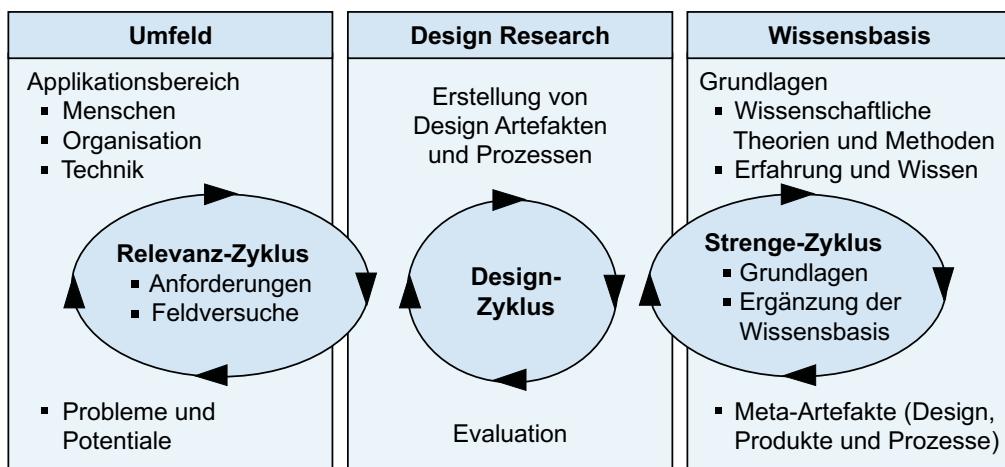


Abbildung 2-2: *Design Research Zyklen nach HEVNER [Hev07], Darstellung in Anlehnung an [Jop21]*

In Ergänzung zu den Design Research Zyklen ist eine **Design Research Methodik** notwendig [BC09]. Die Methodik kann auch als Forschungsdesignansatz verstanden werden. Sie beschreibt Methoden und Richtlinien, die die Durchführung der Forschungsaktivitäten unterstützen und anleiten. Darin inbegriffen sind die Analyse des Problems, die Entwicklung der Artefakte sowie die Validierung der Ergebnisse.

Abbildung 2-3 veranschaulicht den Zusammenhang zwischen der Design Research Methodik, der Design Research sowie des Designs nach BLESSING UND CHAKRABARTI am Beispiel der vorliegenden Arbeit. Die Design Research Methodik beschreibt, wie das Instrumentarium zu entwickeln ist, und bildet somit die Grundlage bzw. den Rahmen der Forschung. Die Design Research entspricht der eigentlichen Entwicklung des Instrumentariums. Das Design entspricht anschließend der Anwendung des Instrumentariums in der Praxis.

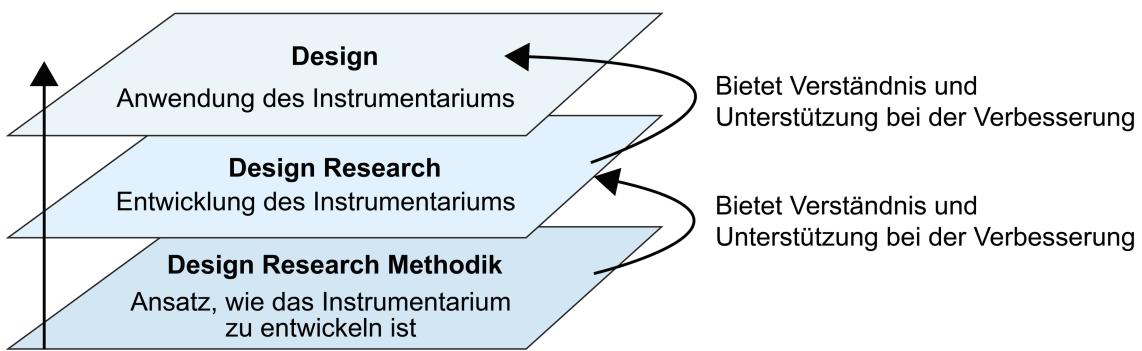


Abbildung 2-3: *Zusammenhang von Design Research Methodik, Design Research sowie dem Design nach [BC09] am Beispiel der vorliegenden Arbeit*

2.2 Design Research Methodology

Mit der Design Research Methodology (DRM) beschreiben BLESSING UND CHAKRABARTI einen **domänenübergreifenden Forschungsdesignansatz** [BC09]. Der Ansatz umfasst die in Abbildung 2-4 veranschaulichten vier wesentlichen Phasen, die grundsätzlich iterativ und bei Bedarf flexibel durchlaufen werden können.

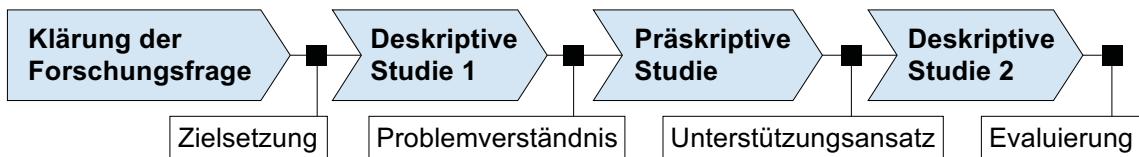


Abbildung 2-4: *Die vier wesentlichen Phasen der Design Research Methodology und ihre Resultate nach [BC09]*

Zunächst erfolgt die **Klärung der Forschungsfrage**. Die Klärung erfolgt meist literaturbasiert. Es gilt, Beweise oder Hinweise zu identifizieren, die das Forschungsziel unterstützen. Darauf basierend werden die aktuelle sowie die gewünschte Situation beschrieben. Resultat der ersten Phase ist entsprechend die Zielsetzung für die Forschung.

Basierend auf der Zielsetzung wird das vorhandene Problem bzw. die Forschungsfrage genauer analysiert. In diesem Zusammenhang wird untersucht, welche Faktoren für die Problemlösung relevant sind und besonderer Berücksichtigung bedürfen. Die Problemanalyse erfolgt im Rahmen einer ersten **deskriptiven Studie**. Dies kann abhängig von der Problemsituation einer empirische Studie, einer Literaturrecherche oder auch weiteren Methoden der Datensammlung wie Beobachtungen oder Interviews entsprechen [BC09]. Ergebnis der zweiten Phase ist ein detailliertes Verständnis des Problems.

In der dritten Phase erfolgt im Rahmen der **präskriptiven Studie** die Entwicklung eines Ansatzes zur Verbesserung des Designs und entsprechend zur Lösung der Problemstellung. Bei der Entwicklung können aufbauend auf Phase 1 und 2 verschiedene Methoden zum Einsatz kommen. Wichtig ist die gezielte Adressierung der zuvor definierten relevanten Faktoren, um eine bedarfsgerechte Lösung zu gewährleisten.

Abschließend wird der Unterstützungsansatz in einer zweiten **deskriptiven Studie** evaluiert. Zum einen wird durch die Evaluierung die Anwendbarkeit des Ansatzes bewertet. Zum anderen ist auch die Nützlichkeit für die Problemstellung entsprechend der zuvor definierten Anforderungen zu evaluieren. Bei der Evaluierung können erneut verschiedene Ansätze der Datensammlung wie Simulationen, Befragungen oder Beobachtungen zum Einsatz kommen. Zum Ende der vierten Phase wurde der Forschungsbeitrag evaluiert und bewertet.

Für die **Vorgehensweise** entsprechend der Design Research Methodology werden sieben Varianten definiert, die unterschiedliche Ausprägungen der einzelnen Phasen beinhalten (Abbildung 2-5). Eine Phase kann beispielsweise ausschließlich *literaturbasiert* und somit durch Analyse von Literatur bearbeitet werden. Eine *umfassende* Bearbeitung ergänzt die Literaturarbeit durch eigene Arbeiten und Entwicklungen des Forschers. Dazu zählt unter anderem auch die Durchführung von Studien. Wird eine Phase nur so weit bearbeitet, bis die relevanten Konsequenzen aufgezeigt werden können, gilt sie als *initial* bearbeitet. Dies ist beispielsweise der Fall, wenn weitere Forschungsaktivitäten durch andere Forscher erfolgen.

2.3 Einordnung und Vorgehen der vorliegenden Arbeit

Die Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung stellen eine komplexe Aufgabe dar. Dies wird in der Problemanalyse in Kapitel 3 ausführlich erläutert. Die Entwicklung eines unterstützenden Instrumentariums erfordert entsprechend einen systematischen Problemlösungsprozess im Sinne des Designs (vgl. Kapitel 2.1). Entsprechend der Design Research erfolgt daher im Rahmen dieser Arbeit die wissenschaftliche Erarbeitung eines Ansatzes zur Unterstützung der Planung und Vorbereitung AR-basierter Validierung. Das adressierte Problem ist bislang kein wesentlicher Teil der Forschung. Es handelt sich somit um ein neues Problem. Es existieren jedoch Ansätze im Stand der Technik, die auf das Problem übertragen werden können. Entsprechend stellt der Stand der Technik die Wissensbasis des Forschungsdesigns dar. Zudem entsprechen

einzelne Bestandteile des Instrumentariums der Definition von GREGOR UND HEVNER folgend *Zweckentfremdungen* (vgl. Abbildung 2-1). Das entwickelte Instrumentarium als Ganzes entspricht aufgrund seines Neuheitsgrads und seiner geringen Reife jedoch einer neuen Lösung. Somit ist das Gesamtergebnis dieser Arbeit als *Erfindung* anzusehen.

		Vorgehen im Forschungsdesign			
		Klärung der Forschungsfrage	Deskriptive Studie 1	Präskriptive Studie	Deskriptive Studie 2
Vorgehensweisen des Forschungsdesigns	1	Literaturbasiert → Umfassend	-	-	-
	2	Literaturbasiert → Umfassend	→ Initial	-	-
	3	Literaturbasiert → Literaturbasiert	→ Umfassend	→ Initial	-
	4	Literaturbasiert → Literaturbasiert	→ Literaturb. / Initial / Umfassend	→ Umfassend	-
	5	Literaturbasiert → Umfassend	→ Umfassend	→ Initial	-
	6	Literaturbasiert → Literaturbasiert	→ Umfassend	→ Umfassend	-
	7	Literaturbasiert → Umfassend	→ Umfassend	→ Umfassend	-

Abbildung 2-5: Vorgehensweisen der Design Research Methodology nach [BC09]

Die Entwicklung des Instrumentariums erfolgte grundsätzlich entsprechend der **Design Research Methodology** nach BLESSING UND CHAKRABARTI [BC09] (s. Kapitel 2.2). Dabei wurden alle vier Phasen durchlaufen, um eine anwendbare und praxistaugliche Lösung für das vorliegende komplexe Problem zu erzielen. Die DRM diente als allgemeiner Rahmen und Struktur für die Forschungsaktivitäten. Die einzelnen Phasen wurden an das konkrete Forschungsvorhaben angepasst. Für die einzelnen Bestandteile des Instrumentariums wurde die DRM jeweils zusätzlich durchlaufen. Dies betrifft die erste deskriptive Studie in Phase 2 sowie die präskriptive Studie in Phase 3. Abbildung 2-6 gibt einen Überblick über die Anwendung der DRM auf die vorliegende Arbeit und die entsprechenden Kapitel, in denen die Phasen jeweils adressiert werden.

Die **Klärung der Forschungsfrage** dieser Arbeit erfolgte durch eine *Literaturrecherche*. Es wurden umfangreiche wissenschaftliche Publikationen und aktuelle Studien herangezogen. Ergänzt wird die Literaturrecherche durch Erkenntnisse aus abgeschlossenen und laufenden Forschungs- und Industrieprojekten im Kontext des betrachteten Forschungsfelds, in denen der Autor dieser Arbeit mitwirkt oder mitgewirkt hat. Das Ergebnis der ersten Phase ist in der Einleitung dieser Arbeit in Kapitel 1 zusammengefasst.

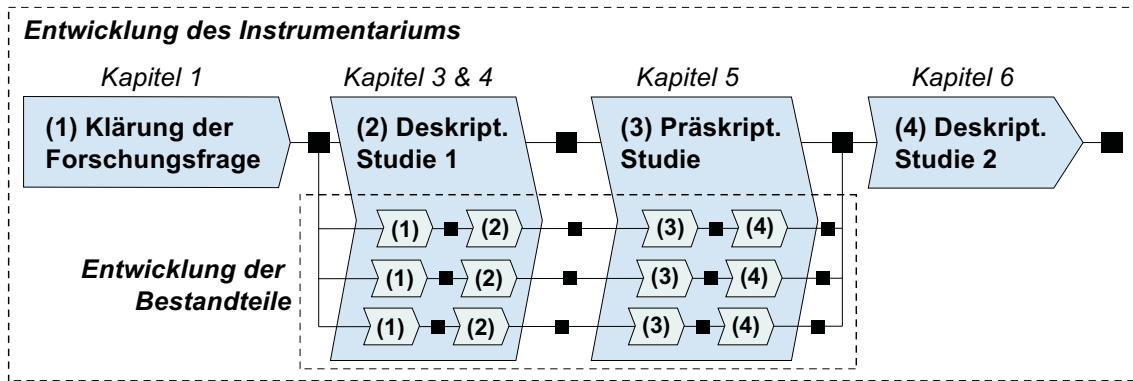


Abbildung 2-6: Anwendung der Design Research Methodology auf die vorliegende Arbeit zur Entwicklung eines Instrumentariums und seiner Bestandteile

In der ersten **deskriptiven Studie** erfolgt zunächst eine ausführliche und detaillierte Problemanalyse. Diese basiert zu großen Teilen ebenfalls auf einer umfangreichen *Literaturrecherche* zu den Schwerpunktthemen dieser Arbeit: Wandel der Produktentwicklung, Validierung und Augmented Reality. Zur Ergänzung und Bestätigung der Literaturrecherche wurde eine *Unternehmensbefragung* durchgeführt. Aus der Befragung konnten Erkenntnisse bzgl. der aktuellen Situation sowie der Anforderungen und Bedarfe zur Validierung und zum Einsatz von Augmented Reality direkt aus der Praxis gewonnen werden. Die Ergebnisse der Befragung werden im Rahmen der Problemanalyse in Kapitel 3 herangezogen. Eine vollständige Übersicht über die Umfrageergebnisse ist zudem Anhang A1 zu entnehmen. Die Problemanalyse belegt den Bedarf an einem Instrumentarium und zeigt drei Handlungsfelder für die Entwicklung eines solchen Instrumentariums auf. Zudem konnten neun Anforderungen an das Instrumentarium definiert werden.

Zur Entwicklung der Lösungen in den drei Handlungsfeldern wurden die vier Phasen der Design Research Methodology zusätzlich durchlaufen. Dies erfolgte in verschiedenen Forschungs- und Industrieprojekten, in denen die jeweiligen Forschungsfragen adressiert wurden. So wurden die Zielsetzungen jeweils individuell geklärt und anschließend das Problemverständnis geschärft. Dazu wurde unter anderem jeweils der Stand der Technik untersucht. Kapitel 4 beschreibt zusammenfassend den für die drei Handlungsfelder untersuchten Stand der Technik. Für die Entwicklung der Lösungen relevante und hilfreiche Ansätze aus den verschiedenen Themenfeldern werden analysiert und anhand der gestellten Anforderungen bewertet. Die Bewertung zeigt, dass keiner der untersuchten Ansätze und auch keine triviale Kombination von Ansätzen alle Anforderungen erfüllt. Somit besteht Handlungsbedarf für die Entwicklung eines Instrumentariums zur Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung.

Die Entwicklung des Instrumentariums und der einzelnen Bestandteile erfolgt anschließend im Rahmen der **präskriptiven Studie**. Einzelne Bestandteile wurden in den verschiedenen Projekten entsprechend der Phasen 3 und 4 der Design Research Methodology entwickelt und im jeweiligen Projektkontext evaluiert. Entsprechende Resultate wurden

bereits wissenschaftlich publiziert (siehe *Liste der veröffentlichten Teilergebnisse* am Anfang dieser Arbeit). Das Zusammenspiel der einzelnen Bestandteile als Instrumentarium zur Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung wird in Kapitel 5 ausführlich beschrieben.

Für die abschließende zweite **deskriptive Studie** werden das Instrumentarium und seine Bestandteile anhand eines konkreten Anwendungsbeispiels aus einem Forschungsverbundprojekt angewendet und erprobt (Kapitel 6). Dabei liegt ein besonderer Fokus auf dem Zusammenspiel der einzelnen Bestandteile zur Planung und Vorbereitung der AR-basierten Validierung. Die Evaluierung wird ergänzt durch Erkenntnisse aus weiteren Forschungs- und Industrieprojekten, in denen die Teilergebnisse dieser Arbeit entwickelt und angewendet wurden.

Das Forschungsdesign der vorliegenden Arbeit entspricht aufgrund der umfassenden Bearbeitung der Studien Ausprägung 7 der von BLESSING UND CHAKRABARTI definierten Vorgehensweisen der Design Research Methodology (vgl. Abbildung 2-5).

3 Problemanalyse

Ziel der Problemanalyse sind Anforderungen an ein *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*. Hierfür werden in Kapitel 3.1 relevante Begriffe definiert und die Ausrichtung der Arbeit beschrieben. In Kapitel 3.2 wird anschließend der Wandel der Produktentwicklung und damit verbundene Veränderungen der Marktleistungen sowie deren Entwicklung analysiert. Kapitel 3.3 beschreibt die Bedeutung von Validierung in der Produktentwicklung und geht auf bestehende Ansätze und Herausforderungen ein. Schwerpunkt von Kapitel 3.4 ist anschließend die Technologie Augmented Reality und damit verbundene Potentiale für die Validierung. In Kapitel 3.5 wird die Problemanalyse zusammengefasst. Darauf aufbauend werden in Kapitel 3.6 Anforderungen an das Instrumentarium abgeleitet und beschrieben.

3.1 Begriffsdefinitionen und Einordnung der Arbeit

Gegenstand dieser Arbeit ist ein *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*. Unter dem Begriff **Instrumentarium** ist eine kohärente Menge an Hilfsmitteln zu verstehen, die eine bestimmte Aufgabe, eine Tätigkeit oder die Erreichung eines Ziels unterstützt. Solche Hilfsmittel können unter anderem Methoden, Richtlinien, Lösungsbausteine sowie Werkzeuge sein. Zudem enthält ein Instrumentarium meist ein Vorgehensmodell, das den Einsatz der Hilfsmittel strukturiert und anleitet.

Die **Produktentwicklung** als Rahmen dieser Arbeit umfasst die interdisziplinäre Konzipierung, den domänen spezifischen Entwurf sowie die Integration der Ergebnisse zu einem Gesamtsystem. Dieses Gesamtsystem entspricht einem Produkt bzw. einer Marktleistung des entwickelnden Unternehmens. Zusammen mit der Produktplanung sowie der Dienstleistungs- und der Produktionssystementwicklung ist die Produktentwicklung Teil der Produktentstehung [GDE+18].

Im Kontext der Produktentwicklung ist **Validierung** eine Aktivität zur Absicherung der Entwicklung. Bei der Validierung wird sichergestellt, dass das Produkt den Kundenanforderungen und -wünschen entspricht und seinen Einsatzzweck erfüllt. Dies entspricht der Beantwortung der Frage, ob das richtige Produkt entwickelt wird [VDI2206:2004].

Mit dem Begriff **Augmented Reality** sind alle Ansätze immersiver Technologien gemeint, bei denen die reale Umgebung durch die Visualisierung digitaler Inhalte erweitert bzw. reale und digitale Inhalte miteinander kombiniert werden. Die Technologie Virtual Reality, bei der lediglich die virtuelle Umgebung visuell wahrgenommen wird, ist daher nicht Teil dieser Arbeit. Eine ausführliche Begriffsdefinition kann dem Leitfaden „Augmented und Virtual Reality“ des BITKOM E.V. entnommen werden, an dem der Autor der vorliegenden Dissertation maßgeblich beteiligt ist [bit21].

Unter dem Begriff **Planung** werden im Rahmen dieser Arbeit die Entscheidungsfindung bzgl. des Einsatzes einer Technologie – konkret der Technologie Augmented Reality – sowie die grobe Gestaltung des Einsatzes verstanden. Darauf aufbauend umfasst die **Vorbereitung** die Konzipierung und die Realisierung eines entsprechenden technischen Systems. Der anschließende Einsatz der Technologie wird von den Begriffen Planung und Vorbereitung nicht abgedeckt.

3.2 Wandel der Produktentwicklung

Die Produktentwicklung umfasst alle Tätigkeiten, die zur Realisierung eines neuen Produkts bzw. einer neuen Marktleistung eines Unternehmens führen. Die Grundlage für die Produktentwicklung bildet das Lastenheft, das Anforderungen an das zu entwickelnde Produkt definiert. Nach GAUSEMEIER ET AL. erfolgen ausgehend vom Lastenheft die Tätigkeiten Produktkonzipierung, domänenspezifischer Entwurf sowie Produktintegration [GDE+18, S. 91]. Zuständig für die Produktentwicklung ist meist die Entwicklungsabteilung eines Unternehmens. Es erfolgt jedoch eine enge Abstimmung mit weiteren Organisationseinheiten, um beispielsweise Aspekte bzgl. der Produktion und des Vertriebs bereits während der Entwicklung zu berücksichtigen. So handelt es sich bei der Produktentwicklung um eine abteilungsübergreifende Aktivität [EM13].

Getrieben durch die aktuellen Trends Digitalisierung, Globalisierung und Nachhaltigkeit verändern sich zum einen die Marktleistungen und zum anderen deren Entwicklungsprozesse. Dieser Wandel der Produktentwicklung wird im Folgenden dargestellt. Getrieben durch die Möglichkeiten der neuartigen Advanced Systems (Abschnitt 3.2.1) lässt sich ein zunehmender Trend der Individualisierung und nutzerzentrierten Entwicklung von Marktleistungen erkennen (Abschnitt 3.2.2). Dieser Trend führt in Kombination mit dem gleichzeitig zunehmenden Zeit- und Kostendruck auf globalen Märkten zu neuen Herausforderungen an die Produktentwicklung (Abschnitt 3.2.3).

3.2.1 Advanced Systems – Die Marktleistung von morgen

Bereits Ende des 20. Jahrhunderts führte die Integration elektronischer Komponenten in zuvor meist rein mechanische Systeme zur Entstehung sogenannter mechatronischer Systeme. Der Begriff **Mechatronik** wurde 1969 in Japan erschaffen und hat sich seitdem in der Industrie etabliert [PGG13]. Mechatronische Systeme umfassen ein physisches Grundsystem, Aktorik und Sensorik sowie eine Informationsverarbeitung (Abbildung 3-1). Mechatronik vereint entsprechend die klassischen Ingenieursdisziplinen Maschinenbau, Elektrotechnik und Informationstechnik [VDI2206:2004]. Über die Sensorik wird die Umgebung wahrgenommen und entsprechende Messwerte an die Informationsverarbeitung weitergegeben. Die Informationsverarbeitung bestimmt notwendige Aktionen als Reaktion auf die Messungen und gibt Signale an die Aktorik weiter. Die Aktorik wirkt auf das Grundsystem ein und sorgt für die gewünschte Beeinflussung relevanter Zu-

standsgrößen. Über eine Mensch-Maschine-Schnittstelle kann zudem der Mensch Einfluss auf das System nehmen. Die Verbindungen zwischen den Systemelementen werden in Stoffflüsse, Energieflüsse und Informationsflüsse unterteilt [PBF+07].

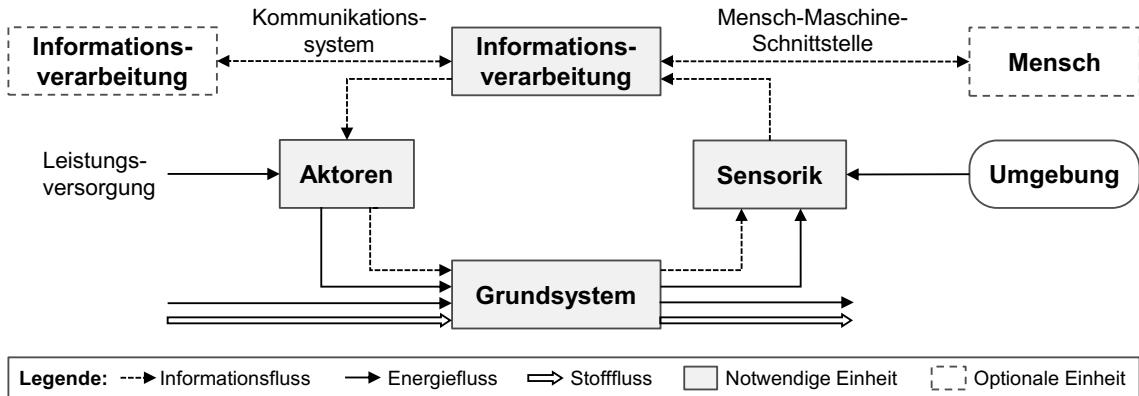


Abbildung 3-1: Grundstruktur mechatronischer Systeme in Anlehnung an [VDI2206:2004]

Nach GAUSEMEIER ET AL. gibt es zwei wesentliche Klassen mechatronischer Systeme [GF06]. Klasse 1 beinhaltet Systeme, bei denen die **räumliche Integration von Mechanik und Elektronik** im Vordergrund steht. Hier liegt der Fokus auf der Miniaturisierung und Funktionsintegration sowie auf Verbindungstechnik. Ziel ist die Realisierung einer hohen Dichte an mechanischen und elektronischen Komponenten auf kleinem Raum. Klasse 2 umfasst **Mehrkörpersysteme mit kontrolliertem Bewegungsverhalten**. Hier liegt der Fokus auf der Regelung des Systems und der Optimierung des Verhaltens. Basiert auf eingehenden Sensordaten analysiert die Informationsverarbeitung das Verhalten und passt dieses bei Bedarf anhand vordefinierter Sollgrößen mittels der Aktorik an.

Die stetige Weiterentwicklung der Informations- und Kommunikationstechnik führte zur Integration intelligenter Funktionalitäten in mechatronische Systeme und zur Herausbildung einer dritten Klasse, den **Intelligenten Technischen Systemen (ITS)**. ITS zeichnen sich gegenüber klassischen mechatronischen Systemen durch eine intelligente Kopplung der Sensorik und Aktorik aus. Durch die Integration von kognitiven Funktionen² sind ITS in der Lage, flexibel auf die Umgebung und den Benutzer zu reagieren und ihr Verhalten anzupassen [Dum10, S. 19]. Auf Basis von Erfahrungswissen entwickeln ITS zudem Strategien zur Bewältigung von Gefahren und zur Reaktion auf zukünftige externe Einflüsse. ITS passen sich dem Benutzer an und zeigen ein stets nachvollziehbares Verhalten. ITS werden entsprechend charakterisiert durch die vier Eigenschaften Adaptivität, Robustheit, Vorausschau und Benutzungsfreundlichkeit [GTD13]. Abbildung 3-2 stellt die

² Kognitive Funktionen umfassen unter anderem das Wahrnehmen und Erkennen der Umgebung, das Enkodieren, Speichern und Erinnern, das Denken und Problemlösen sowie das Steuern von Aktorik [Str96]. Sie ermöglichen somit eine intelligente und flexible Informationsverarbeitung.

Klassen mechatronischer Systeme anhand von Beispielen dar. Wie die Abbildung andeutet, nimmt die Komplexität der Systeme aufgrund der Integration weiterer Funktionalitäten von Klasse zu Klasse zu. Im Kontext technischer Systeme bedeutet eine hohe Komplexität, dass eine hohe Anzahl unterschiedlicher Elemente sowie vielfältige dynamische Beziehungen zwischen den Elementen vorliegen [HWF+12].



Abbildung 3-2: Klassen mechatronischer Systeme in Anlehnung an [Ana15]

Neben der stetigen Weiterentwicklung und Veränderung der physischen Marktleistungen vollzieht sich zudem ein Wandel im Dienstleistungssektor. Die fortschreitende Leistungssteigerung der Informations- und Kommunikationstechnologie sowie die zunehmende Menge an verfügbaren Daten bieten innovative Möglichkeiten zu Gestaltung neuer digitaler Dienstleistungsangebote. So werden vermehrt softwaretechnische Funktionalitäten in die Marktleistungen integriert [GDE+18]. Die zuvor meist getrennt betrachteten Bereiche der Marktleistungen und der Dienstleistungen werden zunehmend miteinander kombiniert und verschmolzen. Es resultieren sogenannte Cyber-Physische-Systeme (CPS). CPS grenzen sich gegenüber ITS vor allem durch die Vernetzung mit anderen Systemen sowie durch die Nutzung weltweit verfügbarer Daten und Dienste ab [Wes17]. Dabei können CPS neben Produkten für Endkunden auch Maschinen, Lagersysteme oder Betriebsmittel umfassen [KWH13, S. 5]. Da die Systeme weit über den aktuellen Stand der Mechatronik hinausgehen, werden sie auch **Advanced Systems** genannt. Für Advanced Systems lassen sich die folgenden Ausprägungen beschreiben [DAG+21]:

- **Autonome Systeme:** Durch die verstärkte Integration kognitiver Funktionalitäten sind Advanced Systems in der Lage, in ihrer Umgebung autonom zu agieren und komplexe Aufgaben eines bestimmten Anwendungsbereichs eigenständig ohne Eingriff des Menschen zu lösen. Dabei lernen sie kontinuierlich im Betrieb und optimieren ihr Verhalten zunehmend.
- **Dynamisch vernetzte Systeme:** Durch eine dynamische Vernetzung sind Advanced Systems in der Lage, im Verbund mit weiteren Systemen zu agieren. So übersteigen

ihre Funktionalität und Leistungsfähigkeit denen einzelner Systeme. Dynamisch vernetzte Systeme verfügen über dynamische Systemgrenzen und können ihre Rollen und Schnittstellen bedarfsgerecht in Abhängigkeit des Gesamtziels variieren.

- **Interaktive soziotechnische Systeme:** Advanced Systems stellen zunehmend eine integrale und zusammenwirkende Einheit zwischen dem technischen System und dem Benutzer dar. Durch eine verbesserte Kognition und Autonomie sind sie in der Lage, sich flexibel den Bedürfnissen der Nutzer anzupassen und eine kontextbasierte Unterstützung zu bieten. Dabei stehen für die Interaktion multimodale Interaktionsmöglichkeiten zur Verfügung (bspw. Sprach- oder Gesten-basiert), mit denen eine hohe Ergonomie erzielt werden kann.
- **Produkt-Service-Systeme:** Advanced Systems ermöglichen zudem eine enge Verzahnung von Sach- und Dienstleistungen. So werden physische Marktleistungen vermehrt mit datenbasierten digitalen Dienstleistungen kombiniert. Die Erfassung und Verarbeitung von Umgebungs-, Nutzer- und Maschinendaten bieten Potential zur Schaffung individueller Mehrwerte und zur Realisierung innovativer Geschäftsmodelle.

Wie die vier beschriebenen Ausprägungen verdeutlichen, zeichnen sich Advanced Systems durch das Zusammenspiel verschiedener Disziplinen aus. Sie vereinen vielfältige Produktmerkmale wie Gestalt, Optik, Funktionalität, Verhalten, Kinematik, Dynamik und Ergonomie in einer Marktleistung. Advanced Systems bieten daher enorme Gestaltungsmöglichkeiten, die zu einer zunehmenden Individualisierung und Nutzerorientierung führen.

3.2.2 Individualisierung und Nutzerorientierung

Unter **Individualisierung** ist grundsätzlich die Änderung oder Gestaltung einer Sache mit dem Ziel, etwas bis zu einem gewissen Grad einzigartiges zu schaffen, zu verstehen [Sch98]. Bei der Individualisierung eines Produkts bzw. einer Marktleistung erfolgt entsprechend die gezielte Gestaltung einer auf individuelle Anforderungen oder Rahmenbedingungen zugeschnittenen Marktleistung. Bereits Mitte des 20. Jahrhunderts erfolgte durch Variantenproduktion eine erste Form der Individualisierung [Uhl03]. In den letzten Jahren hat die Bedeutung der Individualisierung speziell in der Fertigungsindustrie weiter zugenommen [BLF17, S. 369]. Teilweise wird bei Individualisierung und Personalisierung von Produkten sogar von einem neuen Megatrend in der Wirtschaft gesprochen [Rät17-ol].

Es gibt verschiedene **Gründe für den Trend der Individualisierung**. Zum einen herrscht eine verstärkte Volatilität und damit einhergehend ein steigender Wettbewerb auf globalen Märkten. Um sich von Wettbewerbern abzuheben, schneiden Unternehmen ihre Marktleistungen daher zunehmend auf ihre Kunden zu [DAG+21]. Zudem erfordern

die Globalisierung sowie ausgeprägte Nischenmärkte die gezielte Gestaltung der Markt-leistungen mit Hinblick auf individuelle Anforderungen, Normen, Richtlinien und Ge-setze [GDE+18, S. 382, LB06]. Zum anderen ergeben sich im Kontext der Industrie 4.0 neue Möglichkeiten in der Entwicklung sowie der Produktion der Marktleistungen. Pro-duktentwicklungsprozesse werden zunehmend vernetzt und durch digitale Werkzeuge un-terstützt [GDE+18, S. 382]. Durch den Einsatz dieser digitalen Werkzeuge wurden Pro-dukte bereits Anfang des 21. Jahrhunderts häufig individuell für Kunden entwickelt [AR11, S. 15, Vog13]. Durch die fortschreitende Digitalisierung verstärkt sich dieser Trend zunehmend, bis hin zur Produktion kundenindividueller Produkte als Einzelstücke (Losgröße 1) [EUJ18, S. 64]. Dies ist möglich, da durch immer neue und bessere Ferti-gungsverfahren und -technologien heutzutage auch Kleinstmengen und sogar Einzelstü-cke rentabel produziert werden können [KWH13, S. 5]. Eine dieser Fertigungstechnolo-gien, die besondere Potentiale für die Individualisierung bietet, ist beispielsweise Addi-tive Fertigung [JD17].

Im Fokus einer Produktindividualisierung stehen stets die Kundenanforderungen und -wünsche [EUJ18, S. 65, Bog19]. Diese werden typischerweise im **Lastenheft** definiert. Das Lastenheft beinhaltet nach der DIN 69901-5 [Deu09] die Anforderungsspezifikation des Auftraggebers oder des Kunden sowie alle Forderungen zu den gewünschten Leis-tungen. Dabei kann das Lastenheft verschiedene Betrachtungsebenen enthalten [ERZ14]:

- **Geschäftsebene:** Diese Ebene enthält geschäftliche Ziele, die bspw. die Eignung für gewisse Geschäftsmodelle betreffen.
- **Systemebene:** Systemanforderungen beschreiben Anforderungen an die Funktionalität sowie Merkmale des Systems. Dies betrifft unter anderem den konkreten Ein-satzzweck der zu entwickelnden Marktleistung.
- **Komponentenebene:** Diese Ebene umfasst Anforderungen an einzelne Komponen-ten der Marktleistungen, bspw. bzgl. der technischen Ausführung oder Materialei-genschaften.

Für eine bessere Übersicht und Verständlichkeit des Lastenhefts existieren verschiedene **Standards und Referenzstrukturen**. Ein Beispiel ist die Referenzstruktur nach dem VDI/VDE-Standard 3694 „Lastenheft/Pflichtenheft für den Einsatz von Automatisie- rungssystemen“ [VDI/VDE3694:2014]. Ein entsprechendes Lastenheft startet mit Be-schreibungen des Projekts bzw. des zu entwickelnden Produkts sowie des Ist- und des Sollzustands. Anschließend werden die Anforderungen dokumentiert. Sie werden unter-gliedert entsprechend verschiedener Produkt- und Projektaspekte. Es gibt Anforderungen an die Systemtechnik, für Systementwicklung, -inbetriebnahme und -einsatz, sowie An-forderungen an die Qualität und an die Projektabwicklung. Ähnliche Unterteilungen sind auch in anderen Referenzstrukturen vorgesehen. Der IEEE-Standard 1233 zur Erarbei-tung einer Anforderungsspezifikation an ein System [Ins98] sieht eine Unterteilung unter anderem in physische, sicherheitsrelevante oder auch mit dem Informationsmanagement

verbundene Aspekte vor. So existieren für verschiedene Produkttypen und Branchen unterschiedliche Leitfäden und Vorgaben. Die Anforderungen an sich werden meist in natürlicher Sprache formuliert. Zur Formulierung können **Anforderungsschablonen** verwendet werden, die die Struktur der Sätze vorgeben und die Wiederverwendung von Wissen unterstützen [ERZ14, S. 62]. Je nach Produkt, Produkttyp, Branche, Kunde und weiteren Aspekten kann ein Lastenheft entsprechend Anforderungen in beliebigen Formulierungen mit individuellen Inhalten umfassen.

Im Rahmen der Produktentwicklung gilt es, die Eigenschaften der Marktleistung entsprechend der Kundenanforderungen auszuprägen und die Anforderungen somit zu erfüllen [RP09]. Die Berücksichtigung von individuellen Anforderungen wird durch Industrie 4.0 und damit einhergehende Entwicklungs- und Produktionstechnologien zu verschiedenen Zeitpunkten im Wertschöpfungsprozess ermöglicht. Individuelle Anpassungen können im Design, bei der Bestellung, der Planung oder der Produktion sowie im Betrieb erfolgen. Dabei können auch kurzfristige Änderungswünsche berücksichtigt werden [KWH13, S. 19]. Es wird unterschieden zwischen verschiedenen **Kundenauftragsentkopplungspunkten** (Customer Order Decoupling Points, CODP) [BLF17, S. 370]. CODP beschreiben den Zeitpunkt in der Wertschöpfungskette, an dem die Marktleistung an einen speziellen Kunden oder eine Kundengruppe geknüpft und ab diesem Zeitpunkt entsprechend individualisiert wird. Alle Aktivitäten vor dem CODP finden vorausschauend statt. Ab dem CODP erfolgen die Aktivitäten Kundenauftrags-getrieben. Der Kunde stellt Informationen bereit, die als Grundlage für die kundenspezifische Fertigstellung der Marktleistung dienen [Hil97]. Umso früher der CODP, umso mehr Einfluss kann der Kunde entsprechend auf die individuelle Gestaltung der Marktleistung zur Erfüllung seiner Anforderungen nehmen [Hoo15]. Eine gängige Unterscheidung beinhaltet die in Abbildung 3-3 dargestellten vier CODP verortet an den chronologischen Wertschöpfungsphasen *Entwicklung, Herstellung, Montage* und *Auslieferung*:

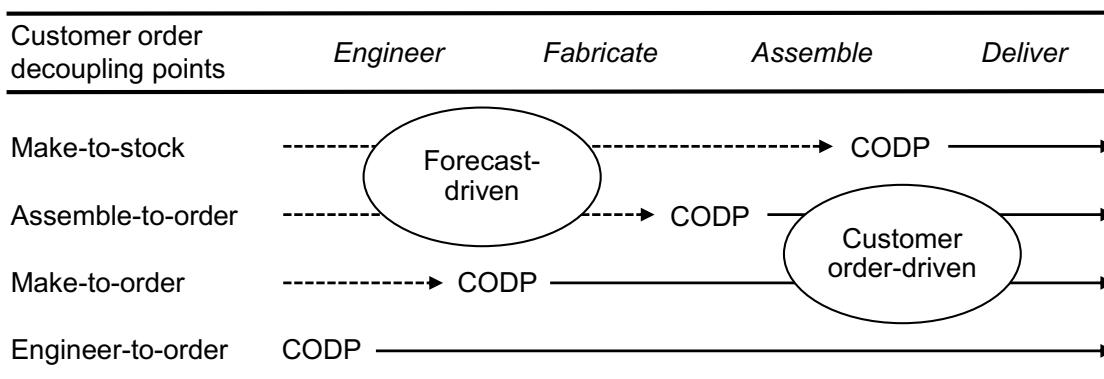


Abbildung 3-3: *Kundenauftragsentkopplungspunkte (CODP) in Anlehnung an [Jan12, S. 38]*

- **Engineer-to-order (ETO):** Bei ETO hat der Kunde die größte Möglichkeit der Einflussnahme auf die Marktleistung und kann bereits während der Entwicklungsphase

mitgestalten. Ausgangspunkt sind Kundenanforderungen und -wünsche, die von Beginn an als Grundlage für die Entwicklung und anschließend für die Fertigung, Montage und Auslieferung eines individuellen Produkts verwendet werden [BLF17, S. 372].

- **Make-to-order (MTO):** Wird das Produkt zunächst vorausschauend entwickelt und direkt im Anschluss individualisiert, handelt es sich um MTO. Hier kann der Kunde die Fertigung des Produkts oder einzelner Komponenten beeinflussen. So können beispielsweise Herstellungsparameter individuell konfiguriert oder das Material ausgewählt werden. Als Synonym für MTO wird teilweise Build-to-order (BTO) verwendet [BLF17, S. 372].
- **Assemble-to-order (ATO):** Bei ATO wurden die Komponenten der Marktleistung bereits produziert bzw. realisiert. Sie werden anschließend entsprechend der Kundenanforderungen individuell zu einem Produkt kombiniert. Dabei kann auch ein Basisprodukt als Grundlage verwendet und mit individuellen Ausprägungen von Zusatzkomponenten individualisiert werden [BLF17, S. 373].
- **Make-to-stock (MTS):** MTS gibt dem Kunden die geringsten Möglichkeiten der Einflussnahme. Die Marktleistung wurde realisiert und ist bereit zur Auslieferung. Der Kunde hat lediglich noch die Möglichkeit der Auswahl von Standard-Komponenten, die ohne individuelle Anpassungen verwendet werden können.

Grundsätzlich kann der Kunde sowohl ein Endnutzer oder eine Gruppe von Endnutzern in einem Business-to-Customer-Markt (B2C) oder ein Unternehmen in einem Business-to-Business-Markt (B2B) sein [BLF17, S. 370]. Speziell im **B2C-Bereich** ist die Veränderung des Marktes hin zum Käufermarkt ein weiterer Treiber des Individualisierungstrends. Der Überhang an Angeboten gegenüber der Nachfrage führt zu einer gestärkten Position der Kunden [Bog19]. Durch vielfältige neue Kanäle können Kunden heutzutage qualifiziertere Kaufentscheidungen treffen und aus einer Vielzahl an Produkten wählen. Kunden haben somit verstärkte Macht gegenüber den produzierenden Unternehmen [BTR19]. Dabei vollzieht sich – unter anderem aus gesellschaftlichen Gründen – ein Trend der Konsumgewohnheiten hin zu individuellen Produkten [JS18]. Kunden erwarten zunehmend auf sie zugeschnittene Produkte, sodass Individualisierung in Zukunft sehr wichtig für das Produktangebot sein wird [KPM17]. Dies zeigt sich bereits an individualisierten Getränken, Verpackungen, Trinkflaschen oder auch Turnschuhen [Cap19]. Zudem wünschen sich Kunden speziell bei Marktleistungen, die für einen langfristigen Gebrauch vorgesehen sind, eine besondere Erfüllung der individuellen Anforderungen und Bedürfnisse [EUJ18, S. 66]. Ein plakatives Beispiel ist die Anschaffung eines Neuwagens. Angefangen von der Lackierung über die Innenausstattung inklusive optischer Details bis hin zu technischen Funktionalitäten und Services hat der Kunde vielfältige Individualisierungsmöglichkeiten [Rät17-ol].

Um sich erfolgreich und nachhaltig am Markt zu positionieren und gegenüber den Wettbewerbern Vorteile zu erzielen, müssen produzierende Unternehmen stärker auf Kundenanforderungen eingehen und sich mit individuellen Bedürfnissen der Kunden befassen [Bog19, S. 6, Rät17-ol, BTR19]. Im **B2B-Bereich** geht es bei der Individualisierung vor allem darum, die Marktleistung so zu gestalten, dass sie in die Wertschöpfungskette und die Infrastruktur des anderen Unternehmens integriert werden kann. So findet Individualisierung im System- und Produktgeschäft und vor allem im Projekt- im Zuliefergeschäft statt [EUJ18, S. 67]. Eine besondere Zunahme der Individualisierung wird unter anderem auch im Maschinen- und Anlagenbau erwartet [DAG+21]. Ein Beispiel aus der Zulieferbranche ist die Fertigung von Getriebemotoren [EUJ18, S. 71]. Abhängig von den gewünschten technischen Spezifikationen und auch des zur Verfügung stehenden Raums können durch Einsatz verschiedener Motoren und Getriebeausprägungen spezifische Drehzahl-Leistungs-Kombinationen erzielt und die Marktleistung so individualisiert werden. Grundsätzlich ist davon auszugehen, dass Individualisierung im B2B-Bereich wesentlich stärker ausgeprägt ist als im B2C-Bereich [EUJ18, S. 68].

Die in Kapitel 3.2.1 beschriebenen **Advanced Systems** vereinen vielfältige Produkteigenschaften in einer Marktleistung. Diese Produkteigenschaften bieten jeweils für sich oder in Kombination enorme Individualisierungspotentiale. Das gilt im Speziellen für integrierte Softwarefunktionalitäten. Ohne physische Komponenten zu verändern oder auszutauschen, können Marktleistungen durch Anpassungen der Software auf Kunden, Nutzer und spezifische Rahmenbedingungen zugeschnitten werden. Des Weiteren ermöglicht das Zusammenspiel unterschiedlicher Systemkomponenten den Austausch einzelner Elemente (bspw. der Sensorik, Aktorik oder Interaktion) und somit die einfache Anpassung der Marktleistung an individuelle Anforderungen und Wünsche des Kunden. Im Kontext der Advanced Systems wird der Trend der Individualisierung aus der Perspektive der Kunden und Nutzer daher weiter zunehmen [DAG+21].

In den letzten Jahren hat sich zudem gezeigt, dass die Digitalisierung der Marktleistungen sowie im speziellen die Kombination von physischen Produkten und digitalen Dienstleistungen einen zusätzlichen Kundennutzen generieren kann. Bei der Entwicklung von Advanced Systems wird der Nutzer daher in den Fokus gestellt und die Ausprägungen der vielfältigen Produktmerkmale individuell auf den jeweiligen Kunden zugeschnitten [DAG+21]. So soll sichergestellt werden, dass die entwickelten Marktleistungen bestehend aus Sachleistungen und Dienstleistungen dem Anbieter, Kunden und Anwender einen konkreten Nutzen stiften. Ein wichtiger Aspekt dieser **Nutzerorientierung** ist Ergonomie. Der Nutzer muss intuitiv und möglichst komfortabel mit der Marktleistung interagieren können. Zudem muss sich das System intelligent dem Nutzer anpassen können, um stets eine hohe Benutzerfreundlichkeit zu gewährleisten [DAG+21].

Laut einer Studie der TU München sind unter anderem steigende und sich ändernde Anforderungen ein Charakteristikum von Produktkomplexität. Daher nimmt die Produkt-

komplexität durch die Individualisierung und Nutzerorientierung in der Produktentwicklung weiter zu [BHL07-ol]. Dies ist einer der Gründe für neue Herausforderungen, die sich im Kontext der Entwicklung heutiger und zukünftiger Marktleistungen ergeben.

3.2.3 Herausforderungen der heutigen Produktentwicklung

Aus Sicht des Herstellers geht es bei der Produktentwicklung darum, ein Produkt bzw. eine Marktleistung zu entwickeln, die profitabel produziert und vermarktet werden kann. Die Profitabilität steht dabei im Zusammenhang mit fünf Charakteristika, die eine erfolgreiche Entwicklung ausmachen [UE12, S. 2f.]:

- **Produktqualität:** Die Produktqualität beschreibt, wie gut das Produkt ist, ob es beispielsweise robust und zuverlässig ist. Zudem umfasst die Produktqualität die Erfüllung der Kundenanforderungen und -bedarfe, was speziell im Kontext der Individualisierung von hoher Relevanz ist. Eine Nichterfüllung bedeutet entsprechend eine geringe Produktqualität. Letztendlich hängt der Preis der Marktleistung, den Kunden bereit sind, zu zahlen, stark von der Produktqualität ab.
- **Produktkosten:** Die Produktkosten umfassen alle Kosten, die für die Herstellung bzw. Realisierung der Marktleistung anfallen. Dazu zählen Ausgaben für Investitionsgüter sowie Werkzeugkosten und Betriebskosten von Maschinen. In Kombination mit dem Verkaufspreis beeinflussen die Produktkosten den Profit, den der Anbieter mit der Marktleistung machen kann. Für einen hohen Profit müssen die Produktkosten entsprechend möglichst gering gehalten werden.
- **Entwicklungszeit:** Die Entwicklungszeit beschreibt die Dauer, die der Anbieter für die Entwicklung der Marktleistung benötigt. Die Entwicklungszeit beeinflusst unter anderem, wie schnell der Anbieter auf Veränderungen des Wettbewerbs und technologische Entwicklungen reagieren kann. Zudem ermöglicht eine kürzere Entwicklungszeit eine kürzere Time-to-Market und entsprechend schnellere wirtschaftliche Erträge aus der entwickelten Marktleistung.
- **Entwicklungskosten:** Die Entwicklungskosten beinhalten alle Ausgaben, die für die Entwicklung notwendig sind. Darunter fallen unter anderem Personalkosten, Ausgaben für Werkzeuge und Labore sowie Studien und Recherchen und auch Materialkosten, beispielsweise für Prototypen. Die Entwicklungskosten machen in der Regel einen großen Teil der Ausgaben aus, die für die Erzielung von Gewinnen durch eine Marktleistung erforderlich sind.
- **Entwicklungsfähigkeit:** Entwicklungsfähigkeit meint die Fähigkeit eines Unternehmens, sich an neue Rahmenbedingungen und sich verändernde Möglichkeiten sowie Anforderungen anzupassen und die Produktentwicklung somit zu verbessern. Entwicklungsfähigkeit beinhaltet, ob Entwicklerteams in der Lage sind, ihre Erfahrungen und zudem neue Technologien so einzusetzen, dass Marktleistungen immer effektiver und wirtschaftlicher entwickelt werden können.

Der beschriebene Wandel der Produktentwicklung führt zu verschiedenen, die fünf Charakteristika direkt oder indirekt betreffenden Herausforderungen. Eine grundsätzliche Herausforderung ist die stetig steigende **Komplexität heutiger Marktleistungen**. Diese resultiert unter anderem aus dem Einfluss der mit der Individualisierung einhergehenden steigenden und sich ändernden Anforderungen [BHL07-ol]. Zudem steigert die zunehmende Interdisziplinarität der Systeme die Komplexität [Ing20-ol]. Funktionalitäten und Eigenschaften der Marktleistungen werden zukünftig verstärkt durch softwaretechnische und elektronische Lösungen realisiert. Ein damit eng verbundener Aspekt ist die Komposition von Marktleistungen aus einzelnen Komponenten. Diese Komponenten werden zum Teil losgelöst entwickelt oder von Zulieferern bezogen. Hinzu kommt die Vernetzung mit weiteren Systemen im Rahmen von sogenannten „System of Systems“ [GDE+18]. Diese führt zu einer Zunahme der Schnittstellenvielfalt und zur Erweiterung der Systemgrenzen. So erfordern Produkt-Service-Systeme als eine Ausprägung von Advanced Systems die integrierte Betrachtung von Sach- und Dienstleistungen, was viele Unternehmen vor Herausforderung bei der Entwicklung stellt [DAG+21]. Grundsätzlich stellt die Beherrschung der Produktkomplexität bereits seit Jahren ein Problem in Unternehmen dar [LB06].

Im Rahmen der Individualisierung werden Marktleistungen gezielt für individuelle Kunden oder Einsatzszenarien entwickelt oder ausgeprägt. Zudem müssen Marktleistungen kontinuierlich an sich ändernde Bedürfnisse und Anforderungen der Kunden angepasst werden. Neben der steigenden Produktkomplexität sorgt die Individualisierung somit für eine zunehmende **Variantenvielfalt** [KP18-ol, GDE+18, S. 382]. Die Vielfalt an Varianten und individuellen Kundenanforderungen stellt Unternehmen vor weitere Herausforderungen bei der Entwicklung der Marktleistungen [BLF17, S. 369]. Zudem werden die maßgeschneiderten Marktleistungen meist als Sonderanfertigungen in kleinen Stückzahlen bis hin zu Losgröße 1 produziert [KP18-ol]. Abbildung 3-4 verdeutlicht diese Entwicklung anhand der Evolution der Fertigungsparadigmen. Seit Mitte des 20. Jahrhunderts nimmt die Produktvielfalt zu, während das Fertigungsvolumen pro Produkt sinkt.

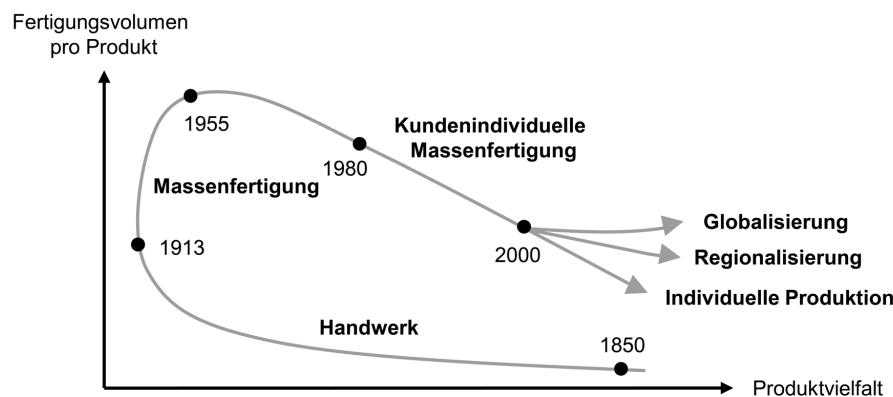


Abbildung 3-4: Die Evolution der Fertigungsparadigmen [Bog19, S. 7]

Gleichzeitig führt der technologische Fortschritt zu einer stetigen **Verkürzung der Produktlebenszyklen** [LB06, GDE+18]. So werden Marktleistungen immer schneller durch neue Generationen mit verbesserter Leistungsfähigkeit oder günstigeren Preisen ersetzt.

Aufgrund der Komplexität und Schnittstellenvielfalt heutiger Marktleistungen ist eine **interdisziplinäre, Abteilungs- und teilweise sogar Unternehmens-übergreifende Zusammenarbeit** in der Entwicklung ein Erfolgskriterium [VDI2206:2004]. Die Systeme müssen von vorneherein ganzheitlich betrachtet werden. Dabei sind auch die nach der Produktentwicklung stattfindenden Phasen des Produktlebenszyklus bereits zu berücksichtigen. Neben den direkt an der Entwicklung beteiligten Disziplinen sind daher unter anderem Fachleute aus der Fertigung, Montage, Logistik sowie dem Service einzubeziehen [ERZ14, S. 9]. So kann das Produkt für die anschließenden Phasen optimiert und frühzeitig Einfluss auf das Kriterium *Produktkosten* genommen werden. Für eine erfolgreiche Zusammenarbeit sind eine effiziente Kommunikation und Kooperation der Fachleute aus den verschiedenen Disziplinen notwendig. Es wird auch vom verteilten Entwickeln gesprochen. Nach Aussage einiger Unternehmen wird die Zusammenarbeit jedoch erschwert, da Mitarbeiter sowohl ihre Arbeitsorte als auch die Arbeitszeiten zunehmend flexibel gestalten wollen [DAG+21].

Im Kontext der Individualisierung müssen für eine erfolgreiche Entwicklung zudem das Kundenproblem und die individuellen Kundenanforderungen und -bedarfe verstanden werden. Viele Entwicklungen scheitern bzw. sind nicht erfolgreich, da die Kundenanforderungen nicht erfüllt werden und auf individuelle Wünsche nicht ausreichend eingegangen wird. Dies betrifft vor allem digitale Dienstleistungen [CFJ20]. Mangelnde Erfüllung der Kundenanforderungen wirkt sich direkt auf die Bewertung des für den Erfolg relevanten Kriteriums der *Produktqualität* aus. Um Marktleistungen erfolgreich zu entwickeln, ist daher eine intensive **Kommunikation mit den Kunden** notwendig [GDE+18, S. 99]. Dies gilt spätestens ab dem Kundenauftragsentkopplungspunkt (CODP, siehe Abschnitt 3.2.2), ab dem eine Marktleistung individualisiert wird. Speziell im Kontext der heutigen Advanced Systems erfordert die menschzentrierte Gestaltung die gezielte Berücksichtigung der Kunden und der Nutzer der Systeme [DAG+21].

So führt die kundenwahrnehmbare Individualisierung in Kombination mit der Komplexitätssteigerung der Marktleistungen sowie der Veränderung der globalen Märkte zu neuen **Herausforderungen in der Entwicklung** heutiger und zukünftiger Marktleistungen [DAG+21]. Individuelle Kundenanforderungen und -bedarfe müssen erfüllt und somit eine hohe *Produktqualität* erzielt werden. Dabei gilt es, die *Produkt- und Entwicklungskosten* trotz geringer Stückzahlen möglichst gering zu halten. Gleichzeitig führen die zunehmenden kürzeren Produktlebenszyklen zu ebenfalls kürzeren *Entwicklungszeiten*. Um am Markt erfolgreich zu sein, müssen die Marktleistungen daher möglichst schnell entwickelt, gefertigt und ausgeliefert werden [PL11, S. 13]. Neben der grundsätzlichen Fähigkeit, individuelle Marktleistungen zu entwickeln, sind für eine erfolgreiche Entwicklung zudem eine hohe Flexibilität und Effektivität ausschlaggebend [WK14, S. 203]. Im

Kontext der *Entwicklungsfähigkeit* sehen Unternehmen eine Veränderung der Unternehmenskultur sowie den Einsatz innovativer Technologien als wichtig an [DAG+21].

Letztendlich müssen Unternehmen ihre Entwicklungszyklen weiter verkürzen und Produkte bei gleicher oder gesteigerter Produktqualität schneller an den Markt bringen [DAG+21]. Ausschlaggebend für die Produktqualität ist die Erfüllung der Kundenanforderungen und -bedarfe. Um den Herausforderungen zu begegnen, müssen Marktleistungen in kurzen Entwicklungszyklen iterativ entwickelt und Teilespekte Schritt für Schritt erprobt und abgesichert werden. Dabei ist die Integration der Kunden und Anwender von enormer Bedeutung. Nur durch effiziente Validierungsaktivitäten mit dem Kunden kann flexibel auf sich ändernde Kunden- und Nutzeranforderungen sowie Rahmenbedingungen reagiert und somit spätere Anpassungen vermieden werden. Zudem kann nur durch eine **frühe und kontinuierliche Validierung** sichergestellt werden, dass die spätere Marktleistung die an sie gestellten Anforderungen erfüllt und entsprechend eine hohe Produktqualität erzielt wird [DAG+21].

3.3 Validierung

Im Duden ist das Verb *validieren* definiert als das Bestimmen der Gültigkeit von etwas. In der DIN EN ISO9000 zu Qualitätsmanagementsystemen wird Validierung definiert als:

„Bestätigung durch Bereitstellung eines objektiven Nachweises, dass die Anforderungen für einen spezifischen beabsichtigten Gebrauch oder eine spezifische beabsichtigte Anwendung erfüllt worden sind“
[ISO9000:2005]

Übertragen auf den Kontext technischer Systeme bedeutet Validierung die Überprüfung, ob ein Produkt für den vorgesehenen Einsatzzweck geeignet ist und einen gewünschten Wert erzielt [VDI2206:2004]. Validierung ist entsprechend von enormer Bedeutung für den Erfolg eines Produkts. Eine genaue Einordnung von Validierung in der Produktentwicklung wird im folgenden Abschnitt 3.3.1 vorgenommen. Im Anschluss werden in Abschnitt 3.3.2 bestehende Ansätze und Werkzeuge zur Validierung analysiert. Im Kontext individueller Marktleistungen wird der Einsatzzweck einer Marktleistung sehr stark durch Kunden und Nutzer beeinflusst. Entsprechend ist es wichtig, Kunden und Nutzer in die Validierung zu integrieren. Dieser Aspekt wird in Abschnitt 3.3.3 untersucht.

3.3.1 Validierung im Kontext der Produktentwicklung

Ziel der Produktentwicklung sind qualitativ hochwertige Marktleistungen, die mit einem möglichst hohen Profit vermarktet werden können. Entscheidend für eine erfolgreiche Produktentwicklung ist daher wie bereits in Abschnitt 3.2.3 beschrieben unter anderem eine hohe **Produktqualität**. Die Bewertung der Produktqualität kann entsprechend verschiedener Faktoren erfolgen. Ein Ansatz ist die Bewertung der Produktqualität anhand

der Präsenz von geforderten Produkteigenschaften, deren Ausprägung objektiv festgestellt werden kann. Umso stärker eine Eigenschaft ausgeprägt ist, umso höher ist die entsprechende Produktqualität. GARVIN spricht dabei vom produktbezogenen Ansatz der Qualitätsbewertung [Gar84]. Ein weiterer speziell im Kontext der Individualisierung und Nutzerorientierung relevanter Ansatz ist die kundenbezogene Bewertung der Produktqualität. Hier stehen die subjektiven Präferenzen sowie Emotionen der Kunden und Nutzer im Fokus. Der Grad der Produktqualität ist entsprechend stark abhängig vom Erfüllungsgrad der Kundenanforderungen und beeinflusst direkt die Kundenzufriedenheit [BSH06]. Weitere für die Produktqualität relevante Faktoren sind die Kosten, unter anderem in Relation zu den Produkteigenschaften, sowie fertigungsbezogene Aspekte.

Zur Erreichung einer hohen Produktqualität ist genaues Wissen über die gestellten Anforderungen und Zielformulierungen sowie zu berücksichtigende Einflussfaktoren notwendig. Speziell im Kontext komplexer Marktleistungen und damit verbundener komplexer Entwicklungsprozesse gibt es jedoch gewisse **Unsicherheiten** im Laufe der Entwicklung. Unsicherheit bedeutet eine Diskrepanz zwischen dem für die Entwicklung benötigten und dem vorhandenen Wissen [WEM+13]. Produktentwicklungsprozesse sind daher nicht von vorneherein planbar und es kommt zu häufigen Änderungen [Ebe15]. Laut der sogenannten Zehnerregel steigen die Änderungskosten mit dem Faktor Zehn je Entwicklungsphase [RLH96]. Neben einer hohen Produktqualität sind die Entwicklungs- und die Produktkosten jedoch ebenfalls von enormer Bedeutung für den Erfolg der Entwicklung (vgl. Abschnitt 3.2.3). Die Kosten müssen daher möglichst gering gehalten werden. Die Möglichkeiten der Kostenbeeinflussung nehmen im Laufe des Produktlebenszyklus kontinuierlich ab [ERZ14, S. 18]. Zudem haben Entscheidungen, die zu Beginn der Entwicklung getroffen werden, Einfluss auf alle weiteren Phasen. Da in der frühen Konzeptionsphase gleichzeitig der größte Gestaltungsspielraum besteht, ist es hier besonders wichtig, den Anforderungen entsprechend gezielt zu entwickeln und somit spätere Änderungen zu vermeiden [ERZ14]. Durch die Vermeidung von Änderungen können zudem die Entwicklungszeiten verkürzt werden, was ebenfalls wichtig ist, um wettbewerbsfähig zu sein. Grundsätzlich streben Unternehmen daher im gesamten Verlauf der Entwicklung eine möglichst frühe **Erkennung von notwendigen Änderungen** an.

Dies spiegelt sich in gängigen Vorgehensmodellen der Produktentwicklung wider. Anstatt eine chronologische Abfolge von Aktivitäten zu durchlaufen, handelt es sich bei der Produktentwicklung meist um ein „Wechselspiel von Aufgaben“ [GDE+18, S. 89]. In einer iterativen Abfolge werden die Aufgaben so lange bearbeitet und dabei das Produkt schrittweise weiterentwickelt, bis eine gewünschte Entwicklungsreife bzw. Produktqualität gegeben ist. So entwickelt sich das Produkt beispielsweise von einem Labormuster über ein Funktionsmuster und ein Vorserienprodukt bis hin zum Serienprodukt. Dabei erfolgt eine regelmäßige **Eigenschaftsabsicherung** zur Überprüfung der Entwicklung in den unterschiedlichen Reifegraden entsprechend der jeweiligen Entwicklungsphasen. Die Eigenschaften von Lösungsvarianten und -konzepten werden fortlaufend mit den Anforderungen abgeglichen und bewertet [VDI2206:2004]. Diese regelmäßige Überprüfung

ermöglicht die Reduzierung der bestehenden Unsicherheiten und die frühzeitige Erkennung und Durchführung notwendiger Änderungen. Zudem kann flexibel auf sich ändernde Anforderungen reagiert und neue Erkenntnisse in anschließenden Entwicklungsiterationen berücksichtigt werden [DAG+21].

Teil der Eigenschaftsabsicherung sind typischerweise die **Validierung und die Verifikation**, die jeweils unterschiedliche Aspekte des Produkts adressieren. Bei der Verifikation geht es vor allem um die formale Absicherung von Spezifikationen. Dies entspricht der Frage, ob die Entwicklung korrekt durchgeführt wird [ERZ14, S. 68]. Validierung dagegen fokussiert das Produkt an sich und beantwortet die Frage, ob das richtige Produkt entwickelt wird [VDI2206:2004]. Dabei ist vor allem die Sicht der Kunden und Nutzer zu berücksichtigen [ABK+16]. Validierung adressiert somit stark die kundenbezogene Bewertung der Produktqualität.

Zur Veranschaulichung der **Validierung im Kontext der Produktentwicklung** wird das erweiterte ZHO-Modell nach ALBERS ET AL. verwendet (Abbildung 3-5). Die drei wesentlichen Bestandteile des ZHO-Modells sind das Zielsystem (Z), das Handlungssystem (H) und das Objektsystem (O) [ABK+16]. Das Zielsystem umfasst die geplanten Eigenschaften und die Anforderungen an das zu entwickelnde Produkt. Entsprechend sind auch die Anforderungen, Bedarfe und Wünsche der Kunden Teil des Zielsystems. Das Objektsystem entspricht den Ergebnissen der Produktentwicklung. Während der Entwicklung enthält es Teillösungen in verschiedenen Reifegraden und zum Ende der Entwicklung das finale Produkt. Beim Handlungssystem handelt es sich um ein sozio-technisches System, das zur Erstellung des Ziel- und des Objektsystems dient. Es beinhaltet für die Entwicklung benötigte Ressourcen, Werkzeuge und Prozesse. Neben den Entwicklern sind auch die Nutzer und Käufer bzw. Kunden der zu entwickelnden Marktleistung als wichtige Stakeholder im Handlungssystem involviert. Im Rahmen des Handlungssystems finden das Design und die Validierung statt (vgl. Abbildung 3-5).

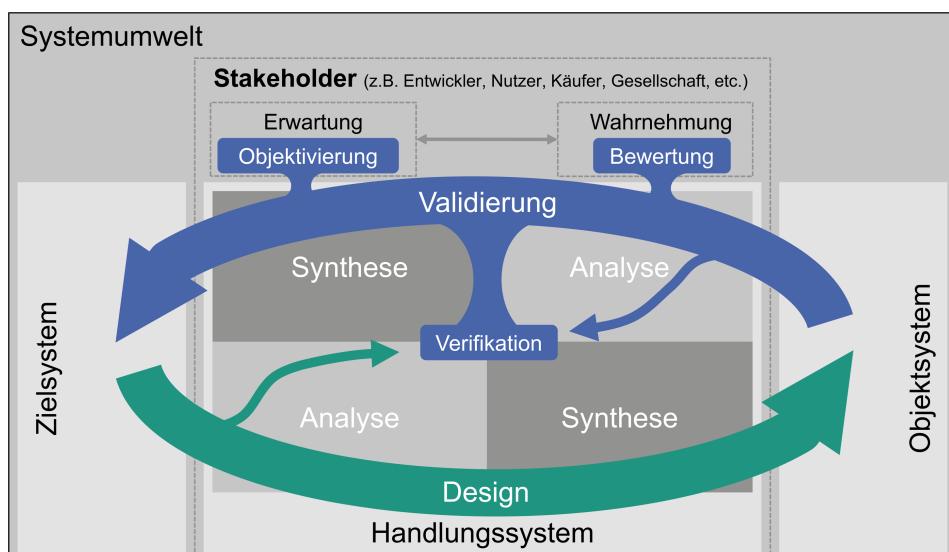


Abbildung 3-5: Erweitertes ZHO-Modell der KaSPro – Karlsruher Schule der Produktentwicklung inkl. der Design- und Validierungsaktivitäten [ABK+16]

Beim Design erfolgt eine Analyse des Zielsystems und darauf aufbauend die Synthese des Objektsystems. So werden Ergebnisse entsprechend des im Zielsystem vorhandenen Wissens und der Anforderungen erarbeitet und ins Objektsystem überführt. Bei der Validierung erfolgt umgekehrt eine Analyse des Objektsystems und darauf aufbauend die Synthese des Zielsystems. Im Rahmen der Analyse wird das jeweils aktuelle Objektsystem den Stakeholdern präsentiert und von ihnen bewertet. Bei der anschließenden Synthese wird überprüft, ob die Informationen im Zielsystem den Erwartungen der Stakeholder entsprechen. Bei Bedarf wird das Zielsystem entsprechend angepasst. So kann Validierung unter anderem zur Auswahl von Lösungsvarianten und entsprechend deren Weiterentwicklung führen [VDI2206:2004]. Zudem erfolgt in positiven Validierungsfällen die Feststellung der Erfüllung einer oder mehrerer Anforderungen oder eines gewünschten Reifegrads der Entwicklung. Umgekehrt führt Validierung zur Identifikation nicht erfüllter Anforderungen und entsprechend zur Berücksichtigung dieser Anforderungen im weiteren Entwicklungsprozess. Des Weiteren kann Validierung dazu führen, dass neue Randbedingungen aufgezeigt oder neue Ziele definiert werden. Validierung generiert somit neues Wissen, das in das Zielsystem überführt und in weiteren Validierungsaktivitäten berücksichtigt wird [Tur14]. Die Aktivitäten des Designs und der Validierung stellen wie in Abbildung 3-5 angedeutet einen Kreislauf dar. So werden die im Laufe der Produktentwicklung erarbeiteten Ergebnisse regelmäßig mit den im Zielsystem auf Kunden- und Anbieternutzen basierenden Zielstellungen und Lastenheftforderungen abgeglichen [Kli16]. Die Resultate und Erkenntnisse der Validierung beeinflussen jeweils den weiteren Verlauf der Entwicklung in der nächsten Iteration. Validierung stellt somit die Erfüllung der Anforderungen und der Ziele und damit verbunden eine hohe Produktqualität sicher.

Die von ALBERS ET AL. auch als „**repetitive Sequenz von Kreations- und Validierungsschritten**“ [ARF+19, S. 430] beschriebene Produktentwicklung spiegelt sich ebenfalls in weiterem Vorgehensmodellen der Produktentwicklung wieder. Grundsätzlich bilden die Anforderungen stets die Grundlage der Produktentwicklung und werden kontinuierlich für die Validierung herangezogen [ERZ14, S. 67]. Dies wird unter anderem auch in der Entwicklungsmethodik für mechatronische Systeme VDI2206 beschrieben. Im Rahmen der Validierung ist hier immer wieder eine Analyse und Evaluation der im Laufe der Entwicklung iterativ entwickelten Lösungsvarianten und Teillösungen bzgl. der im Rahmen der Zielformulierung an sie gestellten Anforderungen vorgesehen [VDI2206:2004].

Aufgrund der Vielfalt und der zunehmenden Interdisziplinarität heutiger Marktleistungen können die gestellten Anforderungen je nach Branche und Produktbereich verschiedenste Produktaspekte adressieren. Konsumgüter haben meist andere Anforderungen als Produktionsmaschinen. Exemplarische Produktbereiche und damit verbundene typische Produktaspekte werden unter anderem von PONN UND LINDEMANN vorgestellt [PL11]. Je nach Produkt und Branche sind Anforderungen bzgl. unterschiedlicher Produktaspekte zu validieren. Mit Validierung wird sichergestellt, dass die entwickelte Marktleistung die

an sie gestellten Anforderungen und die Bedarfe erfüllt und somit eine hohe Produktqualität aufweist. Entsprechend hat Validierung großen Einfluss auf den Verlauf und letztendlich auf den Erfolg eines Produktentwicklungsprozesses. Validierung wird daher auch als **zentrale Aktivität in der Produktentwicklung** gesehen und muss von Anfang an kontinuierlich durchgeführt werden [Alb10, ABK+16]. Dies wird durch eine mit 66 Unternehmensvertretern durchgeführte Umfrage bestätigt [Rei21]. 98% der Befragten stimmen zu, dass eine frühe Validierung wichtiger Bestandteil von Entwicklungsprozessen ist. Die Umfrage bestätigt zudem, dass Validierung wichtig ist für die Einhaltung von Terminzielen (80% Zustimmung) und die Erreichung der Kostenziele (76% Zustimmung), unter anderem, da sie späte Änderungen vermeidet (83% Zustimmung). Des Weiteren sind die meisten der Befragten der Meinung, dass Validierungsaktivitäten einen wichtigen Beitrag für die Berücksichtigung von Kunden- und Anwenderanforderungen (83% Zustimmung) und letztendlich für den Markterfolg des Produkts leisten (82% Zustimmung).

Validierung ist jedoch gleichzeitig auch der **kostspieligste und herausforderndste Prozess** in der Produktentwicklung [Kli16, ABK+16]. Speziell die Komplexität heutiger Systeme führt in vielen Branchen zu speziellen Herausforderungen in der Validierung. So zeigt die Umfrage von REINEMANN unter anderem auch, dass eine frühe Validierung oft nicht ausreichend durchgeführt wird (82% Zustimmung) [Rei21]. Dass Validierung herausfordernd ist und eine unzureichende Validierung dazu führen kann, dass Produkte nicht einsatzfähig sind und Unternehmen unter anderem finanzieller Schaden zugefügt wird, zeigt das Beispiel der Riesen-Windkraftanlage Growian [WEL07-ol]. Growian wurde gefördert vom Bundesforschungsministerium entwickelt und 1983 in Schleswig-Holstein in Betrieb genommen. Die Größe und die Technik waren neu und somit viele Aspekte bspw. bezüglich des Verhaltens des Windrads unklar. Genauere Erkenntnisse über Belastungen und Anforderungen konnten erst nach der Inbetriebnahme gewonnen werden. Die Validierung fand entsprechend erst sehr spät nach der Entwicklung und bereits während des Betriebs statt. Dabei wurde festgestellt, dass das System den Anforderungen und Rahmenbedingungen nicht gerecht wird. Es kam zu Pannen und auch zu Beschädigungen am Windrad, sodass Growian bereits 1987 wieder stillgelegt wurde. Eine frühe Validierung während der Entwicklung hätte viele Kosten und einige Schaden verhindern können. Dies war nach dem damaligen Stand der Technik allerdings nicht möglich. Heutzutage wäre eine frühe Validierung des Riesen-Windrads beispielsweise anhand digitaler Modelle und Simulationen möglich [ABK+16]. Grundsätzlich gibt es verschiedene Ansätze zur Validierung während der Produktentwicklung, auf die im folgenden Abschnitt 3.3.2 eingegangen wird.

3.3.2 Ansätze zur Produktvalidierung

Zunächst lässt sich festhalten, dass es keine festgelegte Vorgehensweise oder Methodik zur Validierung gibt und Validierung im Allgemeinen nicht formal durchzuführen ist

[VDI2206:2004]. Abhängig von der Branche, des Produkttyps und weiteren Rahmenbedingungen müssen Unternehmen die Validierung individuell gestalten und sie situations- und bedarfsgerecht durchführen. Wie in Abschnitt 3.3.1 beschrieben, wird bei der Validierung der jeweils aktuelle Entwicklungsstand mit den Anforderungen abgeglichen und entsprechend abgesichert. Dazu müssen Produktmerkmale analysiert und bewertet werden. Grundlage der Analyse und Bewertung ist stets eine **Repräsentation des Produkts** bzw. des aktuellen Entwicklungsstands.

Um Zeit und Kosten zu sparen, wird das Produkt meist nicht als Ganzes repräsentiert und validiert. Stattdessen werden anhand von **Prototypen** lediglich die für die Absicherung einzelner Aspekte notwendigen Produktmerkmale abgebildet [GDE+18, S. 67]. Der Begriff Prototyp leitet sich aus dem Griechischen ab. „Proto“ bedeutet auf Deutsch „erster“ oder „vorderster“ und „typus“ steht für „Muster“ oder „Urbild“ [Geu96]. Prototypen entsprechen somit meist einem ersten Muster, das zwar einem vereinfachten Versuchsmodell des geplanten Produkts entspricht, diesem aber zumindest in Bezug auf die repräsentierten Produkteigenschaften sehr nah kommt. ULRICH UND EPPINGER definieren Prototypen als eine „Annäherung an das Produkt entlang einer oder mehrerer interessanter Dimensionen“ (übersetzt aus dem Englischen) [UE12, S. 291]. Nach dieser Definition kann alles, was mindestens ein relevantes Merkmal eines Produkts repräsentiert, als Prototyp bezeichnet werden. Nicht relevante Dimensionen bzw. Produktmerkmale werden bei Prototypen entsprechend vernachlässigt. Zum einen lassen sich Prototypen somit schneller, einfacher und günstiger erstellen und vorbereiten. Zum anderen ermöglicht die Reduzierung der repräsentierten Eigenschaften und Funktionalitäten eine Fokussierung auf das Wesentliche und somit eine bessere Beantwortung relevanter Fragestellungen [CD17]. Durch Prototypen lassen sich bereits früh in der Entwicklung mögliche technische Risiken überprüfen, Gestaltungsentscheidungen treffen und das Nutzungserlebnis evaluieren [DH17]. Teilweise werden weitere Begriffe im Kontext von Prototypen verwendet, die nicht einheitlich definiert sind und oft synonym verwendet werden. MAIMANN beschreibt bspw. die Begriffe Mockup und Modell als Arten von Prototypen in verschiedenen Reifegraden [Mai20-ol]. Mockups werden meist früh in der Entwicklung eingesetzt, um ohne signifikante Aufwände erste Ideen und Konzepte zu überprüfen. Modelle sind gegenüber Mockups detailgetreuer. Nach STACHOWIAK entsprechen sie einem abstrakten Abbild des Originals, das für einen bestimmten Zweck angefertigt wird [Sta73]. PONN UND LINDEMANN nennen unter anderem Funktionsmodelle, digitale Geometriemodelle, Kinematik- und Dynamik-Modelle sowie physische Modelle als mögliche Modellarten [PL11, S. 21]. Nach der Definition von MAIMANN entsprechen Prototypen Modellen mit hohem Detailgrad und starker Seriennähe [Mai20-ol]. Im Folgenden wird der Begriff Prototyp entsprechend der gängigen Definition nach ULRICH UND EPPINGER für alle vereinfachten Repräsentationen des zu entwickelnden Produkts während der Entwicklung verwendet [UE12, S. 291].

Als Oberbegriff für die Entwicklung von Prototypen als Produktrepräsentationen im Rahmen der Validierung wird der Begriff **Prototyping** verwendet [UE12, S. 291]. Beim Prototyping werden ausgehend von der Motivation und der Entscheidung zum Einsatz von Prototypen zunächst Werkzeuge, Materialien und Methoden für die Erstellung der Produktrepräsentation ausgewählt. Darauf basierend erfolgt die eigentliche Erstellung der Prototypen entsprechend der jeweiligen Anforderungen und der abzubildenden und zu bewerteten Produktmerkmale [DH17, S. 183]. Anschließend werden Prototypen zur Validierung verwendet und die abgebildeten Produktmerkmale analysiert und bewertet. Für die dazu notwendige Ermittlung von Merkmalswerten beschreibt EL-HAJI unter anderem angelehnt an [Gra06] verschiedene grundsätzliche **Methoden** [El-16] :

- **Inspektion:** Ermittlung von Merkmalswerten mit menschlichen Sinnen oder einfachen technischen Geräten anhand eines materiellen Prototyps
- **Erprobung und Test:** Erfassung von Merkmalswerten mittels technischer Hilfsmittel (bspw. Messgeräten) an materiellen Prototypen, die im Rahmen von spezifischen Szenarien in einen operativen Zustand versetzt werden
- **Berechnung:** Ermittlung von Merkmalswerten basierend auf analytischen und numerischen Berechnungen basierend auf verschiedenen Modellen
- **Kombinierte Methoden:** Beim Einsatz verschiedener Prototypen können die zuvor genannten Methoden kombiniert werden.

Je nach Produkt, zu validierenden Produktmerkmalen und weiteren Rahmenbedingungen werden die Methoden unterschiedlich verwendet. Zur Unterstützung werden zudem verschiedene **Validierungswerkzeuge** eingesetzt. Grundsätzlich werden Prototypen bereits als Validierungswerkzeuge angesehen [El-16]. Weitere Validierungswerkzeuge dienen beispielsweise der Präsentation der Prototypen oder der Ermittlung und Auswertung der Merkmalswerte. Im Folgenden werden verschiedene Ansätze zur Validierung vorgestellt, die verschiedene Prototypen, Methoden und Werkzeuge umfassen.

Ursprünglich wurde Validierung aufgrund der technischen Möglichkeiten lediglich anhand **physischer Prototypen** durchgeführt. Dabei werden funktionale Konzepte und Design-Ideen physisch abgebildet und im Rahmen von Inspektionen, Erprobungen und Tests validiert. Ein Fokus physischer Prototypen liegt auf Gestaltelementen wie der Form, den Materialien oder der Farbe eines Produkts [Geu96]. Zudem waren physische Prototypen oft die einzige Möglichkeit, Funktionalitäten und die Robustheit des geplanten Produkts zu überprüfen. Lange Zeit wurden physische Prototypen mangels Alternativen mit den gleichen Verfahren und Maschinen wie das Serienprodukt hergestellt. Dies war mit einem hohen Aufwand und langen Entwicklungs- und Fertigungszeiten für teilweise komplexe prototypische Komponenten und Teilespekte des Produkts verbunden. Für das Prototyping wurde meist mehr als 25 Prozent der Produktentwicklungszeit verwendet, was oft mehrere Monate bedeutete [EKL+96]. Dieser Fakt verdeutlicht die Relevanz von dreidi-

dimensionalen physischen Produktmodellen für eine aussagekräftige Überprüfung des Produkts während der Entwicklung. Dies gilt speziell bei Produkten, bei denen das Erscheinungsbild von großer Bedeutung ist. Ein Beispiel ist die Automobilbranche. Bereits seit Mitte des 20. Jahrhunderts nutzen Automobil-Designer Clay-Modelle, um realitätsnahe Prototypen des geplanten Fahrzeugs zu erzeugen. Laut Aussagen der Designer ermöglichen ihnen die Modelle eine bessere Überprüfung und Präsentation des Designs anhand von Inspektionen im Gegensatz zu Skizzen und später digitalen Abbildungen [Gib16-ol].

Durch die bereits Ende des 20. Jahrhunderts kürzer werdenden Produktlebenszyklen und schnelleren Innovationen wurde der Bedarf an Modellen und Prototypen zur Validierung zunehmend erhöht [Geu96]. Gleichzeitig führte die Digitalisierung zu einer verstärkt computergestützten Produktentwicklung. Mittels Software-Lösungen wie CAD-Programmen wurden Produktaspekte wie die Gestalt und das Design des geplanten Produkts von Beginn an digital repräsentiert [GDE+18, S. 382]. Um den Bedarf an Prototypen besser und vor allem schneller erfüllen zu können, wurden Technologien und Verfahren entwickelt, um physische Produktmodelle direkt aus den digitalen 3D-CAD-Daten herzustellen. Diese Technologien werden unter dem Begriff **Rapid Prototyping** zusammengefasst. Zu den Rapid Prototyping-Technologien zählen generative Fertigungsverfahren wie Additive Fertigung sowie weitere Computer-Aided-Manufacturing-Technologien (CAM) [Geu96]. Mit additiver Fertigung beispielsweise können dreidimensionale Modelle ohne Zerspanungsvorgänge aus Kunststoff oder Wachs oder auch aus Metallen und weiteren Materialen generiert werden. Diese Modelle sind vor allem geeignet, um optische oder haptische Eigenschaften zu überprüfen [EKL+96]. Dabei ist jedoch abhängig vom Herstellungsverfahren die Wiedergabtreue der Modelle und damit die Vergleichbarkeit mit dem späteren finalen Produkt zu berücksichtigen [Geu96]. Zudem können weitere Produktmerkmale, beispielsweise das Verhalten betreffend, mit Rapid Prototyping-Technologien meist nicht validiert werden.

Grundsätzlich wird für die Erstellung von physischen Prototypen eine gewisse Fertigungszeit benötigt, die mit agilen Arbeitsweisen mit kurzen Entwicklungsiterationen und effizienten Validierungsaktivitäten nicht vereinbar ist [SCK+17]. Gleichzeitig ist der physische Prototypenbau meist material- und kostenintensiv und kann oft erst spät im Entwicklungsprozess erfolgen [ARF+19, S. 429]. Daher wird grundsätzlich eine **Reduzierung der physischen Prototypen** angestrebt [VDI2206:2004]. So erwarten Unternehmen, dass Validierung zunehmend digital oder virtuell durchgeführt wird [DAG+21]. Getrieben und unterstützt wird dies durch die zunehmend computergestützte Produktentwicklung. Für die Entwicklung und die Validierung relevante Produktaspekte wie die Anforderungen, die Gestalt sowie auch die Funktionalität, die Wirkungsweise und das Verhalten werden von Beginn an digital repräsentiert [GDE+18, S. 382]. Die während der Entwicklung im Zielsystem (vgl. Abschnitt 3.3.1) erarbeiteten Artefakte bestehen daher zu großen Teilen aus digitalen Daten, Modellen und Software. Diese digitalen Produktrepräsentationen ermöglichen vielfältige digitale Ansätze zum Prototyping und zur

Validierung. In Kombination mit digitalen Visualisierungstechnologien erlauben sie frühes Ergebnisfeedback und unterstützen so schnelle Entwicklungszyklen [RC16].

Basierend auf digitalen 3D-Modellen lassen sich **digitale Mockups (DMU)** erstellen, die einer realitätsnahen digitalen Abbildung des geplanten Produkts am Rechner entsprechen. Basierend auf DMUs können mittels verschiedener Visualisierungswerzeuge das Design und die Gestalt sowie räumliche Aspekte beispielsweise anhand von Kollisionsprüfungen bewertet werden [RLS+15]. Dies entspricht digitalen Inspektionen. 3D-Modelle werden laut der durchgeführten Unternehmensbefragung von vielen Unternehmen bereits als DMUs zur Validierung eingesetzt (s. Anhang A1). Werden zudem funktionale Aspekte berücksichtigt, wird auch von funktionalen DMUs (FDMU) gesprochen. Mit FDMUs lassen sich entsprechend gewisse Funktionalitäten des Produktes digital erproben und bewerten [ES09].

Eine Erweiterung von DMUs sind **virtuelle³ Prototypen**, die neben der Gestalt und Funktionalitäten weitere Aspekte wie Kinematik und Dynamik abbilden. Darauf basierend ermöglichen digitale **Simulationen** als Validierungswerzeuge die Validierung von über die Gestalt und Optik hinausgehenden Produktmerkmalen [VDI2206:2004]. Bei digitalen Simulationen handelt es sich um Berechnungen, die mittels numerischer Algorithmen auf formalen Modellen durchgeführt werden. Die Ergebnisse der Simulation werden jeweils visualisiert, um sie auswerten zu können [HGA+15, S. 4]. Anhand von Mehrkörpersimulationen kann beispielsweise das kinematische Verhalten von Mehrkörpersystemen digital simuliert und validiert werden. Mit der Finite-Elemente-Methode (FEM) werden Struktur- und Festigkeitsberechnungen oder Spannungsanalysen durchgeführt. Zudem ermöglichen Strömungssimulationen unter anderem die Analyse des Verhaltens von Flüssigkeiten in einem System oder der Luftzirkulation. Virtuelle Prototypen und Simulationen ermöglichen somit schon früh in der Entwicklung die computergestützte Absicherung von Produkteigenschaften, noch bevor physische Prototypen hergestellt werden [ERZ14, S. 175]. So sind Simulationen als Validierungswerzeug grundsätzlich die schnellste und meist günstigste Möglichkeit, die Erfüllung der an das Produkt gestellten Anforderungen zu überprüfen [MGM+17]. Abbildung 3-6 zeigt anhand eines mobilen Roboters Beispiele für verschiedene physische und digitale Arten von Prototypen.

Grundsätzlich sollen wie bereits dargelegt physische Prototypen reduziert werden. Mit digitalen Ansätzen wie Simulationen können jedoch nicht alle Produktmerkmale in allen Reifegraden ausreichend validiert werden. So können zu überprüfende Produktmerkmale teilweise aufgrund ihrer hohen Komplexität nicht ausreichend detailliert digital abgebildet werden oder die Modellierung ist zu aufwendig. Daher lassen sich aufgrund von Abweichungen zwischen dem realen und dem simulierten Verhalten physische Prototypen

³ Der Begriff „virtuell“ meint, dass etwas nicht in der Realität vorhanden ist, dem Betrachter aber sehr realitätsnah erscheint. Im Gegensatz zu etwas Digitalem ist beim Virtuellen entsprechend eine gewisse Realitätsnähe ausschlaggebend. Für weitere Erklärungen siehe bspw. <https://www.dwds.de/wb/virtuell>

und Tests in vielen Fällen nicht vollständig vermeiden [DAG+21]. In diesen Fällen werden sogenannte **X-in-the-Loop-(XiL)-Ansätze** eingesetzt, bei denen Simulationen mit realen Systemen kombiniert werden [MGM+17]. Es handelt sich entsprechend der von EL-HAJI [El-16] verwendeten Definition um kombinierte Methoden. Physische und digitale Prototypen werden kombiniert, sodass nur einzelne Teilsysteme physisch für die Validierung vorliegen müssen. Sie werden in virtuelle Testumgebungen eingebettet und kommunizieren und interagieren über Schnittstellen mit digitalen Simulationen. Abhängig vom Entwicklungsstand und dem notwendigen Grad der Realitätsnähe der Simulation gibt es verschiedene XiL-Ansätze. *Model-in-the-Loop-Ansätze* werden vor allem für die Entwicklung von eingebetteten Systemen eingesetzt. Die Software des eingebetteten Systems wird als Modell abgebildet und in einer simulierten digitalen Systemumgebung getestet und validiert. Bei *Software-in-the-loop* wurde die Software bereits entwickelt und wird in einer simulierten Umgebung getestet [ABK+16]. *Hardware-in-the-loop* fokussiert die Validierung physischer Hardware und darauf laufender Software. Mit dem IPEK-X-in-the-loop-Ansatz existiert zudem ein Ansatz, der die beschriebenen Ansätze aufgreift, sie teilweise erweitert und so eine integrative Betrachtung des Gesamtsystems ermöglicht [AD10, S. 2]. Grundsätzlich bieten XiL-Ansätze trotz der wachsenden Komplexität heutiger Systeme Möglichkeiten zur effizienten und möglichst kostengünstigen Validierung einzelner Produkteigenschaften mit hoher Aussagekraft. Auf höchster Integrationsebene folgen den XiL-Ansätzen komplett physische Prototypen, die für abschließende Tests vor der Markteinführung eingesetzt werden [VDI2206:2004].

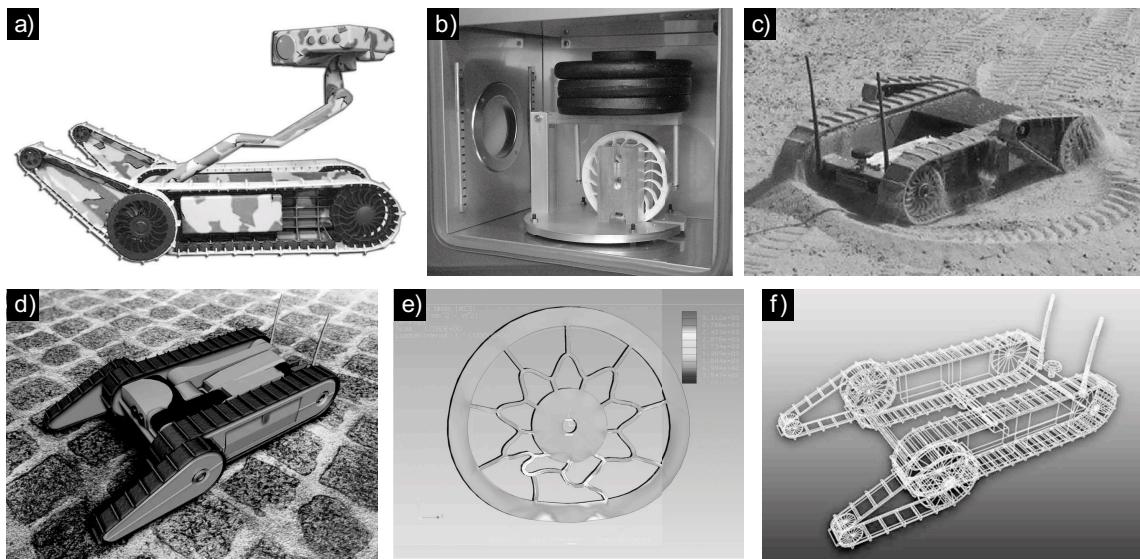


Abbildung 3-6: Verschiedene Prototypen am Beispiel eines mobilen Roboters zusammengestellt aus [UE12]: a) Physisches Anschauungsmodell, b) Physischer Prototyp eines Rads, c) Physisches Gesamtmodell, d) Digitales 3D-Anschaungsmodell, e) Digitales Rad-Modell als Grundlage für Simulationen, f) Dynamik-Simulationsmodell

Der vermehrte Einsatz von Simulationen und XiL-Ansätzen zur Vermeidung von physischen Prototypen und somit zur effizienteren und günstigeren Gestaltung der Validierung

wird von vielen Unternehmen bestätigt [DAG+21]. Gleichzeitig stellen die zunehmende Komplexität sowie die verstärkte Individualisierung der Systeme Unternehmen jedoch vor Herausforderungen bei der Validierung. Mittels der aufgezeigten Validierungsansätze lassen sich viele Produktmerkmale entsprechend der Anforderungen ausreichend überprüfen. Dabei handelt es sich jedoch größtenteils um Eigenschaften, die objektiv und ohne direkte Einbindung der Kunden bewertet werden können. Aspekte, die subjektiv durch die Kunden validiert und bewertet werden müssen, bedürfen der Integration der Kunden in die Validierung.

3.3.3 Kunden-integrierte Validierung

Validierung dient der Sicherstellung, dass das richtige Produkt entwickelt wird und es den gewünschten Zweck erfüllt. Dies entspricht der Überprüfung, ob alle Anforderungen an das Produkt erfüllt werden. Im Kontext der zunehmenden Individualisierung und Nutzerorientierung werden die Anforderungen durch die Kunden definiert oder zumindest stark beeinflusst. Damit das Produkt seinen Zweck erfüllt, müssen entsprechend gewisse vom Kunden geforderte Produktmerkmale gegeben sein. Diese Produktmerkmale können verschiedene Eigenschaften, Funktionalitäten oder auch ein bestimmtes Verhalten sein. Für eine erfolgreiche Entwicklung entsprechend der Anforderungen sollten die Entwickler die Gründe und den Zweck des Produkts verstehen. Dazu ist eine enge Kommunikation mit dem Kunden notwendig [BO20, S. 8]. Zudem kann oft nur der Kunde selbst beurteilen, ob die Anforderungen erfüllt werden und das Produkt seinen Vorstellungen entspricht. Daher ist die **frühe und kontinuierliche Einbindung von Kunden in die Validierung** von enormer Bedeutung, um auf Kundenbedarfe eingehen und Kundenanforderungen besser erfüllen zu können [ARF+19, S. 431]. Die Einbindung von Kunden in die Entwicklung und entsprechende Feedbackmechanismen führen zudem zu effizienterem Arbeiten und stärken die Kundenbindung [KKS19].

Kunden-integrierte Validierung ist zum einen wichtig, wenn ein Produkt individuell für einen spezifischen Kunden entwickelt wird. In diesen Fällen werden die Anforderungen an das zu entwickelnde Produkt direkt vom Kunden definiert. Entsprechend erfolgt die Validierung unter Einbeziehung des spezifischen Kunden. Durch die verstärkte Nutzerorientierung und den Wettbewerbsdruck am Markt sind zum anderen auch bei einer Produktentwicklung für den freien Vertrieb am Markt Kunden in die Entwicklung einzubeziehen. Im Fall dieser Markt-orientierten Entwicklung gibt es keinen spezifischen Kunden. Stattdessen wird ein Produkt entwickelt, das an möglichst viele Kunden verkauft werden soll [GSW14]. Entsprechend gilt es, die Bedarfe und Anforderungen einer großen Menge an potenziellen Kunden mit dem Produkt zu erfüllen. Durch Validierungsaktivitäten in verschiedenen Phasen der Entwicklung müssen Konzepte und Entwicklungsstände mit potenziellen Kunden abgesichert und damit sichergestellt werden, dass das Produkt später am Markt erfolgreich sein kann.

Während bei der individuellen Entwicklung die Anforderungen direkt vom Kunden kommen, ist es zu Beginn der Markt-orientierten Entwicklung für Unternehmen zunächst wichtig, eine **Marktanalyse** durchzuführen. Anhand einer Marktanalyse werden die potenziellen Kunden, das relevante Unternehmensumfeld und weitere Einflussfaktoren analysiert und bewertet. Eingesetzte Methoden zur Identifikation von Kundenbedarfen sind beispielsweise Interviews, Diskussionen in Fokus-Gruppen oder Beobachtungen von Kunden beim Gebrauch eines Produkts [GSW14]. GAUSEMEIER ET AL. beschreiben in [GDE+18] weitere klassische Methoden der **Kundenbefragung** und ordnen sie entsprechend der Art der Informationssuche und des Detaillierungsgrads der Informationen ein. Es wird unterschieden zwischen qualitativen und quantitativen Methoden. Die verschiedenen Methoden ermöglichen durch Kommunikation und Interaktion mit potenziellen Kunden Abschätzungen der Interessen, Bedürfnisse und Wünsche bezüglich eines Produkts. Als eine der bekanntesten Methoden gilt die Conjoint-Analyse. Bei dieser quantitativen Methode können Erkenntnisse zum Kaufverhalten von Kunden und über die Wichtigkeit einzelner Produktmerkmale aus Sicht des Kunden gewonnen werden [BEP+16]. Basierend auf der Marktanalyse und Kundenbefragungen werden anschließend Bedarfe und Wünsche der Kunden antizipiert und daraus Anforderungen und Zielspezifikationen an das zu entwickelte Produkt abgeleitet.

Die anschließende Vorgehensweise der Entwicklung und Validierung ist bei der kunden-spezifischen und der Markt-orientierten Entwicklung ähnlich. Ausgehend von den Anforderungen werden immer wieder Konzepte und Teilespekte des Produkts validiert und die Erkenntnisse in darauffolgenden Entwicklungsiterationen verwendet. Zu Beginn werden verschiedene Produktkonzepte erarbeitet, die die Kundenbedarfe und -anforderungen vermeintlich erfüllen. In einer ersten Validierungsphase gilt es anschließend, die Konzepte vom spezifischen Kunden oder einer Menge potenzieller Kunden validieren zu lassen, um anschließend vielversprechende Konzepte weiterzuentwickeln. Sind die Konzepte weiterentwickelt worden, werden sie in einem weiteren Schritt getestet und somit mit einem höheren Reifegrad validiert. ROBERT ET AL. beschreiben dies in einem iterativen Kunden-integrierten Validierungsprozess, bei dem Prototypen und Konzepte so lange mit Kunden validiert und überarbeitet werden, bis eine ausreichende Produktqualität wahrgenommen wird (Abbildung 3-7).

Grundlage der Validierung ist eine prototypische Repräsentation des Produkts, die das Produktkonzept oder einzelne Produktmerkmale wiedergibt (vgl. Abschnitt 3.3.2). Für die Produktrepräsentation bei der kunden-integrierten Validierung nennen ULRICH UND EPPINGER folgende Möglichkeiten [UE12]:

- **Verbale Beschreibung:** Kurzer Absatz oder stichpunktartige Liste zur Beschreibung des Produktkonzepts.
- **Sketch:** Typischerweise einfache Strichzeichnungen mit Annotationen der Haupt-eigenschaften des Konzepts.

- **Fotos und Renderings:** Fotos oder alternativ Renderings können das geplante Erscheinungsbild wiedergeben. Renderings können nahezu fotorealistisch generiert werden, wenn 3D-Modelle bereits existieren.
 - **Storyboard:** Ein Storyboard entspricht einer Bildergeschichte, die eine mögliche Interaktion mit dem Produkt oder eine Aktion des Produkts kommuniziert.
 - **Video:** Videos ermöglichen eine dynamischere und klarere Kommunikation von Produkteigenschaften und der Art und Weise, wie ein Produkt benutzt wird.
 - **Simulation:** Produktfunktionalitäten können software-technisch simuliert und prototypisch nachgeahmt werden, um bspw. die Interaktion mit dem Produkt validieren zu können.
 - **Interaktive Multimedia-Inhalte:** Repräsentationen wie Fotos, Videos und Simulationen können kombiniert werden. Der Kunde kann interaktiv Inhalte betrachten oder auch Audio-Ausgaben hören, um einen Eindruck vom Produkt zu bekommen.
 - **Physische Anschauungsmodelle:** Anhand physischer Modelle kann das Erscheinungsbild des geplanten Produkts wiedergegeben werden. Oft handelt es sich um Holz- oder Polymerschaum-Modelle, die realitätsnah lackiert werden.
 - **Funktionale Prototypen:** Produktfunktionalitäten können genutzt und bewertet werden. Es besteht die Gefahr, dass Kunden Prototypen für das fertige Produkt halten und die Funktionalität aber teilweise abweicht.

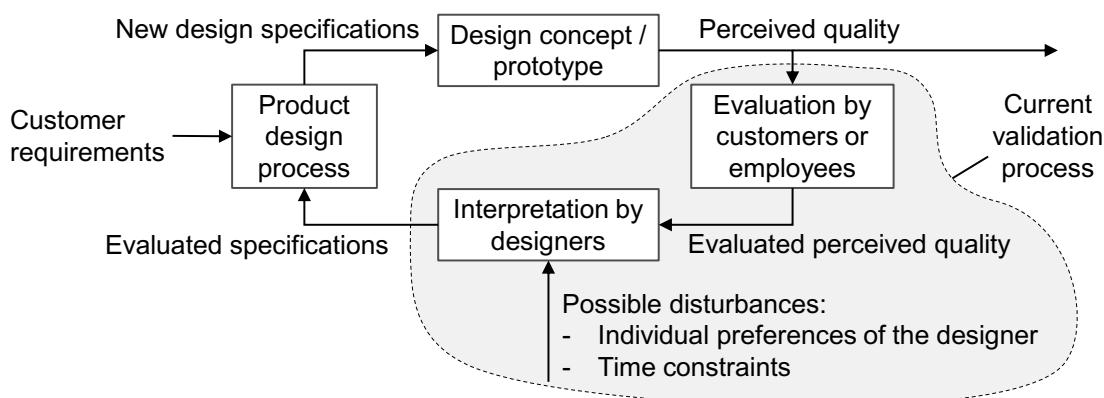


Abbildung 3-7: Generischer iterativer Prozess zur Kunden-integrierten Validierung von Konzepten und Prototypen in Anlehnung an [RBD15, S. 48]

Konkrete Ansätze zur kunden-integrierten Validierung werden vor allem für die **Markt-orientierte Produktentwicklung** beschrieben. In diesem Fall erfolgt die Validierung mit einer Menge potenzieller Kunden. ULRICH UND EPPINGER beschreiben bspw. verschiedene klassische Methoden wie die *Face-to-face Interaktion*, *Interviews per Telefon*, *Briefpost* oder elektronischer *Mail* sowie *Internet*-basierte Umfragen und Produktpräsentationen [UE12, S. 168f.] BLAND UND OSTERWALDER beschreiben weitere innovative Validie-

rungsexperimente, die Markt-orientiertes Kundenfeedback ermöglichen [BO20]. Die Experimente haben das Ziel, Produkt- und Geschäftsideen effizient und aussagekräftig mit potenziellen Kunden zu überprüfen. Es wird unterschieden zwischen *interaktiven Prototypen*, *Call to Action* und *Simulation*. Im weiteren Verlauf der Entwicklung ist dann vor allem auch das Nutzungsverhalten und das Nutzungserlebnis für den Kunden von enormer Bedeutung für den späteren Erfolg des Produkts. Um eine realistische Validierung des Nutzungserlebnisses zu ermöglichen, sollte die **Validierung im realen Kontext** und im Idealfall über einen längeren Zeitraum im Alltag der Menschen durchgeführt werden [DH17, S. 174]. Ein entsprechender Ansatz sind *Platzhalterprototypen*, die einen Produktidee oder ein Produktkonzept repräsentieren und Testpersonen ihrem Alltag begleiten. Die Testpersonen dokumentieren ihren Umgang mit dem Platzhalterprototypen und ihre Eindrücke und reflektieren Möglichkeiten der Nutzung [SPS10]. Ein weiterer Ansatz ist die *Designimprovisation*, bei der Konzepte durch aktives Ausprobieren und die Interaktion zwischen verschiedenen Testpersonen als potenziellen Kunden erlebbar gemacht werden [DH17, S. 177].

Im Falle der **Individualisierung von Produkten für spezifische Kunden** sind konkrete an das Produkt gestellte Anforderungen zu erfüllen. Sie gilt es gezielt zu validieren und auf mögliche Änderungen schnell und flexibel zu reagieren [SCK+17]. Die grundlegenden Anforderungen an das Produkt werden vom Kunden definiert. Sie sind entsprechend durch diesen zu validieren. Aufgrund der zunehmenden Interdisziplinarität sowie der abteilungs- und unternehmensübergreifenden Zusammenarbeit (vgl. Abschnitt 3.2.3) bestehen zudem Entwicklungs-interne Anforderungen. Entsprechend sind die Entwicklungsstände eines Teilsystems unter Umständen durch eine andere Abteilung als interne Kunden zu validieren. Grundsätzlich sind eine intensive **Kommunikation und Kooperation mit internen und externen Kunden** während der Entwicklung notwendig. Diese Zusammenarbeit wird jedoch durch verschiedene Faktoren erschwert. Ein Faktor ist die zunehmende **geografische Trennung** der in ein Entwicklungsprojekt involvierten Akteure. Durch die Globalisierung und die zunehmend verteilte Produktentwicklung und Produktion arbeiten häufig international verteilte Ingenieure an einem Projekt [ERZ14]. Zudem führen die Möglichkeiten der Digitalisierung und ein stärker selbstbestimmtes Arbeiten zu einer zeitlichen und räumlichen Entgrenzung der Arbeit [Bun15]. So erschweren flexible Arbeitsplätze und -zeiten die Zusammenarbeit in Entwicklungsprojekten [GDE+18].

Es mangelt bislang an effizienten Ansätzen zur standortübergreifenden gezielten Validierung von individuellen Produkten und Konzepten mit Kunden. Unternehmen müssen sich daher eigene Mittel und Wege überlegen, um Entwicklungsstände mit ihren Kunden abzusichern. Neben bereits beschriebenen Nachteilen physischer Prototypen wie Kosten und Zeit kommt durch eine räumliche Trennung ein weiterer hinderlicher Aspekt hinzu. Physische Prototypen müssten zur Validierung in vielen Fällen zum Kunden transportiert oder versandt werden. Dies lässt sich mit einer effizienten agilen Arbeitsweise nicht ver-

einen. So setzen die meisten Unternehmen auf digitale Prototypen und Validierungswerzeuge. Laut der durchgeführten Unternehmensbefragung versuchen viele Unternehmen, Anforderungen mittels digitaler Simulationen und interner Tests ohne direkte Einbindung des Kunden zu überprüfen (vgl. Anhang A1). Dies ist jedoch nur bei objektiv bzw. quantitativ zu bewertenden Produkteigenschaften möglich. In anderen Fällen führen Unternehmen die Validierung durch interne Angestellte durch [UE12]. Entsprechende Vorgehensweisen haben ein hohes Risiko, da Angestellte voreingenommen sein und andere Vorstellungen vom Produkt als der Kunde haben können [VB92]. Eine Trennung von Entwicklerteams und Kunden muss vermieden werden, um die Probleme der Kunden zu lösen und ihre Anforderungen zu erfüllen [BO20, S. 11]. Dennoch haben viele der Unternehmen in der Engineering-Branche ko-kreative Prozesse zur Kundeneinbindung noch nicht in die internen Entwicklungsprozesse integriert [KKS19]. Das liegt unter anderem an mangelnden Ressourcen durch die immer kürzer werdenden Entwicklungszyklen und den erhöhten Kostendruck [RBD15]. 72% der Anfang 2021 befragten Unternehmensvertreter haben der Aussage zugestimmt, dass interne und externe Kunden bereits in die Validierung in Entwicklungsprojekten eingebunden werden (Abbildung 3-8a). Diese Einbindung erfolgt unter anderem durch Kundenbefragungen, Testmärkte, die Überprüfung von Checklisten oder den Austausch von Projektmappen. 54% stimmen zudem der Aussage zu, dass Validierung bereits standortübergreifend anhand digitaler Produktmodelle und -informationen durchgeführt wird (Abbildung 3-8b). In diesen Fällen handelt es sich größtenteils um die gemeinsame Betrachtung klassischer Entwicklungsartefakte. Teilweise werden 3D-CAD-Modelle oder daraus abgeleitete Renderings in digitalen Meetings gemeinsam betrachtet und diskutiert (vgl. Anhang A1).

Es lässt sich festhalten, dass verschiedene Methoden für die Integration von Kunden in die Validierung existieren. Diese adressieren vor allem die Validierung im Kontext Markt-orientierter Entwicklungsprojekte. Bei der Entwicklung individueller Produkte für spezifische Kunden ist Kunden-integrierte Validierung noch nicht ausgereift. Unternehmen versuchen, große Teile der Validierung ohne Kunden durchzuführen, um Zeit und Kosten zu sparen. Für subjektiv zu bewertende Produktmerkmale wie beispielsweise das Design oder die Interaktion sind Kunden jedoch definitiv einzubinden. Hier setzen Unternehmen auf den asynchronen Austausch klassischer Entwicklungsartefakte mit den Kunden, um Feedback einzuholen, oder auf digitale Meetings, in denen die Entwicklungsartefakte präsentiert, diskutiert und bewertet werden. Insbesondere bei der Entwicklung heutiger Advanced Systems reicht dies nicht aus. Die Systeme sind zunehmend vernetzt und interaktiv und stehen meist in einem engen Bezug zu ihrem Umfeld. Speziell mögliche Wechselwirkungen mit ihrem Umfeld sind nur schwer zu antizipieren und können bisher nicht ausreichend in der Validierung berücksichtigt werden [DAG+21]. Eine weitere Herausforderung ist die stark menschzentrierte Gestaltung der heutigen Systeme. So spielt die Nutzung der Systeme aus Kundensicht eine große Rolle. Auch die Gebrauchstauglichkeit und verschiedene Nutzungsszenarien müssen in einem sozialen Kontext durch den Kunden validiert werden [CD17]. Die Interaktion zwischen den in der Entwicklung involvierten Stakeholdern muss gefördert und das Produkt für interne und

externe Kunden erlebbar gemacht werden, um es ausreichend zu validieren. Aktuell mangelt es allerdings an geeigneten Validierungsmethoden zur konsequenten Berücksichtigung des Menschen als Entwickler, Käufer und Nutzer in der Validierung heutiger Advanced Systems [DAG+21]. Die bestehenden Vorgehensmodelle und Werkzeuge stoßen an ihre Grenzen, sodass neue Lösungen zur Validierung benötigt werden [GDE+18]. Die innovative Visualisierungs- und Kommunikationstechnologie Augmented Reality (AR) hat das Potential, eine dieser Lösungen zu sein. AR ermöglicht eine Zusammenarbeit, die weit über Telefon- und Videokonferenzen hinausgeht, und bietet vielversprechende Möglichkeiten zur Präsentation von digitalen Prototypen zur Kunden-integrierten Validierung. Somit kann AR als Validierungswerkzeug die Adressierung der genannten Herausforderungen unterstützen.

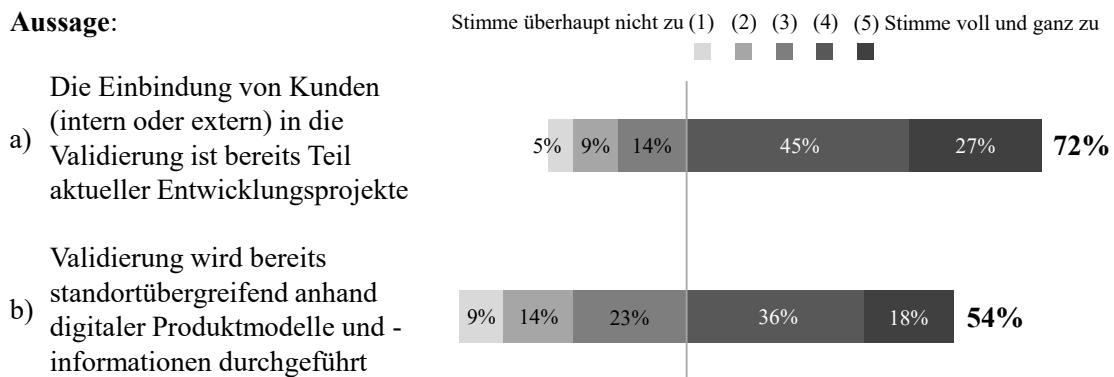


Abbildung 3-8: Umfrageergebnisse a) zur Einbindung von Kunden und b) zum Einsatz dig. Produktmodelle bei der Validierung (vgl. Anhang A1, n=22)

3.4 Augmented Reality

Bei Augmented Reality, kurz AR, handelt es sich um eine innovative Kommunikations- und Visualisierungstechnologie. Im Folgenden wird die Technologie zunächst grundsätzlich vorgestellt (Abschnitt 3.4.1). Anschließend werden die Potentiale von AR im Kontext der Validierung genauer erläutert (Abschnitt 3.4.2). Aufgrund der Komplexität der Technologie sehen sich Unternehmen beim Einsatz der Technologie jedoch mit verschiedenen Herausforderungen konfrontiert. Diese werden in Abschnitt 3.4.3 dargelegt.

3.4.1 Interaktive Erweiterung der Realität

Mittels AR wird die reale Umgebung um digitale bzw. computergenerierte Inhalte erweitert. Im Deutschen wird AR daher oft auch als „Erweiterte Realität“ bezeichnet [DBJ+19]. Der Begriff AR ist nicht eindeutig definiert und wird teilweise unterschiedlich verwendet. ALT verbindet mit AR konkret die Erweiterung der Realität um visuelle Inhalte:

Augmented Reality beschreibt die Ergänzung der visuellen Wahrnehmung des Menschen durch die situationsgerechte Anzeige von rechnergenerierten Informationen [...] [Alt02, S. 3]

Grundsätzlich steht bei der allgemeinen Verwendung des Begriffs AR die visuelle Erweiterung der Realität im Fokus. So wird AR meist über die computergestützte Erweiterung der vom Menschen wahrnehmbaren Realität durch Einblendung digitaler Inhalte im Sichtfeld des Benutzers definiert [TAB19-ol]. AR kann aber auch die Überlagerung anderer Sinneswahrnehmungen einbeziehen, beispielsweise durch akustische, haptische oder auch olfaktorische Erweiterung der Realität [Tön10]. Nach AZUMA wird AR anhand dreier Eigenschaften charakterisiert [Azu97, S. 356]:

- **Kombination** aus realer und virtueller Welt
- **Interaktivität** in Echtzeit
- **Dreidimensionale Registrierung** der virtuellen Inhalte in der realen Umgebung

Die Kombination visueller Inhalte mit der realen Welt und die Registrierung der Inhalte in ihr werden in Abbildung 3-9 veranschaulicht. Durch eine perspektivisch korrekte Darstellung und Positionierung der virtuellen Inhalte können sie mittels Augmented Reality passend mit der Realität kombiniert werden. Die nach AZUMA für AR charakteristische Interaktivität ermöglicht zudem eine Interaktion des Nutzers mit den Inhalten in Echtzeit.



Abbildung 3-9: Das Prinzip der visuellen Kombination von virtuellen Inhalten mit der Realität [Bro19, S. 316]

Zur Abgrenzung gegenüber verwandten Technologien haben MILGRAM UND KISHINO das sogenannte *Virtuality Continuum* erarbeitet (Abbildung 3-10). Das Kontinuum verortet verschiedene Ausprägungen virtueller Technologien entlang einer Achse, die den Anteil der realen und der virtuellen Inhalte angibt. Ganz links befindet man sich in der komplett realen Umgebung, in der keine virtuellen Inhalten vorliegen. Umso weiter man sich auf der Achse nach rechts bewegt, umso größer wird der virtuelle Anteil, bis schließlich ganz rechts zur komplett virtuellen Umgebung. Die komplett virtuelle Umgebung entspricht dem Begriff Virtual Reality oder kurz VR. In VR nimmt der Benutzer seine reale Umgebung zumindest visuell nicht mehr wahr. Alles zwischen realer und virtueller Umgebung definieren MILGRAM UND KISHINO als Mixed Reality, kurz MR. Entsprechend umfasst MR alle Technologien, die die reale und die virtuelle Umgebung vermischen. AR umfasst im Kontinuum den Bereich, in der die reale Umgebung der virtuellen Umgebung überwiegt. Der Fokus der Wahrnehmung liegt entsprechend auf der Realität. Die Realität wird jedoch durch virtuelle Inhalte erweitert und angereichert.

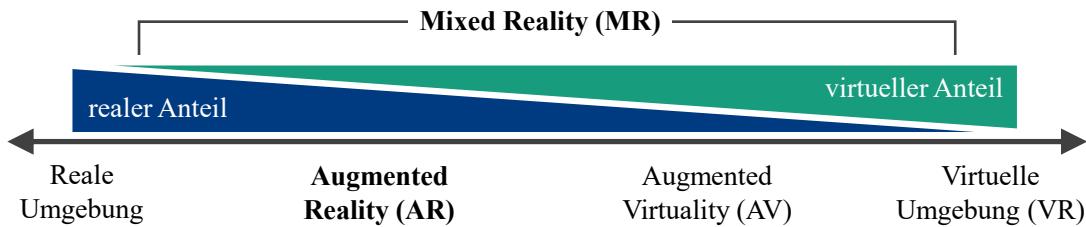


Abbildung 3-10: *Virtuality Continuum in Anlehnung an MILGRAM UND KISHINO [MK94]*

Durch die Kombination der realen und der virtuellen Welt kann der Benutzer der AR-Technologie sowohl mit realen Objekten als auch mit virtuellen Inhalten interagieren. Ebenso ist eine Interaktion zwischen der realen und der virtuellen Welt möglich. In diesem Zusammenhang ist ein wesentlicher Aspekt von AR die Immersion [FR18]. **Immersion** beschreibt den Grad des Effekts, dass virtuelle Inhalte über verschiedene Stimuli als real empfunden werden. Umso höher der Immersionsgrad, umso stärker hat der AR-Benutzer das Gefühl, die virtuellen Inhalte seien real und Teil seiner Umgebung [bit21]. Ein weiterer in diesem Zusammenhang relevanter Begriff ist **Präsenz**. Präsenz meint das Gefühl der Anwesenheit in einer virtuellen Umgebung. Dabei geht es um das Erlebnis als Ganzes. Die Ausprägung der Präsenz ist daher auch abhängig von der Gestaltung und Umsetzung der AR-Umgebung, der Funktionalität und vor allem auch der Interaktion. Je nach technischer Umsetzung und Gestaltung der Anwendung sind mittels AR Erlebnisse mit unterschiedlichen Graden der Immersion und Präsenz möglich.

Grundsätzlich mit AR **mögliche Funktionalitäten** werden unter anderem von BROLL beschrieben [Bro19]. BROLL benennt beispielsweise einen Röntgenblick und virtuelle Löcher als AR-spezifische Möglichkeiten im Bereich der Visualisierung. Der Röntgenblick entspricht dem virtuellen Hineinschauen in ein reales Objekt, in das man ohne AR nicht hineinschauen könnte. Virtuelle Löcher erzeugen den visuellen Eindruck, dass ein Teil eines realen Objekts entfernt oder herausgeschnitten wurde. Im Bereich der Interaktion beschreibt BROLL unter anderem die Selektion durch die Blickrichtung des Nutzers sowie Tangible User Interfaces als Möglichkeiten von AR. Bei Tangible User Interfaces werden reale Objekte in die Interaktion eingebunden und erzeugen einen haptischen Effekt. Neben AR-spezifischen Funktionalitäten bietet AR zudem grundsätzliche Potentiale digitaler Kommunikation und Visualisierung, beispielsweise Kollaboration und Sprachkommunikation.

Mit einer **AR-Lösung** sind alle Teilespekte gemeint, die für die Realisierung eines AR-Erlebnisses notwendig sind. Ziel einer AR-Lösung ist stets die Überlagerung der Realität mit virtuellen Inhalten. Die virtuellen Inhalte werden in einer sogenannten **AR-Szene** beschrieben, die neben den eigentlichen Inhalten auch weitere virtuelle Objekte wie Lichtquellen, Kameras oder Audioquellen enthält [JV19]. Zur Wiedergabe der virtuellen Inhalte im Rahmen der AR-Lösung dient das **AR-System**. Das AR-System beinhaltet geeignete Hardware und Software, um die reale Umgebung in Echtzeit wahrzunehmen und die AR-Szene passend und interaktiv für den Benutzer in ihr wiederzugeben [DBJ+19].

Zu einem AR-System gehört entsprechend ein AR-Endgerät als zentrale Hardware für den Benutzer und zugehörige Software. Die prinzipielle, vereinfachte Struktur eines AR-Systems wird von vielen Autoren ähnlich beschrieben [Azu97, Alt02, Sch08a] (Abbildung 3-11). So verfügt ein AR-System in der Regel über vier wesentliche Hardware- und Software-Komponenten, die über das AR-Endgerät und die Software realisiert werden: 1) Tracking-System, 2) Datenbank-System, 3) Szenengenerator, 4) Anzeigesystem. Diese Komponenten dienen der Erfüllung der laut BROLL fünf notwendigen Schritte zur Realisierung von AR [Bro19]. Die fünf Schritte werden im Folgenden genauer erläutert.

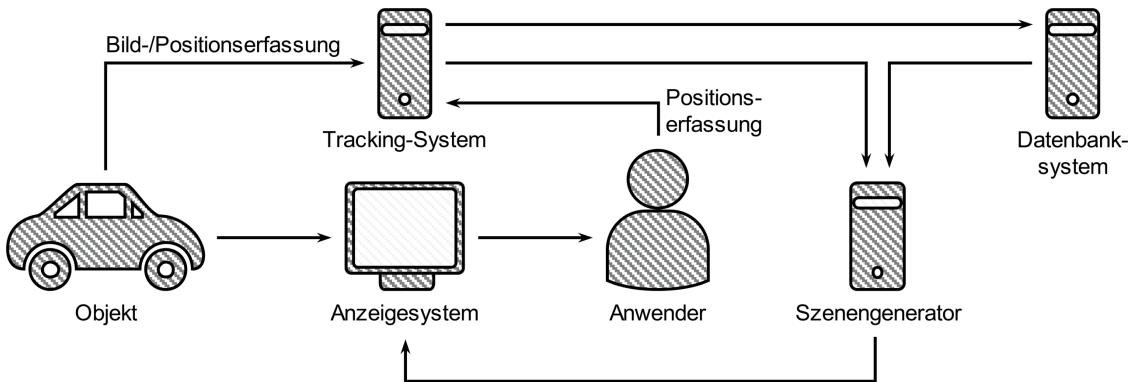


Abbildung 3-11: Vereinfachte Struktur eines AR-Systems [Rei21, S. 99]

Videoaufnahme: Über eine oder mehrere Videokameras als Teil des Tracking-Systems wird die Umgebung des Betrachters kontinuierlich wahrgenommen. Dabei ist das Tracking-System meist Teil des AR-Endgeräts. Der resultierende Videostream dient zum einen dem Tracking (siehe nächster Schritt) und zum anderen je nach technischer Lösung auch später der Wiedergabe der AR-Szene (siehe Ausgabe). Bei neueren AR-Geräten können neben Standard-Kameras auch Tiefenkameras verbaut sein, die dreidimensionale Bilder der Umgebung liefern.

Tracking: Im Kontext AR wird unter Tracking die Bestimmung der Position und der Orientierung des AR-Endgeräts verstanden [Bro19]. Das Tracking-System dient entsprechend der Bewegungsverfolgung. Ziel ist, die Perspektive des Benutzers zu jeder Zeit möglichst genau zu definieren, um die AR-Szene entsprechend korrekt mit der Umgebung kombinieren und passend überlagern zu können. Das Tracking basiert meist auf dem Videostream, aus dem durch spezielle Algorithmen die Bewegung der Kamera und somit des AR-Endgeräts berechnet werden kann [GBH+19].

Registrierung: Bei der Registrierung wird auch von Positionierung der virtuellen Inhalte gesprochen. Anhand der Tracking-Informationen werden die Koordinatensysteme der beobachteten realen Umgebung und der AR-Szene in Relation gesetzt, sodass die virtuellen Inhalte passend in der Realität verankert sind. Bewegt sich der Benutzer in der realen Welt, erscheinen die virtuellen Inhalte fest verankert (registriert) und perspektivisch korrekt an den entsprechenden Positionen [Bro19]. Es wird unterschieden zwischen Marker-basierten und Marker-losen Ansätzen. Bei Marker-basierten Ansätzen wird die AR-Szene

relativ zu einem im Vorhinein festgelegten Marker positioniert. Dieser Marker wird anschließend physisch in der realen Umgebung positioniert. Heutzutage sind sowohl zweidimensionale Grafiken oder Fotos als auch dreidimensionale Objekte als Marker möglich. Die Marker werden von der AR-Software im Videostream erkannt und die relative Position und Orientierung zum AR-Gerät berechnet. Somit kann die bereits im Vorhinein positionierte AR-Szene passend zum Marker und zum Benutzer wiedergegeben werden. Bei Marker-basierten Registrierungs-Ansätzen wird oftmals auch von Marker-basiertem Tracking gesprochen, da die Marker nicht nur die Positionierung der AR-Szene ermöglichen, sondern gleichzeitig auch das Tracking unterstützen. Bei Marker-losen Ansätzen werden räumliche Informationen aus den Videostreams und weiterer Sensorik gewonnen und die AR-Szene während der Anwendung dynamisch in der realen Umgebung positioniert. Dazu werden beispielsweise SLAM-Algorithmen (Simultaneous Localization and Mapping) eingesetzt, die parallel zur Bewegungsverfolgung direkt eine zwei- oder dreidimensionale Karte der Umgebung aufbauen [GBH+19].

Darstellung: Die Darstellung erfolgt im Szenen-Generator und nutzt den Input des Tracking-Systems und bei Bedarf des Datenbanksystems (vgl. Abbildung 3-11). Aus dem Tracking und der Registrierung sind die Position und die Orientierung des AR-Geräts sowie die gewünschte Position und Lage der AR-Szene in der realen Umgebung bekannt. Daraus ergibt sich die für die Darstellung relevante Perspektive des Benutzers auf die AR-Szene. Entsprechend wird die AR-Szene in der korrekten Perspektive gerendert. Das Rendering bildet die Grundlage für die Ausgabe.

Ausgabe: Im letzten Schritt wird das Rendering der AR-Szene über das Anzeigesystem des AR-Systems ausgegeben. Das Anzeigesystem realisiert dabei die Überlagerung der realen Umgebung mit der AR-Szene, sodass der Anwender eine Kombination der beiden Welten als Augmented Reality wahrnimmt. Bei den Anzeigesystemen werden drei wesentliche technische Ansätze unterschieden [Bro19, S. 320]:

- ***Video See-Through (VST)***: Bei VST-Systemen wird für die Ausgabe der im ersten Schritt aufgenommene Videostream verwendet. Die AR-Szene wird perspektivisch korrekt im Videobild überlagert und das kombinierte Videobild über ein Display ausgegeben. Der VST-Ansatz wird vor allem bei handgehaltenen Geräten wie Smartphones und Tablets eingesetzt. Abbildung 3-12 zeigt ein Beispiel, bei dem ein reales Objekt mittels VST auf einem Tablet visuell um weitere Objekte überlagert wird.
- ***Optical See-Through (OST)***: Der OST-Ansatz nutzt semitransparente Anzeigesysteme, die nur dort etwas darstellen, wo virtuelle Inhalte zu sehen sein sollen. OST wird vor allem bei kopfgetragenen AR-Brillen eingesetzt (Englisch: Head-Mounted Display, kurz HMD). Jedem Auge wird über ein eigenes Display die AR-Szene präsentiert. Durch zwei perspektivisch leicht verschobene Bilder wird ein stereoskopischer Effekt erzeugt. Der Nutzer nimmt die AR-Szene dreidimensional und eingebettet in die reale Umgebung wahr. Der Grad der Immersion und der Präsenz ist bei

binokularen AR-Brillen entsprechend höher als bei Tablets mit VST. Abbildung 3-12 deutet auf der rechten Seite ein Beispiel für OST auf einer binokularen AR-Brille an.

- **Projektionsbasiert:** Bei projektionsbasierter AR wird die AR-Szene durch einen Projektor als Anzeigegerät in die reale Umgebung projiziert. Durch diesen Ansatz können entsprechend keine dreidimensionalen Effekte erzielt werden. Der Fokus liegt eher auf der Darstellung veränderter Materialien oder Strukturen oder der Einblendung von Zusatzinformationen wie Erläuterungen oder Anleitungen.

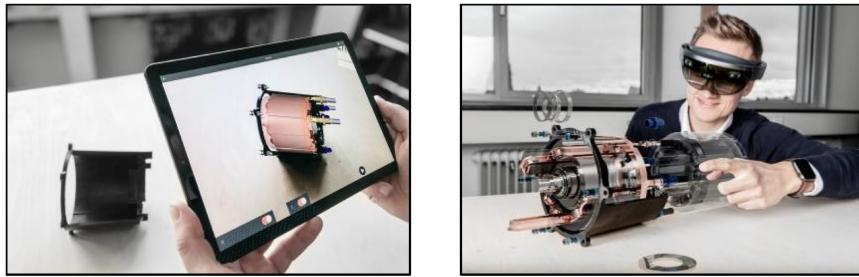


Abbildung 3-12: Beispiele für VST mit Tablet und OST mit AR-Brille [Rei21, S. 101]

Die einzelnen Ansätze bieten unterschiedliche Möglichkeiten der Darstellung von AR-Szenen zur Erweiterung und Anreicherung der realen Umgebung. Zu den einzelnen Ansätzen existieren zudem verschiedenste AR-Endgeräte, die sich in ihren Eigenschaften und Funktionalitäten teilweise stark unterscheiden. Für jeden Anwendungsfall ist daher individuell zu prüfen, welcher Ansatz und welches AR-Endgerät am besten geeignet ist.

Ein über die fünf grundlegenden Schritte zur Realisierung von AR hinausgehender Aspekt ist die **Interaktion**. Im Kontext von AR ist mit Interaktion ein Informationsaustausch zwischen dem menschlichen Nutzer und dem AR-System gemeint. Im Kern der Interaktion steht vor allem die Steuerung des Systems durch den Nutzer. Beim Einsatz von VST-basierten AR-Endgeräten wie Tablets und Smartphones können Standard-Touch-Funktionalitäten auf dem Display zur Interaktion verwendet werden. Für die Interaktion in der dreidimensionalen AR-Szene sind laut DÖRNER ET AL. zudem Menüs, 3D-Widgets, Tangibles, Sprachkommandos und Gesten zur Systemsteuerung weit verbreitet [DGO+19, S. 223]. 3D-Widgets entsprechen 3D-Objekten, die mit Funktionalitäten belegt sind. Tangibles sind reale Objekte, die in die Interaktion einbezogen und mit Aktionen verknüpft werden können.

Durch die Entwicklung sogenannter Smart Glasses wie der Google Glass ist neben Augmented Reality zudem der Begriff **Assisted Reality** geprägt worden [bit21]. Bei Assisted Reality werden dem Benutzer Inhalte auf einem einzelnen Display vor einem Auge im Sichtfeld dargestellt. Es erfolgt meist kein Tracking sowie keine Registrierung. Entsprechend wird die AR-Szene nicht dreidimensional mit der realen Szene kombiniert. Assisted Reality-Brillen verfügen jedoch teilweise über transparente Displays und ermöglichen somit die Wahrnehmung der realen Umgebung bei gleichzeitiger Darstellung und Betrachtung virtueller Inhalte. Somit passt Assisted Reality nicht zu den zuvor genannten Definitionen von AR nach AZUMA oder DÖRNER ET AL.. Assisted Reality lässt sich aber

weit links im Virtualitäts-Kontinuum nach MILGRAM UND KISHINO als schwache Form von AR einordnen (vgl. Abbildung 3-10). Entsprechend wird Assisted Reality im weiteren Verlauf der Arbeit unter dem Begriff AR eingeschlossen.

AR kann als eine Brücke zwischen der realen physischen Welt und den im Kontext technischer Systeme zunehmend parallel entstehenden digitalen Welten gesehen werden [ME19]. So ergeben sich durch den Einsatz von AR **Potentiale entlang des gesamten Lebenszyklus** eines technischen Systems. Bei händischen Tätigkeiten können beispielsweise Informationen und Instruktionen situationsgerecht im Sichtfeld des Benutzers bereitgestellt werden. So werden Werker bei komplexen Montageprozessen in der Produktion unterstützt [KNG19, S. 186]. AR führt dadurch zu einer Effizienz- und Produktivitätssteigerung. Im Handel ermöglichen AR-basierte interaktive Visualisierungen ein realitätsnahes Erleben des späteren Produkts und die Veranschaulichung komplexer Eigenschaften und Funktionalitäten, die sonst nicht ersichtlich wären [PH17]. Gleichzeitig führen virtuelle Produktpräsentationen zu Materialeinsparungen. Über bidirektionale Kommunikation zwischen einem System und dem Benutzer können Systeme zudem AR-basiert gesteuert werden, was zusätzliche AR-Service-Angebote zur Abhebung vom Wettbewerb ermöglicht. AR-Technologien können zudem ein effizienteres Zusammenarbeiten auch über Distanzen hinweg ermöglichen [VDC19-ol, GDE+18, S. 384]. Dies führt zu enormen Zeit- und Kosteneinsparungen und zu Vorteilen im Betrieb und der Wartung [Hau20-ol, Por18-ol]. So kann AR eine entscheidende Rolle bei der Nutzung der Potentiale der Industrie 4.0 einnehmen [Dav15, LNO17]. Auf das konkrete Potential von AR zur Unterstützung der Validierung in der Produktentwicklung wird im folgenden Kapitel genauer eingegangen.

3.4.2 Augmented Reality im Kontext Validierung

In Abschnitt 3.3.2 wurde bereits beschrieben, dass digitale Prototypen grundsätzlich Vorteile gegenüber physischen Prototypen bieten. Die Validierung von subjektiv zu bewertenden Merkmalen, beispielsweise anhand von Inspektionen, ist jedoch auch mit digitalen Prototypen herausfordernd. Die Individualisierung und die Komplexität und Interdisziplinarität der Systeme erfordern eine stärkere Berücksichtigung der Kunden und Nutzer sowie der Umwelt der Systeme bei der Validierung. Dazu ist eine Validierung in realen Einsatzszenarien und mit direkter Integration des Kunden notwendig. Die aktuell verwendeten Validierungsansätze kommen hier an ihre Grenzen [GDE+18]. AR dagegen hat das Potential, genau diese Herausforderungen im Kontext der Validierung zu adressieren und digitale Prototypen mit realen Umgebungen und dem Menschen selbst zu kombinieren.

Ziel der Validierung ist die Reduzierung von Unsicherheiten (s. Abschnitt 3.3.1) und somit der Differenz zwischen dem für die Entwicklung benötigten und dem vorhandenen Wissen. Mittels Validierung soll entsprechend Produktwissen möglichst schnell erarbeitet werden. Beim **Einsatz virtueller Techniken**, zu denen AR gehört, wird vollständiges Produktwissen früher erreicht als bei der klassischen Entwicklung [Ovt05] [GEK01]. So

kann Erarbeitung von Produktwissen im Produktentwicklungsprozess durch **AR zur Präsentation digitaler Prototypen** unterstützt werden. AR ermöglicht die interaktive, dreidimensionale Visualisierung der Prototypen eingebettet in realen Umgebungen. So können immersive Erfahrungen für den Kunden geschaffen werden [AKN+19]. Das Produkt bzw. der Prototyp kann in einer realen Umgebung, bspw. seinem zukünftigen Einsatzort, und in Echtgröße wahrgenommen werden. Der Kunde als Nutzer von AR kann sich dreidimensional um den Prototyp herumbewegen und mit ihm interagieren. Gegenüber klassischen Visualisierungstechniken können Produkte mittels AR somit erlebt anstatt nur betrachtet werden [RFH+19].

Die dreidimensionale Darstellung in der realen Umgebung hat spezielle **Vorteile für die Validierung**. Die Realität ist grundsätzlich dreidimensional. Verfügbare digitale Daten wie Prototypen sind jedoch bisher meist noch an zweidimensionale Wiedergabeformate und Bildschirme gebunden. Dies führt zu einer kognitiven Distanz zwischen der Form der Informationsbereitstellung und dem Kontext, in dem die Informationen angewendet werden sollen [PH17]. Entsprechend ist eine mentale Überführung der zweidimensional wahrgenommenen Informationen in den dreidimensionalen Kontext notwendig. Dieser Prozess benötigt mentale Kapazitäten und führt zu kognitiver Belastung. Gleichzeitig reduziert die Überführung in den dreidimensionalen Kontext die Kapazitäten für andere mentale Aktivitäten [PH17]. Zu diesen Aktivitäten gehören im Kontext der Validierung beispielsweise die Analyse und Bewertung von Produktmerkmalen. Durch die dreidimensionale Darstellung reduziert der Einsatz virtueller Technologien die kognitive Belastung zur Wahrnehmung von digitalen Prototypen [LLM14]. Die zusätzliche Integration der dreidimensionalen Darstellung in der realen Umgebung mittels AR reduziert die kognitive Belastung zusätzlich. Die relevanten Informationen können gegenüber klassischen Darstellungsformen schneller und exakter aufgenommen und verarbeitet werden. Der Kunde kann sich auf die Analyse und Bewertung der Produktmerkmale konzentrieren.

Grundsätzlich stellt AR ein vielversprechendes **Werkzeug zur Präsentation und Vermittlung von Produkteigenschaften** dar. Dies wird unter anderem durch die Unternehmensbefragung bestätigt (s. Anhang A1). Produktmerkmale können mittels AR anschaulich und interessant präsentiert werden [GDE+18, S. 110]. Laut PORTER UND HEPPELMANN ermöglicht AR somit eine bessere Validierung und Optimierung der Entwicklung [PH17]. Durch die grundlegenden und spezifischen Funktionalitäten bietet sich AR vor allem in solchen Situationen an, in denen komplexe Sachverhalte dargestellt und vermittelt werden müssen [Sch08a]. AR ist zudem sinnvoll, wenn eine große Bauteilanzahl oder Variantenvielfalt vorliegt.

Bei der Validierung steht allerdings auch beim Einsatz von AR nicht zwangsläufig das System als Ganzes im Fokus. Vielmehr werden laut der Unternehmensbefragung Potentiale von **AR zur Validierung einzelner Produktmerkmale** anhand spezifischer virtueller Prototypen gesehen. REINEMANN hat im Rahmen seiner Dissertation die Wiedergabe von drei unterschiedlichen Repräsentationsformen bezüglich verschiedener Wie-

dergabedimensionen bewerten lassen (Abbildung 3-13), wobei die Wiedergabedimensionen Produktmerkmalen entsprechen. Als Repräsentationsformen wurden eine Papierskizze, ein CAD-Modell und ein über AR präsentiertes 3D-Modell (AR-Modell) verglichen. Die Testpersonen haben die Wiedergabetreue der Repräsentationsformen jeweils für die verschiedenen Wiedergabedimensionen von Eins bis Fünf bewertet. Ums so höher die Wiedergabetreue, umso besser wird die Wiedergabedimension bzw. das Produktmerkmal wiedergegeben und umso besser kann es verstanden und validiert werden.

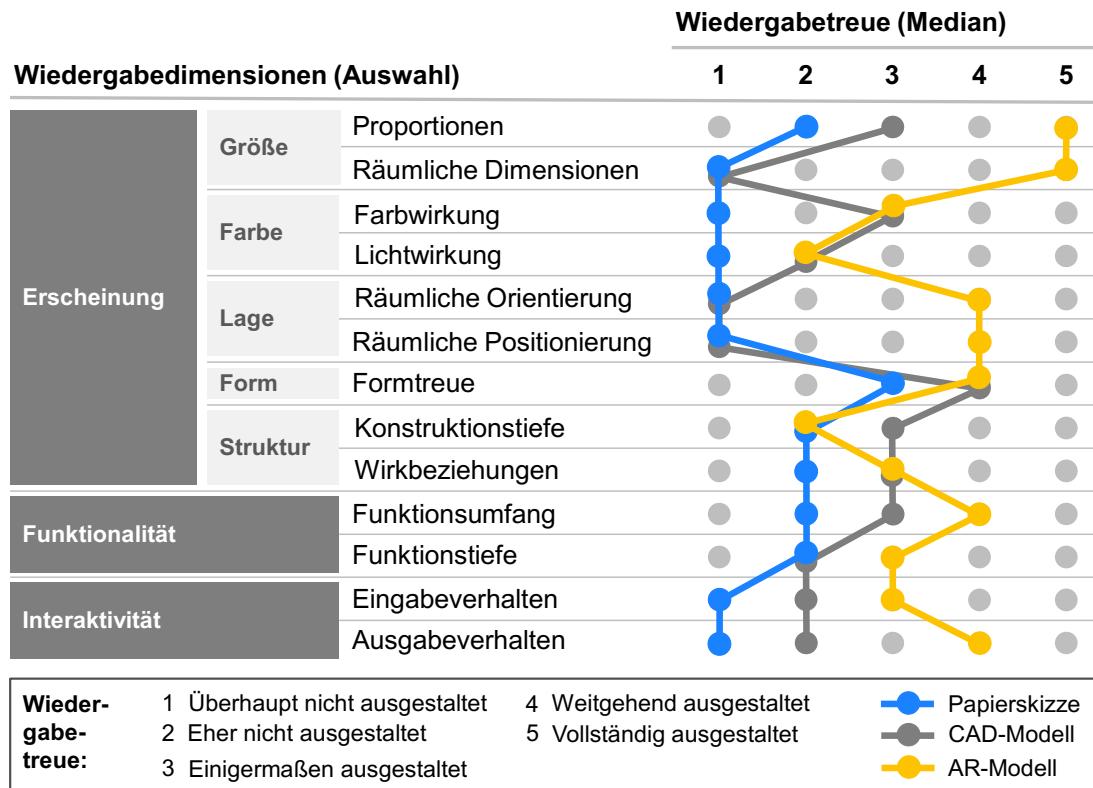


Abbildung 3-13: Bewertung der Wiedergabetreue verschiedener Repräsentationsformen eines Quadrocopters ($n = 68$) (Grafik in Anlehnung an [RFH+19])

Die **Bewertung** zeigt, dass AR bei vielen Produktmerkmalen besser geeignet ist, als die Papierskizze und auch als das CAD-Modell [RFH+19]. Besonders gut eignet sich AR laut der Bewertung für die Wiedergabe der *Größe*. Dies resultiert aus der Möglichkeit immer siverer Technologien, 3D-Modelle in Echtgröße zu präsentieren. Tests in Virtual Reality (VR) haben gezeigt, dass Größe und Proportionen problemlos wahrgenommen und verstanden werden können, unabhängig von der Komplexität der dargestellten Modelle [HŠM+19, S. 1928]. Da die dreidimensionale Visualisierung an sich vergleichbar ist bei VR und AR, lassen sich diese Erkenntnisse auch auf AR übertragen. Die Bewertung von REINEMANN zeigt weitere große Potentiale von AR für die Wiedergabe der *Lage* und der *Positionierung und Orientierung im Raum*. Die im Rahmen dieser Arbeit durchgeführte Unternehmensbefragung bestätigt die Einschätzung, dass beispielsweise Abmaße oder Einbausituationen eines Systems sehr viel besser mit AR direkt in den Räumlichkeiten überprüft werden können (vgl. Anhang A1). Ein Beispiel, in dem diese Potentiale bereits

Verwendung finden, ist die Produktvisualisierungs-App Place von IKEA [Ste19-ol]. Mit der Place-App können Kunden Möbelstücke aus dem IKEA-Katalog direkt über ihr Smartphone mittels AR in ihrer eigenen Wohnung anschauen und die optische Wirkung im Raum sowie den Platzbedarf überprüfen (Abbildung 3-14, links). Somit ermöglicht die App eine bessere Entscheidung gegenüber einer klassischen Darstellung im Katalog oder Online-Shop. Eine weitere Wiedergabedimensionen, in der AR besonders gut bewertet wurde, ist die *Form*. Zudem zeigt die im Rahmen dieser Arbeit durchgeführte Unternehmensbefragung besondere Potentiale von AR für die Validierung der *Optik* und des *Designs*.

Bei der Validierung der *Funktionalität* sind sich die Befragten uneinig bzgl. der Potentiale von AR. Die umfangreiche Bewertung nach REINEMANN schreibt AR jedoch **Vorteile gegenüber den anderen Repräsentationsformen** zu [RFH+19]. So können bspw. komplexe Vorgänge und Zusammenhänge sehr gut mit AR dargestellt werden [Sch08a]. Potentiale der virtuellen und immersiven Präsentation von Produktmerkmalen bezüglich der Vermittlung von Funktionalitäten werden bestätigt durch JASPER [Jas15] und BACKHAUS ET AL. [BGS+14]. Bei der *Struktur* zeigt sich laut der Bewertung nach REINEMANN kein großer Vorteil von AR gegenüber den anderen Repräsentationsformen. Allerdings fördert eine virtuelle Repräsentation von 3D-Modellen in Kombination mit interaktiven Visualisierungsoptionen das Verständnis von Systemstrukturen und -zusammenhängen [FSC18]. Eine Studie von BERG UND VANCE hat gezeigt, dass der Einsatz von VR die Wahrnehmung räumlicher und struktureller Zusammenhänge von Systemkomponenten verbessert [BV16]. In einer Studie von VAN GOETHEM ET AL. wird unter anderem zudem die kontextuelle Wahrnehmung als Vorteil von AR gegenüber Bildschirm-basierten Ansätzen hervorgehoben [vVG+20] (s. Anhang A2.3.4). Auch bei der Wiedergabe der grundsätzlich eher schwierig wiederzugebenden *Interaktivität* werden deutliche Vorteile von AR gegenüber den anderen Repräsentationsformen gesehen (s. Abbildung 3-13) [RFH+19].

Eine über klassische Repräsentationsformen weit hinausgehende Möglichkeit von AR ist die **immersive Visualisierung in realen Umgebungen**. AR ermöglicht eine Kombination aus virtuellen Prototypen und realen Produktkomponenten und Umgebungsobjekten [ARF+19, S. 430]. Mit AR können somit auch einzelne Produktaspekte auf bereits bestehenden Komponenten überlagert werden. AR ermöglicht eine Art Röntgenblick ins Innere von Produkten und somit die Visualisierung von neuen Eigenschaften und Funktionalitäten, die ansonsten nicht ersichtlich wären [PH17]. So wird beispielsweise die Validierung von Erweiterungen oder Aktualisierungen eines bestehenden Systems ermöglicht. Zudem können mittels AR komplexe Simulationsergebnisse oder technische Daten dreidimensional und bei Bedarf direkt am realen System visualisiert werden. Dies vereinfacht die Interpretation, Analyse und Fehlererkennung in räumlicher und logischer Hinsicht [LNO17]. Die Einbettung in reale Situationen ist von besonderem Nutzen im Kontext der zunehmend vernetzten und oft im Systemverbund funktionierenden Advanced

Systems. Räumliche Abhängigkeiten, Schnittstellen und Kommunikations- sowie Materialflussaspekte können durch virtuelle Einbettung in den zukünftigen realen Einsatzort einfacher abgesichert werden. Siemens ermöglicht bspw. eine AR-basierte Visualisierung mittels Tablets, um Maschinen und Anlagen in einer physischen Umgebung erlebbar zu machen und validieren zu können [Sie19-ol] (Abbildung 3-14, rechts).



Abbildung 3-14: Beispiele AR-basierter Validierung: IKEA Place [Ste19-ol] (links), Validierung technischer Systeme bei Siemens [Sie19-ol] (rechts)

Im Kontext der Interaktivität ist ein besonderer Vorteil von AR zudem die direkte **Integration des Menschen** in die Validierung. Dieser Aspekt wird auch in der Umfrage von REINEMANN deutlich. 86% der 66 Befragten stimmen zu, dass AR eine direkte Einbindung des Kunden in die Validierung ermöglicht [Rei21, S. 139]. Bei geeigneter technischer Realisierung kann der Benutzer die virtuellen Prototypen immersiv erleben und mit ihnen sowie mit der realen Umgebung interagieren. Dies schafft einen Mehrwert gegenüber klassischen Simulationen, bei denen der Benutzer lediglich digital repräsentiert und simuliert wird [Gar20-ol]. AR als Validierungswerkzeug ermöglicht somit neben der rein visuellen Absicherung eine Überprüfung von Interaktionsaspekten anhand der direkten Interaktion mit den virtuellen oder auch kombinierten Prototypen durch den Benutzer bzw. Kunden [ARF+19, S. 430]. So können Aspekte der *Ergonomie*, unter anderem Erreichbarkeiten und Bewegungsabläufe, an virtuellen Prototypen realitätsnah validiert werden [ERZ14, S. 185f.]. Durch die Einbettung in der realen Umgebung und die direkte Integration des Menschen lassen sich mit AR zudem *Einsehbarkeiten* und *Verdeckungen*, beispielsweise von großen Maschinen, effizient und aussagekräftig überprüfen.

Beim Einsatz mobiler Geräte kann AR grundsätzlich **ortsunabhängig** eingesetzt werden. Der Benutzer ist nicht an einen festen Arbeitsplatz gebunden, sondern kann die Visualisierung an beliebigen Orten in der Realität platzieren. So kann ein virtueller Prototyp bedarfsgerecht im Büro, in der Maschinenhalle oder auch zu Hause oder in der Natur erlebt werden. Als digitale Kommunikationstechnologie ermöglicht AR zudem eine **standortübergreifende Kommunikation**. So kann die in der Produktentwicklung erforderliche Kommunikation und Kooperation unterschiedlicher Stakeholder über Distanzen hinweg durch AR unterstützt werden. Dabei kann mittels AR eine Zusammenarbeit erfolgen, die über Telefon- und Videokonferenzen hinausgeht [GDE+18, S. 384].

Durch Übertragung des Kamerabilds des AR-Endgeräts kann der Kommunikationspartner in die Perspektive des AR-Benutzers eintauchen. So kann der Entwickler aus der Distanz die Validierung durch den Kunden verfolgen, Eigenschaften erklären und Fragen beantworten. AR ermöglicht entsprechend eine effiziente Validierung vor Ort beim Kunden trotz räumlicher Trennung der Akteure.

Neben den grundsätzlichen Potentialen der AR-basierten Visualisierung und Kommunikation ermöglicht AR **zusätzliche Funktionalitäten**, die die Validierung unterstützen. Zum einen können Validierungsaktivitäten direkt digital dokumentiert und Kundenfeedback dem Entwickler zur Verfügung gestellt werden. Zudem bietet AR verschiedene Möglichkeiten der digitalen Telemetrie. So können die räumliche Bewegung um den Prototypen und auch die Interaktion mit ihm digital verfolgt, analysiert und dokumentiert werden. Unter Umständen lässt dies Rückschlüsse auf die Erfüllung von Kundenanforderungen ziehen. Zudem kann durch zusätzliche oder teilweise integrierte Sensorik die Blickrichtung und auch die Augenbewegung des Benutzers verfolgt und analysiert werden. Entsprechende Eye Tracking-Ansätze ermöglichen unter anderem Aussagen über die Wirkung von Designaspekten, was bspw. beim Verpackungsdesign, bei Werbeelementen oder in der Automobilbranche Anwendung findet [HNA+11]. So bietet AR auch in diesem Kontext Vorteile gegenüber klassischen Validierungsansätzen und -werkzeugen.

Zusammenfassend lässt sich an dieser Stelle festhalten, dass AR Kunden ein frühes Erleben von Produkten noch während der Entwicklung ermöglicht. Kunden können besonders aussagekräftiges Feedback geben, das zur zielgerichteten Gestaltung des geplanten Produkts in die Entwicklung zurückgespielt wird [RC16]. Dabei stellt der Einsatz von AR bei der Validierung einen guten Kompromiss zwischen dem Ressourceneinsatz und dem Produkterlebnis speziell in den frühen Phasen der Entwicklung dar [Ped17]. **AR als Validierungswerkzeug ist ein vielversprechender Ansatz**, um den Herausforderungen der Validierung heutiger Systeme zu begegnen [ARF+19]. Die Potentiale von AR im Kontext der Validierung sind jedoch nur wenigen Unternehmen bekannt. Zudem stellen vor allem die Aufwände zur Realisierung von AR-Lösungen sowie die grundlegende Komplexität der Technologie Herausforderungen für Unternehmen bei der Nutzung der Potentiale dar. Dies wird unter anderem durch die im Rahmen dieser Arbeit durchgeführte Unternehmensbefragung bestätigt (vgl. Anhang A1).

3.4.3 Komplexität und Herausforderungen der Technologie

Laut dem **Gartner Hype Cycle** ist AR bereits über die erste Hype-Phase hinweg [Pan19-ol]. Viele Unternehmen und Investoren haben sich auf die Technologie gestürzt und mit großem Enthusiasmus versucht, die Potentiale zu nutzen. Dies gelang allerdings nur bedingt. Viele Lösungen gingen nicht über einen Prototypen-Status oder Proof-of-Concepts hinaus [KKS17]. Ebenso sind einige AR-Geräte-Hersteller aufgrund mangelnder technologischer Reife und damit verbundener fehlender Nutzerakzeptanz bereits wieder vom

Markt verschwunden [Hau20-ol]. Technologie-Hersteller und Anwender-Unternehmen beginnen sich nun nachhaltiger mit der Technologie und deren erfolgversprechenden Einsatzmöglichkeiten zu befassen [Ska19-ol, Cap18-ol]. AR-Geräte wurden zur Marktreife weiterentwickelt und technologische Fortschritte zu einzelnen Funktionalitäten wie Tracking und Interaktion erzielt.

Die Einschätzung von Gartner zeigt, dass AR aus technologischer Sicht schon sehr weit entwickelt und grundsätzlich für den produktiven Praxiseinsatz geeignet ist. Zudem ist die Technologie an sich Unternehmen weitestgehend bekannt [Rei21]. Dennoch hat die jüngste Forschung gezeigt, dass die Umsetzung für die **industrielle Anwendung eine Herausforderung für Unternehmen** darstellt [EM20]. So lag die Bundesrepublik beim Einsatz von AR in der Industrie zuletzt im internationalen Vergleich deutlich hinter den beiden führenden Nationen USA und China sowie Frankreich [Por18-ol]. Neben teilweise hohen Kosten stellt vor allem die grundlegende Komplexität der Technologie eine kritische Eintrittsbarriere für Unternehmen dar, AR einzusetzen [SPB+20]. Angesichts vielfältiger zu beachtender Aspekte und Herausforderungen ist Technologie-Knowhow im Bereich AR eine Grundvoraussetzung, um die Technologie gezielt und erfolgreich zu nutzen. Gerade mittelständische Unternehmen verfügen jedoch oftmals über unzureichende Ressourcen, um die erforderliche Expertise aufzubauen und an den Potentialen zu partizipieren [KMF+19]. Selbst größere Unternehmen oder Konzerne haben bisher aufgrund mangelnder Erfahrung und Fachkenntnisse gezögert, in eigene Pilotprogramme oder -anwendungen zu investieren [ARC18-ol]. Bevor AR praktisch eingesetzt werden kann, müssen unter anderem der Einsatz von AR strategisch geplant, die technische Umsetzung zielgerichtet konfiguriert und AR-Anwendungssoftware bedarfsgerecht entwickelt werden. Diese drei Aspekte stellen drei wesentliche Herausforderungen für Unternehmen dar. Sie werden im Folgenden genauer betrachtet.

Unternehmen verfügen teilweise über ein Grundverständnis der Technologie und haben eine grobe Vorstellung der bekannten Potentiale. Die Potentiale sowie der damit verbundene Nutzen und Aufwand sind jedoch abhängig von unternehmensspezifischen Prozessen, Arbeitsabläufen und Rahmenbedingungen. Bei der **Planung des Einsatzes von AR** sind Potentiale individuell zu analysieren und der mögliche Nutzen den notwendigen Aufwänden und Kosten gegenüberzustellen. Aufgrund mangelnden Knowhows und fehlender Erfahrung sehen sich Unternehmen dazu jedoch meist nicht in der Lage. Es fehlt das umfassende Verständnis der Möglichkeiten der Technologie zur Visualisierung und Kommunikation. Unternehmen können nicht einschätzen, wie sich Augmented Reality und damit verbundene neue Interaktionsmöglichkeiten auf das Nutzererlebnis bei der Validierung auswirken können [DH17, S. 187]. So können sich Entwickler und Designer oft nicht vorstellen, wie virtuelle Technologien wie AR effektiv in den Entwicklungsprozess integriert werden können [HNC+19, S. 1906]. Neben der Bewertung des Nutzens ist zu überprüfen, ob mögliche Einsparungen über den Aufwänden für die Hardware und den Betrieb von AR liegen [Sch08a]. Dabei ist stets auch zu berücksichtigen, welche Daten und Informationen zur Erschließung des Nutzens benötigt werden. Für die AR-basierte

Validierung ist zu definieren, welche Produktmerkmale in welchem Detailgrad mittels eines virtuellen Prototyps repräsentiert werden sollen. Je nachdem, ob die benötigten Daten bereits vorliegen oder zusätzlich erarbeitet werden müssen, steigt der Aufwand zur Vorbereitung der Validierung. Bei einer Entscheidung für den Einsatz von AR sollten notwendige Aktivitäten zur Vorbereitung von virtuellen Prototypen als Grundlage der Validierung frühzeitig geplant werden, um relevante Daten rechtzeitig verfügbar zu haben. Ein weiterer bei der Planung zu berücksichtigender Aspekt ist die Akzeptanz der Mitarbeiter und Kunden bzgl. neuer Trends und Technologien. So sollten AR-Lösungen auf individueller und organisatorischer Ebene zum Unternehmen selbst und zum Anwender passen [DAG+21].

Nach der Planung des Einsatzes von AR ist die **Konfiguration der AR-Lösung** notwendig. Basis jeder AR-Lösung ist das AR-System bestehend aus AR-Gerät und AR-Anwendungssoftware. Dank des technologischen Fortschritts der jüngsten Zeit ist eine Vielzahl an AR-Geräten am Markt verfügbar, deren funktionale und technische Ausprägungen sich jedoch teilweise stark unterscheiden. Meist wird bei AR an Datenbrillen gedacht, dabei kann je nach Anwendungsfall ein Smartphone oder Tablet das besser geeignete Gerät sein [BH19-ol]. Zu beachtende Aspekte sind unter anderem der Tragekomfort, die Qualität und Farbeigenschaften sowie die Transparenz der Visualisierung oder auch die Tracking-Genauigkeit [ARF+19, S. 438]. Allein bei den Tracking-Systemen existiert eine Vielzahl an mechanischen, inertialen, magnetischen, akustischen, optischen oder elektromagnetischen Ansätzen, die sich in ihrer Qualität und ihren Einsatzmöglichkeiten unterscheiden [Alt02]. Weitere Unterschiede betreffen beispielsweise die Interaktionsmöglichkeiten, die Mobilität und Einsatzdauer sowie Schnittstellen der AR-Endgeräte. Unterschiedliche Anwendungsfälle erfordern unterschiedliche AR-Funktionalitäten oder unterschiedliche Qualitäten der Funktionalitäten. Auch im Kontext der AR-basierten Validierung benötigt die Überprüfung einzelner Produktmerkmale unterschiedliche Funktionalitäten des AR-Systems. Für eine realitätsnahe optische Überprüfung des Designs eines Produkts ist beispielsweise eine sehr gute Auflösung der Visualisierung mit farbgetreuer Darstellung sinnvoll. Die Überprüfung von ergonomischen Aspekten bedarf dagegen freier Hände des Benutzers und einer immersiven Darstellung. Für entsprechende Validierungsaktivitäten sind daher kopfgetragene binokulare AR-Brillen notwendig. Neben den anwendungsbezogenen Anforderungen können zudem weitere Einflüsse wie Umgebungsbedingungen oder unternehmensspezifische Anforderungen und Regularien technische Eigenschaften erfordern oder ausschließen. Beispielsweise kann helle Sonneneinstrahlung die Visualisierung über OST-Displays beeinträchtigen oder eine hohe Umgebungslautstärke eine Sprachsteuerung verhindern. Grundsätzlich ist für den optimalen Einsatz von AR aufgrund der vielfältigen technischen Möglichkeiten und der verschiedenen Einflussfaktoren eine zielgerichtete Auswahl der Funktionalitäten und Eigenschaften des AR-Endgeräts notwendig [ME19]. Dies gilt im Speziellen für den Einsatz von AR als Validierungswerkzeug. Das Erlebnis von virtuellen Prototypen und damit eine aussagekräftige Validierung sind stark abhängig vom eingesetzten AR-Gerät und der Darstellungsqualität [ARF+19]. Zudem wird die Wahrnehmung einzelner Produktmerkmale

durch die Funktionalitäten der AR-Anwendungssoftware beeinflusst und gefördert. Somit ist eine gezielte Konfiguration des AR-Systems entscheidend für eine erfolgreiche Validierung. Aufgrund des Mangels an Knowhow und Erfahrung sind Unternehmen jedoch meist nicht in der Lage, geeignete Hardware und notwendige Software-Funktionalitäten bedarfsgerecht zu identifizieren [HNC+19, S. 1906]. So zeigt die von REINEMANN durchgeführte Unternehmensbefragung, dass die Unterstützung der Planung und Konfiguration wesentlich wichtiger sind als eine Unterstützung der Durchführung und Auswertung der AR-basierten Validierung [Rei21, S. 154].

Im Anschluss an die Konfiguration des AR-Systems wird eine konkrete technische AR-Lösung benötigt. Es gibt bereits eine Vielzahl an AR-Anwendungssoftware am Markt, die von Unternehmen gekauft oder lizenziert werden kann. Diese Software-Lösungen adressieren allerdings nur einzelne Anwendungsfälle und decken längst nicht alle AR-Potentiale ab [ME19]. Zudem besteht teilweise eine Bindung an konkrete Geräte oder Betriebssysteme, was die Flexibilität einschränkt und den Einsatz für Unternehmen aufgrund interner Richtlinien und Restriktionen erschwert. Grundsätzlich sollte AR-Anwendungssoftware für den erfolgreichen Einsatz maßgeschneidert werden [ARC18-ol]. Aufgrund der zuvor genannten Vielfalt an AR-Endgeräten sowie individuellen technischen und funktionalen Anforderungen, speziell im Kontext der Validierung, ist meist eine **individuelle Entwicklung der AR-Anwendungssoftware** notwendig. Die Entwicklung von AR-Anwendungssoftware basiert weitestgehend auf Software Development Kits (SDK), die gewisse grundlegende Software-Funktionalitäten bereitstellen [KMF+19] (s. Abschnitt 4.3.1.2). Zum einen bieten Hersteller von AR-Endgeräten gerätespezifische SDKs an, die speziell die technischen Möglichkeiten der eigenen Geräte adressieren und für die Entwicklung für diese Geräte notwendig sind. Zum anderen gibt es eine Vielzahl weiterer Anbieter-unabhängiger SDKs, die für AR und Kommunikation relevante Funktionalitäten bereitstellen [CWC+19]. All diese SDKs fokussieren jedoch technische Details beziehungsweise grundlegende Funktionalitäten [KMF+19]. Sie bieten keine fertigen Anwendungsfunktionalitäten und bedürfen stets einer individuellen Programmierung, um einsatzbereite AR-Anwendungssoftware zu entwickeln. Die Programmierung erfordert umfangreiches Knowhow und Erfahrung, unter anderem bezüglich der eingesetzten Technik sowie der eingesetzten SDKs [ARC18-ol]. Ein besonderer Fokus sollte stets auf der benutzeroptimierten Gestaltung der AR-Lösung liegen [DGO+19, S. 220]. Interaktive Lösungen, die sowohl dem Benutzer als auch dem Unternehmen signifikanten Nutzen bringen, sind am anspruchsvollsten zu entwickeln [PH17]. Spezielle Herausforderungen ergeben sich durch die Heterogenität an AR-Endgeräten. Die Entwicklung einer immersiven AR-Anwendung für eine binokulare AR-Brille unterscheidet sich teilweise stark von der Entwicklung einer Smartphone-AR-Anwendung. Durch fehlende Standards kann bisher keine ausreichende Kompatibilität zwischen verschiedenen Anwendungsfällen und AR-Endgeräten gewährleistet werden, was die Entwicklung verkompliziert und verlangsamt [Bil18-ol]. Grundsätzlich werden daher bisher hochqualifizierte Experten für die Entwicklung von AR-Anwendungen benötigt [KMF+19]. Aufgrund des demografi-

schen Wandels herrscht speziell im Bereich der Softwareentwicklung jedoch ein Fachkräftemangel [DAG+21]. So werden meist externe Dienstleister zur Entwicklung von AR-Lösungen beauftragt. Zum einen sorgt dies für hohe Entwicklungskosten für die Unternehmen, sowohl bei der initialen Entwicklung als auch bei späteren Anpassungen und Aktualisierungen [PH17]. Zum anderen birgt eine Entwicklung durch Dritte die Gefahr, dass aufgrund mangelnder Einbindung der relevanten Stakeholder die Anforderungen der Industrie nicht erfüllt werden [KMF+19]. Zur Erschließung der Potentiale von AR im Kontext der Validierung, müssen Unternehmen daher in der Lage sein, bedarfsgerechte AR-Anwendungsssoftware für ausgewählte AR-Endgeräte eigenständig zu entwickeln und zu verwalten. Nur dann können die hohen Anforderungen an Effizienz und Kosten einsparungen im Kontext der Validierung erfüllt werden. Allerdings sind es insbesondere der noch hohe Entwicklungsaufwand und die damit verbundene notwendige Expertise, die es Unternehmen erschweren, an den Potentialen von AR zu partizipieren. Zur Nutzung der Potentiale von AR in der Industrie muss die Entwicklung von AR-Anwendungen daher schneller, einfacher und billiger für Unternehmen werden [KMF+19].

Nach der Planung, Konfiguration und Entwicklung einer AR-Lösung, werden in den meisten AR-Anwendungsfällen gewisse Daten benötigt. Daher ist bei AR eine reibungslose **Integration von Daten** in die Anwendungen essentiell [Por18-ol]. Im Fall der AR-basierten Validierung entsprechen diese Daten den virtuellen Repräsentationen der Produktmerkmale in Form von Prototypen. Die Datenintegration ist aktuell jedoch mit hohem Zeit- und Kostenaufwand verbunden [Boy17-ol]. Grundsätzlich bestehen in Unternehmen eine zunehmende Anzahl und Heterogenität an Daten und Modellen, die in unterschiedlichen Software-Tools und Plattform-Umgebungen erstellt und dokumentiert werden. Unternehmen mangelt es an standardisierten Austauschformaten und Datenstrukturen sowie IT-Plattformen zur aufwandsarmen, zentralen Nutzung der Daten [DAG+21]. Für AR fehlt bisher ein einheitliches Datenprotokoll, wodurch die Etablierung von AR im alltäglichen Leben verlangsamt wird [PT20]. Speziell die Integration von CAD-Daten stellt ein großes Problem für Unternehmen dar, insbesondere außerhalb der Software-Ökosysteme der CAD-Systemanbieter [LSR+15]. CAD-Daten müssen von parametrischen Modellen anhand einer Tesselierung in AR-fähige Polygon-Modelle überführt werden. Dabei werden teilweise komplexe Modelle vereinfacht und reduziert, um sie in Echtzeit visualisieren und simulieren zu können [ERZ14, S. 178]. Dieser Aufbereitungsprozess kann durch verschiedene Tools wie beispielsweise PiXYZ unterstützt und teilautomatisiert werden [Els20]. Der Prozess ist dennoch komplex und bedarf gewisser Erfahrung in der Parametrisierung. Es existieren Formate, die sowohl parametrische CAD-Modelle als auch visualisierbare Polygon-Modelle enthalten. Ein Beispiel ist das als ISO-Standard definierte JT-Format [KK13]. Mittels verschiedener kostenpflichtiger Schnittstellen-Tools ist das Format kompatibel zu CAD-Software und kann mittels AR visualisiert werden. Grundsätzlich stellt der Prozess der Datenaufbereitung und -integration weiterhin eine Herausforderung dar. So beschäftigen sich verschiedene Studien und Forschungsvorhaben mit bestehenden Ansätzen und der Erarbeitung von standardisierten Vorgehensweisen unter Einsatz dieser Tools [HLL+19, UVV19].

Die geschilderte Situation sowie die damit verbundenen **Herausforderungen werden durch Unternehmen bestätigt**. Während 88% von 66 Befragten angeben, AR als Technologie zu kennen, haben 86% AR noch nicht zur Validierung eingesetzt (Abbildung 3-15) [Rei21]. Ein wesentlicher Grund dafür ist der Mangel an methodischer Unterstützung zum Einsatz von AR in der Produktentwicklung [ARF+19]. Dies gilt im Speziellen für die weitestgehend individuell zu gestaltende Validierung [ABK+16]. Laut der im Rahmen dieser Arbeit durchgeführten Befragung sind Unternehmen sich teilweise unsicher, welche Auswirkungen der Einsatz von AR auf das jetzige Vorgehen in der Produktentwicklung hat. Als zusätzliche Aktivitäten werden vor allem die frühzeitige Planung der Validierungsaktivitäten, die verstärkte Kommunikation mit den Kunden sowie die Anfertigung von Dokumenten und Modellen als Grundlage der Validierung gesehen. Zudem sehen einige Unternehmen aufgrund mangelnder Expertise die Integration eines AR-Experten in die Validierung als notwendig an. Als Herausforderungen nennen Unternehmen unter anderem die gezielte Potentialidentifikation inklusive der Abschätzung des Kosten-Nutzen-Verhältnisses und die Entwicklung der Anwendungen (vgl. Anhang A1). Zur Adressierung der Herausforderungen und zur Erschließung der Potentiale von AR in der Validierung wünschen sich Produktentwickler methodische Unterstützung [ARF+19].

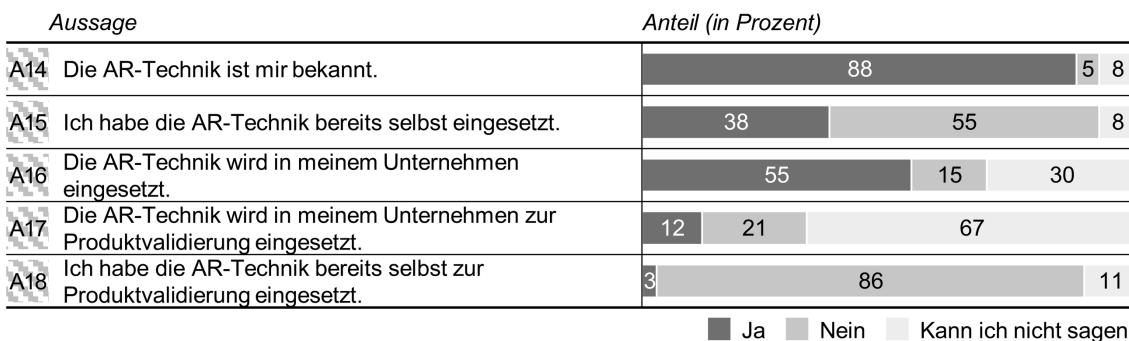


Abbildung 3-15. Umfrageergebnisse zu Erfahrungen mit dem Einsatz von AR (n=66) [Rei21, S. 138]

3.5 Problemabgrenzung

Die Problemanalyse zeigt, dass Produkte vermehrt individuell für Kunden entwickelt werden. Daher ist die Orientierung am Kundennutzen und die frühe Einbeziehung des Kunden ein wichtiger Bestandteil heutiger Entwicklungsprojekte. Dies gilt speziell im Kontext von Advanced Systems. Advanced Systems weisen aufgrund des Zusammenspiels verschiedener Disziplinen und der Integration innovativer Technologien eine hohe Komplexität auf. Sie vereinen zudem vielfältige für den Kunden relevante Produktmerkmale. Dies erschwert die Erfüllung individueller Anforderungen an das System. Kunden, sowohl intern als auch extern, müssen regelmäßig in die Entwicklung heutiger Systeme einbezogen werden und Feedback zum Entwicklungsstand geben. Dabei gilt es speziell, subjektiv zu bewertende Aspekte, die nicht durch Simulationen validiert werden können, durch interne und externe Kunden überprüfen zu lassen. Nur so können den Kundenanforderungen entsprechend qualitativ hochwertige Produkte in konkurrenzfähigen kurzen

Zyklen entwickelt werden. Effiziente Validierungsaktivitäten sind somit von entscheidender Bedeutung für eine erfolgreiche Produktentwicklung, unabhängig von den jeweiligen Kundenentkopplungspunkten. Zum einen wird die kundenintegrierte Validierung jedoch durch eine zunehmende geografische Trennung von Entwicklern und Kunden sowie flexible Arbeitsplätze und -zeiten erschwert. Zum anderen kommen klassische Validierungsansätze, bspw. basierend auf physischen Prototypen oder technischen Dokumenten, mittlerweile aufgrund der steigenden Produktkomplexität schnell an ihre Grenzen.

Wie die Problemanalyse zeigt, können diese Herausforderungen mit der innovativen Visualisierungs- und Kommunikationstechnologie Augmented Reality adressiert werden. AR ermöglicht die interaktive Visualisierung virtueller Inhalte eingebettet in realen Umgebungen. Zur Validierung in der Produktentwicklung ergeben sich dadurch folgende übergeordnete **Nutzenpotentiale**:

- **Dreidimensionale Visualisierung in realen Umgebungen:** Mittels AR können ganze Produkte oder einzelne Produktmerkmale anhand digitaler Produktdaten und -modelle präsentiert werden. Das kann frei im Raum oder präzise und maßstabsgetreu am zukünftigen Einsatzort in der Realität sein. Ohne die Notwendigkeit physischer Prototypen können so auch Produktmerkmale, die einen Bezug zur Realität haben oder die Integration des Kunden selbst erfordern, abgesichert werden. Zudem ermöglicht AR eine dreidimensionale Visualisierung. Eine Überführung zweidimensionaler Darstellungen in den dreidimensionalen Kontext ist im Vergleich zu klassischen digitalen Validierungsansätzen nicht mehr notwendig. Die kognitive Belastung wird reduziert, was zu einem besseren Verständnis der Entwicklung beim Kunden führt. Die bestehenden Einschränkungen digitaler Prototypen können somit durch AR stark verringert werden.
- **Interaktivität:** Mit AR kann der Kunde sich im Rahmen der technischen Möglichkeiten frei im Raum und um die virtuellen Prototypen herumbewegen. So können reale Situationen durchgespielt und für die Validierung relevante Perspektiven eingegommen werden. Zudem können ganze Systeme oder einzelne Produktmerkmale mittels AR interaktiv visualisiert werden. Der Kunde kann mit den dargestellten Inhalten interagieren und sie manipulieren. So wird auch die Überprüfung von funktionalen und ergonomischen Produktmerkmalen ermöglicht. Zudem können komplexe Produkteigenschaften und -funktionalitäten anhand von Animationen und zusätzlichen Visualisierungen veranschaulicht und erklärt werden. Das führt zu einem besseren Verständnis des Produkts gegenüber statischen Produktpräsentationen.
- **Vernetzung:** Neben der Visualisierung ermöglicht AR eine effiziente Vernetzung räumlich verteilter Akteure. Digitale Produktinformationen und -modelle können über das Internet zugänglich gemacht werden. So haben entfernte Kunden direkten Zugriff auf den aktuellen Entwicklungsstand und können ihn über AR erleben. Zudem ermöglicht AR das visuelle Eintauchen in die Situation des Kommunikations-

partners. Durch die gleichzeitige interaktive Visualisierung der digitalen Informationen im Sichtfeld des Kunden wird eine bessere Interaktion und Kommunikation gegenüber Telefon- und Videokonferenzen ermöglicht. Entwickler und Kunden können so auch standortübergreifend effiziente Validierungsaktivitäten gemeinsam durchführen.

Zusammenfassend lässt sich festhalten, dass durch den Einsatz von AR als Validierungswerkzeug die Vorteile von physischen und digitalen Prototypen vereint werden können. Digitale Daten sind meist früh verfügbar und können aufwandsarm auch standortübergreifend bereitgestellt werden. Durch die interaktive Visualisierung mittels AR können Produktmerkmale validiert werden, die bisher lediglich anhand physischer Produktmodelle abgesichert werden konnten. Selbst komplexe Systeme können mit AR früh erlebt werden und Kunden dem Entwickler sehr viel aussagekräftigeres Feedback geben. Somit kann der Einsatz von AR zu einer Effizienzsteigerung in der Produktentwicklung bei gleichzeitig verbesserter Produktqualität entsprechend der Kundenanforderungen führen. Die technologische Komplexität von AR stellt viele Unternehmen jedoch vor grundsätzliche **Herausforderungen** bei der Erschließung der Potentiale [ME19, EM20], speziell im Kontext der Produktvalidierung:

- **Ungewissheit über Potentiale:** Die frühe und kontinuierliche Einbeziehung des Kunden in die Entwicklung zur effizienten Validierung ist ein wichtiger Bestandteil heutiger Entwicklungsprojekte. Dabei ist der Einsatz von Methoden und Werkzeugen sowie die Vorgehensweise zur Validierung individuell zu gestalten. In einzelnen Domänen haben sich verschiedene Validierungswerkzeuge etabliert. Der Einsatz von AR zur Unterstützung der Produktvalidierung wurde dagegen noch nicht ausreichend erforscht. Die verschiedenen stark vom Entwicklungsprojekt abhängigen Potentiale sind Unternehmen daher meist nicht bekannt. Aufgrund des Mangels an Erfahrung und Technologie-Knowhow fällt es vielen Unternehmen somit schwer, an den Potentialen von AR im Kontext der Produktvalidierung zu partizipieren.
- **Komplexe Konfiguration und Entwicklung von AR-Systemen:** AR-Systeme sollten für jeden Anwendungsfall individuell konfiguriert werden, um einen größtmöglichen Nutzen zu erzielen. Zum einen gilt es, ein für den Anwendungsfall geeignetes AR-Endgerät auszuwählen. Zum anderen müssen notwendige Funktionalitäten der Anwendungssoftware individuell definiert werden. Aufgrund der großen Gerätetypen-Vielfalt am Markt und den unterschiedlichen Funktionalitäten und Eigenschaften der Geräte ist Erfahrung und technisches Wissen für die anwendungsfallspezifische Gerät-Auswahl notwendig. Gleiches gilt für die funktionale Konzipierung der Anwendungssoftware. Die Anwendungssoftware muss zudem individuell entwickelt werden, was mit großem Aufwand verbunden ist und Erfahrung bedarf. Unternehmen sind daher meist nicht in der Lage, AR-Systeme für einen konkreten Anwendungsfall bedarfsgerecht zu konfigurieren und zu entwickeln.

- **Mangel an methodischer Unterstützung:** Im Kontext der Produktvalidierung fällt es Unternehmen aufgrund mangelnder methodischer Unterstützung schwer, die Potentiale von AR zu erkennen und zu nutzen [ARF+19, S. 429]. Speziell bei der Identifikation individueller Potentiale sowie der zielgerichteten Konfiguration und Entwicklung der notwendigen AR-Systeme bedürfen Unternehmen Unterstützung. Aufgrund fehlender Ressourcen und mangelnder Erfahrung sind Unternehmen nicht in der Lage, die zusätzlich notwendigen Aktivitäten zur AR-basierten Validierung eigenständig zu definieren und im Rahmen eines Entwicklungsprojekts durchzuführen.

Eine Gegenüberstellung der Nutzenpotentiale und der Herausforderungen verdeutlicht den Bedarf an einem *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*. Aus den Herausforderungen ergeben sich die vier in Abbildung 3-16 dargestellten Handlungsfelder (HF).

HF1: Systematische Planung AR-basierter Validierung

Es bedarf der Unterstützung von Unternehmen bei der Identifikation und Bewertung von Potentialen und dem zielgerichteten Einsatz von AR als Validierungswerkzeug. Zunächst müssen **generische AR-Potentiale** zur Validierung verschiedener Produktmerkmale erarbeitet und dokumentiert werden. Anschließend gilt es, ein Hilfsmittel zur individuellen Bewertung der Potentiale zu entwickeln. So sollen Unternehmen in die Lage versetzt werden, ausgehend von den Anforderungen an das zu entwickelnde System selbstständig zu bestimmen, ob der Einsatz von AR sinnvoll ist und einen Mehrwert bietet. Dabei soll die Planung unabhängig von der Art des Produkts, der Branche, der Kunden (bspw. B2B, B2C oder intern) sowie des Kundenentkopplungspunkts unterstützt werden. Zudem sind grundlegende **Anforderungen** an die Aufbereitung und Integration der zu validierenden Daten zu definieren, um Unternehmen beim bedarfsgerechten Einsatz von AR als Validierungswerkzeug zu unterstützen.

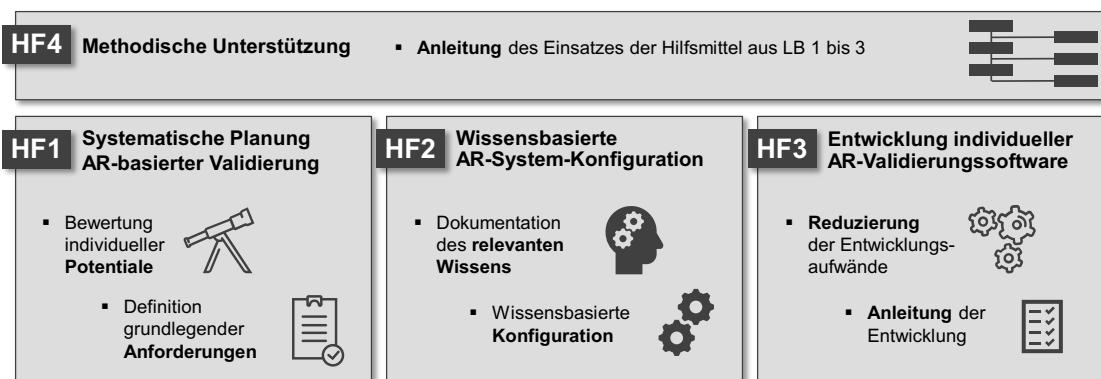


Abbildung 3-16: *Handlungsfelder (HF) des Instrumentariums*

HF2: Wissensbasierte AR-System-Konfiguration

Die bedarfsgerechte Konfiguration des AR-Systems ist ausschlaggebend für die erfolgreiche Nutzung von AR in konkreten Anwendungsfällen. Zum einen liegen dazu notwendiges Wissen und Erfahrung explizit in Form technischer Dokumente sowie implizit bei

AR-Technologie-Experten und -Programmierern vor. Zum anderen ergeben sich Erkenntnisse aus AR-Systemen im praktischen Einsatz, prototypischen AR-Systemen und Proof-of-Concepts. Um Unternehmen ohne spezielles AR-Knowhow bei der individuellen Konfiguration des AR-Systems zu unterstützen, muss **relevantes Wissen** zunächst gesammelt und anschließend dokumentiert werden. Neben Wissen zu technischen Lösungen und Anwendungen sind dabei auch mögliche Einflussfaktoren aus dem Umfeld zu berücksichtigen. Ausgehend von den identifizierten Potentialen und den grundlegenden Anforderungen aus Handlungsfeld 1 muss Unternehmen eine **wissensbasierte Konfiguration** bedarfsgerechter AR-System ermöglicht werden.

HF3: Entwicklung individueller AR-Validierungsanwendungen

Zum aktuellen Zeitpunkt existiert keine Software, die kollaborative interaktive Produktvalidierung mittels AR ermöglicht und alle Potentiale und Anforderungen adressiert. Zur Durchführung von AR-basierten Validierungsaktivitäten ist jeweils eine individuelle AR-Anwendung notwendig, deren Entwicklung AR-Knowhow benötigt und mit umfangreicher Programmierung verbunden ist. Die Anwendung muss Funktionalitäten zur bedarfsgerechten Präsentation der virtuellen Prototypen bereitstellen. Zudem muss dem Kunden die Möglichkeit gegeben werden, Feedback zu den durch die Prototypen repräsentierten Produktmerkmalen zu geben. Bei der Programmierung müssen für unterschiedliche AR-Endgeräte spezifische SDKs eingesetzt werden, die unterschiedliche Schnittstellen und Funktionalitäten bieten. Der Einsatz der SDKs erfordert Erfahrung und Einarbeitung. Unternehmen bedürfen dementsprechend Unterstützung bei der anwendungsspezifischen Entwicklung individueller Validierungsanwendungen. Vor allem müssen die **Entwicklungsauwände reduziert** und eine einfachere und schnellere Entwicklung ermöglicht werden. Durch eine zusätzliche **Anleitung der Entwicklungsschritte** werden Unternehmen somit befähigt, bedarfsgerechte AR-Lösungen als Validierungswerkzeug eigenständig zu entwickeln. Dabei ist die Unterstützung verschiedener Endgeräte und Funktionalitäten notwendig, um eine vollumfängliche Abdeckung der Potentiale zu ermöglichen.

HF4: Methodische Unterstützung

Neben Expertise und Ressourcen mangelt es Unternehmen an methodischer Unterstützung im Kontext der Produktvalidierung [ARF+19, S. 429]. So sind Unternehmen meist nicht in der Lage, notwendige Aktivitäten zur Planung und Vorbereitung von AR-Lösungen als Validierungswerkzeug eigenständig zu definieren und im Rahmen eines Entwicklungsprojekts durchzuführen. Die Hilfsmittel aus den Handlungsfeldern 1 bis 3 bilden die Grundlage zur Unterstützung der systematischen Potentialidentifikation sowie der individuellen Konfiguration und Entwicklung bedarfsgerechter AR-Systeme. Durch eine übergeordnete methodische Unterstützung zum **Einsatz der Hilfsmittel** können Unternehmen ohne spezielles Technologie-Knowhow und -Erfahrung zur eigenständigen Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung befähigt werden.

3.6 Anforderungen an die Arbeit

Aus der Problemanalyse resultieren die folgenden Anforderungen an ein *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*.

Anforderungen an die systematische Planung AR-basierter Validierung

A1) Vollständigkeit: Bei heutigen Marktleistungen können eine Vielzahl unterschiedlicher Produktmerkmale relevant sein, an die wiederum unterschiedlichste zu validierende Anforderungen gestellt werden können. Daraus ergeben sich vielfältige Einsatzmöglichkeiten für AR als Validierungswerkzeug. Es bedarf der vollständigen Identifikation und Dokumentation dieser Potentiale im Kontext der Produktentwicklung, um eine vollständige Erschließung der Potentiale durch Unternehmen zu ermöglichen.

A2) Generalisierung: Die Potentiale müssen allgemeingültig und für verschiedene Systeme, Unternehmen und Branchen wiederverwendbar sein. Die Definition der Potentiale muss daher generisch und somit unabhängig von konkreten Anwendungsbeispielen erfolgen. Nur so können die Übertragbarkeit und Anwendung des Wissens ermöglicht werden.

A3) Korrektheit: Unternehmen ohne umfassendes AR-Knowhow sollen basierend auf den definierten AR-Potentialen und zugehörigen Anforderungen bedarfsgerechte AR-Systeme zur Unterstützung der Validierung entwickeln. Zum einen muss die Bewertung der AR-Potentiale daher wissenschaftlich fundiert und plausibel erfolgen. Zum anderen müssen die Anforderungen an die technische Umsetzung korrekt sein, um einen erfolgreichen Einsatz von AR zu gewährleisten.

Anforderungen an die wissensbasierte AR-System-Konfiguration

A4) Umfassende Dokumentation: Unternehmen sollen durch die wissensbasierte Konfiguration in die Lage versetzt werden, ohne Erfahrung und Technologie-Knowhow AR-Systeme bedarfsgerecht zu konfigurieren. Das dazu relevante Wissen über Anwendungsfälle, AR-Systeme sowie Umfeldeinflüsse muss in umfassender Form gesammelt und dokumentiert werden, um eine korrekte wissensbasierte Konfiguration zu ermöglichen.

A5) Erweiterbarkeit: Aktuell liegt bereits Wissen über AR-Systeme und Anwendungsfälle bei Experten und in der Literatur vor. Durch die Weiterentwicklung der technischen Systeme und die Anwendung der Technologien in immer neuen Anwendungsfällen wird zukünftig neues Wissen entstehen. Um dieses Wissen integrieren und nutzen zu können, muss eine entsprechende Erweiterbarkeit der Dokumentation gegeben sein.

A6) Zugänglichkeit: Das reine Dokumentieren des Wissens reicht nicht aus, um Unternehmen bei der Konfiguration von AR-Systemen zu unterstützen. Das Wissen muss für einen konkreten Anwendungsfall intelligent analysiert und geeignete AR-Geräte und notwendige Anwendungssoftwarefunktionalitäten individuell identifiziert werden. Dazu muss das Wissen effizient und ohne spezielles Knowhow zugänglich gemacht werden.

Anforderungen an die Entwicklung individueller AR-Validierungsanwendungen

A7) Aufwandsarme Entwicklung: Um AR nutzenstiftend als Validierungswerkzeug einzusetzen, sind individuelle AR-Anwendungen notwendig. Bisher ist die Entwicklung dieser Anwendungen mit großem Programmieraufwand und AR-Knowhow verbunden. Unternehmen müssen daher in die Lage versetzt werden, individuelle Validierungsanwendungen entsprechend der eingesetzten Geräte und der spezifischen Anforderungen eigenständig und aufwandsarm zu entwickeln.

Anforderungen an die methodische Unterstützung

A8) Systematische Vorgehensweise: Der erfolgreiche Einsatz von AR als Validierungswerkzeug in der Produktentwicklung bedarf einer systematischen Vorgehensweise zur Planung und Vorbereitung. Daher muss das Instrumentarium eine strukturierte Potentialidentifikation gewährleisten und die anwendungsfallspezifische Konfiguration und Entwicklung des AR-Systems unter Berücksichtigung unternehmensspezifischer Rahmenbedingungen unterstützen.

A9) Anwendbarkeit: Das Instrumentarium muss eine wirtschaftliche Anwendung und einfache Handhabung ermöglichen, auch ohne AR-Knowhow. Die Tätigkeiten müssen mit einem vertretbaren Aufwand durchgeführt werden können, sodass Aufwand und Nutzen stets in einem sinnvollen Verhältnis zueinanderstehen. Eine praktikable Ausführung des Instrumentariums ist essenziell für die Akzeptanz in der Produktentwicklung, bei der es verstärkt auf Effizienz und Schnelligkeit ankommt.

4 Stand der Technik

Dieses Kapitel gibt einen Überblick über den Stand der Technik. Einerseits werden dabei zu der vorliegenden Arbeit verwandte Ansätze adressiert und sich von ihnen abgegrenzt. Andererseits werden relevante Lösungsansätze beschrieben, die als Grundlage und Ideengeber zur Erarbeitung der Ergebnisse dieser Arbeit dienen können.

Die Problemanalyse hat gezeigt, dass bisher keine Lösung zur Planung des Einsatzes von AR als Validierungswerkzeug existiert. Vor diesem Hintergrund werden in Kapitel 4.1 zunächst Arbeiten über Potentiale von AR sowie Ansätze zur Klassifikation von Produktanforderungen und zur Identifikation und Bewertung von Potentialen untersucht. Zudem geht aus der Problemanalyse hervor, dass Technologie- und Erfahrungswissen zu AR-Lösungen bisher lediglich implizit bei Experten vorliegt oder durch individuelle technische Dokumentationen beschrieben wird. In Kapitel 4.2 werden daher Ansätze zur digitalen Repräsentation und Verarbeitung von Wissen sowie zur Konfiguration technischer Systeme untersucht. Im Kontext der Entwicklung von AR-Anwendungen hat sich gezeigt, dass es Unternehmen an Ressourcen und Knowhow mangelt, um individuelle Anwendungen selbstständig zu entwickeln. Aus diesem Grund werden in Kapitel 4.3 Ansätze zur aufwandsarmen Programmierung und zur Reduzierung der Entwicklungsaufwände analysiert. Komplettiert wird der Stand der Technik in Kapitel 4.4 durch eine Bewertung der Ansätze anhand der Anforderungen aus Kapitel 3.6. Die gewählte Reihenfolge der untersuchten Ansätze stellt dabei keine Bewertung dar.

Im Rahmen der Analyse des Stands der Technik wurden weitere Ansätze identifiziert, die als Grundlage zur Erarbeitung von Lösungen in den Handlungsfeldern herangezogen werden können. Diese Ansätze sowie weitere relevante Informationen werden in Anhang A2 als Ergänzung zum Stand der Technik beschrieben.

4.1 Systematische Planung AR-basierter Validierung

Die Lösung zur systematischen Planung AR-basierter Validierung soll allgemeingültig und generisch sein. Ausgehend von an das Produkt gestellten Anforderungen sollen Potentiale von AR für die Validierung korrekt und ganzheitlich hergeleitet werden können. Es bedarf daher einer geeigneten Verarbeitung der Anforderungen, die grundsätzlich beliebig definiert sein können. Entsprechend wird mit dem House of Quality in Abschnitt 4.1.1 zunächst ein Ansatz zur Analyse und Klassifikation von Produktanforderungen analysiert. Anschließend werden Ansätze zur systematischen Identifikation und Bewertung von Potentialen untersucht (Abschnitte 4.1.2, 4.1.3, 4.1.4 und 4.1.5).

4.1.1 Das House of Quality nach KLEIN

Das House of Quality nach KLEIN ist ein zentrales Arbeitsmittel im Quality Function Deployment (QFD) [Kle99]. QFD ist eine Methode zur Planung und Entwicklung eines

Produkts entsprechend der vom Kunden geforderten Qualitätseigenschaften [Rei06]. Ziel ist unter anderem die bestmögliche Erfüllung von Kundenanforderungen. Der QFD-prozess besteht aus den vier Phasen Produktdefinition, Produktdesign, Prozessplanung und Prozesskontrolle. In der ersten Phase, der Produktdefinitionsphase, kommt das House of Quality zum Einsatz (Abbildung 4-1). Durch den Einsatz des House of Quality sollen ausgehend von Kundenanforderungen die Produktidee spezifiziert und die Bedeutung einzelner Qualitätsmerkmale hergeleitet werden. Qualitätsmerkmale entsprechen nicht-funktionalen Anforderungen an ein Produkt [ERZ14, S. 57]. Für das in Abbildung 4-1 verwendete Beispiel eines PKW-Außenspiegels sind dies unter anderem das Design, das Motorgeräusch sowie die Steifigkeit. Die Qualitätsmerkmale werden im House of Quality den Kundenanforderungen in einer Matrix gegenübergestellt. Zunächst wird für jede Anforderung deren Bedeutung aus Kundensicht mit einem Wert zwischen 1 und 5 gewichtet. Dies entspricht der sogenannten Kundengewichtung. Kern des House of Quality ist anschließend die Bewertung der Korrelation zwischen den einzelnen Kundenanforderungen und den Qualitätsmerkmalen. Es wird jeweils bewertet, wie stark die Beziehung zwischen der Anforderung in der Zeile und dem Merkmal in der Spalte ist. Eine 0 bedeutet, es liegt keine Beziehung vor. Eine 9 stellt eine starke Beziehung dar. Dazwischen liegen mögliche (1) und mäßige Beziehungen (3). Die Beziehungswerte werden jeweils mit den Kundengewichtungen multipliziert. Im Beispiel in Abbildung 4-1 wurde die Anforderung „Soll leise sein“ mit 3 gewichtet und ihre Beziehung zum Qualitätsmerkmal Windgeräusch mit 9. Daraus ergibt sich ein gewichteter Beziehungswert von 27. Durch die spaltenweise Summierung der Beziehungswerte können Aussagen über die **Bedeutung der einzelnen Qualitätsmerkmale** getroffen werden. Umso stärker speziell die wichtigen Anforderungen mit hoher Gewichtung ein Qualitätsmerkmal betreffen, umso größer ist dessen Bedeutung. Im Beispiel hat das Qualitätsmerkmal Design mit einer Summe von 57 die höchste Bedeutung, gefolgt von der Verteilgeschwindigkeit und der Enteisbarkeit. Diese Qualitätsmerkmale stellen entsprechend aus Kundensicht die Kernaspekte für das Produkt dar und sollten bei der Entwicklung besondere Berücksichtigung erhalten [Rei06].

Neben der Korrelation zwischen Kundenanforderungen und Qualitätsmerkmalen können im House of Quality weitere Aspekte und Zusammenhänge bewertet werden. Über eine **Korrelationsmatrix** als Dach des House of Quality lassen sich negative und positive Abhängigkeiten zwischen den Qualitätsmerkmalen definieren. Zudem wird für jedes Qualitätsmerkmal ein **Schwierigkeitsgrad** definiert. Hier wird unterschieden zwischen unkritisch (1), kritisch (2) und sehr kritisch (3). Unter anderem ergänzt durch Bewertungen der Konkurrenzfähigkeit bzgl. der Kundenanforderungen sowie der technischen Bewertung der Qualitätsmerkmale stellt das House of Quality ein umfangreiches und vielseitiges Hilfsmittel zur Herleitung wichtiger Produktanforderungen dar.

Bewertung: Das House of Quality nach KLEIN stellt einen interessanten Ansatz zur Bewertung der Bedeutung einzelner Qualitätsmerkmale ausgehend von Produktanforderun-

gen dar. Der Ansatz lässt sich auf beliebige Kundenanforderungen anwenden und ist daher grundsätzlich als generisch zu betrachten. Eine Übertragung des Konzepts auf die vorliegende Problematik ist denkbar. Anstatt der Qualitätsmerkmale könnten generische Produktmerkmale definiert und den Kundenanforderungen gegenübergestellt werden. Mittels einer entsprechenden Bewertung der Beziehungen ließe sich eine Generalisierung der Anforderungen realisieren.

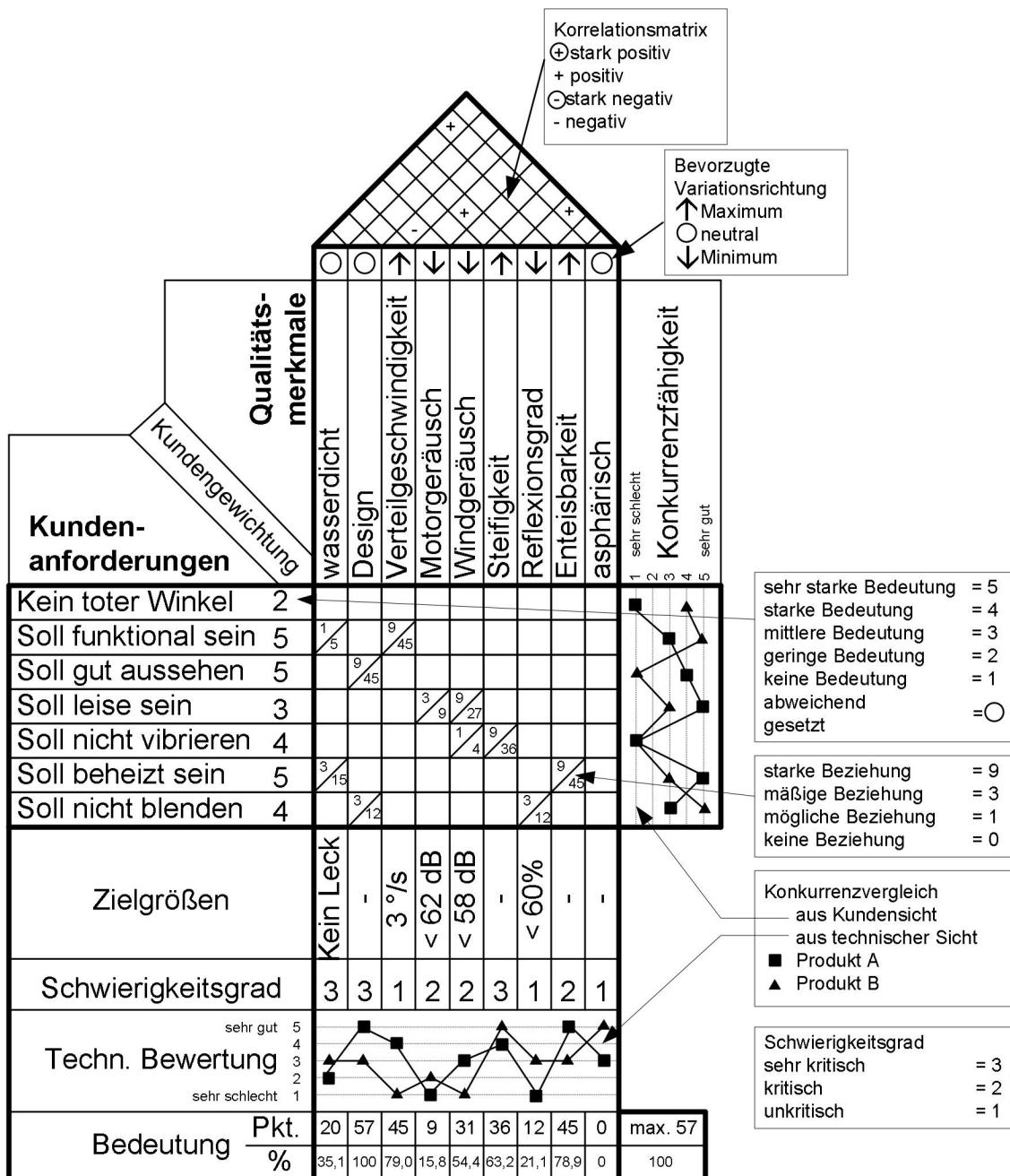


Abbildung 4-1: Das House of Quality nach KLEIN am Beispiel eines PKW-Außenspiegels [Kle99], Grafik aus [Rei06]

4.1.2 Leitfragen-basierte Augmented Reality-Potential-Bewertung nach PALMARINI ET AL.

PALMARINI ET AL. haben einen Leitfragen-basierten Prozess zur Bewertung von Potentiale von AR zur Unterstützung von Wartungsarbeiten entwickelt [PER17]. Die Entwicklung des Prozesses erfolgte in vier Schritten:

- 1) **Identifikation von relevanten Dokumenten:** Im Rahmen einer Literaturrecherche zu den Stichworten „AR“ oder „Augmented Reality“ in Kombination mit „Maintenance“ wurden 29 Veröffentlichungen identifiziert, die sich mit dem Einsatz von AR im Kontext Wartung beschäftigen.
- 2) **Erarbeitung von zusammenfassenden Tabellen je Veröffentlichung:** In Tabellen wurden die wesentlichen Inhalte je Veröffentlichung dokumentiert. Dabei wurden gezielt Informationen zu den vier Aspekten *Hardware*, *Entwicklungssoftware*, *Visualisierungsmethode* sowie *Tracking* extrahiert.
- 3) **Analyse der Tabellen:** Die Tabellen wurden miteinander verglichen und allgemeingültige Aussagen zu den vier Aspekten abgeleitet.
- 4) **Entwicklung des Prozesses:** Basierend auf der Analyse wurde der Prozess zur Bewertung der Potentiale von AR im Kontext Wartung erarbeitet.

Grundlage des Prozesses sind vier sogenannte **Fragenkataloge**. Abbildung 4-2 zeigt den Fragenkatalog zur generellen Einschätzung der Potentiale von AR zur Unterstützung von Wartungsarbeiten. Die drei weiteren Kataloge adressieren die Aspekte *Hardware*, *Entwicklungssoftware* und *Visualisierungsmethode*. Die Kataloge beinhalten Aussagen, für die bei der Anwendung des Prozesses bewertet wird, wie stark sie zutreffen. Eine Bewertung mit *1* entspricht einer absoluten Zustimmung und eine *10* bedeutet, die Aussage trifft gar nicht zu.

Questions	Score (1-10)
<ul style="list-style-type: none"> ▪ The task does not require concentration ▪ The duration or error rate variances are less than 10% among different operators ▪ Flexibility among operators is not required ▪ The system to be maintained is complex ▪ Live data is not required to perform the task ▪ The cost of maintenance is clear ▪ Maitainers do not follow a standard procedure ▪ The maintainer carries out the same operation daily/often ▪ The operation is not in a remote location ▪ The object to be maintained is subject to degradation 	

Abbildung 4-2: *Fragenkatalog zur generellen Einschätzung der Potentiale von AR zur Unterstützung von Wartungsarbeiten in Anlehnung an [PER17, S. 25]*

Für jeden der vier Kataloge wurde eine **Analyse-Skala** erarbeitet. Die Bewertungen aller Aussagen eines Katalogs werden gemittelt und mit der jeweiligen Skala abgeglichen. So

kann anhand der Skala zur Analyse der Eignung von AR (Abbildung 4-3, links) festgestellt werden, ob der Einsatz von AR zur Unterstützung der Wartungsarbeiten sinnvoll ist. Ein niedriger gemittelter Wert bedeutet, AR wird nicht empfohlen. Ein hoher Wert in Richtung 10 bedeutet dagegen, dass AR strengstens empfohlen wird. Im Fall dazwischenliegender Werte ist eine individuelle Abwägung der Potentiale erforderlich. Die zweite Skala adressiert die AR-Hardware. Über die gemittelte Bewertung des entsprechenden Fragenkatalogs kann über diese Skala festgestellt werden, welche Art von AR-Endgerät für den vorliegenden Wartungsfall vermeintlich passend wäre. Niedrige Werte deuten auf ein kopfgetragenes Gerät hin (HMD). Über Mikroprojektoren und kleine und große Tablets verändern sich die Empfehlungen in Richtung stationärer Systeme wie Displays und Projektoren für hohe Werte. Anhand der zwei weiteren Skalen können Entscheidungen bzgl. der Visualisierung und der Entwicklungsumgebung getroffen werden.

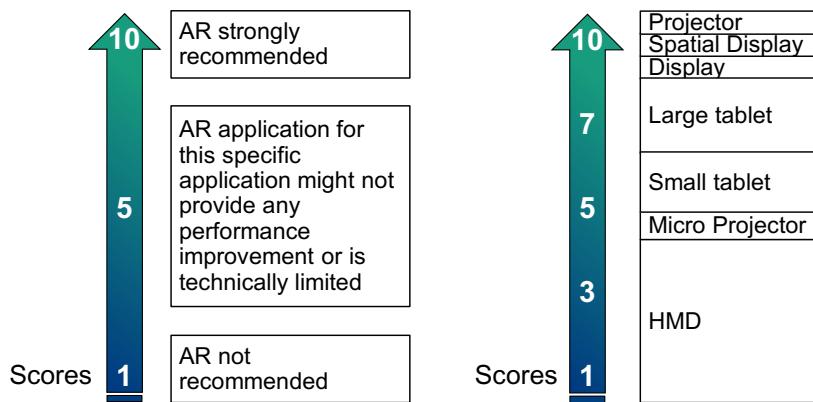


Abbildung 4-3: Skalen zur Analyse der Eignung von AR (links) sowie zur Auswahl eines geeigneten AR-Geräte-Typs (rechts) in Anlehnung an [PER17]

Bewertung: Die Leitfragen-basierte Bewertung von AR-Potentialen nach PALMARINI ET AL. fokussiert das Anwendungsszenario der Wartungsunterstützung und stellt daher keine generische Lösung dar. Zudem wird lediglich eine grobe Einschätzung anhand der Skalen ermöglicht. Einzelne über die Fragen bzw. Aussagen adressierte Aspekte werden durch Mittelung der Werte nur indirekt berücksichtigt. Durch diese Pauschalisierung könnten wichtige Aspekte und Rahmenbedingungen vernachlässigt werden. Der grundlegende Prozess zur Ableitung des Wissens und zur Erstellung der Fragenkataloge kann jedoch auf andere Szenarien und Kontexte übertragen werden. Zudem unterstützt der Ansatz neben der Bewertung der grundsätzlichen Potentiale von AR zur Wartungsunterstützung die AR-Systemkonfiguration. Dies geschieht allerdings nur auf dem Level der Gerätetypen sowie lediglich grob anhand der standardisierten Fragen.

4.1.3 Nutzen-Aufwands-Bewertung von Augmented Reality-Potentialen nach RÖLTGEN

RÖLTGEN hat im Rahmen seiner Dissertation eine Methode zur Bewertung der Potentiale von AR zur Realisierung von Produkt-Service-Systemen (PSS) erarbeitet [Röl21].

Grundlage der Methode sind **16 generische AR-Anwendungsszenarien**. Diese Szenarien wurden von RÖLTGEN UND DUMITRESCU ausgehend von einer Sammlung an AR-Anwendungsbeispielen hergeleitet [RD20]. Zu den Szenarien werden Steckbriefe mit einer Beschreibung, einer Grafik sowie Chancen und Risiken bereitgestellt. Für jedes Szenario ist zudem definiert, wie stark es generische Nutzenpotentiale im Kontext von PSS erfüllt. Zu diesen Nutzenpotentialen zählen beispielsweise *kürzere Wartezeiten, erhöhte Sicherheit* und *verbesserte Fehlervermeidung*. Für die Anwendungsszenarien wird bei Anwendung der Methode nach RÖLTGEN jeweils bewertet, welchen Nutzen und Umsetzungsaufwand sowie welches Umsatzpotential ein entsprechender Service für ein vorliegendes Produkt hätte.

Für die **Bewertung des Nutzens** werden anhand einer Value Proposition Canvas nach OSTERWALDER ET AL. [OPB+14] zunächst Kundenbedarfe identifiziert. Anschließend werden die AR-Nutzenpotentiale den identifizierten Kundenbedarfen in einer Matrix ähnlich des House of Quality nach KLEIN (Abschnitt 4.1.1) gegenübergestellt. Es wird jeweils bewertet, wie gut die AR-Nutzenpotentiale jeweils zur Erfüllung der einzelnen Kundenbedarfe geeignet sind. Daraus ergibt sich eine Bewertung der Relevanz bzw. eine Gewichtung der AR-Nutzenpotentiale. Durch Anwendung der Gewichtung auf die jeweils statisch für die AR-Anwendungsszenarien definierten Nutzenwerte je AR-Nutzenpotential ergibt sich ein individueller Nutzen zwischen 0 und 3 für jedes AR-Anwendungsszenario.

Zur **Bewertung des Umsetzungsaufwands** wurden sechs Kriterien mit jeweils vier vordefinierten Ausprägungen von RÖLTGEN erarbeitet (Abbildung 4-4). Die Kriterien entsprechen charakteristischen Aufwandsaspekten im Kontext AR und die Ausprägungen unterschiedlichen Aufwandsstufen. Den Ausprägungen wird jeweils ein Wert von 0 bis 3 zugeordnet, wobei ein größerer Wert einen geringeren Aufwand bedeutet. Der Anwender der Methode nutzt die Ausprägungen, um jeweils einen Aufwandwert je Kriterium zu definieren. Durch eine Bewertung für alle AR-Anwendungsszenarien und anschließende Mittelung über die Kriterien ergibt sich jeweils ein Aufwandwert zwischen 0 und 3.

Bewertungskriterien zur Abschätzung des Umsetzungsaufwands	
Kriterien	Ausprägungen
IT-Integrationsaufwand bezeichnet alle Aufwendungen zur Einbindung der AR-Hard- und Software in die IT-Infrastruktur (i.d.R. des Kunden)	0 = hoher Aufwand mit kundenindividueller IT-Integration 1 = mittlerer Aufwand mit routinemäßigen Konfigurationstätigkeiten 2 = geringer Aufwand bei homogener IT-Landschaft 3 = kein Integrationsaufwand
Erstellung der virtuellen Inhalte umfasst Aufwände, die zur initialen Erschaffung der AR-Inhalte (z.B. 3D-Modelle) erforderlich sind	0 = Erstellung hochkomplexer virtueller Inhalte 1 = Erstellung virtueller Inhalte mit moderater Komplexität 2 = Erstellung einfacher virtueller Inhalte 3 = kein Aufwand, Integration bestehender virtueller Inhalte
Aktualisierung der virtuellen Inhalte bezeichnet Aufwände für die Überarbeitung und Anpassung der AR-Inhalte im	0 = fortlaufende umfängliche Aktualisierungen erforderlich 1 = gelegentliche Aktualisierungen mit moderatem Aufwand 2 = seltene, geringfügige Aktualisierungen

Abbildung 4-4: Bewertung von Aufwänden verschiedener Kriterien (Auszug) [RöL21]

Nachdem das Umsatzpotential, das für die vorliegende Arbeit nicht relevant ist, bewertet wurde, erfolgt eine Gegenüberstellung des Nutzens und der Umsetzungsaufwände in einer **Portfolio-Darstellung** (Abbildung 4-5). Die Darstellung erfolgt jeweils für einzelne Geschäftsfelder, die gesondert betrachtet werden. Anhand des Portfolios lässt sich intuitiv erkennen, welche AR-Anwendungsszenarien vielversprechend sind und welche eher nicht. Im gezeigten Beispiel hat das Szenario Remote Assistance einen hohen Nutzen und gleichzeitig einen geringen Umsetzungsaufwand. Es ist daher zu priorisieren und zeitnah anzugehen.

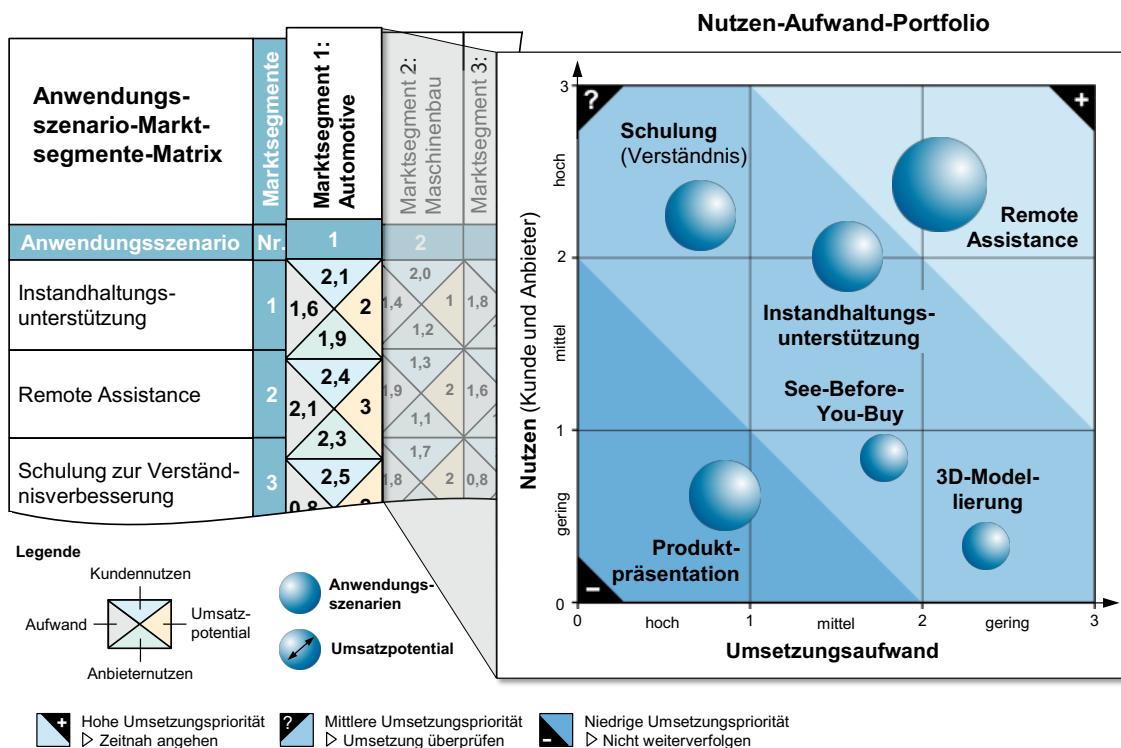


Abbildung 4-5: Gegenüberstellung der Nutzen- und Aufwandswerte von AR-Anwendungsszenarien je Geschäftsfeld in einem Portfolio [Rölt21]

Zwei der 16 Szenarien sind im Kontext Validierung einzuordnen. Im Szenario *See-before-you-buy* werden Produkte noch vor dem Kauf maßstabsgetreu in der späteren Einsatzumgebung visualisiert und so die Kaufentscheidung unterstützt. Dabei lassen sich speziell der Platzbedarf, mögliche Kollisionen sowie die Positionierung von Anschlüssen in realen Umgebungen überprüfen. Beim *Design Review* werden Produktentwürfe während der Entwicklung validiert mit dem Ziel, die Entwicklungskosten und -zeit zu reduzieren. Als Nutzenpotentiale werden beim *Design Review* unter anderem eine kognitive Entlastung sowie ein verbessertes Verständnis genannt.

Bewertung: Die Methode zur Bewertung von AR-Anwendungsszenarien nach RÖLTGEN stellt einen sehr systematischen Ansatz dar. Sie berücksichtigt sowohl den Nutzen als auch den Umsetzungsaufwand und ermöglicht so eine plausible Bewertung der Szenarien. Der Anwender wird unter anderem durch die detaillierten Szenario-Steckbriefe und

den Kriterienkatalog inkl. der Definition der einzelnen Ausprägungen der Aufwände sehr gut unterstützt. Es handelt sich zudem um einen Ansatz, der zwar eine quantitative Bewertung beinhaltet, die finale Entscheidung anhand der Portfolio-Darstellung jedoch dem Anwender überlässt. Die Gefahr, durch eine anhand von festen Metriken vorgenommene automatisierte Bewertung falsche Entscheidungen zu treffen, wird dadurch reduziert. Die Methode adressiert allerdings den speziellen Kontext der Produkt-Service-Systeme und ist nicht allgemeingültig für AR oder den speziellen Kontext Validierung anwendbar. Das grundlegende Konzept lässt sich jedoch übertragen. Daher ist der Ansatz vielversprechend für die systematische Planung AR-basierter Validierung.

4.1.4 Agile Practices Impact Model nach DIEBOLD UND ZEHLER

Das Agile Practices Impact Model (APIM) unterstützt eine bedarfsgerechte Integration von agilen Praktiken in einem Software-Entwicklungsprojekt [DZ15]. Dabei dient das APIM als Grundlage zur Beschreibung der Auswirkungen von agilen Praktiken auf Verbesserungsziele, die erreicht werden sollen. Durch eine individuelle Ausgestaltung und eine systematische Analyse des Modells wird die Identifikation bedarfsgerechter agiler Praktiken ermöglicht. Beim APIM handelt es sich um ein **Graph-basiertes Modell** (Abbildung 4-6), das folgende wesentliche Elemente als Knoten enthält:

- **Agile Praktiken** wie beispielsweise *Code Reviews* oder *Burn-Down Charts*, die für den Einsatz in einem Software-Entwicklungsprojekt in Frage kommen.
- **Auswirkungsmerkmale** entsprechen Aspekten der Entwicklung, die durch die agilen Praktiken beeinflusst werden können, bspw. Qualitäts- oder Zeitaspekte. In Bezug auf diese Auswirkungsmerkmale werden die Verbesserungsziele definiert.

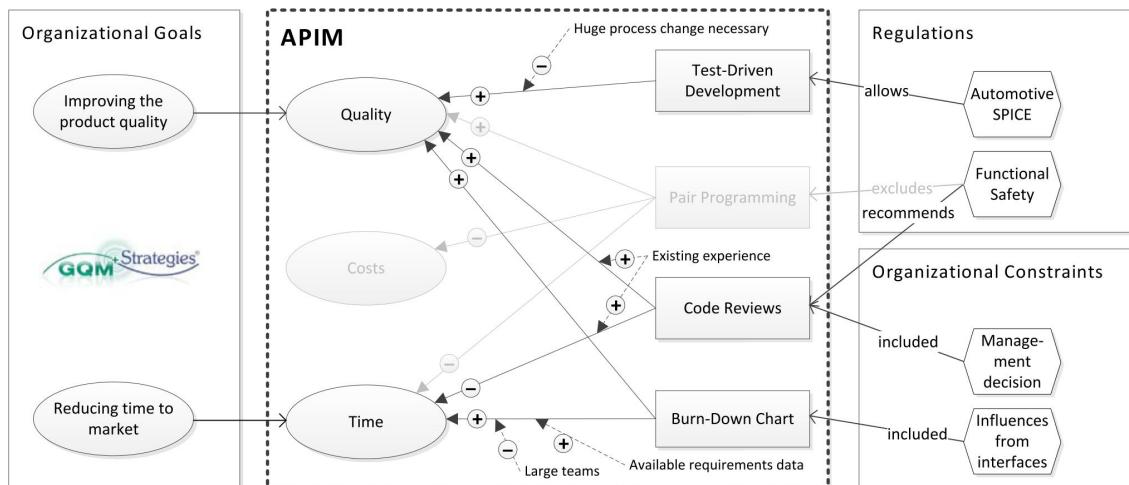


Abbildung 4-6: Das Agile Practices Impact Model (APIM) in der Übersicht [DZ15]

Zwischen den Praktiken und Merkmalen können **Einfluss-Relationen** definiert werden. Einflüsse können positiv oder negativ sein. Zudem können zusätzliche Faktoren beschrieben werden, die sich zusätzlich auf die Einflussrelationen auswirken. Zum Beispiel kann

der positive Einfluss von *Burn-Down-Charts* auf die *Zeit* durch zu große Teams beeinträchtigt werden. Vorhandene Erfahrung mit *Code-Reviews* kann dagegen dem negativen Einfluss auf die *Zeit* positiv entgegenwirken. Als externe Faktoren enthält das APIM zudem die **organisatorischen Ziele** sowie **Regularien und Rahmenbedingungen**. Die organisatorischen Ziele entsprechen der Motivation zur Integration agiler Praktiken. Die Ziele adressieren konkrete Auswirkungsmerkmale und werden entsprechend über Kanten mit ihnen verbunden. Beispielsweise adressiert das Ziel *Verbesserung der Produktqualität* das Auswirkungsmerkmal *Qualität*. Regularien sind unter anderem Standards wie *Automotive SPICE*. Rahmenbedingungen können durch das Management oder auch technische Infrastrukturen gegeben sein. Über Beziehungen zu den agilen Praktiken wird beschrieben, was durch Regularien und Rahmenbedingungen gefordert, empfohlen, erlaubt oder ausgeschlossen wird.

Die **Anwendung des APIM** erfolgt in drei Schritten:

- 1) Basierend auf den definierten organisatorischen Zielen erfolgt eine Auswahl der Auswirkungsmerkmale. Lediglich diese Auswahl wird im weiteren Verlauf betrachtet.
- 2) Durch die Analyse der externen Einflussfaktoren werden die in Frage kommenden agilen Praktiken gefiltert.
- 3) Im dritten Schritt erfolgt die eigentliche Analyse der Einflüsse der in Frage kommenden agilen Praktiken auf die relevanten Auswirkungsmerkmale und darauf basierend eine Auswahl der zu nutzenden agilen Praktik oder Praktiken.

Der genaue Prozess der Analyse im dritten Schritt wird in [DZ15] jedoch nicht adressiert. Laut DIEBOLD UND ZEHLER ist dies eine notwendige Weiterentwicklung des Konzepts basierend auf dem APIM. Bisher ist keine entsprechende Publikation bekannt.

Bewertung: Das APIM stellt einen interessanten Ansatz zur Beschreibung von Wissen, das für die Bewertung und Auswahl agiler Praktiken relevant ist, dar. Übertragen auf den Kontext AR-basierte Validierung könnten die agilen Praktiken den Potentialen von AR und die Auswirkungsmerkmale den Produktmerkmalen entsprechen. Anstatt der organisatorischen Ziele würden die zu analysierenden Produktmerkmale den Input darstellen. Sie könnten im Sinne der Generalisierung mit den Produktmerkmalen in Verbindung gebracht werden. Das von DIEBOLD UND ZEHLER beschriebene Konzept sieht einen Bewertungs- und Auswahlprozess vor, adressiert diesen jedoch nicht spezifischer. Zur Übertragung des Konzepts auf die vorliegende Arbeit müsste dieser Prozess entsprechend entwickelt werden.

4.1.5 Konfiguration AR-basierter Validierungsumgebungen nach REINEMANN

REINEMANN hat im Rahmen seiner Dissertation einen Ansatz zur Bewertung von Potentialen von AR konkret für die Anwendung im Kontext der Validierung erarbeitet [Rei21].

Basierend auf den Dimensionen der Wiedergabetreue nach LIM ET AL. (s. Anhang A2.1.2) wurden die in Abbildung 4-7 dargestellten 22 **Dimensionen der Wiedergabetreue AR-basierter Prototypen** definiert. Aufgrund der besonderen Eigenschaften von AR und der über AR präsentierten Prototypen wurden die Dimensionen ergänzt und angepasst. Sie werden unterteilt in die fünf Kategorien Erscheinung (visuell), Erscheinung (non-visuell), Funktionalität, Interaktivität sowie Meta-Funktionalität. Die 22 Dimensionen entsprechen aus Sicht von REINEMANN ET AL. allen möglichen durch AR-basierte Prototypen abbildbaren Aspekten eines Produkts.

Erscheinung (visuell)										Erschei-nung (non-visuell)	Funktio-nalität	Inter-aktivität	Meta-Funktionalität								
Größe	Farbe			Form	Struktur	Lage		Konstruktions-tiefe	Wirkbeziehungen		Räumliche Orientierung	Räumliche Positionierung	Akustik	Haptik	Funktionsumfang	Funktions-tiefe	Eingabeverhalten	Ausgabeverhalten	Metainformationen	Skalierbarkeit	Variierbarkeit
Proportion	Räumliche Dimension	Lichtwirkung	Farbwirkung	Transparenz	Okklusion	Oberflächenstruktur	Formtreue														

Abbildung 4-7: Dimensionen der Wiedergabetreue AR-basierter Prototypen in Anlehnung an REINEMANN ET AL. [RFH+19]

Basierend auf den 22 Dimensionen der Wiedergabetreue AR-basierter Prototypen wurde ein spezifisches Filter-Fidelity-Profil abgeleitet. Ein Filter-Fidelity-Profil dient der grafischen Darstellung der Bewertung der Wiedergabetreue von Prototypen (s. Anhang A2.1.2). Das Filter-Fidelity-Profil wird für die Konfiguration AR-basierter Validierungs-umgebungen eingesetzt (Abbildung 4-8). Der Ansatz dient der Festlegung des Virtualisierungsgrads, in dem einzelne Merkmale eines Produkts bzw. eines Teilsystems des Produkts bei der Validierung wiedergegeben werden [ARF+19]. Er beinhaltet indirekt die Bewertung der Potentiale von AR zur Validierung der Produktmerkmale.

Als Grundlage dienen sogenannte **Referenzprofile**. Ein Referenzprofil wird empirisch anhand eines Referenzprototyps erarbeitet. So wurde ein Prototyp eines Quadrocopters über ein AR-Gerät einer Testgruppe von 68 Personen präsentiert und anschließend durch die Testpersonen bewertet, wie gut die einzelnen Wiedergabedimensionen ausgestaltet sind. Dabei werden Werte von 1 bis 5 vergeben. 1 bedeutet, die Dimension ist überhaupt nicht ausgestaltet, und 5, die Dimension ist vollständig ausgestaltet. Das Referenzprofil gibt über eine Farbkodierung wieder, in wie viel Prozent der Bewertungen ein Wert jeweils vergeben wurde (Abbildung 4-8, Mitte). Im dargestellten Referenzprofil wurde die Dimension *Proportionen* beispielsweise sehr gut bewertet. Mehr als 90% der Testpersonen haben einen Wert von 4 und mehr als 70% einen Wert von 5 vergeben.

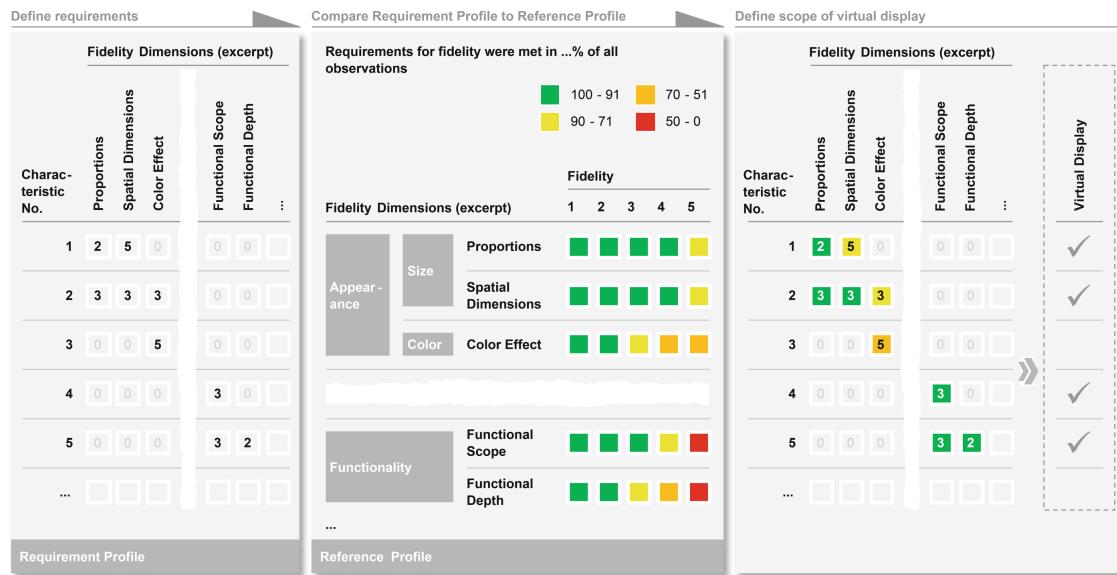


Abbildung 4-8: Konfiguration AR-basierter Validierungsumgebungen basierend auf den 22 Dimensionen der Wiedergabetreue AR-basierter Prototypen [ARF+19, S. 442]

Bei der Konfiguration der Validierungsumgebung wird jeweils eine individuelle **Bewertung der Anforderungen an die Validierung** vorgenommen (Abbildung 4-8, links). Zunächst erfolgt eine Konkretisierung der zu validierenden Produkteigenschaften und Merkmalsausprägungen. Für die einzelnen Merkmale („Characteristic“ in Abbildung 4-8) wird anschließend jeweils bewertet, wie gut sie durch einen Prototyp ausgestaltet sein müssen. Es ergibt sich das **Anforderungsprofil**.

Das Anforderungsprofil wird mit dem Referenzprofil abgeglichen. Für jedes Produktmerkmal wird je Dimension überprüft, wie hoch die prozentuale Bestätigung des im Anforderungsprofil geforderten Grads der Ausgestaltung im Referenzprofil ist (Abbildung 4-8, rechts). Liegt die Bestätigung bei allen Dimensionen für ein Produktmerkmal über 70%, wird dieses Produktmerkmal mittels AR wiedergegeben. Dies ist im Beispiel für die Produktmerkmale 1, 2, 4 und 5 der Fall. Der für Produktmerkmal 4 geforderte Wert von 5 für die Dimension *Farbeffekt* wird im Referenzprofil jedoch nur von 51 bis 70% bestätigt. Entsprechend wird dieses Produktmerkmal nicht virtuell repräsentiert.

Als einen weiteren Konfigurationsschritt wird die **Auswahl des AR-Endgeräts** beschrieben [Rei21]. Die Auswahl wird jedoch nicht weiter erläutert oder unterstützt. Es wird darauf hingewiesen, dass die Auswahl durch die Wahl des Referenzprofils beeinflusst wird. Es wird empfohlen, das AR-Endgerät zu nutzen, das für die empirische Erarbeitung des Referenzprofils zum Einsatz kam. So kann die größte Aussagekraft der Konfiguration erzielt werden. Zudem wird zwar darauf hingewiesen, dass zusätzliche Kriterien wie das finanzielle Budget oder die Praxistauglichkeit bei der Auswahl des Endgeräts berücksichtigt werden sollten. Auf weitere Details wird jedoch nicht eingegangen.

Bewertung: Die 22 Dimensionen der Wiedergabetreue AR-basierter Prototypen entsprechen einem bereits auf den Kontext AR zugeschnittenen Katalog an Produktmerkmalen. Die Dimensionen decken aus Sicht von REINEMANN ET AL. alle möglicherweise für die Validierung mittels AR relevanten Merkmale eines Produkts ab. Sie sind daher für die vorliegende Arbeit, speziell für die Generalisierung der Anforderungen, hoch relevant. Der darauf aufbauende Ansatz zur Konfiguration von AR-basierten Validierungsumgebungen ermöglicht eine systematische Bewertung der Eignung von AR für die Wiedergabe von einzelnen Produktmerkmalen. Anstelle der Produktmerkmale könnten auch konkrete Anforderungen zur Erarbeitung des Anforderungsprofils verwendet werden. Die Bewertung erfolgt jedoch anhand von nicht repräsentativen Referenzprofilen, die an konkrete AR-Endgeräte und exemplarische Prototypen gebunden sind. Somit wird keine allgemeingültige und generische Konfiguration ermöglicht. Der Ansatz enthält als letzten Schritt die Auswahl des AR-Endgeräts. Zum einen wird diese allerdings durch die Wahl des Referenzprofils bestimmt und zum anderen die Berücksichtigung weiterer Kriterien nicht im Detail adressiert.

4.2 Wissensbasierte AR-System-Konfiguration

Unternehmen sollen in die Lage versetzt werden, die AR-System-Konfiguration eigenständig und ohne spezielle Erfahrung und Expertise durchzuführen. Dazu ist relevantes Wissen in geeigneter Form aufzubereiten, zu formalisieren und als Grundlage zur Unterstützung der Konfiguration zu nutzen. Wissen baut grundsätzlich auf Daten und Informationen auf. Daten sind konkrete Fakten, bspw. ein konkreter Wert. Wenn Daten in einen Kontext gebracht werden, entstehen Informationen. So enthält ein konkreter Wert eine Bedeutung. Wenn Informationen miteinander vernetzt und in Zusammenhang gebracht werden, entsteht **Wissen** [VDI5610:2009]. PROBST ET AL. definieren Wissen wie folgt:

„Wissen bezeichnet die Gesamtheit der Kenntnisse und Fähigkeiten, die Individuen zur Lösung von Problemen einsetzen. Dies umfasst sowohl theoretische Erkenntnisse als auch praktische Alltagsregeln und Handlungsanweisungen. Wissen stützt sich auf Daten und Informationen, ist im Gegensatz zu diesen jedoch immer an Personen gebunden. Es wird von Individuen konstruiert und repräsentiert deren Erwartungen über Ursache-Wirkungs-Zusammenhänge“ [PRR12, S. 23]

Bei der Konfiguration von AR-Systemen greifen Experten sowohl auf bekannte Daten und Informationen zur Technologie als auch auf vorhandene Erkenntnisse zurück. Die Experten verfügen somit über für die AR-System-Konfiguration relevantes Wissen und generieren zudem durch neue Erfahrungen neues Wissen. Es wird unterschieden zwischen explizitem und implizitem Wissen. **Explizites Wissen** entspricht Faktenwissen. Es lässt sich mit geeigneten Informations- und Kommunikationstechnologien formalisieren und nutzen. **Implizites Wissen** dagegen ist schwierig zu formalisieren. Es liegt vor allem in Form von Erfahrungen bei Experten vor und ist schwer zu beschreiben [Bau15]. Für

die AR-System-Konfiguration sollen das relevante explizite und auch implizites Wissen genutzt werden, um Unternehmen eine eigenständige Konfiguration zu ermöglichen.

Wissensbasierte Systeme sind Software-Systeme, die Wissen für einen spezifischen Kontext enthalten und dieses Wissen zur Lösung einer Problemstellung einsetzen [Bau15]. Wissensbasierte Systeme stellen das für die Durchführung einer Handlung fehlende Wissen situationsspezifisch bereit. Sie können somit laut der Definition von WANDKE auch als Assistenzsysteme angesehen werden [Wan05]. Als Synonym für wissensbasierte Systeme wird zudem häufig der Begriff Expertensystem verwendet [GR98]. Ein besonderes Merkmal wissensbasierter Systeme ist die strikte Trennung des Wissens von der Logik und Funktionalität des Software-Systems. So bestehen wissensbasierte Systeme im Wesentlichen aus einer **Wissensbasis** und einer **Problemlösungskomponente** [Kur89]. In der Wissensbasis wird relevantes Wissen gespeichert und formalisiert. Über die Problemlösungskomponente wird eine gezielte Analyse und Auswertung der Wissensbasis realisiert. Weitere Details zur Struktur wissensbasierter Systeme sowie eine systematische Vorgehensweise zur Erstellung eines Wissensmodells werden in Anhang A2.2 beschrieben.

Das für die AR-System-Konfiguration relevante Wissen sowie die verfügbaren Wissensquellen sind vielfältig. Teilweise werden einzelne Wissensaspekte durch die in diesem Kapitel beschriebenen Ansätze adressiert. Weitere Informationen zu relevantem Wissen werden in Anhang A2 ergänzt. Das Wissen gilt es, in einer Wissensbasis zu formalisieren. Im folgenden Abschnitt 4.2.1 werden daher zunächst Techniken für die Repräsentation und Verarbeitung von Wissen untersucht. Basierend auf der Wissensbasis soll über eine Problemlösungskomponente die bedarfsgerechte AR-System-Konfiguration realisiert werden. Entsprechend werden in Abschnitt 4.2.2 bestehende Ansätze zur Konfiguration technischer Systeme analysiert.

4.2.1 Repräsentation und Verarbeitung von Wissen

Weltweit liegt eine große Menge von Wissen in natürlicher Sprache vor [Bey13]. Entsprechend der von KOHN beschriebenen systematischen Vorgehensweise zur Entwicklung wissensbasierter Systeme (s. Anhang A2.2.2) muss dieses Wissen in Form von Wissensmodellen geeignet beschrieben werden. Entsprechende Ansätze zur Modellierung werden in Abschnitt 4.2.1.1 analysiert. Mit Blick auf die spätere Analyse des Wissens durch Software im Rahmen der Problemlösungskomponente werden anschließend semantische Technologien (Abschnitt 4.2.1.2) untersucht.

4.2.1.1 Modelle und Modellierungssprachen

Wissen kann unter anderem in Form von Modellen repräsentiert werden. Modelle dienen der Abbildung komplexer Sachverhalte und Zusammenhänge. Sie beschreiben dabei

meist lediglich einen Teilespekt der Realität. Für die Modellerstellung wird eine Modellierungssprache verwendet. Die Modellierungssprache legt die Syntax sowie die Semantik des Modells fest [Bau15]. Bei der Syntax wird zwischen der abstrakten und der konkreten Syntax unterschieden. Die abstrakte Syntax definiert, welche Elemente im Modell enthalten und wie die Elemente miteinander verbunden sein können. Die konkrete Syntax beschreibt die Darstellungsform des Modells. Die Semantik beschreibt mögliche Einschränkungen, bspw. Wertebereiche, sowie die Bedeutung der Elemente und Beziehungen im Modell [VAC+09]. Laut SCHADY kann mit einem Modell spezifisches Wissen über eine Domäne oder ein System inklusive aller relevanten Instanzen und Relationen beschrieben werden [Sch08b]. Es existieren verschiedenste Modellierungssprachen, die für unterschiedliche Einsatzzwecke entwickelt wurden. Im Folgenden werden gängige Modellierungssprachen aus dem Kontext technischer Systeme analysiert.

UML – Unified Modelling Language: Die UML ist eine der gängigsten Modellierungssprachen. Es handelt sich um eine grafische, semiformale Sprache aus dem Bereich der Softwareentwicklung bzw. softwareintensiver Systeme. Mit der UML lassen sich Strukturen, Architekturen und das Verhalten eines Systems sowie Interaktionen mit anderen Systemen anhand verschiedener Arten von Diagrammen modellieren [For07]. Die Grundlage der UML bilden Objekte und Klassen, die wiederum spezifische Attribute und Methoden beinhalten. **Objekte** entsprechen konkreten Sachverhalten, Elementen oder Individuen. Objekte können spezifische Eigenschaften als **Attribute** haben. Zudem können Objekte über **Methoden** verfügen, die das Verhalten des Objekts definieren. So reagieren Objekte entsprechend den Methoden in einer definierten Art und Weise auf Input von außen. Jedes Objekt besitzt eine Identität, sodass es von allen anderen Objekten unterschieden werden kann. Eine **Klasse** entspricht einer Sammlung von Objekten, die über die gleichen Attribute, Methoden und Zusammenhänge zu anderen Objekten verfügen. Dabei haben die Objekte zwar die gleichen Attribute, können aber individuelle Attributwerte aufweisen. Als Beispiel kann eine Klasse *Student* das Objekt *Paul* mit einem *Geburtsdatum* und dem *Einkommen* als Attribute sowie den Methoden *Lernen* und *Feiern* enthalten [For07]. Als ausführbare Einheiten im Kontext der Softwareentwicklung können zudem Komponenten definiert werden. Sie verfügen über festdefinierte Schnittstellen und bestehen in der Regel aus einer Menge von Klassen zur Realisierung der gewünschten Funktionalität. Entwickelt wird die UML durch die Object Management Group (OMG). Eine aktuelle Version der UML ist in der ISO/IEC19505 standardisiert [ISO/IEC19505-1:2012], [ISO/IEC19505-2:2012].

SysML – Systems Modelling Language: Die SysML baut auf der UML auf und erweitert sie. Sie wurde speziell für den Einsatz im Systems Engineering entwickelt und ermöglicht die ganzheitliche und disziplinübergreifende Modellierung technischer Systeme. Dazu bietet die SysML verschiedene Struktur-, Anforderungs- und Verhaltensdiagramme, wobei eine Vielzahl der Diagramme aus der UML direkt übernommen wurde [Wei06]. Lediglich das Anforderungs- und das Zusicherungsdiagramm wurden ergänzt. **Strukturdiagramme** dienen der Beschreibung sowohl informationsverarbeitender als

auch physikalischer Elemente eines Systems. Die in der UML Klassen genannten Elemente heißen in der SysML Blöcke. So beschreibt das *Blockdefinitionsdiagramm* die Bestandteile des Systems sowie die zwischen diesen vorliegenden Beziehungen, Generalisierungen und Abhängigkeiten. Im *internen Blockdiagramm* können die Beziehungen der einzelnen Bestandteile eines Blocks in Form von Ports, Konnektoren und Flüssen näher beschrieben werden. Das *Zusicherungsdiagramm* ermöglicht zudem die Definition von Beziehungen zwischen Eigenschaften von Blöcken und somit die Modellierung physikalischer Gesetze [Dum10]. SysML bietet gute Möglichkeiten, um neben der Struktur, des Verhaltens sowie der Anforderungen auch Zuordnungen und Einschränkungen von Systemeigenschaften zu modellieren. **Anforderungsdiagramme** dienen der Definition funktionaler und nicht-funktionaler Anforderungen. Dabei können verschiedene Beziehungen zwischen den Anforderungen definiert werden. Dazu zählen Ableitungs-, Enthält-, Erfüllungs-, Kopie-, Prüf-, Verfeinerungs- und Verfolgungsbeziehungen [Ana15]. Anforderungsdiagramme werden typischerweise grafisch oder oft auch in tabellarischer Form dargestellt. Zu den **Verhaltensdiagrammen** zählen *Zustands-, Use-Case-, Sequenz- sowie Aktivitätsdiagramme*. Sie dienen unter anderem der Beschreibung der Interaktion des Systems mit dem Benutzer sowie der Systemelemente untereinander. Aufgrund der Vielzahl an Diagrammen erfordert die SysML eine intensive Einarbeitung. Zudem ist die Anwendung der Modellierungssprache aufgrund einer fehlenden Modellierungsmethodik nicht immer eindeutig und intuitiv [Ana15]. Die aktuellste von der Object Management Group (OMG) veröffentlichte Version der SysML ist die 1.6 [OMG19].

CONSENS: Die Modellierungssprache CONSENS wurde am Heinz Nixdorf Institut in Paderborn im Rahmen des SFB 614 als Spezifikationstechnik für komplexe mechatronische Systeme entwickelt [GFD+08]. CONSENS umfasst neben der Modellierungssprache zusätzlich eine Methodik, die die Anwendung der Sprache anleitet. So ermöglicht CONSENS eine ganzheitliche Modellierung disziplinübergreifender mechatronischer Systeme mit dem Ziel eines einheitlichen Systemverständnisses aller Akteure [PBG+14]. CONSENS fokussiert die Konzeption eines Produkts, unterstützt aber auch die des zugehörigen Produktionssystems [GDR+15]. Zur Modellierung des Produkts werden sieben sogenannte **Partialmodelle** verwendet (Abbildung 4-9). Das *Umfeldmodell* beschreibt die Wechselwirkungen zwischen Elementen des Umfelds und dem System. In den *Anwendungsszenarien* werden durch das System zu lösende Probleme und das entsprechend gewünschte Verhalten des Systems definiert. Sämtliche Anforderungen an das System werden in einer Liste in natürlicher Sprache definiert und dokumentiert. *Anforderungen* betreffen neben dem Produkt auch das Produktionssystem. *Funktionen* beschreiben den Zusammenhang zwischen Eingangs- und Ausgangsgrößen, um eine konkrete Funktion zu erfüllen. Sie werden in einer hierarchischen Form mit Subfunktionen dokumentiert. Die *Wirkstruktur* stellt die einzelnen Systemelemente und die Beziehungen zwischen ihnen dar. Die Wirkstruktur umfasst entsprechend alle strukturellen Details des Systems. Das *Verhalten* wird ähnlich zur SysML beispielsweise über Zustands-, Aktivitäts- oder Sequenzdiagramme definiert. So werden Ablaufprozesse und Aktivitäten der interagie-

renden Systemelemente modelliert. Die *Gestalt* des Systems wird anhand von 3D-Modellen beschrieben. Zudem können wechselseitige Beziehungen zwischen den Konstrukten der Partialmodelle abgebildet werden [Ana15].

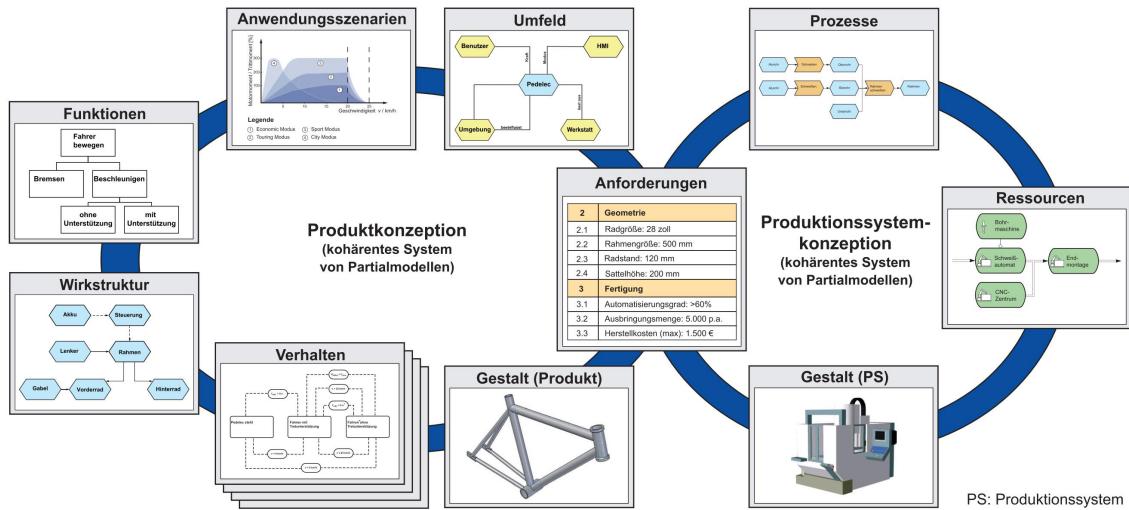


Abbildung 4-9: Partialmodelle von CONSENS zur fachgebietsübergreifenden Beschreibung eines Produkts und des entsprechenden Produktionssystems [GDR+15]

Bewertung: Modelle stellen eine wichtige Grundlage zur Beschreibung von Wissen dar. Laut SCHADY kann mit einem Modell das gesamte relevante Wissen für eine spezifische Domäne abgebildet werden. Im vorliegenden Fall der AR-System-Konfiguration beinhaltet dieses Wissen vor allem die Struktur eines AR-Systems sowie Wechselwirkungen innerhalb des AR-Systems und aus dem Umfeld. Zur Modellierung dieser Sachverhalte bietet UML grundsätzlich gute Möglichkeiten. Sowohl Systemstrukturen als auch Zusammenhänge zwischen Elementen sowie Attribute einzelner Elemente können modelliert werden. Da es sich bei der UML um eine gängige Modellierungssprache aus der Softwareentwicklung handelt, existieren zudem verschiedene Möglichkeiten zur digitalen Verwendung und Verarbeitung von UML-Modellen. SysML und CONSENS fokussieren technische bzw. mechatronische Systeme und deren vollumfängliche Beschreibung. Beide Modellierungssprachen beinhalten daher einige Aspekte, die für das angestrebte wissensbasierte System zur AR-System-Konfiguration nicht relevant sind. Zudem bedarf speziell SysML aufgrund des Mangels an einer zugehörigen Methodik einer zeitintensiven Einarbeitung. Sowohl SysML als auch CONSENS sind zu umfangreich und komplex für die vorliegende Arbeit.

4.2.1.2 Semantische Technologien

Das Konzept des Semantic Web als Weiterentwicklung des World Wide Web (WWW) ermöglicht eine maschinenlesbare Beschreibung der darzustellenden Inhalte [BHL01]. Die dem Semantic Web zugrunde liegenden Standards und Technologien werden als semantische Technologien bezeichnet [BP06]. Im Gegensatz zu den in Abschnitt 4.2.1.1

beschriebenen Modellierungsansätzen ermöglichen semantische Technologien eine **maschinenverständliche Repräsentation** von Wissen. Als Grundlage der Wissensrepräsentation mittels semantischer Technologien dienen **Ontologien**. Ontologien entsprechen einem formalen Repräsentationsmodell. Sie definieren ein wohldefiniertes Vokabular an möglichen Elementen und deren Beziehungen in einem betrachteten Themenfeld [Bau15]. Ontologien ermöglichen somit ein einheitliches Verständnis von Modellen, beispielsweise für den Datenaustausch zwischen Software-Systemen, und die Wiederverwendung von Wissen [BP06]. Eine Ontologie kann ähnlich der zuvor beschriebenen Modellierungsansätze Klassen, Instanzen von Klassen, Eigenschaften und Relationen enthalten. Es wird unterschieden zwischen *Top-Level-, Core- und Domain-Ontologien*. *Top-Level-Ontologien* beschreiben Konzepte, die Themenfeld-übergreifend verwendet werden können. *Core-Ontologien* können in mehreren ähnlichen Bereichen verwendet werden. Und *Domain-Ontologien* beschreiben ein spezifisches Themenfeld im Detail [Obe06].

Semantische Technologien und Ontologien ermöglichen eine maschinenlesbare Aufbereitung von Daten, Informationen und Wissen. Sie unterstützen somit die Arbeit mit großen Informationsmengen und vereinfachen die Datenintegration in wissensbasierten Systemen [Bau15]. Beim Einsatz semantischer Technologien zur Wissensrepräsentation ist das Zusammenspiel verschiedener Ansätze auf verschiedenen **Ebenen der Wissensrepräsentation** notwendig. Eine Übersicht liefert Abbildung 4-10.

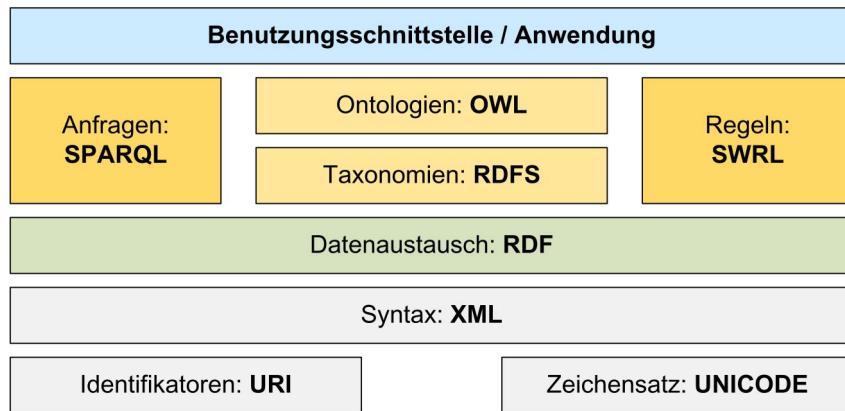


Abbildung 4-10: *Übersicht über die Ebenen der Wissensrepräsentation beim Einsatz semantischer Technologien [Bau15]*

Die technische Grundlage bilden der *Uniform Resource Identifier (URI)*, der **UNICODE**-Zeichensatz sowie die *XML*-Syntax. Der *URI* ermöglicht eine eindeutige Identifikation von Elementen wie sie beispielsweise bei Büchern mittels der ISBN möglich ist [HKR+08]. Die grundlegende Datenmodellierung erfolgt über das **Resource Description Framework (RDF)**. Das RDF ermöglicht die Definition beliebiger Objekte, der sogenannten Resourcen, und zugehöriger Beziehungen zu anderen Resourcen [Bau15]. Die konkreten Informationen werden in *Subjekt-Prädikat-Objekt-Tripeln* repräsentiert, die ei-

nem gerichteten Graphen entsprechen. Das Subjekt entspricht einer spezifischen Resource und stellt den Startknoten dar. Das Prädikat definiert die Beziehung zwischen dem Subjekt und dem Objekt als Endknoten und entspricht somit einer gerichteten Kante. Das Objekt kann dabei sowohl eine Resource als auch ein konkreter Datenwert sein. Alle drei Elemente werden über *URI* spezifiziert und sind somit eindeutig identifizierbar. Abbildung 4-11 zeigt zusammenhängende RDF-Triple am Beispiel eines Buchs. Das Buch als Resource hat einen Datenwert als Titel (Triple 1). Es wird bei einem Verlag verlegt (Triple 2). Der Verlag stellt eine weitere Resource dar und hat einen Datenwert als Namen (Triple 3). Mit RDF-Tripeln können entsprechend komplexe Sachverhalte und Zusammenhänge abgebildet werden. Zur maschinenlesbaren Repräsentation von RDF-Datenmodellen existieren verschiedene auf XML aufbauende Formate. Vom World Wide Web Consortium (W3C) wird der Standard RDF/XML definiert [BGS14-01]. Weit verbreitet ist zudem die Turtle-Syntax [BB11-01].

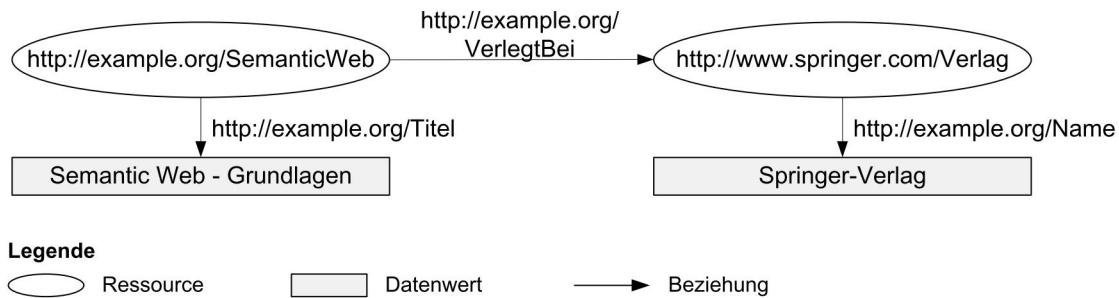


Abbildung 4-11: Zusammenhängende RDF-Tripel am Beispiel eines Buchs [HKR+08]

Aufbauend auf dem RDF wurde das **Resource Description Framework Schema (RDFS)** entwickelt [BG14-01]. Das RDFS definiert ein genaues Vokabular und eine zugehörige Semantik und ermöglicht somit eine standardisierte Verwendung von mittels RDF repräsentiertem Wissen [Deu07]. Als Erweiterung des RDF ermöglicht das RDFS zudem die Notation von Klassen und Klassenhierarchien.

Eine weitere Wissensrepräsentations- und Ontologiesprache ist die **Web Ontology Language (OWL)**. Die OWL wurde im Februar 2004 von der W3C zu einer Empfehlung für das Semantic Web [Mv04-01]. Ziel der OWL ist die Beschreibung von Wissen in einer Form, die auch von Software verstanden werden kann [Dob08]. Ähnlich zum RDFS handelt es sich bei der OWL um eine Sprache, mit der sich Klassen, Beziehungen zwischen Klassen sowie Individuen von Klassen beschreiben lassen.

Semantische Technologien bieten durch die maschinenlesbare Wissensrepräsentation die Möglichkeit der Schlussfolgerung. Sogenannte **Reasoner** sind in der Lage, Beziehungen zwischen den Individuen auszuwerten und daraus neues Wissen zu generieren [Sch09]. Das Schlussfolgern basiert unter anderem auf den Charakteristika der Beziehungen. Hat ein Element *X* eine transitive Beziehung zu Element *Y* und *Y* die gleiche transitive Beziehung zu *Z*, bedeutet dies, dass auch *X* diese Beziehung zu *Z* hat bzw. haben sollte. Ein Reasoner kann somit implizit vorliegendes Wissen, bspw. durch eine Kette an transitiven Beziehungen, identifizieren und als explizites Wissen der Wissensbasis hinzufügen

[Yu14]. Mit Hilfe von **Regeln** und **Restriktionen** können Reasoner zusätzliches Wissen in einer OWL-Ontologie ableiten. Technische Details zur OWL sowie zur Entwicklung OWL-basierter Wissensbasen werden in Anhang A2.2.3 beschrieben.

Ontology Learning bezeichnet den Wissenserwerbsprozess im Kontext des Semantic Web [JCT19]. Es existieren Ansätze, um Wissen automatisiert aus verschiedenen Quellen zu extrahieren und in eine Ontologie zu überführen. Dabei muss zwischen strukturierten und unstrukturierten Quellen unterschieden werden. Ein Großteil des im Internet verfügbaren Wissens liegt unstrukturiert vor. So enthielten 2013 nur knapp 26% von circa 2,2 Milliarden untersuchten Webseiten strukturierte Meta-Daten [MBP15]. Die automatisierte Extraktion von unstrukturiertem Wissen stellt eine Herausforderung des Ontology Learnings dar. Die meist in natürlicher Sprache vorliegenden Informationen gilt es auf die in Ontologien verwendeten Individuen, Klassen und Beziehungen zu übertragen [MHL16]. Dazu müssen unter anderem Ansätze zur intelligenten Textverarbeitung, beispielsweise Natural Language Processing (NLP), eingesetzt werden [WLB12].

Für die Extraktion von Wissen aus strukturierten Quellen existieren verschiedene ausgereiferte Ontology Learning-Ansätze. ZEDLITZ hat im Rahmen seiner Dissertation untersucht, inwiefern UML-Modelle in OWL-Ontologien überführt werden können [Zed13]. Grundsätzlich lässt sich die Semantik von UML-Modellen gut auf Ontologien übertragen. Sowohl UML-Modelle als auch OWL-Ontologien können in einer XML-basierten Form beschrieben werden. So lassen sich **UML-Modelle** anhand von speziellen Transformationsregeln in OWL-Ontologien überführen. Dabei sind jedoch gewisse Einschränkungen zu beachten. Unter anderem lassen sich die in UML möglichen abstrakten Klassen, bestimmte Arten der Generalisierung sowie Stereotypen nicht auf OWL übertragen. Zudem adressieren die übertragbaren Elemente lediglich strukturelle Aspekte eines Modells, bspw. Klassen, Datentypen und Beziehungstypen. Konkrete Individuen und deren Beziehungen lassen sich nicht aus UML in OWL übertragen. Unter anderem liegt das daran, dass UML namenlose Elemente erlaubt, OWL dagegen nur eindeutig identifizierbare Elemente zulässt. Entsprechend können die eigentlichen Inhalte nicht automatisiert aus UML-Modellen in eine Ontologie eingelesen werden.

Da viele Informationen und umfangreiches Wissen in Kalkulationstabellen vorliegen, wurden verschiedene **Tabellen-Mapping-Techniken** entwickelt. Mit Tabellen-Mapping-Techniken lassen sich Inhalte aus Kalkulationstabellen einlesen und darauf aufbauend automatisiert OWL-Ontologien erstellen oder erweitern. Eine dieser Techniken ist die von O'CONNOR an der Stanford University entwickelte *M² Language*. Mit der *M² Language* können generelle Informationen über Klassen und Zusammenhänge und konkrete Individuen und Beziehungen eingelesen und in eine OWL-Ontologie überführt werden [OHM10]. Somit lassen sich sowohl die strukturellen als auch die inhaltlichen Aspekte automatisiert übertragen. Abbildung 4-12 zeigt eine von KÜGLER ET AL. beschriebene Vorgehensweise zur Überführung von Informationen aus Kalkulationstabellen in eine OWL-Ontologie [KSW18]. Ausgangspunkt ist eine Kalkulationstabelle, die alle re-

levanten Informationen enthält. Um die Informationen in eine OWL-Ontologie übertragen zu können, müssen sie zunächst vorverarbeitet werden. Bei der Vorverarbeitung werden die Informationen in eine durch die Mapping-Technik vorgegebene Form überführt. Im Import-Schema wird über Transformationsregeln, beispielsweise basierend auf der *M² Language*, genau definiert, welche Daten aus welchen Zellen in welche OWL-Elemente zu überführen sind. Im Beispiel in Abbildung 4-12 entsprechen die Einträge in Spalte *A* Individuen der Klasse *Requirement*. Sternchen hinter den Spalten-Buchstaben bedeuten, dass nicht eine spezifische Zelle, sondern alle Zellen einer Spalte eingelesen werden sollen. Der Inhalt der Zellen in Spalte *A* wird jeweils als eindeutige ID des Individuums verwendet. Den Individuen werden eine Beschreibung als *rdfs:comment* und eine Versionsnummer als *owl:versionInfo* aus den Spalten *C* und *B* zugeteilt. Die Informationen werden als Tripel in der OWL-Ontologie angelegt und repräsentiert.

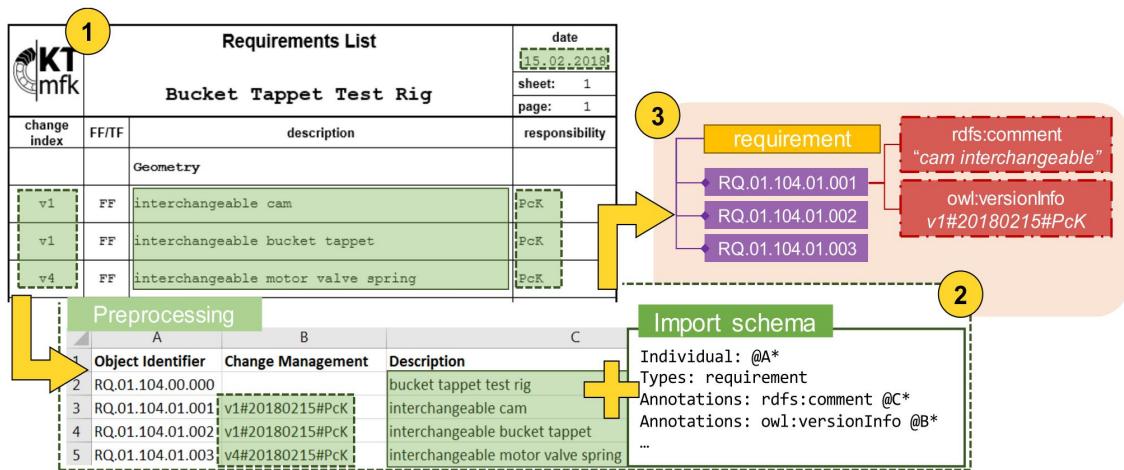


Abbildung 4-12: Vorgehensweise zur Überführung von Informationen mittels Excel-Tabelle in eine OWL-Ontologie [KSW18]

Bewertung: Wie die in Abschnitt 4.2.1.1 beschriebenen Modellierungsansätze ermöglichen auch semantische Technologien eine vollumfängliche Beschreibung von Wissen zu einer ausgewählten Domäne bzw. einem Anwendungsgebiet. Großer Vorteil semantischer Technologien gegenüber klassischen Modellierungsansätzen ist die maschinenlesbare Repräsentation von Wissen. Über Regeln und Reasoner kann zudem aufwandsarm zusätzliches Wissen generiert werden. Dank einer über Ontologien genau spezifizierten Semantik kann das Wissen von Software verstanden werden. Eine Wissensrepräsentation über semantische Technologien bzw. Ontologien ist somit vielversprechend für das zu entwickelnde wissensbasierte Konfigurationssystem. Als Ontologiesprache bietet sich die von der W3C empfohlene OWL an. Für das Anlegen des umfangreichen Wissens in der Wissensbasis stellen Tabellen-Mapping-Techniken wie die *M² Language* einen vielversprechenden Ansatz dar. Ein manuelles Anlegen einzelner Tripel für jede Information wäre sehr aufwendig. Die Überführung von UML-Modellen scheint entweder nicht vollumfänglich möglich oder zu komplex zu sein. Zudem würde eine Repräsentation des ge-

samten relevanten Wissens mittels UML oder CONSENS zu sehr komplexen und unüberschaubaren Modellen führen. Tabellen-Mapping-Techniken dagegen ermöglichen eine automatisierte Überführung von Wissen aus Tabellen in OWL-Ontologien. Bei geeigneter Strukturierung des Wissens ermöglichen Tabellen zudem eine einfache Erweiterung der Informationen.

4.2.2 Konfiguration technischer Systeme

Im Folgenden werden verschiedene Ansätze zur Konfiguration technischer Systeme untersucht, die für die Entwicklung der Problemlösungskomponente der wissensbasierten AR-System-Konfiguration hilfreich sein können.

4.2.2.1 AR-System-Klassifikation nach FELLMANN ET AL.

FELLMANN ET AL. beschreiben ein **Klassifikationsschema** für AR-Systeme im Kontext des Technischen Kundendiensts [FHM+15]. Das Schema umfasst drei anwendungsbezogene und zwei technologiebezogene Merkmalsgruppen, die insgesamt 14 Merkmale von AR-Systemen umfassen. Die anwendungsbezogenen Merkmale betreffen das Anwendungsszenario, die Mobilität sowie die Benutzerschnittstelle. Technologiebezogen sind die Software und die Hardware des AR-Systems. Zu jedem Merkmal gibt es eine feste Auswahl an möglichen Ausprägungen. Das Klassifikationsschema wird von FELLMANN ET AL. auf acht ausgewählte AR-Systeme angewendet. Dabei werden für jedes AR-System die zutreffenden Merkmalsausprägungen bestimmt. Abbildung 4-13 zeigt dies am Beispiel des AR-Systems „ARAS“. ARAS kommt im *Gesundheitswesen* zum Einsatz und ist *multiuserfähig*. Das System ist *drahtgebunden*, wird *stationär* eingesetzt und hat lediglich eine *niedrige bis mittlere* Beeinträchtigung des Nutzers. Die Benutzerschnittstelle ermöglicht eine *taktile* sowie *optische* Eingabe und eine *visuelle* und *akustische* Ausgabe. Die ARAS-Technologie ermöglicht *Tracking* zur Objektidentifikation sowie die Darstellung von *3D-Grafiken* und *akustische* Informationswiedergabe. Dabei findet die Datenverarbeitung *lokal* statt. Die Daten werden dagegen *verteilt* gespeichert. Mit einer *Kamera*, einem *Zeigegerät*, einem *Mikrofon* sowie einem *Tastengerät* verfügt ARAS über verschiedene Eingabegeräte. Ausgabegeräte sind ein *Bildschirm* in Form eines *Head-Mounted Displays (HMD)* sowie ein *Lautsprecher*. Ein *ortsfester Computer* stellt die Plattform dar.

Das Klassifikationsschema kann zudem zur Definition von Anforderungen an ein AR-System verwendet werden. FELLMANN ET AL. definieren die Anforderungen für den konkreten Anwendungsfall des Technischen Kundendiensts [FHM+15]. Es ergibt sich eine wie in Abbildung 4-13 für ein AR-System dargestellte Klassifikation. Diese **Anforderungs-Klassifikation** für einen vorliegenden Anwendungsfall kann dann mit allen vorhandenen Klassifikationen von AR-Systemen abgeglichen und so ein geeignetes AR-System identifiziert werden.

Merkmalsklasse		Merkmale		Merkmalsausprägung					
Anwendungsbezogen	Anwendungsszenario	Einsatzbereich	Herstellung & Produktion	Entertainment	Allg. Kundendienst				
			Entwicklung & Design	Allgemeinwesen	Gesundheitswesen				
			Marketing	Militär	Ausbildung				
			nicht multiuserfähig		multiuserfähig				
	Mobilität	Multiuserfähig	stationär	eingeschränkt	uneingeschränkt				
		Transportierbarkeit	drahtgebunden	drahtlos lokal	drahtlos unabhängig				
		Datenverbindung	niedrig	mittel	hoch				
	Benutzerschnittstelle	Nutzerbeeinträchtigung	taktile	gestisch	optisch	akustisch			
		Eingabe	haptisch	visuell	akustisch				
		Ausgabe	GPS	Barcode	RFID	Tracking			
		Objektdentifikation	3D-Grafik	2D-Grafik	akustisch				
Technologiebezogen	Software	Informationsdarstellung	lokal	kooperativ	entfernt				
		Datenverarbeitung	lokal	verteilt	entfernt				
		Datenhaltung	Kamera	Zeigegerät	Tastengerät	Sonstiges Eingabegerät			
		Eingabegeräte	Bildschirm	Head-Mounted Display	Projektor	Virtual-Retinal Display			
	Hardware	Ausgabegeräte	Lautsprecher	haptisches Ausgabegerät	sonstiges Ausgabegerät				
		tragbares Kleingerät	tragbares Kleingerät	tragbarer Computer	ortsfester Computer	sonstige Plattform			

Abbildung 4-13: Beispielhafte Anwendung des Klassifikationsschemas nach FELLMANN ET AL. auf das AR-System "ARAS" in Anlehnung an [FHM+15]

Bewertung: Das Klassifikationsschema nach FELLMANN ET AL. stellt eine gute Möglichkeit zur Beschreibung spezifischer AR-Systeme dar. Das Schema ist für den Anwendungsfall des Technischen Kundendiensts ausgelegt. Das Konzept ließe sich auf andere Anwendungsfelder übertragen und auch generalisieren. Allerdings erfolgt die Identifikation geeigneter Geräte manuell anhand eines Abgleichs der AR-Systeme mit den Anforderungen. Entsprechend stellt der Ansatz nur bedingt eine Unterstützung für die komplexe AR-System-Konfiguration dar. Zudem wird nur ein geringer Teil des für die Konfiguration relevanten Wissens abgedeckt. Beispielsweise werden Einflüsse aus dem Umfeld nicht berücksichtigt.

4.2.2.2 Bewertung von AR-Systemen nach RÖLTGEN

RÖLTGEN beschreibt in seiner Dissertation einen Leitfaden zur Bewertung und Auswahl eines AR-Systems für eine konkrete Idee eines Produkt-Service-Systems [Röl21]. Im ersten Schritt wird anhand von 15 **Eignungsindikatoren** bestimmt, ob ein Handheld-AR-System oder eine Datenbrille empfehlenswert sind (Abbildung 4-14). Für jeden Indikator wird durch den Anwender jeweils bewertet, inwiefern die jeweilige Aussage zutrifft. Durch die Mittelung der Antwort-Werte ergibt sich ein Wert zwischen 0 und 3 als Handlungsempfehlung. Werte in Richtung 0 bedeuten, Handheld-Systeme sind empfehlenswert. In diesem Fall wird abhängig von individuellen Anforderungen wie dem Budget, der Displaygröße oder auch dem Gewicht ein handelsübliches Smartphone oder Tablet ausgewählt. Geht der gemittelte Wert Richtung 3, ist eine Datenbrille geeignet. In diesem Fall erfolgt eine detaillierte Bewertung verfügbarer Datenbrillen.

Ähnlich zum Ansatz von FELLMANN ET AL. nutzt RÖLTGEN ein Anforderungsprofil sowie Steckbriefe zu Datenbrillen für die Auswahl einer geeigneten Datenbrille. Das **Anforderungsprofil** umfasst Merkmalsausprägungen zu leistungsbezogenen und ergonomiebezogenen Anforderungen. Adressiert werden unter anderem die Display-Auflösung, das

Sichtfeld, die Verkabelung sowie die Konnektivität einer Datenbrille. Anhand eines Leitfragenkatalogs wird zudem die Berücksichtigung von Einflüssen aus der Umgebung unterstützt (Abbildung 4-15).

Bewertung der Eignung von Datenbrillen und Handheld-Systemen					
Indikatoren zur Eignungsbewertung		trifft nicht zu	trifft eher nicht zu	trifft eher zu	trifft voll zu
1	Beide Hände müssen zur Tätigkeitsausführung frei sein.	<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input checked="" type="checkbox"/> 3
2	Zwischen Primär- und Sekundäraufgabe wird häufig gewechselt.	<input type="checkbox"/> 0	<input checked="" type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3
3	Die Dauer der Sekundäraufgabe ist kurz.	<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input checked="" type="checkbox"/> 2	<input type="checkbox"/> 3
4	Der Informationsbedarf für einzelne Arbeitsschritte ist gering.	<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input checked="" type="checkbox"/> 3
...	
15	Schwarze Darstellungen (z.B. Schatten) sind nicht erforderlich.	<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input checked="" type="checkbox"/> 2	<input type="checkbox"/> 3
Handlungsempfehlung (Durchschnittswert)					
Handheld-System empfehlenswert		0	1	2	3
Datenbrille empfehlenswert					

Abbildung 4-14: Bewertung der Eignung von Datenbrillen und Handheld-Systemen [RöL21]

Der **Leitfragenkatalog** stellt eine informative Sammlung und Aufbereitung von Einflüssen aus der Umgebung dar. Leitfragen adressieren einzelne Aspekte wie beispielsweise die Lichtverhältnisse. Anhand einer Erläuterung werden mögliche Einflüsse zu diesem Aspekt jeweils informativ beschrieben. Der Leitfragenkatalog dient als Unterstützung bei der Auswahl eines AR-Geräts. Er kann verwendet werden, um Anforderungen an AR-Systeme abzuleiten und darauf basierend einzelne Geräte auszuschließen.

Nr.	Leitfrage	Erläuterung
1	Umgebung	
1.1	Lichtverhältnisse: Unter welchen Lichtverhältnissen muss die AR-Applikation funktionieren?	Die Lichtverhältnisse werden bestimmt durch natürliche (Tageslicht) und künstliche Lichtquellen (z.B. Leuchten). Insbesondere optische Trackingverfahren erfordern zur räumlichen Positionserfassung eine ausreichende Beleuchtung der Umgebung (vgl. Leitfrage 3.1). Hierbei sind zeitliche Schwankungen der Lichtverhältnisse (z.B. abhängig von der Tageszeit) zu berücksichtigen. Einer mangelnden Ausleuchtung kann mit einer zusätzlichen aktiven Beleuchtung entgegen gewirkt werden. Hierzu zählt Licht innerhalb des durch den Menschen wahrnehmbaren Spektrums (380 bis 780nm) sowie nicht sichtbares Licht (z.B. Infrarot), das entweder unstrukturiert oder strukturiert (d.h. durch Projektion bekannter Muster) eingebracht werden kann [SH16, S. 106ff.], [KR13, S. 36].
1.2	Fremdkörper und Feuchtigkeit: Welchen Einflüssen durch Fremdkörper und Feuchtigkeit ist	Abhängig von der Einsatzumgebung kann das AR-System mit Fremdkörpern (z.B. Staub) und Feuchtigkeit (z.B. Sprühwasser) in Kontakt kommen. Um die Funktionsfähigkeit und den sicheren Einsatz zu gewährleisten, muss das AR-System definierte Eigenschaften aufweisen. Die Eignung einzelner Systeme unter den Umgebungsbedingungen wird in der Norm DIN EN 60068-2-36 festgelegt.

Abbildung 4-15: Leitfragenkatalog zur Erhebung von Anforderungen an AR-Systeme [RöL21]

Basierend auf den für das Anforderungsprofil verwendeten Merkmalen und Merkmalsausprägungen beschreibt RÖLTGEN eine Auswahl an konkreten Datenbrillen. **Datenbrillen-Steckbriefe** geben Auskunft darüber, welche Anforderungen eine Datenbrille erfüllt (s. Abbildung A-17 in Anhang A2.2.5). Zudem enthalten die Steckbriefe weitere Informationen zur Marktverfügbarkeit und zum Preis der Datenbrille. RÖLTGEN hat im Rahmen seiner Arbeit Steckbriefe zu sechs gängigen Datenbrillen erstellt. Diese dienen als Wissensbasis für die Auswahl einer geeigneten Datenbrille [RöL21].

Zur Auswahl einer geeigneten Datenbrille erfolgt ein Abgleich des Anforderungsprofils mit den Datenbrillen-Steckbriefen, um die Datenbrille zu identifizieren, die die Anforderungen am besten erfüllt. Zu besseren Auswertung des Abgleichs dient eine **Portfolio-Darstellung** (Abbildung 4-16). Auf der Ordinate wird der Erfüllungsgrad der leistungsbezogenen Anforderungen dargestellt und auf der Abszisse der Erfüllungsgrad der ergonomiebezogenen Anforderungen. Zusätzlich werden der Preis über die Größe und die Marktreife über die Farbe der im Portfolio dargestellten Kreise veranschaulicht. So lassen sich die Ergebnisse des Abgleichs zur Bewertung der Datenbrillen intuitiv verstehen und auswerten. Datenbrillen, die im oberen rechten Bereich des Portfolios positioniert sind, haben hohes Eignungspotential und sollten ausgewählt werden. Kommen mehrere Datenbrillen in Frage, können zudem der Preis und die Verfügbarkeit bei der Entscheidung berücksichtigt werden.

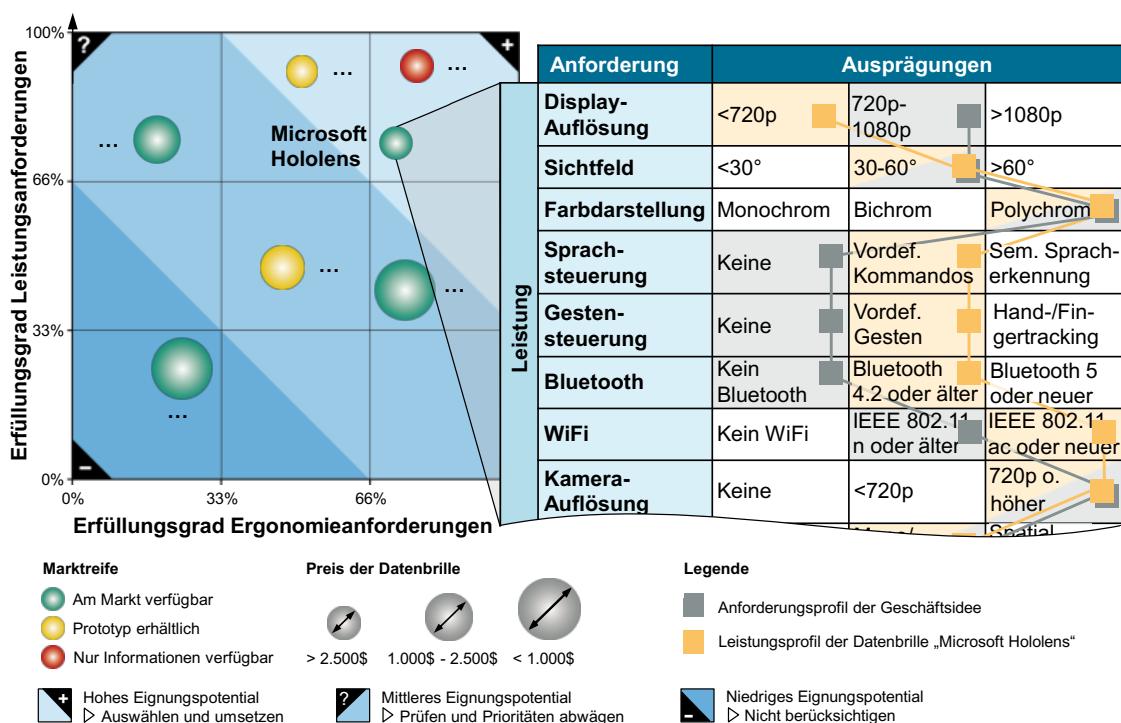


Abbildung 4-16: Bewertung der Anforderungserfüllung von Datenbrillen anhand der Anforderungsschablone [RöL21]

Bewertung: RÖLTGEN beschreibt einen sehr systematischen Ansatz zur Bewertung der Eignung von AR-Geräten für einen Anwendungsfäll. Der Ansatz ähnelt grundsätzlich dem von FELLMANN ET AL. (vgl. Abschnitt 4.2.2.1). RÖLTGEN beschreibt jedoch eine konkrete Vorgehensweise und stellt mit dem Leitfragenkatalog und der Portfolio-Darstellung wichtige Hilfsmittel bereit. Der Leitfragenkatalog regt die Berücksichtigung von Umwelteinflüssen an, eine genaue Einschätzung dieser Einflüsse wird allerdings dem Anwender überlassen. Die grundsätzliche Idee des Abgleichs von Anforderungen mit der Spezifikation von Datenbrillen ist vielversprechend und sinnvoll. Die Anwendung des Ansatzes erfolgt jedoch manuell entsprechend der Vorgehensweise. Sowohl der Abgleich des Anforderungsprofils mit den Steckbriefen als auch die Darstellung im Portfolio müssen durch den Anwender realisiert werden. Ein Software-Werkzeug, das diese Aktivitäten automatisiert übernimmt, wäre sinnvoll und hilfreich. Zudem wurden nur sechs Datenbrillen analysiert und in Steckbriefen dokumentiert.

4.2.2.3 Konfiguration von Fahrsimulatoren nach HASSAN

HASSAN hat im Rahmen seiner Dissertation ein **Rahmenwerk zur Entwicklung rekonfigurierbarer Fahrsimulatoren** erarbeitet [Has14]. Dabei wird davon ausgegangen, dass bereits verschiedene modulare Fahrsimulatoren vorliegen. Ziel des Rahmenwerks ist die bedarfsgerechte Konfiguration weiterer Fahrsimulatoren aus den vorhandenen Modulen für spezifische Simulationsszenarien. Das Rahmenwerk setzt sich aus einem Katalog an Lösungselementen, einer Konfigurationssoftware sowie einem Vorgehensmodell zusammen (Abbildung 4-17).

Das Vorgehensmodell umfasst die *Entwicklungsphase* und *Varianten-Generierungsphase* mit jeweils drei Arbeitsschritten [HGA+15]. Die drei Arbeitsschritte der *Entwicklungsphase* werden einmalig initial von einem Entwickler durchgeführt:

- 1) **Systemspezifikation:** Im ersten Arbeitsschritt wird das System Fahrsimulator mittels CONSENS spezifiziert (s. Abschnitt 4.2.1.1). Dazu werden zunächst Anwendungsszenarien und Anforderungen definiert und Umwelteinflüsse identifiziert. Zudem werden die Funktionshierarchie und die Wirkstruktur eines Fahrsimulators erstellt. Ergebnis ist eine detaillierte Systembeschreibung.
- 2) **Hauptkomponenten:** Basierend auf der Systemspezifikation werden die Hauptkomponenten eines Fahrsimulators identifiziert. Es wird unterschieden zwischen Schlüsselkomponenten und optionalen Komponenten. Zu den Hauptkomponenten zählen unter anderem Eingabe- und Visualisierungsgeräte als Hardware, Fahrzeugmodelle und Intelligent Interfacing Modules (IIM) im Bereich der Software sowie Simulationsrechner und Schnittstellen als Ressourcen. Die Hauptkomponenten definieren die grundlegende Struktur eines Fahrsimulators.

3) **Konfigurationsmechanismus:** In diesem Schritt wird ein Mechanismus entwickelt, mit dem sich basierend auf den definierten Hauptkomponenten und deren Lösungselementen Fahrsimulatoren zusammenstellen lassen. Der Mechanismus beinhaltet eine Konsistenzprüfung und eine Kompatibilitätsprüfung und stellt somit sicher, dass die gewählten Lösungselemente kompatibel sind und über passende Schnittstellen logisch oder funktional miteinander verbunden werden können. Als Grundlage dienen eine *Abhängigkeitsmatrix*, die Abhängigkeiten der Hauptkomponenten untereinander definiert, sowie eine *Konsistenzmatrix*, die in Phase 4 individuell konfiguriert werden muss.

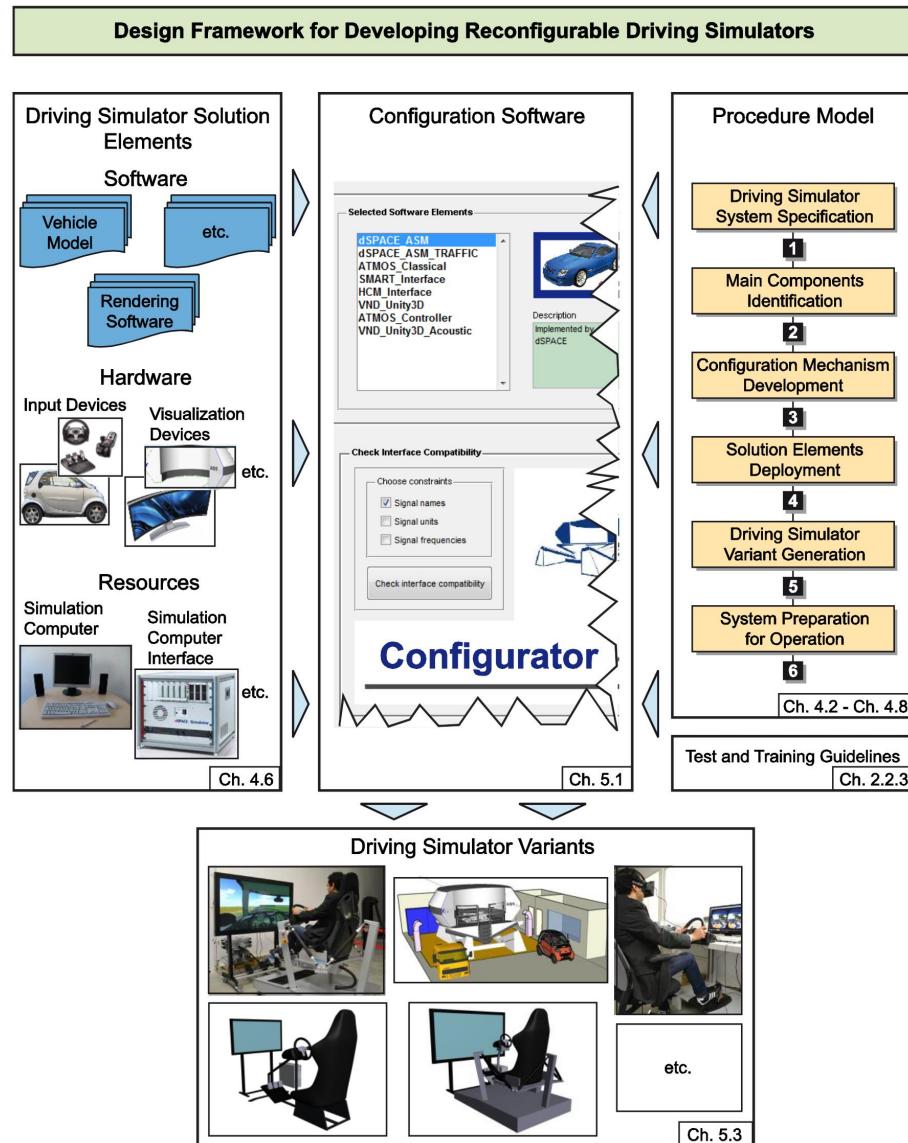


Abbildung 4-17: Rahmenwerk zur Konfiguration von Fahrsimulatoren [Has14]

Nach den ersten drei Arbeitsschritten ist die grundsätzliche Entwicklung des Rahmenwerks abgeschlossen. Die drei Arbeitsschritte der *Varianten-Generierungs-Phase* werden dann jeweils bei Anwendung des Rahmenwerks durch den Anwender durchgeführt. Zur

Unterstützung des Anwenders bei diesen Arbeitsschritten stellt HASSAN ein **Software-Werkzeug** bereit:

- 4) **Lösungselemente:** Für die in Phase 2 identifizierten Hauptkomponenten eines Fahrsimulators werden konkrete Lösungselemente als technische Umsetzungsmöglichkeiten definiert und in einer Datenbank gespeichert. Dazu werden die bestehenden Fahrsimulatoren und die durch sie verfügbaren technischen und funktionalen Elemente verwendet. Ergebnis ist eine Art morphologischer Kasten mit allen Lösungselementen als Datengrundlage der Konfiguration. Zur Vervollständigung des Konfigurationsmechanismus müssen zudem die Konsistenzen zwischen den einzelnen Lösungselementen in der Konsistenzmatrix aus Schritt 3 konfiguriert werden.
- 5) **Variantenerstellung:** Aus dem in Phase 4 erstellen morphologischen Kasten werden zu verwendende Lösungselemente schrittweise durch den Anwender ausgewählt. So erfolgt die eigentliche Systemkonfiguration manuell. Nach der Auswahl erfolgt jedoch eine automatisierte Überprüfung der Kompatibilitäten und der Konsistenz durch das Software-Werkzeug. Ist diese Überprüfung erfolgreich, werden automatisch ein Anschlussplan für die Hardware sowie eine Schnittstellen-Definition für die Software erstellt und als Ergebnis der des fünften Arbeitsschritts ausgegeben.
- 6) **System-Betriebsvorbereitung:** In Schritt 6 erfolgt die technische Vorbereitung des Fahrsimulators. Entsprechend des Anschlussplans wird die Hardware manuell vorbereitet und miteinander verbunden. Die Software-Einrichtung erfolgt automatisiert durch eine von HASSAN entwickelte Assistenz-Software. Software-Lösungselemente werden aus einer Datenbank auf die entsprechenden Ressourcen geladen und automatisch anhand der Schnittstellen-Definition miteinander verknüpft. Am Ende von Schritt 6 ist die Konfiguration und Vorbereitung des Fahrsimulators abgeschlossen.

Bewertung: Das von HASSAN erarbeitete Rahmenwerk ermöglicht die Konfiguration und Entwicklung bedarfsgerechter Fahrsimulatoren. Die Entwicklungs- und die Varianten-Generierungs-Phase können mit der Erstellung einer Wissensbasis und deren Auswertung über die Problemlösungskomponente verglichen werden. Durch die ersten Phasen wird die grundsätzliche Struktur des zu konfigurierenden Systems festgelegt und mögliche Lösungselemente definiert. In den letzten Phasen werden konkrete Lösungselemente entsprechend der individuellen Situation bedarfsgerecht konfiguriert. Jedoch ist die Menge an möglichen Lösungselementen beim Ansatz von HASSAN auf vorhandenen Fahrsimulatoren beschränkt. Es handelt sich somit um keine allgemeingültige Lösung. Die Konfiguration wird durch ein Software-Werkzeug unterstützt, das auch zum Teil automatisiert agiert. Allerdings erfolgt die eigentliche Konfiguration manuell durch den Anwender. Dieser wählt die Lösungselemente der Hauptkomponenten aus. Für eine plausible und sinnvolle Konfiguration ist entsprechend weiterhin Erfahrung und Expertise notwendig. Die automatische Überprüfung der Konsistenz und der Kompatibilität unterstützen die Konfiguration jedoch und nehmen dem Anwender aufwendige Arbeit ab. Grundsätzlich stellt die Kombination aus einer Datenbank mit möglichen Lösungselementen und der

durch ein Software-Werkzeug unterstützen Konfiguration einen interessanten und sinnvollen Ansatz dar.

4.2.2.4 Checklisten-basierte Evaluierung von AR-Displays nach PAELKE ET AL.

PAELKE ET AL. haben eine Checkliste erarbeitet, die Projektmanager, Designer und Entwickler bei der Auswahl von AR-Displays für industrielle AR-Anwendungen unterstützt [PBM+18]. Die Checkliste kann eingesetzt werden, um den Auswahlprozess schneller zu machen und zu verbessern. Sie orientiert sich am internationalen Standard für mensch-zentrierte Gestaltung von interaktiven Systemen [ISO9241-210:2010] und wurde unterstützt durch eine systematische Literaturrecherche sowie eine Experten-Fokusgruppe hergeleitet.

Die Checkliste unterstützt konkret die Analyse und Spezifikation des Anwendungskontextes. Der Anwendungskontext muss zunächst verstanden werden, um anschließend ein passendes AR-Display für den konkreten Anwendungsfall auswählen zu können. Der Anwendungskontext umfasst bei PAELKE ET AL. drei Aspekte: 1) Den Nutzer oder die Nutzergruppe, 2) die Ziele und Aufgaben der Nutzer, 3) die Systemumgebung. Die Checkliste besteht aus drei Tabellen für die drei Aspekte. Jede Tabelle enthält Charakteristika für den jeweiligen Aspekt, die die Auswahl eines AR-Displays beeinflussen können. Für jedes Charakteristikum gibt es zudem eine Erklärung, wann es zutreffend ist für einen konkreten Anwendungsfall. Für die Analyse und Spezifikation des Anwendungskontextes werden die zutreffenden Charakteristika jeweils in den Tabellen angekreuzt.

Die erste Tabelle adressiert die zukünftigen Nutzer des AR-Systems. Sie enthält zunächst Charakteristika bzgl. der Erfahrung der Nutzer, also ob sie Anfänger oder Experten bzgl. AR sind. Zudem adressieren weitere Charakteristika die motorischen Fähigkeiten sowie mögliche Beeinträchtigungen und Angewohnheiten der Nutzer. Wichtig ist bspw., ob Brillenträger unter den Nutzern sind oder eine grundsätzliche Akzeptanz für die Technologie zu erwarten ist.

Die zweite Tabelle adressiert die Aufgaben und Ziele der Nutzer (Tabelle 4-1). Hier werden zunächst die Natur der Aufgabe sowie die Häufigkeit der Nutzung spezifiziert. Zudem wird definiert, ob bereits AR-Systeme im Einsatz sind sowie ob Gefahren und Sicherheitsrisiken bestehen.

Tabelle 4-1: Checkliste für Ziele und Aufgaben in Anlehnung an [PBM+18]

Ziele und Aufgaben	Begründung (kreuzen Sie das Kästchen an, wenn ...)
<i>Natur der Aufgabe</i>	
<input type="checkbox"/> Linear	... die Aufgaben überwiegend einem linearen Verlauf folgen
<input type="checkbox"/> Dynamisch	... die Aufgaben überwiegend einem dynamischen Verlauf folgen
<input type="checkbox"/> Stationär	... die Aufgaben an einen bestimmten Ort gebunden sind
<input type="checkbox"/> Mobil	... die Aufgaben an verschiedenen Orten ausgeführt werden

<i>Häufigkeit der Nutzung</i>	
<input type="checkbox"/> Langfristig	... der Benutzer normalerweise mindestens 2 Stunden pro Schicht mit dem
<input type="checkbox"/> Kurzfristig	... der Benutzer normalerweise weniger als 2 Stunden pro Schicht mit dem System interagiert
<input type="checkbox"/> Häufig	... der Benutzer normalerweise mindestens einmal pro Woche mit dem System interagiert
<input type="checkbox"/> Sporadisch	... der Benutzer normalerweise weniger als einmal pro Woche mit dem System interagiert
<i>Interdependenzen</i>	
<input type="checkbox"/> Andere AR-Systeme	... andere AR-Systeme bereits für ähnliche Aufgaben in der Organisation im Einsatz sind
<i>Gesundheit und Sicherheit</i>	
<input type="checkbox"/> Gefahr	... die Arbeitsumgebung allgemein gefährlich ist
<i>Fehlerpotential</i>	
<input type="checkbox"/> Sicherheitskritisch	... Fehler schwerwiegende Konsequenzen haben

Die dritte Tabelle dient der Spezifikation der Systemumgebung (Tabelle 4-2). Hier werden Anforderungen an die Hardware und Software sowie Eigenschaften der physischen Umgebung definiert. Zudem wird spezifiziert, welche sozialen und kulturellen Aspekte relevant sind.

Tabelle 4-2: Checkliste für Systemumgebungen in Anlehnung an [PBM+18]

Umgebung	Begründung (kreuzen Sie das Kästchen an, wenn ...)
<i>Hardware</i>	
<input type="checkbox"/> Hohe Auflösung	... die Informationen in hoher Auflösung gezeigt werden müssen
<input type="checkbox"/> Hohe Geschwindigkeit	... die Informationen schnell verarbeitet werden müssen
<input type="checkbox"/> Multimodalität	... mehr als eine Interaktionsmodalität (unabhängige Kanäle für Ein- und Ausgabe) erforderlich ist
<input type="checkbox"/> Niedrige Kosten	... das System kostengünstig sein muss
<input type="checkbox"/> Wenig Wartung	... das System zuverlässig mit wenig Wartung laufen soll
<input type="checkbox"/> Mobiles System	... das System transportierbar sein muss
<i>Software</i>	
<input type="checkbox"/> Open Source	... der Schwerpunkt auf Kosteneffizienz und der Vermeidung von Anbieterabhängigkeit liegen soll
<input type="checkbox"/> Proprietär	... der Schwerpunkt auf Qualität und zuverlässigen Anbietern einschließlich Supportdiensten liegen soll
<i>Physische Umgebung</i>	
<input type="checkbox"/> Thermische Bedingungen	... deutlich wärmer oder kälter als Raumtemperatur ist
<input type="checkbox"/> Beleuchtung	... es sehr hell ist
<input type="checkbox"/> Lärm	... es lauter als normal ist
<input type="checkbox"/> Robustheit	... die Umgebung allgemein rau ist
<input type="checkbox"/> Reinraum	... die Tätigkeit nur in Reinräumen durchgeführt werden kann

<i>Soziale Aspekte</i>	
<input type="checkbox"/> Datenschutz	... das AR-System mit sensiblen Daten wie persönlichen Nutzerdaten umgehen muss
<i>Kulturelle Aspekte</i>	
<input type="checkbox"/> Diebstahlrisiko	... das System an Orten eingesetzt wird, an denen das Diebstahlrisiko hoch ist
<input type="checkbox"/> Fluktuation der Arbeitskräfte	... die Nutzer des AR-Systems häufig wechseln

Die ausgefüllte Checkliste bzw. die ausgefüllten Tabellen dienen anschließend als Grundlage für die Auswahl eines AR-Displays. Dazu gibt es zwei Optionen:

- **Option A:** Die Checkliste wird an Experten im Bereich AR übergeben. Die Experten leiten Anforderungen daraus ab und geben eine Empfehlung für ein passendes AR-Display zurück.
- **Option B:** Die Auswahl eines AR-Displays erfolgt durch den Anwender der Checkliste selbst. Zur Unterstützung der Entscheidung geben PAELKE ET AL. zusätzliche Informationen, Erklärungen und Einschätzungen zu den verschiedenen Charakteristika in Textform [PBM+18]. Unter anderem gehen sie darauf ein, dass Sehbehinderungen bei den Nutzern ein Hindernis bei stereoskopischen AR-Displays, also vor allem AR-Brillen, sind.

Bewertung: Die Checkliste von PAELKE ET AL. bestehend aus drei Tabellen stellt ein gutes Hilfsmittel zur Beschreibung des Anwendungskontextes dar. Sie unterstützt, ein Verständnis aufzubauen für relevante Aspekte bei der Auswahl eines AR-Displays. Das Ausfüllen der Tabellen ist aufgrund der Erklärungen für die einzelnen Charakteristika sehr gut möglich, auch ohne AR-Expertise. Durch die Checkliste wird jedoch lediglich ein Teil der AR-Display-Auswahl unterstützt. Die eigentliche Auswahl basierend auf der Checkliste wird von PAELKE ET AL. nicht fokussiert und nur geringfügig adressiert. Option A für die Nutzung der Checkliste erfordert die Einbeziehung externer Experten. Die von den Autoren ergänzend beschriebenen Informationen für Option B, also die eigenständige Auswahl, stellen lediglich eine Unterstützung dar. Einerseits sind für fundierte Entscheidungen umfangreichere Erklärungen und weiteres Wissen notwendig. Andererseits werden die Informationen in Fließtext-Form beschrieben und müssen vom Anwender entsprechend gelesen, analysiert und individuell auf die Checkliste angewendet werden. Eine definierte Vorgehensweise dazu wird von den Autoren nicht beschrieben. Zudem stellt die Checkliste ein manuelles Hilfsmittel dar, das durch den Anwender händisch ausgefüllt und ausgewertet werden muss. Die von PAELKE ET AL. erarbeiteten Charakteristika sowie die für AR-Display-Auswahl beschriebenen Zusatzinformationen können jedoch für die Erarbeitung einer Lösung in Handlungsfeld 2, insbesondere der Erstellung einer Wissensbasis, herangezogen werden.

4.2.2.5 AR-System-Konfiguration nach SCHILLING

SCHILLING hat mit seiner Dissertation das Ziel verfolgt, potenziellen Anwendern von AR ein Handwerkszeug zur Verfügung zu stellen, mit dem ein AR-System entsprechend einer spezifischen Aufgabenstellung konfiguriert werden kann. Ergebnis der Dissertation ist eine Konfigurationssystematik für AR-Systeme im Kontext der Produktentstehung [Sch08a]. Die Grundlage der Systematik bildet eine Variantenmatrix, die SCHILLING basierend auf einer Analyse der verschiedenen Systemkomponenten eines AR-Systems definiert. Neben *Technik*-Komponenten enthält die **Variantenmatrix** auch Aspekte der *Aufgabe* und der *Funktion* (Abbildung 4-18).

Aufgabe	Design & Konstruktion	Berechnung	Planung & Logistik	Prototypenbau	Erprobung	Fertigung	Qualitäts- sicherung	
	<ul style="list-style-type: none"> - Entwurf - Prüfung - Varianten - Baureihen 	<ul style="list-style-type: none"> - FEM - MKS - Strömung - Kinematik 	<ul style="list-style-type: none"> - Hallenbau - Arbeitsabläufe - Transportwege 	<ul style="list-style-type: none"> - Entwurf - Vergleich - Dokumentation 	<ul style="list-style-type: none"> - Statusinformationen 	<ul style="list-style-type: none"> - Fertigungs simulation - Maschineneinrichtung - Wartung & Schulung 	<ul style="list-style-type: none"> - Soll/IlsVergleich - Fehlererfassung - Produktion - Arbeitsfolge - Logistik 	
Funktion	Anzeige	Anwenderführung	Interaktion	Dokumentation	Messen & Vergleichen			
	<ul style="list-style-type: none"> - Werte - CAD-Daten - Messdaten - Hewise 	<ul style="list-style-type: none"> - Navigation - Arbeitsfolgen - intuitiv - Aufgabenabhängig 	<ul style="list-style-type: none"> - keine - Datenhandschuh - Hand und Gesten ohne Eingabegeräte - Maus, Spacemouse, und andere Eingabegeräte 	<ul style="list-style-type: none"> - Werte - Modelle - Bilder - Videos 	<ul style="list-style-type: none"> - Real-Virtuell - Virtuell-Virtuell - Real-Real - optische Sichtkontrolle - Qualitativ/quantitativ 			
Technik	Kamera	Tracking	Displaysystem	Szenengenerator	Interaktionstool	Software		
	<ul style="list-style-type: none"> - Video PAL - Video HD - Firewire - S-Video - Netzwerk - USB - Cameralink - Fotoapparat 	<ul style="list-style-type: none"> - mechanisch - intertial - magnetisch - ultraschall - optisch - elektromagnetisch - hybrid 	<ul style="list-style-type: none"> - HMD (OST) - HMD (VST) - VRD - PDA - TFT - Projektion - Aufprojektion - Lautsprecher - Force-feedback - Touch-Feedback 	<ul style="list-style-type: none"> - 1-Chip-System - mobile Systeme - Standard-PC - Multi-CPU/GPU-Systeme - Cluster - Verteilte Systeme 	<ul style="list-style-type: none"> - Messstift - Handschuh - PC-Maus - Sprache 	<ul style="list-style-type: none"> - AR-Toolkit - Metaio - ARVIKA - VRCom - Studierstube 		

Abbildung 4-18: AR-System- Variantenmatrix in Anlehnung an SCHILLING [Sch08a]

Zudem hat SCHILLING **Einflussfaktoren auf AR im Produktentstehungsprozess** zu den fünf Aspekten Integration, Betrieb, Anwender, Hardware und Software definiert und untersucht (Abbildung 4-19). Die Einflussfaktoren werden jedoch nicht weiter im Detail beschrieben. Basierend auf den identifizierten Einflussfaktoren enthält die Systematik eine **Wechselwirkungsmatrix**. In dieser ist definiert, welche Wechselwirkungen zwischen Komponenten und Einflussfaktoren bestehen. Als weiteres Hilfsmittel dient ein **Fragenkatalog**, der die Auswahl einzelner Komponenten sowie die Definition vorliegender Einflussfaktoren anleitet. Der Fragenkatalog gibt die Auswahlmöglichkeiten an und informiert den Anwender über bestehende Wechselwirkungen zu anderen Komponenten und Einflussfaktoren.

Vor der Anwendung der Systematik sind jeweils drei Schritte durchzuführen:

- 1) Anforderungen an das AR-System definieren
- 2) Fragenkatalog entsprechend dem Stand der Technik aktualisieren
- 3) Fragen entsprechend ihrer Priorität ordnen

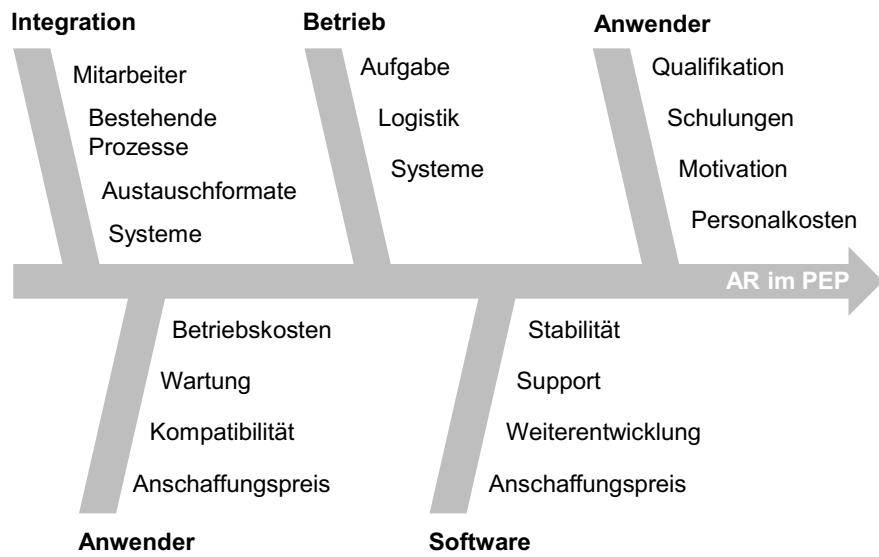


Abbildung 4-19: Einflussfaktoren auf AR im PEP in Anlehnung an [Sch08a]

Durch den dritten Schritt werden die Fragen in eine Reihenfolge gebracht und so der Ablauf der System-Konfiguration festgelegt. Die **System-Konfiguration** erfolgt anhand des in Abbildung 4-20 dargestellten Prinzips. Es wird begonnen mit der Beantwortung der ersten Frage in Zeile 1 des durch den Anwender geordneten Fragenkatalogs. Aus den Auswahlmöglichkeiten wird eine Ausprägung der System-Komponente oder des Einflussfaktors durch den Anwender ausgewählt. Anschließend werden die Wechselwirkungen ausgewertet. Wechselwirkungen beeinflussen bzw. reduzieren die Auswahlmöglichkeiten der weiteren Fragen im Fragenkatalog. Diese Reduzierung wird manuell vorgenommen. Zudem wird die Reihenfolge der Fragen angepasst. Ist die in der Reihenfolge der Fragen als nächstes anstehende Komponente nicht durch die Wechselwirkungen der zuvor konfigurierten Komponente betroffen, wird die Frage nach hinten verschoben. Die durch die Wechselwirkungen beeinflussten Komponenten und Einflussfaktoren werden stattdessen zuerst behandelt. Diese Vorgehensweise soll eine bessere Berücksichtigung von Inkompatibilitäten ermöglichen und die Konsistenz der Konfiguration gewährleisten. Nachdem Schritt-für-Schritt alle Fragen im Fragenkatalog abgearbeitet wurden, ist das AR-System konfiguriert.

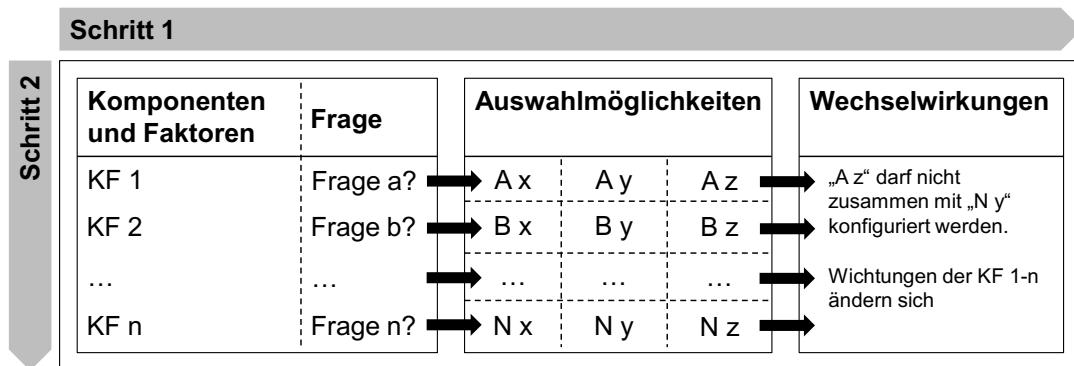


Abbildung 4-20: AR-System-Konfigurationssystematik in Anlehnung an [Sch08a]

Bewertung: SCHILLING beschreibt eine systematische Vorgehensweise zur Konfiguration von AR-Systemen unter Berücksichtigung verschiedener Einflussfaktoren und Wechselwirkungen zwischen Komponenten. Die zugrundeliegende Variantenmatrix ist jedoch nicht sehr detailliert und für die Varianz heutiger AR-Systeme nicht ausreichend. Die Wechselwirkungen werden grundsätzlich umfangreich adressiert. Allerdings wird nur angegeben, dass eine Wechselwirkung zwischen zwei Komponenten oder Einflussfaktoren bestehen kann. Der genaue Einfluss muss letztendlich durch den Anwender eingeschätzt werden. Somit bietet der Fragenkatalog nur bedingt Unterstützung. Er leitet den Anwender lediglich an. So bleibt die Auswahl der System-Komponenten als wichtigster Schritt der Konfiguration komplett dem Anwender überlassen. Dieser muss entsprechend weiterhin über Erfahrung und Expertise zu AR verfügen, um eine plausible Auswahl treffen zu können. Übergreifend handelt es sich bei der Systematik zudem um eine manuelle Lösung. Ein Software-Werkzeug, das die Auswertung der Wechselwirkungen automatisiert übernimmt, wäre hilfreich.

4.2.2.6 Wissensbasierte Produktionssystemkonzipierung nach BAUER

BAUER hat ein **Planungswerkzeug** zur wissensbasierten Produktionssystemoptimierung entwickelt [Bau15]. Das Planungswerkzeug unterstützt den Anwender bei der Entwicklung neuer Produktionssysteme basierend auf **Lösungswissen**, das in vorangegangenen Entwicklungsprojekten erarbeitet wurde. Das Planungswerkzeug entspricht einem wissensbasierten System. Es basiert auf den folgenden vier Bestandteilen (Abbildung 4-21):

- **Spezifikationstechnik:** Ein wesentlicher Bestandteil des Planungswerkzeugs ist eine Spezifikationstechnik für Produktionssysteme. Hierzu wird die zuvor beschriebene Spezifikationstechnik CONSENS verwendet. Wie in Abschnitt 4.2.1.1 beschrieben, umfasst CONSENS verschiedene Partialmodelle. BAUER verwendet die für Produktionssysteme relevanten Partialmodelle *Prozesse* und *Ressourcen* (Abbildung 4-9). Über diese Partialmodelle können Produktionssysteme inklusive Materialelementen, Materialflüssen, Werkzeugen, Akteuren, Vorrichtungen und weiteren Aspekten modelliert werden. Für den Einsatz der Modelle im Kontext des Planungswerkzeugs definiert BAUER Richtlinien und Bedingungen über eine Vielzahl an Regeln. Diese Regeln ermöglichen ein einheitliches Verständnis der Modellierung.
- **Ontologie:** Über die sogenannte ProSyCo-Ontologie (Production System Concept) soll dem Anwender Zugang zu Lösungswissen ermöglicht werden. Die Ontologie definiert, wie die semantische Beschreibung von Lösungselementen erfolgt. Dazu besteht die Ontologie aus mehreren Teiltonologien. Unter anderem hat BAUER eine Ontologie zur Prozessmodellierung und eine Ontologie zur Ressourcenmodellierung erarbeitet. Die Teiltonologien sind teilweise sehr komplex, beinhalten vielfältige Klassen und mehrstufige Generalisierungsebenen. Sie ermöglichen eine sehr detaillierte Spezifikation von verschiedenen Aspekten eines Produktionssystems. Ein be-

sonderer Fokus liegt auf den Schnittstellen und Wechselwirkungen zu anderen Komponenten, die für die modulare Konzipierung eines Produktionssystems von enormer Bedeutung sind. Über die ProSyCo-Ontologie lässt sich so zum einen Fachwissen zu Materialien und Prozessen repräsentieren. Zum anderen dient die Ontologie der Dokumentation von konkreten Lösungselementen. Die Lösungselemente entsprechen beispielsweise modularen Fertigungs- oder Montagesystemen. Entsprechend der Ontologie wird eine Wissensbasis aufgebaut, die die Grundlage des wissensbasierten Systems zur Produktionssystemkonzipierung bildet.

- **Problemlösungskomponente:** Die Problemlösungskomponente ermöglicht den situationsbedingten Zugriff auf das Lösungswissen. Sie unterstützt den Anwender bei der Konzipierung eines neuen Produktionssystems. Unter anderem wird der Anwender bei der Definition notwendiger Produktionsprozesse und Prozessparameter zur Herstellung eines Materials unterstützt. Zudem können gezielt Fertigungs- und Montagesysteme zur Realisierung eines Prozesses aus der Menge an Lösungselementen identifiziert werden. Über die Ontologie sind alle notwendigen Informationen einheitlich beschrieben und können gezielt abgefragt werden. Die Problemlösungskomponente unterstützt den Anwender bei gezielten Anfragen an die Wissensbasis sowie der Aufbereitung und Überführung der Lösungselemente in die Modellierung des neuen Produktionssystems mit CONSENS. Die Spezifikation eines neuen Produktionssystems muss so nicht komplett erfolgen. Stattdessen können Teile der CONSENS-Modelle aus vorangegangenen Projekten aufwandsarm übernommen werden. Die Problemlösungskomponente wurde von BAUER prototypisch in Form eines Software-Werkzeugs implementiert.
- **Vorgehensmodell:** Der Einsatz des Planungswerkzeugs zur wissensbasierten Produktionssystemkonzipierung wird durch ein übergreifendes Vorgehensmodell angeleitet.

Bewertung: Das Planungswerkzeug von BAUER bietet Entwicklern von Produktionssystemen eine sehr gute Unterstützung. Vorhandenes Wissen kann basierend auf der Ontologie eindeutig und einheitlich beschrieben und wiederverwendet werden. Durch den Einsatz semantischer Technologien erfolgt die Identifikation und Verarbeitung des Lösungswissens softwarebasiert und teilautomatisiert. Dies stellt einen enormen Vorteil gegenüber anderen Ansätzen wie der AR-System-Konfiguration nach BAUER oder der Bewertung von AR-Geräten nach RÖLTGEN dar. Der Fokus des Planungswerkzeugs liegt auf der Nutzung von Wissen für die Spezifikation eines neuen Systems. Die Wissensbasis des Planungswerkzeugs enthält daher spezifisches Wissen zu konkreten Lösungselementen. Es handelt sich nicht um allgemeines Wissen zur Auswahl von System-Komponenten. Der Ansatz geht über die reine Auswahl von System-Komponenten bzw. eines Geräts, wie sie im Fall der AR-System-Konfiguration angestrebt wird, hinaus. Zudem erfolgt keine Berücksichtigung möglicher äußerer Einflüsse. Das grundlegende Konzept stellt jedoch einen vielversprechenden Ansatz dar, der sich mit verschiedenen Anpassungen auf die AR-System-Konfiguration übertragen lässt.

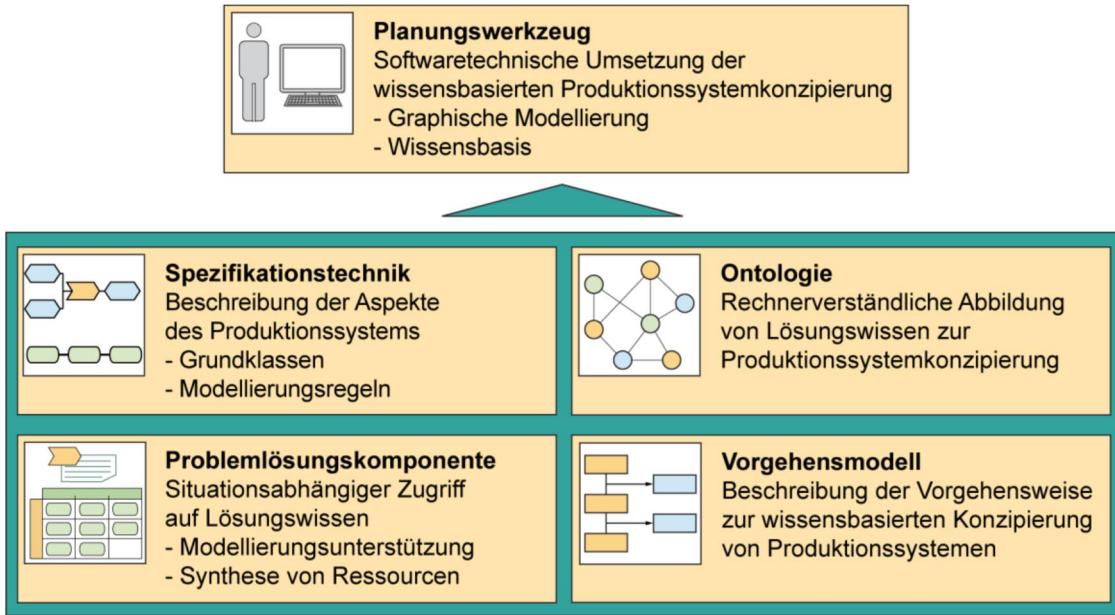


Abbildung 4-21: Struktur der wissensbasierten Produktionssystemkonzipierung nach BAUER [Bau15]

4.3 Aufwandsarme Entwicklung von AR-Validierungssoftware

Im dritten Handlungsfeld dieser Arbeit sollen Unternehmen in die Lage versetzt werden, individuelle Validierungsanwendungen für ausgewählte AR-Geräte eigenständig zu entwickeln. Die Entwicklung muss daher für Unternehmen möglichst aufwandsarm ermöglicht werden. Zunächst wird in Abschnitt 4.3.1 die Entwicklung von AR-Anwendungen grundsätzlich untersucht. Bereits existierende Ansätze zur aufwandsarmen Entwicklung von AR-Anwendungen werden in Abschnitt 4.3.2 behandelt. Das Kapitel schließt ab mit einer Untersuchung bestehender AR-Anwendungssoftware im Kontext der Produktvalidierung in Abschnitt 4.3.3.

4.3.1 Entwicklung von AR-Anwendungen

Zunächst wird in Abschnitt 4.3.1.1 der grundsätzliche Entwicklungsprozess einer AR-Anwendung analysiert. Anschließend werden in Abschnitt 4.3.1.2 Software Development Kits (SDK) als Grundlage der Entwicklung untersucht.

4.3.1.1 AR-Entwicklungsprozess

Die Entwicklung einer AR-Anwendung entspricht grundsätzlich der Entwicklung einer Softwareanwendung. Als methodischer Rahmen für die Softwareentwicklung wurde der **Software Development Lifecycle (SDLC)** definiert. Der SDLC beschreibt den grundlegenden Software-Entwicklungsprozess inklusive notwendiger Aktivitäten sowie zu beachtende Richtlinien, Standards und Prinzipien [Mur12]. Die Aktivitäten innerhalb des

SDLC sind klar definiert, die Aufteilung auf übergeordnete Phasen variiert jedoch. So werden je nach Publikation sechs oder sieben Phasen des SDLC beschrieben [Shy17], [BP13]. Eine Variante des SDLC sind die folgenden sieben Phasen (Abbildung 4-22):

- **Planung:** Die zu entwickelnde Software und das entsprechende Entwicklungsprojekt werden geplant. Dazu werden unter anderem Risiken analysiert sowie ökonomische, operative und technische Machbarkeitsstudien durchgeführt.
- **Analyse:** Anforderungen an die Software und die Entwicklung werden spezifiziert.
- **Design:** Basierend auf den Anforderungen werden die Software-Architektur, die Funktionalität sowie das Erscheinungsbild konzipiert.
- **Implementierung:** Basierend auf dem Konzept erfolgt die eigentliche Entwicklung der Software unter Berücksichtigung von Richtlinien und Vorgaben.
- **Testen:** Anhand von Tests wird die Software auf korrekte Funktionalität entsprechend den Anforderungen überprüft und bei Bedarf Fehler behoben.
- **Deployment:** Die Software wird auf den entsprechenden Endgeräten installiert oder als fertige Software am Markt angeboten.
- **Wartung und Updates:** Die Software muss je nach Art und Einsatzszenario regelmäßig gewartet und aktualisiert werden.

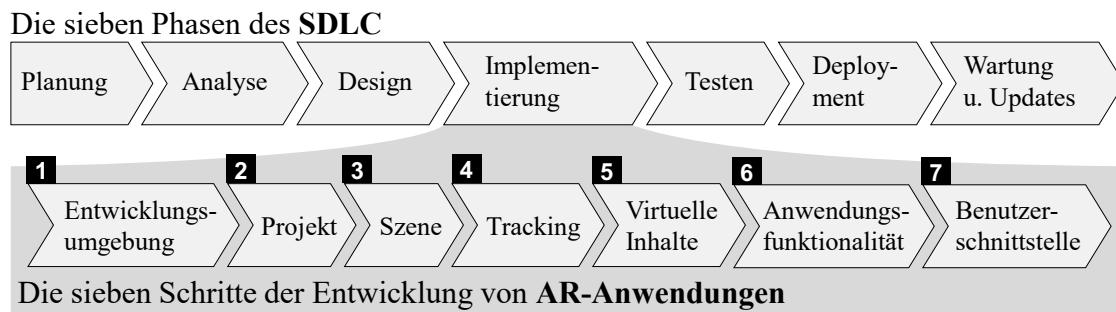


Abbildung 4-22: Die sieben Phasen des SDLC und die sieben wesentlichen Schritte der spezifischen Entwicklung von AR-Anwendungen

Der SDLC wird als sukzessive Abfolge der Phasen beschrieben. Dies dient lediglich als eine Art Leitlinie. Je nach Projekt werden die Phasen in unterschiedlichen Reihenfolgen, mehrfach oder parallel zueinander bearbeitet [Jor19-ol]. Zudem existieren verschiedene Modelle für den konkreten Prozessablauf, die am SDLC orientiert sind. Beispiele sind das Wasserfall-Modell, Scrum oder auch das V-Modell der Softwareentwicklung [BP13].

Die grundsätzlichen Phasen des SDLC sind auch bei der Entwicklung einer AR-Anwendung zu durchlaufen. Die Implementierungsphase gestaltet sich aufgrund der funktionalen Besonderheiten von AR jedoch sehr spezifisch. Es existiert zum aktuellen Zeitpunkt kein formaler Prozess oder Standard zur Entwicklung von AR-Anwendungen. Im Internet

ist jedoch eine Vielzahl an Online-Kursen und Tutorials verfügbar, die die Entwicklung beschreiben und Entwickler bei der Erstellung eigener Anwendungen anleiten [Buc19-ol], [Bri21-ol], [Gaj20-ol], [Mic21-ol], [Per20], [PTC21-ol], [Tut21-ol]. Aus diesen Quellen können die in Abbildung 4-22 dargestellten sieben Schritte der spezifischen **Entwicklung von AR-Anwendungen** abgeleitet werden:

- 1) **Entwicklungsumgebung:** Zur Entwicklung einer AR-Anwendung wird eine spezielle Entwicklungsumgebung benötigt. Diese stellt eine grafische Benutzeroberfläche sowie grundlegende Funktionen für die Entwicklung bereit. Je nach Plattform, für die die AR-Anwendung implementiert werden soll, müssen spezielle Pakete installiert werden. Eine der gängigsten Entwicklungsumgebungen ist *Unity* (s. Anhang A2.3.1). Neben der eigentlichen Entwicklungsumgebung sind ein Code-Editor (bspw. Microsoft® *Visual Studio*⁴) und unter Umständen plattformspezifische Werkzeuge wie *Xcode*⁵ zu installieren.
- 2) **Projekt:** Innerhalb der Entwicklungsumgebung muss ein Projekt angelegt werden. Es stehen meist Vorlagen bereit, bspw. für eine „3D-Anwendung“. Im Projekt müssen je nach AR-Endgerät spezielle Software-Frameworks oder Software Development Kits (SDK) als Grundlage der Entwicklung geladen werden. Neben SDKs für AR-Grundlagen können zusätzliche SDKs zur Unterstützung der Anwendungssoftware eingebunden werden (s. Abschnitt 4.3.1.2).
- 3) **Szene:** Über eine oder mehrere Szenen werden die virtuellen Inhalte inkl. der gesamten Funktionalität sowie der Interaktion beschrieben. Szenen sind hierarchisch als Graphen aufgebaut. Sie enthalten sowohl visuelle als auch funktionale nicht sichtbare Objekte. Im dritten Schritt finden entsprechend der grundlegende Aufbau und die Konfiguration der Szene als Rahmen der Implementierung statt.
- 4) **Tracking:** Eine Besonderheit von AR-Anwendungen ist das Tracking. Die Bewegung eines AR-Endgeräts in der realen Umgebung muss präzise verfolgt werden, um virtuelle Inhalte positionsgenau darstellen zu können. Es existieren verschiedene Tracking- und Positionierungsansätze (vgl. Abschnitte 3.4.1 und Anhang A2.2.4). In Schritt 4 wird das Tracking in der Szene konfiguriert und alle notwendigen Vorbereitungen getroffen.
- 5) **Virtuelle Inhalte:** Die Darstellung der virtuellen Inhalte wird vorbereitet. Zum einen sind die virtuellen Inhalte bspw. in Form von 3D-Modellen zu erstellen. Zum anderen müssen die Inhalte in der Szene integriert werden. Dies kann entweder statisch erfolgen.

⁴ Microsoft® *Visual Studio* ist verfügbar unter <https://visualstudio.microsoft.com/de/>

⁵ Bei *Xcode* handelt es sich um eine spezifische Integrierte Entwicklungsumgebung (IDE) von Apple für *iOS*- und *macOS*-Software

gen oder dynamisch durch Realisierung entsprechender Schnittstellen zu einer Datenbank oder einem Server. Die Darstellung der virtuellen Inhalte wird im Szenengraphen integriert und vorbereitet.

- 6) **Anwendungsfunktionalität:** Im sechsten Schritt werden die Anwendungsfunktionalitäten und die damit verbundene Funktionslogik der AR-Anwendung implementiert und im Szenengraphen integriert. Einzelne Funktionalitäten werden programmiert und logisch sowie funktional miteinander in Verbindung gebracht. Meist werden für die Steuerung der Funktionalität spezielle Manager-Objekte im Szenengraphen hinzugefügt. In den Manager-Objekten laufen alle relevanten Informationen, unter anderem über Zustände der Anwendung sowie Benutzereingaben, zusammen.
- 7) **Benutzerschnittstelle:** Aufbauend auf der Anwendungsfunktionalität wird die Benutzerschnittstelle realisiert. Sie ermöglicht die Interaktion zwischen Software und Benutzer. Grundlegende Elemente für Benutzerschnittstellen werden von Entwicklungsumgebungen bereitgestellt. Spezielle Interaktionslösungen wie beispielsweise Gesteuerung sind zudem über SDKs verfügbar (s. Abschnitt 4.3.1.2).

Während der Implementierung erfolgen bereits Tests einzelner Aspekte. AR-spezifische Aspekte wie das Tracking oder die Interaktion über Gesten sind nur auf dem jeweiligen AR-Endgerät zu testen. Daher wird die AR-Anwendung für aussagekräftige Tests meist bereits während der Entwicklung auf dem Endgerät installiert. Spätestens nach Abschluss der Entwicklung erfolgt die Installation in der Deployment-Phase. In dieser Phase sind je nach AR-Endgerät die zusätzlichen in Phase 1 genannten Software-Werkzeuge notwendig. Eine Installation auf *iOS*- oder *macOS*-basierten Geräten benötigt beispielsweise *Xcode*. Die Installation auf einer Microsoft HoloLens läuft dagegen über *Visual Studio*.

Bewertung: Der SDLC beschreibt den übergeordneten Prozess der Software-Entwicklung. Die ersten drei Phasen (Planung, Analyse, Design) werden durch die Planung der AR-basierten Validierung in den Handlungsfeldern 1 und 2 abgedeckt. In Handlungsfeld 3 steht darauf aufbauend die Implementierung im Fokus. Eng mit der Implementierung verbunden findet das Testen statt. Das Handlungsfeld 3 soll zudem auch die Phase Deployment unterstützen. Wartung und Updates der AR-Anwendung können bei Bedarf ebenfalls anhand der angestrebten Lösung durchgeführt werden. Besonders relevant für das Handlungsfeld 3 sind die sieben Schritte der spezifischen Entwicklung von AR-Anwendungen. Ziel des Handlungsfelds ist eine Lösung zur aufwandsarmen Entwicklung individueller AR-Anwendungen im Kontext Validierung. Bei der Entwicklung sind die sieben Schritte zu durchlaufen. Entsprechend sollte die angestrebte Lösung sich an diesen Schritten orientieren und sie gezielt unterstützen. Die Schritte können somit als Grundlage für eine strukturiertes Vorgehen genutzt werden.

4.3.1.2 Software Development Kits (SDK)

Software Development Kits (SDK) stellen Sammlungen von Programmierfunktionalitäten in einem installierbaren Paket bereit. Dabei handelt es sich um Funktionalitäten, die von Software-Entwicklern als Grundlage für die Entwicklung individueller Anwendungen verwendet werden können. SDKs adressieren meist spezifische Themen- oder Anwendungsfelder. SDKs können in der Entwicklungsumgebung beim Anlegen eines Projekts integriert und bei der Entwicklung von AR-Anwendungen verwendet werden.

Es existiert eine große Anzahl an SDKs, die **grundlegende AR-Funktionalitäten** bereitstellen. In verschiedenen wissenschaftlichen Veröffentlichungen wurde bereits versucht, AR-SDKs zu analysieren und gegenüberzustellen [AG15], [VP21]. Allerdings decken diese Arbeiten nur eine kleine Auswahl an SDKs ab oder sind bereits veraltet. Die ersten AR-SDKs wurden unabhängig von Geräten entwickelt und fokussierten die Erkennung von Markern in einem Videobild zur Positionierung von AR-Inhalten. Vorreiter in diesem Bereich war die Metaio GmbH aus München [LBM+09]. Mit dem *Metaio SDK* veröffentlichte die GmbH eines der ersten SDKs, das Marker-Tracking zur Darstellung von virtuellen Inhalten auf mobilen Geräten ermöglichte. Mittlerweile decken AR-SDKs vielfältige für AR relevante Funktionalitäten ab. Neben verschiedenen Tracking-Lösungen zählen dazu unter anderem Möglichkeiten der Umgebungserkennung und -rekonstruktion, AR-spezifische Interaktionsfunktionalitäten sowie eine automatische Bestimmung der Lichtverhältnisse. Die bekanntesten Anbieter von geräteunabhängigen AR-SDKs sind PTC mit dem *Vuforia AR SDK*⁶ sowie *Wikitude*⁷. Beide bieten ähnliche Funktionalitäten, können in *Unity* verwendet werden und unterstützen alle gängigen Plattformen. Sie sind beim kommerziellen Einsatz jedoch mit Lizenzkosten verbunden. Gleiches gilt für das Tracking-SDK *visionLib* [Vis22-ol]. Bei *visionLib* handelt es sich um ein neuartiges SDK, das fortschrittliches 3D-Objekt-Tracking basierend auf CAD-Daten ermöglicht.

Neben den geräteunabhängigen SDKs gibt es eine Reihe an **SDKs für spezielle Plattformen**. Diese SDKs werden von Gerätsherstellern entwickelt und angeboten und stellen auf die Geräte angepasste Funktionalitäten bereit. Bekannteste Beispiele sind das *ARKit* von Apple® und *ARCore* von Google®. *ARKit* und *ARCore* bieten ähnliche Funktionalitäten. Sie sind jedoch jeweils auf die jeweiligen Geräte und Betriebssysteme angepasst und optimiert. Beide SDKs ermöglichen markerloses und 2D- sowie 3D-Marker-basiertes Tracking basierend auf der Kamera und weiteren Sensoren der mobilen Geräte. Neben grundlegenden Funktionalitäten werden zudem innovative Ansätze wie Motion Capture

⁶ Verfügbar unter <https://www.ptc.com/en/products/vuforia>

⁷ Verfügbar unter <https://www.wikitude.com/>

und dynamische Berechnung von Verdeckungen (Okklusion) unterstützt. *ARKit* und *ARCore* sind kostenlos zu nutzen. Neben diesen beiden existieren weitere gerätegebundene SDKs, unter anderem *Lumin SDK*⁸ für die AR-Brillen der Firma Magic Leap®.

Microsoft® stellt mit dem ***Mixed Reality Toolkit (MRTK)*** das umfangreichste SDK zur Entwicklung von Mixed Reality-Anwendungen in *Unity* bereit. Es kann kostenlos heruntergeladen und verwendet werden. COULTER ET AL. nennen vier wesentliche Merkmale des *MRTK* [SCP+21-ol]:

- Das *MRTK* stellt ein plattformübergreifendes Eingabesystem sowie **modulare Bausteine** für eine räumliche dreidimensionale Benutzeroberfläche bereit. Die modularen Bausteine bieten vielfältige Möglichkeiten und vereinfachen die Entwicklung enorm. Sie adressieren unter anderem die in Abbildung 4-23 dargestellten Aspekte. Weitere Details zu den Bausteinen und Funktionalitäten des *MRTK* werden in Anhang A2.3.3 beschrieben.
- Das *MRTK* ermöglicht schnelle **Simulation** von AR-Anwendungen direkt im *Unity*-Editor. Unter anderem erlaubt die Simulation von Gesten über Tastatur und Maus ein aufwandsarmes Testen der Entwicklung ohne Installation auf dem Endgerät.
- Das *MRTK* bietet ein **Framework**, das durch eigene Komponenten erweitert werden kann. Zudem ist ein Austausch von Komponenten zwischen Entwicklern möglich.
- Das *MRTK* bietet eine umfangreiche **Plattform-Unterstützung**. Neben den Microsoft-eigenen HoloLens-Geräten und Windows Mixed Reality-Headsets laufen *MRTK*-basierte Anwendungen unter anderem auch auf iOS- und Android-basierten mobilen Geräten.

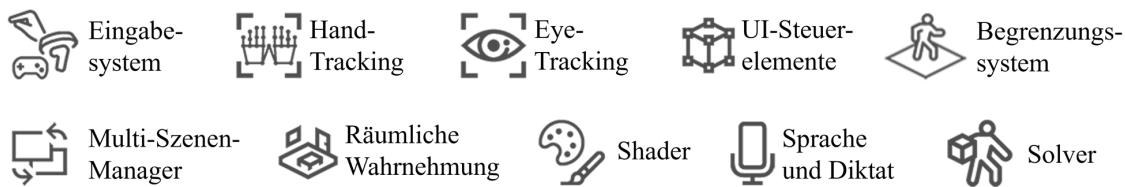


Abbildung 4-23: Auszug aus den vom MRTK adressierten Aspekten nach [SCP+21-ol]

Abgesehen vom *MRTK* können die SDKs der Geräte-Hersteller meist nur auf den Hersteller-eigenen Plattformen eingesetzt werden. Mit dem ***AR Foundation-Framework*** bietet *Unity* eine eigene kostenlose Lösung, die dieser Situation entgegenwirkt [Uni22n-ol]. Dabei unterstützt *AR Foundation* iOS- und Android-basierte Geräte sowie Magic Leap- und HoloLens-Brillen. Für die einzelnen Plattformen sind normalerweise die einzelnen SDKs *ARKit*, *ARCore*, *Lumin SDK* und *Win10 SDK* notwendig. Für HoloLens-Geräte steht wie zuvor beschrieben auch das *MRTK* zur Verfügung. *MRTK* basiert auf

⁸ Weitere Informationen unter <https://developer.magicleap.com/en-us/home>

dem *Win10 SDK*. Wie in Abbildung 4-24 dargestellt, vereint *AR Foundation* die Funktionalität der gerätespezifischen SDKs und stellt eine einheitliche Programmierschnittstelle (API) bereit. *AR Foundation* kann bei der Entwicklung einer AR-Anwendung zur Programmierung gewisser AR-Funktionalitäten verwendet werden. Die AR-Anwendung kann anschließend auf den vier unterstützten Plattformen installiert und verwendet werden. Dazu greift *AR Foundation* im Hintergrund auf die entsprechenden grundlegenden Funktionalitäten der gerätespezifischen SDKs zurück. *AR Foundation* unterstützt jedoch lediglich einen Teil der durch die SDKs zur Verfügung gestellten Funktionalitäten. Weitere Ausführungen dazu liefert Anhang A2.3.3.

Neben grundlegenden AR-Funktionalitäten werden in AR-Anwendungen **weitere Anwendungsfunktionalitäten** realisiert. Als Grundlage der Entwicklung dieser Anwendungsfunktionalitäten stehen verschiedene spezifische SDKs zur Verfügung. In Ergänzung zu den eigentlichen AR-Endgeräten können unter anderem externe Interaktionslösungen eingesetzt werden. Zur softwaretechnischen Integration der Lösungen stehen entsprechende SDKs bereit. Ein Beispiel ist das *Ultraleap Plugin* für *Unity*, mit dem Gesten-Interaktion über den *Ultraleap-Sensor* implementiert werden kann [Ult22-ol]. Für die Einbindung des *SenseGlove* als haptisches Interaktionsgerät steht beispielsweise das *SenseGlove SDK* bereit [Sen22-ol]. Zur Realisierung eines kompletten kollaborativen Spielemanagements bietet das *PlayFab SDK* von Microsoft umfangreiche Möglichkeiten [Mic22b-ol]. Datenbanken können mit *MongoDB* aufwandsarm in ein *Unity*-Projekt integriert werden [Mon22-ol]. Google ermöglicht mit dem *Maps SDK* die Einbindung von standortbezogenen Kartendaten [Goo22-ol]. Ein sehr bekanntes SDK ist zudem *OpenCV*. Über den Asset Store kann *OpenCV* auch in *Unity*-Anwendungen für komplexe Bildverarbeitungsfunktionalitäten eingesetzt werden [Ope22-ol]. Zudem bietet *Unity* selbst unter anderem *Machine Learning Agents* zur Realisierung von Funktionalitäten der Künstlichen Intelligenz an [Uni22o-ol].

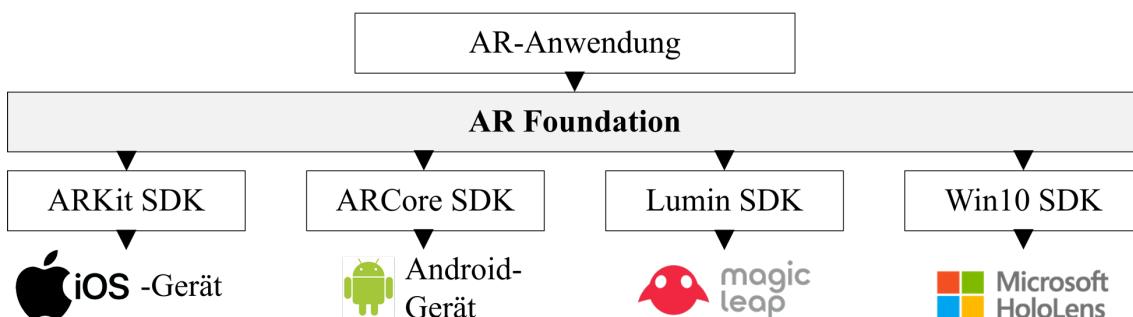


Abbildung 4-24: *AR Foundation-Framework*

Zur Entwicklung von AR-Lösungen als Werkzeuge zur kollaborativen Validierung sind **Netzwerk- und Kollaborationsfunktionalitäten** von besonderer Bedeutung. Dabei geht es unter anderem um Kommunikation über Audio und Video sowie um die Synchronisation von Inhalten. Beispielsweise kann es bei kollaborativen virtuellen Anwendungen wichtig sein, die aktuelle Position der anderen Nutzer in Echtzeit zu visualisieren. Zudem

sind Interaktionen von Nutzern, beispielsweise das Bewegen eines Objekts in der gemeinsamen virtuellen Welt, zu den anderen Nutzern zu synchronisieren. Ein gängiges Multiplayer- und Netzwerk-SDK ist beispielsweise *Photon PUN* (Photon Unity Networking). Eine detaillierte Analyse von Netzwerk-SDKs erfolgt in Anhang A2.3.3.

Bewertung: AR-SDKs stellen grundlegende Funktionalitäten von AR bereit. Ein AR-SDK ist daher notwendig zur Realisierung einer AR-Anwendung. Für die vorliegende Arbeit ist es sinnvoll, geräteunabhängige SDKs einzusetzen, da anderenfalls eine Nutzung auf verschiedenen Endgeräten bzw. Plattformen erschwert wird. Mit dem *MRTK* existiert ein umfangreiches und gleichzeitig kostenloses SDK, das sich auf mehreren Plattformen nutzen lässt. Ein besonderer Fokus liegt auf der Gestaltung von dreidimensionalen Benutzerschnittstellen. Diese verstärken die Immersion und bieten innovative Möglichkeiten im Kontext der Validierung. Andere grundlegende SDKs wie das *ARKit* und *ARCore* bieten diese Möglichkeiten nicht. Die Interaktionselemente des *MRTK* können jedoch auch auf Android- und iOS-basierten Geräten genutzt werden. In diesem Fall erfolgt die Interaktion nicht über Gesten, sondern über Touch. Das *MRTK* stellt insgesamt einen vielversprechenden Ansatz zur Unterstützung der Entwicklung interaktiver AR-Anwendungen als Validierungswerkzeug dar. Relevant sind zudem die Netzwerk- und Kollaborations-SDKs. Ein entsprechendes SDK ist definitiv für die Entwicklung von kolaborativen AR-basierten Validierungsanwendungen notwendig. Ein vielversprechendes und ebenfalls kostenloses SDK existiert mit *Photon PUN*.

4.3.2 Ansätze zur aufwandsarmen AR-Entwicklung

Eine große Herausforderung bei der Entwicklung von AR-Anwendungen ist wie bereits in Abschnitt 3.4.3 beschrieben die große Vielfalt an AR-Endgeräten. Geräte haben unterschiedliche funktionale und technische Eigenschaften und laufen mit teilweise individuellen Betriebssystemen. Zur Anwendungsentwicklung werden daher oft vom Hersteller bereitgestellte SDKs zur Verfügung gestellt (s. Abschnitt 4.3.1.2). Wird eine Anwendung für eine spezielle AR-Plattform entwickelt, lässt sie sich daher nicht auf andere Plattformen übertragen. Entsprechend ist erneuter und teilweise doppelter Programmieraufwand notwendig. Um dieses Problem zu umgehen und eine aufwandsarme AR-Entwicklung zu ermöglichen, werden verschiedene Ansätze verfolgt. Mit *OpenXR* wird eine standardisierte Programmierschnittstelle (Application Programming Interface, API) für AR und VR und verschiedene Plattformen entwickelt (Abschnitt 4.3.2.1). Im Forschungsprojekt *AcRoSS* wurde versucht, die Entwicklung von AR-Anwendungen für Produkt-Service-Systeme (PSS) durch wiederverwendbare Bausteine zu ermöglichen (Abschnitt 4.3.2.2). Zudem existieren verschiedene Software-Werkzeuge, die die Erstellung von AR-Anwendungen ohne Programmierung (No-Code) ermöglichen (Abschnitt 4.3.2.3).

4.3.2.1 OpenXR

OpenXR ist ein offener lizenzzfreier **Standard für XR-Anwendungen**, zu denen unter anderem AR und VR zählen. Der Standard wird seit 2017 von einer Arbeitsgruppe unter der Leitung der Khronos Group erarbeitet [Khr22-ol]. Motivation des *OpenXR*-Standards ist die zuvor beschriebene Heterogenität der Endgeräte. Für verschiedene Geräte müssen verschiedene SDKs bzw. APIs verwendet werden. Dies wird auch als API-Fragmentierung bezeichnet. Das Ziel von *OpenXR* ist es, der API-Fragmentierung durch eine standardisierte API entgegenzuwirken. *OpenXR* dient entsprechend als Schnittstelle zwischen XR-Anwendungen und verschiedenen XR-Plattformen (Abbildung 4-25). Dabei unterstützt *OpenXR* alle gängigen Plattformen, darunter Windows Mixed Reality, Vive, Oculus, SteamVR und Magic Leap. Durch den *OpenXR*-Standard soll die Kompatibilität einer XR-Anwendung zu den verschiedenen Plattformen gewährleistet werden. So wird die Übertragung einer Anwendung auf andere Geräte vereinfacht. Die Entwicklung des Standards wird durch eine Reihe großer Unternehmen unterstützt, unter anderem Google, Nvidia, Microsoft, Samsung und auch Unity.

OpenXR stellt kein SDK und keine Entwicklungsumgebung dar. *OpenXR* ermöglicht lediglich eine **einheitliche Programmierung von Funktionalitäten**, sodass eine Anwendung auf verschiedenen Plattformen lauffähig ist. Zu den von *OpenXR* abgedeckten Funktionalitäten gehören unter anderem die Beschreibung der Pose eines Endgeräts, die räumliche Interaktion inklusive Hand- und Blickverfolgung sowie räumliche Anker zur Lokalisierung. Der Funktionsumfang wird im Rahmen der fortlaufenden Entwicklung kontinuierlich erweitert und aktualisiert. Anbieter von SDKs können auf dem Standard aufbauen und so eine bessere Plattform-Unterstützung ihrer Lösungen erzielen. Unter anderem integriert das *MRTK* seit Juni 2021 bereits den *OpenXR*-Standard [Sem22-ol], [Mic23-ol].

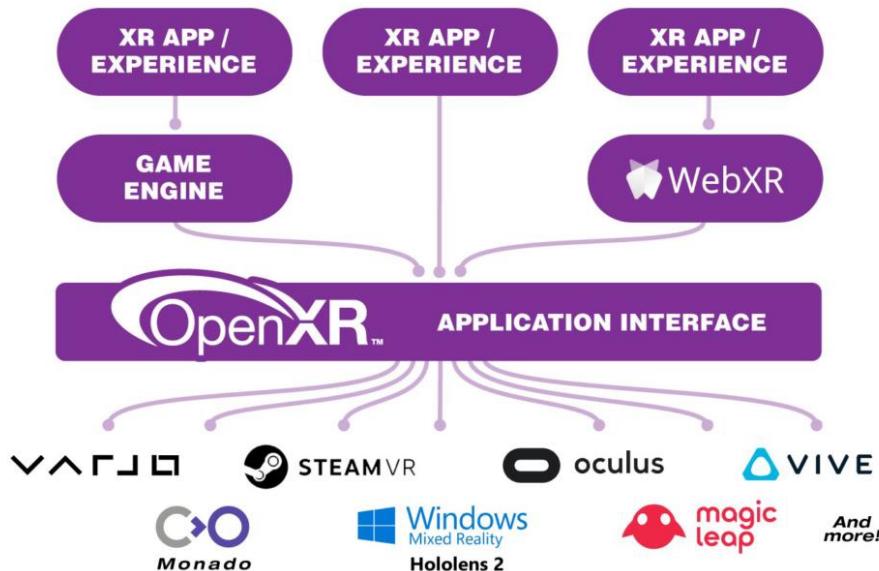


Abbildung 4-25: Übersicht über die Funktion des *OpenXR*-Standards als Schnittstelle zwischen XR-Anwendungen und verschiedenen Plattformen [Khr22-ol]

Bewertung: Der *OpenXR*-Standard stellt einen sinnvollen und hilfreichen Ansatz zur Entwicklung plattformunabhängiger XR-Anwendungen dar. Aufgrund der einfacheren Übertragbarkeit einer Anwendung auf verschiedene Geräte kann *OpenXR* als Ansatz zur aufwandsarmen Entwicklung betrachtet werden. Aufgrund seiner Funktion als Schnittstelle zwischen der Anwendungsentwicklung und den Plattformen ist *OpenXR* jedoch nur bedingt für die vorliegende Arbeit relevant. Es sollte darauf geachtet werden, dass verwendete SDKs auf dem *OpenXR*-Standard aufbauen, um eine hohe Wiederverwendbarkeit und Übertragbarkeit der Lösungen zu ermöglichen.

4.3.2.2 Forschungsprojekt AcRoSS

Ziel des Forschungsprojekts *AcRoSS*⁹ war es, Unternehmen bei der Konzipierung und Implementierung von **AR-basierten Produkt-Service-Systemen** zu unterstützen. Unter anderem sollte dazu eine aufwandsarme Entwicklung von AR-Anwendungen ermöglicht werden. Im Kern des Konzepts steht eine AR-Service-Plattform, auf der modulare Funktionsbausteine zur Verfügung gestellt werden [KMF+19]. Diese Bausteine adressieren beispielsweise die Erkennung von QR-Codes, Sprach- und Gestensteuerung, 3D-Instruktionsvisualisierung, Tracking-Ansätze sowie die Konnektivität. Die Bausteine können über die Plattform zu individuellen AR-Anwendungen kombiniert und konfiguriert und anschließend auf unterschiedlichen Plattformen installiert und verwendet werden. Nach diesem Konzept kann eine Entwicklung von AR-Anwendungen ohne Programmieraufwände erfolgen.

Um dies zu ermöglichen werden die einzelnen Bausteine semantisch beschrieben. Beispielsweise werden die Ein- und Ausgabewerte sowie die genaue Funktionalität definiert. In einem visuellen Tool auf der Plattform können die Bausteine basierend auf einer speziellen Logik zur Überprüfung der Semantik passend miteinander kombiniert werden. Entsprechend enthält die **AR-Service-Plattform** die verschiedenen Service-Bausteine, eine Web-Applikation als Benutzerschnittstelle sowie ein Knowledge-Management im Hintergrund (Abbildung 4-26). Die Kommunikation erfolgt über verschiedene Protokolle, unter anderem MQTT, REST und HTTP. Ausgeführt werden die Services in einer Laufzeitumgebung auf der Plattform. Auf dem eigentlichen AR-Endgerät läuft lediglich die **AcRoSS-Starter-App**, die mit den zentral ausgeführten Services kommuniziert und die jeweilige Funktionalität dem Anwender über eine AR-basierte Benutzerschnittstelle zur Verfügung stellt und visualisiert. Über die Starter-App können zudem weitere lokale oder gerätespezifische Services realisiert werden. In Situationen, in denen keine schnelle Internetverbindung vorhanden ist, kann das sogenannte **Companion Device** eingesetzt werden. Das Companion Device entspricht einer mobilen Recheneinheit, die eine lokale

⁹ Das BMWi-geförderte Verbundprojekt lief über 3 Jahre von 2016 bis 2019. Weitere Informationen sind verfügbar unter: https://www.digitale-technologien.de/DT/Redaktion/DE/Downloads/Publikation/SSW_Factsheet_AcRoSS.html

Ausführung der Services und Kommunikation mit dem AR-Endgerät ermöglicht [KMF+19].

Im Rahmen der Pilotprojekte von AcRoSS wurden zwei **prototypische AR-Anwendungen auf zwei unterschiedlichen AR-Endgeräten** entwickelt. Eine Anwendung dient der Reparatur-Unterstützung auf einer binokularen Microsoft HoloLens (Abbildung 4-27, links). Die komplette Benutzerinteraktion wurde dreidimensional und über Gestensteuerung umgesetzt. Die zweite Anwendung visualisiert Montage-Arbeitsschritte in einer zweidimensionalen Form auf einer monokularen RealWear HMT-1 (Abbildung 4-27, rechts). Hier findet die Interaktion per Sprache und Touch statt. Die Anwendungen nutzen entsprechend des Projektkonzepts auf der Plattform laufende Services, beispielsweise zur Fehlerbaumanalyse und zur Identifikation von Fehlern. Bei der Entwicklung der beiden Anwendungen wurde festgestellt, dass die generische Programmierung und Verwendung von AR-Bausteinen für verschiedene Endgeräte nicht realisierbar sind. Aufgrund der unterschiedlichen Charakteristiken verschiedener AR-Geräte und der individuellen SDKs wurden die AR-spezifischen Funktionalitäten lokal in der jeweiligen Starter-App realisiert [KMF+19]. Dies betrifft beispielsweise das Tracking, die unterschiedlichen Visualisierungstechniken sowie die Interaktion. Das Konzept der generischen AR-Bausteine konnte in *AcRoSS* entsprechend nur bedingt realisiert werden.

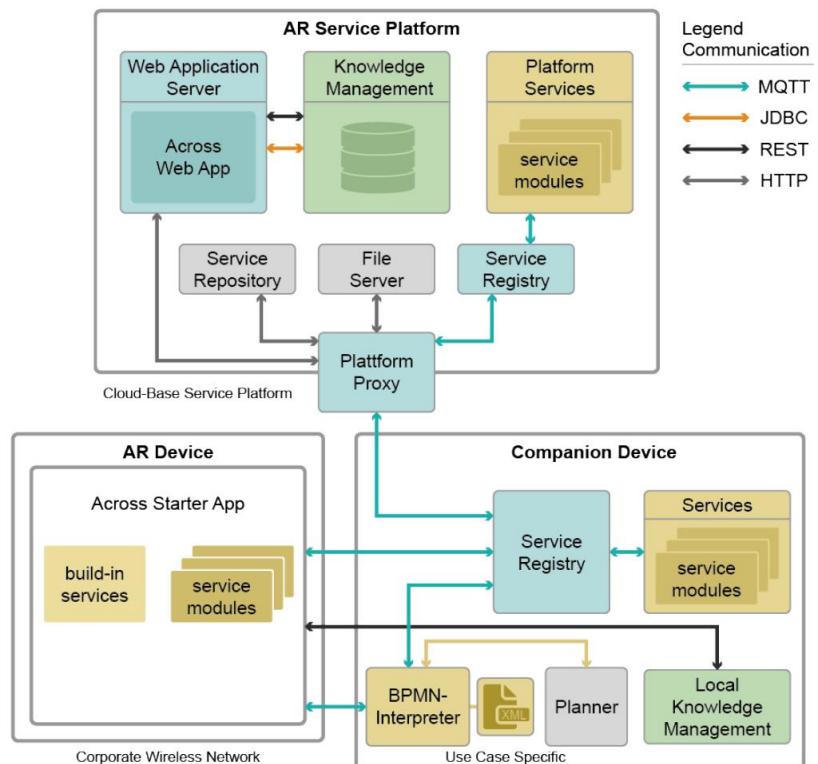


Abbildung 4-26: Die Struktur des AcRoSS-Ökosystems zur Laufzeit [KMF+19, S. 664]

Zu ähnlichen Erkenntnissen sind SPEICHER ET AL. im Kontext ihres **XD-AR-Frameworks** gekommen [SHY+18]. Sie stellen Funktionalitäten zentral auf einem Server be-

reit. Unter anderem wird über den Server die Kommunikation zwischen mehreren Geräten und das Management der Anwendungsfunktionalität realisiert. Die eigentliche AR-Funktionalität wird jedoch lokal basierend auf gerätespezifischen SDKs umgesetzt.



Abbildung 4-27: Pilotanwendungen im Projekt AcRoSS: Reparatur-Unterstützung auf der Microsoft HoloLens (links), Montage-Anleitung auf der RealWear HMT-1 (rechts) [KMF+19]

Bewertung: Im Rahmen des *AcRoSS*-Projekts wurde ein sehr vielversprechender Ansatz zur aufwandsarmen Entwicklung von AR-Anwendungen verfolgt. Das Konzept sieht eine Wiederverwendung modularer Funktionsbausteine ohne die Notwendigkeit von Programmierung vor. Es hat sich jedoch herausgestellt, dass die zentrale Bereitstellung und Ausführungen nicht praktikabel ist für AR-spezifische Bausteine. Aufgrund der Heterogenität der Endgeräte und der SDKs ist weiterhin eine individuelle Programmierung und lokale Ausführung auf den Geräten notwendig. Das Projekt wurde bereits 2019 abgeschlossen. Seitdem sind neue Ansätze zur geräteübergreifenden Entwicklung von AR-Anwendungen erarbeitet worden, bspw. das *MRTK* oder *AR Foundation*. Das grundsätzliche Konzept der modularen Bausteine ist vielversprechend für die Arbeit. Für die Validierung relevante Funktionalitäten könnten identifiziert und modular bereitgestellt werden. Der bisher untersuchte Stand der Technik lässt jedoch vermuten, dass eine generische Lösung zur geräteübergreifenden AR-Entwicklung ohne Einsatz einer Entwicklungsumgebung wie bspw. *Unity* weiterhin nicht möglich ist. Es muss entsprechend ein neuer Ansatz erarbeitet werden, der die Programmieraufwände innerhalb der Entwicklungsumgebung reduziert und vereinfacht.

4.3.2.3 No-Code AR-Editoren

Die Problemanalyse hat gezeigt, dass Unternehmen meist die Expertise und Erfahrung mit der Technologie AR fehlt. Im Kontext der Entwicklung von AR-Anwendungen betrifft dies vor allem auch einen Mangel an Programmierkenntnissen. Der *AcRoSS*-Ansatz (s. vorheriger Abschnitt 4.3.2.2) fokussierte eine Internetplattform-basierte AR-Entwicklung ohne Programmierung bei gleichzeitiger Unterstützung verschiedener AR-Plattformen. Es hat sich gezeigt, dass dieser Ansatz bisher nicht praktikabel ist. Eine Untersu-

chung von PRICE AND BARNES zeigt jedoch, dass eine Programmierung basierend auf vordefinierten Blöcken vor allem Anfängern deutlich einfacher fällt [PB15]. Das Konzept der vordefinierten Blöcke greifen verschiedene No-Code AR-Editoren auf. Sie ermöglichen die **Erstellung von AR-Anwendungen ohne Programmierung**. Im Folgenden wird eine exemplarische Auswahl dieser Editoren analysiert.

Einer der ersten No-Code AR-Editoren wurde von MOTA ET AL. im Bildungsbereich entwickelt [MRD+16]. **VEDILS** bietet eine grafische Benutzeroberfläche zur Erstellung von AR-basierte Lernszenarien nach dem Baukastenprinzip. Verschiedene Funktionsblöcke können zur Modellierung komplexer Abläufe miteinander kombiniert und verschachtelt werden. Ein ähnlicher Ansatz wird mit *Pacelab WEAVR* bzw. dem **WEAVR Creator** verfolgt [PAC22-ol]. Dabei handelt es sich um ein *Unity*-Plugin, das die Erstellung von virtuellen Trainingsanwendungen ohne Programmierung ermöglicht. Die Erstellung der Anwendung und die Konfiguration der Anwendungslogik sowie des Informationsflusses erfolgen Graph-basiert. Zielgruppe des *WEAVR Creator* sind vor allem Großkunden, die virtuelle Trainings realisieren wollen. Entsprechend ist *WEAVR Creator* mit Lizenzgebühren verbunden. Mit **BUNDLAR**¹⁰ existiert zudem eine Web-Plattform, über die einfache AR-Anwendungen für iOS- und Android-basierte mobile Geräte erstellt werden können. *BUNDLAR* ermöglicht Marker-Tracking und darauf basierend die Darstellung einfacher Inhalte. Ebenso wie *WEAVR Creator* ist auch *BUNDLAR* kostenpflichtig zu lizenzieren. Einen sehr bekannten und zudem kostenlosen No-Code AR-Editor stellt Apple mit dem **Reality Composer** bereit [App22a-ol]. Der *Reality Composer* kann als eigenständige Anwendung ausgeführt werden. Er bietet eine übersichtliche grafische Benutzeroberfläche zur Konfiguration der AR-Anwendungen. Vordefinierte oder eigene 3D-Modelle können in eine Anwendung importiert werden. Über die Benutzeroberfläche lässt sich anschließend unter anderem die Art der Positionierung der Inhalte in der realen Umgebung sowie die zu verwendenden Materialien konfigurieren (Abbildung 4-28, links). Zudem stellt der *Reality Composer* vorbereitete Funktionalitäten und Verhaltensweisen als Bausteine bereit (Abbildung 4-28, rechts). Die AR-Anwendungen können anschließend in Apple-spezifischen Formaten exportiert und über spezielle Apps auf iOS-basierten mobilen Geräten zur Visualisierung mittels AR verwendet werden.

Bewertung: No-Code AR-Editoren ermöglichen die Entwicklung von AR-Anwendungen ohne Programmierung. Es ist eine Reihe dieser Editoren verfügbar, die größtenteils nur spezifische Plattformen unterstützen. Zudem werden meist lediglich sehr einfache Funktionalitäten wie das Darstellen von 3D-Modellen auf einem Marker ermöglicht. Komplexere Funktionalitäten sind dagegen auf spezifische Anwendungsfälle bezogen. Beispielsweise bietet *WEAVR Creator* Funktionsbausteine für Trainings-Szenarios. Der *Reality Composer* von Apple bietet grundsätzlich gute Möglichkeiten zur Erstellung an-

¹⁰ Weitere Informationen unter <https://bundler.com/>

sprechender AR-Erlebnisse. So können aufwandsarm Produktpräsentations- oder Marketing-Apps entwickelt werden. Allerdings ist der Reality Composer auf iOS-Geräte beschränkt. Keiner der Ansätze ermöglicht die aufwandsarme Entwicklung einer AR-Anwendung mit den für die Validierung notwendigen Funktionalitäten. Entsprechend stellen bestehende No-Code AR-Editoren keine praktikable Lösung zur Unterstützung der geräteübergreifenden AR-Anwendungsentwicklung im Kontext der Validierung dar.

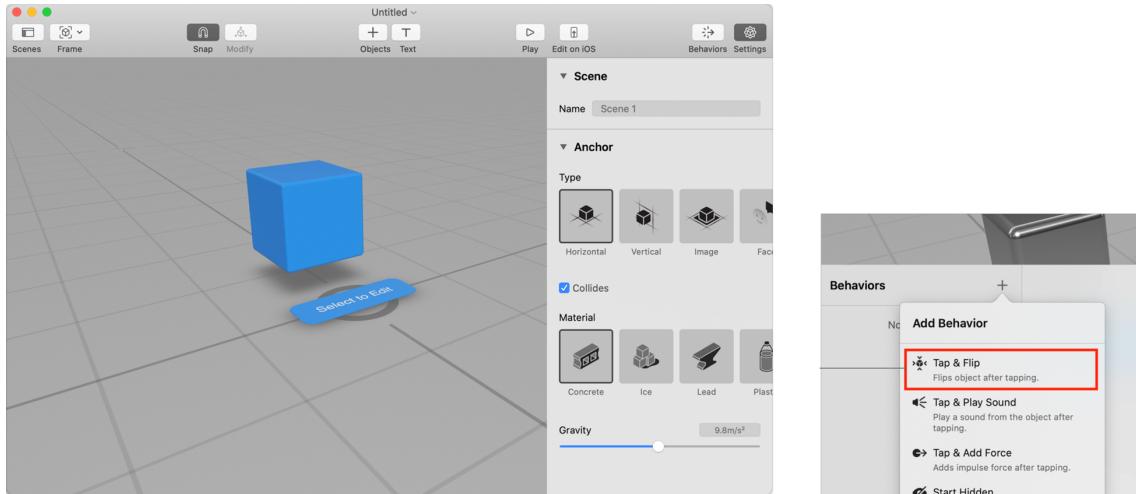


Abbildung 4-28: Reality Composer von Apple: Konfiguration einer Szene (links) und Auswahl vordefinierter Funktionalitäten und Verhaltensweisen (rechts) [App22a-ol]

4.3.3 AR-Anwendungssoftware im Kontext Validierung

Der Einsatz von AR als Validierungswerkzeug ist zum einen nicht in der Industrie etabliert und zum anderen kein präzenter Gegenstand der Forschung. Entsprechend sind nur wenige Berichte und wissenschaftliche Veröffentlichungen über AR-Anwendungssoftware im Kontext der Validierung bekannt. Viele Berichte befassen sich mit dem Einsatz der verwandten Technologie VR im Bereich Building Information Modeling (BIM) [ZC18], [BWY+10]. Unter anderem beschreiben PAES ET AL. konkrete VR-Anwendungen für Design Reviews von BIM-Daten [PIP21]. Der Einsatz von AR zur Unterstützung der Validierung wird ebenfalls meist anhand konkreter Anwendungsszenarien beschrieben. Es handelt sich entsprechend um sehr spezifische Anwendungssoftware, die konkrete Anforderungen erfüllt und nicht auf andere Validierungssituationen übertragbar ist. Zudem basieren diese Ansätze oft auf mittlerweile veralteter Technik. VERLINDEN ET AL. beschreiben beispielsweise ein Projektions-AR-System für Marker-basierte Design Reviews [VHN09]. BORDEGONI ET AL. nutzen bei ihrem Mixed-Prototyping-Ansatz zur Produktbewertung prototypische Brillen-Systeme und ein Spiegel-basiertes 3D-Visualisierungssystem [BCC+09]. Einen einfachen Smartphone- und Marker-basierten Ansatz beschreiben MAIR ET AL. [MRS14]. All diese AR-Anwendungen stellen keine generischen Lösungen zur Unterstützung der Produktvalidierung dar. Im Folgenden werden verschiedene Ansätze zu AR-Anwendungen im Kontext der Produktvalidierung beschrieben, die

im Rahmen dieser Arbeit für die aufwandsarme Entwicklung individueller Anwendungssoftware relevant sein können. Zunächst wird die Vorgehensweise zum Aufbau einer AR-basierten Validierungsumgebung nach REINEMANN untersucht (4.3.3.1). In Abschnitt 4.3.3.2 wird anschließend eine Auswahl wissenschaftlicher und kommerzieller Lösungen im Kontext AR-basierter Validierung analysiert. Für die Validierung ist zudem wichtig, dass Feedback zu den durch die Prototypen repräsentierten Produktmerkmalen gegeben werden kann. In Abschnitt 4.3.3.3 werden daher abschließend Ansätze zur Annotation in AR-Anwendungen behandelt.

4.3.3.1 Aufbau AR-basierter Validierungsumgebungen nach REINEMANN

REINEMANN beschreibt im Rahmen seiner Dissertation neben der Konfiguration auch den Aufbau einer AR-basierten Validierungsumgebung [Rei21]. Dabei wird eine AR-Szene zur Darstellung auf einem ausgewählten AR-Endgerät entwickelt. Der beschriebene Aufbau der Validierungsumgebung basiert auf einer unveröffentlichten Bachelorarbeit. Im Rahmen dieser Bachelorarbeit wurden zwei exemplarische AR-Anwendungen entwickelt, eine in der Entwicklungsumgebung *Unity* und eine in der Software Vuforia™ Studio (s. Abschnitt 4.3.3.2). Die aus der Bachelorarbeit abgeleitete **Vorgehensweise zum Aufbau der Validierungsumgebung** umfasst vier Schritte, die in Textform von REINEMANN angeleitet werden:

- 1) **Definition der Geometrie:** Im ersten Schritt erfolgt die Erstellung der notwendigen Geometrien in Form von CAD-Modellen. REINEMANN beschreibt das typische Vorgehen der Geometrie-Erstellung mittels CAD-Werkzeugen. Um eine flüssige Visualisierung auf mobilen AR-Geräten zu ermöglichen, wird eine Bereinigung und Reduzierung bestehender komplexer CAD-Modelle vorgeschlagen.
- 2) **Erstellen eines Visualisierungsmodells:** Ausgehend von den CAD-Modellen werden Polygon-Modelle für die Visualisierung erarbeitet. Dazu ist wie in Anhang A2.3.2 beschrieben eine Konvertierung bzw. Tesselierung notwendig. REINEMANN gibt allgemeine Hinweise zum Setzen von Material- und Oberflächeneigenschaften der Modelle über Shader und weist darauf hin, dass spezielle Shader beispielsweise für die Berechnungen von Verdeckungen von Objekten benötigt werden. Konkrete Details zu diesen Aspekten fehlen jedoch.
- 3) **Verknüpfung von virtuellen und physischen Modellen:** Im dritten Schritt erfolgt die Konfiguration von Tracking-Lösungen zur passenden Positionierung der Modelle in der realen Umgebung. Es wird beschrieben, dass sowohl 2D-Marker-Tracking als auch 3D-Objekt-Tracking sowie eine freie Positionierung verwendet werden können. Als konkretes SDK wird *Vuforia* genannt. Zudem werden verschiedene Hinweise gegeben, die es bei der Nutzung der verschiedenen Tracking-Ansätze zu berücksichtigen gilt.

4) Implementierung von Funktionalität, Interaktivität und Verhalten: Im letzten Schritt werden die Visualisierungsmodelle in der AR-Szene mit Funktionalität, Interaktivität und Verhalten versehen. Funktionalitäten dienen der Repräsentation funktionaler Produktmerkmale. Funktionalitäten können laut REINEMANN über Animationen realisiert werden. Animationen beschreiben eine zeitliche Reihenfolge an Zuständen des Modells, meist bzgl. der Transformation einzelner Komponenten. Über Interaktion kann der Benutzer beispielsweise zwischen verschiedenen Varianten eines Produkts oder alternativen Merkmalsausprägungen wechseln. Als Möglichkeiten der Interaktion werden Sprache, Gesten oder Touch genannt.

Nach Abschluss der vier Schritte kann die AR-Szene auf das entsprechende AR-Endgerät übertragen werden. An dieser Stelle wird lediglich darauf verwiesen, dass dies zum einen mit *Unity* auf verschiedenen Plattformen möglich ist. Alternativ kann eine mit Vuforia™ Studio erstellte AR-Szene über die zusätzliche Software Vuforia™ View ebenfalls auf verschiedenen Plattformen genutzt werden.

Bewertung: REINEMANN beschreibt eine systematische Vorgehensweise zum Aufbau von AR-basierten Validierungsumgebungen. Der Ablauf entspricht grundsätzlich dem in Abschnitt 4.3.1.1 beschriebenen typischen Entwicklungsprozess von AR-Anwendungen. REINEMANN beschreibt die einzelnen Schritte jedoch nur sehr allgemein und nicht im Detail. Zudem wird lediglich auf bestehende Software-Werkzeuge zur Nutzung verwiesen. Die durch die Vorgehensweise erarbeiteten AR-Szenen ermöglichen eine interaktive Visualisierung von 3D-Modellen. Möglichkeiten für Feedback werden nicht gegeben. Zudem findet keine Kollaboration statt, AR wird lediglich lokal durch einen Benutzer verwendet. Entsprechend ist der Aufbau einer AR-basierten Validierungsumgebung nach REINEMANN nicht hilfreich für die vorliegende Arbeit.

4.3.3.2 Bestehende Lösungen im Kontext AR-basierter Produktvalidierung

AR-basierte Validierung ist weder in der Industrie etabliert noch in der Forschung präsent. Viele wissenschaftliche Aktivitäten befassen sich mit der verwandten Technologie VR in der Produktentwicklung, da VR einen höheren Reifegrad aufweist. Unter anderem untersucht WOLFARTSBERGER die Potentiale von VR für Design Reviews im Engineering [Wol19]. Im Folgenden werden neben AR- auch VR-Lösungen analysiert, da es sich bei beiden Technologien um immersive Visualisierungsformen handelt und VR-Potentiale bzw. -Funktionalitäten teilweise auf AR übertragen werden können.

JASPER hat im Rahmen seiner Dissertation untersucht, inwiefern interaktive Visualisierungen die Merkmalspräsentation im Rahmen einer **Conjoint-Analyse** (vgl. Abschnitt 3.3.3) unterstützen können [Jas15]. Mittels VR konnte flexibel zwischen verschiedenen Merkmalsausprägungen und Funktionalitäten gewechselt und diese interaktiv und anschaulich präsentiert werden. Durch die immersive Visualisierung konnten die Probanden

auch komplexe Sachverhalte nachvollziehen. Die Analyse konnte effizienter und aussagekräftiger durchgeführt werden. Zudem kann die bei klassischen Ansätzen häufig auftretende Ermüdung durch wenig ansprechende Stimuli vermieden werden [GDE+18].

AR wurde beispielsweise zur Validierung verschiedener Fahrzeuginterieurs eingesetzt [DKM+19]. Abbildung 4-29 zeigt ein entsprechendes projektionsbasiertes AR-System. Durch die Möglichkeiten von AR lassen sich verschiedene Interieur-Varianten auf einem **physischen Prototyp** eines Armaturenbretts darstellen und somit haptische Erlebnisse schaffen. Dies ermöglicht sowohl die Bewertung verschiedener Designs als auch von Ergonomie-Aspekten wie Einsehbarkeiten und Erreichbarkeiten.

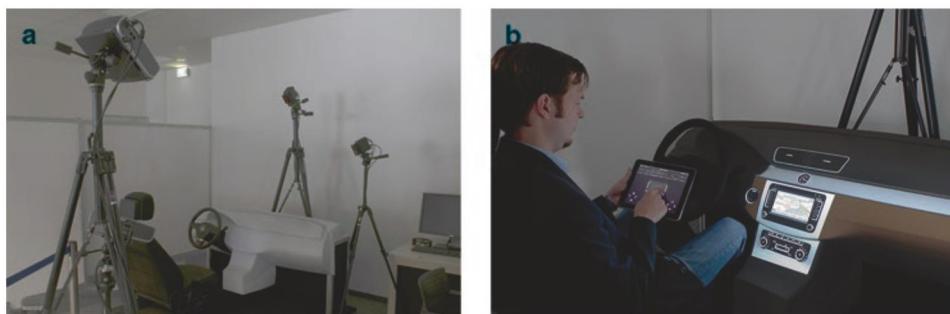


Abbildung 4-29: AR-basierte Validierung verschiedener Fahrzeug-Interieurs unter Einbindung physischer Elemente, Bildausschnitt aus [DKM+19, S. 375]

Eine Recherche zu kommerziellen Software-Lösungen zur Unterstützung der Validierung hat gezeigt, dass keine explizit für die kundenintegrierte Produktvalidierung vorgesehene Lösung existiert. Viele AR- und VR-Software-Lösungen adressieren die immersive Visualisierung von Produkten. Entsprechende Lösungen können zur Bewertung von Produktmerkmalen eingesetzt werden. Die Software wird jedoch meist für spezielle Themenfelder oder Anwendungsszenarien angeboten, beispielsweise zur Visualisierung von Architekturmodellen. Zudem bestehen einige VR-basierte Software-Lösungen zur virtuellen Layout-Planung. Mit dem **Augmented Reality Engineering Space (AR 3S)**¹¹ bietet das Münchener Unternehmen Holo-Light verschiedene AR-Funktionalitäten für den industriellen Einsatz. Mit *AR 3S* können qualitativ hochwertige 3D-Konstruktionsdaten interaktiv und kollaborativ mittels AR visualisiert werden (Abbildung 4-30, a). Die Software läuft auf mobilen Endgeräten sowie auf HoloLens-Geräten. Durch geeignete Interaktionskonzepte, unter anderem Gesten-basiert, bietet *AR 3S* dem Betrachter die Möglichkeit, die 3D-Modelle zu rotieren, zu skalieren und zu verschieben sowie spezielle Ansichten wie Schnitte zu erzeugen. So können Konstruktionsdaten immersiv während der Entwicklung überprüft werden. *AR 3S* wird unter anderem in der Automobilbranche eingesetzt [Hol20-ol]. PTC bietet mit dem bereits in Abschnitt 4.3.3.1 erwähnten **Vuforia™ Studio** ein Autorensystem zur Erstellung von AR-Visualisierungsinhalten [PTC22b-ol].

¹¹ Weitere Informationen unter <https://holo-light.com/products/ar3s/>

In *Vuforia™ Studio* erstellte Inhalte werden in eine lokale oder Cloud-basierte Datenbank geladen. Von dort können sie mit der *Vuforia™ View*-App auf zuvor konfigurierten AR-Markern, genannt *ThingMark*, dargestellt werden (Abbildung 4-30, b). Die Inhalte können mit Animationen und Zusatzinformationen versehen werden, die über Interaktions-elemente in der *View*-App entsprechend verwendet werden. Mit **Vuforia™ Chalk** bietet PTC zudem eine Smartphone-basierte AR-Lösung zur Unterstützung von Service-Mitarbeitern und Technikern [PTC22c-ol]. Ähnlich zu der von BRUNO ET AL. im folgenden Abschnitt 4.3.3.3 beschriebenen Lösung ermöglicht *Chalk* die Positionierung von virtuellen Annotationen in realen Umgebungen. Mit der *Chalk*-App können Freihandzeichnungen auf dem Bildschirm erstellt und in der realen Szene eingefügt werden (Abbildung 4-30, c). Von Kollegen erstellte Annotationen werden asynchron anderen Technikern an der entsprechenden Stelle angezeigt und können so für die Wissensvermittlung und Kommunikation verwendet werden. IKEA stellt potentiellen Kunden die Smartphone-App **Place** bereit [Ste22-ol]. Mit der App können Kunden Produkte aus dem IKEA-Portfolio als 3D-Modelle virtuell in ihren eigenen Räumlichkeiten visualisieren (Abbildung 4-30, d). Die *Place*-App stellt lediglich eine einfache, aber kundenfreundliche Möglichkeit der Validierung fertiger Produkte dar.

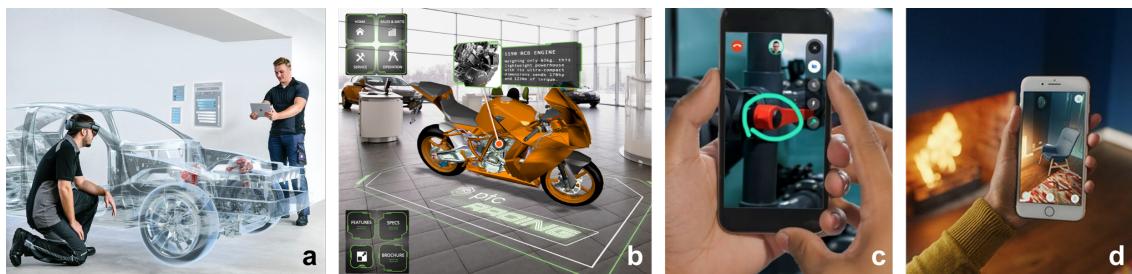


Abbildung 4-30: Drei exemplarische Software-Lösungen im Kontext Validierung: a) *Holo-Light AR 3S* [XRG21-ol], b) *Vuforia™ View*-App¹², c) *Vuforia™ Chalk* (Quelle: PTC.com), d) *IKEA Place*-App [Ste22-ol]

Bewertung: Die Recherche zu bestehenden AR-Software-Lösungen im Kontext der Produktvalidierung belegt, dass zum aktuellen Zeitpunkt keine fertige Software verfügbar ist, die alle Anforderungen erfüllt und einen generischen Einsatz für verschiedene Branchen und Produkttypen ermöglicht. Bestehende Software-Lösungen adressieren lediglich Teilespekte und sind für ausgewählte Anwendungsszenarien konzipiert. Es handelt sich zudem entweder um wissenschaftliche Prototypen oder mit Lizenzkosten verbundene kommerzielle Angebote. Es bestätigt sich der Bedarf an einer Lösung, die Unternehmen in die Lage versetzt, eigene AR-Validierungssoftware zugeschnitten auf die individuellen Bedarfe und Anforderungen zu entwickeln. Die beschriebenen sowie weitere Ansätze lie-

¹² Die Grafik wurde aus dem Google Play Store verwendet (<https://play.google.com/store/apps/details?id=com.ptc.vuforiaview>)

fern jedoch hilfreiche Informationen bzgl. für die Validierung relevanter Funktionalitäten. Zudem verdeutlichen die beschriebenen Ansätze die Notwendigkeit bedarfsgerechter Technik und Funktionalität. Für eine effiziente und effektive Validierung müssen sowohl geeignete Hardware als auch zielgerichtete Softwarefunktionalitäten verwendet werden. Somit liefern die Ansätze wichtige Informationen sowohl für die AR-System-Konfiguration in Handlungsfeld 2 als auch für die aufwandsarme Entwicklung der Software in Handlungsfeld 3.

4.3.3.3 Ansätze für Feedback in AR

AR bietet die Möglichkeit, **virtuelles Feedback** direkt am dreidimensionalen Modell zu annotieren. Das ermöglicht eine effizientere Abgabe des Feedbacks und verbessert den Informationsaustausch zwischen verschiedenen Akteuren [BBM+19]. Feedback lässt sich sowohl über Handheld- als auch Kopfgetragene AR-Geräte dreidimensional am 3D-Modell und somit direkt dort, wo es relevant ist, positionieren. Im Folgenden werden verschiedene Ansätze analysiert, die entsprechendes Feedback ermöglichen.

CHANG ET AL. haben eine Lösung für 3D-Annotationen auf der HoloLens entwickelt [CNL+17]. Die Lösung nutzt die Gesten-Erkennung der HoloLens und ermöglicht darauf basierend **Freihand-Annotationen** (Abbildung 4-31).



Abbildung 4-31: Gesten-basierte Freihand-Annotation auf einer HoloLens [CNL+17]

Der Benutzer kann durch eine Klick-Geste die Aufnahme einer Annotation starten. Die anschließende Bewegung der Hand im Raum wird verfolgt und eine entsprechende dreidimensionale Linie gezeichnet. Dabei gibt es zwei Möglichkeiten. Die Linie kann frei im Raum gezeichnet werden oder auf einer im Hintergrund liegenden Oberfläche. Im letzten Fall wird die in Blickrichtung hinter der Geste liegende Fläche detektiert und als Malfläche verwendet. Dies kann beispielsweise eine Wand oder ein Objekt im Raum sein. Bei beiden Annotations-Varianten wird über einen Kreis als 3D-Cursor jeweils die aktuelle Zeichenposition visualisiert. Dies ist entweder frei in der Luft vor dem Benutzer (Abbildung 4-31, Mitte) oder passend auf der Hintergrundfläche. Der Cursor unterstützt den Benutzer beim Zeichnen. Zur besseren Darstellung werden die Freihand-Annotationen automatisch überarbeitet und optimiert. Unter anderem findet eine zeitliche und geometrische Glättung statt. Ergebnis der Optimierung sind anschauliche 3D-Annotationen. Ein Beispiel ist rechts in Abbildung 4-31 dargestellt. Neben Freihand-Annotationen ermöglichen die Lösung von CHANG ET AL. zudem das Gesten-basierte Zeichnen von **3D-Pfeilen**.

Beim Klick wird der Endpunkt gesetzt und beim Loslassen der Geste die Pfeilspitze. Da die Annotations-Funktionalität vor allem als Grundlage für Nutzertests entwickelt wurde, wird bisher keine Dokumentation der 3D-Annotationen zur späteren Einsicht unterstützt.

Eine für die **Annotation von Design-Variationen** vorgesehene AR-Lösung wird von BRUNO ET AL. in [BBM+19] beschrieben. Die Smartphone-Anwendung nutzt sowohl Marker-basiertes Tracking als auch die Feature-basierte Analyse und Rekonstruktion der realen Umgebung, um Annotationen platzieren zu können. Dazu werden das *Vuforia SDK* und das mittlerweile nicht mehr verfügbare *Google Tango*¹³ verwendet. Über das Tracking können Konstruktionsdaten in Form von 3D-Modellen mit den entsprechenden realen Objekten überlagert werden (Abbildung 4-32). Anhand der AR-basierten Visualisierung können Abweichungen zwischen der geplanten und der tatsächlichen Konstruktion erkannt werden. Zur Dokumentation von erkannten Fehlern und Anmerkungen können über eine Benutzeroberfläche 3D-Annotationen mit Kommentaren hinzugefügt werden. Die 3D-Annotationen entsprechen einfachen 3D-Kugeln, die entsprechend verschiedenen Arten von Anmerkungen in unterschiedlichen Farben dargestellt werden (Abbildung 4-32, rechts). Die hinzugefügten Annotationen können auch auf anderen Geräten von anderen Personen an den entsprechenden Stellen am Produkt betrachtet werden. Ähnliche Annotations-Funktionalitäten werden auch in kommerziellen Software-Werkzeugen angeboten. Ein Beispiel ist *Vuforia™ Chalk* von PTC.

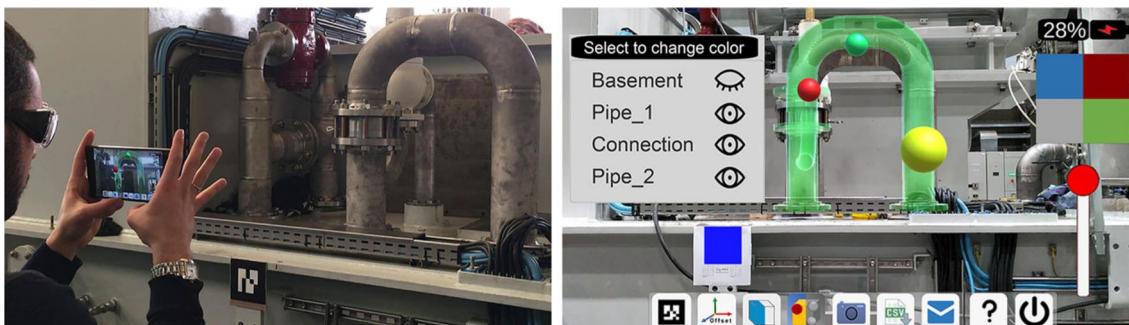


Abbildung 4-32: *Smartphone-basierte Annotation* [BBM+19, S. 882]

Einen anderen Feedback-Ansatz verfolgen ARBELÁEZ-ESTRADA UND OSORIO-GÓMEZ [AO13]. Sie nutzen eine Marker-basierte Smartphone-AR-Anwendung, um **Screenshots der Visualisierung eines Modells** zu erstellen. Die Screenshots können mit einem Wert von 1 bis 5 bewertet und anschließend gespeichert werden (Abbildung 4-33).

Bewertung: Die Abgabe von Feedback während der Validierung ist eine wichtige Funktionalität einer AR-Anwendung zur Unterstützung der Validierung. Die beschriebenen

¹³ Mit *Tango* hat Google zeitweise eine Hardware- und Software-Kombination für AR basierend auf 3D-Sensoren angeboten. Mittlerweile ist aus dem *Tango*-Projekt das *ARCore SDK* entstanden, das unabhängig von spezieller Sensorik und lediglich unter Einsatz der Standard-Kamera eingesetzt werden kann.

Ansätze stellen verschiedene Lösungen dar. Screenshot-basiertes Feedback nach ARBÉLÉZ-ESTRADA UND OSORIO-GÓMEZ nutzt die räumlichen und interaktiven Möglichkeiten von AR nicht aus. Ansätze für räumlich positionierte Annotationen ähnlich zu der Lösung von BRUNO ET AL. adressieren eher die asynchrone Zusammenarbeit von Konstrukteuren. Sie sind nicht für Feedback in Validierungsanwendungen konzipiert. Die Gesten-basierten Freihand-Annotationen nach CHANG ET AL. stellen jedoch einen viel-versprechenden Ansatz zur Abgabe von Feedback an 3D-Modellen dar. Entsprechende Annotationen nutzen die natürlichen Interaktionsmöglichkeiten von AR-Brillen aus und sind intuitiv zu benutzen. Zum einen müssten Freihand-Annotationen allerdings um weitere Feedback-Möglichkeiten ergänzt werden, beispielsweise durch Text-Annotationen. Zum anderen müsste ein geeigneter Ansatz zur Verwendung von Freihand-Annotationen sowohl auf Brillen als auch auf mobilen Geräten wie Smartphones und Tablets erarbeitet werden.

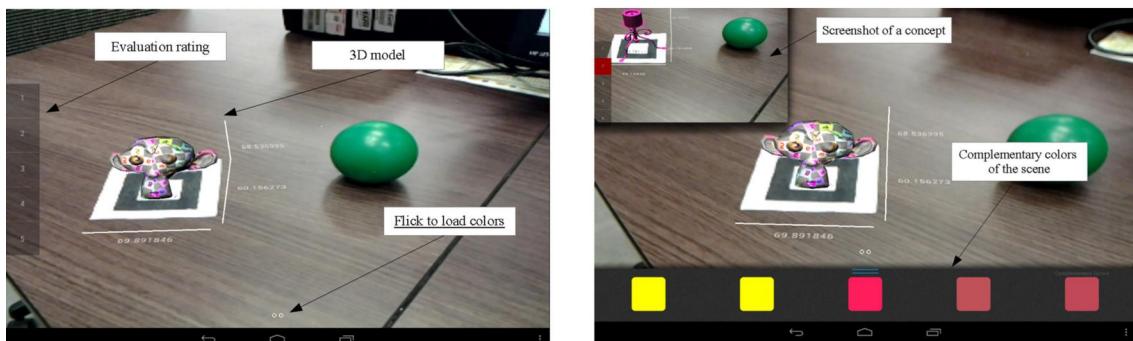


Abbildung 4-33: Einfaches Feedback durch Aufnahme und Bewertung von Screenshots in einer Smartphone-AR-Anwendung [AO13, S. 396]

4.4 Bewertung und Handlungsbedarf

Ein Vergleich des Stands der Technik mit den in Kapitel 3.6 abgeleiteten Anforderungen führt zu folgender Bewertung, die Abbildung 4-34 zusammenfasst:

A1) Ganzheitlichkeit: Es besteht bisher keine ganzheitliche Beschreibung der Potentiale von AR im Kontext Validierung. Es konnten jedoch verschiedene Ansätze und Quellen identifiziert werden, die die Definition von AR-Potentialen im Kontext der Validierung unterstützen können. Eine hilfreiche Grundlage zur systematischen Planung AR-basierter Validierung stellen unter anderem die Wiedergabedimensionen von Prototypen nach REINEMANN sowie verschiedene Arbeiten, die die grundsätzliche Funktionalität von AR beschreiben. Die hilfreichen Ansätze gilt es im Rahmen der vorliegenden Arbeit für die Erarbeitung eines Hilfsmittels zu nutzen und darauf aufzubauen.

A2) Generalisierung: Die identifizierten Ansätze zur Bewertung von Potentialen adressieren spezifische Anwendungsfälle oder decken lediglich Teilaspekte in generischer Form ab. Einen interessanten Ansatz zur Generalisierung individueller Anforderungen stellt das House of Quality nach KLEIN dar. Es besteht kein vollumfänglicher Ansatz, der die systematische Planung AR-basierter Validierung generisch unterstützt.

A3) Korrektheit: Die der Bewertung von AR-Potentialen zugrundeliegenden Informationen müssen fundiert und korrekt sein. Da keine Quelle die Potentiale von AR im Kontext Validierung explizit und ganzheitlich beschreibt, sind einzelne Informationen zwar grundsätzlich korrekt, sie müssen jedoch in den korrekten Zusammenhang und durch Expertenwissen ergänzt und angepasst werden.

Bewertungsskala:		Anforderungen	Planung			Konfiguration			SW	Vorgehen	
			Ganzheitlichkeit	Generalisierung	Korrektheit	Umfassende Dokumentation	Erweiterbarkeit	Zugänglichkeit			
			A1	A2	A3	A4	A5	A6	A7	A8	A9
Systematische Planung	House of Quality nach KLEIN		○	○	●	○	○	○	○	●	●
	Leitfragen-basierte Bewertung von AR-Potentialen nach PALMARINI ET AL.		●	○	●	○	○	○	○	●	●
	Nutzen-Aufwands-Bewertung von AR-Potentialen nach ROELTGEN		●	○	●	●	○	○	●	●	●
	Agile Practices Impact Model (APIM) nach DIEBOLD UND ZEHLER		●	○	●	○	○	●	○	●	●
	Konfiguration AR-basierte Validierungsumgebungen nach REINEMANN		○	○	●	○	○	●	○	●	●
Wissensbasierte AR-Systemkonfiguration	Modelle und Modellierungssprachen		○	●	○	○	●	○	○	○	○
	Semantische Technologien		○	●	○	○	●	●	○	○	○
	AR-System-Klassifikation nach FELLMANN ET AL.		○	●	○	○	●	○	○	○	●
	Bewertung von AR-Systemen nach ROELTGEN		○	○	○	○	○	●	○	●	●
	Konfiguration von Fahrsimulatoren nach HASSAN		○	○	○	○	●	○	○	●	●
	Checklisten-basierte Evaluierung von AR-Displays nach PAELKE ET AL.		○	○	○	○	●	○	○	●	●
	AR-System-Konfiguration nach SCHILLING		●	●	○	○	○	●	○	●	●
Aufwandsarme AR-Softwareentwicklung	Wissensbasierte Produktionssystemkonzipierung nach BAUER		○	○	○	○	●	○	○	●	●
	AR-Entwicklungsprozess		○	○	○	○	○	○	○	●	●
	Augmented Reality-Software Development Kits (SDK)		○	○	○	○	○	○	●	○	●
	OpenXR		○	○	○	○	○	○	●	○	○
	Forschungsprojekt AcRoSS		○	○	○	○	○	○	○	●	●
	No-Code AR-Editoren		○	○	○	○	○	○	●	○	●
	Aufbau einer AR-basierten Validierungsumgebung nach REINEMANN		○	○	○	○	○	○	○	●	○
	Anwendungen AR/VR-basierter Produktvalidierung		○	○	●	○	○	○	●	○	○
Ansätze für Feedback in AR			○	○	○	○	○	○	●	○	●

Abbildung 4-34: Bewertung des Stands der Technik anhand der Anforderungen

A4) Umfassende Dokumentation: Die AR-System-Konfiguration bedarf einer umfassenden Dokumentation des relevanten Wissens. Das relevante Wissen liegt jedoch in vielfältigen verteilten Quellen vor. Die analysierten Ansätze zur Konfiguration von AR-Systemen und Validierungsumgebungen sind entweder veraltet oder unterstützen lediglich auf einem hohen Niveau. Entsprechend ist das zugrundeliegende Wissen nicht ausreichend detailliert. Im Rahmen der vorliegenden Arbeit gilt es entsprechend, das notwendige Wissen aus der Vielzahl unterschiedlicher Quellen zu in einer Wissensbasis sammeln und zu formalisieren, um so eine umfassende Dokumentation zu erzielen. Als technische Grundlage der Repräsentation und Verarbeitung großer Wissensmengen wurde mit semantischen Technologien ein vielversprechender Ansatz identifiziert.

A5) Erweiterbarkeit: Die Erweiterbarkeit der Wissensbasis ist aufgrund der stetig fortschreitenden Entwicklung der Technologie AR wichtig für eine nachhaltige Lösung zur Unterstützung der AR-System-Konfiguration. Diese Anforderung wird unter anderem durch semantische Technologien und entsprechende Ansätze des Ontology Learnings adressiert. Einzelne Ansätze zur Konfiguration technischer Systeme sehen zudem eine Erweiterung der verwendeten Wissensbasis vor. Keiner dieser Ansätze adressiert die Konfiguration von AR-Systemen. Die Konzepte der identifizierten Ansätze lassen sich jedoch auf ein wissensbasiertes System zur AR-System-Konfiguration übertragen.

A6) Zugänglichkeit: Im Kontext der AR-System-Konfiguration ist der einfache Zugang für Unternehmen zum relevanten Wissen von großer Bedeutung. Das Wissen muss von den Unternehmen bedarfsgerecht und zielgerichtet verwendet werden können. Keiner der identifizierten Ansätze ermöglicht dies für den vorgesehenen Kontext. Verschiedene Ansätze machen Wissen für andere Themenfelder zugänglich, beispielsweise die Nutzen-Aufwands-Bewertung von AR-Potentialen nach RÖLTGEN oder die Konfiguration AR-basierter Validierungsumgebungen nach REINEMANN. Im Kontext der AR-System-Konfiguration macht SCHILLING relevantes Wissen manuell zugänglich und PAELKE ET AL. unterstützen die Evaluierung von AR-Displays durch eine Checkliste. All diese Ansätze stellen eher Vorgehensweisen zur Nutzung von Wissen. Eine automatische Auswertung von Wissen, was einer maximalen Zugänglichkeit entsprechen würde, findet nicht statt. Wissensbasierte Systeme und semantische Technologien stellen jedoch einen vielversprechenden Ansatz dar, um Wissen für die AR-System-Konfiguration zugänglich zu machen.

A7) Aufwandsarme Entwicklung: Die Recherche kommerziell und wissenschaftlich verfügbarer Software-Lösungen zeigt, dass keine Software für alle Validierungsszenarien verwendet werden kann. Dies bestätigt die Notwendigkeit der aufwandsarmen Entwicklung individueller AR-Validierungssoftware. Diese wird jedoch von keinem der analysierten Ansätze ermöglicht. *Unity* stellt eine gute Grundlage der Entwicklung dar. Verschiedene grundlegende Funktionalitäten werden zudem von SDKs bereitgestellt. Darauf aufbauend gilt es, ein Konzept zu erarbeiten, dass für die Validierung relevante Funktionen auf verschiedenen AR-Endgeräten wiederverwendbar macht. Für die Definition der relevanten Funktionalitäten kann auf die Funktionsbeschreibungen der analysierten AR-Anwendungssoftware im Kontext Validierung zurückgegriffen werden.

A8) Systematische Vorgehensweise: Im Stand der Technik wurden verschiedene Ansätze untersucht, die eine systematische Vorgehensweise beinhalten. Keiner der Ansätze adressiert vollumfänglich die Planung und Vorbereitung AR-basierter Validierung. Die Ansätze betreffen jeweils lediglich Teilespekte der vorliegenden Arbeit. Das House-of-Quality nach KLEIN sowie die Nutzen-Aufwands-Bewertung von AR-Potentialen nach RÖLTGEN beinhalten beispielsweise hilfreiche Aspekte zur Gestaltung der Planung AR-basierte Validierung. Für die Vorgehensweise zur AR-System-Konfiguration beschreiben RÖLTGEN und HASSAN hilfreiche Ansätze. Für die aufwandsarme AR-Entwicklung stellt zudem der AR-Entwicklungsprozess eine wichtige Grundlage dar. So bedarf jedes der

drei Handlungsfelder einer individuellen Vorgehensweise, die übergeordnet in einem gemeinsamen Vorgehensmodell zusammengefasst werden muss.

A9) Anwendbarkeit: Die meisten der analysierten Ansätze lassen sich grundsätzlich auf geeignete Art und Weise von Anwendern nutzen. Jedoch können lediglich No-Code AR-Editoren sowie die systematischen Ansätze nach RÖLTGEN die Anforderung nach Anwendbarkeit komplett erfüllen. Viele der anderen Ansätze sind weiterhin mit Erfahrung verbunden oder nicht ausreichend angeleitet. So wird der Einsatz durch unerfahrene Anwender erschwert. Im Rahmen dieser Arbeit gilt es daher umso mehr, Lösungen zu schaffen, die von Unternehmen ohne spezielle Erfahrung im Bereich AR mit entsprechender Anleitung genutzt werden können.

Keiner der untersuchten Ansätze und auch keine triviale Kombination bestehender Ansätze erfüllen alle an die Arbeit gestellten Anforderungen. Viele Ansätze adressieren Teilaspekte des Gesamtproblems. Es existiert jedoch keine übergreifende Lösung, die Unternehmen in die Lage versetzt, die Technologie AR bedarfsgerecht als Validierungswerkzeug zu nutzen. Es besteht demnach dringender Handlungsbedarf für ein *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*.

5 Instrumentarium zur Planung und Vorbereitung AR-basierter Validierung

Dieses Kapitel bildet den Kern der vorliegenden Arbeit. Es stellt das *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung* vor. Das Instrumentarium hat den Anspruch, dem aufgezeigten Handlungsbedarf (Kapitel 3.5 und 4.4) unter Berücksichtigung des Stands der Technik (Kapitel 1) gerecht zu werden und die an die Arbeit gestellten Anforderungen (Kapitel 3.6) zu erfüllen. Kapitel 5.1 gibt zunächst einen Überblick über das Instrumentarium und seine wesentlichen Bestandteile. Das Instrumentarium soll sich aus vier Lösungsbausteinen (LB) zusammensetzen, die den in Kapitel 3.5 aufgezeigten Handlungsfeldern entsprechen. Erster Lösungsbaustein ist eine Systematik zur Nutzen-Aufwands-Analyse der Potentiale von AR im Kontext Validierung, die Unternehmen basierend auf grundlegenden AR-Potentiale und den vorliegenden Anforderungen einen systematischen Einstieg in die Planung der Validierungsaktivitäten ermöglicht. Die Systematik wird in Kapitel 5.2 vorgestellt. Kapitel 5.3 beschreibt die wissensbasierte AR-System-Konfiguration basierend auf einer AR-System-Wissensbasis und einem interaktiven Hilfsmittel zur Konfiguration. Durch den dritten Lösungsbaustein werden Unternehmen zur aufwandsarmen Entwicklung individueller AR-Validierungsanwendungen befähigt (Kapitel 5.4). Das in Kapitel 5.5 beschriebene übergreifende Vorgehensmodell leitet den Einsatz der Methoden und Hilfsmittel für die verschiedenen Aktivitäten an, ausgehend von der Planung über die Konfiguration des AR-Systems bis hin zur Entwicklung der AR-Validierungssoftware. Das Instrumentarium ermöglicht somit die bedarfsgerechte Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung.

In diesem Kapitel wird das erarbeitete Instrumentarium aus methodischer Sicht beschrieben. Die durchgängige Anwendung und Evaluierung des Instrumentariums sind anschließend Gegenstand von Kapitel 6.

5.1 Das Instrumentarium im Überblick

Das *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung* besteht aus vier Lösungsbausteinen (LB). Die Lösungsbausteine adressieren die Herausforderungen in den vier in Kapitel 3.5 aufgezeigten Handlungsfeldern (Abbildung 5-1):

- Eine **Systematik zur Nutzen-Aufwands-Analyse von AR-Potentialen** (Kapitel 5.2) unterstützt Unternehmen bei der eigenständigen Bewertung der individuellen Potentiale für den Einsatz von AR als Validierungswerkzeug. Zudem ermöglicht die Systematik die Definition grundlegender Anforderungen an die Datenvorbereitung für die Validierung.

- Durch die **wissensbasierte AR-System-Konfiguration** (Kapitel 5.3) wird Unternehmen eine zielgerichtete AR-Geräte-Auswahl sowie die Definition eines bedarfsgerechten Funktionsumfangs der AR-Validierungssoftware ermöglicht. Dazu wird relevantes Wissen in einer AR-System-Wissensbasis formalisiert und durch ein interaktives Hilfsmittel zugänglich gemacht. So werden Unternehmen zur anwendungsfallspezifischen Konfiguration des AR-Systems ohne spezielles AR-Knowhow befähigt.
- Um die Entwicklung zu vereinfachen, wird ein **Werkzeugkoffer zur aufwandsarmen Entwicklung individueller AR-Validierungsanwendungen** bereitgestellt. Der Werkzeugkoffer beinhaltet ein AR-Validierungs-Framework, das für die Validierung relevante Funktionalitäten als wiederverwendbare Funktionsbausteine sowie weitere Hilfsmittel enthält. In Kombination mit einem Leitfaden ermöglicht das Framework eine aufwandsarme Entwicklung individueller AR-Validierungsanwendungen (Kapitel 5.4) zum Einsatz in der Produktentwicklung.
- Ein übergreifendes **Vorgehensmodell** (Kapitel 5.5) leitet den Einsatz der Methoden und Hilfsmittel für die verschiedenen Aktivitäten an, ausgehend von der Planung über die Konfiguration des AR-Systems bis hin zur Entwicklung der AR-Validierungsanwendung. Das Vorgehensmodell versetzt Unternehmen somit in die Lage, die bedarfsgerechte Planung und Vorbereitung von AR-Lösungen als Validierungswerkzeug eigenständig durchzuführen.



Abbildung 5-1: Lösungsbausteine (LB) des Instrumentariums

5.2 Systematik zur Nutzen-Aufwands-Analyse von AR-Potentialen

Die Systematik zur Nutzen-Aufwands-Analyse von AR-Potentialen bildet den Einstieg in die Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung. Ausgangspunkt der Systematik sind Anforderungen an das zu entwickelnde Produkt, die es zu validieren gilt. Ziel der Systematik ist eine **Entscheidung für oder gegen den Einsatz von AR als Validierungswerkzeug**. Dabei setzt der Einsatz der Systematik voraus, dass die Validierung grundsätzlich digital durchgeführt werden soll. Es gilt entsprechend

zu entscheiden, ob AR als Validierungswerkzeug für die vorliegenden Anforderungen sinnvoll ist und einen Mehrwert gegenüber alternativen digitalen Ansätzen bietet. Aufgrund mangelnden Technologie-Knowhows und fehlender Erfahrung kann diese Entscheidung von vielen Unternehmen nicht eigenständig getroffen werden. Die Systematik ermöglicht daher die individuelle Analyse und Bewertung der grundsätzlichen Potentiale von AR zur Unterstützung der Validierung der vorliegenden Anforderungen. Zum einen wird der Nutzen von AR zur Validierung der Anforderungen bewertet. Dem Nutzen werden zum anderen die entsprechenden notwendigen Aufwände gegenübergestellt. Darüber hinaus wird ein Vergleich mit anderen digitalen Ansätzen unterstützt. So kann der Anwender abschätzen, ob AR wirklich notwendig bzw. sinnvoll ist, um eine aussagekräftige Validierung zu ermöglichen. Zu den alternativen Ansätzen zählen zum einen die klassische 2D-Darstellung digitaler Daten und Modelle am Bildschirm und zum anderen Virtual Reality als immersive Visualisierungstechnologie. Ein Vergleich mit dem Einsatz rein physischer Prototypen wird von der Systematik nicht adressiert. Eine Entscheidung zwischen rein physischer und digital unterstützter Validierung sollte entsprechend im Vorhinein getroffen werden.

Im folgenden Abschnitt 5.2.1 wird zunächst ein Überblick über die Systematik gegeben. Anschließend werden der AR-Potential-Bewertungsgraph (Abschnitt 5.2.2) und das interaktive Hilfsmittel (Abschnitt 5.2.3) als die zwei wesentlichen Bestandteile der Systematik im Detail vorgestellt.

5.2.1 Die Systematik im Überblick

Ausgangssituation und Input für die Systematik ist eine Menge an Anforderungen, die möglicherweise mittels AR validiert werden sollen. Wie der Stand der Technik zeigt, werden die Anforderungen an das Produkt im Lastenheft definiert und liegen meist in natürlicher Sprache vor. Sie sind stets individuell und können beliebige Produktaspekte bzw. -merkmale betreffen (vgl. Abschnitt 3.2.2). Die Systematik soll jedoch eine **allgemeingültige, generische Lösung** darstellen, die unabhängig vom Produkt, vom Unternehmen und von den Anforderungen eingesetzt werden kann. Daher werden die Potentiale von AR zur Unterstützung der Validierung sowie die Nutzen- und Aufwandsaspekte grundsätzlich generisch definiert und bewertet. Durch eine **Generalisierung** der jeweils vorliegenden Anforderungen wird bei jedem Einsatz der Systematik eine Übertragung dieser generischen Definitionen und Bewertungen auf die individuelle Situation realisiert. Somit ermöglicht die Systematik die Analyse und Bewertung der Potentiale von AR für beliebige individuelle Anforderungen. Das versetzt Unternehmen in die Lage, zu entscheiden, ob AR als Validierungswerkzeug eingesetzt wird.

Die Systematik setzt sich aus zwei wesentlichen Bestandteilen zusammen (Abbildung 5-2). Den Kern der Systematik bildet der **AR-Potential-Bewertungsgraph**. Der Bewertungsgraph ist inspiriert durch das Agile Practices Impact Model (APIM) (Abschnitt

4.1.4), bei dem die Auswahl verschiedener agiler Praktiken durch in einer Graphen-Struktur dokumentiertes Wissen unterstützt wird. Die grundsätzliche Idee des Bewertungsgraphen wurde im Rahmen einer bisher unveröffentlichten Bachelorarbeit erarbeitet, betreut durch den Autor der vorliegenden Arbeit [KE21]. Ähnlich zum APIM wird im AR-Potential-Bewertungsgraph das für die Nutzen-Aufwands-Analyse der AR-Potentiale relevante Wissen dokumentiert und miteinander in Beziehung gebracht. Der Bewertungsgraph besteht aus einem statischen Teil und einem dynamischen Teil.

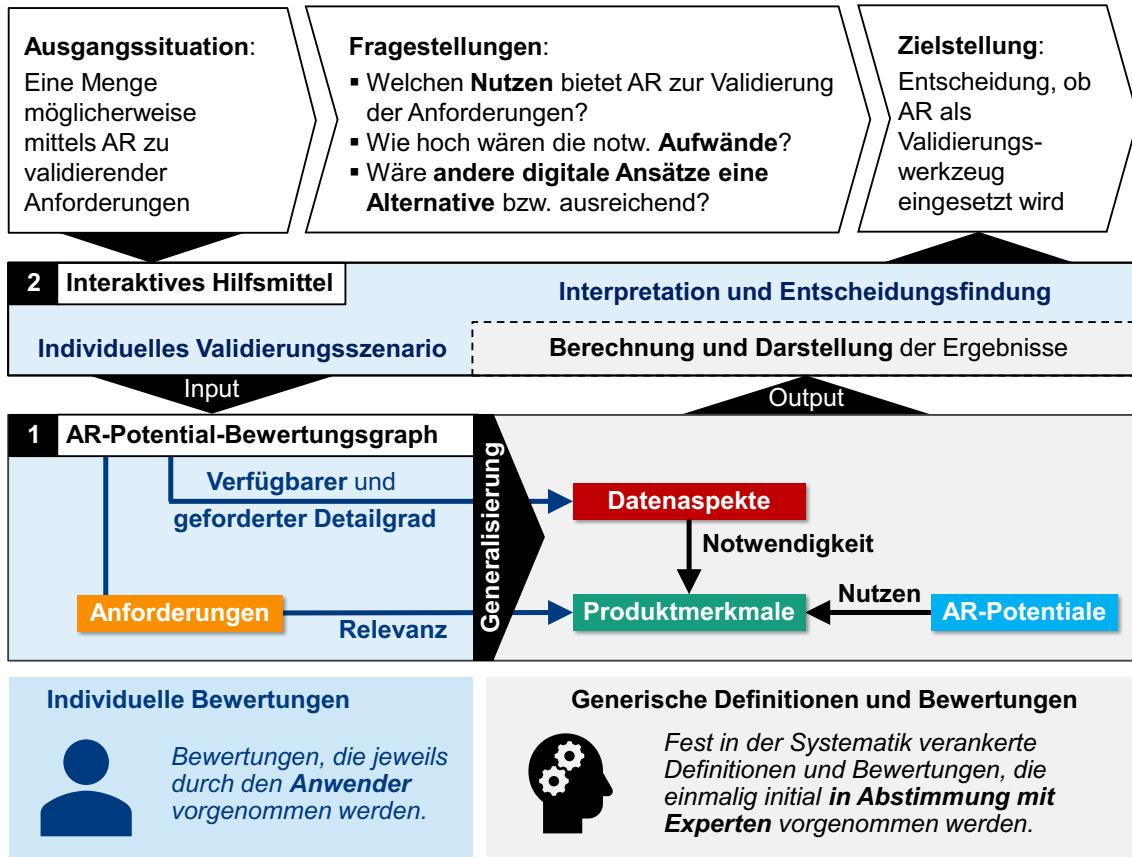


Abbildung 5-2: Die Systematik zur Nutzen-Aufwands-Analyse von AR-Potentialen im Überblick

Der statische Teil (graue Bereiche in Abbildung 5-2) enthält **generische Definitionen und Bewertungen**, die einmalig initial durch Experten vorgenommen werden. Die Definitionen und Bewertungen werden statisch in der Systematik verwendet und bilden die Grundlage der Nutzen-Aufwands-Analyse von AR-Potentialen im Kontext der Validierung. Folgende drei Arten von Knoten sind im statischen Teil vorhanden:

- Generische **Produktmerkmale**, die durch die individuellen Anforderungen adressiert werden können. Die Produktmerkmale umfassen alle möglicherweise für die Validierung relevanten Aspekte eines Produkts.
- **AR-Potentiale**, die den grundsätzlichen funktionalen Möglichkeiten von AR zur Unterstützung der Validierung entsprechen.

- Generische **Datenaspekte**, die der Repräsentation des Produkts oder einzelner Produktmerkmale dienen und für die Validierung relevant sein können. Basierend auf den Datenaspekten werden entsprechend Prototypen erstellt.

Die Knoten bilden alle für die Bewertung von AR-Potentialen im Kontext der Validierung erforderlichen Aspekte ab. Über gewichtete Kanten werden die Knoten miteinander in Beziehung gebracht und Bewertungen durchgeführt. So erfolgt zum einen eine Nutzenbewertung zwischen den einzelnen Produktmerkmalen und den AR-Potentialen. Dabei wird jeweils der **Nutzen** eines AR-Potentials zur Validierung der einzelnen Produktmerkmale bewertet. Details dazu folgen in Abschnitt 5.2.2.1. Zudem werden die Produktmerkmale mit den Datenaspekten in Beziehung gebracht. An den entsprechenden Kanten erfolgt eine Bewertung der **Notwendigkeit** der einzelnen Datenaspekte für die Validierung der jeweiligen Produktmerkmale. Die Notwendigkeit einzelner Datenaspekte wird später für die Bewertung der notwendigen Aufwände verwendet (Abschnitt 5.2.2.2).

Im dynamischen Teil des Bewertungsgraphen werden jeweils bei jeder Anwendung der Systematik **individuelle Bewertungen durch den Anwender** vorgenommen. Als Knoten enthält dieser Teil des Graphen die zu analysierenden individuellen **Anforderungen**. Durch die Bewertung der **Relevanz** der generischen Produktmerkmale für jede Anforderung erfolgt eine **Generalisierung** der individuellen Anforderungen. Ähnlich zum Ansatz des House of Quality nach KLEIN (Abschnitt 4.1.1) kann so die individuelle Relevanz der generischen Produktmerkmale für die geplante Validierung bestimmt werden. Darauf aufbauend erfolgen alle weiteren Analyse- und Bewertungsschritte basierend auf den generischen Bewertungen im statischen Teil des Bewertungsgraphen. Zur individuellen Bewertung der Aufwände definiert der Anwender zudem den für die Validierung **gewünschten sowie den bereits verfügbaren Detailgrad einzelner Datenaspekte**. So ermöglicht der AR-Potential-Bewertungsgraph basierend auf einer generischen Bewertung der AR-Potentiale eine individuelle Analyse und Bewertung der Potentiale entsprechend der vorliegenden Anforderungen und somit eine allgemeingültige Lösung. Wie dies genau erfolgt, wird in Abschnitt 5.2.2.2 beschrieben.

Aufbauend auf dem AR-Potential-Bewertungsgraph wird dem Anwender ein **interaktives Hilfsmittel** zur Nutzen-Aufwands-Analyse bereitgestellt. Es dient als Benutzeroberfläche zum Einsatz des AR-Potential-Bewertungsgraphen. Zum einen erfolgen über das Hilfsmittel die individuellen Bewertungen als Anwendereingaben. Entsprechend der Anwendereingaben erfolgen die notwendigen Berechnungen der Nutzen-Aufwands-Analyse basierend auf den generischen Definitionen und Bewertungen im Bewertungsgraphen. Die Ergebnisse werden wiederum über das Hilfsmittel ausgegeben und dargestellt. Der Anwender wertet die Ergebnisse aus und nutzt sie zur Entscheidungsfindung. Eine detaillierte Beschreibung des Hilfsmittels erfolgt in Abschnitt 5.2.3.

5.2.2 AR-Potential-Bewertungsgraph

Im AR-Potential-Bewertungsgraph wird das für die Nutzen-Aufwands-Analyse von AR-Potentialen im Kontext Validierung relevante Wissen dokumentiert. Dies entspricht den generischen Definitionen und Bewertungen im statischen Teil des Graphen. Zudem nimmt der Anwender über das interaktive Hilfsmittel individuelle Bewertungen im dynamischen Teil des Graphen vor. Der Bewertungsgraph dient anschließend als Grundlage für Berechnungen der Nutzen-Aufwands-Analyse. Zusätzlich zu den in Abbildung 5-2 dargestellten Zusammenhängen werden den Knoten im AR-Potential-Bewertungsgraph individuelle Attribute zugeordnet. Zudem werden aus den Bewertungen zwischen den Knoten weitere Attribute abgeleitet. Um alle Informationen gemeinsam in einem Modell darstellen zu können, wurde angelehnt an das House of Quality nach KLEIN (Abschnitt 4.1.1) die **City-of-Augmented-Reality-based-Validation (CARV)** entwickelt (Abbildung 5-3). Die Funktion und Verwendung der CARV werden im Folgenden anhand der in Abbildung 5-3 dargestellten Schritte erläutert. Die dargestellten Werte entsprechen grundsätzlich den in der Systematik verwendeten Definitionen und Bewertungen. Lediglich die Anforderungen und entsprechende Bewertungen sind exemplarisch definiert.

5.2.2.1 Generische Definitionen und Bewertungen

Die generischen Definitionen und Bewertungen sind fest im Bewertungsgraph und somit in der Systematik verankert. Sie ermöglichen einen allgemeingültigen Einsatz der Systematik. Zunächst wurden die Produktmerkmale, AR-Potentiale und Datenaspekte basierend auf dem Stand der Technik sowie der Technologie-Expertise und der Erfahrung des Autors definiert. Ebenso wurden die Bewertungen des Nutzens, der Notwendigkeit der Datenaspekte sowie der Aufwände zunächst durch den Autor durchgeführt. Die Definitionen und Bewertungen wurden anschließend mit Fachexperten aus der Produktentwicklung und dem Themenfeld Augmented Reality diskutiert und angepasst. Dabei wurden die Bewertungen durch Anwendung an exemplarischen Anforderungen so abgestimmt, dass die Nutzen-Aufwands-Analyse zu realistischen, den Einschätzungen der Experten entsprechenden Ergebnissen führt. Die generischen Definitionen und Bewertungen entsprechen den in Abbildung 5-3 in weißen Kreisen dargestellten Schritten.

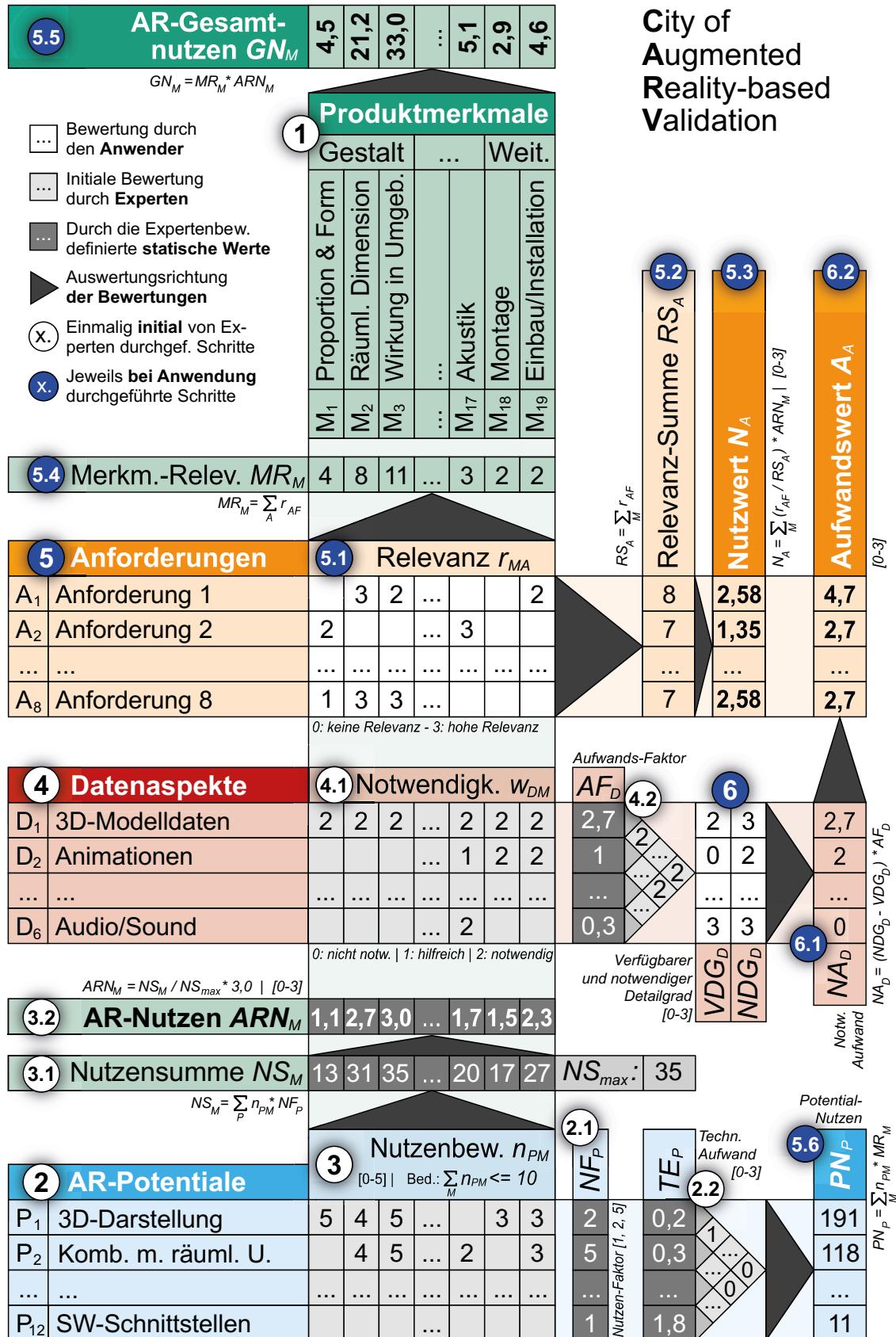


Abbildung 5-3: City-of-Augmented-Reality-based-Validation (CARV)

(1) Produktmerkmale

Es wurden 19 generische Produktmerkmale (M_1-M_{19}) definiert. Sie entsprechen möglichen durch die Anforderungen adressierten Aspekten des Produkts. Sie sind generisch definiert und haben den Anspruch, alle für die Nutzen-Aufwands-Analyse von AR-Potentialen relevanten Aspekte abzudecken. Als Grundlage der Definition wurden die Hauptmerkmale eines Produkts nach PAHL ET AL. herangezogen (s. Anhang A2.1.1). Sie wurden unter Berücksichtigung der von LIM ET AL. allgemein sowie der von REINEMANN ET AL. speziell für AR-basierte Prototypen beschriebenen Wiedergabedimensionen angepasst und überarbeitet (s. Abschnitt 4.1.5 und Anhang A2.1.2). Von REINEMANN ET AL. wurden Wiedergabedimensionen, die beim Einsatz von AR nicht relevant sind, bereits vernachlässigt. Dazu zählen beispielsweise das Gewicht oder der Härtegrad. Prototypen entsprechen Repräsentationen von Produkten, die eine Auswahl an Produktmerkmalen wiedergeben. Entsprechend können die Wiedergabedimensionen von Prototypen auch als Produktmerkmale gesehen werden. So wurden für die Bewertung von AR zur Unterstützung der Validierung die in Abbildung 5-4 dargestellten generischen Produktmerkmale definiert. Sie werden unterteilt in sieben Kategorien:

Gestalt	Proportionen und Form	Interaktion	Benutzerschnittstelle
	Räumliche Dimension, Maße, Abstände		Einsehbarkeit
	Lage und Wirkung in realer Umgebung		Erreichbarkeit
Optik	Farb-/Lichtwirkung	Struktur	Systemstruktur
	Transparenz		Konnektivität
	Oberflächenstruktur		Anschlüsse u. externe Wirkbeziehungen
Funktionalität	Internes Systemverhalten	Nonvisuelle Erscheinung	Haptik
	Endnutzer-Funktionalität		Akustik
	Kinematik	Weitere	Montage/Zusammenbau
			Einbau/Installation

Abbildung 5-4: Generische Produktmerkmale unterteilt in sieben Kategorien

- Die *Gestalt*-Produktmerkmale umfassen geometrische Aspekte sowie deren Bezug zur realen Umgebung.
- Die Kategorie *Optik* beinhaltet Aspekte der visuellen Erscheinung des Produkts.
- Die *Funktionalität* umfasst sowohl das interne Verhalten und die Kinematik als auch die für den Benutzer relevante nach außen gerichtete Funktionalität.
- Die Kategorie *Interaktion* beinhaltet Aspekte, die bei der Interaktion zwischen Benutzer und Produkt relevant sind.
- *Struktur*-Produktmerkmale adressieren sowohl die innere Struktur des Produkts als auch Zusammenhänge mit externen Elementen und Systemen.

- Zu den *Nonvisuellen* Produktmerkmalen zählen die durch weitere Sinne wahrgenommenen Aspekte der Haptik und Akustik.
- Die Kategorie *Weitere* umfasst die für die Produktion und Inbetriebnahme des Produkts relevanten Aspekte der Montage und des Zusammenbaus.

(2) AR-Potentiale

AR-Potentiale entsprechen den grundsätzlichen Möglichkeiten von AR zur Unterstützung der Validierung (Abbildung 5-5). Für die Definition der Potentiale wurden unter anderem verschiedene Arbeiten und Publikationen herangezogen, die die grundsätzliche Funktionsweise von AR und damit verbundene Potentiale beschreiben. Weitere Erkenntnisse bzgl. der Potentiale von AR zur Unterstützung der Validierung konnten aus den von RÖLTGEN definierten AR-Anwendungsszenarien (vgl. Abschnitt 4.1.3) sowie den vorhandenen AR-Anwendungen im Kontext der Validierung (s. Abschnitt 4.3.3) gewonnen werden. So wurden 12 generische AR-Potentiale (P_1-P_{12}) in den Kategorien *Darstellung*, *Interaktion*, *Anbindung* und *Weitere* definiert.

		Alleinstellungsmerkmal	Nutzen-Faktor
		AR / VR / 3D	
Darstellung	Kombination mit realer Umgebung	AR-spezifisch	5
	Dreidimensionale Darstellung	AR oder VR	2
	Animation der Inhalte		1
	Funktionale Interaktivität		1
	Flexible Inhaltsbearbeitung	Digital 3D	1
	Darstellung von Meta-Informationen		1
	Personalisierte Ansichten		1
Interaktion	Einbeziehung des Benutzers	AR oder VR	2
	Natürliche Benutzereingaben		2
	Tangible User Interface		2
	Akustische Ausgabe	Digital 3D	1
Anbindung	Software-Schnittstellen	Digital 3D	1
Weitere	Integriertes virtuelles Feedback	AR oder VR	2
	Digitale Telemetrie und Dokumentation		2
	Automatisierte Benutzerführung		1
	Ortsunabhängigkeit	Digital 3D	1
	Kollaboration		1

Abbildung 5-5: AR-Potentiale im Kontext Validierung inklusive Nutzen-Faktoren

(2.1) Nutzen-Faktor: Die durch die Potentiale repräsentierten funktionalen Möglichkeiten sind teilweise auch mit anderen digitalen Ansätzen realisierbar. Entsprechend wird für jedes Potential bewertet, ob es ein Alleinstellungsmerkmal von AR ist oder ob es sich ebenfalls mit Virtual Reality (VR) oder sogar klassischen digitalen 3D-Visualisierungen

am Bildschirm realisieren lässt. Für diese drei Unterscheidungen wurde jeweils ein Nutzen-Faktor NF_P definiert, der für die Nutzen-Aufwands-Analyse relevant ist (s. Abschnitt 5.2.2.2). AR-spezifische Potentiale erhalten einen Faktor von 5. VR-fähige Potentiale werden mit 2 bewertet. Potentiale, die auch mittels digitaler 3D-Lösungen genutzt werden können, haben eine Faktor von 1. Die AR-Potentiale sind unterteilt in vier Kategorien:

- Die Kernfunktionalität von AR ist die dreidimensionale **Darstellung** von digitalen bzw. virtuellen Modellen. Ein Alleinstellungsmerkmal von AR und somit AR-spezifisch ist die Kombination und Einbettung der Darstellungen in realen Umgebungen. Die dreidimensionale Darstellung dagegen ließe sich auch mit VR realisieren. Zudem bietet AR weitere Möglichkeiten der Darstellung, die auch mit klassischen digitalen Darstellungsansätzen möglich sind. Inhalte können interaktiv verändert oder flexibel ausgetauscht werden. Zudem können Zusatzinformationen dargestellt und individuelle Ansichten eines Prototyps dargestellt werden.
- Bei der **Interaktion** ermöglicht AR wie auch VR die Einbeziehung des Nutzers sowie natürliche und taktile Benutzereingaben. So können unter anderem auch physische Interaktionsobjekte als Tangible User Interfaces in die Anwendungen integriert und mit virtuellen Modellen kombiniert werden. Ebenso sind akustische Ausgaben möglich, die sich auch mit anderen Ansätzen realisieren lassen.
- Alle digitalen Ansätze bieten die Möglichkeit der **Anbindung** weiterer Software und digitaler Systeme, um beispielsweise Daten auszutauschen.
- **Weitere** Möglichkeiten beinhalten virtuelles dreidimensionales Feedback direkt am 3D-Modell. So kann der Benutzer beispielsweise durch natürliche Interaktionslösungen aussagekräftiges Feedback an den relevanten Stellen des Prototypen geben. Zudem ermöglichen sowohl AR als auch VR eine digitale Telemetrie und Dokumentation und somit eine realitätsnahe Auswertung der Benutzerinteraktion und des Benutzerverhaltens. Grundsätzlich kann mit AR und weiteren digitalen Ansätzen eine automatisierte Benutzerführung ähnlich einer schrittweisen Präsentation des Produkts erfolgen. Zudem wird je nach technischer Umsetzung ein ortsunabhängiger Einsatz der Anwendung sowie eine kollaborative Validierung ermöglicht.

(2.2) Technischer Aufwand: Um dem Anwender Auskunft über die voraussichtlichen Aufwände zur Nutzung der AR-Potentiale zu geben, wurde für jedes AR-Potential P ein technischer Aufwand TA_P bestimmt. Die Aufwände ergeben sich aus einer qualitativen Gegenüberstellung in Form einer Korrelationsmatrix wie sie auch im House of Quality nach KLEIN (Abschnitt 4.1.1) eingesetzt wird. Für jedes AR-Potential wird jeweils bewertet, ob die Aufwände zur Nutzung des Potentials geringer, gleich oder größer sind als die der anderen AR-Potentiale (Abbildung 5-6). Dazu werden Zahlenwerte 0, 1 und 2 verwendet. Der Vergleich berücksichtigt, dass entsprechende Entwicklungsbausteine (s. Kapitel 5.4.3.2) bereitstehen und die notwendigen Daten vorbereitet sind. Aus der Gegenüberstellung der Aufwände und einer anschließenden Skalierung ergeben sich technische Aufwände von 0 bis 3. Eine **3D-Darstellung** lässt sich beispielsweise mit geringen

technischen Aufwänden realisieren ($TA_{PI} = 0,2$), da es sich um eine grundlegende Funktionalität der Technologie AR handelt und über die Geräte direkt realisiert wird. Die höchsten Aufwände sind für die Integration einer *funktionalen Interaktivität* notwendig. Diese muss zunächst programmiert und anschließend in die Anwendungslogik integriert und bei Bedarf an die Benutzerschnittstellen angebunden werden.

Vergleich der Aufwände je AR-Potential (vorausgesetzt, entsprechende Entwicklungsbausteine können eingesetzt werden und notwendige Daten sind vorbereitet)													Aufwands-Faktor						
	3D-Darstellung	Kombination mit realer Umgebung	Animation	Funktionale Interaktivität	Flexible, interaktive Inhaltsbearbeitung	Darstellung von Meta-Informationen	Personalisierte Ansichten	Einbeziehung des Nutzers	Natürliche Benutzereingaben	Akustische Ausgabe	Tangible User Interface	Software-Schnittstellen	Integriertes Virtuelles Feedback	Digitale Telemetrie und Dokumentation	Automatisierte Benutzerführung	Ortsunabhängigkeit	Kollaboration	Summe	%
3D-Darstellung	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	0,7
Kombination mit realer Umgebung	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	3	1,1
Animation	2	2	1	1	1	1	1	1	0	1	0	0	1	0	0	2	0	13	4,8
Funktionale Interaktivität	2	2	1	2	2	2	2	2	2	2	2	1	2	1	2	2	2	29	11
Flexible, interaktive Inhaltsbearbeitung	2	2	1	0	1	1	1	1	1	1	1	0	1	0	0	2	1	15	5,5
Darstellung von Meta-Informationen	2	2	1	0	1	0	2	2	1	1	0	0	0	0	0	2	1	15	5,5
Personalisierte Ansichten	2	2	1	0	1	2	2	2	2	2	1	1	0	1	2	2	23	8,5	
Einbeziehung des Nutzers	2	2	1	0	1	0	0	1	1	1	1	0	0	0	0	2	1	12	4,4
Natürliche Benutzereingaben	2	2	2	0	1	0	0	1	1	1	2	0	0	0	0	2	1	15	5,5
Akustische Ausgabe	2	2	1	0	1	1	0	1	1	0	0	0	0	0	0	2	0	11	4
Tangible User Interface	2	1	2	0	1	1	0	1	1	2	1	0	0	0	0	2	1	15	5,5
Software-Schnittstellen	2	2	2	1	2	2	1	2	0	2	1	2	1	2	2	2	25	9,2	
Integriertes Virtuelles Feedback	2	2	1	0	1	2	1	2	2	2	2	0	0	0	0	2	1	20	7,4
Digitale Telemetrie und Dokumentation	2	2	2	1	2	2	2	2	2	2	2	1	2	1	2	2	29	11	
Automatisierte Benutzerführung	2	2	2	0	2	2	1	2	2	2	2	1	2	1	2	1	26	9,6	
Ortsunabhängigkeit	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0,7	
Kollaboration	2	2	2	0	1	1	0	1	1	2	1	0	1	0	1	2	17	6,3	
																		Summe: 272	

(Zeile im Vergleich zu Spalte: 0 - weniger ... , 1 - gleicher ... , 2 - mehr Aufwand)

Abbildung 5-6: Gegenüberstellung der Aufwände zur Nutzung der einzelnen AR-Potentiale zur Definition der Aufwandsfaktoren AF_P je AR-Potential

(3) Nutzenbewertung

Im Rahmen der Nutzenbewertung werden jeweils die Knoten der einzelnen Produktmerkmale mit denen der AR-Potentiale über gewichtete Kanten verbunden. In der CARV entspricht dies einer Matrix-basierten Gegenüberstellung. Für jede Kombination aus einem Potential P und einem Merkmal M wird eine Nutzenbewertung n_{MP} zwischen 0 und 5 definiert. 0 bedeutet, das AR-Potential hat keinen Nutzen zur Validierung des Produktmerkmals. 5 bedeutet einen entsprechenden sehr hohen Nutzen. Eine Bedingung bei der Bewertung ist, dass die Summe der Nutzenwerte je AR-Potential maximal 10 betragen darf. Dies stellt bei weiteren Berechnungen eine Vergleichbarkeit und Skalierbarkeit der Bewertungen sicher. Die *3D-Darstellung* sowie die *Kombination mit der räumlichen Umgebung* haben beispielsweise einen sehr hohen Nutzen zur Validierung der *Lage und Wirkung im Raum*.

(3.1) Nutzensumme: Basierend auf der Nutzenbewertung kann die Nutzensumme NS_M für jedes Merkmal berechnet werden. Sie ergibt sich als Summe aller Nutzenbewertungen je Produktmerkmal jeweils faktorisiert mit den Nutzen-Faktoren der AR-Potentiale. Umso höher die Nutzensumme, umso stärker kann das entsprechende Produktmerkmal mittels AR validiert werden.

(3.2) AR-Nutzen: Die Nutzensummen werden anschließend auf Werte zwischen 0 und 3 skaliert. So ergibt sich der AR-Nutzen ARN_M je Produktmerkmal. Der AR-Nutzen stellt einen sehr wichtigen Teil der Analyse der Potentiale dar. Basierend auf der Generalisierung der Anforderungen (Schritt 5.1) dient der AR-Nutzen der AR-Potentiale als Hauptindikator für die Nutzenbewertung bzgl. der individuellen Anforderungen (Schritt 5.3).

(4) Datenaspekte

Zur Repräsentation des Produkts oder einzelner Produktmerkmale dienen digitale bzw. virtuelle Prototypen, die basierend auf verschiedenen Datenaspekten erstellt werden müssen. Als mögliche Grundlage für die Prototypen wurden daher die in Abbildung 5-7 dargestellten sechs generischen Datenaspekte definiert. Ein grundlegender Datenaspekt von AR sind **3D-Modelldaten**. Darauf aufbauend können **Animationen** zur Präsentation einzelner Produktmerkmale und Veranschaulichung komplexer Sachverhalte dienen. **Funktionalitäten** entsprechen der realen Funktionalität des Produkts oder stellen diese zumindest angenähert nach. Zudem können **Simulationsdaten**, die während der Entwicklung entstehen, für die Validierung relevant sein. Durch die Möglichkeit der Kombination virtueller und realer Objekte, können auch **Materialien** in die Validierung einbezogen werden. Mit **Audio**-Daten können zudem akustische Aspekte validiert oder relevante Umgebungsgeräusche integriert werden. Je nach Validierung sind unterschiedliche Ausprägungen der Datenaspekte notwendig oder gewünscht. Zur Berechnung der notwendigen Aufwände zur Vorbereitung der Validierung wurden daher für jeden Datenaspekt **drei Detailgrade** definiert. Die Idee der Detailgrade basiert auf den von RÖLTGEN verwendeten Aufwandstufen für die Bewertung von AR-Anwendungsszenarien (vgl. Abbildung 4-4). Bei 3D-Modelldaten entspricht Detailgrad 1 beispielsweise groben Form-Modellen, die unter Umständen für eine erste Abschätzung der Produktgröße ausreichen. Über genaue Geometrie-Modelle steigert sich der Detailgrad bis hin zu texturierten Detail-Modellen. Diese sind beispielsweise notwendig zur Validierung von Designaspekten.

(4.1) Notwendigkeit der Datenaspekte: Über gewichtete Kanten werden die Datenaspekte jeweils mit den Produktmerkmalen verbunden, was in der CARV einer weiteren Matrix-Gegenüberstellung entspricht. Es wird jeweils die Notwendigkeit w_{DM} des Datenaspekts D zur Validierung des Produktmerkmals M bewertet. Dabei wird unterschieden, ob der Datenaspekt notwendig ist (2), nicht notwendig ist, aber hilfreich sein kann (1), oder grundsätzlich nicht relevant ist (0). **3D-Modelldaten** sind beispielsweise für die meisten Produktmerkmale relevant. Für die Validierung der *Montage* und des *Einbaus* sind zudem *Animationen* zu verwenden. *Animationen* können auch zur Validierung von *Akustik* beitragen, beispielsweise zur Darstellung von Schallwellen, sie sind aber nicht

erforderlich. Wenn Datenaspekte als nicht notwendig bewertet sind, können sie selbstverständlich dennoch zur Validierung verwendet werden. Die Bewertung soll lediglich eine grundlegende Bewertung der notwendigen Aufwände ermöglichen.

Detailgrad:	1	2	3	Aufwandsfaktor (AF_D)
3D-Modelldaten	Große Form	Genaue Geometrie	Texturierte Detail-Modelle	2,7
Animationen	Einfache Hinweise	Transformationen	Komplexe Animationen	1
Funktionalitäten	Kontinuierlich	Reaktiv/Interaktiv	Analytisch	3
Simulationsdaten	Werte und Graphen	Darstellung am Modell	Komplexe Visualisierungen	2,3
Materialien	Materialprobe	Form-Prototypen	Material in Form	0,7
Audio/Sound	Einfache Töne	Samples	Reale Aufnahmen	0,3

Abbildung 5-7: Datenaspekte und entsprechende Detailgrade (1 entspricht den geringsten Details und 3 den höchsten) sowie Aufwandsfaktoren AF_D

(4.2) Aufwands-Faktor: Für jeden Datenaspekt wurde zudem ein Aufwands-Faktor definiert. Die Aufwandsfaktoren der einzelnen Datenaspekte ergeben sich wie die technischen Aufwände der AR-Potentiale aus einer qualitativen Gegenüberstellung der Aufwände in einer Korrelationsmatrix. Für jeden Datenaspekt wird jeweils anhand von Zahlenwerten bewertet, ob die Aufwände zur Erstellung der Daten geringer, gleich oder größer sind als die der anderen Datenaspekte (Abbildung 5-8). Der Vergleich der Aufwände erfolgt pauschal. Grundsätzlich können die Aufwände für die verschiedenen Datenaspekte individuell variieren. Eine pauschale Betrachtung reicht für eine grundsätzliche Aufwandsabschätzung und -bewertung im Rahmen der Systematik jedoch aus. Durch eine zeilenweise Summierung und anschließende Skalierung ergeben sich somit pauschale Aufwands-Faktoren AF_D von 0 bis 3 für die Datenaspekte. Der größte Aufwandsfaktor wurde für die Vorbereitung der Funktionalitäten hergeleitet, gefolgt von den 3D-Modelldaten. Geringste Aufwände ergeben sich für Audio-Daten.

5.2.2.2 Anwender-Bewertungen und deren Auswertung

Die Anwender-Bewertungen werden bei jeder Anwendung der Systematik individuell entsprechend des vorliegenden Validierungs-Szenarios vorgenommen.

(5) Anforderungen

Zunächst werden die individuellen zu analysierenden Anforderungen in der CARV eingetragen. Dies entspricht der Integration als Knoten in den AR-Potential-Bewertungsgraph (s. Abbildung 5-2). Es müssen nicht alle an das Produkt gestellten Anforderungen eingetragen werden. Der Anwender kann entscheiden, welche Anforderungen möglicherweise mittels AR validiert und entsprechend im Rahmen der Systematik analysiert werden sollen.

Vergleich der Aufwände für die einzelnen Datenaspekte		3D-Modelldaten	Animationen	Funktionalitäten	Simulationsdaten	Materialien	Audio/Sound	Summe	%	Aufwandsfaktor
3D-Modelldaten		2	1	1	2	2	2	8	27	2,7
Animationen		0	0	0	1	2	3	10	1	
Funktionalitäten		1	2		2	2	2	9	30	3
Simulationsdaten		1	2	0		2	2	7	23	2,3
Materialien		0	1	0	0		1	2	6,7	0,7
Audio/Sound		0	0	0	0	1		1	3,3	0,3
		Summe: 30								
(Zeile im Vergleich zu Spalte: 0 - weniger ... , 1 - gleicher ... , 2 - mehr Aufwand)										

Abbildung 5-8: Gegenüberstellung der Aufwände für die Erstellung und Aufbereitung der einzelnen Datenaspekte zur Definition der pauschalen Aufwandsfaktoren je Datenaspekt

(5.1) Relevanz: Anschließend erfolgt eine individuelle Bewertung der Relevanz der Produktmerkmale für die vorliegenden Anforderungen. Die Relevanzbewertung dient der Generalisierung der individuell definierten Anforderungen basierend auf den generischen Produktmerkmalen. Die Relevanz r_{MA} eines Produktmerkmals M für eine Anforderung A wird mit Werten von 0 bis 3 bewertet. 0 bedeutet, das Produktmerkmal ist nicht relevant für die Anforderung. Die Anforderung betrifft dieses Produktmerkmal entsprechend in keiner Weise. Eine 3 stellt die maximale Relevanz dar. Die Anforderung betrifft somit sehr stark das Produktmerkmal.

(5.2) Relevanz-Summe: Basierend auf der Relevanzbewertung erfolgt für jede Anforderung die Berechnung der Relevanz-Summe RS_A .

(5.3) Nutzwert: Über die Relevanz-Summe lässt sich eine prozentuale Gewichtung der Produktmerkmale je Anforderung bestimmen. Der Nutzwert N_A je Anforderung ergibt sich als gewichtete Summe der AR-Nutzen ARN_M über die Produkterkmale. Entsprechend beschreibt der AR-Nutzwert jeweils, wie groß der Nutzen von AR zur Validierung der Anforderung ist. Durch die Generalisierung der Anforderungen lässt sich die generische Nutzenbewertung somit auf die individuellen Anforderungen anwenden.

(5.4) Merkmal-Relevanz: Die Merkmal-Relevanz MR_M für ein Produktmerkmal ergibt sich als Summe der Relevanzwerte r_{AM} , die für das Merkmal vergeben wurden.

(5.5) AR-Gesamtnutzen: Um dem Anwender der Systematik weitere Hilfestellung bei der Entscheidungsfindung zu geben, wird der AR-Gesamtnutzen GN_M je Produktmerkmal berechnet. Dieser ergibt sich jeweils aus der Multiplikation des AR-Nutzen ARN_M mit der Merkmal-Relevanz. Der AR-Gesamtnutzen gibt Auskunft darüber, wie hoch der Nutzen von AR für die einzelnen Produktmerkmale im vorliegenden Validierungsszenario ist.

(5.6) Potential-Nutzen: Um später die Auswahl geeigneter AR-Potentiale zu unterstützen, wird zudem der Potential-Nutzen PN_P für jedes AR-Potential berechnet. Er ergibt sich jeweils aus der Summe der Merkmal-Relevanz MR_M multipliziert mit der Nutzenbewertung n_{PM} über alle Produktmerkmale. Wenn die Produktmerkmale, die sich gut mit einem AR-Potential validieren lassen, von hoher Relevanz sind, ergibt sich ein hoher Potentialnutzen für das Potential. Der Potentialnutzen ist entsprechend ein Indikator für den Nutzen der einzelnen Potentiale für die vorliegenden Anforderungen. Je höher der Wert, desto höher der Nutzen.

(6) Beurteilung der Datenaspekte

Für die Bewertung der notwendigen Aufwände für die Vorbereitung der Daten sind zwei Aspekte durch den Anwender zu beurteilen. Zum einen ist der **notwendige Detailgrad NDG_D** der einzelnen Datenaspekte zu definieren. Dieser ist beispielsweise abhängig vom Fortschritt der Entwicklung und von individuellen Anforderungen. So kann eine frühe Validierung anhand eines groben Prototyps gewünscht sein oder auch die Validierung kleinsten Details anhand eines realitätsnahen Prototyps mit hohem Detailgrad. Zudem definiert der Anwender über den **verfügbarer Detailgrad VDG_D** , in welcher Form einzelne Datenaspekte bereits vorliegen.

(6.1) Notwendiger Aufwand: Für die Vorbereitung der Datenaspekte ist lediglich die Differenz der notwendigen und der verfügbaren Detailgrade relevant. Der notwendige Aufwand NA_D je Datenaspekt ergibt sich entsprechend jeweils aus der Differenz der Detailgrade multipliziert mit dem jeweiligen Aufwands-Faktor AF_D .

(6.2) Aufwandswert: Basierend auf den Aufwänden je Datenaspekt können Aufwandswerte A_A für die einzelnen Anforderungen bestimmt werden. Sie ergeben sich aus der Summierung der jeweils für die Anforderung notwendigen Datenaspekte.

Die CARV beschreibt die grundlegende Struktur der Bewertung und die durchgeführten Berechnungen. Zur Anwendung der CARV im Rahmen der Nutzen-Aufwands-Analyse der AR-Potentiale sowie zur Entscheidungsfindung bzgl. des Einsatzes von AR als Validierungswerkzeug dient das im Folgenden beschriebene interaktive Hilfsmittel.

5.2.3 Interaktives Hilfsmittel

Das interaktive Hilfsmittel dient als Benutzerschnittstelle zum AR-Potential-Bewertungsgraph. Es wird in Form einer **Excel-Arbeitsmappe** bereitgestellt. Die Arbeitsmappe beinhaltet die verschiedenen in der CARV verwendet Bewertungs- und Korrelationsmatrizen. Sie werden zur einfacheren Bearbeitung und Nutzung in einem angepassten Format verwendet. So werden die Expertenbewertungen und Anwendereingaben in unterschiedlichen Tabellenblättern vorgenommen. Die generischen Definitionen und Bewertungen als statischer Teil des Graphen sind bereits fest in der Arbeitsmappe vorkonfiguriert (s. Abbildung A-25 in Anhang A3). Das entsprechende Tabellenblatt ist für den Anwender nicht relevant. Die Inhalte werden lediglich für die zuvor beschriebenen Berechnungen

verwendet. Über vorbereitete Eingabetabellen kann der Anwender die zu analysierenden Anforderungen definieren und die notwendigen individuellen Bewertungen vornehmen. Alle Berechnungen zur Nutzen-Aufwands-Analyse werden anschließend über in der Arbeitsmappe integrierte Makros realisiert. Die Ergebnisse der Nutzen-Aufwands-Analyse werden in einem weiteren Tabellenblatt der Arbeitsmappe ausgegeben und dargestellt.

Der Einsatz der Excel-Arbeitsmappe zur Nutzen-Aufwands-Analyse der Potentiale von AR zur Validierung der Anforderungen wird durch ein **Vorgehensmodell** in Form eines Phasen-Meilenstein-Diagramms angeleitet (Abbildung 5-9). Das Vorgehen der Nutzen-Aufwands-Analyse unterteilt sich in vier Phasen. Für jede Phase gibt das Vorgehensmodell die notwendigen Tätigkeiten, zu verwendende Hilfsmittel sowie das zu erwartende Resultat an. In den folgenden Abschnitten werden die vier Phasen genauer vorgestellt.

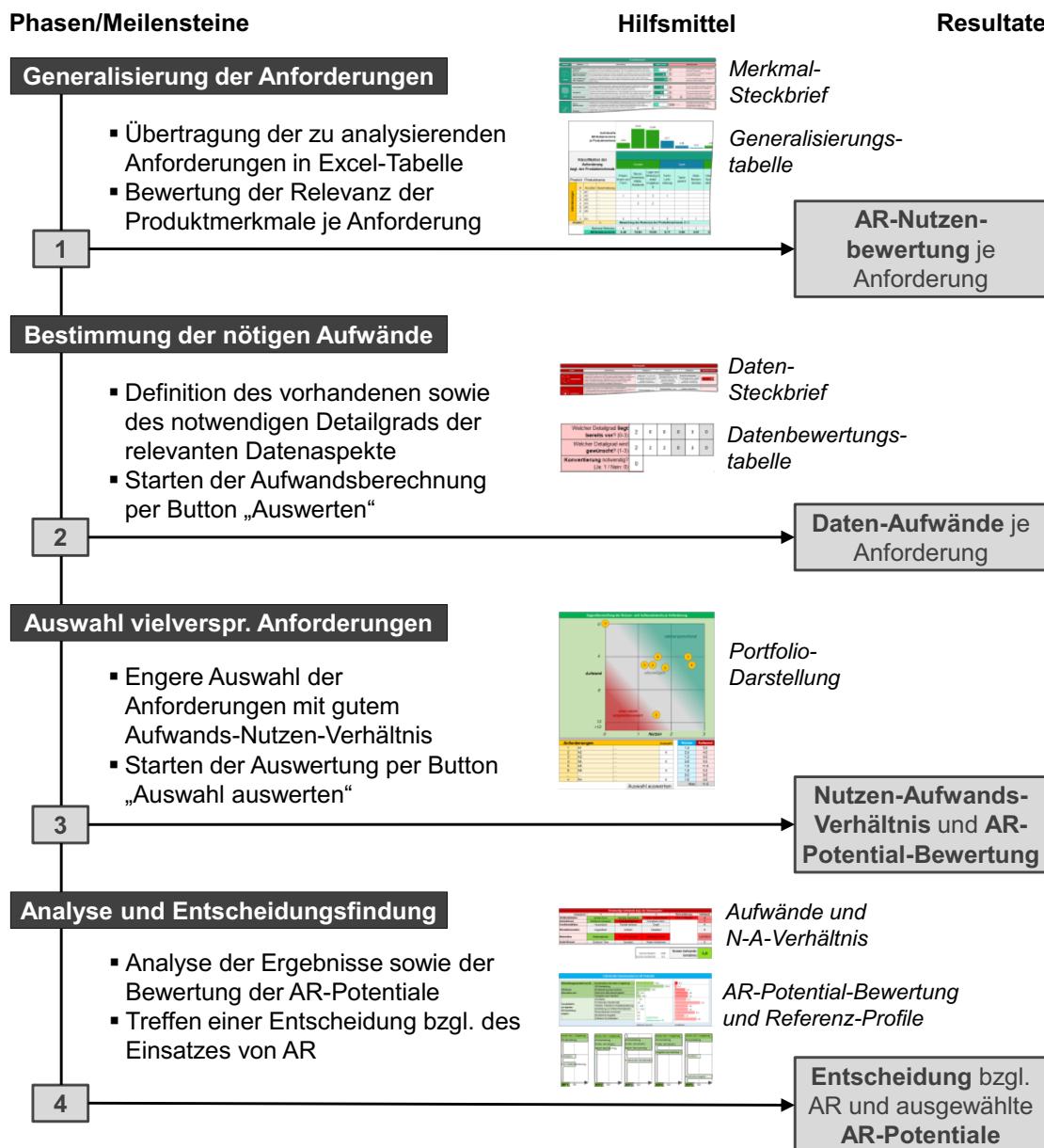


Abbildung 5-9: Vorgehensmodell zur Nutzen-Aufwands-Analyse

5.2.3.1 Generalisierung der Anforderungen

In der ersten Phase der Nutzen-Aufwands-Analyse erfolgt die Generalisierung der zu analysierenden Anforderungen. Die dazu notwendigen Definitionen und Bewertungen durch den Anwender erfolgen in der **Eingabetabelle** (Abbildung 5-10). Links im Gelb dargestellten Bereich können die zu analysierenden Anforderungen inkl. Kurztitel und Beschreibung zeilenweise eingetragen werden. Diese entsprechen den Anforderungs-Knoten im Graphen. Die generisch definierten Produktmerkmale sind bereits spaltenweise vordefiniert. Über die sich daraus ergebende Matrix-Struktur werden die Produktmerkmale mit den Anforderungen in Verbindung gebracht und die Relevanz-Bewertung vorgenommen. In den weißen Feldern wird entsprechend die Relevanz der in den Spalten eingetragenen Produktmerkmale für die Anforderungen in den Zeilen bewertet. Eine leere Zelle entspricht dabei einer Bewertung mit 0.

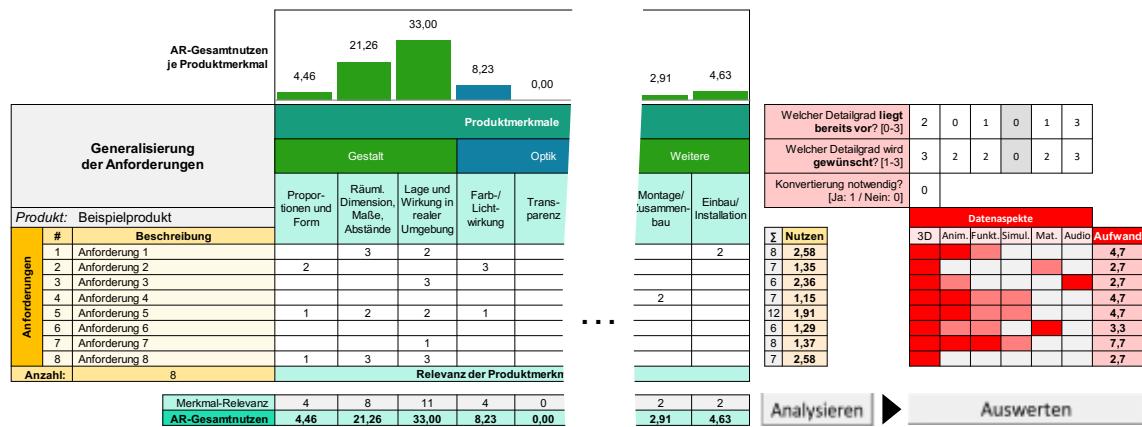


Abbildung 5-10: Ausschnitt der Eingabetabelle für den Anwender zur Bewertung der Relevanz der Produktmerkmale je Anforderung (links) sowie der Datenaspekte (rechts)

Zur Unterstützung der Bewertung dient der **Produktmerkmal-Steckbrief**, der die einzelnen Produktmerkmale genauer beschreibt (Abbildung 5-11). Für die spätere Entscheidungsfindung enthält der Steckbrief zudem Informationen zum AR-Nutzen sowie zur Notwendigkeit der Datenaspekte je Produktmerkmal.

Produktmerkmale							
Kategorie	Merkmal	Beschreibung		Nutzen von AR		Relevante Daten	
Gestalt	Proportionen und Form	Proportionen betreffen das Größenverhältnis der Elemente des Produkts untereinander sowie deren grundsätzliche Form und Positionierung zueinander. Eine 3D-Visualisierung ist auch ohne AR sinnvoll.		1,1			Es sind lediglich 3D-Modelle im Detailgrad 2 „Geometrie“ notwendig.
	Räumliche Dimension, Maße und Abstände	Das Merkmal betrifft die Größe des Produkts sowie Maße und Abstände im Bezug zur realen Umgebung. Eine dreidimensionale Einbettung in die reale Umgebung mittels AR bringt daher einen großen Nutzen.		2,7			Es sind lediglich 3D-Modelle im Detailgrad 2 „Geometrie“ notwendig.
	Lage und Wirkung in realer Umgebung	AR bietet sehr gute Möglichkeiten, um das Produkt dreidimensional in seiner realen Umgebung zu positionieren. So können die Wirkung des Produkts, die optische Erscheinung oder auch der Platzbedarf validiert werden.		3,0			Es sind 3D-Modelle notwendig, bei Bedarf mit optischen Details (Detailgrad 3)
Optik	Farb-/Lichtwirkung	Die Farbe und möglicherweise Muster sowie die grundsätzliche optische Wirkung des Produkts sind meist abhängig von der Umgebung. Daher bietet sich eine Validierung mittels AR an. So kann der Benutzer zudem verschiedene Varianten aus freien Perspektiven betrachten.		2,1			Es sind 3D-Modelle mit nachvollziehbaren optischen Details notwendig.
	Transparenz	Transparenz betrifft die Lichtdurchlässigkeit und -undurchlässigkeit des Produkts.					

Abbildung 5-11: Ausschnitt des Produktmerkmal-Steckbriefs (vgl. Anhang A3)

Basierend auf der Relevanz-Bewertung werden automatisch erste Berechnungen durchgeführt und entsprechende Attributwerte in der Eingabetabelle dargestellt. Für jedes Produktmerkmal wird die AR-Gesamtnutzen GN_M berechnet und ausgegeben sowie grafisch dargestellt (Abbildung 5-10, oben). Ergebnis der Generalisierung sind jeweils die **Nutzwerte N_A** für die einzelnen Anforderungen. Sie werden entsprechend ausgegeben.

5.2.3.2 Bestimmung der notwendigen Aufwände

Nach der Generalisierung der Anforderungen werden in der zweiten Phase die notwendigen Aufwände bestimmt. Zunächst wird dazu über den Button „Analysieren“ ein Makro gestartet, das basierend auf der Relevanz-Bewertung analysiert, welche Datenaspekte notwendig sind für die Validierung. Das Ergebnis wird rechts in der Eingabetabelle (Abbildung 5-10) in Matrixform dargestellt. Dunkelrote Zellen zeigen an, dass der Datenaspekt in der Spalte für die Validierung der Anforderung in der Zeile notwendig ist. Hellrote Zellen bedeuten, dass der Datenaspekt hilfreich sein kann, aber nicht notwendig ist. Im Eingabebereich über dieser Matrix kann der Anwender für die notwendigen Datenaspekte angeben, in welchem Detailgrad sie bereits vorliegen (VDG_D) und welcher Detailgrad gefordert ist (GDG_D). Datenaspekte, die für keine Anforderung notwendig sind, werden ausgegraut. Für 3D-Modelldaten wird zudem angegeben, ob eine Konvertierung in ein AR-fähiges Format notwendig ist. Ist dies der Fall, erhöht sich das Aufwandsdelta um 1.

Zur Unterstützung der Bewertung wird der **Datenaspekt-Steckbrief** bereitgestellt (Abbildung 5-12). Im Steckbrief werden die einzelnen Datenaspekte und die jeweiligen Detailgrade im Detail beschrieben. Durch den Button „Auswerten“ wird ein weiteres Makro gestartet und die **Aufwandswerte Ad** je Anforderung berechnet und ganz rechts in der Eingabetabelle ausgegeben.

Datenaspekte					
Aspekt	Beschreibung	Detailgrad 1	Detailgrad 2	Detailgrad 3	Pauschaler Aufwand
3D-Modelldaten	3D-Modelldaten stellen die Grundlage der AR-Visualisierung dar. Sie werden während der Entwicklung als CAD-Koordinatendaten erstellt. Für die Nutzung mittels AR müssen sie je nach eingesetzten CAD-Tool in AR-fähige Formate (bspw. FBX) überführt werden. Alternativ können 3D-Visualisierungsmodelle mittels verschiedener Modellierungs-Tools erstellt werden. Je nach Anwendung können verschiedene Detailgrade notwendig sein.	Große Form – Für eine erste Abschätzung der Größe und der Proportionen reichen oft grobe Modelle. Sie können aufwandsarm erstellt werden.	Genaue Geometrie – Genaue Geometrien sind meist in den CAD-Modellen vorhanden. Sie können mit gewissen Aufbereitungsaufwänden verwendet werden.	Texturierte Detail-Modelle – Für die Validierung optischer Aspekte sind realitätsnahe 3D-Modelle notwendig. Sie bedürfen aufwendiger Vorbereitungen.	2,7
Animationen	Animationen sind eine sehr gute Möglichkeit, um Funktionalitäten, Interaktionsmöglichkeiten und vor allem komplexe Sachverhalte zu vermitteln und zu veranschaulichen. Mit Animationen können Produkte und deren Vorgänge dynamisch dargestellt und vom Kunden besser verstanden werden. Animationen lassen sich in Modellierungs- oder teilweise auch CAD-Tools erstellen. Alternativ können Animationen durch Software-Funktionalitäten	Einfache Hinweise – Die Einblendung einfacher Hinweise oder die Hervorhebung von Komponenten.	Transformationen – Durch Verschiebung und Rotation einzelner virtueller Elemente	Komplexe Animationen – Komplexe Animationen können	

Abbildung 5-12: Ausschnitt des Datenaspekt-Steckbriefs (vgl. Anhang A3)

5.2.3.3 Auswahl vielversprechender Anforderungen

RÖLTGEN beschreibt im Rahmen seiner Dissertation mehrfach die Vorteile einer grafischen Gegenüberstellung von Bewertungskriterien in einem Portfolio (Abschnitte 4.1.3 und 4.2.2.2). Daher wird zur besseren Auswertung der Bewertungsergebnisse im interaktiven Hilfsmittel ebenfalls ein Portfolio eingesetzt. Basierend auf den zuvor berechneten jeweiligen Nutzen- und Aufwandswerten werden die Anforderungen in einem **Portfolio** dargestellt (Abbildung 5-13). Der Nutzen kann zwischen 0 und 3 liegen. Die Aufwände

liegen zwischen 0 und 19, wobei Aufwände über 12 pauschal als Maximum im Portfolio gewertet werden. Im Portfolio werden drei Bereiche unterschieden. Anforderungen, die ein gutes **Nutzen-Aufwands-Verhältnis** haben, werden als *vielversprechend* für die Validierung mittels AR eingestuft. Anforderungen mit einem eher hohen Aufwand bei gleichzeitig geringem Nutzen, sind *eher nicht empfehlenswert* für den Einsatz von AR. Dazwischen liegen Anforderungen, für die AR als Validierungswerkzeug *abzuwägen* ist. Eine pauschale Empfehlung kann hier nicht gegeben werden. Die Definition der Bereiche des Portfolios sowie die Achsen-Skalierung haben sich durch eine exemplarische Anwendung und die Abstimmung mit den Experten als sinnvoll ergeben.

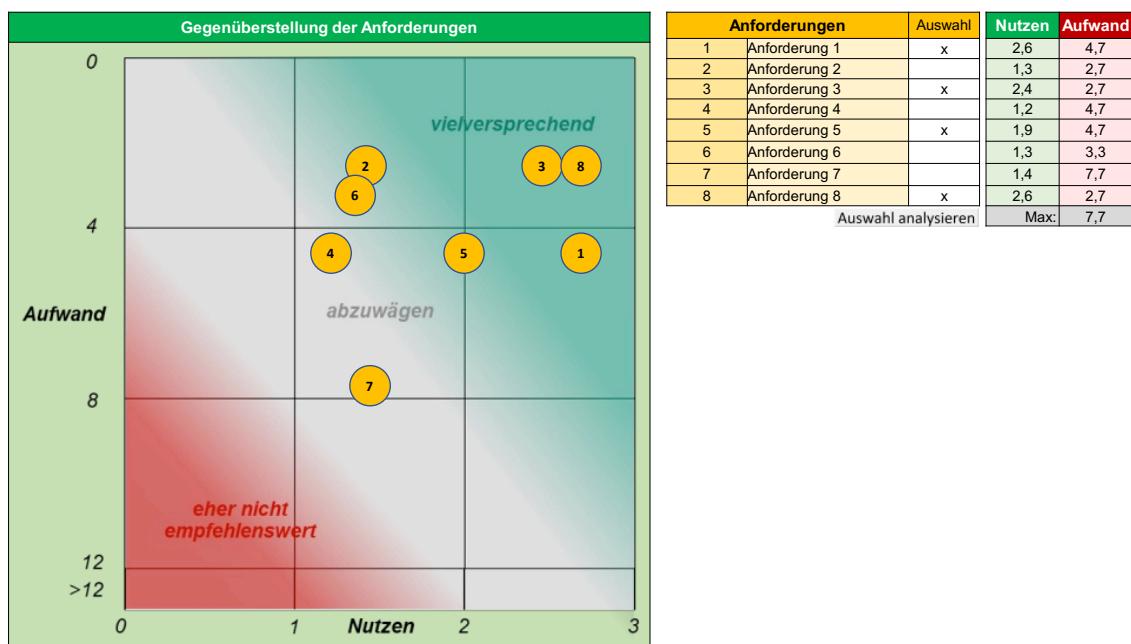


Abbildung 5-13: Portfolio-Darstellung der Anforderungen entsprechend ihrer Nutzen- und Aufwandswerte

Ziel der Systematik ist keine pauschale Entscheidung für oder gegen den Einsatz von AR für alle Anforderungen. Ergebnis kann auch sein, dass AR nur für einen Teil der Anforderungen eingesetzt wird. Basierend auf der Portfolio-Darstellung kann entsprechend eine **Auswahl von Anforderungen** zur weiteren Analyse erfolgen. Die Auswahl erfolgt durch Setzen eines „X“ in der Spalte neben den Anforderungen. Über den Button „Auswahl analysieren“ (s. Abbildung 5-13) wird eine detaillierte Analyse gestartet, um die Potentiale von AR für die ausgewählten Anforderungen zu untersuchen und zu bewerten.

5.2.3.4 Auswertung und Entscheidungsfindung

Im dargestellten Beispiel wurden die Anforderungen 1, 3, 5 und 8 für eine detailliertere Analyse ausgewählt. Das Ergebnis der detaillierten Analyse ist eine Übersicht über die notwendigen Aufwände bzgl. der Datenaspekte. Dabei wird dem Anwender je Datenaspekt dargestellt, welcher Detailgrad gefordert ist, in welchem Detailgrad der Datenaspekt

bereits vorliegt und welche konkreten Aufwände sich daraus basierend auf den pauschalen Aufwands-Faktoren AF_D ergeben (Abbildung 5-14). Zudem wird die **Summe der Aufwände** gebildet. Dabei werden die Aufwände zur Erstellung von 3D-Modelldaten nur einmal eingerechnet, da 3D-Modelldaten nur einmalig erstellt werden müssen und zur Repräsentation verschiedener Merkmale bzw. Anforderungen verwendet werden können. Die Nutzensumme dagegen ergibt sich durch einfache Summierung der Nutzenwerte je Anforderung.

Notwendige Aufwände bzgl. der Datenaspekte				Konvertierung		Aufwand
Detailgrad:	1	2	3	CAD -> Polygon		
3D-Modelldaten	Große Form	Genaue Geometrie	Textur, Detail-Modell			2,6666667
Animationen	Einfache Hinweise	Transformationen	Komplexe Anim.			4
Funktionalitäten	Vereinfacht	Partiell definiert	Exakt			0
Simulationsdaten	Angenähert	Einfach	Detailliert			0
Materialien	Materialprobe	Form-Prototypen	Material in Form			0
Audio/Sound	Einfache Töne	Samples	Reale Aufnahmen			0
Summe Nutzen:			9,40	Nutzen-Aufwands-Verhältnis:		1,4
Summe Aufwände:			6,7			

Abbildung 5-14: Ausgabe der Aufwände sowie des Nutzen-Aufwands-Verhältnisses

Das **Nutzen-Aufwands-Verhältnis** entspricht dem Quotienten der Nutzensumme und der Aufwandssumme. Das Verhältnis ist der grundlegende Indikator, an dem die Entscheidung bzgl. AR als Validierungswerkzeug festgemacht werden kann. Ist das Verhältnis größer als 1, überwiegt der Nutzen den Aufwänden. Entsprechend ist der Einsatz von AR vielversprechend. Umso größer der Wert ist, umso sinnvoller wird der Einsatz von AR eingeschätzt. Werte kleiner als 1 bedeuten, die Aufwände übersteigen den Nutzen. Entsprechend wird AR als Validierungswerkzeug in diesen Fällen nicht empfohlen.

Zusätzlich zum auf die Anforderungen bezogenen Nutzen-Aufwands-Verhältnis werden die AR-Potentiale genauer untersucht und bewertet. Zunächst wird der konkret auf die ausgewählten Anforderungen bezogene **Potential-Nutzen PN_P** je AR-Potential berechnet. In Form von Balkendiagrammen werden diese Werte den entsprechenden **technischen Aufwänden TA_P** für die technische Realisierung gegenübergestellt (Abbildung 5-15). Die Potentialnutzen-Werte geben den Nutzen der einzelnen AR-Potentiale zur Validierung der vorliegenden ausgewählten Anforderungen an. Die Darstellung ermöglicht zudem eine **Gegenüberstellung von AR und den alternativen digitalen Ansätzen**. Ist der Potentialnutzen des AR-Potentials *Kombination mit realer Umgebung* als einzigem Alleinstellungsmerkmal von AR hoch oder sogar höher als die der anderen AR-Potentiale, ist dies ein gutes Indiz dafür, dass AR sinnvoll ist als Validierungswerkzeug. Überwiegt dagegen der Potentialnutzen der Potentiale, die auch mit einer klassischen digitalen 3D-Darstellung möglich sind, ist der Einsatz von AR eher nicht sinnvoll.

Zur Unterstützung der Auswertung der Darstellung wird der **AR-Potential-Steckbrief** bereitgestellt (Abbildung 5-16). Dieser gibt neben einer allgemeinen Beschreibung der Potentiale auch jeweils die notwendigen Aufwände an. Diese Aufwände sollten bei der Auswertung und Entscheidungsfindung berücksichtigt werden. Für die Potentiale, die für AR oder AR und VR gelten, sind die Aufwände eher gering. Dies ist der Fall, da der

Nutzen grundsätzlich aus der Art der Darstellung resultiert und keine speziellen Vorbereitungen benötigt. Weitere Potentiale basieren eher auf einer speziellen Vorbereitung der Daten oder benötigen spezielle Software-Funktionalitäten. Die Darstellung des Potentialnutzen und der entsprechenden Aufwände unterstützt entsprechend die Entscheidung für oder gegen AR als Validierungswerkzeug.

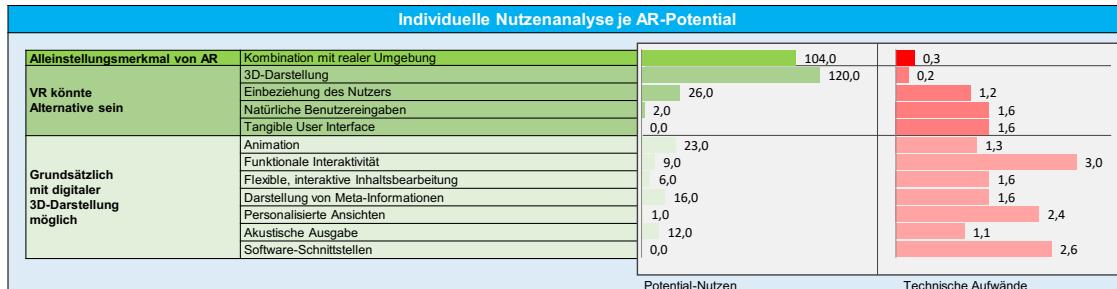


Abbildung 5-15: Gegenüberstellung des Nutzens und der technischen Aufwände

AR-Potentiale				
Kategorie	Potential	Technik	Beschreibung	Aufwände
Darstellung	Kombination mit realer Umgebung	AR-spezifisch	Mittels AR können virtuelle Prototypen in die reale Umgebung eingebettet und mit ihr kombiniert werden. Dies ermöglicht eine Validierung im realen Kontext. Dabei können Räumlichkeiten relevant sein oder auch die Einbindung vorhandener physischer Teilsysteme oder haptischer Elemente.	0,3
	Dreidimensionale Darstellung	AR oder VR	AR ermöglicht eine interaktive dreidimensionale Darstellung digitaler Daten und Modelle. Benutzer können sich frei um die Modelle herum bewegen. Bei einer technischen Umsetzung mit binokularen AR-Brillen werden die Inhalte zudem dreidimensional und immersiv vom Benutzer wahrgenommen.	0,2
	Animation der Inhalte	Digitale 3D-Visualisierung ausreichend	Die dargestellten Inhalte können durch Animationen angereichert werden. So können Vorgänge oder komplexe Sachverhalte anschaulich dargestellt und veranschaulicht werden.	1,3
	Funktionale Interaktivität		In AR-Anwendungen können die originalen Funktionalitäten integriert oder angebunden werden. Beispielsweise können Software-Algorithmen in visuellen virtuellen Prototypen kombiniert werden und die Funktionalität realitätsnah abbilden und entsprechend darstellen.	3,0
	Flexible Inhaltsbearbeitung		Dargestellte Inhalte können flexibel ausgetauscht, verändert oder ausgetauscht werden. So lassen sich beispielsweise aufwandsarm verschiedene Varianten validieren oder Einblicke ins Innere eines Produkts ermöglichen.	1,6
	Darstellung von Meta-Informationen		Meta-Informationen können als zusätzliche Visualisierungen direkt am 3D-Modell erscheinen oder auch eine Preisangabe etc. sein, die für die Validierung an den Produkten für einzelne Personen relevant sind.	

Abbildung 5-16: Ausschnitt des AR-Potential-Steckbriefs (vgl. Anhang A3)

Um den Anwender bei seiner Entscheidung zusätzlich zu unterstützen, wurden sogenannte **Referenzprofile (RP)** erarbeitet (Abbildung 5-17). Referenzprofile ähneln den LIEBRECHT ET AL. beschriebenen Grundtypen von Industrie 4.0 im Mittelstand [LKS+19]. Sie entsprechen generischen Ausprägungen der Potentialnutzen-Werte, die auf eine spezielle Validierungssituation hindeuten. Entspricht das Verhältnis der individuell vorliegenden Potentialnutzen-Werte ungefähr dem eines Referenzprofils, kann davon ausgegangen werden, dass eine ähnliche Validierungssituation vorliegt. Da die Potentialnutzen-Werte abhängig von der Anzahl der Anforderungen unterschiedlich skaliert sind, ist das Verhältnis der Werte ausschlaggebend und nicht die absoluten Zahlen. Für jedes Referenzprofil werden eine Beschreibung der vorliegenden Situation und eine Empfehlung bzgl. des Einsatzes von AR als Validierungswerkzeug gegeben. Im Rahmen der vorliegenden Arbeit wurden basierend auf Erfahrungswerten und Technologie-Expertise bereits fünf Referenzprofile definiert:

- **RP1 - Optisches Erscheinungsbild im Raum:** Entsprächen die Potentialnutzen-Werte ungefähr dem RP1, geht es sehr wahrscheinlich um die Validierung des optischen Erscheinungsbilds des Produkts in einer realen Umgebung. Entsprechend ist

eine dreidimensionale im Raum eingebettete Darstellung von hoher Bedeutung. Animationen und eine flexible Inhaltsgestaltung können die Validierung unterstützen. Für dieses Referenzprofil wird der Einsatz von **AR empfohlen**, da andere Ansätze keine Validierung direkt in der realen Umgebung ermöglichen.

- **RP2 - Ergonomie-Test im Kontext realer Räumlichkeiten:** Ist die dreidimensionale Darstellung im realen Raum unter Einbeziehung des Nutzers von größter Bedeutung, geht es um die Validierung ergonomischer Aspekte, die im Kontext der Räumlichkeit zu betrachten sind. Die Integration natürlicher Interaktionsformen kann dies unterstützen. Aufgrund des Bezugs zur realen Umgebung ist der Einsatz von **AR vielversprechend** und bringt einen Mehrwert. Zur immersiven Einbeziehung des Nutzers ist jedoch eine binokulare **AR-Brille notwendig**.

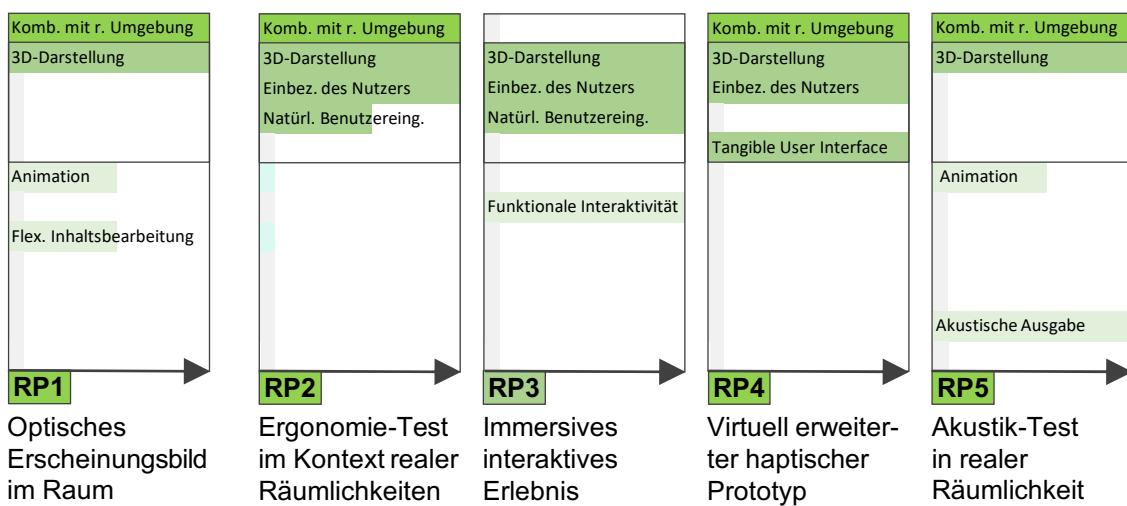


Abbildung 5-17: Referenzprofile (RP) zur Auswertung der Potentialnutzen-Werte

- **RP3 - Immersives interaktives Erlebnis:** RP3 beschreibt das immersive, interaktive Erleben eines Produkts mittels natürlicher Benutzereingaben. Aufgrund der Einbeziehung des Nutzers ist eine immersive Darstellung mittels AR oder VR notwendig. **VR reicht aus** in diesem Fall, da die reale Umgebung keinen Einfluss hat.
- **RP4 - Virtuell erweiterter haptischer Prototyp:** In diesem Profil ist erneut eine dreidimensionale Darstellung im realen Raum unter Einbeziehung des Nutzers von größter Bedeutung. Hochrelevant ist zudem die Integration eines Tangible User Interfaces. Entsprechend handelt es sich um eine Validierung, die anhand eines virtuell erweiterten haptischen Prototypen durchgeführt wird. Ein Teilespekt des Produkts liegt in diesem Fall physisch vor und wird durch eine passgenaue AR-basierte Visualisierung angereichert. Dies kann beispielsweise ein Bedienelement einer vorhandenen Maschine sein, zu dem verschiedene Designvarianten überlagert werden. Der Einsatz von **AR ist erforderlich**.

- **RP5 - Akustik-Test in realer Räumlichkeit:** Neben der dreidimensionalen Darstellung im Raum ist die Ausgabe akustischer Signale von höchster Bedeutung. Entsprechend handelt es sich um die Validierung von akustischen Aspekten in der realen Räumlichkeit. Dies kann bspw. die Überprüfung von Betriebsgeräuschen und deren Wahrnehmbarkeit eines Produkts im Büro oder der Maschinenhalle sein.

Basierend auf dem Nutzen-Aufwands-Verhältnis sowie der Bewertung der AR-Potentiale kann der Anwender eine Entscheidung für oder gegen den Einsatz von AR als Validierungswerkzeug treffen. Dabei kann eine Auswahl von Anforderungen definiert werden, die mittels AR validiert werden sollen. Zudem wird der Anwender bei der Auswahl nutzenstiftender AR-Potentiale unterstützt. Die ausgewählten AR-Potentiale werden im weiteren Verlauf der Planung und Vorbereitung der AR-basierten Validierung bei der AR-System-Konfiguration berücksichtigt sowie im Rahmen der Entwicklung der AR-Anwendungssoftware realisiert.

5.3 Wissensbasierte AR-System-Konfiguration

Für den erfolgreichen Einsatz von AR und eine gezielte Nutzung der Potentiale speziell im Kontext Validierung ist eine bedarfsgerechte Konfiguration des AR-Systems notwendig. Diese Konfiguration umfasst sowohl das AR-Endgerät an sich als auch die zu implementierenden AR-Softwarefunktionalitäten. Aufgrund der Komplexität und Vielfältigkeit der Technologie ist Expertise und Erfahrung notwendig, um fundierte Konfigurationsentscheidungen treffen zu können. Vielen Unternehmen mangelt es am nötigen Wissen. Sie sind daher meist nicht in der Lage, die AR-System-Konfiguration selbstständig durchzuführen. Ziel des zweiten Lösungsbausteins ist daher die **Unterstützung der individuellen Konfiguration bedarfsgerechter AR-Systeme**.

Als **Ausgangslage der AR-System-Konfiguration** liegen die im Rahmen der Nutzen-Aufwands-Analyse zur Nutzung ausgewählten AR-Potentiale vor (s. Kap. 5.2). Zudem sollten individuelle Rahmenbedingungen der Validierung bei der AR-System-Konfiguration berücksichtigt werden. Ziel ist die Auswahl eines geeigneten AR-Endgeräts sowie die Definition der AR-Softwarefunktionalitäten, die in die Validierungssoftware integriert werden sollen. Dazu sind verschiedene Fragestellungen zu beantworten. Welche AR-Endgeräte kommen in der individuell vorliegenden Situation grundsätzlich in Frage? Welche Anwendungssoftware wird benötigt oder kann helfen? Und welche Umfeldfaktoren haben Einfluss auf das AR-System und sind entsprechend zu berücksichtigen oder vorzubereiten? Um Unternehmen bei der Beantwortung dieser Fragen zu unterstützen, soll das relevante Wissen gesammelt, formalisiert und zugänglich gemacht werden. Darauf aufbauend soll eine wissensbasierte AR-System-Konfiguration Unternehmen in die Lage versetzen, eigenständig und ohne umfangreiche Erfahrung und Expertise geeignete AR-Endgeräte auszuwählen und relevante Softwarefunktionalitäten zu definieren.

Für die AR-System-Konfiguration wird über ein **wissensbasiertes System** realisiert, das an der Struktur wissensbasierter Systeme nach KURBEL orientiert ist (Anhang A2.2). Die

Grundlage der AR-System-Konfiguration bildet eine **AR-System-Wissensbasis** (Abbildung 5-18, Abschnitt 5.3.1). In ihr wird das für die Konfiguration relevante Wissen zusammengeführt und aufbereitet. Die Analyse von Ansätzen zur Repräsentation und Verarbeitung von Wissen im Stand der Technik hat gezeigt, dass semantische Technologien vielfältige Möglichkeiten und Vorteile gegenüber klassischen Modellierungsansätzen bieten (s. Abschnitt 4.2.1.2). Dies wurde unter anderem von BAUER anhand seines Planungswerkzeugs zur wissensbasierten Produktionssystemplanung bestätigt (Abschnitt 4.2.2.6). Daher wird die Wissensbasis basierend auf der Web Ontology Language (OWL) realisiert. Die Repräsentation des Wissens mittels OWL als semantischer Technologie ermöglicht eine digitale Verarbeitung und somit halbautomatisierte Auswertung durch Softwarealgorithmen.

Diese Verarbeitung und Auswertung findet über das **interaktive Konfigurationswerkzeug** statt (Abschnitt 5.3.2). Das Werkzeug dient als Schnittstelle zwischen Anwender und AR-System-Wissensbasis und macht das Wissen zugänglich. Der Anwender kann zu nutzende AR-Potentiale sowie Rahmenbedingungen als Input geben. Darauf basierend erfolgt ein gezielter Zugriff sowie eine entsprechende Auswertung des Wissens in der Wissensbasis. Als Ergebnis werden in Frage kommende AR-Endgeräte identifiziert und zu implementierende Softwarefunktionalitäten definiert und die Ergebnisse entsprechend für den Anwender ausgegeben.



Abbildung 5-18: Übersicht der wissensbasierten AR-System-Konfiguration

5.3.1 AR-System-Wissensbasis

Wie die Problemanalyse sowie der Stand der Technik zeigen, sind verschiedene Wissensaspekte für die Konfiguration von AR-Systemen relevant. Dazu zählt zunächst technisches und funktionales Wissen über AR-Systeme sowie über Einflüsse aus dem Umfeld. Zudem ist Anwendungs- und Erfahrungswissen aufzubereiten, beispielsweise zu Anforderungen an das AR-System für verschiedene Anwendungsfälle oder zu Abhängigkeiten zwischen AR-System und Umfeld. Für die bedarfsgerechte Auswahl eines AR-Endgeräts

ist zudem Wissen über am Markt verfügbare Geräte, deren Technik sowie weiteren Eigenschaften relevant. Dieses Wissen ist bisher verteilt auf viele verschiedene Quellen verfügbar. Dabei wird das Wissen meist in natürlicher Sprache und damit in einer Mensch-orientierten Form repräsentiert. Zudem liegt Erfahrungs- und Anwendungswissen vor allem in den Köpfen von Experten vor. Um dieses gesamte Wissen zugänglich und für die AR-System-Konfiguration nutzbar zu machen, wurde die AR-System-Wissensbasis konzipiert und realisiert. Vorarbeiten zur AR-System-Wissensbasis erfolgten im Rahmen einer unveröffentlichten Masterarbeit, betreut durch den Autor der vorliegenden Arbeit [ME20]. Die Ergebnisse der Masterarbeit wurden überarbeitet und erweitert und auf das Anwendungsszenario der vorliegenden Arbeit übertragen. Daraus resultierende Teilergebnisse dieser Arbeit wurden bereits in einer gemeinsamen wissenschaftlichen Veröffentlichung publiziert (siehe *Liste der veröffentlichten Teilergebnisse*). Die Erarbeitung der Wissensbasis erfolgt angelehnt an die Vorgehensweise zur Erstellung von Wissensmodellen nach KOHN (Anhang A2.2.2) in drei Schritten (Abbildung 5-19).

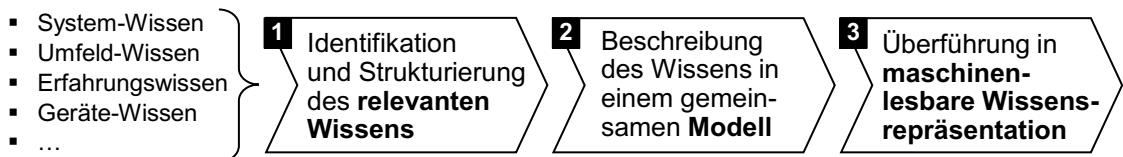


Abbildung 5-19: Die drei Schritte zur Erarbeitung der AR-System-Wissensbasis

Im ersten Schritt wird das relevante Wissen zunächst identifiziert, gesammelt und strukturiert (Abschnitt 5.3.1.1). Anschließend wird ein Modell erarbeitet, mit dem sich das gesamte Wissen inklusive aller Zusammenhänge beschreiben lässt (Abschnitt 5.3.1.2). Das resultierende Wissens-Modell wird basierend auf semantischen Technologien in eine maschinenlesbare Repräsentationsform überführt (Abschnitt 5.3.1.3). Letztendlich liegt das Wissen in einer *.owl*-Datei (Web Ontology Language) vor und kann somit digital verarbeitet und mittels Software zugänglich gemacht werden. Im Folgenden werden die drei Schritte zur Erstellung der AR-System-Wissensbasis im Detail beschrieben.

5.3.1.1 Relevantes Wissen

Zunächst wurde das relevante Wissen aus einer Vielzahl an Quellen gesammelt und strukturiert. Einige dieser Quellen werden im Anhang A2.2 beschrieben. Grundlage der Konfiguration ist das AR-System an sich inklusive seiner funktionalen und technischen Komponenten (vgl. Kapitel 3.4.1 sowie Anhang A2.2.4). Zudem existiert eine Vielzahl spezifischer AR-Endgeräte, aus denen bedarfsgerecht eine Auswahl getroffen werden muss (Anhang A2.2.5). Neben Quellen, in denen die Technologie AR sehr allgemein beschrieben wird, bspw. von DÖRNER ET AL. [DBG+19] und BILLINGHURST ET AL. [BCL15], wurde auch Wissen aus sehr technischen und detaillierten Veröffentlichungen wie der VON LEE ET AL. über Displays [LZW19] berücksichtigt. Neben Wissen über Einflüsse aus dem Anwendungskontext, das unter anderem von RÖLTGEN (Abschnitt 4.2.2.2), PAELKE ET AL. (4.2.2.4) und SCHILLING (Abschnitt 4.2.2.5) beschrieben wird, wurden Berichte

und Publikationen über die Anwendung von AR in verschiedenen Anwendungsszenarien untersucht (Abschnitt 4.3.3 und Anhang A2.2.6). Bei der Strukturierung des Wissens wird zunächst unterschieden zwischen *System*, *Umfeld*, *Software* sowie konkreten *Endgeräten*. Zwischen diesen Aspekten werden anschließend *Wechselwirkungen* definiert.

Das System- und Umfeld-Wissen ist grundsätzlich eher statisch und kann gut abgegrenzt werden. Allerdings kann es beliebig detailliert definiert werden. Daher galt es, eine geeignete Detaillierungstiefe zu finden. Zum einen muss eine ausreichende Varianz zur Unterscheidung verschiedener Geräte, Anwendungssituationen und Rahmenbedingungen gewährleistet werden. Zum anderen darf die Komplexität der AR-System-Konfiguration nicht unnötig erhöht werden, um die Praxistauglichkeit und Anwendbarkeit weiterhin sicherzustellen. Das Wissen über AR-Endgeräte sowie erfahrungsbasiertes und anwendungsbezogenes Wissen sind dagegen nicht endlich und sehr dynamisch. Es kommen stetig neue Geräte auf den Markt und neue Erfahrungen werden mit AR gemacht. Daher wird dieses Wissen zunächst exemplarisch betrachtet. Die Wissensbasis ist jedoch so gestaltet, dass neues und zusätzliches Wissen aufwandsarm integriert und die Wissensbasis somit kontinuierlich erweitert und aktualisiert werden kann (s. Abschnitt 5.3.1.3).

System

Ein AR-System setzt sich aus verschiedenen technischen und funktionalen Komponenten zusammen. Bestehende Beschreibungen von AR-System-Strukturen behandeln meist Systeme für einen konkreten Anwendungsfall (s. Anhang A2.2.4). Eine generische Beschreibung fehlt bisher. So wurden basierend auf den bestehenden spezifischen System-Strukturen sowie weiterem über AR-Systeme bekanntem und veröffentlichtem Wissen mögliche Komponenten eines AR-Systems definiert. Jede Komponente kann durch verschiedene technische Ausprägungen realisiert werden. Die Komponenten und ihre Ausprägungen erfüllen die Funktion eines morphologischen Kastens zur Spezifikation von AR-Endgeräten. Jedes AR-Endgerät lässt sich als eine Kombination von technischen Ausprägungen der einzelnen Komponenten beschreiben. Morphologische Kästen wurden auch von HASSAN für Fahrsimulatoren (Abschnitt 4.2.2.3) sowie von SCHILLING für AR-Systeme (Abschnitt 4.2.2.5) eingesetzt. Die von SCHILLING als morphologischer Kasten verwendete Variantenmatrix ist jedoch nur sehr grob definiert und zudem aufgrund des technologischen Fortschritts nicht mehr aktuell. Sie kann daher für die vorliegende Arbeit nicht verwendet werden. So wurden die Komponenten und ihre Ausprägungen entsprechend dem aktuellen Stand der Technik neu definiert (Abbildung 5-20).

Bei den Komponenten wird unterschieden zwischen essenziellen und optionalen Komponenten. Zu den essenziellen Komponenten muss mindestens eine Ausprägung im System enthalten sein. So hat jedes AR-System ein *Anzeigesystem*, beispielsweise *Optical See-Through* (OST), und ein *Interaktionssystem*, das unter anderem über *Gesten* realisiert werden kann. Essenziell sind zudem ein *Datenspeicher*, ein *Prozessor* und eine *Stromversorgung*, die beispielsweise per *Kabel* erfolgt. Optionale Komponenten sind unter anderem eine *Netzwerk-* und *Daten-Verbindung* sowie *Sensorik* oder auch *Audio-*

Equipment. So kann ein AR-System über ein *Mikrofon* oder *Lautsprecher* verfügen. Dies ist aber grundsätzlich nicht erforderlich. Zudem kann ein AR-System Zukaufkomponenten beinhalten. Einige AR-Systeme benötigen beispielsweise ein *Smartphone*, auf dem die Anwendungssoftware läuft. Insgesamt wurden für die 16 Komponenten 96 technische Ausprägungen definiert.

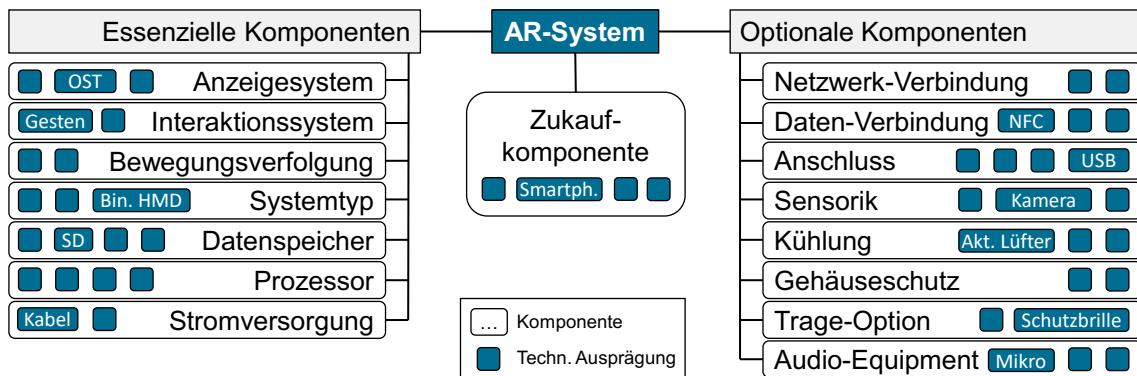


Abbildung 5-20: Komponenten eines AR-Systems mit Beispielen der jeweils möglichen technischen Ausprägungen

Umfeld

Abbildung 5-21 zeigt die identifizierten Umfeldfaktoren, die Einfluss auf die AR-Systemkonfiguration haben können oder für einzelne Systemkomponenten zu berücksichtigen sind. Dazu wurde neben der Definition von Einflussfaktoren nach SCHILLING und dem Leitfragenkatalog von RÖLTGEN auf einer Vielzahl weiterer unter anderem in Anhang A2.2.7 genannter fachspezifischer und technischer Quellen aufgebaut. Die Umfeldfaktoren können zu drei wesentlichen Aspekten gruppiert werden: der Einsatzumgebung, der Tätigkeit, zu der auch der Nutzer zählt, sowie der Umwelt. Für jeden Umfeldfaktor wurden mögliche Ausprägungen definiert. Beim Thema *Arbeitsschutz* kann bspw. eine *Helmpflicht* bestehen oder der *Nutzer* kann *Brillenträger* sein. Beides gilt es bei der System-Konfiguration zu berücksichtigen, da das Tragen einer AR-Brille beeinflusst wird. Umweltbedingungen wie die *Lautstärke* oder die *Lichtverhältnisse* sind ebenso zu berücksichtigen wie der *Einsatzort* oder die *Arbeitsumgebung* als Aspekte der Einsatzumgebung. Bzgl. der *IT-Infrastruktur* kann bspw. relevant sein, bspw. ob eine *Cloud*-Anbindung möglich ist oder ein *drahtloses Netzwerk* vorhanden ist. Für die 16 Umfeldfaktoren wurden insgesamt 69 Ausprägungen identifiziert. Diese decken nach dem aktuellen Wissensstand alle für die AR-System-Konfiguration relevanten Umfeld-Aspekte ab.

Software

Die in Abschnitt 5.2.2 definierten *AR-Potentiale* werden über Anwendungssoftware realisiert, die später auf dem AR-Endgerät läuft. Dabei basiert die Anwendungssoftware auf verschiedenen Funktionalitäten, die verwendet und miteinander kombiniert werden können. Bisher wurden 28 Softwarefunktionalitäten definiert (s. Abbildung 5-22). Die Softwarefunktionalitäten adressieren verschiedene Aspekte. Es gibt Funktionalitäten für die

Präsentation und Manipulation von virtuellen Inhalten, beispielsweise zum *Ein- und Ausblenden*, *Variantenwechsel* oder *Skalieren*. Im Bereich der Interaktion gibt es unter anderem die Möglichkeit der *Gestensteuerung*, die als Funktionalität integriert werden kann. Zudem kann *Feedback* am virtuellen Prototyp ermöglicht werden. Für kollaborative Anwendungen ist beispielsweise *Audiokommunikation* notwendig.

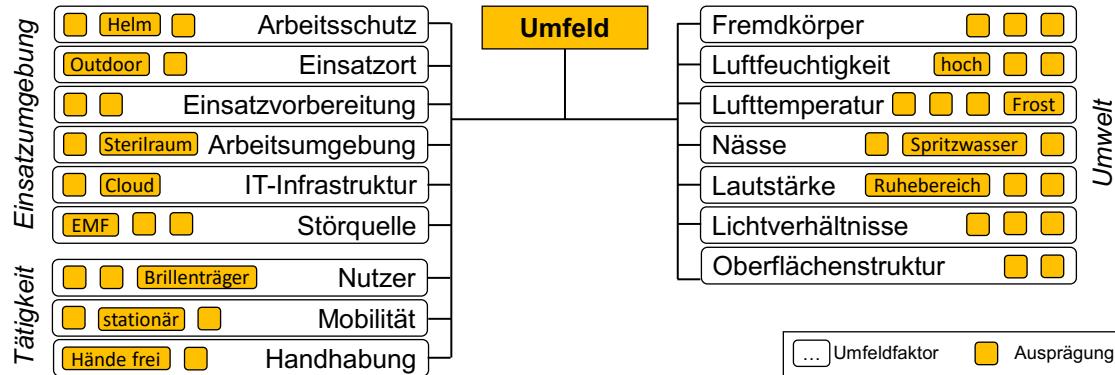


Abbildung 5-21: Übersicht der identifizierten Umfeldfaktoren mit Beispielen der konkreten Ausprägungen

Für die Implementierung von AR-Softwarefunktionalitäten existieren verschiedene Software Development Kits (SDKs). Die SDKs stellen grundlegende Funktionalitäten bereit, auf denen die Anwendungssoftware aufgebaut sein kann. Neben grundsätzlichen AR-SDKs wie dem *MRTK* oder dem *ARKit* ermöglichen weitere SDKs spezifische Funktionalitäten (s. Abschnitt 4.3.1.2). Dazu zählen beispielsweise *Photon* (Kollaboration), *OpenCV* (Bildverarbeitung) oder auch *Vuforia* und *VisionLib* (beide Tracking).

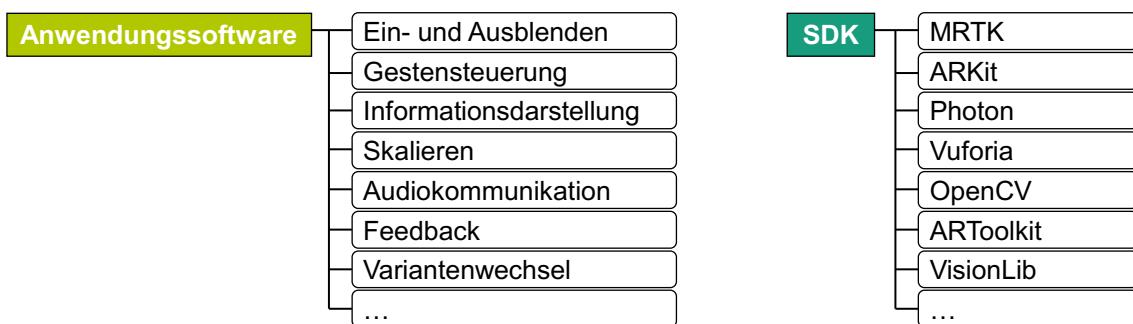


Abbildung 5-22: Auszug aus den Softwarefunktionalitäten für die Anwendungssoftware sowie den SDKs für die Entwicklung von AR-Anwendungen

Endgeräte

Um basierend auf verschiedenen Anforderungen und Rahmenbedingungen ein AR-Endgerät auswählen zu können, muss Wissen über am Markt verfügbare AR-Endgeräte vorhanden sein. So wurde bislang ausgehend vom Stand der Technik Wissen zu 18 Endgeräten erarbeitet und dokumentiert. Als Quellen dienen hier vor allem technische Spezifikationen, Produktpräsentationen und Datenblätter (s. Anhang A2.2.5). Für jedes AR-End-

gerät wird definiert, welche technischen und funktionalen Ausprägungen der Systemkomponenten enthalten sind oder unterstützt werden. Zusätzlich werden Gerät-Attribute wie der *Preis*, das *Gewicht* und weitere technische Details wie die Größe des *Sichtfelds* oder die *Auflösung* der Displays definiert. Zudem ist für die spätere Entwicklung der Anwendungssoftware relevant, welche SDKs ein AR-Endgerät unterstützt oder sogar erfordert.

Wechselwirkungen

Ausschlaggebend für die bedarfsgerechte AR-System-Konfiguration sind vor allem Wechselwirkungen zwischen den oder innerhalb von Aspekten. Wie der Stand der Technik zeigt, sind hier vielfältige Aspekte zu berücksichtigen (Anhang A2.2.7). Wichtigste Wechselwirkungen sind **Erfordernisse**, **Ausschlüsse** und **Beeinflussungen**. Zum einen können einzelne Ausprägungen von Systemkomponenten Wechselwirkungen untereinander aufweisen. Beispielsweise erfordert *GPS-Tracking* einen *GPS-Sensor* und ein *Smartphone* als Systemtyp schließt ein *Optical See-Through*-Anzeigesystem aus. Zum anderen können Abhängigkeiten auch auf Komponentenebene bestehen. Ein *Cloud*-Datenspeicher erfordert beispielsweise eine *Netzwerk*-Anbindung. Entsprechend ist eine der möglichen Ausprägungen dieser Komponente gefordert, welche konkret ist dabei egal. Zudem kann eine Ausprägung aus einer Menge an alternativen Ausprägungen gefordert sein. Es handelt sich entsprechend um eine Oder-Beziehung. Als Beispiel erfordert eine Handhabung mit *beiden Händen frei* ein *binokulares HMD* (Head-Mounted Display) oder ein *monokulares HMD*. Systemkomponenten können auch durch die Anwendungssoftware erforderlich sein. Beispielsweise erfordert *Marker-Tracking* eine *Kamera* als Sensorik. Wechselwirkungen zwischen dem Umfeld und dem System spielen eine besonders große Rolle bei der System-Konfiguration, da das Umfeld sehr vielfältig sein und gleichzeitig großen Einfluss auf das System haben kann. So erfordert ein *mobiler Einsatz* der AR-Lösung beispielsweise eine Stromversorgung per *Batterie* und ein *Outdoor*-Einsatzort schließt ein *kabelgebundenes Netzwerk* aus. Entsprechend kann die Auswahl an möglichen AR-Endgeräten durch Umfeldfaktoren stark eingeschränkt werden. Beeinflussung dagegen schränkt die Auswahl eines AR-Endgeräts nicht direkt ein, sollten aber beim späteren Einsatz berücksichtigt werden. Beispielsweise erschwert eine *hohe Umgebungs-lautstärke* die *Sprachsteuerung* und eine *starke Sonneneinstrahlung* hat Einfluss auf die Wirkungsweise einer *Kamera* und darauf basierenden Funktionalitäten wie dem *Marker-Tracking*.

5.3.1.2 Modellierung des Wissens

Das zuvor beschriebene Wissen wird im zweiten Schritt in einem gemeinsamen Modell zusammengeführt und formalisiert. Wichtig dabei ist, dass das Wissen grundsätzlich dynamisch ist, sich ändern und zukünftig erweitert werden kann. Zudem soll es digital verarbeitet und analysiert werden können. Die Modellierung muss daher entsprechend einer klaren Struktur erfolgen. Zu diesem Zweck wurde die **AR-System-Ontologie** entwickelt (Abbildung 5-23). Eine Ontologie stellt eine formale Definition von Begriffen und deren

Beziehungen für einen bestimmten Anwendungsbereich dar. Die Ontologie definiert die Syntax und Semantik und legt fest, welche Arten von Aussagen möglich und erlaubt sind. Eine Ontologie dient entsprechend als Grundlage für semantische Technologien und ermöglicht eine automatisierte Wissensverarbeitung über Software-Werkzeuge (s. Abschnitt 4.2.1.2). Der Umfang einer Ontologie wird durch den Anwendungskontext und den Nutzen bestimmt. Die Anwendungsdomäne ist in diesem Fall das AR-System. Dementsprechend enthält die Ontologie alle Wissensaspekte, die aus Sicht des Autors für eine bedarfsoorientierte AR-System-Konfiguration relevant sind.

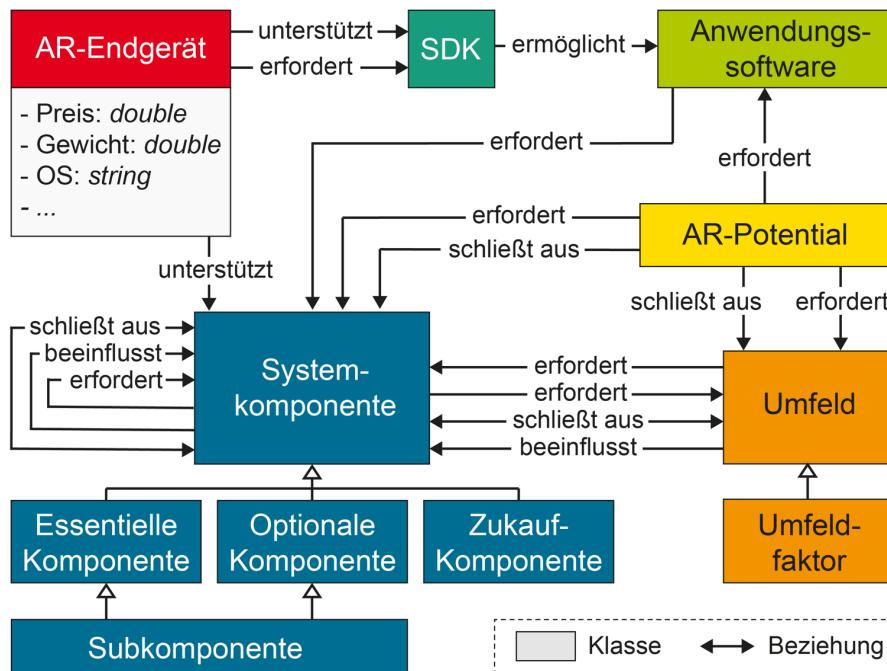


Abbildung 5-23: AR-System-Ontologie als Grundlage der digitalen Wissensbasis

Die AR-System-Ontologie umfasst entsprechend des im vorherigen Abschnitt beschriebenen relevanten Wissens verschiedene Klassen. Neben Systemkomponenten sind das Umfeld, Anwendungssoftware, AR-Potentiale sowie SDKs in der Ontologie erlaubt. Zudem wurde die Klasse AR-Endgerät zur Beschreibung konkreter Produkte integriert. Für jede dieser Klassen können spezifische Ausprägungen als Instanzen definiert werden. Die Klassen stellen entsprechend Sammlungen von Instanzen dar, die dem gleichen Typ entsprechen. Instanzen einer Klasse müssen die gleiche Funktionalität bieten und sich gegenseitig ersetzen können. Dies wurde bereits bei der Strukturierung des Wissens berücksichtigt (vgl. Abschnitt 5.3.1.1). Systemkomponenten werden unterteilt in essentielle, optionale und Zukauf-Komponenten. Essentielle und optionale Komponenten werden weiter untergliedert in Subkomponenten. Zu den Subkomponenten existieren jeweils verschiedene technische Ausprägungen als Instanzen, beispielsweise die Instanz *RGB-Kamera* für die Subkomponente *Sensorik*. Einzelne Umfeldeinflüsse werden über Umfeldfaktoren als Unterklasse der Klasse Umfeld gruppiert. Umfeldfaktoren sind beispielsweise die *IT-Infrastruktur* oder der *Benutzer* (vgl. Abbildung 5-21). Die Klassenhierarchie impliziert auch, dass Instanzen, die einer Unterklasse zugeordnet sind, auch der

Oberklasse zugeordnet werden. Über Beziehungen werden die für die AR-System-Konfiguration relevanten Wechselwirkungen beschrieben. Dabei wird durch die Ontologie genau festgelegt, welche Beziehungen zwischen den Klassen bzw. deren Instanzen bestehen können. Systemkomponenten haben *schließt aus-*, *beeinflusst-* und *erfordert-* Beziehungen untereinander und zu Umfeldfaktoren. Anwendungssoftware kann Systemkomponenten erfordern. Durch die Integration der AR-Potentiale können für sie erforderliche oder ausgeschlossene Systemkomponenten und Umfeldfaktoren beschrieben werden. Wie zuvor beschrieben, erfordert die Nutzung der AR-Potentiale zudem konkrete Anwendungssoftware, die wiederum durch SDKs ermöglicht werden kann. Ein AR-Endgerät kann Systemkomponenten unterstützen bzw. beinhalten. Diese Beziehung dient der technischen und funktionalen Beschreibung konkreter Geräte. Zudem unterstützen AR-Endgeräte einzelne SDKs oder erfordern diese sogar. Weitere Informationen zu konkreten AR-Endgeräten werden über Attribute beschrieben. So hat ein AR-Endgerät unter anderem einen Preis, ein Gewicht oder ein Betriebssystem (Operating System, OS). Diese Attribute können später bei der Auswahl geeigneter Endgeräte verwendet werden.

Die AR-System-Ontologie bildet die Grundlage für ein spezifisches Modell, das das gesamte identifizierte Wissen enthält. Ein spezifisches Modell ist nach SCHADY ein Gesamtmodell eines bestimmten Systems mit allen relevanten Instanzen und Beziehungen [Sch08b]. Das Modell enthält alle technischen Ausprägungen der Systemkomponenten, alle möglichen Ausprägungen der Umgebungsfaktoren sowie die definierten AR-Potentiale und Funktionalitäten der Anwendungssoftware. Zudem ist eine Auswahl gängiger AR-Endgeräte und SDKs enthalten. Außerdem werden die zwischen den Aspekten identifizierten Beziehungen beschrieben. Das konkrete Modell ist aufgrund der großen Anzahl von Instanzen und Beziehungen grafisch nicht darstellbar. Aus diesem Grund wird auf eine Visualisierung verzichtet. Auf Umfang und Inhalt des spezifischen Modells wird im folgenden Abschnitt eingegangen.

5.3.1.3 Maschinenlesbare Wissensrepräsentation

Das basierend auf der AR-System-Ontologie modellierte Wissen wird im nächsten Schritt in eine maschinenlesbare digitale Form überführt. Aufgrund der Empfehlung der W3C als Standardisierungsorganisation für semantische Technologien wurde die **Web Ontology Language (OWL)** als konkrete Beschreibungssprache ausgewählt. Wie im Stand der Technik beschrieben, können Informationen mittels semantischer Technologien so beschrieben werden, dass ihre Semantik und Bedeutung automatisiert von Software erkennbar ist (vgl. Abschnitt 4.2.1.2). Das Wissen wird entsprechend über Software halb-automatisiert zugänglich gemacht. Somit eignen sich semantische Technologien besonders gut für wissensbasierte Systeme wie der vorliegenden AR-System-Konfiguration. Die Repräsentation des Wissens über OWL ermöglicht die Verarbeitung und Analyse des Wissens mittels digitaler Algorithmen und somit die Realisierung des interaktiven Konfigurationswerkzeugs (Abschnitt 5.3.2).

Eine auf der OWL basierende Wissensbasis besteht aus **zwei Komponenten**, der *TBox* und der *ABox* (s. Anhang A2.2.3). Die *TBox* beschreibt die Struktur des abgebildeten Wissens. Dazu zählen Klassen und deren Hierarchie sowie mögliche Beziehungs- und Attributtypen. Zur Beschreibung der Struktur des vorliegenden Wissens werden die folgenden OWL-Elemente verwendet:

- *owl:Class* für die in der Ontologie beschriebenen Klassen
- *owl:SubClass* zur Definition von Generalisierungen (bspw. bzgl. essenzieller und optionaler Systemkomponenten)
- *owl:Datatypes* für Typen von Attributen (Integer, String etc.)
- *owl:DatatypeProperties* für die konkreten Attribute (bspw. den Preis)
- *owl:ObjectProperties* für Typen von Beziehungen (erfordert etc.)

In der *ABox* wird das eigentliche Wissen dokumentiert. Entsprechend enthält die *ABox* konkrete Individuen als Instanzen der Klassen und ihre Klassenzugehörigkeiten, Beziehungen zwischen Individuen sowie konkrete Attributwerte.

Eine Besonderheit stellen sogenannte **Punning-Klassen** dar. Die Web Ontology Language erlaubt lediglich Beziehungen zwischen einzelnen Individuen als 1-zu-1-Beziehungen. Beziehungen zu Klassen als Ganzes und Oder-Beziehungen zu mehreren Elementen sind nicht möglich. Bei *Netzwerk-Verbindung* handelt es sich beispielsweise um eine Klasse, zu der es spezifische Individuen gibt. Die Information, dass ein *Cloud-Datenspeicher* eine *Netzwerk-Verbindung* benötigt, kann entsprechend nicht in der Wissensbasis abgebildet werden. Punning-Klassen sind ein gängiger Ansatz im Kontext semantischer Technologien, um diese Problematik zu umgehen. Für jede Klasse wird ein zusätzliches Individuum erstellt und der Wissensbasis als Instanz einer neuen Meta-Klasse *Punning* hinzugefügt [W3C08-ol]. So können Beziehungen zu den *Punning*-Individuen definiert und durch geeignete Auswertungsmechanismen entsprechende Schlüsse auf die jeweilige Klasse und die ursprüngliche Information gezogen werden.

Für den Import von Wissen in eine Wissensbasis im Rahmen des sogenannten Ontology Learning existieren verschiedene Ansätze (s. Abschnitt 4.2.1.2). So kann Wissen unter anderem aus UML-Modellen direkt eingelesen werden. Allerdings gilt dies nur für Elemente der *TBox*. Das Überführen von Wissen in die *ABox* wird nicht unterstützt. Daher wurde ein neues **Konzept zur aufwandsarmen Überführung des modellierten Wissens** in die *TBox* und die *ABox* einer OWL-basierten Wissensbasis entwickelt (Abbildung 5-24). Das Konzept sieht vor, dass neu gewonnenes Wissen bei Bedarf hinzugefügt oder angepasst werden kann. Bei den dargestellten Aufgaben wird daher unterschieden zwischen einmalig durchzuführenden Aufgaben zur Vorbereitung der Wissensbasis (schwarz) sowie Aufgaben, die jeweils bei Anpassung oder Erweiterung des Wissens durchgeführt werden müssen (rot).

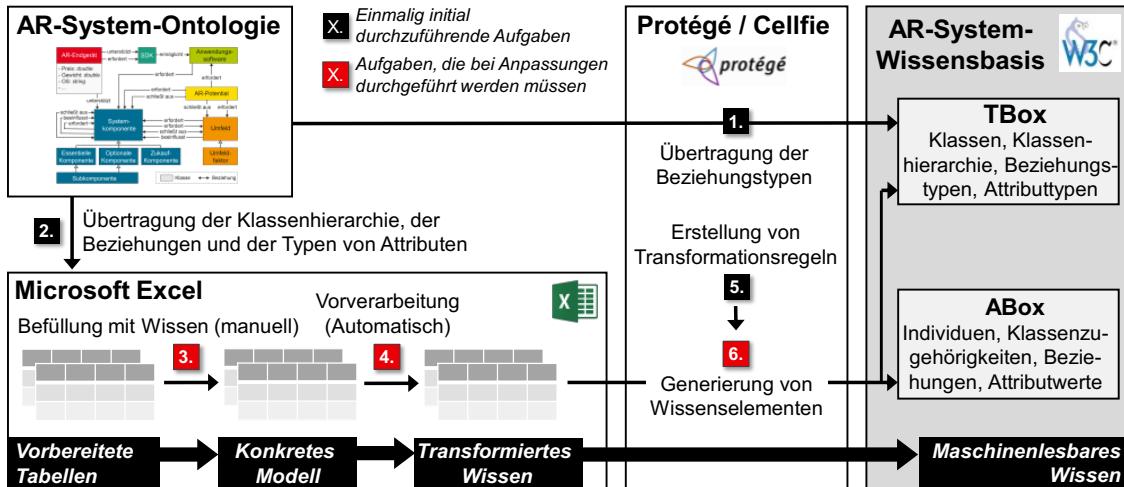


Abbildung 5-24: Überführung des entsprechend der AR-System-Ontologie modellierten Wissens in eine maschinenlesbare Wissensrepräsentation

Ausgangspunkt und Grundlage der AR-System-Wissensbasis stellt die AR-System-Ontologie dar. In **Schritt 1** werden die in der Ontologie enthaltenen Beziehungstypen (*erfordert, schließt aus, hat Einfluss auf, unterstützt, ermöglicht*) in die TBox überführt. Dazu wird das Softwarewerkzeug **Protégé** eingesetzt (s. Anhang A2.2.8). Die *erfordert*-Beziehung wird als transitiv definiert und die *schließt-aus*-Beziehung als symmetrisch. Die eigentliche Abbildung des Wissens zum aufwandsarmen Einlesen in die Wissensbasis erfolgt über **Excel**. Diese Entscheidung wurde getroffen, da Excel-Arbeitsmappen mittels des in Protégé verwendbaren Plugins **Cellfie** automatisiert eingelesen und Wissen in die TBox und die ABox überführt werden kann.

Anlegen des Wissens in Excel-Arbeitsmappen

In **Schritt 2** erfolgt entsprechend eine Übertragung der Klassenhierarchie, der Beziehungen sowie der Typen von Attributen in Excel-Arbeitsmappen. In den resultierenden vorbereiteten **Excel-Arbeitsmappen** erfolgt die manuelle Beschreibung des Wissens als **Schritt 3**. In einer Arbeitsmappe werden die Individuen für die einzelnen Klassen definiert (Abbildung 5-25). Mittels integrierter **Makros**, die über Button in den Arbeitsmappen aktiviert werden können, werden automatisiert weitere Arbeitsmappen für weitere Wissensaspekte generiert. Dabei werden für jede Klasse automatisch auch Individuen der *Punning*-Klasse generiert, um Klassenbeziehungen definieren zu können.

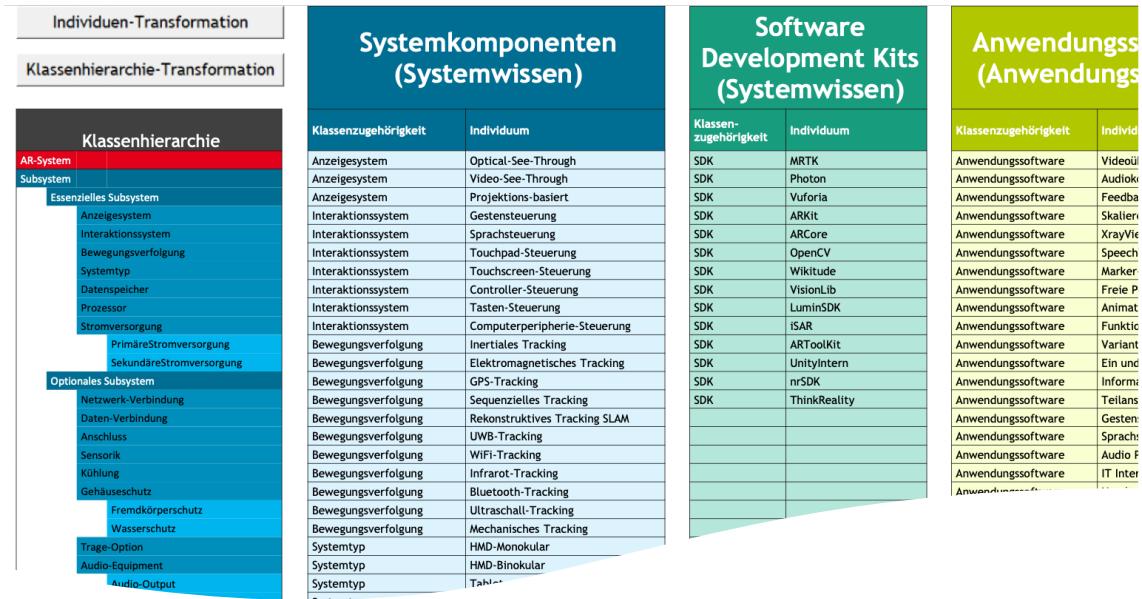


Abbildung 5-25: Ausschnitt der Excel-Arbeitsmappe zur Definition der Klassenhierarchie (links) sowie der Individuen der einzelnen Klassen

Grundsätzlich erfolgt die Definition der Beziehungen über eine Matrix-basierte Gegenüberstellung und Bewertung anhand von Zahlenwerten. Abbildung 5-26 zeigt einen Ausschnitt aus der Arbeitsmappe zur Definition der Beziehungen der einzelnen AR-Potentiale. In den Zellen wird jeweils definiert, ob eine Beziehung zwischen dem AR-Potential in der Spalte und dem Element in der Zeile besteht. Eine 3D-Visualisierung schließt bspw. ein projektions-basiertes Anzeigesystem aus (Wert 1 in Zelle D10). Zudem erfordert die *Kombination mit der realen Umgebung eine Bewegungsverfolgung* (Wert 2 in Zelle E19). Bei dieser Beziehung wird der Punning-Ansatz ausgenutzt und eine Beziehung zum *Punning-Individuum System_Bewegungsverfolgung* definiert. Über den Punning-Ansatz lassen sich zudem Oder-Beziehungen abbilden, die mit OWL nicht direkt abbildbar sind. *Natürliche Benutzereingaben* erfordern *Gestensteuerung* oder *Sprachsteuerung*. Dies kann definiert werden, indem die Punning-Klasse *Interaktionssystem* erfordert ist und gleichzeitig die weiteren Individuen der Klasse (*Touchpad-Steuerung* etc.) ausgeschlossen werden. Bei der späteren Auswertung werden diese Sachverhalte entsprechend berücksichtigt (s. Abschnitt 5.3.2.2). Die Makros für die Vorbereitung der Arbeitsmappen sind so konzipiert, dass bereits definierte Beziehungen auch bei Änderungen an den Individuen, bspw. beim Löschen oder Hinzufügen einzelner Elemente, beibehalten werden und kein Wissen verloren geht. In weiteren Arbeitsmappen wird das vollständige Wissen inklusive der verschiedenen in der Ontologie beschriebenen Beziehungen sowie der Geräte-Spezifikationen definiert. Das Wissen liegt nun als konkretes Modell in Excel vor.

für Potentiale		Potential	DreiD-Visualisierung	Kombination mit realer Umgebung	Animation	Funktionale Interaktivität	Flexible Inhaltsgestaltung	Darstellung von Meta-Infos	Personalisierte Ansichten	Einbeziehung des Benutzers	Natürliche Benutzerergaben	Tangible UI	Akustische Ausgabe	Software Schnittstellen	Integriertes virtuelles Feedback	Digitale Telemetrie	Automatisierte Benutzerführung	Ortsabhängigkeit	Kollaboration
schließtAus	1																		
erfordert	2																		
	3																		
	4																		
	5																		
	6																		
	7																		
	8																		
PunningKlasse	System_Anzeigesystem																		
Anzeigesystem	Optical-See-Through																		
Anzeigesystem	Video-See-Through																		
Anzeigesystem	Projektions-basiert	1																	
PunningKlasse	System_Interaktionssystem																		
Interaktionssystem	Gestensteuerung																		
Interaktionssystem	Sprachsteuerung																		
Interaktionssystem	Touchpad-Steuerung																		
Interaktionssystem	Touchscreen-Steuerung																		
Interaktionssystem	Controller-Steuerung																		
Interaktionssystem	Tasten-Steuerung																		
Interaktionssystem	Computerperipherie-Steuerung																		
PunningKlasse	System_Bewegungsverfolgung	2																	
Bewegungsverfolgung	Inertiales Tracking																		
Bewegungsverfolgung	Elektromagnetisches Tracking																		
Bewegungsverfolgung	GPS-Tracking																		
Bewegungsverfolgung	Sequenzielles Tracking																		
Bewegungsverfolgung	Rekonstruktives Tracking SLAM																		
Bewegungsverfolgung	UWB-Tracking																		
Bewegungsverfolgung	WiFi-Tracking	1																	

Abbildung 5-26: Ausschnitt der Excel-Arbeitsmappe zur Definition der Beziehungen der AR-Potentiale inkl. einer Oder-Beziehung

Transformation des Wissens

In einer OWL-basierten Wissensbasis wird jede **Information als Axiom** integriert. Ein Axiom besteht jeweils aus den drei Bestandteilen Subjekt, Prädikat und Objekt. Subjekt und Objekt werden über das Prädikat in Beziehung gebracht. Dabei kann das Prädikat sowohl einer Beziehung (*Marker-Tracking erfordert Kamera*) entsprechen als auch einer Klassenzugehörigkeit (*USB istIndividuumDerKlasse Anschluss*) oder einem Attribut (*HoloLens2 hatPreis 3.500\$*). Für die Überführung des Wissens mittels Cellfie in die Wissensbasis muss das Wissen in **Schritt 4** daher zunächst vorverarbeitet werden (vgl. Anhang A2.2.8). Basierend auf Makros erfolgt dabei eine Transformation in eine Axiom-Darstellung. Abbildung 5-27 zeigt exemplarisch einen Ausschnitt der bereits in Cellfie geladenen Arbeitsmappe mit dem transformierten Wissen für die Beziehungen *erfordert* und *hatEinflussAuf*. Insgesamt enthält die erarbeitete Excel-Datei **12 Arbeitsmappen** und **17 Makros**.

Einlesen des Wissens in die Wissensbasis

Das transformierte Wissen kann nun mittels Cellfie in die Wissensbasis eingelesen werden. Dazu sind in **Schritt 5** einmalig **Transformationsregeln** in Cellfie anzulegen. Über die Transformationsregeln wird die Struktur einzelner Axiome und die Position der Bestandteile in der Arbeitsmappe definiert. Die in Abbildung 5-27 blau hinterlegte Trans-

formationsregel für die *erfordert*-Beziehungen gibt beispielsweise an, dass sich das Subjekt jeweils in Spalte *J* und das Objekt in Spalte *L* befindet und zwischen ihnen das Prädikat *erfordert* gilt. Entsprechend der zugehörigen jeweiligen Start- und Endzeile wird für jede Zeile ein entsprechendes Axiom mit den Werten aus der Arbeitsmappe erzeugt und der Wissensbasis hinzugefügt. Struktur-relevante Axiome wie Klassenbeziehungen werden der TBox hinzugefügt und Informationen über Individuen und Beziehungen der ABox. So liegt das in Excel beschriebene Wissen nach **Schritt 6** in einer maschinenlesbaren Form in der Wissensbasis vor.

The screenshot shows the Cellfile application interface. At the top, it displays the target ontology: urn:absolute:AR_WB_211217. The main window contains two parts: an Excel spreadsheet and a transformation rules editor.

Excel Spreadsheet (Top):

	I	J	K	L	M	N	O	P
1								
2			erfordert	99			hatEinflussAuf	59
3	Hololens 2		erfordert	MRTK	Umfeld_Fremdkörper	hatEinflussAuf	Mechanisches Tracking	
4	Magic Leap		erfordert	Lumin	Hoheluftfeuchtigkeit	hatEinflussAuf	Ultraschall-Tracking	
5	Gestensteuerung		erfordert	RGB-D-Kamera	Hoheluftfeuchtigkeit	hatEinflussAuf	Ultraschall-Sensor	
6	Gestensteuerung		erfordert	Umfeld_Handhabung	Hohelufttemperatur	hatEinflussAuf	Infrarot-Tracking	
7	Sprachsteuerung		erfordert	System_Audio-Input	Hohelautstärke	hatEinflussAuf	Sprachsteuerung	
8	Sprachsteuerung		erfordert	Ruhebereich	Hohelautstärke	hatEinflussAuf	Mikrofon	
9	Touchpad-Steuerung		erfordert	Umfeld_Handhabung	StarkeBeleuchtung	hatEinflussAuf	Optical-See-Through	
10	Touchscreen-Steuerung		erfordert	Umfeld_Handhabung	StarkeBeleuchtung	hatEinflussAuf	Projektions-basiert	
11	Controller-Steuerung		erfordert	System_Anchluss	StarkeBeleuchtung	hatEinflussAuf	Mechanisches Tracking	
12	Controller-Steuerung		erfordert	Umfeld_Handhabung	StarkeBeleuchtung	hatEinflussAuf	RGB-Kamera	
13	Tasten-Steuerung		erfordert	Umfeld_Handhabung	SchwankendeBeleuchtung	hatEinflussAuf	Optical-See-Through	
14	Computerperipherie-Steuerung		erfordert	System_Anchluss	SchwankendeBeleuchtung	hatEinflussAuf	Projektions-basiert	
15	Computerperipherie-Steuerung		erfordert	Umfeld_Handhabung	SchwankendeBeleuchtung	hatEinflussAuf	Mechanisches Tracking	
16	Inertiales Trakinn		erfordert	Curoskon	SchwankendeBeleuchtung	hatEinflussAuf	Ultraschall-Sensor	

Transformation Rules (Bottom):

Add	Edit	Delete	Load Rules	Save Rules	Save As...		
<input checked="" type="checkbox"/>	Sheet Name	Start Column	End Column	Start Row	End Row	Rule	Comment
<input checked="" type="checkbox"/>	K_Annotatione B	C		3	6	Individual: @B* Annotations: rdfs:comment @C*	
<input checked="" type="checkbox"/>	T_Relationen	J	L	3	278	Individual: @* Facts: erfordert @L*	
<input checked="" type="checkbox"/>	T_Attribute	S	V	3	145	Individual: @T* Facts: @U* @V* (xsd:boolean)	
<input checked="" type="checkbox"/>	T_Individuen	T	U	3	216	Individual: @T* Types: @U*	

Buttons: Generate Axioms

Abbildung 5-27: Einlesen von Excel-Arbeitsmappen (oben) und automatisiertes Überführen der enthaltenen Informationen in eine OWL-Wissensbasis basierend auf Transformationsregeln (unten) in Cellfile

OWL als semantische Technologie ermöglicht zudem die automatisierte Ableitung von implizit vorhandenem Wissen (vgl. Abschnitt 4.2.1.2). In Protégé sind dazu bereits verschiedene **Reasoner** integriert. Reasoner werten Beziehungsketten innerhalb einer Wissensbasis aus. In der AR-System-Wissensbasis betrifft dies zum einen die zuvor als transitiv definierten *erfordert*-Beziehungen. Wenn ein Individuum *X* ein Individuum *Y* erfordert und *Y* zudem Individuum *Z* erfordert, bedeutet dies, dass *X* ebenfalls *Z* erfordert. Zum anderen betrifft es die symmetrischen *schließt-aus*-Beziehungen. Wenn *X* *Y* ausschließt, schließt *Y* umgekehrt auch *X* aus. So kann Wissen, das implizit im Modell vorhanden ist, automatisch abgeleitet und der Wissensbasis explizit hinzugefügt werden.

Nach der Durchführung aller Schritte liegt die AR-System-Wissensbasis in maschinenlesbarer Form als eine **.owl-Datei** vor. Insgesamt enthält die im Rahmen der Arbeit erarbeitete Wissensbasis bereits **1656 Axiome, 48 Klassen und 283 Individuen**.

5.3.2 Interaktives Konfigurationswerkzeug für AR-Systeme

Auf der AR-System-Wissensbasis setzt das interaktive Konfigurationswerkzeug auf. Es dient als Schnittstelle zwischen der Wissensbasis und dem Anwender und macht das Wissen zugänglich. Input für das Konfigurationswerkzeug sind die durch den Anwender ausgewählten zu nutzenden AR-Potentiale sowie individuelle Rahmenbedingungen und Anforderungen der Validierung. Basierend auf diesem Input wird das Wissen über AR-Systeme analysiert und dem Anwender geeignete AR-Endgeräte, zu implementierende Softwarefunktionalitäten sowie zu berücksichtigende Aspekte als Ergebnis ausgegeben.

Das Konfigurationswerkzeug wurde als **Java-basierte Softwareanwendung** realisiert. Als Entwicklungsumgebung wird die **Eclipse IDE for Java Developers**¹⁴ verwendet. Als Input erhält die Softwareanwendung die Wissensbasis in Form der **.owl-Datei**. Der Zugriff auf die Wissensbasis erfolgt über die **SWRLAPI**. Die SWRLAPI unterstützt OWL-basierte SWRL-Regeln sowie SQWRL-Queries zum gezielten Zugriff auf in der Wissensbasis enthaltenes Wissen (vgl. Anhang A2.2.3). Über grafische Benutzeroberflächen ermöglicht die Softwareanwendung Eingaben durch den Anwender und gibt Analyse-Ergebnisse zurück.

Im folgenden Abschnitt 5.3.2.1 wird zunächst die Struktur der entwickelten Softwareanwendung beschrieben. Anschließend werden die einzelnen Schritte der System-Konfiguration im Detail erläutert (Abschnitt 5.3.2.2).

5.3.2.1 Software-Struktur

Die entwickelte Softwareanwendung setzt sich aus **46 Klassen** zusammen, die in **acht funktionalen Packages** organisiert sind. Abbildung 5-28 gibt einen Überblick über die Struktur der Software sowie die wesentlichen Zusammenhänge. Dabei wurde die Darstellung zur Vereinfachung auf die wichtigsten Klassen reduziert. Die farblich hervorgehobenen Bereiche stellen die Packages dar. Graue Packages enthalten Controller- und Hilfs-Funktionalitäten. Im *configurationTool*-Package erfolgt der Start der Anwendung sowie die Kontrolle des gesamten Konfigurationsablaufs inklusive der Darstellung der grafischen Interaktionsfenster. Das grün dargestellte Package *owlSerialization* realisiert den Zugriff auf die Wissensbasis und die Serialisierung des Wissens. In diesem Package kommt die SWRLAPI zum Einsatz. Die eigentliche Analyse und Auswertung des Wis-

¹⁴ <https://www.eclipse.org/downloads/packages/release/2021-12/r/eclipse-ide-java-developers>

sens finden im *knowledgeAnalysis*-Package, in Blau dargestellt, statt. In der Klasse *Analyzer* laufen alle relevanten Informationen zusammen. Über die gelb dargestellten Packages werden verschiedene Aspekte der Anwendereingaben sowie die Ausgabe von Ergebnissen realisiert. Zur Interaktion mit dem Anwender werden jeweils interaktive grafische Benutzeroberflächen in Form von Anwendungsfenstern verwendet. Sie basieren auf der Java-Klasse *JFrame*, über die sich unter anderem Textfelder, Eingabemasken und Interaktionselemente wie Button integrieren lassen.

5.3.2.2 Interaktive AR-System-Konfiguration

Ausgangspunkt der AR-System-Konfiguration sind zu nutzende AR-Potentiale. Die AR-Potentiale haben gewisse Anforderungen an Systemkomponenten und Umfeldfaktoren oder schließen diese aus. Zudem erfordern sie gewisse Softwarefunktionalitäten. So ergibt sich durch die Auswahl von AR-Potentialen eine initiale Menge an geforderten und ausgeschlossenen Aspekten. Diese wiederum haben weitere Abhängigkeiten, sodass sich eine Kette an Beziehungen und Wechselwirkungen ergibt. Ausgehend von den zu nutzenden AR-Potentialen erfolgt entsprechend eine schrittweise Analyse des Wissens. Dabei kann der Anwender an verschiedenen Stellen weitere Konfigurationen vornehmen und die Auswertung beeinflussen. Wenn alle Konfigurationen definiert und ausgewertet sind, erfolgt ein Abgleich mit den AR-Endgeräten. So können entsprechend der vorliegenden Anforderungen und Rahmenbedingungen in Frage kommende AR-Endgeräte und weitere zu berücksichtigende Informationen identifiziert werden. Im Folgenden wird die schrittweise AR-System-Konfiguration anhand der in Abbildung 5-28 dargestellten elf Schritte genauer beschrieben.

Schritt 1: Die Softwareanwendung wird über die main-Funktion in der Klasse *SystemConfigurator* gestartet. Als Parameter wird der **Dateipfad zur Wissensbasis** übergeben und im *QueryEngineProvider* zum Zugriff auf die Datei gesetzt. Der *UIController* wird gestartet. Er koordiniert im weiteren Verlauf der Konfiguration die Anzeige der verschiedenen Interaktionsfenster. Zudem initiiert der *UIController* eine Instanz der Klasse *Analyzer*, über die Analyse und Auswertung stattfindet.

Schritt 2: Über den *MetaInformationInquirer* werden **grundlegende Informationen** aus der Wissensbasis ausgelesen und in Form eines *MetaInformationContainer* an den *Analyzer* übergeben. Dazu zählen unter anderem Listen der möglichen AR-Potentiale, Softwarefunktionalitäten sowie der AR-Endgeräte. Die Anfragen an die Wissensbasis werden über *QueryHandler* realisiert. Ein *QueryHandler* nutzt SWRLAPI-Funktionalitäten, um über SQWRL-Anfragen (s. Abschnitt 4.2.1.2) auf die Wissensbasis zuzugreifen und Wissen gezielt abzufragen. *QueryHandler* können als *QueryThreads* parallel und somit effizienter ausgeführt werden.

Schritt 3: Nachdem die möglichen AR-Potentiale eingelesen wurden, wird dem Anwender über ein Interaktionsfenster (*PotentialSelectionFrame*) die Möglichkeit gegeben, die zu nutzenden **AR-Potentiale** auszuwählen.

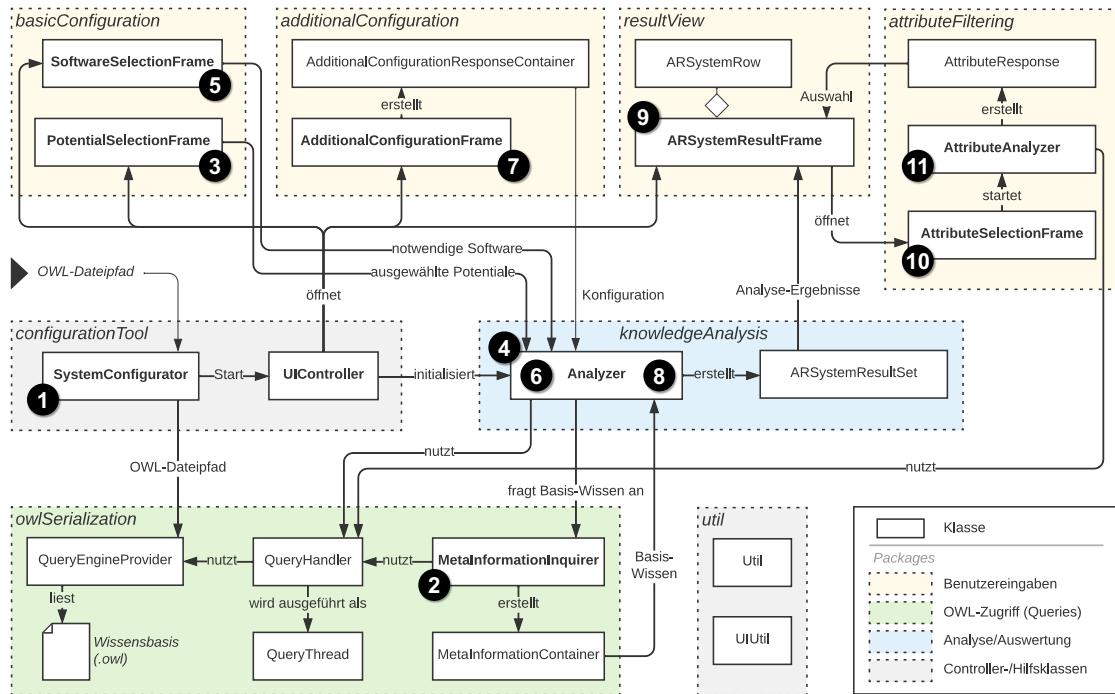


Abbildung 5-28: Software-Struktur des Konfigurationswerkzeugs inkl. Packages und wichtigster Klassen und Zusammenhänge sowie nummerierter Analyse- und Auswertungsschritte

Schritt 4: Die gewählten AR-Potentiale werden dem *Analyzer* übergeben. Dieser analysiert die Beziehungen der AR-Potentiale durch gezielte Anfragen an die Wissensbasis und erstellt initiale Listen mit **geforderten und ausgeschlossenen Systemkomponenten und Umfeldfaktoren** sowie **benötigten Softwarefunktionalitäten**. Geforderte und ausgeschlossene *Punning*-Klassen für Systemkomponenten und Umfeldfaktoren werden gesondert abgefragt und dokumentiert. Sie bedürfen im weiteren Verlauf einer individuellen Analyse (s. Schritt 9).

Schritt 5: Dem Anwender werden über ein weiteres Interaktionsfenster (*SoftwareSelectionFrame*) die für die AR-Potentiale benötigten Softwarefunktionalitäten präsentiert. Der Anwender erhält über das Interaktionsfenster die Möglichkeit, zusätzliche gewünschte **Softwarefunktionalitäten** auszuwählen. Wird ein Tracking zur Positionierung der virtuellen Prototypen benötigt, muss der Anwender in diesem Zusammenhang zudem eine Tracking-Lösung wählen. Zur Unterstützung dieser Entscheidung wird ein **Tracking-Steckbrief** mit relevanten Informationen und Eigenschaften je Tracking-Lösung bereitgestellt (Abbildung 5-29).

Schritt 6: Im *Analyzer* werden die ausgewählten Softwarefunktionalitäten analysiert und die durch sie geforderten Systemkomponenten identifiziert und in den entsprechenden Listen ergänzt. Anschließend erfolgt eine Analyse der in den Listen gesammelten geforderten und ausgeschlossenen Systemkomponenten und Umfeldfaktoren. Für jedes Ele-

ment werden die von ihm ausgehenden **Beziehungen analysiert** und entsprechende weitere geforderte oder ausgeschlossene Elemente in den Listen ergänzt. Durch einen Reasoneur wurden transitive Beziehungsketten bereits im Vorhinein analysiert und entsprechendes Wissen in der Wissensbasis integriert. Eine rekursive Analyse der sich möglicherweise durch die neu identifizierten geforderten Elemente ergebenden Beziehungsketten ist daher nicht notwendig.

Tracking-Varianten	Beschreibung	Vorteile	Nachteile
2D-Marker-Tracking	Virtuelle Inhalte werden anhand eines 2D-Markers, der in der Umgebung angebracht ist, positioniert. Es können Grafiken und Fotos verwendet werden.	<ul style="list-style-type: none"> - sehr genaue Position - (Dyn.) Positionsanpassung - Marker dienen der eindeutigen Identifikation 	<ul style="list-style-type: none"> - Marker muss in der Umgebung vorhanden sein - u.U. störende Beeinträchtigung der Umgebung
3D-Objekt-Tracking	Reale 3D-Objekte werden anhand ihrer Geometrie ohne zusätzliche Marker erkannt und virtuelle Inhalte entsprechend positioniert.	<ul style="list-style-type: none"> - genau Position - kein Marker notwendig - gleiche Objekte einsetzbar - dynamisches Tracking 	<ul style="list-style-type: none"> - komplexe Einrichtung - Ansatz noch nicht ausgereift - Objekt muss vorhanden sein
Freie Positionierung	Virtuelle Inhalte können beliebig und beweglich in der realen Umgebung positioniert werden. Es sind keine Marker oder Vorbereitungen notwendig.	<ul style="list-style-type: none"> - keine Vorbereitungen der Umgebung notwendig - unabhängig - Flexible Positionierung 	<ul style="list-style-type: none"> - keine sehr präzise Positionierung möglich - Interaktion durch den Benutzer notwendig
Statische Positionierung	Virtuelle Inhalte werden an einer festen Position in der Umgebung dargestellt. Die Position ist abhängig von der initialen Position des AR-Endgeräts.	<ul style="list-style-type: none"> - keine Vorbereitungen der Umgebung notwendig - unabhängig - geringe Komplexität 	<ul style="list-style-type: none"> - Position abhängig von Position des Gerät beim Start der Anwendung - keine Positionsanpassung

Abbildung 5-29: Steckbrief zu den verfügbaren Tracking-Varianten

Schritt 7: Die bisher identifizierten erforderlichen und ausgeschlossenen Systemkomponenten und Umfeldfaktoren werden dem Anwender über ein Interaktionsfenster ausgegeben. Der Anwender erhält zudem die Möglichkeit, weitere **individuelle Anforderungen und Rahmenbedingungen** zu definieren. Über Interaktionselemente in Form von Checkboxen kann definiert werden, ob ein einzelnes Element oder auch eine Klasse gefordert oder ausgeschlossen wird (Abbildung 5-30). Nach Bestätigung der Auswahl werden die Elemente in den entsprechenden Listen ergänzt und eine Auswertung und Ergänzung der Beziehungen der neuen Elemente durchgeführt. An dieser Stelle liegen die für die Analyse der AR-Endgeräte relevanten Informationen vor.

Schritt 8: Die in der Wissensbasis enthaltenen **AR-Endgeräte werden analysiert**. Zunächst wird für die geforderten Systemkomponenten überprüft, welche Geräte diese unterstützen. Für die geforderten *Punning*-Klassen wird analysiert, ob mindestens eine mögliche Ausprägung der entsprechenden Komponente von einem Gerät unterstützt wird. Dabei wird anhand der Liste der ausgeschlossenen Systemkomponenten überprüft, dass es sich bei der unterstützten Ausprägung nicht gleichzeitig um eine ausgeschlossene handelt. Zudem wird untersucht, ob ein AR-Endgerät die notwendigen Softwarefunktionalitäten ermöglicht. Softwarefunktionalitäten können durch verschiedene SDKs ermöglicht werden. Es wird daher überprüft, ob ein AR-Endgerät jeweils mindestens eins der möglichen SDKs unterstützt. Ist dies nicht der Fall, unterstützt das AR-Endgerät die Softwarefunktionalität nicht. AR-Endgeräte, die alle geforderten Aspekte unterstützen, gelten zunächst als in Frage kommend. Die Ergebnisse der Analyse werden in Form eines *ARSysystemResultSet* dokumentiert. Dabei wird für jedes AR-Endgerät gespeichert, welche erforderli-

chen Systemkomponenten nicht unterstützt werden, welche Konflikte mit ausgeschlossenen Systemkomponenten bestehen sowie welche notwendigen Softwarefunktionalitäten nicht unterstützt werden.

The screenshot shows a software interface for defining individual requirements. The window is titled '7' and contains three main sections:

- Essentielle Komponenten** (Essential Components):
 - Anzeigesystem** (Display System):
 - Projektions-basiert
 - Optical-See-Through
 - Video-See-Through
 - Bewegungsverfolgung** (Movement Tracking):
 - Rekonstruktives Tracking SLAM
 - GPS-Tracking
 - Elektromagnetisches Tracking
 - Mechanisches Tracking
 - Infrarot-Tracking
 - WiFi-Tracking
 - Sequenzielles Tracking
 - Ultraschall-Tracking
 - UWB-Tracking
 - Inertiales Tracking
 - Bluetooth-Tracking
 - Datenspeicher** (Data Storage):
 - Cloudspeicher
 - MemoryCard
 - Festplatte
 - Interner Speicher
- Optionale Komponenten** (Optional Components):
 - Anschluss** (Connection):
 - DisplayPort
 - Apple-Lightning
 - USB
 - HDMI
 - Audio-Input** (Audio Input):
 - Mikrofon
 - Kopfhörer Mikrofon
 - Bluetooth Mikrofon
 - Audio-Output** (Audio Output):
 - Kopfhörer
 - Lautsprecher
 - Bluetooth Kopfhörer
 - Daten-Verbindung** (Data Connection):
 - ZigBee
 - Bluetooth
 - FIR-IrDA
 - NFC
 - Fremdkörperschutz** (Foreign Object Protection):
 - IP5x
 - IP4x
 - IP6x
- Umgebungs faktoren** (Environmental Factors):
 - Arbeitsschut zvorschrift** (Workplace Safety Regulation):
 - Schutzhelm-Vorschrift
 - Schutzbrillen-Vorschrift
 - Ohropax-Vorschrift
 - Arbeitsumgebung** (Work Environment):
 - Büro
 - Werkstatt
 - Reinraum
 - Sterilraum
 - Einsatzort** (Deployment Site):
 - Outdoor
 - Indoor
 - Einsatzvorbereitung** (Deployment Preparation):
 - Nicht Präparierte Umgebung
 - Präparierte Umgebung
 - Fremdkörper** (Foreign Object):
 - Größer 50mm
 - Kleiner 2k5mm
 - Kleiner 1mm
 - Kleiner 50mm
 - Kleiner 12mm
 - Handhabung** (Handling):
 - Keine Hand frei mit Freiraum
 - Eine Hand frei mit Freiraum
 - Beide Hände frei
 - Keine Hand frei
 - Eine Hand frei
 - IT-Infrastruktur** (IT Infrastructure):
 - UMTS-Netz
 - FünG-Netz
 - Cloud
 - Steckdose
 - WLAN-Router

Abbildung 5-30: Interaktionsfenster zur Definition individueller Anforderungen

Schritt 9: Die Ergebnisse der AR-Endgeräte-Analyse werden dem Anwender in einem Fenster (*ARSystemResultFrame*) ausgegeben (Abbildung 5-31).

Den Hauptteil des Fensters nimmt eine scrollbare Liste ein, in der zeilenweise die **AR-Endgeräte mit zugehörigen Informationen** dargestellt werden. In der ersten Spalte werden jeweils die geforderten, aber nicht unterstützten Systemkomponenten ausgegeben. Die zweite Spalte enthält die Ausprägungen und zugehörigen Systemkomponenten, zu denen ein Konflikt besteht. Dies ist der Fall, wenn eine Punning-Klasse gefordert ist, ein AR-Endgerät aber lediglich eine Ausprägung dieser Klasse unterstützt, die ausgeschlossen wurde. In der dritten Spalte werden die Softwarefunktionalitäten gelistet, die vom AR-Endgerät nicht unterstützt werden. Zudem werden in der fünften Spalte die von einem AR-Endgerät geforderten SDKs angegeben. Dies dient bei Auswahl eines Geräts als Information für die anschließende Softwareentwicklung. AR-Endgeräte, die alle Anforderungen erfüllen, werden in Fettschrift dargestellt und mit einem grünen Hintergrund hervorgehoben. Rechts neben den AR-Endgeräten werden erkannte **Konflikte** und mögliche

Beeinflussungen ausgegeben. Konflikte bedeuten, dass ein Element gleichzeitig erfordert und ausgeschlossen ist und die Konfiguration angepasst werden sollte. Die Beeinflussungen umfassen alle zu den erforderlichen Systemkomponenten und Umfeldfaktoren bestehenden Einfluss-Beziehungen. Diese möglichen Beeinflussungen gilt es bei der späteren Validierung unter den gegebenen Rahmenbedingungen zu berücksichtigen. Als weitere Informationen werden dem Anwender im oberen Teil des Fensters die geforderten sowie die ausgeschlossenen **Umfeldfaktoren** gegeben. Diese gilt es ebenfalls bei der Validierung zu berücksichtigen. Zudem werden oben rechts die zu implementierenden **Softwarefunktionalitäten** und mögliche unterstützende SDKs als Information für die Implementierung ausgegeben.

Umfeld-Rahmenbedingungen

Geforderte bzw. vorhandene Umfeldfaktoren:		Ausgeschlossene Umfeldfaktoren:		Ausgewählte bzw. benötigte Software (inkl. mögl. SDKs)	
StarkeBeleuchtung		FunIG-Netz		Skalieren (MRTK, ARCore, ARKit, ARToolKit, Wikitude)	
HöheLautstärke				Marker-Tracking (VisionLab, OpenCV, Vuforia, ARKit, ARToolKit, Wikitude)	
ODER (EineHandfreiMeFreiraum, BeideHändeFrei, EineHandFrei)					

Relevante Software und mögliche SDKs

AR-System	Nicht unterstützte Komponenten	Konflikt mit ausgeschlossener Komponente	Mangelnde SW-Unterstützung	Erforderliche SDKs	Inkompatibilitäten
HoloLens2				MRTK	
VuzixBlade	System_SekundäreStromversorgung		Skalieren		
ODGR-7	System_Interaktionssystem		Skalieren		
LumusDK-52	System_Bewegungsverfolgung System_SekundäreStromversorgung		Skalieren		
MadGazeGlow	System_Datenspeicher System_SekundäreStromversorgung				
AppleiPad					
VuzixBladeSafety	System_SekundäreStromversorgung				
RokidProjectAURORA	System_Interaktionssystem System_SekundäreStromversorgung				
SamsungGalaxyA51					
EPSONMoverioBT-300					
VuzixM4000	System_SekundäreStromversorgung				
LightformLF2	System_PrimaryStromversorgung System_Bewegungsverfolgung System_SekundäreStromversorgung	System_PrimaryStromversorgung/Kabel	Speichergröße		
LumusDKVision	System_Datenspeicher System_SekundäreStromversorgung		Video-Framerate		
MagicLeap			Videoauflösung		

AR-Endgeräte und Analyse-Ergebnisse

AR-Endgerät	Attribut	Wert
HoloLens2	Akkulaufzeit_Ah	16.9
HoloLens2	Akkulaufzeit_h	1080
HoloLens2	Anzeigeauflösung	2K 3.2
HoloLens2	Anzeigehelligkeit	1.3M nits per eye
HoloLens2	Arbeitsspeicher	1230 x 1080
HoloLens2	FieldOfView	720px per eye
HoloLens2	Gewicht	16:9 Contrast > 500:1
HoloLens2	Kameraauflösung	
HoloLens2	Linsendurchlässigkeit	
HoloLens2	Preis	
HoloLens2	Prozessorne	
HoloLens2	Speichergröße	
HoloLens2	Video-Framerate	
HoloLens2	Videoauflösung	

Attribute Selection Frame

Nichts ausgewählt

✓ Nichts ausgewählt

16.9
1080
2K 3.2
1.3M nits per eye
1230 x 1080
720px per eye
16:9 Contrast > 500:1

OK

Attribute filtern

Abbildung 5-31: Ausgabefenster für die Analyse-Ergebnisse (Schritt 9) inkl. Eingabemaske zur Filterung nach Attributen (Schritt 10)

Schritt 10: Über einen Button im Ergebnisfenster kann ein zusätzliches Fenster zum **Filtern von Attributen** (*AttributeSelectionFrame*) angezeigt werden (s. Abbildung 5-31). Es bietet Eingabefelder zur Definition von Minimal- oder Maximalwerten für alle numerischen Attribute und eine Dropdown-Liste für das textbasierte Attribut *Anzeigeauflösung*. Der Anwender kann entsprechend weitere Rahmenbedingungen definieren, beispielsweise das maximal verfügbare Budget, und so die AR-Endgeräte weiter filtern.

Schritt 11: Über einen Button im *AttributeSelectionFrame* wird die Filterung bestätigt und die Werte an den *AttributeAnalyzer* übergeben. Dieser überprüft die Attributwerte der AR-Endgeräte durch gezielte Anfragen an die Wissensbasis und gibt die **gefilterten**

Ergebnisse als *AttributeResponse* an das Ergebnisfenster (*ARSystemResultFrame*, s. Schritt 9) weiter. Die Ergebnisdarstellung wird entsprechend angepasst, sodass nun lediglich die AR-Endgeräte als in Frage kommend hervorgehoben werden, die sowohl den grundsätzlichen Anforderungen als auch den gefilterten Attributen entsprechen. In Abbildung 5-31 trifft dies nur auf die *HoloLens 2* zu. Zur Information für den Anwender werden alle AR-Endgeräte, die zumindest die Attributfilterung erfüllen, in grüner Schrift dargestellt. Diese finale Darstellung dient als Grundlage für die Auswahl eines geeigneten AR-Endgeräts. Unterstützend für den Anwender können die durch RÖLTGEN erarbeiteten AR-Endgeräte-Steckbriefe verwendet werden (s. Abbildung A-17). Die Steckbriefe stellen grundsätzliche Informationen und technische Spezifikationen bereit, die bei der Auswahl eines AR-Endgeräts berücksichtigt werden können.

Das interaktive Konfigurationswerkzeug bietet dem Anwender die Möglichkeit, ohne eigene Erfahrung und Technologie-Expertise eine individuelle AR-System-Konfiguration durchzuführen. Dazu wird das in der AR-System-Wissensbasis abgebildete für die Konfiguration relevante Wissen automatisiert ausgelesen und ausgewertet. Entsprechend der für die Validierung vorliegenden Anforderungen und Rahmenbedingungen kann der Anwender ein bedarfsgerechtes AR-Endgerät identifizieren. Dabei werden individuelle Kriterien wie beispielsweise das Budget direkt berücksichtigt. Der Anwender erhält Informationen über relevante Umfeldfaktoren und möglichen sich aus der System-Konfiguration ergebenden Beeinflussungen. Diese können bei der späteren Durchführung der Validierung gezielt berücksichtigt werden, um einen erfolgreichen Einsatz der Technologie als Validierungswerkzeug zu ermöglichen. Zudem erhält der Anwender Informationen über die für die Ausnutzung der Potentiale von AR zu implementierenden Softwarefunktionalitäten. Entsprechend ist an dieser Stelle die technische **Planung der AR-basierten Validierung abgeschlossen**. Im nächsten Schritt, der Vorbereitung der Validierung, gilt es nun, die Anwendungssoftware entsprechend der definierten Softwarefunktionalitäten zu implementieren.

5.4 Werkzeugkoffer zur aufwandsarmen Entwicklung individueller AR-Validierungsanwendungen

Im Anschluss an die AR-System-Konfiguration erfolgt die Entwicklung der AR-Validierungsanwendung. Diese entspricht zusammen mit der Vorbereitung der virtuellen Prototypen der technischen **Vorbereitung der AR-basierten Validierung**. Nach Abschluss der Entwicklung liegt eine bedarfsgerechte AR-Anwendung auf einem zielgerichtet ausgewählten AR-Endgerät vor, die als Validierungswerkzeug eingesetzt werden kann.

Der Stand der Technik zeigt, dass bisher keine AR-Validierungsanwendung existiert, mit der sich alle Potentiale der Technologie AR zur Unterstützung der Validierung vollumfänglich erschließen lassen. Unter anderem die Studie von VAN GOETHEM ET AL. (Anhang A2.3.4) zeigt, dass die AR-Anwendungsfunktionalitäten individuell auf das jeweilige Va-

lidierungsszenario zugeschnitten sein sollten, um einen größtmöglichen Nutzen zu erzielen. Weder am Markt verfügbare AR-Lösungen (Abschnitt 4.3.3.2) noch No-Code-Editoren (Abschnitt 4.3.2.3) ermöglichen dies. Entsprechend muss eine **individuelle AR-Anwendungsentwicklung** erfolgen. Unternehmen sind bisher jedoch nicht in der Lage, die komplexe und aufwendige Entwicklung entsprechender AR-Validierungsanwendungen eigenständig durchzuführen. Im Rahmen dieses Handlungsfelds war daher das Ziel, die Entwicklungsaufwände zu reduzieren und Unternehmen zur Entwicklung individueller AR-Validierungsanwendungen zu befähigen.

AR-Anwendungen werden in speziellen Entwicklungsumgebungen entwickelt. Eine grundlegende Unterstützung erfolgt bereits durch Software Development Kits, die in den Entwicklungsumgebungen verwendet werden können (Abschnitt 4.3.1.2). Sie stellen grundlegende Funktionalitäten bereit und erfordern weiterhin relativ hohe Programmieraufwände. Im Forschungsprojekt AcRoSS wurde dagegen ein Ansatz zur individuellen Realisierung von AR-Lösungen ohne die Notwendigkeit von einer Entwicklungsumgebung und Programmierung erforscht (Abschnitt 4.3.2.2). Ziel des Ansatzes war, modulare Funktionsbausteine auf einer Webplattform zu individuellen AR-Anwendungen miteinander zu kombinieren und von dort direkt auf AR-Endgeräte zu installieren. Das Projekt hat gezeigt, dass sich AR-Anwendungen aufgrund der unterschiedlichen Eigenschaften und Anforderungen nicht außerhalb einer Entwicklungsumgebung individuell für verschiedene AR-Endgeräte konfigurieren und realisieren lassen. Aus diesem Grund wurde im Rahmen dieser Arbeit ein Mittelweg zwischen der Bereitstellung grundlegender Funktionalitäten und der kompletten Vermeidung von Entwicklungsarbeit erarbeitet. Die Entwicklung findet weiterhin in einer AR-Entwicklungsumgebung statt. Zur Unterstützung der Entwicklung wurde ein Konzept für einen Werkzeugkoffer erarbeitet. Der Werkzeugkoffer enthält modulare für die Validierung relevante Anwendungsfunktionen sowie verschiedene weitere Hilfsmittel und grafische Benutzeroberflächen. So können die **Entwicklungsaufwände sowie -komplexität stark reduziert** werden. Gleichzeitig ermöglicht die Entwicklungsumgebung eine flexible Adressierung gerätespezifischer Aspekte und somit die geräteübergreifende Anwendung des Werkzeugkoffers. Das Konzept wurde teilweise bereits softwaretechnisch umgesetzt. Exemplarische Werkzeuge und Hilfsmittel sind einsatzbereit im Werkzeugkoffer vorhanden und wurden bereits für einen konkreten Anwendungsfall verwendet und erprobt (s. Kapitel 6).

Im folgenden Abschnitt 5.4.1 wird zunächst das übergreifende Konzept des Werkzeugkoffers vorgestellt, gefolgt von der Beschreibung eines Leitfadens zur Anleitung der Entwicklung (Abschnitt 5.4.2). Anschließend werden mit Templates (Abschnitt 5.4.3), Funktionsbausteinen (Abschnitt 5.4.3.2) und Hilfsmitteln (Abschnitt 5.4.3.3) die drei wesentlichen Werkzeugarten im Detail erläutert. Einzelne Hilfsmittel werden zum Teil aufgrund ihrer engen Verzahnung mit dem Leitfaden, den Templates und den Bausteinen bereits in den entsprechenden Abschnitten erläutert.

5.4.1 Das Konzept des Werkzeugkoffers im Überblick

Der Werkzeugkoffer zur aufwandsarmen Entwicklung individueller AR-Validierungsanwendungen unterstützt die *Implementierung* als mittlere Phase im Software Development Lifecycle (SDLC) (Abschnitt 4.3.1.1). Die im SDLC vorangehenden Phasen *Planung*, *Analyse* und *Design* sind zu Beginn der Implementierung bzw. Entwicklung bereits abgeschlossen. Die *Planung* wird durch die Systematik zur Nutzen-Aufwands-Analyse aus Handlungsfeld 1 (Kapitel 5.2) unterstützt. Die *Analyse* und das *Design* erfolgen im Rahmen der AR-System-Konfiguration in Handlungsfeld 2 (Kapitel 5.3). Lediglich das Konzept des Erscheinungsbilds der AR-Validierungssoftware als Teil der *Design*-Phase sind individuell durch das Unternehmen zu erarbeiten. Alternativ beinhaltet der Werkzeugkoffer ein vordefiniertes Erscheinungsbild, das verwendet werden kann. Abbildung 5-32 zeigt das Konzept des Werkzeugkoffers und seine vier wesentlichen Komponenten im Überblick.

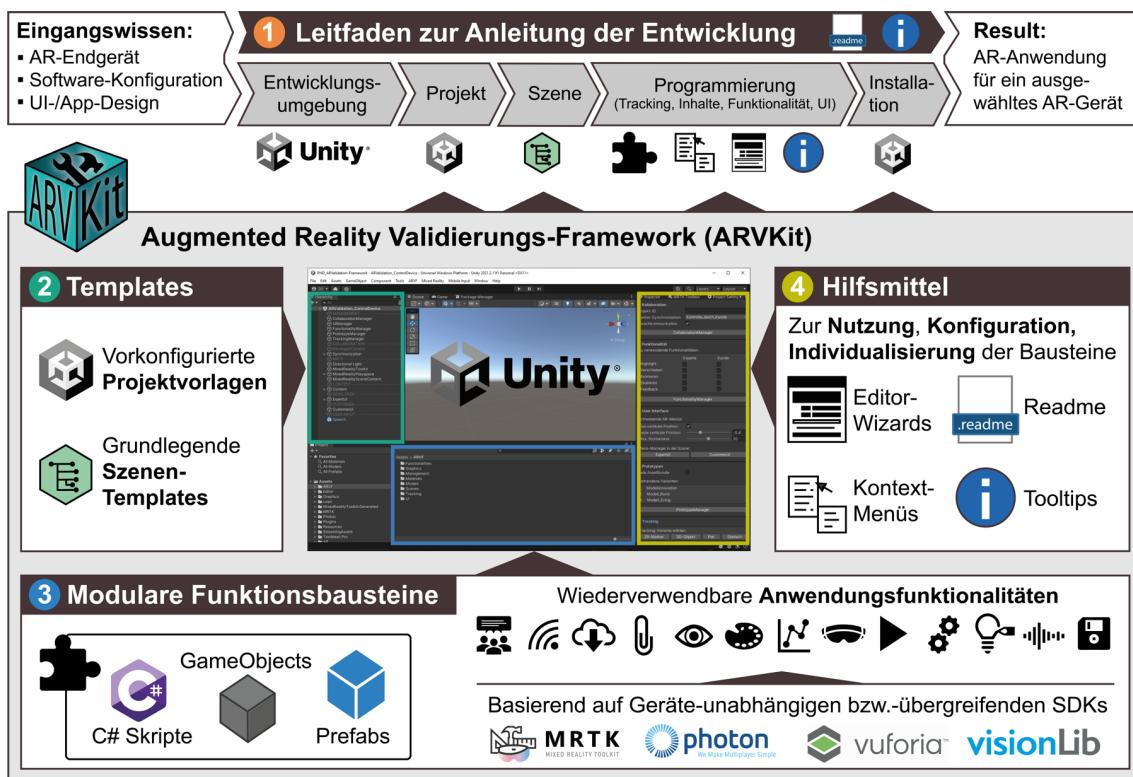


Abbildung 5-32: Werkzeugkoffer zur aufwandsarmen Entwicklung individueller AR-Validierungssoftware

Nach Abschluss der *Planung*, *Analyse* und des *Designs* liegen als **Eingangswissen** das ausgewählte AR-Endgerät, die zu implementierenden Software-Funktionalitäten sowie das User Interface (UI)- und App-Design vor. Basierend auf diesem Wissen erfolgt die Entwicklung der AR-Validierungsanwendung unterstützt durch den Werkzeugkoffer. Als übergeordnete Komponente erhält der Werkzeugkoffer einen **Leitfaden zur Anleitung der Entwicklung (1)**. Der Leitfaden ist orientiert am Entwicklungsprozess von AR-An-

wendungen als spezifische Ausprägung der *Implementierungs*-Phase des SDLC (Abschnitt 4.3.1.1). In der ersten Phase muss die *Entwicklungsumgebung* ausgewählt werden. Unity stellt die gängigste Entwicklungsumgebung für AR-Anwendungen dar. Unity bietet zudem vielfältige Möglichkeiten zur Vereinfachung und Unterstützung der Entwicklung (Details sind Anhang A2.3.1 zu entnehmen). Aus diesem Grund wurde **Unity als grundlegende Entwicklungsumgebung** für den Werkzeugkoffer ausgewählt. Für die weiteren Phasen der Entwicklung werden verschiedene Hilfsmittel bereitgestellt. Diese Hilfsmittel werden im **Augmented Reality Validierungs-Framework (ARVKit)** als Kern des Werkzeugkoffers zusammengefasst.

Für die Phasen *Projekt* und *Szene* werden vorkonfigurierte *Projektvorlagen* sowie **grundlegende Szenen** als **Templates (2)** bereitgestellt. Für die eigentliche Entwicklung und Programmierung der AR-Validierungsanwendung stehen **modulare Funktionsbausteine (3)** zur Verfügung. Sie wurden basierend auf Geräte-unabhängigen bzw. -übergreifenden SDKs entwickelt. Dies ermöglicht eine Verwendung der Bausteine auf verschiedenen Endgeräten. Im Gegensatz zu klassischen SDKs stellen die Bausteine Funktionalitäten auf Anwendungsebene bereit. Diese entsprechen einzelnen Anwendungsfunktionalitäten, die im Kontext der Produktvalidierung relevant sind. Sie bauen auf den in Handlungsfeld 2 definierten Softwarefunktionalitäten (Abschnitt 5.3.1.1) sowie Erkenntnissen aus den im Stand der Technik untersuchten Software-Lösungen (Abschnitt 4.3.3) auf. Beispielsweise werden Aspekte der Visualisierung, Interaktion, Kommunikation oder auch der Datenintegration adressiert. Die Bausteine werden in Form von *C#-Skripten* und *Prefabs* (vgl. Anhang A2.3.1) im Asset-Bereich bereitgestellt. Über geeignete Schnittstellen und Funktionen können die Bausteine aufwandsarm miteinander verbunden und so zu individuellen AR-Anwendungen kombiniert werden. Somit werden die Entwicklungs- und vor allem die Programmieraufwände stark reduziert. Nachteil der Bereitstellung von Bausteinen auf Anwendungsebene ist eine reduzierte Flexibilität für den Entwickler. Da die Bausteine jedoch auf den Anwendungskontext Validierung zugeschnitten sind und anzunehmen ist, dass die Entwickler über geringe AR-Programmiererfahrung verfügen, kann der Aspekt der reduzierten Flexibilität vernachlässigt werden. Um die Entwicklung zusätzlich zu unterstützen, werden weitere **Hilfsmittel (4)** im Rahmen des ARVKit-Frameworks bereitgestellt. Ein *Readme* sowie *Tooltips* ergänzen den Leitfaden und leiten den Entwickler in der Verwendung der Bausteine an. Über *Editor-Wizards* werden zudem bedarfsgerechte grafische Benutzeroberflächen, beispielsweise zur Individualisierung der Anwendung, bereitgestellt. Zusätzlich ermöglichen *Kontext-Menüs* die einfache Nutzung und Konfiguration von Bausteinen direkt in der Szene.

Resultat der Entwicklung ist eine bedarfsgerechte AR-Anwendung für ausgewählte AR-Endgeräte. Im Software Development Lifecycle (SDLC) folgen auf die Entwicklung die Phasen *Testen*, *Deployment* und *Wartung und Updates*. Das *Testen* sollte bereits parallel zur Entwicklung kontinuierlich stattfinden. Dies wird durch den Leitfaden adressiert. Das *Deployment* wird durch den Leitfaden ebenfalls beschrieben, um Unternehmen bis zur

einsatzbereiten AR-Anwendung anzuleiten. Die Phase *Wartung und Updates* ist individuell durch die Unternehmen durchzuführen und zu gestalten. Updates und Anpassungen an einer AR-Anwendung können durch Einsatz des Werkzeugkoffers selbstständig durchgeführt werden. Die Abläufe entsprechen den im Leitfaden beschriebenen Aktivitäten zur Entwicklung einer Anwendung. Bei Updates und Anpassungen können einzelne Schritte gezielt ausgeführt und andere ausgelassen werden. Entsprechend unterstützt der Werkzeugkoffer in Kombination mit den Lösungen aus den Handlungsfeldern 1 und 2 den gesamten Entwicklungsprozess individueller AR-Validierungsanwendungen. Unternehmen werden entsprechend befähigt, AR-Anwendungen als Validierungswerzeuge eigenständig vorzubereiten.

5.4.2 Leitfaden zur Anleitung der Entwicklung

Abbildung 5-33 zeigt den Leitfaden in der Übersicht. Entwickler werden schrittweise in der Entwicklung individueller AR-Validierungsanwendungen angeleitet. Das durch den Leitfaden beschriebene **Vorgehen** entspricht grundsätzlich dem im Abschnitt 4.3.1.1 beschriebenen Prozess zur Entwicklung von AR-Anwendungen. Dabei wird gezielt der Einsatz der im ARVKit verfügbaren Werkzeuge adressiert, um eine aufwandsarme Entwicklung zu ermöglichen. Der Leitfaden wird unterstützt durch weitere Hilfsmittel, unter anderem das speziell für den Einsatz des ARVKit erarbeitete Konfigurationsmenü (s. Abschnitt 5.4.3.3), das wesentliche Einstellungsmöglichkeiten in einem zentralen Fenster bündelt. In diesem Abschnitt erfolgt zunächst eine grundsätzliche Beschreibung der einzelnen Entwicklungsschritte. Anschließend werden in den Abschnitten 5.4.3, 5.4.3.2 und 5.4.3.3 Details zu den Templates, Funktionsbausteinen und Hilfsmitteln beschrieben.

- 1) **Entwicklungsumgebung:** Als Entwicklungsumgebung ist **Unity** festgelegt. Die jeweils aktuelle Version kann über das das **Unity Hub**¹⁵ kostenlos installiert werden. Je nach AR-Endgerät ist das Betriebssystem für die Entwicklungsumgebung passend zu wählen. Für eine HoloLens- oder Android-AR-Anwendung ist Windows zu nutzen. Die Entwicklung einer AR-Anwendung für iOS-Geräte erfordert dagegen ein MacOS-basiertes Gerät (bspw. Apple iMac oder MacBook). Für den kommerziellen Einsatz von Unity ist eine Pro-Lizenz zu beschaffen. Neben Unity wird zudem ein Code-Editor benötigt. Ein gängiger Editor ist beispielweise **Microsoft Visual Studio**, das sowohl auf Windows und MacOS kostenlos verfügbar ist.
- 2) **Projekt:** Im Werkzeugkoffer sind verschiedene **Unity-Projektvorlagen** enthalten. Entsprechend des ausgewählten AR-Endgeräts ist die passende Projektvorlage zu wählen und zu öffnen. Die Projektvorlage dient als Grundlage für die weitere Entwicklung. Alle relevanten Software Development Kits sind bereits integriert und notwendige Konfigurationen für die spätere Installation sind vordefiniert. Zudem enthält

¹⁵ Verfügbar unter <https://unity.com/download>

jedes Projekt eine Readme-Datei, die beim Öffnen Projekts im Unity-Inspector dargestellt wird und die Arbeit mit dem jeweiligen Projekt anleitet (s. Abschnitt 5.4.3.3)

- 3) **Szene:** Wie in den Ergänzungen zum Stand der Technik (Anhang A2.3.1) beschrieben wird, erfolgt die eigentliche Erstellung von AR-Anwendungen in sogenannten Szenen. Innerhalb der Projektvorlagen stehen **Szenen-Templates** bereit, die bereits eine Grundstruktur für verschiedene Grundtypen von AR-Validierungsanwendungen in Form eines Szenengraphen enthalten. Bei den Szenen-Templates wird unterschieden zwischen Einzelnutzer-Anwendungen und kollaborativen Anwendungen zur gemeinsamen Durchführung von Validierungsaktivitäten. Basierend auf den Templates können individuelle Szenen erstellt werden.

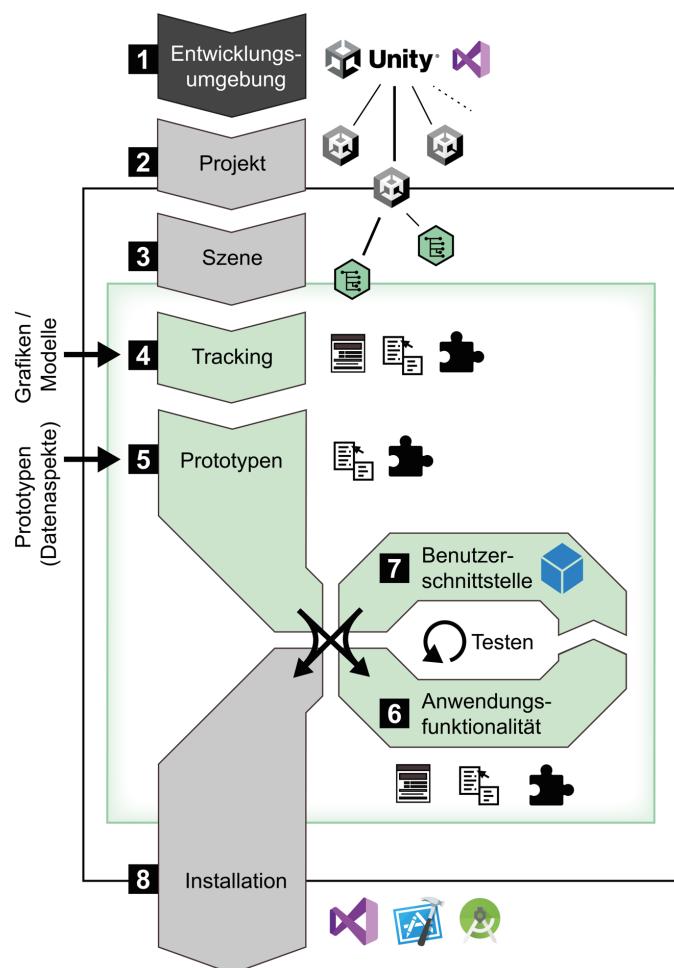


Abbildung 5-33: Leitfaden zur Anleitung der Entwicklung individueller AR-Validierungsanwendungen in der Entwicklungsumgebung Unity

Die folgenden Entwicklungsschritte 4 bis 7 finden anschließend in der jeweiligen Szene statt. Sie entsprechen der eigentlichen inhaltlichen und funktionalen Entwicklung und Gestaltung der AR-Validierungsanwendung. In diesen Schritten kommen die Funktionsbausteine und Hilfsmittel zum Einsatz.

- 4) **Tracking:** In der Szene wird zunächst das Tracking konfiguriert. Es dient der passenden Positionierung der virtuellen Prototypen in der realen Welt. Je nach Anforderungen und entsprechender AR-System-Konfiguration können **unterschiedliche Tracking-Lösungen** zum Einsatz kommen. Die einzelnen Lösungen werden als Funktionsbausteine bereitgestellt (s. Abschnitt 5.4.3.2). Die grundsätzliche Auswahl und Integration der Trackinglösungen erfolgt aufwandsarm über Button im Konfigurationsmenü (s. Abschnitt 5.4.3.3). Bei 2D-Marker- und 3D-Objekt-Tracking sind Marker vorzubereiten und zu integrieren. Weitere Tracking-Konfigurationen können anschließend direkt am entsprechenden Baustein im Unity-Inspector vorgenommen werden.
- 5) **Prototypen:** Nachdem das Tracking konfiguriert wurde, können die Prototypen integriert werden. Entsprechende Modelle und weitere Datenaspekte werden in Unity importiert und in die Szene integriert. Zur Anbindung an die Anwendungslogik und die Benutzerschnittstellen stehen verschiedene Funktionsbausteine bereit (s. Abschnitt 5.4.3.2). Neben der festen Integration von Daten in die Anwendung enthält der Werkzeugkoffer auch Bausteine zum Import von lokalen oder Online verfügbaren Daten während der Laufzeit.

Die Schritte 6 und 7 sind sowohl als optional als auch als iterativ anzusehen. Sie können je nach AR-System-Konfiguration gar nicht oder mehrfach durchlaufen werden.

- 6) **Anwendungsfunktionalität:** Mittels der modularen Bausteine werden Anwendungsfunktionalitäten in die AR-Validierungsanwendung integriert. Die Bausteine liegen als Assets im Unity-Projektordner vor und können per Drag-and-Drop verwendet werden. Sie sind so weit wie möglich Plattform-unabhängig konzipiert und realisiert, sodass sie auf verschiedenen Plattformen verwendet werden können. Für einzelne Bausteine wird die Integration durch das Konfigurationsmenü sowie individuelle Kontextmenüs unterstützt und vereinfacht. Zusätzliche Erklärungen zu Bausteinen und deren Komponenten erfolgen in Readme-Dateien sowie sogenannten Tooltips (s. Abschnitt 5.4.3.3).
- 7) **Benutzerschnittstelle:** Für viele der Anwendungsfunktionalitäten werden spezifische Benutzerschnittstellen in Form von Interaktionsmenüs benötigt. Bei Benutzerschnittstellen handelt es sich um die Elemente, die grundsätzlich gerätespezifisch sind. Die genaue Realisierung der Benutzerschnittstellen wird zudem stark von der AR-System-Konfiguration beeinflusst. Des Weiteren können speziell bei an Kunden herausgegebenen Anwendungen individuelle Design-Richtlinien zu berücksichtigen sein. Die Benutzerschnittstellen lassen sich daher nicht als flexibel einsetzbare Bausteine vorbereiten. Sie sind grundsätzlich individuell zu gestalten und als Prefabs im Projekt anzulegen. Durch die Referenzierung der Prefabs in den jeweiligen Funktionsbausteinen werden sie bedarfsgerecht angezeigt und ermöglichen eine entsprechende Interaktion. Details dazu folgen in Abschnitt 5.4.3.2.

8) Installation: Nachdem alle notwendigen und gewünschten Funktionalitäten integriert wurden, kann die Installation auf dem entsprechenden AR-Endgerät erfolgen. Diese erfolgt mittels der standardmäßigen Prozesse in Unity. Wie in Schritt 2 bereits beschrieben, sind die grundsätzlichen Einstellungen für das jeweilige Gerät bereits im Projekt konfiguriert. Individuelle Einstellungen wie die App-Bezeichnung oder ein Logo können in den Projekteinstellungen vorgenommen werden. Je nach Zielplattform sind entsprechend der standardmäßigen Installationsprozesse weitere Software-Werkzeuge als Schnittstelle zum AR-Endgerät notwendig. Die Installation auf Universal Windows Platform (UWP)-Geräten wie beispielsweise der Microsoft HoloLens oder HoloLens 2 erfordert *Visual Studio*. Android-Geräte werden über *Android Studio* angesprochen und iOS-Geräte über Apples *Xcode*. Alternativ ist auch ein Export zur Bereitstellung der AR-Validierungsanwendung in einem entsprechenden App Store möglich.

5.4.3 ARVKit - Augmented Reality-Validierungs-Framework

Die verschiedenen Werkzeuge zur Unterstützung der Entwicklung von AR-Validierungsanwendungen werden im ARVKit zusammengefasst. Im Folgenden werden zunächst die Templates beschrieben (Abschnitt 5.4.3.1), gefolgt von den modularen Funktionsbausteinen (Abschnitt 5.4.3.2) sowie den weiteren Hilfsmitteln (Abschnitt 5.4.3.3).

5.4.3.1 Templates

Bei der Entwicklung einer neuen AR-Anwendung in Unity muss zunächst ein neues Projekt angelegt werden. In diesem Projekt werden anschließend relevante Software Development Kits (SDKs) importiert und konfiguriert. Neben für einzelne Geräte grundsätzlich notwendigen SDKs gehören dazu auch SDKs, die spezielle Anwendungsfunktionen unterstützen. Der SDK-Integrationsprozess ist aufgrund verschiedener Quellen und Arten der Bereitstellung der SDKs meist aufwands- und vor allem zeitintensiv. Ebenso ist für die Konfiguration einzelner SDKs Erfahrung oder umfangreiche Einarbeitung notwendig. Daher werden im Rahmen des ARVKit vorkonfigurierte **Unity-Projektvorlagen** bereitgestellt. Projektvorlagen adressieren einzelne AR-Endgeräte bzw. -Plattformen und enthalten bereits die relevanten und kompatiblen SDKs. Abbildung 5-34 gibt einen Überblick über die bisher entsprechend den gängigsten AR-Plattformen konzipierten Projektvorlagen.

Aufgrund der vorhandenen Kompatibilität zwischen den gängigen UWP-, Android- und iOS-Geräten kann eine übergreifende Projektvorlage für diese Plattformen verwendet werden. Die Vorlage enthält das plattformübergreifende *AR Foundation-Framework* für grundsätzliche AR-Funktionalitäten (s. Abschnitt 4.3.1.2). Aufgrund der vielfältigen Anwendungsmöglichkeiten sowie der Unterstützung des *OpenXR*-Standards wird zudem das *MRTK* (s. Abschnitt 4.3.1.2 sowie Anhang A2.3.3) auf allen unterstützten Plattformen

eingesetzt. Es bietet innovative Visualisierungs- und Interaktionsmöglichkeiten für verschiedene AR-Endgeräte. Für die Kommunikation und Kollaboration wurde *Photon* ausgewählt und integriert. Mit *Vuforia* ist zudem ein SDK enthalten, das verschiedene Tracking-Lösungen unterstützt. In Ergänzung zu *Vuforia* bietet *visionLib* 3D-Objekt-Tracking basierend auf CAD-Daten. Neben den SDKs werden den Projektvorlagen zudem Plugins aus dem Unity Asset Store hinzugefügt. Die Plugins stellen ähnlich zu SDKs verschiedene weitere Funktionalitäten bereit und dienen als Grundlage für die Hilfsmittel (s. Abschnitt 5.4.3.3). Die UWP-Projektvorlage wurde im Rahmen dieser Arbeit bereits erstellt und im Rahmen der Evaluierung des Instrumentariums eingesetzt und erprobt (s. Kapitel 6). Die weiteren Projektvorlagen wurden konzipiert, aber noch nicht realisiert. Die Magic Leap-Plattform benötigt das *LuminSDK* als Grundlage der Entwicklung. Das *MRTK* und *visionLib* werden nicht unterstützt und sind entsprechend nicht in der Projektvorlage enthalten. Gleiches gilt für die Vorlagen für nreal- und Lenovo-Geräte. Die Geräte benötigen das *nrSDK* bzw. das *ThinkReality* SDK. Diese SDKs bieten unter anderem auch Tracking- und Interaktionsmöglichkeiten, die aufgrund der mangelnden Unterstützung des *MRTK* und von *Vuforia* entsprechend für die Entwicklung der Funktionsbausteine verwendet werden. Als Plattform-übergreifendes SDK wird in diesen Vorlagen *Photon* für die Kommunikation und Kollaboration verwendet. Weitere Projektvorlagen können für weitere oder neu an den Markt gebrachte AR-Geräte konzipiert und im Werkzeugkoffer entsprechend ergänzt werden.

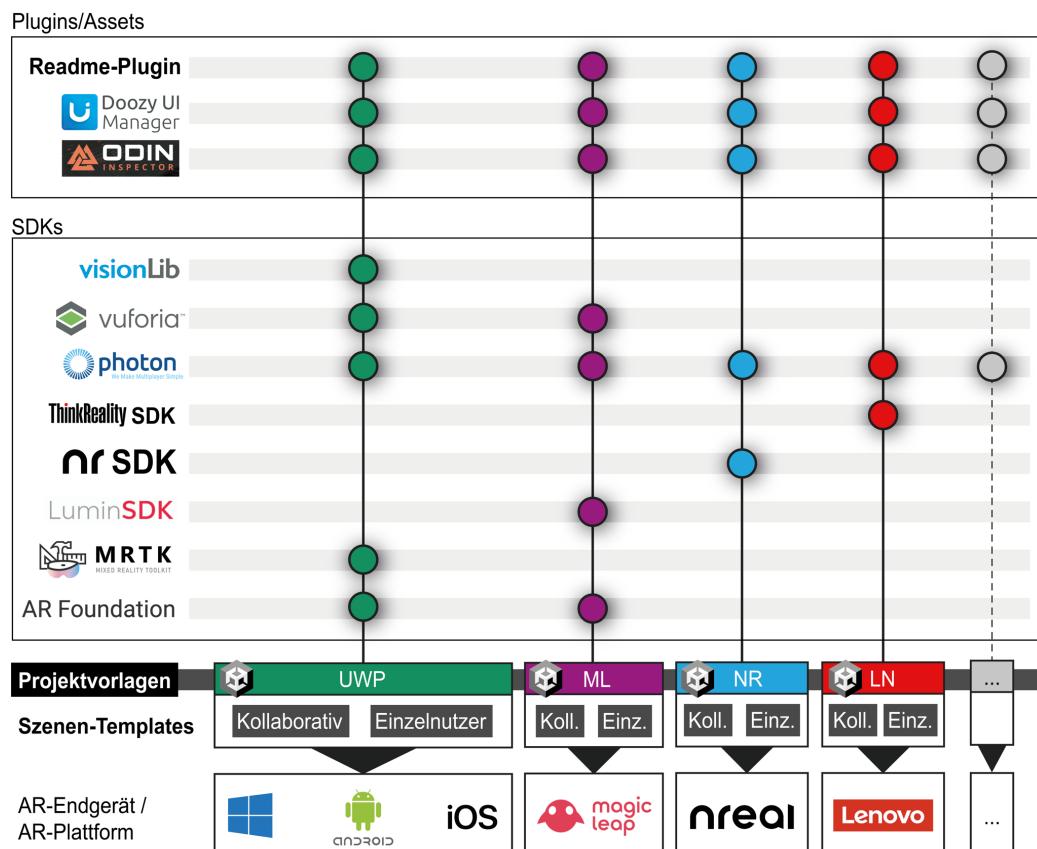


Abbildung 5-34: Übersicht über Projektvorlagen und Szenen-Templates

In jeder Projektvorlage sind zwei **Szenen-Templates** für die grundsätzlichen Arten von AR-Validierungsanwendungen vorgesehen. Die Templates umfassen vorkonfigurierte Szenengraphen, die für die jeweiligen Anwendungen grundlegenden Elemente und Funktionalitäten enthalten. Ein Template adressiert die lokale Einzelnutzer-Validierung auf einem AR-Endgerät. Der Kunde kann sich das Produkt oder Produktmerkmale mit einer entsprechenden AR-Validierungsanwendung bei sich vor Ort alleine anschauen und validieren. Alternativ kann die Validierung bei einem persönlichen Treffen zwischen Entwickler und Kunde erfolgen. Das zweite Template adressiert jeweils die standortübergreifende kollaborative Validierung, bei der zwei oder mehr Personen miteinander verbunden sind. Entsprechende Kommunikations- und Synchronisationsfunktionalitäten sind in der Szene bereits eingerichtet. Zusätzlich sind Teile der Anwendungsfunktionalität bzw. der entsprechenden Funktionsbausteine bereits in Szenen-Templates integriert. Dazu zählen unter anderem Bausteine für das Tracking, die Darstellung verschiedener Varianten oder auch eine virtuelle Laser-Pointer-Funktionalität für den Entwickler.

Eine Besonderheit der kollaborativen Szenen ist, dass die darin entwickelte Anwendung auf verschiedenen Endgeräten installiert werden kann. Über **C#-Präprozessoranweisungen** wird beim Starten der Anwendung überprüft, welches Endgerät vorliegt. Die Abfrage `#if UNITY_IOS` erkennt beispielsweise, dass die Anwendung auf einem iOS-Gerät installiert wurde. Darauf basierend können entsprechende Einstellungen und Anpassungen der Szene automatisch während der Laufzeit vorgenommen werden. So können **unterschiedliche Funktionalitäten und Benutzerschnittstellen für den Entwickler und den Kunden** in einer Szene bereitgestellt werden. Die kollaborative Szene der UWP-Projektvorlage beispielsweise ermöglicht zum einen eine Installation der AR-Validierungsanwendung auf HoloLens-Geräten für den Kunden. Zudem wird eine Installation auf klassischen Windows-Geräten (Desktop oder Surface) als Anwendung für den Entwickler ermöglicht. In diesem Fall werden die AR-Funktionalitäten automatisch deaktiviert und eine spezielle Benutzerschnittstelle für den Entwickler dargestellt.

Die Szenen-Templates werden dem Entwickler beim Erstellen einer neuen Szene zur Auswahl angezeigt. Neben dem Namen und einer Beschreibung des Templates kann eine Grafik zur Veranschaulichung verwendet werden. Wird eine Szene basierend auf einem Template erstellt, enthält die Szene bereits die vorkonfigurierten GameObjects und Funktionalitäten. Zum vereinfachten Verständnis einer Szene enthalten die Szenengraphen leere und ausgeblendete GameObjects als Trennung zwischen einzelnen Aspekten. Abbildung 5-35 zeigt beispielhaft die kollaborative Szene der UWP-Projektvorlage und die optische Unterteilung des Szenengraphen.

Im Bereich MANAGEMENT erfolgt die Kontrolle und Steuerung der Anwendung inklusive der Kollaboration, der Benutzerschnittstellen, der Prototypen, des Trackings sowie der Funktionalitäten. Der Bereich COLLABORATION ist spezifisch für die kollaborativen Szenen-Templates und adressiert unter anderem die Synchronisation der Inhalte zwischen den Kommunikationspartnern. Im MRTK-Bereich liegen grundlegende AR-Objekte. Der CONTENT-Bereich enthält ein GameObject *Content* als Wurzelknoten für alle

virtuellen Inhalte. Alle Prototypen-Modelle sowie erstelltes Feedback etc. werden der Szene unter diesem Knoten hinzugefügt. Das ermöglicht eine automatisierte Verarbeitung und Anbindung an die Anwendungslogik basierend auf zur Verfügung gestellten Bausteinen. Die Bereiche DEVELOPER, CUSTOMER und USER INPUT dienen der Erstellung von Benutzerschnittstellen und Interaktionslösungen für den Entwickler und den Kunden. Im kollaborativen Template sind grundsätzliche Benutzerschnittstellen-Elemente für den Entwickler bereits integriert. Dies betrifft bspw. die Darstellung verschiedener Produktvarianten oder die Aktivierung des virtuellen Laser-Pointers. Je nach eingesetzter Plattform werden entsprechend der Präprozessoranweisungen entweder der Bereich DEVELOPER oder CUSTOMER deaktiviert. Die genauen Funktionalitäten der Elemente und der verwendeten Bausteine werden im folgenden Abschnitt 5.4.3.2 beschrieben.

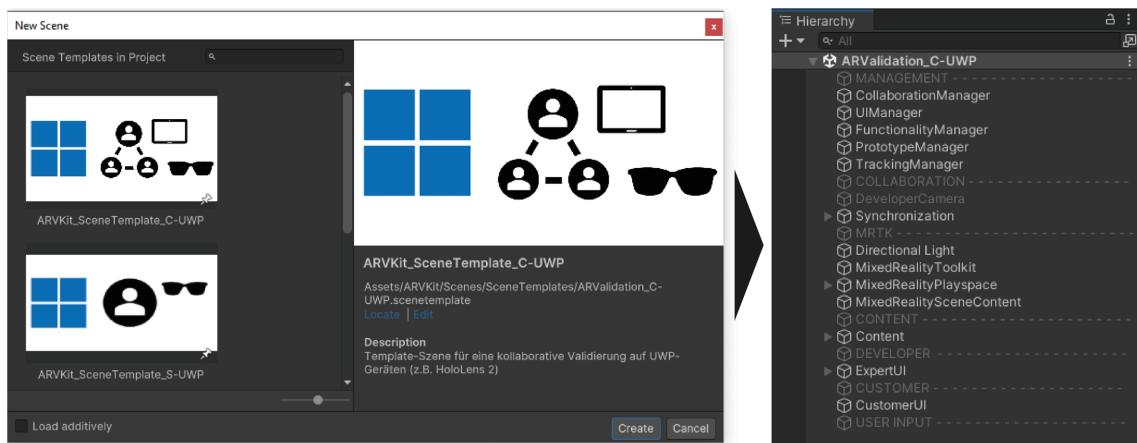


Abbildung 5-35: Erstellung einer neuen Szene basierend auf einem Template

5.4.3.2 Modulare Funktionsbausteine

Den Kern des Werkzeugkoffers und des ARVKit bilden die modularen Funktionsbausteine. Ziel der Bausteine ist es, einzelne für die Validierung relevante Funktionalitäten modular so bereitzustellen, dass sie aufwandsarm auf verschiedenen AR-Endgeräten wiederverwendet werden können. Wie unter anderem im Projekt AcRoSS festgestellt wurde, stellen insbesondere die AR-Benutzerschnittstellen ein Problem bei der Geräte-übergreifenden AR-Anwendungsentwicklung dar. Ein grundlegender Aspekt des Baustein-Konzepts ist daher die klare **Trennung von Anwendungsfunktionalitäten und Benutzerschnittstellen**. Anwendungsfunktionalitäten werden so weit wie möglich basierend auf Unity-integrierten Funktionalitäten sowie plattformunabhängigen SDKs entwickelt. Sie können somit einheitlich in den verschiedenen Projektvorlagen für die verschiedenen AR-Plattformen verwendet werden. Die Benutzerschnittstellen können durch die Entkopplung von den Anwendungsfunktionalitäten problemlos an verschiedene Plattformen angepasst bzw. speziell für die Plattformen erstellt werden. Durch definierte Softwareschnittstellen können die Anwendungsfunktionalitäten und die Benutzerschnittstellen anschließend passend miteinander verbunden und die Anwendungslogik integriert werden. Abbildung 5-36 zeigt das Konzept und die Funktionsweise der modularen Bausteine.

Szenen-Management

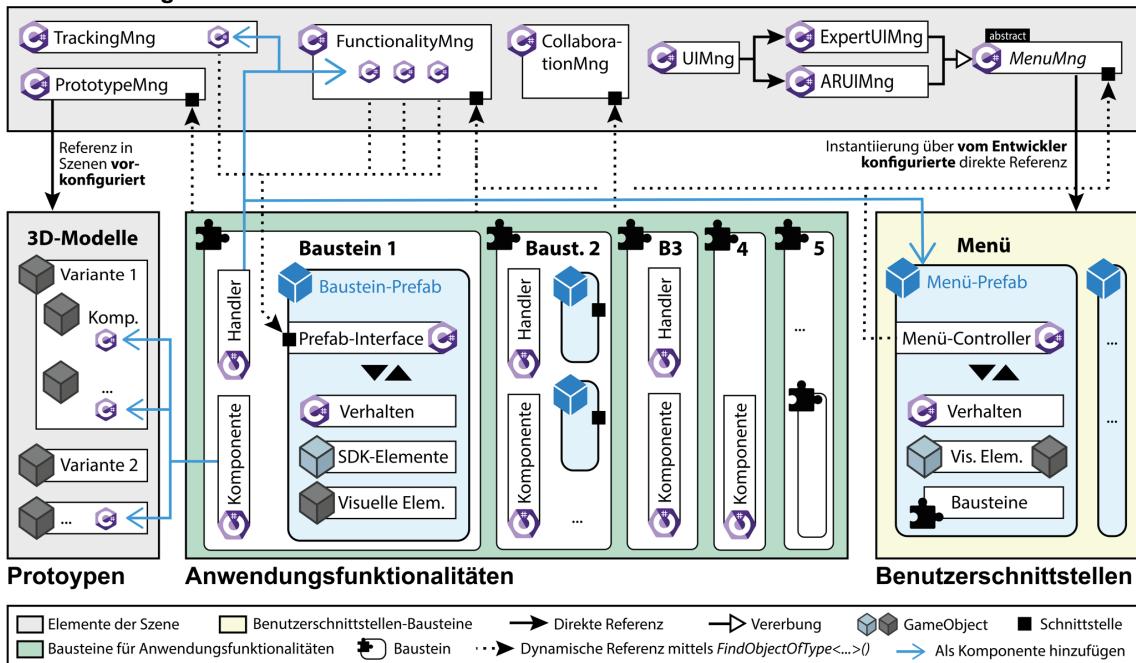


Abbildung 5-36: Das Konzept der modularen Funktionsbausteine in der Übersicht

Die grauen Bereiche entsprechen den in der Szene vorhandenen Objekten, die den Rahmen für den Einsatz der Funktionsbausteine bilden. Für die Integration und Steuerung der Bausteine sind verschiedene übergeordnete Manager im **Szenen-Management** notwendig. Diese sind in den Szenen-Templates in Form von einzelnen Skripten im Bereich MANAGEMENT bereits integriert (vgl. Abbildung 5-35). Zudem enthält die Szene unter dem Content-Knoten hinzugefügte **Prototypen**, deren Basis grundsätzlich 3D-Modelle sind. Bei den Bausteinen wird unterschieden zwischen **Anwendungsfunktionalitäten** (grüner Bereich) und **Benutzerschnittstellen** (gelber Bereich). Neben dieser Unterscheidung stellt der Einsatz dynamischer Referenzierung eine wesentliche Grundlage des Bausteinkonzepts dar. Üblicherweise werden einzelne Szenen-Objekte und ihre Komponenten direkt miteinander referenziert, um eine Kommunikation und einen Datenaustausch zwischen ihnen zu ermöglichen. Das vorliegende Konzept setzt dagegen größtenteils auf **dynamische Referenzierung** mittels der Unity-Funktion `FindObjectOfType<...>()`. Die Funktion ermöglicht die Verbindung der Bausteine mit den Manager-Objekten und umgekehrt während der Laufzeit ohne eine vorherige Kenntnis darüber, welche Anwendungsfunktionen und Benutzerschnittstellen eingesetzt werden. Die Kommunikation zwischen den Benutzerschnittstellen und Anwendungsfunktionalitäten erfolgt ausschließlich über das Szenen-Management. Es besteht keine direkte Verbindung. Dies ermöglicht einen sehr flexiblen und einfacher Austausch verschiedener Benutzerschnittstellen. Eine Voraussetzung dafür ist, dass die Namen der Manager-Objekte feststehen und einheitliche Schnittstellen definiert sind. Dies wird im ARVKit über die Szenen-Templates sowie die Gestaltung der Bausteine gewährleistet.

Bausteine für **Anwendungsfunktionalitäten** entsprechen einer Menge an Skripten und Prefabs. Dabei gibt es verschiedene Möglichkeiten, die in Abbildung 5-36 beispielhaft an fünf Bausteinen angedeutet werden. Bausteine mit visuellen Komponenten enthalten meist ein **Baustein-Prefab**, das aus GameObjects und Skripten zur Verhaltensbeschreibung besteht (Baustein 1). Die GameObjects können aus SDKs verwendet werden oder individuell erstellten visuellen Elementen entsprechen. Die Baustein-Prefabs werden bei Bedarf visuell in die Szene integriert. Für die Anbindung an das Szenen-Management sowie die Benutzerschnittstellen stehen für die meisten Bausteine sogenannte **Handler** zur Verfügung. Sie werden den Manager-Objekten oder Benutzerschnittstellen als Komponenten hinzugefügt und ermöglichen die Steuerung des Verhalten sowie der Darstellung der Baustein-Prefabs. Dazu verfügen die Prefabs über eine fest definierte Schnittstelle, das *Prefab-Interface*. Dieses Interface empfängt Funktionsaufrufe und kommuniziert entsprechend mit den Prefab-internen Skripten und Objekten. Einzelne Funktionalitäten benötigen zudem Skripte, die als **Komponenten** den Prototypen oder Teilmodellen der Prototypen hinzugefügt werden. Grundsätzlich sind die **Baustein-Prefabs**, **-Handler** und **-Komponenten** optional und auf unterschiedliche Art und Weise miteinander kombinierbar. Ein Baustein kann beispielsweise mehrere Prefabs enthalten (Baustein 2) oder lediglich auf einem *Handler* und einer *Komponente* basieren (Baustein 3). Ebenso sind Bausteine möglich, die ohne *Handler* funktionieren (Baustein 4) oder ganze Bausteine wiederverwenden (Baustein 5). Abbildung 5-37 zeigt eine Übersicht der bisher im Konzept des ARVKit enthaltenen Funktionsbausteine unterteilt in die Kategorien Kollaboration, Prototypen, Tracking, Validierungsfunktionalität und User Interface.

Funktionsbausteine:	Prototypen	Validierungsfunktionalität	User Interface
Tracking Kollaboration			
Videoübertragung	Bewegungstracking	Verschieben	Pointer
Audiokommunikation	Umgebungsrekonstruktion	Ein-/Ausblenden	Meta-Infos
Synchronisation	Lokaler Prototyp	IT-Anbindung	MenuController
Live-Streaming	Cloud-Prototyp	Feedback	SpeechToText
2D-Marker-Tracking	Variantenwechsel	Telemetrie	Gestensteuerung
3D-Objekt-Tracking	TUI	Highlight	Sprachsteuerung
Freie Positionierung	Audio-Player	Anforderungsdarst.	Headtracking
Statische Position.	Animationsdarstellung	Röntgenblick	Eye Tracking
	Simulationsdatenvisual.	Skalieren	Drag & Drop
	Funktionswiedergabe	Teilansichten	

Abbildung 5-37: Übersicht der durch die Bausteine adressierten Funktionalitäten

Bausteine für **Benutzerschnittstellen** werden als **Menü-Prefabs** in den Projekten erstellt. Sie basieren auf visuellen und interaktiven Interaktionselementen aus Unity oder speziellen SDKs sowie Skripten zur Beschreibung des Interaktionsverhaltens. Zudem können einzelne User Interface-Bausteine in Menüs verwendet werden. Ein *Menü-Controller* dient jeweils als Schnittstelle in das Szenen-Management und zur Anbindung an die Anwendungsfunktionalitäten.

Im Folgenden werden die verschiedenen Aspekte der Funktionsbausteine (s. Abbildung 5-37) und ihre **Verwendung und Funktionsweise** in Kombination mit den Benutzerschnittstellen genauer erläutert:

- **Kollaboration:** Wie die Problemanalyse zeigt, ist eine standortübergreifende Kollaboration in vielen Validierungsszenarien von entscheidender Bedeutung (Kapitel 3.3). Dies wird unter anderem durch Untersuchungen von PAES UND IRIZARRY bestätigt (Anhang A2.3.5). Entsprechend wurden im Rahmen des ARVKit verschiedene Bausteine für die Realisierung kollaborativer AR-Anwendungen konzipiert und realisiert. Aufgrund der guten Plattform-Unterstützung sowie der Verfügbarkeit von Tutorials und einer großen Entwickler-Community wurde *Photon PUN* (Anhang A2.3.3) als Netzwerk-SDK ausgewählt. Die Bausteine adressieren eine *Videoübertragung* und *Audio-Kommunikation* zwischen den Kommunikationspartnern sowie die *Synchronisation* und *Live-Streaming*. *Synchronisation* und *Live-Streaming* sind hilfreich für die gemeinsame Betrachtung von Prototypen. Die Bausteine sind bereits in den kollaborativen Szenen-Templates in Form des **Kollaborationsmanagers** (*CollaborationMng*) integriert. Über das Konfigurationsmenü als grafische Benutzerschnittstelle (s. Abschnitt 5.4.3.3) kann die Kollaboration konfiguriert werden. Zum einen kann eine Projekt-ID angegeben werden, die als standardmäßige ID zur **Vernetzung der Kommunikationspartner** verwendet wird. Zusätzlich können einzelne Funktionalitäten wie die *Audio-Kommunikation* aktiviert bzw. deaktiviert werden. Je nach Anforderung und Bedarf kann die *Synchronisation* der Perspektive sowie der Positionierung der Prototypen konfiguriert werden. Eine Möglichkeit ist, dass der Kunde die Kontrolle über die Position der Prototypen hat, beispielsweise basierend auf einer Tracking-Lösung. Andererseits kann der Entwickler die Positionierung aus der Distanz vornehmen. Die Position wird jeweils entsprechend automatisiert und live beim anderen angepasst. Als Alternative kann es sinnvoll sein, dass Kunde und Entwickler die Inhalte jeweils unabhängig voneinander positionieren können.
- **Prototypen:** Als Datentyp für die 3D-Modelle der Prototypen wird entsprechend der Empfehlung von Unity das FBX-Dateiformat genutzt (s. Anhang A2.3.2). Es wird von vielen Modellierungs- und Konstruktionswerkzeugen als Exportformat unterstützt oder lässt sich alternativ durch das kostenlose Werkzeug Blender konvertieren. Einzelne Prototypen können hierarchisch aufgebaut sein und aus Teilmustern bestehen. FBX ermöglicht zudem die Integration von Animationen und weiteren Datenaspekten zum direkten Import in Unity. Zur Integration der Prototypen gibt es verschiedene Möglichkeiten. Grundsätzlich werden die Prototypen-Modelle unter dem *Content*-Knoten eingebunden. Zum einen lassen sich Prototypen direkt in Unity in die Anwendung einbinden. Alternativ stehen Bausteine zum *lokalen Import* vom AR-Gerät sowie zum Laden von Dateien aus einer *Cloud*-Umgebung bereit. Das Management der Prototypen erfolgt über den **Prototypenmanager** (*PrototypeMng*), der eine direkte Referenz auf das *Content*-Objekt und alle darunter hinzugefügten Prototypen hat. Der Prototypenmanager verfügt unter anderem über den bereits integrierten Baustein zur *Darstellung von Varianten*. Alle unter dem *Content*-Knoten hinzugefügten Prototypen werden automatisch als Varianten angesehen. Die Varianten können durch entsprechende Benutzerschnittstellen angezeigt und ausgeblendet und

so dem Kunden verschiedene Produktmöglichkeiten oder Merkmalsausprägungen präsentiert werden. Jedes Teilmodell der einzelnen Varianten wird beim Starten der Anwendung durch den Prototypenmanager automatisch mit Komponenten versehen, die es auf dem jeweils verwendeten Gerät per Klick auswählbar machen. Das Auswählen dient später als Auslöser zum Anzeigen des Hauptmenüs, das die verschiedenen Funktionalitäten bereitstellt. Die Integration der weiteren für die Prototypen relevanten Datenaspekte wird über spezifische Bausteine adressiert. Die Baustein-Komponenten können den Teilmodellen der Prototypen hinzugefügt und entsprechend mit den jeweiligen Datenaspekten versehen werden. Die Datenaspekte können später in der Anwendung durch die Funktionsbausteine verwendet werden. So können beispielsweise sowohl bereits im FBX-Modell enthaltene als auch nachträglich in Unity erstellte Animationen über die *AnimatorHandler*-Komponente in die Anwendung eingebunden werden. Die Integration von Materialien in die Validierung erfolgt über Tangible User Interfaces (TUI). Einzelnen Modellen kann die Komponente *TangibleUserInterface* des TUI-Bausteins hinzugefügt werden. Diese Modelle werden dadurch nur auf der Expertenseite dargestellt, da sie beim Kunden physisch vorliegen. Entsprechend der verwendeten weiteren Bausteine mit Validierungsfunktionalitäten werden den Teilmodellen zudem bei Bedarf die in den Bausteinen enthaltenen notwendigen Komponenten hinzugefügt.

- **Tracking:** Für den Aspekt Tracking sind sechs grundsätzliche Bausteine vorgesehen. Ein *2D-Marker-Tracking*-Baustein wurde bereits realisiert. Als alternative Positionierungslösungen stehen *3D-Modell-Tracking*, *freie Positionierung* sowie eine *statische Positionierung* an einer festen Stelle im Raum zur Verfügung. *2D-Marker-Tracking* wird aufgrund der guten Platform-Unterstützung über das *Vuforia SDK* realisiert. *Vuforia* ermöglicht zudem *3D-Objekt-Tracking*. In der UWP-Projektvorlage ermöglicht ein auf *visionLib* basierender Baustein zudem ein *3D-Objekt-Tracking* basierend auf CAD-Daten. Auf den nicht-unterstützten Plattformen, beispielsweise nreal und Lenovo, werden individuelle Tracking-Bausteine angeboten. Hierzu stellen die gerätespezifischen SDKs *nrSDK* sowie *ThinkReality SDK* Funktionalitäten bereit. Als Grundlage für weitere Anwendungsfunktionalitäten dienen die Bausteine *Bewegungstracking* sowie *Umgebungsrekonstruktion*. Die Bausteine werden über den **Trackingmanager** (*TrackingMng*) verwaltet und in der Szene integriert. Durch den Trackingmanager hinzugefügten Baustein-*Handle*n kann eine Kommunikation mit den Tracking-Prefabs erfolgen.
- **Validierungsfunktionalitäten:** Die Validierungsfunktionalitäten werden über den **Funktionalitätsmanager** (*FunctionalityMng*) integriert und gesteuert. Dem Funktionalitätsmanager sind alle zur Verfügung stehenden Bausteine bekannt. Über das Konfigurationsmenü als grafische Benutzeroberfläche (s. Abschnitt 5.4.3.3) kann der Entwickler auswählen, welche Validierungsfunktionalitäten zum Einsatz kommen sollen. Bei kollaborativen Validierungsanwendungen kann die Konfiguration individuell für Entwickler und Kunde erfolgen. Beispielsweise kann der Entwickler die

Möglichkeit bekommen, zwischen Varianten zu wechseln, und der Kunde nicht. Wird eine Funktionalität aktiviert, wird der entsprechende Baustein-*Handler*, falls vorhanden, automatisch dem Tracking- oder Funktionalitätsmanager hinzugefügt. Zudem werden bei Bedarf die Prototypen mit Baustein-*Komponenten* versehen. Gleichzeitig werden dem Baustein entsprechende Konfigurationsmöglichkeiten für die Benutzerschnittstellen aktiviert. Das aktuelle Konzept des ARVKit umfasst Bausteine zu 15 Validierungsfunktionalitäten. Die Funktionalitäten adressieren unter anderem die Visualisierung der Prototypen, beispielsweise durch *Skalieren*, *Verschieben*, *Ein- und Ausblenden* oder einen *Röntgenblick* ins Innere. Einen sehr wichtigen Baustein für die Validierung stellen laut Stand der Technik zudem *Animationen* dar (Anhang A2.3.5). Die Erstellung und Dokumentation von *Feedback* sind ebenfalls von großer Bedeutung. Der *Feedback*-Baustein wird im weiteren Verlauf dieses Abschnitts genauer vorgestellt. Der *Pointer*-Baustein bietet eine standortübergreifende Funktionalität ähnlich einem Laser-Pointer. Der Baustein ist inspiriert durch die Arbeiten von BRUNO ET AL. (Abschnitt 4.3.3.3). Je nach Funktionskonfiguration müssen die für die Funktionalitäten zu verwendenden Menü-Prefabs in den entsprechenden UI-Managern konfiguriert werden.

- **User Interface:** Die übergreifende Koordination der Benutzerschnittstellen sowie der Benutzerinteraktion erfolgt über den **UI-Manager** (*UIMng*). Entsprechend der zuvor beschriebenen Verwendung der kollaborativen Szenen sowohl für den Kunden als auch den Experten enthalten diese Szenen sowohl den **AR-UI-Manager** (*ARUIMng*) als auch den **Experten-UI-Manager** (*ExpertUIMng*). Über die beiden Manager erfolgt die Konfiguration der individuellen Benutzerschnittstellen für die Funktionsbausteine. Zudem agieren die Manager als Knoten, an dem die jeweiligen Menüs angehängt werden. Je nach eingesetztem Endgerät wird der nicht benötigte Manager beim Starten der Validierungsanwendung durch den UI-Manager automatisch deaktiviert. Die beiden Manager sind von der **abstrakten Klasse** **MenuMng** abgeleitet. Diese abstrakte Klasse beschreibt alle relevanten Schnittstellen zur Benutzerinteraktion sowie zum Anzeigen und Ausblenden von Menüs. Die Schnittstellen werden von den beiden Managern individuell realisiert. Dank der abstrakten Klasse muss bei der Funktionalität der Menüs sowie bei der Anbindung der Validierungsfunktionalitäten an die Menüs nicht zwischen unterschiedlichen Plattformen unterschieden werden. Durch die Funktion *FindObjectOfType<MenuMng>()* wird der jeweils vorhandene Manager automatisch referenziert und die jeweiligen Funktionsaufrufe über diesen ausgeführt. Eine entsprechende Funktionalität wird über den *MenuController*-Baustein bereitgestellt. Die Koordination der nicht dauerhaft angezeigten Menüs erfolgt komplett **Prefab-basiert** über die *MenuMng*-Klasse. Es kann jeweils nur ein Menü aktiv sein. Beim Öffnen eines neuen Menüs wird das aktuell angezeigte Menü automatisch geschlossen. Dies ermöglicht eine flexible Integration und Anpassungen von individuellen Menüs, ohne die umfangreiche Anpassung einer komplizierten Menü-Logik. Die einzelnen zu verwendenden Menü-Prefabs werden über die beiden Menü-Manager konfiguriert. Ein Start-Menü und ein Hauptmenü

müssen immer vorhanden sein. Abhängig von der Konfiguration der Validierungsfunktionalitäten sind zudem weitere Menü-Prefabs zu erstellen und über den jeweiligen Manager zu referenzieren. Wird eine Funktionalität beispielsweise über das Hauptmenü aufgerufen, wird das aktuelle Menü geschlossen und das entsprechende Menü-Prefab in der Szene instanziert und dargestellt. Zur Erstellung von Menüs können neben den standardmäßigen Visualisierungs- und Interaktionselementen verschiedene Bausteine des ARVKit verwendet werden. So stehen unter anderem *Gesten-* und *Sprachsteuerungsfunktionalitäten* bereit. Sprache kann auch zur Eingabe von Text basierend auf dem *SpeechToText*-Baustein verwendet werden. *Head-* und *Eye-Tracking* ermöglichen innovative Auswertungen im Kontext der Validierung. *Drag & Drop* kann für die Manipulation von und Interaktion mit virtuellen Inhalten hilfreich sein. Über eine dynamische Referenzierung erfolgt entsprechend die Kommunikation der Menüs mit dem Funktionalitätsmanager in der Szene. Über die jeweiligen Baustein-Handler erfolgt dann die Steuerung der Funktionalitäten in den Bausteinen.

Im Folgenden wird eine Auswahl an Bausteinen exemplarisch im Detail vorgestellt, um das Baustein-Konzept zu veranschaulichen. Die ausgewählten Bausteine werden im Kontext des Feedback-Bausteins verwendet.

Der **Drag-and-Drop-Baustein** setzt sich aus einem *GestureDragAndDropHandler* und einer abstrakten Klasse *DragAndDrop* zusammen (Abbildung 5-38, links). Die *DragAndDrop*-Klasse definiert die Funktionsaufrufe *Startpoint*, *Drag* und *Drop* als Schnittstellen. Von der abstrakten Klasse lassen sich individuelle *DragAndDrop*-Skripte ableiten, die ein konkretes Verhalten für die Schnittstellenfunktionen implementieren. Die individuellen Skripte werden als Komponenten einzelnen Objekten hinzugefügt und führen bei Aufruf der Schnittstellenfunktionen das entsprechende Verhalten aus. Ein Beispiel ist die Komponente *DragAndDropArrow*, die auf dreidimensionale Pfeile angewendet werden kann und im *Feedback*-Baustein verwendet wird. Der *GestureDragAndDropHandler* wird in Menü-Prefabs verwendet. Er erhält eine Referenz auf das Zielobjekt (*target*), auf das die *Drag-and-Drop*-Funktionalität angewendet werden soll. Der Handler reagiert auf die Gesten-Ereignisse *OnPointerDown*, *OnPointerUp* und *OnPointerDragged*. Tritt eines der Ereignisse auf, wird die entsprechende Schnittstellenfunktion der *DragAndDrop*-Komponente des referenzierten Zielobjekts aufgerufen und so die *Drag-and-Drop*-Funktionalität auf dieses Objekt angewendet.

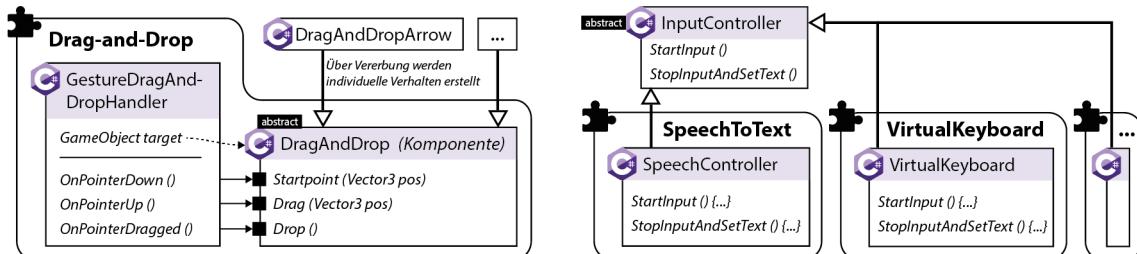


Abbildung 5-38: Exemplarische Bausteine für den Aspekt User-Interface: *Drag-and-Drop* (links) und verschiedene Texteingabe-Funktionalitäten (rechts)

Für die **Eingabe von Text** stellt das ARVKit verschiedene von der abstrakten Klasse *InputController* abgeleitete Bausteine bereit (Abbildung 5-38, rechts). Als Schnittstellenfunktionen sind *StartInput* und *StopInputAndSetText* definiert. Der Baustein **SpeechToText** implementiert diese Funktionen beispielsweise basierend auf einer Speech-to-Text-Funktionalität, die über das *SpeechController*-Skript realisiert wird. Der **VirtualKeyboard-Baustein** dagegen nutzt eine virtuelle Tastatur für die Eingabe von Text. Die verschiedenen von *InputController* abgeleiteten Bausteine zur Eingabe von Text können durch die einheitliche Schnittstellendefinition einfach und flexibel ausgetauscht werden.

Unter anderem inspiriert durch die Arbeiten von CHANG ET AL. (Abschnitt 4.3.3.3) wurden verschiedene Feedback-Möglichkeiten entwickelt. Feedback durch den Kunden zu einzelnen Produktmerkmalen ist von höchster Bedeutung für die Validierung. Beim Einsatz von AR besteht dabei das Potential, das Feedback dreidimensional am Produkt bzw. an den Prototypen zu verorten. Basierend auf der Problemanalyse und dem Stand der Technik wurden daher zunächst **dreidimensionale Text-Annotationen** sowie **3D-Pfeile** als Feedback-Komponenten konzipiert und im Baustein realisiert. Beide nutzen die Potentiale der dreidimensionalen Visualisierung aus. Abbildung 5-39 gibt einen Überblick über den Feedback-Baustein und seine Komponenten. Die Grundlage des Feedbacks bilden je ein Prefab für die Text-Annotationen sowie für die 3D-Pfeile. Text-Annotationen werden über die im *MRTK* bereitgestellten *Tooltips* realisiert. Als Prefab-Interface dient das *Annotation*-Skript. Es stellt Funktionen zum Setzen des Textes sowie der Position des Textlabels bereit. Über die *SetTarget*-Funktion kann zudem das Zielobjekt der Annotation definiert werden. Die Linie der Annotation wird automatisch zum definierten Zielobjekt ausgerichtet. Die *Tooltips* werden zudem mit einem *Billboard*-Skript versehen, durch das das Text-Label stets zum Betrachter ausgerichtet wird. Über eine *NearInteractionGrabbable*-Komponente des *MRTK* wird realisiert, dass das Textlabel gegriffen und verschoben werden kann. Das Prefab für den 3D-Pfeil ist ähnlich aufgebaut. Es basiert auf einem eigens für den Baustein erstellten Pfeil-Modell und nutzt das *Arrow*-Skript als Prefab-Interface. Über das Skript können die Position der Pfeilspitze und des Pfeilendes gesetzt sowie das Material bzw. die Farbe verändert werden. Über eine *NearInteractionGrabbable*-Komponente kann der gesamte Pfeil gegriffen und bewegt werden. Zudem verfügen Pfeile über die zuvor beschriebene *DragAndDropArrow*-Komponente des *Drag-And-Drop*-Bausteins (*DaD*). Die Komponente ermöglicht das Setzen der Pfeilspitze sowie des Pfeilendes basierend auf Drag-and-Drop-Input. Neben den Prefabs enthält der Feedback-Baustein den **FeedbackHandler**. Dieser wird entsprechend des zuvor beschriebenen Baustein-Konzepts dem Funktionalitätsmanager der Szene hinzugefügt. Der *FeedbackHandler* steuert die Abgabe von Feedback und speichert alle Annotationen und Pfeile in Listen. Er stellt zudem Schnittstellenfunktionen zur Erstellung, Anpassung und Löschung von Annotationen und Pfeilen bereit. Bei Erstellung von Annotationen oder Pfeilen werden entsprechend die Prefabs initialisiert, in der Szene hinzugefügt und eine Referenz in der jeweiligen Liste gespeichert. Zudem werden die Feedback-Komponenten über die Schnittstellenfunktionen der Prefabs konfiguriert. Bei Löschung oder Anpassung von Feedback erfolgt der Zugriff auf die jeweiligen Objekte über die Listen.

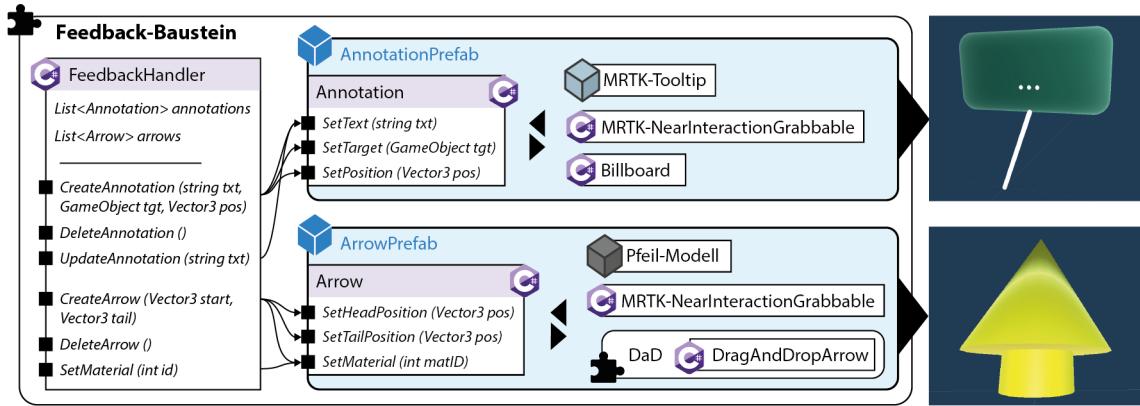


Abbildung 5-39: Der Funktionsbaustein „Feedback“ inklusive der für die Entwickler relevanten Komponenten und Schnittstellen

Der Feedback-Baustein lässt sich auf verschiedenen AR-Plattformen nutzen. Für die Plattformen sind jeweils individuelle Benutzerschnittstellen zu verwenden. Die Benutzerschnittstellen können beliebig gestaltet sein. Sie kommunizieren über die dynamische Referenzierung mit dem *Funktionalitätsmanager* und nutzen die Schnittstellenfunktionen des entsprechenden *FeedbackHandlers* zur Erstellung und Anpassung von Annotationen und Pfeilen. Im Rahmen der vorliegenden Arbeit wurden bereits verschiedene Menü-Prefabs für die HoloLens 2 realisiert. Diese Prefabs können in der UWP-Projektvorlage entweder direkt oder alternativ als Beispiele zur Erstellung individueller Menü-Prefabs verwendet werden. Zunächst enthält der Werkzeugkasten bereits ein **Startmenü**- sowie **ein Hauptmenü-Prefab**. Über das Startmenü können eine Projekt-ID eingegeben und die Validierung gestartet werden. Zur Eingabe der Projekt-ID wird das *VirtualKeyboard* (s. Abbildung 5-38) verwendet. Das Hauptmenü soll wie bereits beschrieben bei Auswählen eines Prototypenteilmodells erscheinen. Abhängig von der Funktionskonfiguration im Funktionalitätsmanager sowie der Verfügbarkeit einzelner Funktionen (bspw. Animationen oder Meta-Infos) zum Teilmodell werden die Elemente des Hauptmenüs automatisch angepasst.

Zudem wurden bereits die für den Einsatz des Feedback-Funktionsbausteins relevanten Menü-Prefabs exemplarisch realisiert. Abbildung 5-40 zeigt die vier **Feedback-Menü-Prefabs** inklusive ihrer Komponenten.

Alle exemplarisch realisierten HoloLens 2-Menü-Prefabs basieren auf *NearMenus* des MRTK. *NearMenus* schweben frei vor dem Benutzer und können per Finger-Gesten wie reale Knöpfe benutzt werden. Zudem nutzt jedes Menü den Baustein *MenuControl*, der die Kommunikation mit dem Menu-Manager und das Öffnen und Schließen von Menüs bereitstellt. Das *HL2FeedbackMenu* dient der Auswahl zwischen einem 3D-Pfeil und einer Annotation. Für die Erstellung von Annotationen steht das *HL2AnnotationMenu* bereit. Die Erstellung wird durch die *AnnotationCreator*-Komponente gesteuert. Durch Klicken auf einen Aufnehmen-Button wird der zuvor beschriebene und im Prefab enthaltene *SpeechToText*-Baustein verwendet, um Gesprochenes in Text zu überführen und als In-

halt der Annotation zu setzen. Zudem können alle Annotationen, die bisher zum ausgewählten Prototypen-Teilmodell erstellt wurden, gelöscht werden. Die Erstellung von 3D-Pfeilen kann über das *HL2ArrowMenu* und der darin enthaltenen *ArrowCreator*-Komponente erfolgen. Das Prefab nutzt den *Handler* des *Drag-and-Drop*-Bausteins als Gegenstück zur *Drag-and-Drop*-Komponente *DragAndDropArrow*, die im Pfeil-Prefab enthalten ist (s. Abbildung 5-39). So wird die Ausführung von *Drag-and-Drop*-Gesten genutzt, um 3D-Pfeile frei in der Luft zu erstellen. Beim ersten Klick wird die Pfeilspitze gesetzt. Anschließend kann bei gehaltener Klick-Geste das Pfeil-Ende beliebig positioniert werden. Das *HL2ArrowMenu* ermöglicht zudem eine Farbauswahl und das Speichern des erstellten Pfeils. Das *HL2ArrowDeleteMenu* stellt lediglich einen Button zum Löschen eines 3D-Pfeils bereit.

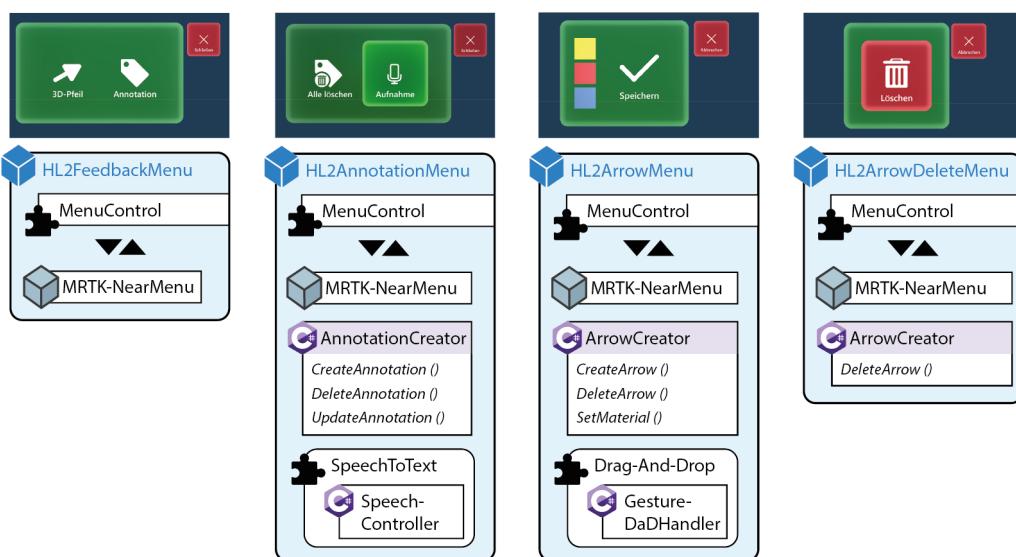


Abbildung 5-40: Vordefinierte Menüs zum Einsatz des Feedback-Funktionsbausteins auf der HoloLens2 mittels Gesten und Sprache

5.4.3.3 Hilfsmittel

Die Anleitung der Entwicklung durch den Leitfaden wird durch zusätzliche Hilfsmittel direkt in Unity unterstützt. Beim Öffnen einer Projektvorlage wird zunächst ein **Readme** im Inspector angezeigt (Abbildung 5-41). Das Readme beschreibt das ARVKit im Allgemeinen und grob den Prozess der Entwicklung individueller AR-Validierungsanwendungen. Weitere Readmes sind in den konkreten Szenen-Templates vorgesehen, um die Entwicklung im Detail zu beschreiben. Die Anleitung entspricht dabei grundsätzlich den im Leitfaden beschriebenen Entwicklungsschritten innerhalb einer Szene. Readmes haben den Vorteil, dass sie direkt in der Entwicklungsumgebung angezeigt werden und ein Wechsel zwischen verschiedenen Fenstern vermieden werden kann.



Abbildung 5-41: *Readme zum Einsatz des ARVKit*

Das bereits im vorherigen Abschnitt beschriebene **Konfigurationsmenü** wurde individuell für die Entwicklung von AR-Validierungsanwendungen basierend auf dem ARVKit gestaltet. Es lässt sich über die Unity-Menüleiste öffnen und beliebig in das modulare Layout von Unity integrieren (Abbildung 5-42). Das Menü bündelt die wesentlichen Konfigurationsmöglichkeiten der verschiedenen Manager in einer grafischen Benutzeroberfläche. Unter anderem kann das Tracking aufwandsarm über Button konfiguriert werden. Zudem werden die einzelnen Manager-Objekte über individuelle Button verlinkt. Beim Klicken auf einen Button wird das entsprechende Objekt in der Szene ausgewählt und die Konfigurationsmöglichkeiten im Inspector dargestellt. Im Inspector können detaillierte Einstellungen vorgenommen werden. Über das Konfigurationsmenü können unter anderem auch die zu verwendenden Funktionalitäten für den Entwickler und den Kunden aktiviert werden. Bei Aktivierung werden automatisch die zu setzenden Referenzen für die Menü-Prefabs als Benutzerschnittstellen aktiviert. Nicht verwendete Einstellungsmöglichkeiten werden entsprechend ausgeblendet. So wird die grafische Benutzeroberfläche auf das Nötigste reduziert. Dies erfolgt zudem bei allen Inspector-Ansichten zu einzelnen Komponenten der verschiedenen in den Szenen oder den Bausteinen verwendeten Objekten. Dazu wird das Plugin **Odin Inspektor** eingesetzt. Alle für den Entwickler nicht relevanten Aspekte werden ausgeblendet. Durch Gruppierungen und ausklappbare Menübereiche erfolgt zudem eine sinnvolle Strukturierung der Konfigurationsmöglichkeiten. Insgesamt kann so die Komplexität der Entwicklung stark reduziert werden.

Zusätzliche Informationen zu einzelnen Objekten und deren Komponenten werden über sogenannte **Unity-Tooltips** angezeigt (Abbildung 5-43, links). Tooltips erscheinen, wenn die Maus länger auf einer Komponente verweilt. Bei Unklarheiten über eine spezielle Komponente können entsprechend hilfreiche Informationen über Tooltips gewonnen

werden. Verschiedene Funktionalitäten zur Konfiguration und inhaltlichen Gestaltung einer Szene werden zudem über ein spezielles **ARVKit-Kontextmenü** angeboten (Abbildung 5-43, rechts). Das Menü erscheint, wenn ein Rechtsklick auf einem Objekt ausgeführt wird. Prototypen-Objekten können über das Kontext-Menü beispielsweise Komponenten zur Ergänzung weiterer Datenaspekte aus den jeweiligen Funktionsbausteinen hinzugefügt werden. Die Erstellung von Objekten sowie das Hinzufügen von Funktions-Komponenten wird für den Entwickler somit stark vereinfacht.

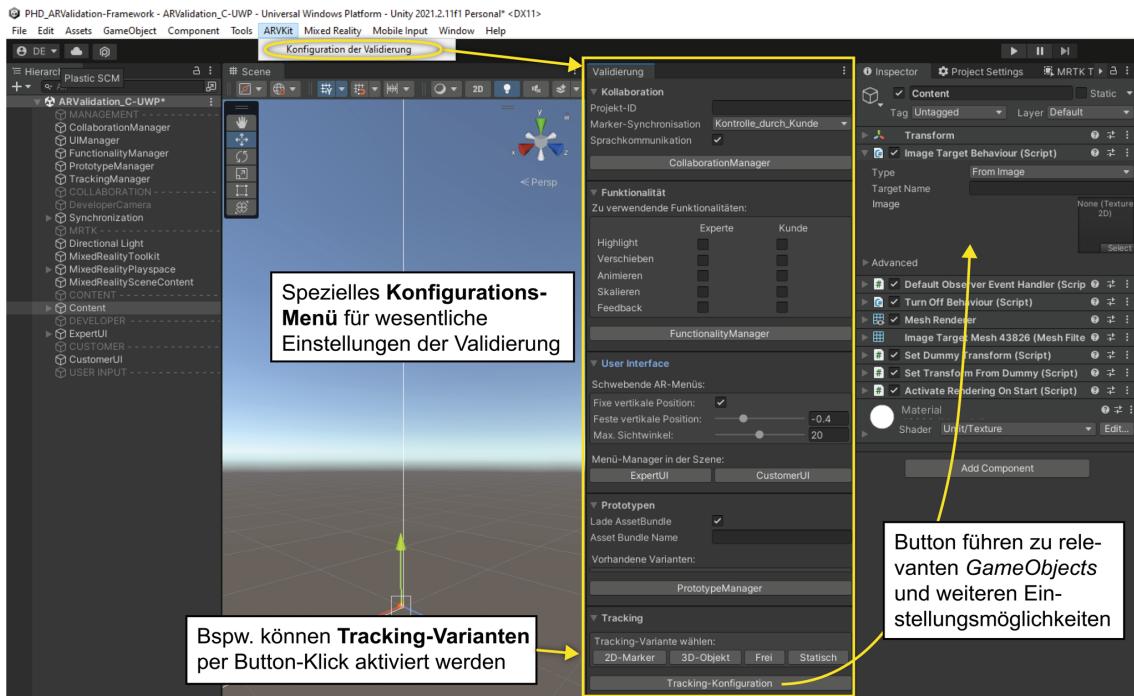


Abbildung 5-42: Konfigurationsmenü mit wesentlichen Einstellungsmöglichkeiten und Referenzierungen relevanter GameObjects in der Szene

Durch die verschiedenen Elemente des ARVKit und den darauf aufbauenden Leitfaden zur Anleitung der Entwicklung versetzt der Werkzeugkoffer Unternehmen auch ohne spezielles Knowhow in die Lage, individuelle AR-Validierungsanwendungen eigenständig in Unity zu entwickeln und auf bedarfsgerechten AR-Endgeräten zu installieren. Unternehmen werden befähigt, die AR-basierte Validierung technisch vorzubereiten.

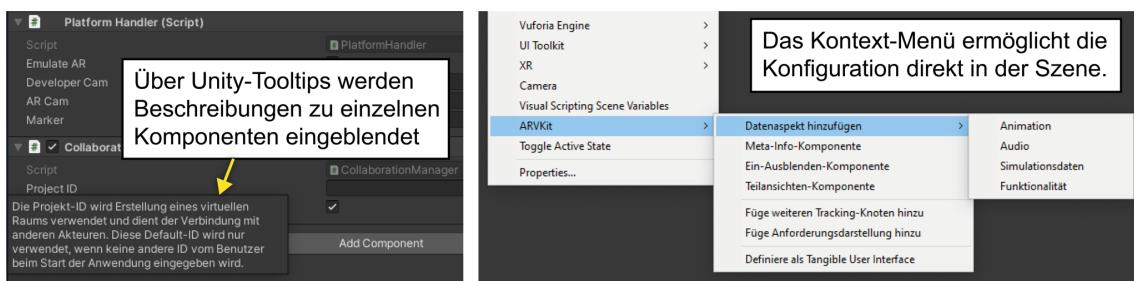


Abbildung 5-43: Beispielhafter Unity-Tooltip (links) sowie das Kontext-Menü (rechts)

5.5 Vorgehensmodell

Die drei zuvor beschriebenen Lösungsbausteine bilden den Kern des entwickelten Instrumentariums. Die Lösungsbausteine stellen verschiedene Hilfsmittel bereit und unterstützen Unternehmen somit bei Potentialbewertung, der AR-System-Konfiguration sowie der Anwendungsentwicklung im Kontext der Produktvalidierung. Für jede Phase wird ein konkretes Vorgehen zum Einsatz der Hilfsmittel und zur Durchführung der notwendigen Tätigkeiten beschrieben. Vervollständigt wird das Instrumentarium durch das übergreifende Vorgehensmodell. Das Vorgehensmodell bildet die Klammer um die drei anderen Lösungsbausteine und beschreibt ihre Verwendung zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung. Es wird in Form eines **Phasen-Meilenstein-Diagramms** bereitgestellt (Abbildung 5-44). Die folgenden Phasen sind bei der Planung und Vorbereitung der AR-basierten Validierung zu durchlaufen:

- 1) **Analyse der zu validierenden Produktanforderungen:** Zu Beginn des Einsatzes des Instrumentariums liegen die möglicherweise mittels AR zu validierenden Anforderungen an das Produkt vor. Anhand der Systematik zur Nutzen-Aufwands-Bewertung werden die Anforderungen generalisiert und die verfügbaren sowie gewünschten Detailgrade der Datenaspekte definiert. Basierend auf ersten Analysen können für die Validierung mittels AR vielversprechende Anforderungen ausgewählt und zusätzliche Auswertungen durchgeführt werden. Ergebnis der ersten Phase ist anschließend eine Nutzen-Aufwands-Bewertung der AR-Potentiale.
- 2) **Entscheidung bzgl. AR als Validierungswerkzeug:** Die Nutzen-Aufwands-Bewertung dient als Grundlage für die Entscheidungsfindung. Sie gibt Auskunft über den potentiellen Nutzen des Einsatzes von AR sowie die dafür notwendigen Aufwände. Unter Berücksichtigung individueller Rahmenbedingungen wird eine Entscheidung für oder gegen AR als Validierungswerkzeug getroffen. Zudem werden die im Rahmen der Validierung zu nutzenden Potentiale von AR definiert. Sie dienen als Grundlage für die anschließende AR-System-Konfiguration.
- 3) **AR-System-Konfiguration:** Basierend auf Lösungsbaustein 2 erfolgt die AR-System-Konfiguration. Input für die Konfiguration sind die zu nutzenden AR-Potentiale sowie individuelle Rahmenbedingungen und Anforderungen an die Validierung. Die vom Konfigurations-Werkzeug ermittelten zu nutzenden Softwarefunktionalitäten können bei Bedarf ergänzt werden. Basierend auf der Auswertung der AR-System-Wissensbasis kann ein geeignetes AR-Endgerät ausgewählt werden. Ergebnis ist bedarfsgerechte AR-System-Konfiguration.
- 4) **Entwicklung der AR-Validierungsanwendung:** Basierend auf der AR-System-Konfiguration wird die Validierungsanwendung für das ausgewählte AR-Endgerät in Unity entwickelt. Die Entwicklung wird durch den Werkzeugkoffer und das darin enthaltene ARVKit unterstützt und vereinfacht. Die Anwendung kann anschließend auf dem AR-Endgerät installiert und bereitgestellt werden. Ergebnis der vierten Phase ist somit eine individuelle AR-Lösung als Validierungswerkzeug.

Ausgehend von Anforderungen an ein sich in der Entwicklung befindendes Produkt ermöglicht das Instrumentarium somit angeleitet durch das übergreifende Vorgehensmodell die Planung und Vorbereitung einer AR-basierten Validierung.

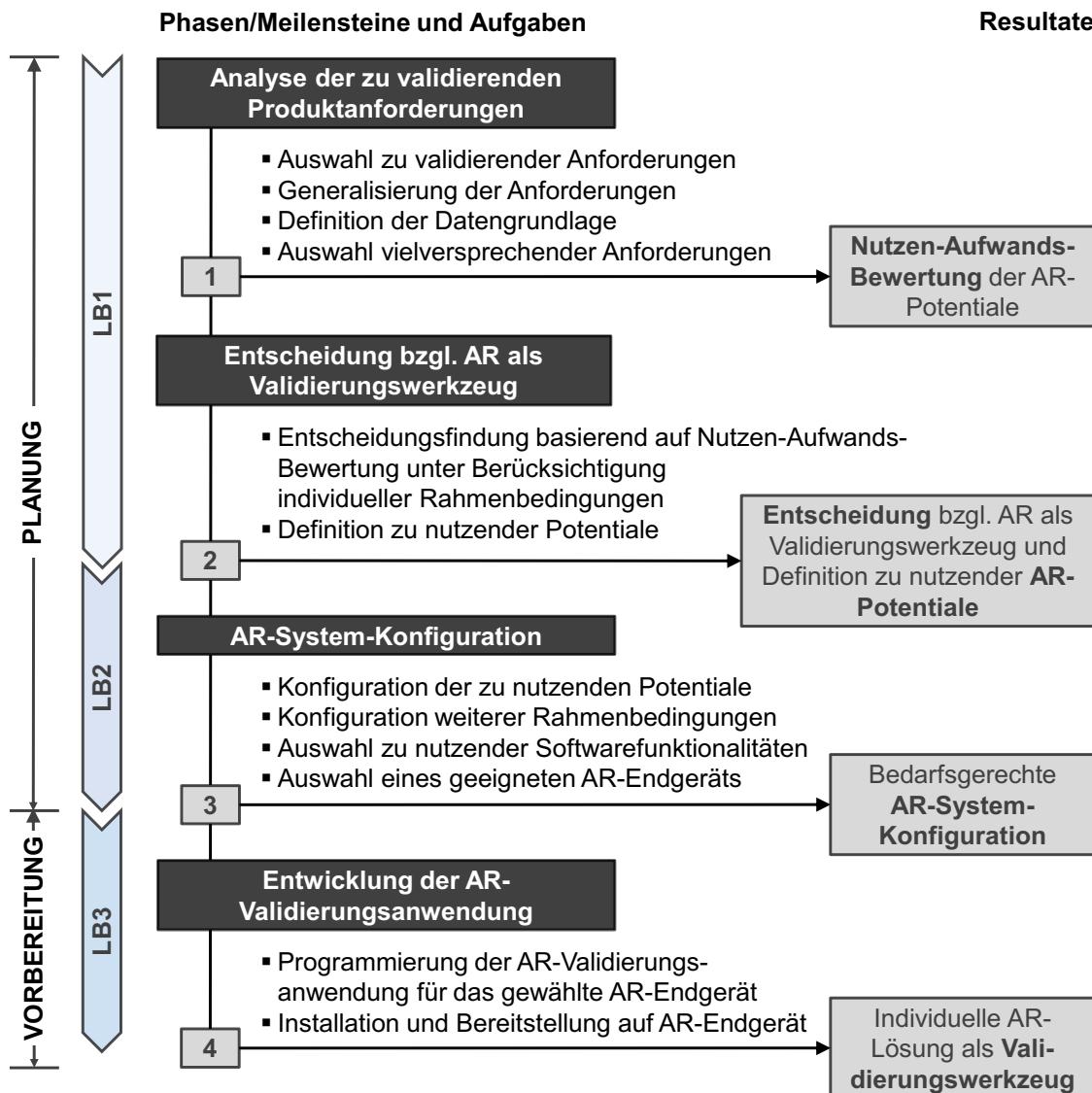


Abbildung 5-44: Übergeordnetes Vorgehensmodell zur Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung

6 Anwendung und Bewertung des Instrumentariums

Das vorliegende Kapitel beschreibt die Anwendung und Bewertung des *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung* an einem exemplarischen Anwendungsfall. Zunächst wird der Anwendungsfall in Kapitel 6.1 vorgestellt. Anschließend werden die drei grundlegenden Lösungsbausteine des Instrumentariums entsprechend des übergeordneten Vorgehensmodells für den Anwendungsfall angewendet (Kapitel 6.2). Abschließend wird das Instrumentarium anhand der Anforderungen aus der Problemanalyse bewertet (Kapitel 6.3).

6.1 Exemplarischer Anwendungsfall: Regal-Bediengerät

Der exemplarische Anwendungsfall ist Teil eines Pilotprojekts aus dem Forschungsverbundprojekt *DigiKAM*. Das Projekt ist zwar bereits abgeschlossen, die für die vorliegende Arbeit relevanten Informationen wurden jedoch bereits während der Projektlaufzeit zusammen mit den beteiligten Unternehmen erarbeitet. Sie können entsprechend für die Anwendung und Bewertung des Instrumentariums herangezogen werden.

Beim im Pilotprojekt beteiligten Unternehmen handelt es sich um ein **mittelständisches Unternehmen** aus dem Maschinen- und Anlagenbau, das **industrielle Regal- und Lagersysteme** fertigt und vertreibt. Im Rahmen des Pilotprojekts sollte ein neues und individuelles **Regal-Bediengerät** unter Einsatz der innovativen Fertigungstechnologie 3D-Druck entwickelt werden. Für die Entwicklung war entsprechend ein auf 3D-Druck spezialisiertes ebenfalls am Projekt beteiligtes Unternehmen zuständig. Dieser 3D-Druck-Dienstleister repräsentiert den Entwickler und ist somit Adressat und Nutzer des Instrumentariums. Der Hersteller für Real- und Lagersysteme stellt wiederum den Kunden der Entwicklung dar. Die beiden Akteure wurden zunächst über eine im Rahmen des Forschungsverbundprojekts entwickelte Internetplattform miteinander vernetzt. Anschließend sollte die gesamte Entwicklung standortübergreifend über diese Plattform abgewickelt werden.

Bisherige Regalbediengeräte waren groß und schwer und ergonomisch nicht sinnvoll gestaltet (Abbildung 6-1, links). Im Pilotprojekt wurde daher 3D-Scanning eingesetzt, um basierend auf einem Handabdruck einen ergonomischen Griff für ein neues Regal-Bediengerät zu konstruieren und zu fertigen (Abbildung 6-1, Mitte). Der Griff lag bereits früh in der Entwicklung als gefertigter Prototyp beim Kunden vor. Basierend auf dem Griff wurden verschiedene Konzepte für die Bedienelemente des Regal-Bediengeräts von den 3D-Druck-Experten erarbeitet (Abbildung 6-1, rechts). Dabei galt es verschiedene vom Kunden an das Regal-Bediengerät gestellte Anforderungen zu erfüllen (Tabelle 6-1). Die Anforderungen betreffen verschiedene Aspekte wie das Material, die Form und Gestalt sowie die Nutzung des Regal-Bediengeräts. Aufgrund der hohen Individualisierung, die unter anderem durch den Einsatz von 3D-Druck-Technologien möglich wird, ist eine Validierung der entwickelten Konzepte für die Bedienelemente von großer Bedeutung.

Nur durch eine effiziente Einbindung des Kunden in die Entwicklung kann ein den Anforderungen entsprechendes, gegenüber dem vorherigen Regal-Bediengerät optimiertes Produkt entwickelt werden. Die Validierung wird jedoch durch die räumliche Trennung der Entwickler und des Kunden erschwert. Entsprechend soll durch Einsatz des Instrumentariums zunächst festgestellt werden, ob AR einen Mehrwert für die Validierung bietet und wenn ja, welche Potentiale der Technologie für die Validierung genutzt werden sollten. Anschließend gilt es, ein bedarfsgerechtes AR-System zu konfigurieren und eine entsprechende Validierungsanwendung als Validierungswerkzeug zu entwickeln.

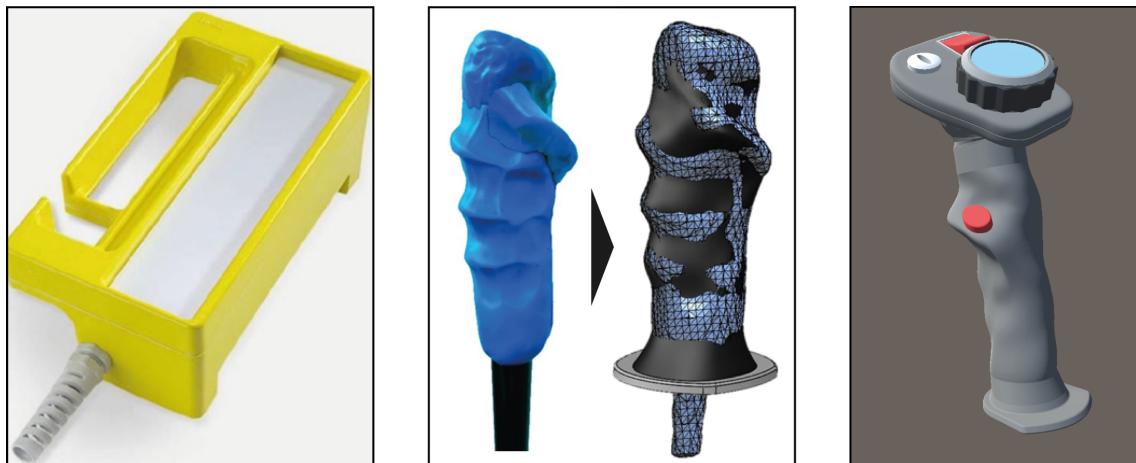


Abbildung 6-1: Anwendungsfall Regal-Bediengerät: ursprüngliche Art des Regalbediengeräts (links), 3D-Scan eines Handabdrucks und entsprechende Konstruktionsdaten (Mitte), Variante eines neuen Designs des Regal-Bediengeräts (rechts)

Tabelle 6-1: An das Regal-Bediengerät gestellte Kundenanforderungen

1	Material	Das Gerät muss leicht und gleichzeitig robust sein.
2	Formfaktor	Das Gerät sollte einen guten Formfaktor und sinnvolle Maße haben.
3	Griff	Das Gerät muss gut und fest in der Hand liegen bzw. gegriffen werden können. Das Gerät wird von verschiedenen Personen benutzt.
4	Handschuhe	Das Gerät muss mit Handschuhen bedienbar sein
5	Einhand-Bedienung	Das Gerät sollte größtenteils mit einer Hand bedienbar sein. Schalter und Bedienelemente sollten gut erreichbar sein.
6	Sicherheit	Das Gerät benötigt einen Sicherheitsschalter zur Verhinderung einer ungewollten Steuerung
7	Notaus	Das Gerät benötigt einen im Notfall einfach auszulösenden Notaus-Schalter
8	Benutzung	Das Gerät muss einfach und intuitiv zu benutzen sein
9	Steuerung	Das Gerät muss an das Regalsteuerungssystem angebunden werden
10	Aufbau	Das Gerät muss einfach zusammengebaut und bei Bedarf Teile ausgetauscht werden können
11	Funktionsumfang	Das Gerät muss die notwendigen Steuerungsfunktionalitäten für das Regalsystem bieten

6.2 Anwendung des Instrumentariums

In diesem Kapitel wird die Anwendung des Instrumentariums am zuvor vorgestellten Anwendungsfall beschrieben. Das Forschungsverbundprojekt *DigiKAM* und somit das Pilotprojekt, aus dem der Anwendungsfall stammt, waren zum Zeitpunkt der Anwendung des Instrumentariums bereits abgeschlossen. Im Rahmen des Projekts wurde jedoch bereits eine AR-basierte Validierung eingesetzt. Die dazu notwendigen Schritte der Planung und Vorbereitung wurden gemeinsam durch die Projektpartner durchgeführt. Entsprechend waren sowohl AR-Experten, 3D-Druck-Experten als Entwickler sowie das beteiligte mittelständische Unternehmen als Kunde an der Erarbeitung der Ergebnisse beteiligt. Weitere Informationen zu den Projekt-Aktivitäten im Kontext der AR-basierten Validierung finden sich in zwei Publikationen des Autors dieser Arbeit (Vorarbeiten [EBA+19] und [EAD22], siehe Übersicht am Anfang dieser Arbeit). Zum einen liegen durch das Projekt somit die für die Anwendung des Instrumentariums nötigen Informationen zu den Kundenanforderungen, den Umfeld- sowie den Rahmenbedingungen vor. Zum anderen können die durch das Projekt vorliegenden Ergebnisse mit den Resultaten der Anwendung des Instrumentariums abgeglichen und zur Bewertung herangezogen werden.

Folgende Situation stellt die **Ausgangslage der Anwendung des Instrumentariums** dar: Der 3D-Druck-Dienstleister hat zunächst einen individuellen Griff und darauf aufbauend verschiedene Konzepte für die Bedienelemente entwickelt. Der Griff liegt bereits als gedruckter Prototyp beim Kunden vor. Der Entwickler muss nun sicherstellen, dass das finale Bedienkonzept die Anforderungen erfüllt. Entsprechend ist eine Validierung durch den Hersteller von Lager- und Regalsystemen als Kunden notwendig. In diesem Zusammenhang benötigt der Entwickler Feedback vom Kunden bzgl. der weiteren Entwicklung der Bedienelemente. Der Entwickler stellt sich die Frage, ob AR als Validierungswerkzeug für das vorliegende Validierungsszenario sinnvoll ist und einen Mehrwert bietet. Da kein spezielles Knowhow und lediglich wenig Erfahrung bzgl. AR vorhanden sind, kann diese Frage nicht ohne Unterstützung beantwortet werden. Aus diesem Grund wendet der Entwickler das Instrumentarium an die darin enthaltenen Lösungsbausteine an.

In den folgenden Abschnitten wird die Anwendung der Lösungsbausteine entsprechend des Vorgehensmodells aus Kapitel 5.5 im Detail beschrieben (vgl. Abbildung 5-44). Ausgangspunkt sind die in Tabelle 6-1 zusammengefassten Anforderungen an das Regal-Bediengerät. Basierend auf diesen Anforderungen erfolgt zunächst die Analyse der zu validierenden Produktanforderungen in Phase 1 (Abschnitt 6.2.1). Anschließend wird in Phase 2 eine Entscheidung bzgl. AR als Validierungswerkzeug getroffen (Abschnitt 6.2.2). Im Rahmen von Phase 3 wird daraufhin ein bedarfsgerechtes AR-System konfiguriert (Abschnitt 6.2.3) und in Phase 4 eine entsprechende AR-Validierungsanwendung entwickelt (Abschnitt 6.2.4).

6.2.1 Phase 1 – Analyse zu validierender Produktanforderungen

In Phase 1 kommt die Systematik zur Nutzen-Aufwands-Analyse zum Einsatz (Lösungsbaustein 1 des Instrumentariums). Zunächst wird das in Form einer Excel-Arbeitsmappe bereitgestellte **interaktive Hilfsmittel zur Nutzen-Aufwands-Analyse** der AR-Potentiale geöffnet. Entsprechend des in Abbildung 5-9 dargestellten Vorgehensmodells werden anschließend die die ersten drei Phasen der Nutzen-Aufwands-Analyse durchlaufen. Diese Phasen werden im Folgenden mit 1.1 bis 1.3 nummeriert, um eine Verwechslung mit den Phasen des übergeordneten Vorgehensmodells zu vermeiden.

Phase 1.1: In der Eingabetabelle werden zunächst die Anforderungen eingetragen. Durch die Bewertung der Relevanz der generischen Produktmerkmale je Anforderung erfolgt die **Generalisierung**. Genauere Erklärungen zu den Produktmerkmalen werden dem **Produktmerkmal-Steckbrief** entnommen (s. Anhang A3). Der Steckbrief vermittelt ein eindeutiges Bild der Produktmerkmale und unterstützt entsprechend die Bewertung. Die Anforderungen 2 bis 5 betreffen vor allem die *räumliche Dimension* als Produktmerkmal. Beim Formfaktor (Anforderung 2) sind zudem die *Proportionen* von besonderer Bedeutung. Bei Anforderungen 2 und 3 bzgl. des Griffes und der Bedienung mit Handschuhen spielt die *Haptik* eine besondere Rolle. Das Produktmerkmal *Benutzerschnittstelle* wird stark durch die Anforderungen 4 bis 8 adressiert. Dies betrifft die Positionierung der Bedienelemente und deren Benutzung. Gleichzeitig ist bei diesen Anforderungen auch das Produktmerkmal *Erreichbarkeit* relevant. Zur intuitiven Benutzung des Regal-Bediengeräts (Anforderung 8) ist es wichtig, dass die Benutzerschnittstellen einsehbar und ersichtlich sind. Anforderung 9 adressiert die Steuerung des Regals somit lediglich das Produktmerkmal *Konnektivität*. Für den Aufbau (Anforderung 10) sind die *Systemstruktur* sowie die *Montage* relevant. Anforderung 12 betrifft lediglich die *Endbenutzerfunktionalität*. Die Bewertung der führt durch die im Hintergrund stattfindenden Berechnung automatisch zu individuellen **Nutzenwerten** je Anforderung (Abbildung 6-2, links).

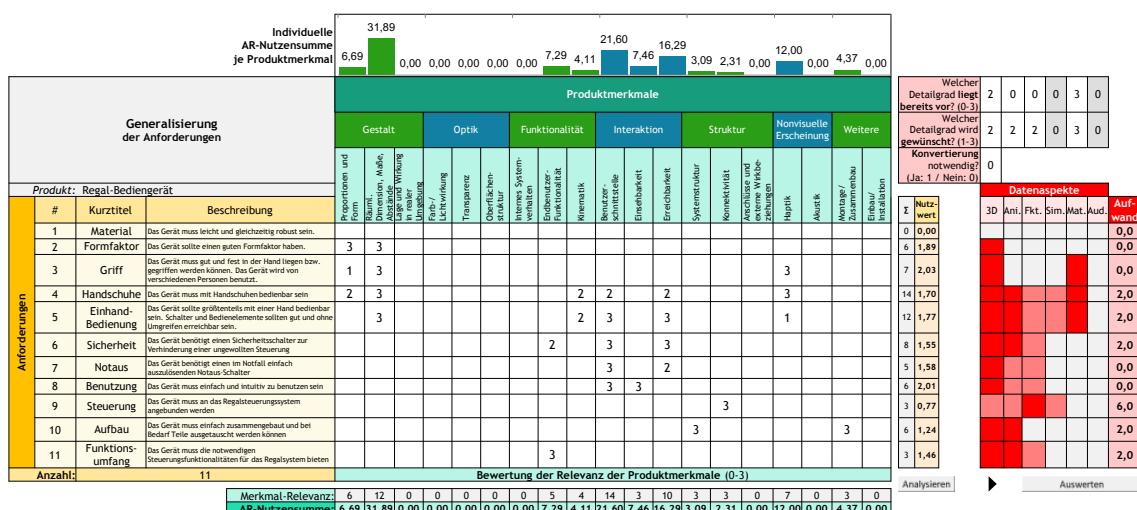


Abbildung 6-2: Eingabetabelle des interaktiven Hilfsmittels zur Nutzen-Aufwands-Analyse für den Anwendungsfall des Regal-Bediengeräts

Phase 1.2: Im rechten Teil der Eingabetabelle werden die als Grundlage der Validierung verfügbaren sowie die geforderten **Datenaspekte** definiert (Abbildung 6-2, rechts). Hier wird der **Daten-Steckbrief** (s. Anhang A3) unterstützend hinzugezogen. Simulations- und Audio-Daten werden aufgrund der vorherigen Bewertung der Anforderungen nicht benötigt. Konstruktionsdaten der Bedienkonzepte liegen bereits als 3D-Modelle entsprechend des Detailgrads 2 vor (vgl. Abbildung 5-7). Die Konstruktion erfolgte mit Konstruktionswerkzeugen der Firma Autodesk, die einen direkten Export im FBX-Format ermöglichen. Eine Konvertierung ist entsprechend nicht notwendig. Da es beim Regal-Bediengerät vor allem um die Gestalt und die Struktur geht, reichen 3D-Modelldaten im Detailgrad 2 für die Validierung aus. Detaillierte, texturierte Visualisierungsmodelle sind nicht notwendig. Über Transformations-Animationen (Detailgrad 2) soll die Funktionsweise der Bedienelemente, bspw. Schieberegler oder Drehregler, veranschaulicht werden. Für die Validierung der Steuerung des Regals sind zudem interaktive Funktionalitäten notwendig (ebenfalls Detailgrad 2). Weder Animationen noch Funktionalitäten liegen bereits vor. Material wird entsprechend des Detailgrad 3 in der finalen Form benötigt, um ergonomische Aspekte zu validieren. In Form des 3D-gedruckten Griff-Prototypen liegt dieser Datenaspekt entsprechend bereits in Detailgrad 3 vor. Die Bewertung der Datenaspekte führt zu individuellen **Aufwandswerten** für die einzelnen Anforderungen.

Phase 1.3: Die Portfoliodarstellung veranschaulicht das Nutzen-Aufwands-Verhältnis der einzelnen Anforderungen (Abbildung 6-3, 1). Grundsätzlich weisen die meisten Anforderungen ein gutes Nutzen-Aufwands-Verhältnis auf. Über die Auswahlmöglichkeit (Abbildung 6-3, 2) werden die Anforderungen 2, 3, 4, 5 und 8 für die weitere Analyse ausgewählt. Anforderungen 6 und 7 haben ebenfalls vielversprechendes Potential. Der Notaus-Schalter (Anforderung 7) wurde jedoch bereits prototypisch direkt am physischen 3D-Druck-Modell erprobt und der Sicherheitsschalter (Anforderung 6) war in den vorhandenen Konzepten noch nicht enthalten. Basierend auf der Auswahl der Anforderungen erfolgt die weitere Auswertung. Sie wird durch den Button „Auswahl auswerten“ gestartet. Das **Nutzen-Aufwands-Verhältnis** liegt im vorliegenden Fall bei 2,5 und ist entsprechend hoch. Der Einsatz von AR als Validierungswerkzeug ist dementsprechend sinnvoll. Zudem wird eine Übersicht der notwendigen Aufwände dargestellt (Abbildung 6-3, 4). 3D-Modelldaten sowie Materialien sind in den erforderlichen Detailgraden bereits vorhanden und entsprechend grün dargestellt. Somit sind lediglich Aufwände für die Erstellung von Transformations-Animationen notwendig, was in Summe einem Aufwand von 2 entspricht. Als weitere Entscheidungsgrundlage wird der Potentialnutzen der einzelnen AR-Potentiale in einem Balkendiagramm visualisiert (s. Abschnitt 5.2.2.2). Im vorliegenden Validierungsszenario hat die **3D-Darstellung** den größten Nutzen, gefolgt von der **Einbeziehung des Nutzers**. Diese Potentiale allein wären auch mit VR realisierbar. Die **Kombination mit der realen Umgebung** weist jedoch ebenfalls einen hohen Wert auf. Sie ist nur mit AR möglich. Ein weiteres für die Validierung nützliches Potential stellen **Tangible User Interfaces** dar. Ein Abgleich mit den **Referenzprofilen** (RP, Abbildung 5-17) ergibt, dass das vorliegende Validierungsszenario stark dem **RP4 „Virtuell erweiterter haptischer Prototyp“** entspricht. In diesem Profil werden physische Objekte als Tangible

User Interface eingesetzt und mit virtuellen Prototypen erweitert. Im vorliegenden Anwendungsfall kann der 3D-gebrückte Griff als Tangible User Interface verwendet und an ihm die Bedienelemente virtuell erweitert werden. Somit lassen sich ergonomische und haptische Aspekte der Bedienkonzepte effizient validieren. Entsprechend liegt ebenfalls eine gewisse Ähnlichkeit zum RP2 "Ergonomie-Test im Kontext realer Räumlichkeiten" vor, bei dem natürliche Benutzereingaben verwendet werden.

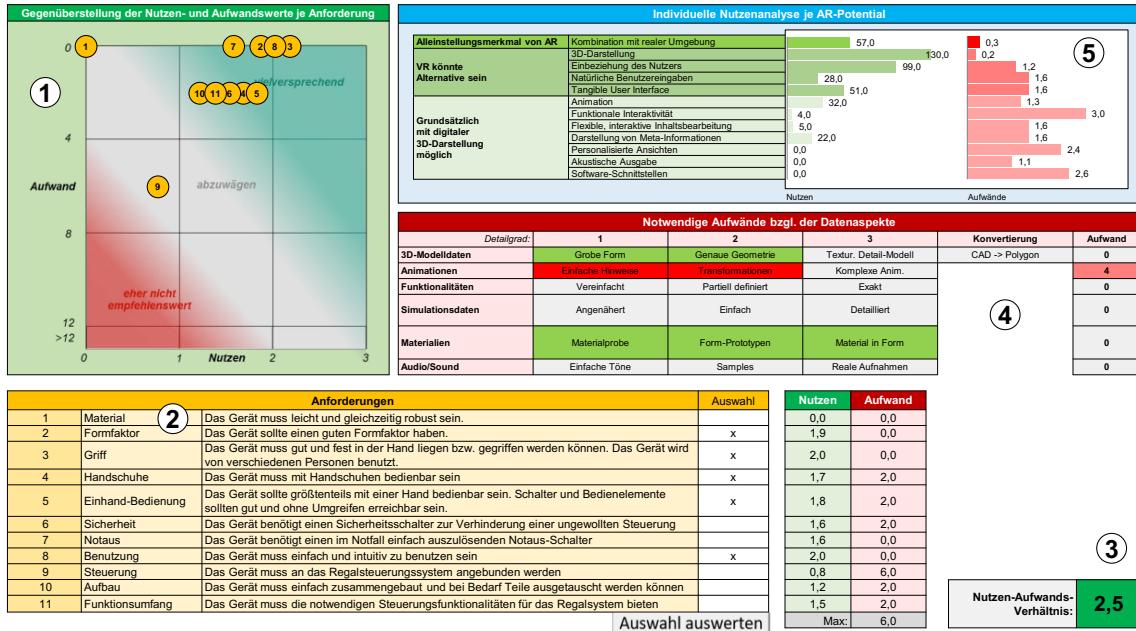


Abbildung 6-3: Ergebnisse der Nutzen-Aufwands-Analyse im interaktiven Hilfsmittel:
(1) Darstellung der Anforderungen im Portfolio, (2) Nutzen- und Aufwandswerte der einzelnen Anforderungen inkl. Auswahlmöglichkeit zur detaillierten Analyse, (3) Nutzen-Aufwands-Verhältnis, (4) notwendige und vorhandene Aufwände, (5) Potentialnutzen und technische Aufwände je AR-Potential

6.2.2 Phase 2 – Entscheidung bzgl. AR als Validierungswerkzeug

Phase 2 des übergeordneten Vorgehensmodells entspricht der vierten und abschließenden Phase der Systematik zur Nutzen-Aufwands-Analyse (Phase 1.4).

Basierend auf dem Nutzen-Aufwands-Verhältnis sowie dem Abgleich mit den Referenzprofilen und damit verbundenen Empfehlungen wird eine **Entscheidung für den Einsatz von AR als Validierungswerkzeug** getroffen. Unterstützt durch die Potentialnutzenwerte sowie den **AR-Potential-Steckbrief** werden zudem folgende **AR-Potentiale** zur Nutzung ausgewählt:

- Kombination mit realer Umgebung
- 3D-Darstellung
- Einbeziehung des Nutzers

- Natürliche Benutzereingaben
- Tangible User Interfaces
- Animationen
- Flexible interaktive Inhaltsbearbeitung
- Darstellung von Meta-Informationen

Die *flexible interaktive Inhaltsbearbeitung* sowie die *Darstellung von Meta-Informationen* sind entsprechend der Potentialnutzen nicht notwendig. Sie können jedoch eingesetzt werden, um dem Kunden verschiedene Varianten sowie hilfreiche Zusatzinformationen zu einzelnen Bedienelementen zu präsentieren. Basierend auf den ausgewählten Potentialen wird anschließend die AR-System-Konfiguration durchgeführt.

6.2.3 Phase 3 – AR-System-Konfiguration

Die zu nutzenden AR-Potentiale dienen als Grundlage für die AR-System-Konfiguration. Diese wird mit Hilfe des wissensbasierten Konfigurationswerkzeugs als zweitem Lösungsbaustein des Instrumentariums durchgeführt. Nach dem Start des interaktiven Werkzeugs (s. Abschnitt 5.3.2) erscheint zunächst das Fenster zur **Auswahl der zu nutzenden AR-Potentiale** (Abbildung 6-4, Schritt 1). Die Darstellung der Potentiale erfolgt dynamisch, nachdem alle Potentiale aus der Wissensbasis ausgelesen wurden. Die Darstellungsreihenfolge der Potentiale ist abhängig von der Reihenfolge des Auslesens aus der Wissensbasis. Entsprechend erfolgt bisher keine Gruppierung oder Sortierung. Durch Aktivierung der Button erfolgt die entsprechende Auswahl der zu nutzenden AR-Potentiale. Zusätzlich zu den durch die AR-System-Konfiguration empfohlenen Potentialen wurden im vorliegenden Fall *Kollaboration* sowie *Integriertes virtuelles Feedback* gewählt. Diese Potentiale sind als übergeordnet anzusehen und nach Bedarf bzw. abhängig von der Validierungssituation zu nutzen. Im vorliegenden Fall arbeiten der Entwickler und der Kunde des Regal-Bediengeräts standortübergreifend zusammen. Entsprechend ist eine kollaborative Anwendung erforderlich. Zudem wird die Möglichkeit des Feedbacks gewünscht, damit der Entwickler auch im Nachgang an die Validierung das Ergebnis der Validierung einsehen und für die Weiterentwicklung berücksichtigen kann.

Nach der Bestätigung der AR-Potentiale erfolgen die ersten Berechnungen. Anschließend wird die grafische Benutzeroberfläche zur **Konfiguration der Softwarefunktionalitäten** angezeigt (Abbildung 6-4, Schritt 2). Entsprechend des in Abschnitt 5.3.2.2 beschriebenen Vorgehens muss durch den Anwender eine Tracking-Lösung ausgewählt werden. Darüber wird über ein zusätzliches Info-Fenster informiert. Bei der Validierung des Regal-Bediengeräts soll der bereits vorhandene Griff als tangible User Interface eingesetzt werden. Er soll vom Kunden in der Hand gehalten werden. Die Bedienelemente sollen virtuell passend zum Griff dargestellt werden. Entsprechend ist eine dynamische und au-

tomatische Positionierung entsprechend der Position und Ausrichtung des Griffes notwendig. Ein *3D-Objekt-Tracking* ist nicht realisierbar, da große Teile des Griffes durch die Hand des Kunden verdeckt und eine Erkennung somit nicht möglich ist. Der Griff verfügt allerdings über ein flaches Endstück am oberen Ende. Auf diesem lässt sich ein 2D-Marker anbringen. Aus diesem Grund wurde das *2D-Marker-Tracking* ausgewählt. Zusätzlich zu den vorgegebenen, notwendigen Softwarefunktionalitäten zur Nutzung der AR-Potentiale wurde *Speech-to-Text* ausgewählt, um eine einfache Eingabe von Text zu ermöglichen. Nach Bestätigung der Auswahl durch den entsprechenden Button erfolgt die weitere Analyse basierend auf der AR-System-Wissensbasis.

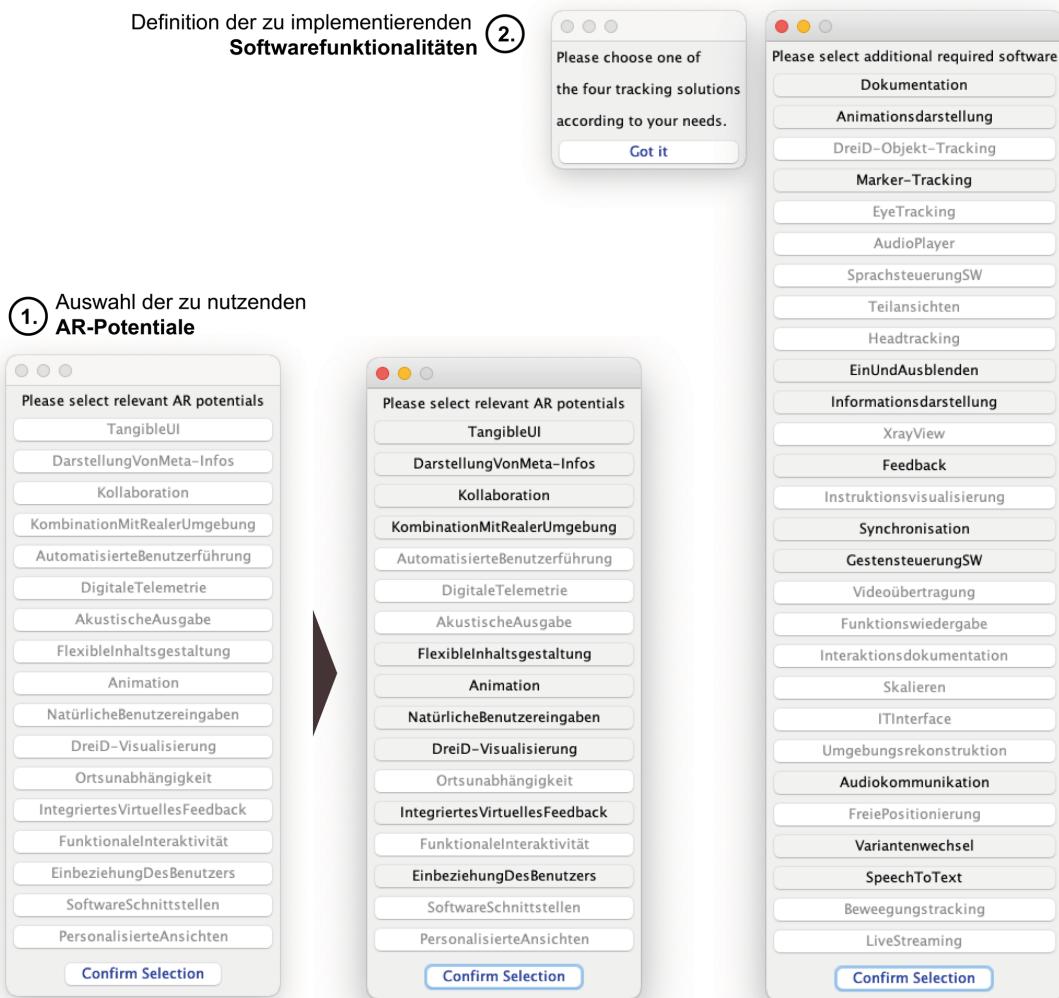


Abbildung 6-4: Interaktionsfenster der AR-System-Konfiguration

Die **Ergebnisse der Analyse** werden dem Entwickler in der entsprechenden grafischen Benutzerschnittstelle präsentiert (Abbildung 6-5). Die Ausgabe zeigt, welche Systemkomponenten erforderlich sind oder ausgeschlossen werden und welche Umfeldfaktoren erfüllt sein müssen oder zu vermeiden bzw. nicht realisierbar sind.



Abbildung 6-5: Ausschnitt der Konfiguration inkl. der individuellen Ergänzungen (rote Kreise) (Die komplette Konfiguration ist in Anhang A4 zu finden)

Unter anderem ist für den vorliegenden Validierungsfall ein *projektionsbasiertes Anzeigesystem* ausgeschlossen. Zudem muss ein *binokularer Systemtyp* eingesetzt werden, um eine 3D-Darstellung zu ermöglichen. Weitere Konfigurationen wurden entsprechend individuell vorliegender **Rahmenbedingungen und Anforderungen** vorgenommen. Diese Konfigurationen sind in Abbildung 6-5 mit roten Kreisen markiert. Zum einen soll die Validierung *Indoor* durchgeführt werden. Dabei sollen auch Personen, die eine *Alltagsbrille* tragen, an der Validierung beteiligt werden. Um die Kommunikation mit weiteren Personen zu gewährleisten, werden *Kopfhörer* ausgeschlossen. Das betrifft auch ein *Kopfhörer-Mikrofon* sowie *Bluetooth-Kopfhörer*. Gleichzeitig ist bereits durch die vorherige Analyse *Audio-Output* erforderlich. Entsprechend muss das AR-Endgerät einen

Lautsprecher besitzen, um diese Anforderung zu erfüllen. Zudem sind beim Kunden kein *5G-Netz* sowie keine *Cloud* zum dynamischen Einlesen von Daten vorhanden. Die Validierung soll zudem *mobil* durchgeführt werden. Entsprechend wird ein *kabelgebundenes* Gerät ausgeschlossen. Über den Button wird die Konfiguration bestätigt.

Basierend auf der Konfiguration werden alle in der AR-System-Wissensbasis vorhandenen AR-Endgeräte analysiert und mit den Anforderungen abgeglichen. Als Ergebnis werden die AR-Endgeräte inklusive der Analyse-Ergebnisse ausgegeben. Zur weiteren Analyse und Filterung werden für den vorliegenden Fall drei Attribute über das **Attribut-Filterungs-Fenster** spezifiziert (s. Schritt 10 in Abschnitt 5.3.2.2). Zum einen ist eine *minimale Akkulaufzeit* von einer Stunde gewünscht, um genug Zeit für die Validierung zu haben. Zudem besteht beim Kunden, dem Hersteller von Regal- und Lagersystemen, das Vorhaben, sich ein AR-Endgerät anzuschaffen und die Technologie für verschiedene Anwendungsfälle im Unternehmen zu nutzen. Das dazu verfügbare Budget liegt bei circa 4000€. Diese werden entsprechend als *maximaler Preis* definiert. Das **Ergebnis der Analyse** inklusive der Filterung ist in Abbildung 6-6 dargestellt. Im unteren Bereich werden die aktuell 17 in der Wissensbasis enthaltenen AR-Endgeräte und ihre Bewertung zeilenweise dargestellt. Wie das Ausgabefenster anhand der grünen Hinterlegung veranschaulicht, sind lediglich die *HoloLens 2* sowie die *Magic Leap* entsprechend der grundlegenden Anforderungen sowie der Attribut-Spezifikation für den vorliegenden Validierungsfall geeignet. Für die anderen Geräte ist jeweils angegeben, aus welchen Gründen sie die Anforderungen nicht erfüllen. Bei den meisten Geräten liegt es daran, dass sie keine *Gestensteuerung* ermöglichen, die für natürliche Benutzereingaben jedoch notwendig ist. Andere haben einen unpassenden *Systemtypen* oder es handelt sich um *kabelgebundene* Geräte. Als Beispiel ist hier das Apple iPad zu nennen. Beim iPad handelt es sich um ein Tablet, dass entsprechend keine immersive *3D-Darstellung* unter *Einbeziehung des Nutzers* und zudem keine *Gestensteuerung* ermöglicht. Die in grüner Schrift dargestellten Geräte entsprechen der Attribut-Filterung, erfüllen jedoch nicht die grundlegenden Anforderungen. Sie sind daher nicht geeignet.

Rechts neben der Auswertung der AR-Endgeräte wird der Entwickler falls notwendig über zu berücksichtigende Einflüsse einzelner Komponenten oder Umfeldfaktoren informiert. Dass dieser Bereich fehlt bzw. leer ist, bedeutet, dass es im vorliegenden Fall **keine besonderen Einflüsse** zu berücksichtigen gibt. In der oberen Zeile des Ausgabefensters werden zunächst die geforderten bzw. vorhandenen sowie die auszuschließenden bzw. nicht vorhandenen Umfeldfaktoren gelistet. Im vorliegenden Fall betrifft letzteres vor allem die **nicht vorhandene Cloud**.

Oben rechts im Ausgabefenster werden die **zu implementierenden Softwarefunktionalitäten** inklusive der SDKs, durch die sie realisiert werden können, präsentiert. Für den vorliegenden Validierungsfall sind dies die folgenden mit den jeweiligen SDKs:

- Ein- und Ausblenden: *Unity-intern*
- Informationsdarstellung: *MRTK, nrSDK, Unity-intern, ThinkReality*

Geforderte bzw. vorhandene Umfeldfaktoren:		Ausschlossene Umfeldfaktoren:		Ausgewählte bzw. benötigte Software: (link): mögl. SDKs)	
Ab-System	Nicht unterstützte Komponenten	Konflikt mit ausschlossener Komponente	Mangelnde SW-Unterstützung	Erforderliche SDKs	
ODGR-7	Alltags-Brille System_Headset Gestensteuerung KGB-D-Kamera indoor		GestensteuerungSW		
VuzixBlade	Alltags-Brille System_Prozessor System_Audio-Output System_SekundäreStromversorgung Gestensteuerung KGB-D-Kamera	System_Audio-Output[Bluetoothkopfhörer] System_Prozessor[LowPerformance]	GestensteuerungSW		
Hololens2				MTK	
LumusDK-52	Alltags-Brille System_Netzwerk-Verbindung System_Prozessor System_Audio-Output System_SekundäreStromversorgung Gestensteuerung KGB-D-Kamera	System_Interaktionssystem[Tasten-Steuerung]	GestensteuerungSW	Erforderliches SDK: MRTK	
MadGazeGlow	Nicht unterstützte Komponente: Gestensteuerung		Konflikt mit ausgeschlossener Komponente: Systemtyp Tablet		GestensteuerungSW
AppleiPad	Alltags-Brille System_Prozessor Gestensteuerung	System_Systemtyp[Tablet]		ARKit	
RokidProjectAURORA	Alltags-Brille System_Interaktionssystem Gestensteuerung KGB-D-Kamera		GestensteuerungSW		
VuzixBladeSafety	Alltags-Brille System_Prozessor System_Audio-Output System_SekundäreStromversorgung Gestensteuerung KGB-D-Kamera	System_Audio-Output[Bluetoothkopfhörer] System_Prozessor[LowPerformance]	GestensteuerungSW		
SamsungGalaxyA51	Alltags-Brille System_Interaktionssystem Gestensteuerung KGB-D-Kamera	System_Systemtyp[Smartphone]		ARCore	
VuzixM4000	Alltags-Brille System_Systemtyp System_SekundäreStromversorgung Gestensteuerung KGB-D-Kamera	System_Systemtyp[HMD-Monokular]	GestensteuerungSW		
EPSONMoverioBT-300	Alltags-Brille System_Prozessor System_Interaktionssystem Gestensteuerung KGB-D-Kamera	System_Interaktionssystem[Tasten-Steuerung, Touchpad-Steuerung] System_Prozessor[LowPerformance]	GestensteuerungSW		
LightformLF2	Alltags-Brille System_Prozessor System_Bewegungsverfolgung System_Interaktionssystem Gestensteuerung System_SekundäreStromversorgung System_Audio-Input System_Audio-Output KGB-D-Kamera	System_Prozessor[Projektor] System_Interaktionssystem[Computerperipherie-Steuerung] System_PräziseStromversorgung[Kabelgebunden]	GestensteuerungSW		
LenovoThinkRealityA3	Alltags-Brille System_Prozessor System_Datenleser System_SekundäreStromversorgung Gestensteuerung KGB-D-Kamera	System_PräziseStromversorgung[Kabelgebunden]		ThinkReality	
LumusDKVision	Alltags-Brille System_Interaktionssystem System_Prozessor System_SekundäreStromversorgung Gestensteuerung KGB-D-Kamera	System_Interaktionssystem[Tasten-Steuerung]	GestensteuerungSW		
MagicLeap				Lumin	
VuzixM400	Alltags-Brille System_Systemtyp System_SekundäreStromversorgung Gestensteuerung KGB-D-Kamera	System_Systemtyp[HMD-Monokular]	GestensteuerungSW	Erforderliches SDK: Lumin	
MadGazeX5	Alltags-Brille System_Prozessor System_Audio-Output System_Systemtyp System_SekundäreStromversorgung KGB-D-Kamera	System_Systemtyp[HMD-Monokular] System_Audio-Output[Bluetoothkopfhörer] System_Prozessor[LowPerformance]	GestensteuerungSW		

Abbildung 6-6: Ergebnis der AR-System-Konfiguration nach der Attributfilterung

- Animationsdarstellung: *Unity-intern*
- Variantenwechsel: *Unity-intern*
- Feedback: *MRTK, ARCore, ARKit, Unity-intern*
- Dokumentation: *OpenCV, MRTK, Unity-intern, Photon, iSAR visionLib, OpenCV, Vuforia, ARCore, Wikitude, ARKit, nrSDK, ARToolkit, ThinkReality*
- 2D-Marker-Tracking: *Photon*
- Audio-Kommunikation: *Photon*
- Synchronisation: *Photon, iSAR*
- Speech-to-Text: *MRTK, Unity-intern*
- Gestensteuerung: *MRTK, ThinkReality, Lumin, nrSDK*

Die zu entwickelnde Software adressiert entsprechend eine **kollaborative Validierung**, die virtuelle Bedienkonzepte über ein 2D-Marker-Tracking auf dem 3D-gedruckten Griff visualisiert. Kunde und Entwickler können über Audio kommunizieren. Dabei wird die vom Kunden betrachtete Szene zum Entwickler synchronisiert. Der Entwickler kann dem Kunden verschiedene Varianten der Bedienkonzepte präsentieren und diese über Animationen sowie Zusatzinformationen erläutern. Der Kunde wiederum kann Feedback zu den Prototypen geben, dass der Entwickler in Rahmen der weiteren Entwicklung berücksichtigen kann. Die Interaktion beim Kunden erfolgt über Gesten und Sprach-Input und ist somit natürlich und intuitiv. Das konfigurierte AR-System ermöglicht so eine effiziente und aussagekräftige Validierung des Regal-Bediengeräts.

Aufgrund verschiedener Aspekte wurde die Microsoft **HoloLens 2** als AR-Endgerät ausgewählt. Zum einen handelt es sich um ein verbreitetes Gerät, für das bereits industrielle Software-Lösungen für spezifische Anwendungsfälle verfügbar sind. Dies ist förderlich entsprechend des Vorhabens des Unternehmens, AR für weitere Anwendungsfälle einzusetzen. Beim Hersteller Microsoft handelt es sich zudem grundsätzlich um einen etablierten Hardware- und Software-Anbieter, sodass eine Integration des Windows-basierten AR-Endgeräts in die Unternehmens-IT einfacher zu gestalten ist. Hinzu kommt, dass das von der HoloLens verwendete *MRTK* im Gegensatz zum *Lumin SDK* vielfältigere Möglichkeiten bietet und für die meisten der zu implementierenden Softwarefunktionalitäten geeignet ist. Die entsprechend basierend auf dem *MRTK* entwickelten Lösungen lassen sich zudem auf Android- und iOS-basierte AR-Endgeräte übertragen. Die folgende Anwendung des Werkzeugkoffers zur AR-Anwendungsentwicklung erfolgte somit für die HoloLens 2 entsprechend der zuvor beschriebenen Software-Konfiguration.

6.2.4 Phase 4 – Entwicklung der AR-Validierungsanwendung

Die Validierungsanwendung wird basierend auf dem Werkzeugkoffer zur AR-Anwendungsentwicklung (Lösungsbaustein 3 des Instrumentariums) durch den Autor dieser Arbeit entwickelt. Ein weiterer dem grundsätzlichen Konzept des ARVKit entsprechender Ansatz wurde und wird im Rahmen verschiedener Projekte entwickelt und angewendet. So können unter anderem Erkenntnisse aus dem Forschungsprojekt *EMERGE* sowie einzelnen Industrieprojekten, in denen das grundsätzliche Konzept des Werkzeugkoffers bereits angewendet wurde, in die in Kapitel 6.3 folgende Bewertung des Instrumentariums einfließen.

Zielplattform der Entwicklung ist entsprechend der vorangegangenen AR-System-Konfiguration die **Microsoft HoloLens 2** als AR-Endgerät. Zudem handelt es sich um eine kollaborative Anwendung, bei der der Entwickler und der Kunde miteinander vernetzt sind und der Entwickler aus der Ferne an der Validierung teilhaben und die zu validierenden Bedienkonzepte präsentieren und erklären kann. Als Gerät für den Entwickler wird ein **Desktop- bzw. Surface-Gerät mit Windows 10** ausgewählt. Somit handelt es

sich sowohl beim Kunden- als auch beim Expertengerät um Windows- bzw. UWP-Geräte. Entsprechend dem Leitfaden zur Anleitung der Entwicklung (s. Abbildung 5-33) erfolgt die Entwicklung wie folgt:

Schritt 1 – Entwicklungsumgebung: *Unity* wird mittels *Unity Hub* in der Version 2021.2.11f1 installiert. Zudem wird die *Microsoft Visual Studio Community Edition 2019* installiert.

Schritt 2 – Projekt: Entsprechend der AR-System-Konfiguration wird die *UWP-Projektvorlage* ausgewählt und in Unity geöffnet.

Schritt 3 – Szene: Da eine kollaborative AR-Validierungsanwendung zwischen zwei UWP-Geräten entwickelt werden soll, wird innerhalb der Projektvorlage das kollaborative Szenen-Template *ARValidation_C-UWP* verwendet. Die neue Szene wird als *ARValidation_C-UWP_Bediengeräet* gespeichert. Links in Abbildung 6-7 sind der entsprechende Szenengraph sowie die vordefinierten Entwickler-Benutzerschnittstellen der vorkonfigurierten Szene zu sehen. Die entsprechend der AR-System-Konfiguration zu nutzenden Funktionalitäten **Audio-Kommunikation** und **Synchronisation** sind in der Szene bereits integriert. Im unteren Bereich der Unity-Benutzeroberfläche sind zudem die verschiedenen Bausteine im Projektordner verfügbar. Das **Validierungs-Konfigurationsmenü** wird geöffnet und im Layout platziert. In diesem wird anschließend die Projekt-ID definiert, die zur Verbindung des Entwicklers und des Kunden dient (1.).

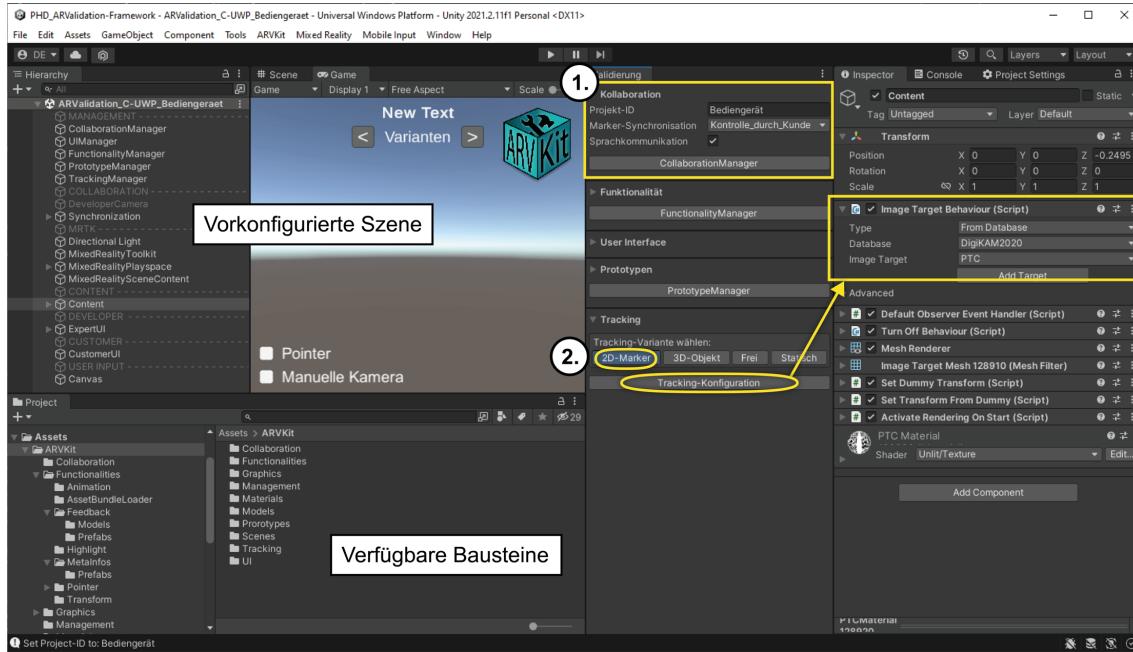


Abbildung 6-7: Ansicht in Unity nach dem Erstellen einer neuen Szene

Schritt 4 – Tracking: In der vorbereiteten Szene wird als nächstes das Tracking konfiguriert (Abbildung 6-7, rechts). Im Konfigurationsfenster wird über den entsprechenden Button das **2D-Marker-Tracking** aktiviert (2.). Über den „Tracking-Konfiguration“ Button wird anschließend das Tracking-Objekt ausgewählt und im Inspector der 2D-Marker

konfiguriert. Dazu muss zuvor eine *Vuforia-Marker-Datenbank* importiert werden. Diese kann Online im *Vuforia Developer-Portal*¹⁶ angelegt und heruntergeladen werden. Im Inspector kann anschließend ein Marker aus der Datenbank ausgewählt werden.

Schritt 5 – Prototypen: Im Rahmen der Nutzen-Aufwands-Analyse der AR-Potentiale (Kapitel 5.2) wurden **3D-Modelldaten**, **Animationen** sowie **Materialien** als relevante **Datenaspekte** für die Prototypen identifiziert. Das *Material* liegt bereits in Form des 3D gedruckten Griffes beim Kunden vor. Es soll die Grundlage für die Validierung der Bedienkonzepte bilden. Zudem liegen bereits drei Bedienkonzepte als 3D-Konstruktionsdaten vor. Sie werden als **3D-Modelle** im FBX-Format exportiert. Anschließend werden sie in Unity importiert und unter dem *Prototypes*-Knoten in die Szenenhierarchie integriert. Zusätzlich zu den Bedienkonzepten muss das 3D-Modell des Griffes importiert und integriert werden. Das Griff-Objekt wird über das Kontext-Menü als **Tangible User Interface** definiert, wodurch ihm eine *Tangible-UI*-Komponente hinzugefügt wird (Abbildung 6-8, rechts). Wie das Konfigurationsmenü zeigt, werden somit die Bedienkonzepte automatisch als vorhandene **Varianten** erkannt und der Griff gesondert behandelt. Im Rahmen des Pilotprojekts, in dem bereits eine AR-basierte Validierung für den vorliegenden Anwendungsfall durchgeführt wurde, wurden die 3D-Modelldaten über eine Online-Datenbank dem Kunden zur Verfügung gestellt. Sie wurden somit nach dem Start der Validierungsanwendung eingelesen und in die Szene integriert. Ein entsprechender Baustein liegt vor. Da das Projekt bereits abgeschlossen ist, steht die Datenbank aktuell nicht zur Verfügung. Entsprechend wird im Rahmen der Anwendung des Instrumentariums die integrierte Prototypen-Variante gewählt.

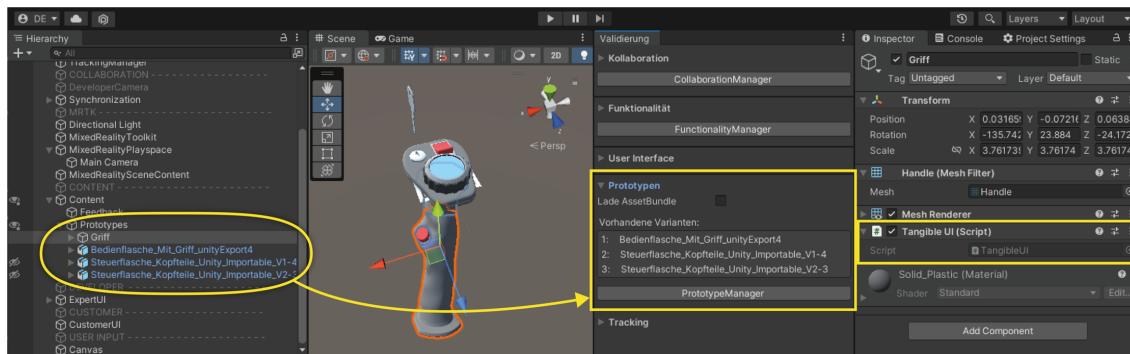


Abbildung 6-8: Integration der 3D-Modelle für die Bedienkonzepte in die Szene sowie Definition des Griffes als TUI

Für die Erstellung und Integration der **Animationen** gibt es grundsätzlich verschiedene Möglichkeiten. Animationen können vor dem Export in den entsprechenden Konstruktions- bzw. Modellierungswerkzeugen erstellt werden. Da Unity bereits für die Anwendungsentwicklung im Einsatz ist, werden die Animationen im vorliegenden Fall direkt in

¹⁶ <https://developer.vuforia.com/vui/develop/databases>

Unity erstellt. Über das Kontext-Menü können einem Prototypen-Teilmodell ein vorbereiteter *Animator* sowie die *AnimatorHandler*-Komponente des Animations-Bausteins hinzugefügt werden (Abbildung 6-9). Nach der Erstellung einer Animation werden das Starten und Stoppen über sogenannte Trigger im Animator-Graphen definiert. Die Trigger *Animate* und *Stop* sind bereits vordefiniert. Sie werden beim Klicken der Animations-Button im Animations-Menü gesetzt und bewirken über den *AnimatorHandler* entsprechend das Starten und Stoppen der Animation.

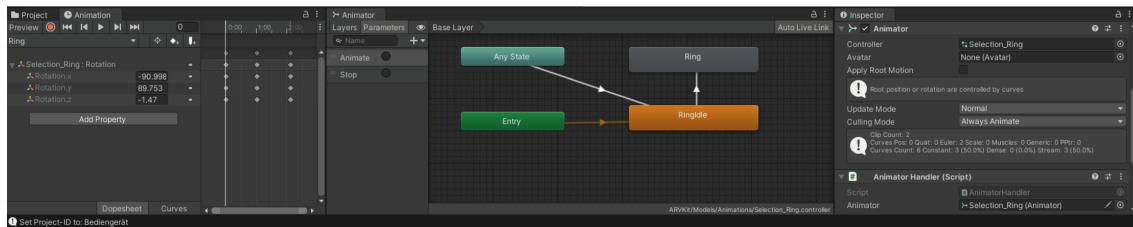


Abbildung 6-9: Erstellung und Integration von Animationen

Eine Anwendungsfunktionalität, die bereits im Rahmen der Prototypen-Integration vorzubereiten ist, ist die **Informationsdarstellung**. Über das Kontext-Menü werden einzelnen Teilmustern der Prototypen sogenannte *Meta-Infos* hinzugefügt. Dies erfolgt über Eingabefenster für Text, die als Teil des *Informationsdarstellungs*-Bausteins verwendet werden (Abbildung 6-10). Durch einen Button-Klick wird dem ausgewählten Teilmustell ein entsprechende *MetaInfo*-Komponente hinzugefügt. Wird das Teilmustell während der Validierung angeklickt, wird die entsprechende Information automatisch dargestellt.

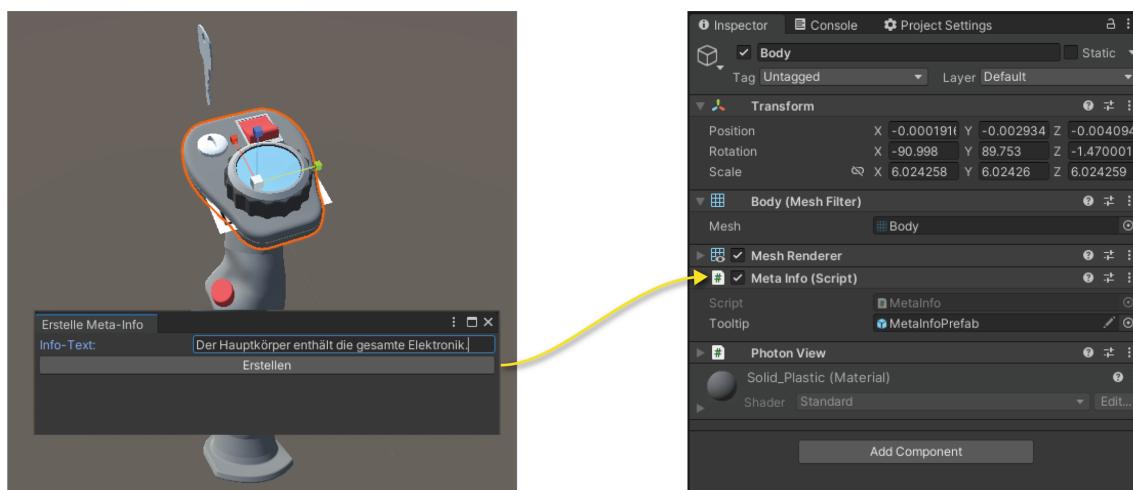


Abbildung 6-10: Erstellen von Meta-Infos über ein Eingabefenster

Schritt 6 – Anwendungsfunktionalität: Zusätzlich zu den bereits integrierten Funktionalitäten werden einzelne Anwendungsfunktionalitäten über das **Konfigurationsmenü** aktiviert oder deaktiviert. Für den vorliegenden Validierungsfall erfolgt entsprechend der AR-System-Konfiguration die links in Abbildung 6-11 dargestellte Funktionskonfiguration. Der Entwickler bzw. Experte kann hervorheben, verschieben und **animieren**. Für den

Kunden wird ebenfalls die Highlight-Funktionalität aktiviert. Zudem erhält der Kunde die Möglichkeit, **Feedback** abzugeben.

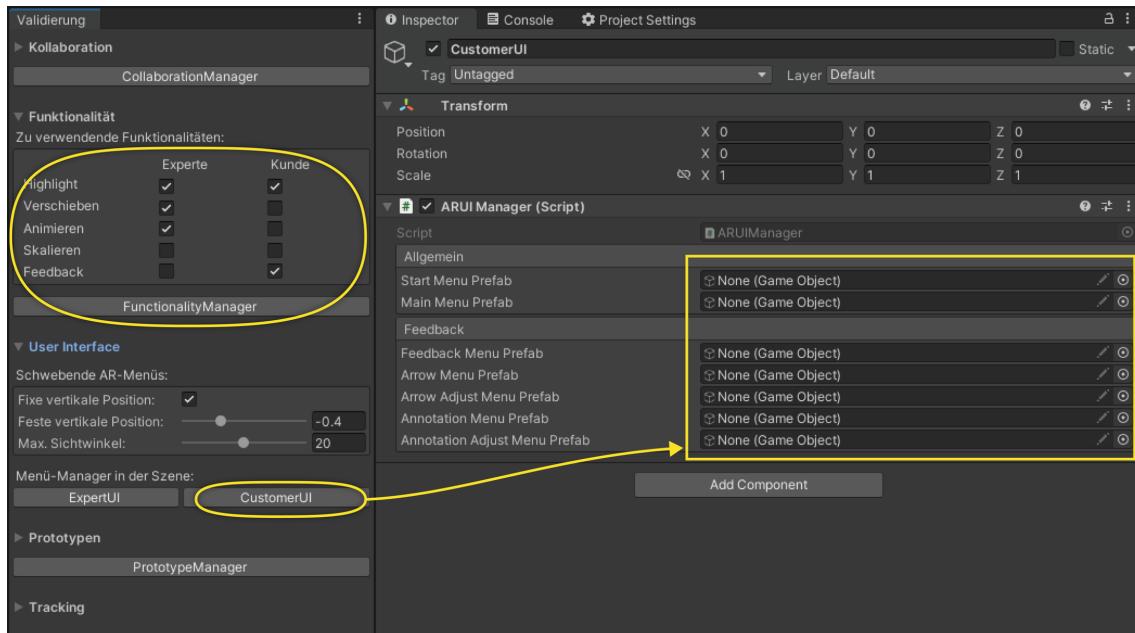


Abbildung 6-11: Funktionskonfiguration sowie Referenzierung der entsprechenden Prefabs für die Benutzerschnittstellen

Schritt 7 – Benutzerschnittstellen: Entsprechend der Funktionskonfiguration werden die Referenzen für die notwendigen Menü-Prefabs in den beiden Menü-Managern aktiviert. Ein Klick auf den „CustomerUI“-Button führt zum *ARUIManager*, der als Menü-Manager für den Kunden agiert (Abbildung 6-11, rechts). Hier sind standardmäßig Prefabs für das Start- sowie das Haupt-Menü zu referenzieren. Zudem sind fünf Menü-Prefabs als Benutzerschnittstellen im Kontext der Feedback-Funktion zu definieren. Die Highlight-Funktionalität benötigt kein spezielles Menü, sie erfolgt direkt aus dem Hauptmenü heraus. Aus dem Projektordner heraus werden die bereits im ARVKit exemplarisch enthaltenen Menü-Prefabs für die HoloLens 2 an den entsprechenden Stellen referenziert. Unter anderem werden die in Abbildung 5-40 dargestellten Menü-Prefabs verwendet. Entsprechend der AR-System-Konfiguration werden somit **Gestensteuerung** und **Speech-to-Text** verwendet. Alternativ können an dieser Stelle individuelle Menüs erstellt und referenziert werden. So sind sowohl das Design als auch die Art der Interaktion frei zu wählen.

Schritt 8 – Installation: Nach der Konfiguration der Anwendungsfunktionalität sowie der Referenzierung der Menü-Prefabs ist die AR-Validierungsanwendung bereit zur Installation. Dafür werden über das Unity-Menü die *Build Settings* aufgerufen (Abbildung 6-12, links). Die grundsätzlichen Einstellungen sind durch die Projektvorlage sowie das Szenen-Template bereits vorgenommen. In den Player Settings wird ein individueller Anwendungsname sowie ein Icon gesetzt. Anschließend wird die Anwendung über den Build-Button zunächst als Visual Studio-Projekt erstellt und anschließend über Visual

Studio auf der HoloLens installiert. Zudem wird die gleiche Anwendung für den Entwickler auf einem Desktop-Gerät installiert. Durch die zuvor beschriebenen Präprozessoranweisungen werden die jeweils nicht notwendigen Funktionalitäten und Benutzerschnittstellen automatisch beim Starten der Anwendung deaktiviert.

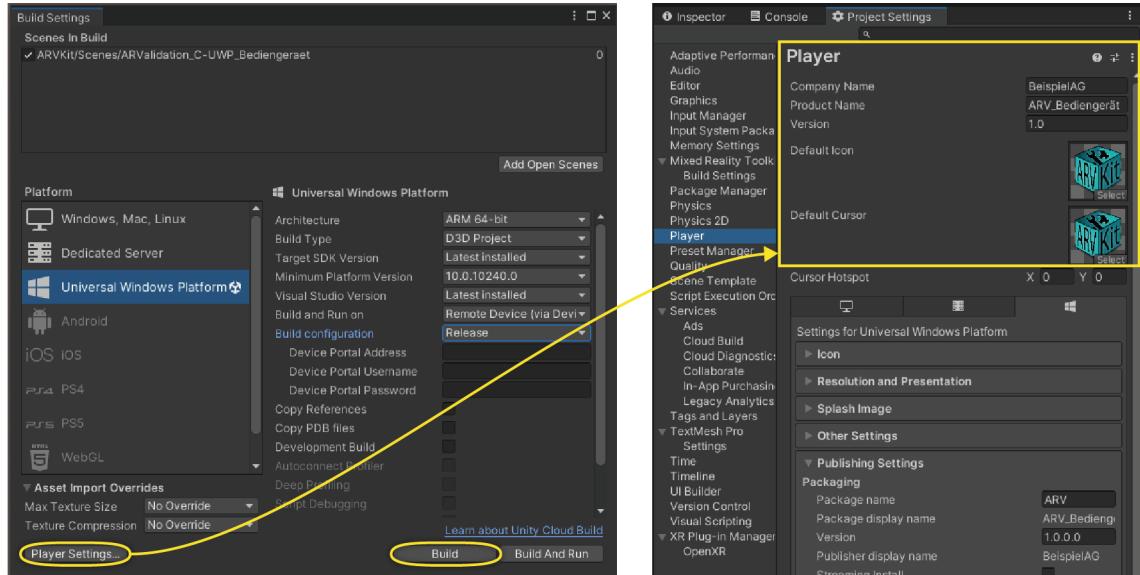


Abbildung 6-12: *Build Settings* und *Player Settings* zur Vorbereitung der Installation

Nach der Installation liegt das geplante und konfigurierte kollaborative AR-System als einsatzbereites Validierungswerkzeug vor. Grundsätzlich endet an dieser Stelle die Unterstützung des Instrumentariums. Zu besseren Veranschaulichung der Potentiale wird im Folgenden der **Einsatz der AR-Lösung zur Validierung der Bedienkonzepte** exemplarisch beschrieben. Der Ablauf sowie die Inhalte und Funktionalitäten entsprechen dabei der bereits im Projekt *DigiKAM* durchgeführten Validierung. Zunächst wird der 2D-Marker ausgedruckt und am 3D-gedruckten Griff angebracht (Abbildung 6-13, links). Die virtuellen Inhalte bzw. Prototypen werden entsprechend passend auf dem Griff visualisiert (Abbildung 6-13, rechts).

Nachdem die Anwendungen für den Experten und den Kunden gestartet wurden, erscheinen jeweils die **Startmenüs**, in denen die *Projekt-ID* eingegeben werden muss. Nach der Bestätigung verbinden sich die beiden Anwendungen und die Synchronisation der Perspektive wird gestartet. Entsprechend wird die Sicht des Kunden auf die virtuellen Prototypen auf die Visualisierung des Experten am Desktop übertragen (Abbildung 6-14). Der Entwickler sieht das Produkt, in diesem Fall das Regal-Bediengerät, entsprechend stets aus der Perspektive des Kunden. Über Audio-Kommunikation können der Entwickler und der Kunde miteinander reden.



Abbildung 6-13: 3D-gedruckter Griff mit angebrachtem Marker (links) und AR-Sicht des Kunden auf virtuelle Prototypen (rechts)¹⁷

Im weiteren Verlauf der Validierung kann der Kunde nun die verschiedenen auf dem realen Griff visualisierten Bedienkonzepte validieren. Durch AR hat der Kunde die Möglichkeit, die Ergonomie sowie die Benutzung der einzelnen Bedienelemente entsprechend der zu validierenden Anforderungen zu testen und zu bewerten. Nachdem dem Kunden, wie in Abbildung 6-14 zu sehen, zunächst eine erste Variante des Bedienkonzepts vorgestellt wurde, hat der Entwickler über das Varianten-Menü am oberen Bildschirm-Rand zu einer zweiten Variante gewechselt. Abbildung 6-15 zeigt darauf aufbauend eine Reihe an Screenshots der AR-Sicht des Kunden sowie der Sicht des Experten zur Veranschaulichung der Funktionalitäten und ihrer Anwendung im Kontext der Validierung des Regal-Bediengeräts:

1+2: Der Kunde hat den Hauptkörper des Bedienkonzepts angeklickt. Dieser wird entsprechend in Gelb hervorgehoben. Es erscheint die dazu vorhandene **Meta-Information** als Label (oben rechts) und das Hauptmenü wird angezeigt. Das Hauptmenü enthält entsprechend der Funktionskonfiguration Button für *Highlight*, *Feedback* und *Ausblenden*, wobei *Ausblenden* eine Standard-Funktion darstellt. Durch einen virtuellen Klick auf den **Highlight**-Button startet der Hauptkörper langsam in Rot zu blinken. Dies geschieht sowohl beim Kunden als auch beim Entwickler.

3+4: Der Entwickler hat die **manuelle Kamerakontrolle** aktiviert (Haken unten links). Er hat somit die Kontrolle über die Perspektive und kann gezielt auch kleine Modelle anklicken und auswählen. Nach Auswahl des Drehrads erscheint auch hier ein Label mit Zusatzinfos sowie das Entwickler-Hauptmenü. Da das Drehrad über eine Animation verfügt, enthält das Hauptmenü unter anderem einen Button „Animieren“. Dieser öffnet das Animationsmenü, das lediglich aus einem „Play“-Button besteht. Der Button startet die jeweilige **Animation** sowohl beim Entwickler als auch beim Kunden. Gleichzeitig wechselt der Button in einen „Stopp“-Button. Neben der Animation des Drehrads wurden eine

¹⁷ Die HoloLens 2 ermöglicht keine gleichzeitige Nutzung der Kamera für Marker-Tracking und Screenshots. Entsprechend wurden die Screenshots der AR-Sicht ohne Marker-Tracking aufgenommen.

Animation zur Veranschaulichung der Schlüssel-Funktionalität sowie des Kippschalters erstellt.

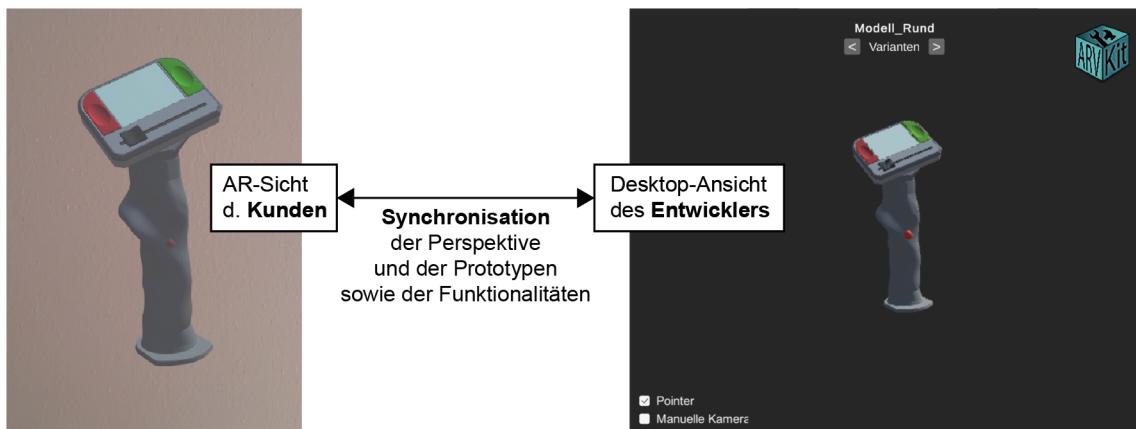


Abbildung 6-14: Synchronisation der AR-Sicht des Kunden zum Entwickler

5: Der Kunde hat das 3D-Pfeil-Menü geöffnet und erstellt über eine Drag-and-Drop-Geste einen **3D-Pfeil**. Der Pfeil kann anschließend virtuell gegriffen und verschoben werden und taucht gleichzeitig beim Entwickler auf.

6+7: Über den „Aufnahme“-Button des Annotations-Menüs kann der Kunde per *Speech-to-Text* eine **Annotation** erstellen. Sie taucht nach der Erstellung auch beim Entwickler auf und zeigt und dokumentiert die Meinung des Kunden zu einzelnen Produktmerkmalen. Die Annotationen und Pfeile unterstützen die Diskussion und dienen dem Entwickler als Grundlage für mögliche Anpassungen im Anschluss an die Validierung. Unter anderem findet der Kunde das Drehrad gut, da es auch gut mit Handschuhen entsprechend der Anforderung 4 zu bedienen ist. Zudem hat der Kunde festgestellt, dass der Knopf am Griff zu weit vorne positioniert und schwer zu bedienen ist (Anforderungen 5). Über den roten 3D-Pfeil hat der Kunde angezeigt, wo der Knopf besser positioniert wäre.

8+9: Über die Verschieben-Funktion kann der Entwickler den Knopf bereits während der Validierung an die gewünschte Stelle verschieben und diese vom Kunden live überprüfen lassen.

10: Sollte sich Feedback bereits während der Validierung erledigt haben, kann es durch die entsprechenden Menüs gelöscht werden.

Mit der basierend auf dem Werkzeugkoffer entwickelten AR-Validierungsanwendung lassen sich speziell die zuvor für die AR-basierte Validierung ausgewählten Anforderungen validieren. Weitere Anforderungen können selbstverständlich zusätzlich im Rahmen der Validierung betrachtet werden. Durch die dreidimensionale Darstellung in Kombination mit dem realen Griff lassen sich besonders gut die Aspekte der Bedienung und der Erreichbarkeiten im Kontext der Einhand-Bedienung überprüfen. Durch den Hersteller von Regal- und Lagersystemen als Kunden im *DigiKAM*-Projekt wurde die AR-basierte Validierung des Regal-Bediengeräts als sehr hilfreich und **vielversprechende Lösung** für die Zukunft bewertet.

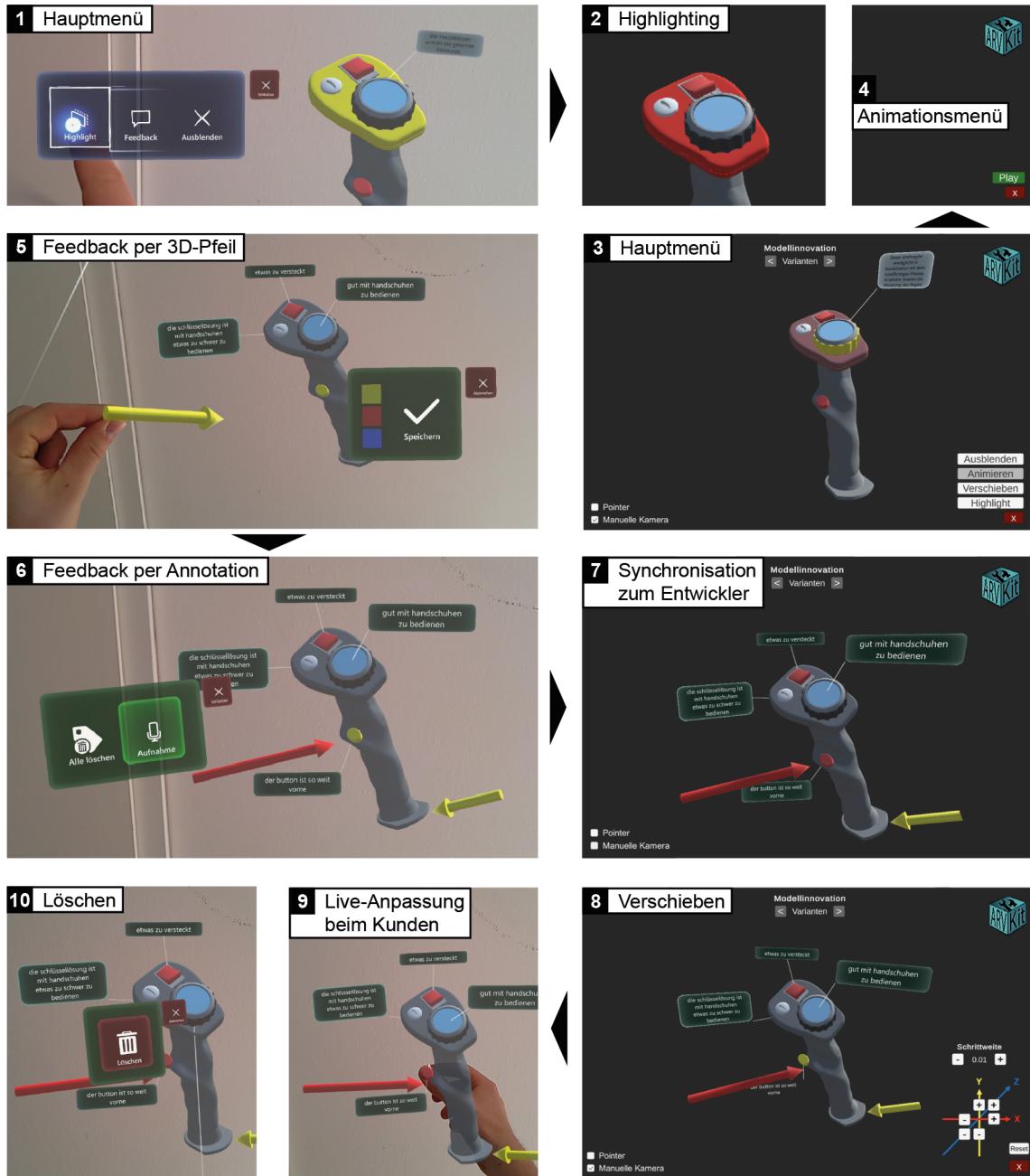


Abbildung 6-15: Screenshots der AR-Sicht sowie der Entwickler-Sicht zur Veranschaulichung der Validierung des Regal-Bediengeräts mittels AR

6.3 Bewertung des Instrumentariums

Abschließend wird das erarbeitete *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung* anhand der neun identifizierten Anforderungen aus Kapitel 3.6 bewertet. Zu diesem Zweck wird für jede Anforderung beschrieben, inwiefern sie durch das Instrumentarium und die darin enthaltenen Hilfsmittel erfüllt wird. Als Bewertungsgrundlage werden unter anderem Erkenntnisse

aus den Forschungsverbundprojekten *DigiKAM*, aus dem der exemplarische Anwendungsfallstammt, und *EMERGE* sowie Industrieprojekten herangezogen.

A1) Ganzheitlichkeit: Heutige Systeme vereinen unterschiedliche Disziplinen und umfassen entsprechend eine Vielzahl unterschiedlicher Produktmerkmale. Die sich daraus ergebenden vielfältigen Möglichkeiten für den Einsatz von AR als Validierungswerkzeug gilt es volumnäglich zu adressieren. Dies wurde erreicht, indem zunächst *grundsätzliche Funktionalitäten der Technologie AR* analysiert wurden. Diese wurden ergänzt durch Erkenntnisse anderer wissenschaftlicher Arbeiten, unter anderem den *AR-Anwendungsszenarien* nach RÖLTGEN. Aus den identifizierten Funktionalitäten wurden Potentiale zur Unterstützung der Validierung abgeleitet, die eine volumnägliche Erschließung der Potentiale durch die Unternehmen ermöglichen (vgl. Abschnitt 5.2.2.1).

A2) Generalisierung: Das Instrumentarium soll allgemeingültig und für verschiedene Systeme, Unternehmen und Branchen wiederverwendbar sein. Die dazu notwendige Generalisierung wird durch eine von den konkreten Anforderungen unabhängige Bewertung der Potentiale von AR für die Validierung erreicht. Die Bewertung erfolgt stattdessen bzgl. generischer Produktmerkmale, die unter anderem basierend auf den *Hauptmerkmale eines Produkts* nach PAHL ET AL. basieren. Durch die Bewertung der Relevanz dieser Produktmerkmale bzgl. der konkreten Anforderungen erfolgt die notwendige Generalisierung (vgl. Abschnitt 5.2.3.1).

A3) Korrektheit: Die das Instrumentarium anwendenden Unternehmen verfügen über kein oder lediglich geringes AR-Knowhow. Sie sind daher auf die Plausibilität und Korrektheit der Ergebnisse angewiesen. Um diese zu gewährleisten, wurden die Definitionen und Bewertungen zunächst initial durch den Autor basierend auf dem *Stand der Technik* definiert. Sie wurden anschließend gemeinsam mit *Experten* diskutiert und angepasst. Zudem wurden die Bewertungen anhand verschiedener Beispielanwendungen getestet und optimiert. Die Definitionen und Bewertungen stellen somit korrekte und für die Anwendung plausible Ergebnisse dar.

A4) Umfassende Dokumentation: Das für die AR-System-Konfiguration relevante Wissen wurde aus einer *Vielzahl unterschiedlicher Quellen* zusammengetragen. Es wurden unter anderem wissenschaftliche Arbeiten, technische Dokumentationen und Spezifikationen sowie Erfahrungsberichte herangezogen. Zur Modellierung aller Aspekte in einem gemeinsamen Modell wurde die *AR-System-Ontologie* erarbeitet. Entsprechend der Ontologie konnten auch komplizierte Zusammenhänge und Abhängigkeiten zwischen einzelnen Aspekten modelliert werden. Das gesamte gesammelte Wissen wurde entsprechend der Ontologie beschrieben und in die digitale *AR-System-Wissensbasis* überführt. Diese entspricht der umfassenden Dokumentation des relevanten Wissens.

A5) Erweiterbarkeit: AR ist eine Technologie, die sich teilweise noch in der Entwicklung befindet. Es kommen immer neue AR-Endgeräte an den Markt und neue Sensorik und Funktionalitäten werden integriert. Zudem wird immer neues Erfahrungswissen durch neue Anwendungsfälle gewonnen, das für die AR-System-Konfiguration relevant

ist. Entsprechend ist die Erweiterbarkeit der Wissensbasis von enormer Bedeutung für den langfristigen Einsatz des Instrumentariums. Die Erweiterbarkeit wird durch das in Abschnitt 5.3.1 beschriebene *Vorgehen zur Modellierung und Überführung des Wissens in eine maschinenlesbare Form* gewährleistet. In der zugrundeliegenden *Excel-Arbeitsmappe* kann neues Wissen aufwandsarm hinzugefügt werden. Es wird anschließend durch erneutes Einlesen mittels *Protégé* in das bestehende Wissen integriert. Zusätzlich kann durch neue Erkenntnisse implizit vorhandenes Wissen durch *Reasoner* automatisch identifiziert und der Wissensbasis explizit hinzugefügt werden.

A6) Zugänglichkeit: Das umfangreiche und teilweise komplizierte Wissen muss zugänglich sein, damit es Unternehmen einen Mehrwert bringt. Dies wird über das *interaktive Konfigurationswerkzeug* (Abschnitt 5.3.2) realisiert. Das Konfigurationswerkzeug analysiert das in der AR-System-Wissensbasis vorhandene Wissen für einen konkreten Validierungsfall automatisiert und zielgerichtet. So wird das Wissen entsprechend des individuellen Benutzer-Inputs von Software ausgewertet und ein geeignetes AR-Endgerät sowie zu nutzende Softwarefunktionalitäten vorgeschlagen. Es bedarf keiner manuellen Analyse. Somit ist das Wissen zugänglich für Unternehmen und kann aufwandsarm genutzt werden.

A7) Aufwandsarme Entwicklung: Für eine effiziente und aussagekräftige Validierung sind bedarfsgerechte AR-Validierungsanwendungen notwendig. Unternehmen müssen befähigt werden, diese Anwendungen eigenständig zu entwickeln. Dazu muss die Entwicklung aufwandsarm gestaltet werden. Der in Kapitel 5.4 beschriebene *Werkzeugkoffer* stellt verschiedene Hilfsmittel bereit, die Unternehmen bei der Entwicklung individueller AR-Validierungsanwendungen unterstützen und die Entwicklungsaufwände reduzieren. Kern des Werkzeugkoffers sind modulare Funktionsbausteine, die auf verschiedenen AR-Plattformen wiederverwendet werden können. Die Bausteine können angeleitet durch einen Leitfaden und unterstützt durch grafische Benutzeroberflächen und weitere Hilfsmittel flexibel zu individuellen Anwendungen kombiniert werden. Unternehmen werden so in die Lage versetzt, eigenständig und mit geringem Aufwand AR-Anwendungen als Validierungswerzeuge zu entwickeln.

A8) Systematische Vorgehensweise: Das in Kapitel 5.5 beschrieben Vorgehensmodell bildet einen systematischen Rahmen für den Einsatz des Instrumentariums zur Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung. Das Vorgehen ist unterteilt in den drei Handlungsfeldern entsprechenden Phasen der Nutzen-Aufwands-Bewertung, der AR-System-Konfiguration sowie der Anwendungsentwicklung. Die drei Phasen werden ergänzt durch die zwischenzeitlich nach der Nutzen-Aufwands-Analyse zu treffende Entscheidung für oder gegen AR als Validierungswerzeug. Für drei Hauptphasen werden zusätzliche Vorgehensweisen beschrieben, die Unternehmen im Detail in der Anwendung der Bestandteile des Instrumentariums anleiten. Dabei werden an verschiedenen Stellen auch die relevanten unternehmensspezifischen Anforderungen und Rahmenbedingungen berücksichtigt.

A9) Anwendbarkeit: Das Instrumentarium muss für Unternehmen ohne AR-Erfahrung und -Knowhow effizient anwendbar sein, um die Potentiale der AR-basierten Validierung zu nutzen. Bei der Erarbeitung der einzelnen Bestandteile wurde sich daher an verschiedenen Arbeiten aus dem Stand der Technik orientiert, darunter verschiedene Arbeiten von RÖLTGEN. Durch geeignete Hilfsmittel und eine starke *Automatisierung durch Software-Werkzeuge* werden die Aufwände stark reduziert, was in einem sehr guten *Nutzen-Aufwands-Verhältnis* resultiert. So erfolgt die Auswertung der Nutzen-Aufwands-Analyse semiautomatisiert durch Excel-Makros und die AR-System-Konfiguration durch das interaktive Konfigurationswerkzeug. Die Entwicklung der AR-Validierungsanwendungen wird zudem durch grafische Benutzeroberflächen unterstützt. Die Anwendung und Bewertung des Instrumentariums bestätigen die Anwendbarkeit (Kapitel 6.2).

Das vorgestellte *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung* erfüllt somit alle Anforderungen in vollem Umfang. Das Instrumentarium ermöglicht zunächst ausgehend von den an das Produkt gestellten Anforderungen eine individuelle Nutzen-Aufwands-Analyse von AR-Potentiälen. Darauf aufbauend unterstützt das Instrumentarium eine bedarfsgerechte AR-System-Konfiguration sowie die eigenständige Entwicklung von entsprechenden AR-Validierungsanwendungen auf einem ausgewählten AR-Endgerät. Das Instrumentarium wurde anhand des exemplarischen Anwendungsfalls „Regal-Bediengerät“ angewendet und evaluiert.

7 Zusammenfassung und Ausblick

Getrieben durch die Trends Digitalisierung, Globalisierung und Nachhaltigkeit vollzieht sich ein **Wandel der Marktleistungen sowie der Entstehungsprozesse**. Die Marktleistungen werden getrieben durch den stetigen technologischen Fortschritt immer komplexer und bieten immer neue Möglichkeiten. Heutige intelligente, cyber-physische Systeme, auch Advanced Systems genannt, vereinen Mechanik, Elektronik und Software in einem System. Durch das Zusammenspiel der verschiedenen Disziplinen kombinieren Advanced Systems vielfältige Produktmerkmale wie Optik, Gestalt, Funktionalität, Verhalten, Kinematik oder auch Ergonomie. Dies birgt großes Potential für Produktvarianten und eine individuelle Ausgestaltung des Produkts. So nimmt der bereits seit Jahren erkennbare, durch digitale Entwicklungswerzeuge getriebene Trend der Individualisierung im Kontext von Advanced Systems zu. Die Produktmerkmale werden gezielt für Kunden ausgeprägt, um individuelle Kundenanforderungen bestmöglich zu erfüllen.

In Kombination mit dem gleichzeitig steigenden Zeit- und Kostendruck auf den globalen Märkten führt die Individualisierung jedoch zu großen Herausforderungen in der Produktentwicklung. Unternehmen müssen ihre Entwicklungszyklen weiter verkürzen und ihre Produkte unter Berücksichtigung der individuellen Kundenanforderungen schneller an den Markt bringen, um weiterhin erfolgreich zu sein. Dies erfordert eine frühe und kontinuierliche **Validierung der Entwicklung unter Einbeziehung des Kunden**. Bei der Validierung werden die entwickelten Lösungen mit den Kundenanforderungen abgeglichen und sichergestellt, dass das richtige Produkt entwickelt wird. Klassische Validierungsansätze wie physische Prototypen sind nicht zeit- und kosteneffizient. Digitale Ansätze wie Simulationen ermöglichen dagegen eine Validierung subjektiv zu betrachtender Produktmerkmale. Die Einbindung des Kunden wird zudem durch eine zunehmende geografische Trennung der Akteure erschwert. Speziell im Kontext der komplexen und interdisziplinären Advanced Systems mangelt es daher bisher an geeigneten Validierungsansätzen. Mit **Augmented Reality (AR)** existiert eine innovative Visualisierungs- und Kommunikationstechnologie, die enorme Potentiale zur Adressierung der bestehenden Herausforderungen bietet. Mittels AR können digitale Daten und Modelle dreidimensional und immersiv in realen Umgebungen visualisiert und erlebt werden. AR ermöglicht so zum einen eine aussagekräftige Bewertung einzelner Produktmerkmale durch den Kunden. Zum anderen ermöglicht AR die digitale Vernetzung räumlich verteilter Akteure. Durch den Einsatz von AR als Validierungswerkzeug besteht somit das Potential, effiziente Validierungsaktivitäten standortübergreifend durchzuführen und den Kunden kontinuierlich in die Entwicklung einzubeziehen. Allerdings handelt es sich bei AR um eine grundsätzlich komplexe Technologie, die noch nicht in Unternehmen etabliert ist. Unternehmen sehen sich daher mit **Herausforderungen** in der Nutzung von AR konfrontiert. Speziell im Kontext der Validierung fällt es vielen Unternehmen aufgrund des Mangels an Knowhow und Erfahrung schwer, die Potentiale zu erkennen und zu nutzen.

Um den Herausforderungen zu begegnen und die Potentiale von AR im Kontext der Validierung zu nutzen, müssen drei **Handlungsfelder** erschlossen werden. Zunächst müssen Unternehmen bei der *systematischen Planung AR-basierter Validierung* unterstützt werden. Es gilt, Unternehmen zu befähigen, Potentiale von AR zur Validierung von Produkten oder Produktmerkmalen eigenständig zu identifizieren und den entsprechenden Einsatz von AR als Validierungswerkzeug zu planen. Im Rahmen der weiteren Planung sind zudem bedarfsgerechte AR-Systeme zu konfigurieren. Für diese Konfiguration ist umfangreiches Wissen aus verschiedensten Quellen relevant. Dieses Wissen muss gesammelt, formalisiert und im Rahmen einer *wissensbasierten AR-System-Konfiguration* Unternehmen zugänglich gemacht werden. Das dritte Handlungsfeld betrifft die *Entwicklung individueller AR-Validierungsanwendungen*. Entsprechend der AR-System-Konfiguration sind Unternehmen zu befähigen, bedarfsgerechte AR-Validierungsanwendungen auf ausgewählten AR-Endgeräten eigenständig zu entwickeln und so die notwendigen Validierungswerkzeuge vorzubereiten.

Im Rahmen dieser Arbeit wurden vielfältige existierende Ansätze zur systematischen Planung, zu wissensbasierten Systemen und System-Konfiguration sowie zur aufwandsarmen Entwicklung von AR-Anwendungen untersucht. Die betrachteten Ansätze unterstützen lediglich einzelne Aspekte der Handlungsfelder oder sind für andere Anwendungsbereiche konzipiert. Potentiale von AR werden grundsätzlich in verschiedenen übergreifenden Publikationen zur Technologie oder in spezifischen Anwendungsberichten beschrieben. Systematische Ansätze zur Untersuchung von Potentialen wie die *Nutzen-Aufwands-Bewertung* nach RÖLTGEN oder das *Agile Practices Impact Model (APIM)* nach DIEBOLD UND ZEHLER sind nicht auf AR-basierte Validierungswerkzeuge bezogen. Für die AR-System-Konfiguration relevantes Wissen steht in einer Vielzahl unterschiedlicher, größtenteils unstrukturierter Quellen zur Verfügung. Gesammeltes Wissen lässt sich unter anderem über *semantische Technologien* formalisieren und digital verarbeiten. So ermöglichen *wissensbasierte Systeme* eine automatisierte Verarbeitung und Analyse von Wissen durch Softwarealgorithmen. Es bestehen bereits verschiedene Ansätze zur AR-System-Konfiguration, unter anderem durch RÖLTGEN, FELLMANN ET AL. und SCHILLING. Diese Ansätze berücksichtigen jedoch nicht alle relevanten Aspekte und bedürfen zudem weiterhin vorheriger Expertise im Themenfeld AR. Ein vielversprechender Ansatz, der auf semantischen Technologien beruht, wird von BAUER im Kontext *Produktionssystemkonzipierung* beschrieben. Zur Unterstützung der AR-Anwendungsentwicklung stehen zahlreiche *Software Development Kits (SDK)* zur Verfügung. SDKs stellen jedoch nur grundlegende Funktionalitäten als Grundlage der Programmierung bereit. Im Forschungsprojekt *AcRoSS* wurde daher versucht, AR-Anwendungen ohne Programmierung basierend auf modularen Bausteinen zu erstellen. Usability-Untersuchungen von PAES UND IRIZARRY beschreiben zudem wichtige Aspekte, die es bei AR-Anwendungen zu berücksichtigen gilt. Einer Vielzahl weiterer *Erfahrungs- und Anwendungsberichten* sowie *Studien* können hilfreiche Erkenntnisse und Anregungen für die funktionale Gestaltung und Realisierung von AR-Validierungsanwendungen entnommen werden. Keiner der un-

tersuchten Ansätze entspricht einer vollumfänglichen Lösung zur Unterstützung der Planung und Vorbereitung AR-basierter Validierung. Daher besteht Handlungsbedarf für ein übergreifendes *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung*.

Das entwickelte Instrumentarium nutzt einzelne Ansätze aus dem Stand der Technik, ergänzt oder kombiniert sie und überträgt sie bei Bedarf auf den Kontext Augmented Reality-basierte Validierung. Das Ergebnis ist ein Instrumentarium, das sich ergänzend zu den vorhandenen Entwicklungsprozessen und damit verbundenen Vorgehensmodellen einsetzen lässt. Ausgehend von den Anforderungen an das zu entwickelnde Produkt unterstützt das Instrumentarium Unternehmen bei der Entscheidung, ob AR als Validierungswerkzeug sinnvoll ist, sowie bei der anschließenden Planung und Vorbereitung der AR-basierten Validierung. Dazu stellt das Instrumentarium die folgenden vier **Lösungsbausteine** bereit:

- Eine **Systematik zur Nutzen-Aufwands-Analyse von AR-Potentialen**, die basierend auf einem AR-Potential-Bewertungsgraph sowie einem interaktiven Hilfsmittel die individuelle Identifikation und Bewertung von Potentialen unterstützt,
- eine auf einer AR-System-Wissensbasis aufbauenden und über ein interaktives Konfigurationswerkzeug realisierten **wissensbasierte AR-System-Konfiguration** zur Auswahl eines geeigneten AR-Endgeräts sowie der bedarfsgerechten Konzipierung der Anwendungssoftware,
- ein **Werkzeugkoffer zur aufwandsarmen Entwicklung individueller AR-Validierungsanwendungen**, der wiederverwendbare Vorlagen, Funktionsbausteine und weitere Hilfsmittel für die Entwicklungsumgebung Unity sowie einen Leitfaden zur Anleitung der Entwicklung bereitstellt, und
- ein **übergeordnetes Vorgehensmodell**, das den Einsatz der zuvor beschriebenen Lösungsbausteine zur Planung und Vorbereitung AR-basierter Validierung systematisch anleitet.

Anhand der Validierung eines Regal-Bediengeräts als exemplarischem Anwendungsfall aus einem Forschungsprojekt wurde das Instrumentarium angewendet und bewertet. Dazu wurden die Phasen des Vorgehensmodells vollständig durchlaufen und die Hilfsmittel aus den Lösungsbausteinen für die Planung und Vorbereitung der AR-basierten Validierung im Kontext des exemplarischen Anwendungsfalls angewendet. Die exemplarische Anwendung und Bewertung haben gezeigt, dass das *Instrumentarium zur Planung und Vorbereitung Augmented Reality-basierter Validierung in der Produktentwicklung* die gestellten Anforderungen in vollem Umfang erfüllt.

Die Entwicklung des Instrumentariums erfolgte grundsätzlich entsprechend der Design Research Methodology nach BLESSING UND CHAKRABARTI (vgl. Kapitel 2.2 und 2.3). Die vier Phasen der Methodik wurden komplett durchlaufen. Zudem wurde die Design Research Methodology zusätzlich zur Entwicklung der Bestandteile des Instrumentariums in

verschiedenen Forschungs- und Industrieprojekten angewendet. Teilergebnisse aus diesen Projekten wurden bereits veröffentlicht und auf wissenschaftlichen Konferenzen präsentiert. Durch das systematische Forschungsdesign konnte eine wissenschaftlich fundierte Entwicklung gewährleistet werden. Durch die Berücksichtigung der Anforderungen und Bedarfe aus der Praxis, unter anderem basierend auf der im Rahmen der ersten deskriptiven Studie durchgeführten Unternehmensbefragung, konnten praxistaugliche Lösungen sichergestellt werden.

Die von HEVNER für jede Forschungsaktivität als notwendig angesehenen drei Design Research Zyklen (s. Kapitel 2.1) können eindeutig in dieser Arbeit identifiziert werden. Der *Relevanz-Zyklus* zur Verknüpfung der Design-Aufgabe mit dem Umfeld erfolgte ausführlich durch die Problemanalyse (Kapitel 3). Entsprechende durch das Umfeld beeinflusste Handlungsfelder sowie Anforderungen an das Instrumentarium wurden identifiziert. Die Erarbeitung der Wissensbasis im Rahmen des *Strenge-Zyklus* als Grundlage der Entwicklung erfolgte durch die ausführliche Erarbeitung des Stands der Technik (Kapitel 4). Der *Design-Zyklus* entspricht darauf aufbauend der Entwicklung des Instrumentariums und seiner Bestandteile (Kapitel 5) sowie deren Evaluierung anhand des exemplarischen Anwendungsfalls (Kapitel 6).

Bezüglich der Planung und Vorbereitung AR-basierter Validierung in der Produktentwicklung besteht weiterer **Forschungsbedarf**. Zunächst wird bei Potentialbewertung bisher vorausgesetzt, dass die Validierung grundsätzlich digital durchgeführt werden soll. Ein direkter Vergleich mit dem Einsatz klassischer physischer Prototypen wird von der Systematik zur Nutzen-Aufwands-Bewertung von AR-Potentialen bislang nicht adressiert. Dieser Aspekt sollte zukünftig integriert werden. Zudem ist eine Erweiterung des Instrumentariums auf Virtual Reality (VR) zu prüfen. Aufgrund der Ähnlichkeit der Technologien bietet sich diese an und wäre für alle Lösungsbauusteine grundsätzlich denkbar. Im Rahmen der AR-System-Konfiguration gilt es zukünftig zunächst, weiteres relevantes Wissen und neue Erkenntnisse in die AR-System-Wissensbasis zu integrieren. Im Falle einer Erweiterung des Instrumentariums auf den Einsatz von VR müssten zudem notwendige Anpassungen an der System-Ontologie, der System-Wissensbasis sowie dem Konfigurationswerkzeug identifiziert und vorgenommen werden. Grundsätzlich sind im Werkzeugkoffer zur aufwandsarmen Entwicklung weitere Funktionsbausteine zu konzipieren und zu realisieren. Zudem gilt es, bei Veröffentlichung neuer AR-Endgeräte entsprechende Templates zu integrieren und den Einsatz neuer SDKs zu prüfen. Gleiches gilt für eine mögliche Erweiterung auf VR.

8 Abkürzungsverzeichnis

AM	Additive Manufacturing
API	Application Programming Interface
APIM	Agile Practices Impact Model
AR	Augmented Reality
ARVKit	AR Validation Framework
ABox	Assertional Box
ATS	Assemble-to-order
B2B	Business-to-business
B2C	Business-to-customer
bzgl.	bezüglich
bspw.	beispielsweise
bzw.	beziehungsweise
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CARV	City-of-Augmented-Reality-Validation
CODP	Customer-order-decoupling-point
DaD	Drag-and-Drop
DMU	Digital Mockup
DRM	Design Research Methodology
ETO	Engineer-to-order
HF	Handlungsfeld
HMD	Head-mounted Display
IMU	Intertial Measurement Unit
inkl.	inklusive
ITS	Intelligente technische Systeme
JT	Jupiter Tessellation
LB	Lösungsbaustein

MRTK	Mixed Reality Toolkit
MTO	Make-to-order
MTS	Make-to-stock
OWL	Web Ontology Language
QFD	Quality Function Deployment
PSS	Produkt-Service-System
RP	Referenzprofil
s.	siehe
SDK	Software Development Kit
SDLC	Software Development Lifecycle
SLAM	Simultaneous Localization and Mapping
SysML	Systems Modelling Language
TBox	Terminological Box
u.a.	unter anderem
UI	User Interface
UML	Unified Modelling Language
UWP	Universal Windows Platform
vgl.	vergleiche
VR	Virtual Reality
z.B.	zum Beispiel

9 Literaturverzeichnis

- [ABK+16] ALBERS, A.; BEHRENDT, M.; KLINGLER, S.; MATROS, K.: Verifikation und Validierung im Produktentstehungsprozess. In: Lindemann, U. (Hrsg.): Handbuch Produktentwicklung. Hanser, München, 2016, S. 541–569
- [AD10] ALBERS, A.; DÜSER, T.: Implementation of a vehicle-in-the-loop development and validation platform. 33. FISITA World Automotiv Congress, Budapest, 2010
- [AG15] AMIN, D.; GOVILKAR, S.: Comparative study of augmented reality SDKs: International Journal on Computational Sciences & Applications (IJCSA). Band 5, 2015
- [AHM+19] ALBERS, A.; HEIMICKE, J.; MÜLLER, J.; SPADINGER, M.: Agility and its Features in Mechatronic System Development: A Systematic Literature Review: ISPIM Innovation Conference – Celebrating Innovation: 500 Years Since daVinci. ISPIM, 16.-19. Juni, Florence, Italy, 2019
- [AKN+19] ALIPRANTIS, J.; KONSTANTAKIS, M.; NIKOPOULOU, R.; MYLONAS, P.; CARIDAKIS, G.: Natural Interaction in Augmented Reality Context. VIPERC@ IRCDL, 2019
- [Alb10] ALBERS, A.: Five Hypotheses about Engineering Processes and their Consequences. Proceedings of the TMCE 2010, Ancona, I, April 12-16, 2010, 2010
- [Alt02] ALT, T.: Augmented Reality in der Produktion. Dissertation, Universität Magdeburg, 2002
- [Ana15] ANACKER, H.: Instrumentarium für einen lösungsmusterbasierten Entwurf fortgeschritten mechatronischer Systeme. Dissertation, Heinz Nixdorf Institut, Universität Paderborn, 2015
- [AO13] ARBELÁEZ-ESTRADA, J. C.; OSORIO-GÓMEZ, G.: Augmented Reality Application for Product Concepts Evaluation. Procedia Computer Science, 25, 2013, S. 389–398
- [App22a-ol] APPLE: Creating 3D Content with Reality Composer | Apple Developer Documentation. Unter: https://developer.apple.com/documentation/realitykit/creating_3d_content_with_reality_composer, 27. Juni 2024
- [App22b-ol] APPLE: iPad Pro. Unter: <https://www.apple.com/de/ipad-pro/>, 27. Juni 2024
- [AR11] ABELE, E.; REINHART, G.: Zukunft der Produktion – Herausforderungen, Forschungsfelder, Chancen. Carl Hanser Fachbuchverlag, s.l., 2011
- [ARC18] ARC ADVISORY GROUP: Augmented Reality für die Fertigung – Digitaler Wandel für Fachkräfte, 2018
- [ARC18-ol] ARC ADVISORY GROUP: Maximizing Digital Transformation with Augmented and Virtual Reality | ARC Advisory. Unter: <https://www.arcweb.com/blog/maximizing-digital-transformation-augmented-virtual-reality>, 14. Mai 2020
- [ARF+19] ALBERS, A.; REINEMANN, J.; FAHL, J.; HIRSCHTER, T.: Augmented Reality for Product Validation: Supporting the Configuration of AR-Based Validation Environments. In: Chen, J. Y.; Fragomeni, G. (Hrsg.): Virtual, Augmented and Mixed Reality. Applications and Case Studies. Lecture Notes in Computer Science, Springer International Publishing, Cham, 2019, S. 429–448
- [Azu97] AZUMA, R. T.: A survey of augmented reality. Presence: Teleoperators & Virtual Environments, (6)4, 1997, S. 355–385
- [Bau15] BAUER, F.: Planungswerkzeug zur wissensbasierten Produktionssystemkonzipierung. Dissertation, Heinz Nixdorf Institut, Universität Paderborn, 2015
- [BB11-ol] BECKETT, D.; BERNERS-LEE, T.: Turtle - Terse RDF Triple Language – W3C Team Submission 28 March 2011. Unter: <https://www.w3.org/TeamSubmission/turtle/>, 27. Juni 2024
- [BBM+19] BRUNO, F.; BARBIERI, L.; MARINO, E.; MUZZUPAPPA, M.; D'ORIANO, L.; COLACINO, B.: An augmented reality tool to detect and annotate design variations in an Industry 4.0 approach. The International Journal of Advanced Manufacturing Technology, (105)1, 2019, S. 875–887
- [BC09] BLESSING, L.; CHAKRABARTI, A.: DRM: A Design Reseach Methodology. Springer, London, 2009

- [BCC+09] BORDEGONI, M.; CUGINI, U.; CARUSO, G.; POLISTINA, S.: Mixed prototyping for product assessment: a reference framework. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, (3)3, 2009, S. 177–187
- [BCL15] BILLINGHURST, M.; CLARK, A.; LEE, G.: A Survey of Augmented Reality. *Foundations and Trends® in Human–Computer Interaction*, (8)2-3, 2015, S. 73–272
- [BCT07] BREITMAN, K.; CASANOVA, M. A.; TRUSZKOWSKI, W.: *Semantic Web: Concepts, Technologies and Applications*. Springer London, London, 2007
- [BEP+16] BACKHAUS, K.; ERICHSON, B.; PLINKE, W.; WEIBER, R.: *Multivariate Analysemethoden – Eine anwendungsorientierte Einführung*. 14. Auflage, Springer Gabler, Berlin, 2016
- [Bey13] BEYERER, J.: *visIT – Interoperabilität*, Fraunhofer IOSB, Karlsruhe, 2013
- [BG14-ol] BRICKLEY, D.; GUHA, R. V.: *RDF Schema 1.1 – W3C Recommendation*. Unter: <https://www.w3.org/TR/rdf-schema/>, 27. Juni 2024
- [BG21] BENDER, B.; GERICKE, K.: Entwickeln der Anforderungsbasis: Requirements Engineering. In: Bender, B.; Gericke, K. (Hrsg.): *Pahl/Beitz Konstruktionslehre – Methoden und Anwendung erfolgreicher Produktentwicklung*. 9. Auflage, Springer Berlin Heidelberg, Berlin, Heidelberg, 2021, S. 169–209
- [BGS+14] BACKHAUS, K.; GAUSEMEIER, J.; STÖCKLEIN, J.; JASPER, J.; WESTHOFF, K.; GRAFE, M.: *VR-basierte Conjoint-Analyse zur frühzeitigen Ermittlung von Kundenpräferenzen. Digitales Engineering zum Planen, Testen und Betreiben technischer Systeme – im Rahmen der 17. IFF Wissenschaftstage*, Magdeburg, 2014
- [BGS14-ol] BECKETT, D.; GANDON, F.; SCHREIBER, G.: *RDF 1.1 XML Syntax – W3C Recommendation*. Unter: <http://www.w3.org/TR/rdf-syntax-grammar/>, 27. Juni 2024
- [BH19-ol] BOUVERET, C.; HUMAN, S.: *Augmented Reality in der Industrie: Herausforderungen, Potenziale, Chancen*. Unter: <https://www.industry-of-things.de/augmented-reality-in-der-industrie-herausforderungen-potenziale-chancen-a-882695/>, 27. Juni 2024
- [BHL01] BERNERS-LEE, T.; HENDLER, J.; LASSILA, O.: The Semantic Web – A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. In: SCIENTIFIC AMERICAN (Hrsg.): *Scientific American*. Georg von Holtzbrinck Publishing Group, New York, NY, 2001, S. 34–43
- [BHL07-ol] BRAUN, S. C.; HELLENBRAND, D.; LINDEMANN, U.: *Kostentransparenz in der Mechatronik - Eine Studie über Komplexitäts- und Kostentreiber mechatronischer Produkte*. Unter: <https://www.shaker.de/de/content/catalogue/index.asp?lang=de&ID=8&ISBN=OND-00000-0000004>, 27. Juni 2024
- [Bil18-ol] BILYK, V.: *8 Challenges with Augmented Reality App Development*. Unter: <https://the-appsolutions.com/blog/development/augmented-reality-challenges/>, 14. Mai 2020
- [bit21] BITKOM - BUNDESVERBAND INFORMATIONSWIRTSCHAFT, TELEKOMMUNIKATION UND NEUE MEDIEN E.V (Hrsg.): *Augmented und Virtual Reality – Potenziale und praktische Anwendung immersiver Technologien*, 2021
- [BLF17] BOGNER, E.; LÖWEN, U.; FRANKE, J.: Systematic Consideration of Value Chains with Respect to the Timing of Individualization. *Procedia CIRP*, (60), 2017, S. 368–373
- [BO20] BLAND, D. J.; OSTERWALDER, A.: *Testing business ideas*. John Wiley & Sons Inc, Hoboken New Jersey, 2020
- [Bög18-ol] BÖGEMANN, I.: *Produktlebenszyklen werden immer kürzer, Ihre Entwicklungszeiten auch?* Unter: <https://blog.mb-collaborations.com/de/produktlebenszyklen-kuerzer-entwicklungszeiten-geringer>, 27. Juni 2024
- [Bog19] BOGNER, E.: *Strategien der Produktindividualisierung in der produzierenden Industrie im Kontext der Digitalisierung*. Dissertation, Friedrich-Alexander-Universität, 2019
- [Boy17-ol] BOYAJIAN, L.: *The 3 biggest challenges facing augmented reality*. Unter: <https://www.networkworld.com/article/3174804/the-3-biggest-challenges-facing-augmented-reality.html>, 27. Juni 2024
- [BP06] BLUMAUER, A.; PELLEGRINI, T.: *Semantic Web und semantische Technologien – Zentrale Begriffe und Unterscheidungen*. In: Pellegrini, T.; Blumauer, A. (Hrsg.): *Semantic Web – Wege zur vernetzten Wissensgesellschaft*. X.media.press, Springer, Dordrecht, 2006

- [BP13] BHUVANESWARI, T.; PRABAHARAN, S.: A survey on software development life cycle models: International Journal of Computer Science and Mobile Computing (IJCSMC) – A Monthly Journal of Computer Science and Information Technology. 2. Auflage, 2013
- [Bri21-ol] BRICKNELL, C.: Creating a Simple AR App. Unter: <https://msd-makerspaces.gitbook.io/next-lab/augmented-reality/guides/unity-and-vuforia-1/vuforia-primer>, 27. Juni 2024
- [Bro19] BROLL, W.: Augmentierte Realität. In: Dörner, R.; Broll, W.; Grimm, P.; Jung, B. (Hrsg.): Virtual und Augmented Reality (VR/AR) – Grundlagen und Methoden der Virtuellen und Augmentierten Realität. 2. Auflage, Springer Berlin Heidelberg, Imprint: Springer Vieweg, Berlin, 2019, S. 315–356
- [BSH06] BAUER, H. H.; STOKBURGER, G.; HAMMERSCHMIDT, M.: Marketing Performance – Messen - Analysieren - Optimieren. Gabler, Wiesbaden, 2006
- [BTR19] BERNHARD KÖLML; THOMAS PFEFFERLE; REBECCA BULANDER: Mega-Trend Individualisierung: Personalisierte Produkte und Dienstleistungen am Beispiel der Verpackungsbranche: Dialogmarketing Perspektiven 2018/2019. Springer Gabler, Wiesbaden, 2019, S. 243–260
- [Buc19-ol] BUCKLEY, D.: How to Create an Augmented Reality App in Unity. Unter: <https://vrgamedevelopment.pro/how-to-create-an-augmented-reality-app-in-unity/>, 27. Juni 2024
- [Bun15] BUNDESMINISTERIUM FÜR ARBEIT UND SOZIALES: Grünbuch Arbeiten 4.0 – Arbeit weiter denken, 2015
- [BV16] BERG, L. P.; VANCE, J. M.: An Industry Case Study: Investigating Early Design Decision Making in Virtual Reality. Journal of Computing and Information Science in Engineering, (17)1, 2016
- [BWY+10] BASSANINO, M.; WU, K.-C.; YAO, J.; KHOSROSHAHI, F.; FERNANDO, T.; SKJÆRBÆK, J.: The Impact of Immersive Virtual Reality on Visualisation for a Design Review in Construction: 2010 14th International Conference Information Visualisation. IEEE, 2010
- [Cap18-ol] CAPGEMINI: AR VR Virtual Augmented Reality Unternehmen Fertigung Betrieb Automotive. Unter: <https://www.capgemini.com/de-de/news/augmented-virtual-reality-ar-vr-studie/>, 14. Mai 2020
- [Cap19] CAPODIECI, I.: Markenbedeutungen im Zeitalter der Individualisierung. Journal für korporative Kommunikation, 2019, S. 64–74
- [CBH+17] CHATZOPoulos, D.; BERMEJO, C.; HUANG, Z.; HUI, P.: Mobile Augmented Reality Survey: From Where We Are to Where We Go. IEEE Access, (5), 2017, S. 6917–6950
- [CCG+14] CARICATO, P.; COLIZZI, L.; GNINI, M. G.; GRIECO, A.; GUERRIERI, A.; LANZIOTTO, A.: Augmented reality applications in manufacturing: a multi-criteria decision model for performance analysis. IFAC Proceedings Volumes, (47)3, 2014, S. 754–759
- [CD17] CHRISTOFORAKOS, L.; DIEFENBACH, S.: Erfolgreiches Prototyping im Ideenstadium der Produktentwicklung. Gesellschaft für Informatik e.V, 2017
- [CFJ20] CORINNA WINKLER; FRANZiska M. BONGERS; JAN H. SCHUMANN: Kundenintegration und Individualisierung bei digitalen Dienstleistungsinnovationen – Entwicklung eines Methodenbaukastens und Strategietoolkits: Dienstleistungsinnovationen durch Digitalisierung. Springer Gabler, Berlin, Heidelberg, 2020, S. 3–47
- [CNL+17] CHANG, Y. S.; NUERNBERGER, B.; LUAN, B.; HÖLLERER, T.; O'DONOVAN, J.: Gesture-based augmented reality annotation: 2017 IEEE Virtual Reality (VR), 2017, S. 469–470
- [Col12-ol] COLUMBIA UNIVERSITY - COMPUTER GRAPHICS AND USER INTERFACE LAB: MARS - Mobile Augmented Reality Systems. Unter: <https://graphics.cs.columbia.edu/projects/mars/>, 27. Juni 2024
- [Cra13] CRAIG, A. B.: Understanding augmented reality – Concepts and applications. Morgan Kaufmann, Waltham, MA, 2013
- [CWC+19] CHEN, Y.; WANG, Q.; CHEN, H.; SONG, X.; TANG, H.; TIAN, M.: An overview of augmented reality technology. Journal of Physics: Conference Series, (1237), 2019, S. 22082

- [DAG+21] DUMITRESCU, R.; ALBERS, A.; GAUSEMEIER, J.; STARK, R.; RIEDEL, O. (Hrsg.): Advanced Systems Engineering – Wertschöpfung im Wandel – Leistungsstand des Engineering in Wirtschaft und Wissenschaft. Paderborn, 2021
- [Dar21-ol] DARKRIFT NETWORKING: DarkRift 2. Unter: <https://www.darkriftnetworking.com/>, 1. Februar 2022
- [Dav15] DAVIES, R.: Industry 4.0 – Digitalisation for productivity and growth. EPRS - European Parliamentary Research Service, 2015
- [DBG+19] DÖRNER, R.; BROLL, W.; GRIMM, P.; JUNG, B. (Hrsg.): Virtual und Augmented Reality (VR/AR) – Grundlagen und Methoden der Virtuellen und Augmentierten Realität. 2. Auflage, Springer Berlin Heidelberg; Imprint: Springer Vieweg, Berlin, 2019
- [DBJ+19] DÖRNER, R.; BROLL, W.; JUNG, B.; GRIMM, P.; GÖBEL, M.: Einführung in Virtual und Augmented Reality. In: Dörner, R.; Broll, W.; Grimm, P.; Jung, B. (Hrsg.): Virtual und Augmented Reality (VR/AR) – Grundlagen und Methoden der Virtuellen und Augmentierten Realität. 2. Auflage, Springer Berlin Heidelberg; Imprint: Springer Vieweg, Berlin, 2019, S. 1–42
- [Deu07] DEUTSCH, J.-O.: Entwicklung einer Infrastruktur zur regelbasierten Verarbeitung von RDF/OWL Spezifikationen. Masterarbeit, Universität Hildesheim, 2007
- [Deu09] Projektmanagement - Projektmanagementsysteme - Teil 5: Begriffe, 2009
- [DGD+05] DJURIĆ, D.; GAŠEVIĆ, D.; DEVEDŽIĆ, V.; DAMJANOVIĆ, V.: A UML Profile for OWL Ontologies. In: Aßmann, U.; Aksit, M. (Eds.): Model driven architecture – European MDA Workshops: Foundations and Applications, MDAFA 2003 and MDAFA 2004, Twente, The Netherlands, June 26 - 27, 2003 and Linköping, Sweden, June 10 - 11, 2004 ; revised selected papers. Lecture Notes in Computer Science, 3599, Springer, Berlin, 2005, pp. 204–219
- [DGO+19] DÖRNER, R.; GEIGER, C.; OPPERMANN, L.; PAELKE, V.; BECKHAUS, S.: Interaktionen in Virtuellen Welten. In: Dörner, R.; Broll, W.; Grimm, P.; Jung, B. (Hrsg.): Virtual und Augmented Reality (VR/AR) – Grundlagen und Methoden der Virtuellen und Augmentierten Realität. 2. Auflage, Springer Berlin Heidelberg; Imprint: Springer Vieweg, Berlin, 2019, S. 219–265
- [DH17] DIEFENBACH, S.; HASSENZAHL, M.: Psychologie in der nutzerzentrierten Produktgestaltung – Mensch-Technik-Interaktion-Erlebnis. Die Wirtschaftspsychologie, Springer, Berlin, 2017
- [DHD+07] DRUMMOND, N.; HORRIDGE, M.; DAMERON, O.; RECTOR, A.; WANG, H.: A Practical Introduction to Protégé OWL, University of Manchester, 2007
- [DKM+19] DÖRNER, R.; KUHLEN, T. W.; MATTHYS, G.; BOGEN, M.; RILLING, S.; GERNDT, A.; DODIYA, J.; HERTKORN, K.; HULIN, T.; HUMMEL, J.; SAGARDIA, M.; WOLFF, R.; BERNSTEIN, A.-C.; UTZIG, S.; KÜHNERT, T.; BRUNNETT, G.; BLUM, L.; MENK, C.; BADE, C.; SCHREIBER, W.; ALEXANDER, T.; KLEIBER, M.; OPPERMANN, L.; BRUDER, G.; STEINCKE, F.; ZENDER, R.; GEIGER, C.: Fallbeispiele für VR/AR. In: Dörner, R.; Broll, W.; Grimm, P.; Jung, B. (Hrsg.): Virtual und Augmented Reality (VR/AR) – Grundlagen und Methoden der Virtuellen und Augmentierten Realität. 2. Auflage, Springer Berlin Heidelberg; Imprint: Springer Vieweg, Berlin, 2019, S. 357–392
- [Dob08] DOBRATZ, S.: Einführung in Semantic Web, HU Berlin, 2008
- [DS06] DRUMMOND, N.; SHEARER, R.: The Open World Assumption, University of Manchester, 2006
- [Dum10] DUMITRESCU, R.: Entwicklungssystematik zur Integration kognitiver Funktionen in fortgeschrittene mechatronische Systeme. Dissertation, Heinz Nixdorf Institut, Universität Paderborn, 2010
- [DZ15] DIEBOLD, P.; ZEHLER, T.: The agile practices impact model: idea, concept, and application scenario. In: Pfahl, D.; Bendraou, R.; Turner, R.; Kuhrmann, M.; Hebig, R.; Maggi, F. (Hrsg.): Proceedings of the 2015 International Conference on Software and Systems Process (ICSSP). International Conference on Software and Systems Process ICSSP, Tallinn, Estonia, ICPS: ACM international conference proceeding series, Association for Computing Machinery, Inc, New York, NY, 2015, S. 92–96

- [Ebe15] EBEL, B.: Modellierung von Zielsystemen in der interdisziplinären Produktentstehung. Dissertation, Institut für Produktentwicklung IPEK, Karlsruher Institut für Technologie, 2015
- [EGL16] ELIA, V.; GNONI, M. G.; LANZIOTTO, A.: Evaluating the application of augmented reality devices in manufacturing from a process point of view: An AHP based model. *Expert Systems with Applications*, (63), 2016, S. 187–197
- [EKL+96] EYERER, P.; KELLER, B.; LÜCK, T.; ESCHL, J.; DUSEL, K.-H.; STIERLEN, P.; ELSNER, P.; SHEN, J.; BAUMANN, F.; VÖLKLE, D.: *Rapid Prototyping & Rapid Tooling: Generative Fertigungsverfahren und Prozeßketten in der Produktentwicklung*, 1996
- [El-16] EL-HAJI, M.: Ontologie-basierte Definition von Anforderungen an Validierungswerzeuge in der Fahrzeugtechnik. Dissertation, Institut für Fahrzeugsystemtechnik, Karlsruher Institut für Technologie, 2016
- [Els20] ELSTNER, M.: Use cases of extended reality in the construction industry. Thesis - Double Degree Program in Civil- and Construction Engineering, Faculty of Technology, LAB University of Applied Sciences Lappeenranta, 2020
- [EM13] EHRLENSPIEL, K.; MEERKAMM, H.: *Integrierte produktentwicklung: Denkabläufe, methodeneinsatz, zusammenarbeit*. Carl Hanser Verlag GmbH Co KG, 2013
- [EM20] EGGER, J.; MASOOD, T.: Augmented reality in support of intelligent manufacturing – A systematic literature review. *Computers & Industrial Engineering*, (140), 2020, S. 106195
- [Eps22-ol] EPSON: Smartglasses. Unter: <https://www.epson.de/products/see-through-mobile-viewer>, 18. Januar 2022
- [ERZ14] EIGNER, M.; ROUBANOV, D.; ZAFIROV, R.: *Modellbasierte virtuelle Produktentwicklung*. Springer, 2014
- [ES09] EIGNER, M.; STELZER, R.: *Product Lifecycle Management – Ein Leitfaden für Product Development und Life Cycle Management*. Springer, 2009
- [EUJ18] EVA BOGNER; ULRICH LÖWEN; JÖRG FRANKE: Bedeutung der zukünftigen Produktion kundenindividueller Produkte in Losgröße 1. In: Redlich, T.; Moritz, M.; Wulfsberg, J. P. (Hrsg.): *Interdisziplinäre Perspektiven zur Zukunft der Wertschöpfung*. Springer Fachmedien, Wiesbaden, 2018, S. 63–75
- [Fel14] FELLMANN, M.: *Web Ontology Language (OWL) und Anfragesprachen – OWL*, Institut für Informationsmanagement, Universität Osnabrück, 2014
- [FHM+15] FELLMANN, M.; HEITMANN, C.; METZGER, D.; NOBBE, L.; THOMAS, O.: *TKD 4.0 - Klassifikation; Einordnung und Bewertung der Einsatzpotenziale von Augmented-Reality-Anwendungen für den Technischen Kundendienst*, 2015
- [For07] FORBRIG, P.: *Objektorientierte Softwareentwicklung mit UML*. 3. Auflage, Lehrbücher zur Informatik, Hanser, München, 2007
- [FR18] FINK, C.; ROSEDALE, P.: *Charlie Fink's metaverse – An AR enabled guide to AR & VR*. Cool Blue Media, United States, 2018
- [FSC18] FREEMAN, I.; SALMON, J.; COBURN, J.: A bi-directional interface for improved interaction with engineering models in virtual reality design reviews. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, (12)2, 2018, S. 549–560
- [Gaj20-ol] GAJSEK, D.: *HoloLens 2 Setup Guide - Prepare your HoloLens 2 for Development in Unity | Circuit Stream*. Unter: <https://circuitstream.com/blog/hololens-2-setup-guide/>, 31. Januar 2022
- [Gar20-ol] GARTNER: *Augmented Reality (AR)*. Unter: <https://www.gartner.com/en/information-technology/glossary/augmented-reality-ar>, 12. Mai 2020
- [Gar84] GARVIN, D. A.: What Does “Product Quality” Really Mean? *Sloan management review* 25, 1984
- [Gas07-ol] GASEVIC, D.: *UMLtoOWL: Converter from UML to OWL*. Unter: <https://www.sfu.ca/~dgasevic/projects/UMLtoOWL/#downloads>, 27. Juni 2024
- [GBH+19] GRIMM, P.; BROLL, W.; HEROLD, R.; HUMMEL, J.: *VR/AR-Eingabegeräte und Tracking*. In: Dörner, R.; Broll, W.; Grimm, P.; Jung, B. (Hrsg.): *Virtual und Augmented Reality (VR/AR) – Grundlagen und Methoden der Virtuellen und Augmentierten Realität*. 2.

- Auflage, Springer Berlin Heidelberg; Imprint: Springer Vieweg, Berlin, 2019, S. 117–162
- [GDE+18] GAUSEMEIER, J.; DUMITRESCU, R.; ECHTERFELD, J.; PFÄNDER, T.; STEFFEN, D.; THIELEMANN, F.: Innovationen für die Märkte von morgen – Strategische Planung von Produkten, Dienstleistungen und Geschäftsmodellen. Hanser, München, 2018
- [GDR+15] GAUSEMEIER, J.; DUMITRESCU, R.; RAMMIG, F.; SCHÄFER, W.; TRÄCHTLER, A.: Entwurf mechatronischer Systeme, Heinz Nixdorf Institut, Universität Paderborn, 2015
- [Geu96] GEUER, A.: Einsatzpotential des Rapid Prototyping in der Produktentwicklung. iwb Forschungsberichte, Berichte aus dem Institut für Werkzeugmaschinen und Betriebswissenschaften der Technischen Universität München Band 100, Springer Berlin Heidelberg, Berlin, Heidelberg, 1996
- [GF06] GAUSEMEIER, J.; FELDMANN, K. (Hrsg.): Integrative Entwicklung räumlicher elektronischer Baugruppen. Carl Hanser Verlag, München, 2006
- [GFD+08] GAUSEMEIER, J.; FRANK, U.; DONOTH, J.; KAHL, S.: Spezifikationstechnik zur Beschreibung der Prinziplösung selbstoptimierender Systeme des Maschinenbaus (Teil 1 und 2). Konstruktion - Fachaufsatzz Mechatronik, 7/8/9, 2008
- [GH13] GREGOR, S.; HEVNER, A. R.: Positioning and Presenting Design Science Research for Maximum Impact. MIS Quarterly, (37)2, 2013, S. 337–355
- [Gib16-ol] GIBSON, D. K.: Why car designers stick with clay. Unter: <http://www.bbc.com/autos/story/20161111-why-car-designers-stick-with-clay>, 2. Juli 2021
- [GL19] GLOVER, J.; LINOWES, J.: Complete Virtual Reality and Augmented Reality Development with Unity – Leverage the power of Unity and become a pro at creating mixed reality applications. Packt Publishing Limited, Birmingham, 2019
- [GM12] GRÜNWALD, A.; MOSER, T.: umlTUowl - A Both Generic and Vendor-Specific Approach for UML to OWL Transformation, Institute of Information Systems Engineering, TU Wien, 2012
- [Goo21-ol] GOOGLE: Glass. Unter: <https://www.google.com/glass/start/>, 18. Januar 2022
- [Goo22-ol] GOOGLE DEVELOPERS: Maps SDK for Unity Overview. Unter: https://developers.google.com/maps/documentation/gaming/overview_musk, 1. Februar 2022
- [GR98] GIARRATANO, J. C.; RILEY, G.: Expert systems – principles and programming. 3. Auflage, PWS Publishing Company, 1998
- [Gra06] GRADY, J. O.: System requirements analysis. Elsevier Academic Press, Amsterdam, 2006
- [GSW14] GRIMM, R.; SCHULLER, M.; WILHELMER, R.: Marktanalyse. In: Grimm, R.; Schuller, M.; Wilhelmer, R. (Hrsg.): Portfoliomanagement in Unternehmen – Leitfaden für Manager und Investoren. Springer-Gabler, Wiesbaden, 2014, S. 59–109
- [GTD13] GAUSEMEIER, J.; TSCHIRNER, C.; DUMITRESCU, R.: Der Weg zu Intelligenten Technischen Systemen. Industrie-Management, (29)Nr.1, 2013, S. 49–52
- [Haa14] HAAS, J.: A history of the unity game engine. Interactive Qualifying Project, Worcester Polytechnic Institute, 2014
- [Har11] HARTUNG, M.: Ontologie-Management – Kapitel 2: Ontologiesprachen. Vorlesungsskript, Institut für Informatik, Universität Leipzig, 2011
- [Har17-ol] HARDI, J.: Grocery Tutorial. Unter: <https://github.com/protegeproject/cellfile-plugin/wiki/Grocery-Tutorial>, 27. Juni 2024
- [Has14] HASSAN, B.: A design framework for developing a reconfigurable driving simulator. Dissertation, Fakultät Maschinenbau, Universität Paderborn, 2014
- [Hau20-ol] HAUCK, M.: Die Zukunft gehört den Hologrammen. Unter: <https://www.sueddeutsche.de/digital/augmented-reality-datenbrille-hololens-microsoft-1.4756247>, 14. Mai 2020
- [Hef16-ol] HEFLIN, J.: AN INTRODUCTION TO THE OWL WEB ONTOLOGY LANGUAGE. Unter: <http://www.cse.lehigh.edu/~heflin/IntroToOWL.pdf>, 27. Juni 2024
- [Hev07] HEVNER, A. R.: A Three Cycle View of Design Science Research. Scandinavian Journal of Information Systems, 19(2), 2007, S. 87–92

- [HGA+15] HASSAN, B.; GAUSEMEIER, J.; ABDELGAWAD, K.; BERSSENBRÜGGE, J.; GRAFE, M.: Systematik zur Entwicklung von rekonfigurierbaren Fahrsimulatoren. Augmented und Virtual Reality in der Produktentstehung, 2015
- [HGB01] HUA, H.; GAO, C.; BROWN, L.: Using a Head-Mounted Projective Display in Interactive Augmented Environments, Beckman Institute, University of Illinois, 2001
- [Hil97] HILDEBRAND, V. G.: Individualisierung als strategische Option der Marktbearbeitung – Determinanten und Erfolgswirkungen kundenindividueller Marketingkonzepte. Forum Marketing, Deutscher Universitätsverlag, Wiesbaden, 1997
- [HKR+08] HITZLER, P.; KRÖTZSCH, M.; RUDOLPH, S.; SURE, Y.: Semantic Web – Grundlagen. SpringerLink Bücher, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008
- [HLL+19] HAN, Y.-S.; LEE, J.; LEE, J.; LEE, W.; LEE, K.: 3D CAD data extraction and conversion for application of augmented/virtual reality to the construction of ships and offshore structures. International Journal of Computer Integrated Manufacturing, (32)7, 2019, S. 658–668
- [HNA+11] HOLMQVIST, K.; NYSTRÖM, M.; ANDERSSON, R.; DEWHURST, R.; JARODZKA, H.; VAN WEIJER, J. DE: Eye Tracking – A comprehensive guide to methods and measures. OUP Oxford, 2011
- [HNC+19] HOANG, D.; NADERI, E.; CHENG, R.; ARYANA, B.: Adopting Immersive Technologies for Design Practice: The Internal and External Barriers. Proceedings of the Design Society: International Conference on Engineering Design, (1)1, 2019, S. 1903–1912
- [HO18-ol] HARDI, J.; O'CONNOR, M.: Cellfie-Plugin. Unter: <https://github.com/protegeproject/cellfie-plugin>, 27. Juni 2024
- [Hol20-ol] HOLO-LIGHT: BMW Uses AR 3S in Prototyping. Unter: <https://holo-light.com/bmw-uses-ares-in-prototyping/>, 4. Februar 2022
- [Hoo15] HOOSHMAND, Y.: Transparenzerhöhung bei der Entwicklung von individualisierten Produkten in der Einzelfertigung. Dissertation, Uni Duisburg Essen, 2015
- [Hor01] HORVÁTH, I.: A contemporary survey of scientific research into engineering design: Proceedings of the International Conference on Engineering Design (ICED), 2001, S. 21–23
- [Hor11] HORRIDGE, M.: A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition 1.3, University of Manchester, 2011
- [Hou20] HOUSE, B.: Choosing the right netcode for your game. Unity Blog, 2020, 08.09.2020
- [Hou21-ol] HOUGAARD, K.: Behind the scenes: Speeding up Unity workflows – Unity Blog. Unter: <https://blog.unity.com/technology/behind-the-scenes-speeding-up-unity-workflows>, 27. Juni 2024
- [HŠM+19] HORVAT, N.; ŠKEC, S.; MARTINEC, T.; LUKAČEVIĆ, F.; PERIŠIĆ, M. M.: Comparing Virtual Reality and Desktop Interface for Reviewing 3D CAD Models. Proceedings of the Design Society: International Conference on Engineering Design, (1)1, 2019, S. 1923–1932
- [HWF+12] HABERFELLNER, R.; WECK, O. DE; FRICKE, E.; VÖSSNER, S.: Systems Engineering – Grundlagen und Anwendung. 12. Auflage. Zürich: Orell Füssli, 2012, 978-3280040683
- [IDG19] IDG RESEARCH SERVICES: Studie Virtual Reality / Augmented Reality, 2019
- [IEC60529:2013] INTERNATIONAL ELECTROTECHNICAL COMMISSION: Degrees of protection provided by enclosures (IP Code), 2013
- [Ing20-ol] INGENIEUR.DE: Interdisziplinäre Konstruktionsmethoden und -prozesse - ingenieur.de. Unter: <https://www.ingenieur.de/fachmedien/konstruktion/produktentwicklung/interdisziplinaere-konstruktionsmethoden-und-prozesse/>, 27. Juni 2024
- [Ins98] IEEE guide for developing system requirements specifications, 1998
- [ISO/IEC18039:2019] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION; INTERNATIONAL ELECTROTECHNICAL COMMISSION: Information Technology – Computer Graphics, Image Processing And Environmental Data Representation And Coding Of Audio, Picture, Multimedia And Hypermedia Information – Mixed And Augmented Reality (MAR) Reference Model, ISO Copyright Office, Genf, 2019

- [ISO/IEC19505-1:2012] OBJECT MANAGEMENT GROUP: Information technology - Object Management Group Unified Modeling Language (OMG UML), Infrastructure, 2012
- [ISO/IEC19505-2:2012] OBJECT MANAGEMENT GROUP: Information technology - Object Management Group Unified Modeling Language (OMG UML) - Part 2: Superstructure, 2012
- [ISO14306:2017] ISO – INTERNATIONAL ORGANIZATION FOR STANDARDIZATION: Industrial automation systems and integration — JT file format specification for 3D visualization, 2017
- [ISO9000:2005] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION: Quality management systems – Fundamentals and vocabulary;. ISO 9000:2005, ISO Copyright Office, Geneva, 2005
- [ISO9241:2006] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION: Ergonomics of human-system interaction, 2006
- [ISO9241-210:2010] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION: Ergonomics of human-system interaction – Part 210: Human-centered design for interactive systems, 2010
- [Jan12] JAN OLHAGER: The Role of Decoupling Points in Value Chain Management. In: Jodlbauer, H.; Olhager, J.; Schonberger, R. J. (Eds.): Modelling Value – Selected Papers of the 1st International Conference on Value Chain Management. Physica-Verlag, Heidelberg, 2012, pp. 37–47
- [Jas15] JASPER, J.: Virtual Reality-Based Product Representations in Conjoint Analysis – Empirical Insights Into Its Applicability for Early Customer Integration in the Development Process of a Technical Innovation. Dissertation, Fachbereich Wirtschaftswissenschaften, Universität Münster, 2015
- [JCT19] JIOMEKONG, A.; CAMARA, G.; TCHUENTE, M.: Extracting ontological knowledge from Java source code using Hidden Markov Models. Open Computer Science, (9)1, 2019, S. 181–199
- [JD17] JOHANNA SPALLEK; DIETER KRAUSE: Entwicklung individualisierter Produkte durch den Einsatz Additiver Fertigung: Additive Manufacturing Quantifiziert. Springer Vieweg, Berlin, Heidelberg, 2017, S. 69–83
- [Jop21] JOPPEN, R.: Systematik zur Bewertung von Industrie 4.0 Anwendungen in der Produktion. Dissertation, Heinz Nixdorf Institut, Universität Paderborn, 2021
- [Jor19-ol] JORY MACKAY: Software Development Process: How to Pick The Process That's Right For You | Planio. Unter: <https://plan.io/blog/software-development-process/#the-sdlc-what-is-the-software-development-lifecycle-and-why-is-it-so-important-to-have-one>, 29. März 2021
- [JS18] JOSUPEIT, S.; SCHMID, H.-J.: Individualisierung mit und durch additive(r) Fertigung. In: Berscheid, A.-L.; Riegraf, B. (Hrsg.): Wissenschaft im Angesicht »großer gesellschaftlicher Herausforderungen« – Das Beispiel der Forschung an hybriden Leichtbaumaterialien. transcript Verlag, 2018, S. 19–34
- [JV19] JUNG, B.; VITZTHUM, A.: Virtuelle Welten. In: Dörner, R.; Broll, W.; Grimm, P.; Jung, B. (Hrsg.): Virtual und Augmented Reality (VR/AR) – Grundlagen und Methoden der Virtuellen und Augmentierten Realität. 2. Auflage, Springer Berlin Heidelberg; Imprint: Springer Vieweg, Berlin, 2019, S. 79–115
- [KBB+18] KIM, K.; BILLINGHURST, M.; BRUDER, G.; DUH, H. B.-L.; WELCH, G. F.: Revisiting Trends in Augmented Reality Research: A Review of the 2nd Decade of ISMAR (2008–2017). IEEE transactions on visualization and computer graphics, (24)11, 2018, S. 2947–2962
- [KE21] KONDRATENKO, E.; ECKERTZ, DANIEL: Systematik zur Identifikation von Potentialen AR-basierter Produktvalidierung. unveröffentlichte Bachelorarbeit, Heinz Nixdorf Institut - Fachgruppe Advanced Systems Engineering, Universität Paderborn, 2021
- [KHD+13] KOHLER, K.; HOCHREUTER, T.; DIEFENBACH, S.; LENZ, E.; HASSENZAHL, M.: Durch schnelles Scheitern zum Erfolg: Eine Frage des passenden Prototypen? German UPA e.V, 2013
- [Khr11-ol] KHRONOS GROUP: COLLADA - 3D Asset Exchange Schema. Unter: <https://www.khronos.org/api/collada>, 1. Februar 2022

- [Khr22-ol] KHRONOS GROUP: OpenXR - High-performance access to AR and VR —collectively known as XR— platforms and devices. Unter: <https://www.khronos.org/openxr/>, 2. Februar 2022
- [KHV13] KATZENBACH, A.; HANDSCHUH, S.; VETTERMANN, S.: JT Format (ISO 14306) and AP 242 (ISO 10303): The Step to the Next Generation Collaborative Product Creation. 411. Auflage, IFIP International Conference on Digital Product and Process Development Systems, Springer, 2013
- [KK13] KOVACS, G.L.; KOCHAN, D. (Eds.): Digital Product and Process Development Systems – IFIP TC 5 International Conference, NEW PROLAMAT 2013, Dresden, Germany, October 10-11, 2013, Proceedings. Springer Berlin Heidelberg, v.411 Berlin/Heidelberg, 2013
- [KKS17] KLIMANT, P.; KOLLATSCH, C.; SCHUMANN, M.: Augmented Reality Solutions in Mechanical Engineering: Proceedings of the ASME 12th International Manufacturing Science and Engineering Conference. Los Angeles, California, USA, The American Society of Mechanical Engineers, New York, N.Y., 2017
- [KKS19] KÜNZEL, M.; KRAUS, T.; STRAUB, S.: Kollaboratives Engineering – Grundzüge und Herausforderungen der unternehmensübergreifenden Zusammenarbeit beim Engineering von Produkten und begleitenden Services. Studie im Auftrag des Bundesministeriums für Wirtschaft und Energie (BMWi) im Rahmen der Begleitforschung zum Technologieprogramm PAiCE. Berlin, 2019
- [Kle99] KLEIN, B.: QFD - Quality function deployment – Konzept, Anwendung und Umsetzung für Produkte und Dienstleistungen. expert-Verl.; Linde, Renningen-Malmsheim, 1999
- [Kli16] KLINGLER, S.: Eine Methode zur effizienten und effektiven Unterstützung der kontinuierlichen Validierung im Kontext der PGE - Produktgenerationsentwicklung, 2016
- [KMF+19] KÜSTER, T.; MASUCH, N.; FÄHNDRICH, J.; TSCHIRNER-VINKE, G.; TASCHNER, J.; SPECKE, M.; IBEN, H.; BAUMANN, H.; SCHMID, F.; STÖCKLEIN, J.; RÖLTGEN, D.; TRINOGA, M.: A Distributed Architecture for Modular and Dynamic Augmented Reality Processes. IEEE 17th International Conference on Industrial Informatics (INDIN), 2019, S. 663–670
- [KNG19] KIM, S.; NUSSBAUM, M. A.; GABBARD, J. L.: Influences of augmented reality head-worn display type and user interface design on performance and usability in simulated warehouse order picking. Applied ergonomics, (74), 2019, S. 186–193
- [Koh14] KOHN, A.: Entwicklung einer Wissensbasis für die Arbeit mit Produktmodellen. Dissertation, Fakultät für Maschinenwesen, Technische Universität München, 2014
- [KP10] KREVELEN, VAN, D.W.F.; POELMAN, R.: A Survey of Augmented Reality Technologies, Applications and Limitations, 2010
- [KP18-ol] KOBYLINSKA, A.; PEREIRA MARTINS, F.: Produktentwicklung setzt auf Evolution. Unter: <https://www.com-magazin.de/praxis/business-it/produktentwicklung-setzt-evolution-1571339.html>, 27. Juni 2024
- [KPM17] KPMG AG WIRTSCHAFTSPRÜFUNGSGESELLSCHAFT (Hrsg.): KPMG Consumer Barometer 2/2017 – Trends und Treiber im Sektor Consumer Markets. Thema: Produktindividualisierung. Hamburg, 2017
- [KPR05] KLOSE, R.; PENLINGTON, J.; RUCKELSHAUSEN, A.: Usability study of 3D Time-of-Flight cameras for automatic plant phenotyping, Leibniz-Institut für Agrartechnik Potsdam-Bornim e.V., University of Applied Sciences Osnabrück, 2005
- [KSS19] KHAN, Z. M. A.; SAEIDLLOU, S.; SAADAT, M.: Ontology-based decision tree model for prediction in a manufacturing network. Production & Manufacturing Research, (7)1, 2019, S. 335–349
- [KSW18] KÜGLER, P.; SCHLEICH, B.; WARTZACK, S.: Consistent digitalization of engineering design – an ontology-based approach, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2018
- [Kur89] KURBEL, K.: Entwicklung und Einsatz von Expertensystemen – Eine anwendungsorientierte Einführung in wissensbasierte Systeme. Springer Berlin Heidelberg, Berlin, Heidelberg, 1989

- [KWH13] KAGERMANN, H.; WAHLSTER, W.; HELBIG, J.: Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0 – Abschlussbericht des Arbeitskreises Industrie 4.0, 2013
- [LB06] LINDEMANN, U.; BAUMBERGER, G. C.: Individualisierte Produkte. In: Lindemann, U.; Reichwald, R.; Zäh, M. (Hrsg.): Individualisierte Produkte – Komplexität beherrschen, in Entwicklung und Produktion. VDI-Buch, Springer, Berlin, 2006, S. 7–16
- [LB17] LINOWES, J.; BABELINSKI, K.: Augmented Reality for Developers – Build practical augmented reality applications with Unity, ARCore, ARKit, and Vuforia. Packt Publishing Limited, Birmingham, 2017
- [LBM+09] LIEBERKNECHT, S.; BENHIMANE, S.; MEIER, P.; NAVAB, N.: A dataset and evaluation methodology for template-based tracking algorithms. In: Institute of Electrical and Electronics Engineers (Ed.). 2009 8th IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 10/19/2009 - 10/22/2009, Orlando, FL, USA, IEEE Service Center, Piscataway, NJ, 2009, pp. 145–151
- [LCN+19] LEE GEON-HEE; CHOI PYEONG-HO; NAM JEONG-HWAN; HAN HWA-SEOP; LEE SEUNG-HYUN; KWON SOON-CHUL: A Study on the Performance Comparison of 3D File Formats on the Web. *International journal of advanced smart convergence*, (8)1, 2019, S. 65–74
- [Lei19-ol] LEIBSON, S.: IMUs zur exakten Standortbestimmung: Teil 2 – höhere Präzision durch IMU-Software. Unter: <https://www.digikey.ch/de/articles/imus-for-precise-location-part-2-how-to-use-imu-software-for-greater-precision>, 1. Juli 2024
- [Lei20-ol] LEINHOS, S.: UML2OWL. Unter: <http://diplom.oooyoo.de>, 1. Juli 2024
- [Len22-ol] LENOVO: ThinkReality A3 Smart Glasses. Unter: <https://www.lenovo.com/us/en/thinkrealitya3>, 27. Juni 2024
- [Li14] LI, L.: Time-of-Flight Camera – An Introduction, 2014
- [LKS+19] LIEBRECHT, C.; KRODEL, T.; STRICKER, N.; LANZA, G.: Ausprägungen von Industrie 4.0 im Mittelstand. *Zeitschrift für wirtschaftlichen Fabrikbetrieb*, (114)6, 2019, S. 398–403
- [LLM14] LIU, Y.; LATHER, J.; MESSNER, J.: Virtual Reality to Support the Integrated Design Process: A Retrofit Case Study. American Society of Civil Engineers, 2014
- [LNO17] LI, W.; NEE, A. Y. C.; ONG, S. K.: A State-of-the-Art Review of Augmented Reality in Engineering Analysis and Simulation. *Multimodal Technologies and Interaction*, (1)3, 2017, S. 17
- [LSR+15] LORENZ, M.; SPRANGER, M.; RIEDEL, T.; PÜRZEL, F.; WITTSTOCK, V.; KLIMANT, P.: CAD to VR – A Methodology for the Automated Conversion of Kinematic CAD Models to Virtual Reality. In: Elsevier Procedia (Hrsg.): 48th CIRP International Conference on Manufacturing Systems CMS, 2015, S. 358–363
- [LST08] LIM, Y.-K.; STOLTERMAN, E.; TENENBERG, J.: The anatomy of prototypes. *ACM Transactions on Computer-Human Interaction*, (15)2, 2008, S. 1–27
- [LZW19] LEE, Y.-H.; ZHAN, T.; WU, S.-T.: Prospects and challenges in augmented reality displays. *Virtual Reality & Intelligent Hardware*, (1)1, 2019, S. 10–20
- [mag22-ol] MAGIC LEAP: Magic Leap 1. Unter: <https://www.magicleap.com/en-us/magic-leap-1>, 17. Januar 2022
- [Mai20-ol] MAIMANN, M.: Mockup vs Model vs Prototype – Understanding the Difference Between a Working Prototype and a Production-Ready Design. Unter: <https://intelligent-product.solutions/process-blog/what-is-a-prototype/>, 1. Juli 2021
- [MB08] MCHENRY, K.; BAJCSY, P.: An overview of 3d data content, file formats and viewers. Image Spatial Data Analysis Group, National Center for Supercomputing Applications - Technical Report, 2008
- [MBP15] MEUSEL, R.; BIZER, C.; PAULHEIM, H.: A Web-scale Study of the Adoption and Evolution of the schema.org Vocabulary over Time. In: Akerkar, R.; Dikaiakos, M.; Achilleos, A.; Omitola, T. (Hrsg.): Proceedings of the 5th International Conference on Web Intelligence, Mining and Semantics - WIMS '15. the 5th International Conference, 13.07.2015 - 15.07.2015, Larnaca, Cyprus, ACM Press, New York, New York, USA, 2015, S. 1–11

- [ME19] MASOOD, T.; EGGER, J.: Augmented reality in support of Industry 4.0—Implementation challenges and success factors. *Robotics and Computer-Integrated Manufacturing*, (58), 2019, S. 181–195
- [ME20] MÖLLER, M.; ECKERTZ, DANIEL: Analyse und Modellierung interaktiver Augmented Reality-Systeme im industriellen Einsatz. unveröffentlichte Masterarbeit, Heinz Nixdorf Institut - Fachgruppe Advanced Systems Engineering, Universität Paderborn, 2020
- [met11-ol] METAIO GMBH: metaio veröffentlicht die fortschrittlichste mobile Augmented Reality Software ab sofort kostenlos, Pressemitteilung - PresseBox. Unter: <https://www.pressebox.de/inaktiv/metaio-gmbh/metaio-veroeffentlicht-die-fortschrittlichste-mobile-Augmented-Reality-Software-ab-sofort-kostenlos/boxid/469598>, 27. Juni 2024
- [MGM+17] MATTHIESEN, S.; GWOSCH, T.; MANGOLD, S.; GRAUBERGER, P.; STECK, M.; CERSOWSKY, S.: Frontloading in der Produktentwicklung von Power-Tools durch frühe Validierung mit Hilfe von leistungsskalierten Prototypen. *Stuttgarter Symposium der Produktentwicklung*, 2017
- [MHL16] MARTINEZ-RODRIGUEZ, J. L.; HOGAN, A.; LOPEZ-AREVALO, I.: Information Extraction meets the Semantic Web: A Survey, *Cinvestav Tamaulipas*, 2016
- [Mic21-ol] MICROSOFT: Tutorials - Mixed Reality. Unter: <https://docs.microsoft.com/en-us/windows/mixed-reality/develop/unity/tutorials>, 31. Januar 2022
- [Mic22a-ol] MICROSOFT: HoloLens 2: Find Specs and Features. Unter: <https://www.microsoft.com/en-us/d/hololens-2/91pnzzznwcp?activetab=pivot:techspecstab>, 18. Januar 2022
- [Mic22b-ol] MICROSOFT: Unity3D (C#) SDK - PlayFab. Unter: <https://docs.microsoft.com/en-us/gaming/playfab/sdks/unity3d/>, 27. Juni 2024
- [Mic22c-ol] MICROSOFT: Dokumentation- Mixed Reality Toolkit. Unter: <https://docs.microsoft.com/de-de/dotnet/api/Microsoft.MixedReality.Toolkit?view=mixed-reality-toolkit-unity-2020-dotnet-2.7.0>, 27. Juni 2024
- [Mic22d-ol] MICROSOFT: Setting up Photon Unity Networking - Mixed Reality. Unter: <https://docs.microsoft.com/en-us/windows/mixed-reality/develop/unity/tutorials/mr-learning-sharing-02>, 27. Juni 2024
- [Mic22e-ol] MICROSOFT: Microsoft HoloLens | Mixed Reality Technology for Business. Unter: <https://www.microsoft.com/en-us/hololens>, 17. Januar 2022
- [Mic23-ol] MICROSOFT: OpenXR - Mixed Reality. Unter: <https://docs.microsoft.com/de-de/windows/mixed-reality/develop/native/openxr>, 27. Juni 2024
- [Mil07] MILTON, N. R.: *Knowledge Acquisition in Practice – A Step-by-step Guide*. Springer London, London, 2007
- [Mir22-ol] MIRROR: Network | Unity Asset Store. Unter: <https://assetstore.unity.com/packages/tools/network/mirror-129321>, 27. Juni 2024
- [MK94] MILGRAM, P.; KISHINO, F.: A Taxonomy of Mixed Reality Visual Displays. *IEICE Transactions on Information and Systems*, (vol. E77-D, no. 12)12, 1994, S. 1321–1329
- [MLB+16] MACLARTY, I.; LANGEVINE, L.; BOSSCHE, M. V.; ROSS, P.: *Using SWRL for Rule-Driven Applications*, 2016
- [Mon22-ol] MONGODB: Realm SDKs. Unter: <https://docs.mongodb.com/realm/sdk/>, 27. Juni 2024
- [MPP12-ol] MOTIK, B.; PATEL-SCHNEIDER, P. F.; PARSIA, B.: *OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition) – W3C Recommendation*. Unter: <https://www.w3.org/TR/owl2-syntax/>, 26. Januar 2022
- [MRD+16] MOTA, J. M.; RUIZ-RUBE, I.; DODERO, J. M.; FIGUEIREDO, M.: *Visual Environment for Designing Interactive Learning Scenarios with Augmented Reality*. International Association for Development of the Information Society, 2016
- [MRS14] MAIR, G. M.; ROBINSON, A.; STORR, J.: Applying Augmented Reality to the Concept Development Stage of the Total Design Methodology. In: Shumaker, R.; Lackey, S. (Eds.): *Virtual, Augmented and Mixed Reality. Applications of Virtual and Augmented Reality – 6th International Conference, VAMR 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014, Proceedings, Part II*. SpringerLink Bücher, 8526, Springer, Cham, 2014

- [MSA20] MOURTZIS, D.; SIATRAS, V.; ANGELOPOULOS, J.: Real-Time Remote Maintenance Support Based on Augmented Reality (AR). *Applied Sciences*, (10)5, 2020, S. 1855
- [Mur12] MURCH, R.: *The Software Development Lifecycle - A Complete Guide*, 2012
- [Mus15] MUSEN, M. A.: The Protégé Project: A Look Back and a Look Forward. *AI matters*, (1)4, 2015, S. 4–12
- [Mv04-ol] MCGUINNESS, D. L.; VAN HARMELEN, F.: OWL Web Ontology Language Overview – W3C Recommendation. Unter: <https://www.w3.org/TR/owl-features/>, 26. Januar 2022
- [NCF+03] NOY, N. F.; CRUBÉZY, M.; FERGUSON, R. W.; KNUBLAUCH, H.; TU, S. W.; VENDETTI, J.; MUSEN, M. A.: Protégé-2000: an open-source ontology-development and knowledge-acquisition environment: AMIA ... Annual Symposium proceedings. *AMIA Symposium – Open Source Expo*, 2003
- [NMR+15] NAQVI, N. Z.; MOENS, K.; RAMAKRISHNAN, A.; PREUVENEERS, D.; HUGHES, D.; BERBERS, Y.: To cloud or not to cloud. In: Wainwright, R. L.; Corchado, J. M.; Bechini, A.; Hong, J. (Hrsg.): *Proceedings of the 30th Annual ACM Symposium on Applied Computing - SAC '15*. 13.04.2015 - 17.04.2015, Salamanca, Spain, ACM Press, New York, New York, USA, 2015, S. 555–562
- [nre22-ol] NREAL: Nreal Light – Ready-to-wear Mixed Reality Glasses. Unter: <https://www.nreal.ai/light/>, 17. Januar 2022
- [Obe06] OBERLE, D.: *Semantic Management of Middleware*. Springer US, Vol. 1 Boston, MA, 2006
- [O'C09] O'CONNOR, M.: Importing Data into Protégé-OWL, Stanford Center for Biomedical Informatics Research, Stanford University, 2009
- [OD09] O'CONNOR, M.; DAS, A.: SQWRL: a query language for OWL: Proceedings of the 5th International Workshop on OWL: Experiences and Directions (OWLED 2009). Chantilly, VA, United States, 2009
- [Odi22-ol] ODIN INSPECTOR: Odin Inspector and Serializer | Improve your workflow in Unity. Unter: <https://odininspector.com/>, 31. Januar 2022
- [OHM10] O'CONNOR, M.; HALASCHEK-WIENER, C.; MUSEN, M.: M²: a Language for Mapping Spreadsheets to OWL: OWLED. Band 614, 2010
- [OMG19] OBJECT MANAGEMENT GROUP: OMG Systems Modeling Language (OMG SysML™) – Version 1.6, 2019
- [OPB+14] OSTERWALDER, A.; PIGNEUR, Y.; BERNARDA, G.; SMITH, A.: *Value proposition design – How to create products and services customers want*. Wiley, Hoboken, NJ, 2014
- [Ope22-ol] OPENCV: OpenCV for Unity - Unity Asset Store. Unter: <https://assetstore.unity.com/packages/tools/integration/opencv-for-unity-21088>, 1. Februar 2022
- [Ovt05] OVTCHAROVA, J.: Life Cycle Engineering Konzepte. ICIDO Summit "Next Generation VR", 2005
- [PAC22-ol] PACE AEROSPACE & IT: Pacelab WEAVR. Unter: <https://pace.txtgroup.com/products/extended-reality/pacelab-weavr/>, 3. Januar 2021
- [Pan19-ol] PANETTA, K.: 5 Trends Appear on the Gartner Hype Cycle for Emerging Technologies, 2019. Unter: <https://www.gartner.com/smarterwithgartner/5-trends-appear-on-the-gartner-hype-cycle-for-emerging-technologies-2019/>, 27. Juni 2024
- [PB15] PRICE, T. W.; BARNES, T.: Comparing Textual and Block Interfaces in a Novice Programming Environment: Proceedings of the eleventh annual International Conference on International Computing Education Research. ICER, ICER '15, Association for Computing Machinery, 2015, S. 91–99
- [PBF+07] PAHL, G.; BEITZ, W.; FELDHUSEN, J.; GROTE, K.-H.: *Konstruktionslehre – Grundlagen erfolgreicher Produktentwicklung ; Methoden und Anwendung*. 7. Auflage, Springer, Berlin, 2007
- [PBG+14] PETERSEN, M.; BANDAK, S.; GAUSEMEIER, J.; IWANEK, P.; SCHNEIDER, M.: Methodik zur Berücksichtigung von Wechselwirkungen zwischen Produkt und Produktionsystem in den frühen Phasen der Produktentwicklung – Ein Praxisbeispiel. *Digitales Engineering zum Planen, Testen und Betreiben technischer Systeme*, IFF-Wissenschaftstage, 2014

- [PBM+18] Paelke, V.; Büttner, S.; Mucha, H.; Röcker, C.: A Checklist Based Approach for Evaluating Augmented Reality Displays in Industrial Applications. In: Trzcielinski, S. (Hrsg.): Advances in Ergonomics of Manufacturing: Managing the Enterprise of the Future. Advances in Intelligent Systems and Computing, Springer International Publishing, Cham, 2018, S. 225–234
- [Ped17] Peddie, J.: Augmented reality – where we will all live, 2017
- [PER17] Palmarini, R.; Erkoyuncu, J. A.; Roy, R.: An Innovative Process to Select Augmented Reality (AR) Technology for Maintenance. Procedia CIRP, (59), 2017, S. 23–28
- [Per20] Perera, M.: Build your first HoloLens 2 Application with Unity and MRTK 2.3.0: AR/VR Journey: Augmented & Virtual Reality Magazine, 2020
- [PGG13] Pannaga, N.; Ganesh, N.; Gupta, R.: Mechatronics – An Introduction to Mechatronics. International Journal of Engineering Research & Technology (IJERT), (2)8, 2013
- [PH17] Porter, M. E.; Heppelmann, J. E.: A Manager's Guide to Augmented Reality – Why Every Organization needs an Augmented Reality Strategy: Harvard Business Review, 2017, S. 45–57
- [Pho22-ol] PHOTON: PUN – The Ease-of-use of Unity's Networking with the Performance & Reliability of Photon Realtime. Unter: <https://www.photonengine.com/pun>, 1. Februar 2022
- [PI18] Paes, D.; Irizarry, J.: A Usability Study of an Immersive Virtual Reality Platform for Building Design Review: Considerations on Human Factors and User Interface. In: Wang, C. (Ed.): Construction Information Technology. Construction Research Congress 2018, April 2–4, 2018, New Orleans, Louisiana, American Society of Civil Engineers, Reston, 2018, pp. 419–428
- [PIP21] Paes, D.; Irizarry, J.; Pujoni, D.: An evidence of cognitive benefits from immersive design review: Comparing three-dimensional perception and presence between immersive and non-immersive virtual environments. Automation in Construction, (130), 2021
- [PL11] Ponn, J.; Lindeemann, U.: Konzeptentwicklung und Gestaltung technischer Produkte: systematisch von Anforderungen zu Konzepten und Gestaltlösungen. Springer-Verlag, 2011
- [Por18-ol] PORSCHE CARS NORTH AMERICA, INC: Porsche "Tech Live Look" Pioneers Augmented Reality In U.S. Auto Repairs. Unter: <https://www.prnewswire.com/news-releases/porsche-tech-live-look-pioneers-augmented-reality-in-us-auto-repairs-300655775.html>, 27. Juni 2024
- [PRR12] Probst, G. J. B.; Raub, S.; Romhardt, K.: Wissen managen – Wie Unternehmen ihre wertvollste Ressource optimal nutzen. 7. Auflage, Springer Gabler, Wiesbaden, 2012
- [PSM08] Papagiannakis, G.; Singh, G.; Magnenat-Thalmann, N.: A survey of mobile and wireless technologies for augmented reality systems. Computer Animation and Virtual Worlds, (19)1, 2008, S. 3–22
- [PT20] Pacholczyk, D.; Trzaska, M.: Common Data Format in Visual Assembling Guidance Using Augmented Reality. In: Arai, K.; Bhatia, R.; Kapoor, S. (Hrsg.): Proceedings of the Future Technologies Conference (FTC) 2019 – Volume 1. 2020, Cham, Advances in Intelligent Systems and Computing, Springer International Publishing, Cham, 2020, S. 306–317
- [PTC19] PTC: Buyer's Guide für industrielle Augmented Reality – Die richtigen Lösungen für ihr Unternehmen finden, 2019
- [PTC21-ol] PTC: Getting Started with Vuforia Engine in Unity | VuforiaLibrary. Unter: <https://library.vuforia.com/articles/Training/getting-started-with-vuforia-in-unity.html>, 27. Juni 2024
- [PTC22a-ol] PTC: Vuforia Enterprise Augmented Reality (AR) Software. Unter: <https://www.ptc.com/de/products/vuforia>, 27. Juni 2024
- [PTC22b-ol] PTC: Vuforia Studio Augmented Reality for Industrial Enterprise | PTC. Unter: <https://www.ptc.com/en/products/vuforia/vuforia-studio>, 27. Juni 2024
- [PTC22c-ol] PTC: Vuforia Chalk Augmented Reality (AR) Remote Assistance | PTC. Unter: <https://www.ptc.com/en/products/vuforia/vuforia-chalk>, 27. Juni 2024

- [Rät17-ol] RÄTSCH, C.: Individualisierung und Personalisierung von Produkten – Christian Rätsch Blog. Unter: <https://christianraetsch.de/individualisierung-und-personalisierung-von-produkten/>, 20. Mai 2021
- [RBD15] ROBERT, S.; BJÖRN, F.; DANIEL, F.: Efficient Validation During Product Development Using a Self-optimizing Inspection System. *Procedia CIRP*, (33), 2015, S. 47–52
- [RC16] RUNDE, C.; CANNAROZZI, M.: Delphi-Studie Digitales Engineering 2025. Virtual Dimension Center, 2016
- [RD20] RÖLTGEN, D.; DUMITRESCU, R.: Classification of industrial Augmented Reality use cases. *Procedia CIRP*, (91), 2020, S. 93–100
- [Rea21-ol] REALWEAR: RealWear HMT-1. Unter: <https://realwear.at/hmt-1>, 27. Juni 2024
- [Rei06] REICHLE, M.: Bewertungsverfahren zur Bestimmung des Erfolgspotenzials und des Innovationsgrades von Produktideen und Produkten. Dissertation, Institut für Konstruktionstechnik und Technisches Design, Universität Stuttgart, 2006
- [Rei21] REINEMANN, J.: Entwicklung einer Systematik zur Validierung interaktiver Produkte in Augmented-Reality-Umgebungen in der Frühen Phase im Modell der PGE – Produktgenerationsentwicklung. Dissertation, IPEK - Institut für Produktentwicklung, Karlsruher Institut für Technologie, 2021
- [RFH+19] REINEMANN, J.; FAHL, J.; HIRSCHTER, T.; ALBERS, A.: Augmented Reality in der Produktvalidierung: Potenziale und Grenzen in frühen Entwicklungsphasen. 5th International Conference on Electrical Engineering and Electronics (EEE'19), Lissabon, P, August 21 - 23, 2019, 2019
- [RLH96] REINHART, G.; LINDEMANN, U.; HEINZL, J.: Qualitätsmanagement – Ein Kurs für Studium und Praxis. Springer Berlin Heidelberg, Berlin, Heidelberg, 1996
- [RLS+15] RIASCOS, R.; LEVY, L.; STJEPANDIĆ, J.; FRÖHLICH, A.: Digital Mock-up. In: Stjepandić, J.; Wognum, N.; J.C. Verhagen, W. (Eds.): Concurrent Engineering in the 21st Century – Foundations, Developments and Challenges. Springer International Publishing, Cham, 2015, pp. 355–388
- [Röl21] RÖLTGEN, D.: Systematik zur strategischen Planung von Augmented Reality für das Produkt-Service-Geschäft. Dissertation, Heinz Nixdorf Institut, Universität Paderborn, 2021
- [RP09] REICHWALD, R.; PILLER, F.: Interaktive Wertschöpfung – Open Innovation, Individualisierung und neue Formen der Arbeitsteilung. 2. Auflage, Gabler, Wiesbaden, 2009
- [SAM09] SCHUMANN, M.; ACHILLES, S.; MÜLLER, S.: Analysis by synthesis techniques for markerless tracking. 6. Workshop der GI Fachgruppe VR/AR, Braunschweig, 2009
- [SB15] SATTER, K.; BUTLER, A.: Competitive Usability Analysis of Immersive Virtual Environments in Engineering Design Review. *Journal of Computing and Information Science in Engineering*, (15)3, 2015
- [SCB17] SICARU, I.; CIOCIANU, C.; BOIANGIU, C.-A.: A survey on Augmented Reality: *Journal of Information Systems & Operations Management*, 2017
- [Sch08a] SCHILLING, T.: Augmented Reality in der Produktentstehung. Dissertation, Fakultät für Maschinenbau, Technische Universität Ilmenau, 2008
- [Sch08b] SCHADY, R.: Methode und Anwendungen einer wissensorientierten Fabrikmodellierung, Otto-von-Guericke-Universität Magdeburg, 2008
- [Sch09] SCHMIDT, J.: Einführung Ontologiemanagement, Institut für Informatik, Universität Leipzig, 2009
- [Sch19] SCHMIDT, G.: Augmented Reality-Konzepte zur Verbesserung der Distanzeinschätzung für die Navigation von chirurgischen Instrumenten, Institut für Simulation und Graphik, Otto-von-Guericke-Universität Magdeburg, 2019
- [Sch98] SCHNEIDER, P.: Produktindividualisierung als Marketing-Ansatz. Dissertation, Universität St. Gallen, 1998
- [SCK+17] SCHMIDT, T. S.; CHAHIN, A.; KÖBLER, J.; PAETZOLD, K.: AGILE DEVELOPMENT AND THE CONSTRAINTS OF PHYSICALITY: A NETWORK THEORY-BASED CAUSE-AND-EFFECT ANALYSIS, 2017

- [SCP+21-ol] SEMPLE, K.; COULTER, D.; PARK, Y.; EVELEIGH, K.: MRTK-Unity Developer Documentation. Unter: <https://docs.microsoft.com/de-de/windows/mixed-reality/mrtk-unity/>, 29. März 2021
- [SDH+15] SYBERFELDT, A.; DANIELSSON, O.; HOLM, M.; WANG, L.: Visual Assembling Guidance Using Augmented Reality. *Procedia Manufacturing*, (1), 2015, S. 98–109
- [Sem22-ol] SEMPLE, K.: MRTK 2.7 is Out Now with Official Support for OpenXR - Microsoft Tech Community. Unter: <https://techcommunity.microsoft.com/t5/mixed-reality-blog/mrtk-2-7-is-out-now-with-official-support-for-openxr/ba-p/2397313>, 2. Februar 2022
- [Sen22-ol] SENSEGLOVE: SenseGlove for developers. Unter: <https://www.senseglove.com/developer/>, 1. Februar 2022
- [SH16] SCHMALSTIEG, D.; HOLLERER, T.: Augmented reality – Principles and practice. Addison-Wesley, Boston, 2016
- [SHY+18] SPEICHER, M.; HALL, B. D.; YU, A.; ZHANG, B.; ZHANG, H.; NEBELING, J.; NEBELING, M.: XD-AR: Challenges and Opportunities in Cross-Device Augmented Reality Application Development. *Proceedings of the ACM on Human-Computer Interaction*, (2)EICS, 2018, S. 1–24
- [Shy17] SHYLESH, S.: A Study of Software Development Life Cycle Process Models: National Conference on Reinventing Opportunities in Management, IT, and Social Sciences, 2017
- [Sie19-ol] SIEMENS DIGITAL INDUSTRIES SOFTWARE: Siemens adds augmented reality for enhanced design visualization and collaboration to Solid Edge 2020. Unter: <https://www.plm.automation.siemens.com/global/en/our-story/newsroom/solid-edge-2020/61815>, 28. Juli 2021
- [Sie20-ol] SIEMENS DIGITAL INDUSTRIES SOFTWARE: JT Open Toolkit | Siemens Software. Unter: <https://www.plm.automation.siemens.com/global/en/products/plm-components/jt-open-toolkit.html>, 1. Februar 2022
- [Ska19-ol] SKARREDGHOST: Augmented reality is reaching a mature state according to Gartner. Unter: <https://skarredghost.com/2019/09/04/augmented-reality-mature-gartner/>, 12. Mai 2020
- [SPB+20] SU, S.; PERRY, V.; BRAVO, L.; KASE, S.; ROY, H.; COX, K.; DASARI, V. R.: Virtual and Augmented Reality Applications to Support Data Analysis and Assessment of Science and Engineering. *1558-366X*, (22)3, 2020, S. 27–39
- [SPS10] SPROLL, S.; PEISSNER, M.; STURM, C.: From product concept to user experience. In: Hvannberg, E. T. (Ed.): *Proceedings of the 6th Nordic Conference on Human-Computer Interaction – Extending boundaries* ; Reykjavik, Iceland - October 16 - 20, 2010. the 6th Nordic Conference, 10/16/2010 - 10/20/2010, Reykjavik, Iceland, ACM, New York, NY, 2010, p. 473
- [SSA+00] SCHREIBER, A. T.; SCHREIBER, G.; AKKERMANS, H.; ANJEWIERDEN, A.; SHADBOLT, N.; HOOG, R. DE; VAN VELDE, W. DE; WIELINGA, B.; SHADBOLT, N. R.: *Knowledge engineering and management – The CommonKADS methodology*. 3rd Edition, MIT Press, Cambridge, Mass., 2000
- [Sta22-ol] STANFORD CENTER FOR BIOMEDICAL INFORMATICS RESEARCH: Protégé – A free, open-source ontology editor and framework for building intelligent systems. Unter: <https://protege.stanford.edu/>, 27. Januar 2022
- [Sta73] STACHOWIAK, H.: Allgemeine Modelltheorie. Springer, Wien, 1973
- [Ste19-ol] STEUERWALD, K.: IKEA Unternehmensblog - Mit der IKEA App per Augmented Reality einrichten. Unter: <https://ikea-unternehmensblog.de/article/2019/ikea-place-app>, 28. Juli 2021
- [Ste22-ol] STEUERWALD, K.: Augmented Reality mit der IKEA Place App. Unter: <https://www.ikea.com/de/de/this-is-ikea/corporate-blog/ikea-place-app-augmented-reality-puba55c67c0>, 4. Februar 2022
- [Str08] STRANG, T.: Lokalisierungsverfahren. Shaker, Aachen, 2008

- [Str96] STRUBE, G. (Hrsg.): Wörterbuch der Kognitionswissenschaft. Klett-Cotta, Stuttgart, 1996
- [TAB19-ol] TAB - BÜRO FÜR TECHNIKFOLGEN-ABSCHÄTZUNG BEIM DEUTSCHEN BUNDESTAG: Virtuelle und erweiterte Realitäten (VAR): Was das ist und was sie uns bringen | TAB. Unter: <https://www.tab-beim-bundestag.de/de/aktuelles/20190625.html>, 12. Mai 2020
- [Teg06] TEGTMEIER, A.: Augmented Reality als Anwendungstechnologie in der Automobilindustrie. Dissertation, Otto-von-Guericke-Universität Magdeburg, 2006
- [TLM14] TÜRK, D.; LEUTENACKER, B.; MEBOLDT, M.: Experience the relevance of testing in engineering design education. Proceedings of the 10th international CDIO conference, 2014
- [Tön10] TÖNNIS, M.: Augmented Reality – Einblicke in die Erweiterte Realität. Informatik im Fokus Band 0, Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg, 2010
- [Tur14] TURKI, T.: Bedeutung von Erfahrungswissen in der Produktentwicklung und Ansätze zu dessen Evaluierung und Transfer am Beispiel studentischer Gruppen. Dissertation, Institut für Produktentwicklung, Karlsruher Institut für Technologie, 2014
- [Tut21-ol] TUTORIALS FOR AR: Day 1 Creating an AR App for iOS using Unity and ARKit. Unter: <https://tutorialsforar.com/creating-an-ar-app-for-ios-using-unity-and-arkit/>, 31. Januar 2022
- [UE12] ULRICH, K. T.; EPPINGER, S. D.: Product design and development. 5. Auflage, McGraw-Hill/Irwin, New York, 2012
- [Uhl03] UHLMANN, E.: Wandel der Fabrik durch Produktindividualisierung. In: Reinhart, G.; Zäh, M. F. (Hrsg.): Marktchance Individualisierung. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, S. 119–127
- [Ult22-ol] ULTRALEAP: Ultraleap Plugin for Unity - Ultraleap for Developers. Unter: <https://developer.leapmotion.com/unity>, 1. Februar 2022
- [Uni22a-ol] UNITY TECHNOLOGIES: Unity - Manual: Prefabs. Unter: <https://docs.unity3d.com/Manual/Prefabs.html>, 27. Juni 2024
- [Uni22b-ol] UNITY TECHNOLOGIES: Unity - Manual: Unity's interface. Unter: <https://docs.unity3d.com/2022.1/Documentation/Manual/UsingTheEditor.html>, 27. Juni 2024
- [Uni22c-ol] UNITY TECHNOLOGIES: Unity - Manual: Asset workflow. Unter: <https://docs.unity3d.com/2022.1/Documentation/Manual/AssetWorkflow.html>, 27. Juni 2024
- [Uni22d-ol] UNITY TECHNOLOGIES: What platforms are supported by Unity? Unter: <https://support.unity.com/hc/en-us/articles/206336795-What-platforms-are-supported-by-Unity->, 27. Juni 2024
- [Uni22e-ol] UNITY TECHNOLOGIES: Unity - Manual: GameObjects. Unter: <https://docs.unity3d.com/2022.1/Documentation/Manual/GameObjects.html>, 27. Juni 2024
- [Uni22f-ol] UNITY TECHNOLOGIES: Unity - Manual: Built-in packages. Unter: <https://docs.unity3d.com/2022.1/Documentation/Manual/pack-build.html>, 31. Januar 2022
- [Uni22g-ol] UNITY TECHNOLOGIES: Netcode | Unity Multiplayer Networking. Unter: <https://docs-multiplayer.unity3d.com/>, 27. Juni 2024
- [Uni22h-ol] UNITY TECHNOLOGIES: Pixyz. Unter: <https://unity.com/products/pixyz>, 27. Juni 2024
- [Uni22i-ol] UNITY TECHNOLOGIES: Unity - Manual: Multiplayer and Networking. Unter: <https://docs.unity3d.com/Manual/UNet.html>, 1. Februar 2022
- [Uni22j-ol] UNITY TECHNOLOGIES: Unity - Manual: Supported Model file formats. Unter: <https://docs.unity3d.com/2020.1/Documentation/Manual/3D-formats.html>, 27. Juni 2024
- [Uni22k-ol] UNITY TECHNOLOGIES: Unity - Scripting API: ContextMenu. Unter: <https://docs.unity3d.com/ScriptReference/ContextMenu.html>, 31. Januar 2022

- [Uni22l-ol] UNITY TECHNOLOGIES: Unity - Manual: Unity User Manual 2022.1 (beta). Unter: <https://docs.unity3d.com/2022.1/Documentation/Manual/UnityManual.html>, 31. Januar 2022
- [Uni22m-ol] UNITY TECHNOLOGIES: Download Unity. Unter: <https://unity3d.com/get-unity/download>, 31. Januar 2022
- [Uni22n-ol] UNITY TECHNOLOGIES: Unity's AR Foundation Framework. Unter: <https://unity.com/unity/features/arfoundation>, 2. Februar 2022
- [Uni22o-ol] UNITY TECHNOLOGIES: Machine Learning Agents. Unter: <https://unity.com/products/machine-learning-agents>, 1. Februar 2022
- [Uni22p-ol] UNITY TECHNOLOGIES: Unity - Scripting API: ScriptableWizard. Unter: <https://docs.unity3d.com/ScriptReference/ScriptableWizard.html>, 31. Januar 2022
- [Uni22q-ol] UNITY TECHNOLOGIES: Unity. Unter: <https://unity.com/>, 31. Januar 2022
- [UVV19] URBASA, U.; VRABIĆA, R.; VUKAŠINOVICA, N.: Displaying Product Manufacturing Information in Augmented Reality for Inspection. *Procedia CIRP*, (81), 2019, S. 832–837
- [VAC+09] VOGEL, O.; ARNOLD, I.; CHUGHTAI, A.; IHLER, E.; KEHRER, T.; MEHLIG, U.; ZDUN, U.: Software-Architektur – Grundlagen - Konzepte - Praxis. 2. Auflage, Spektrum Akademischer Verlag, Heidelberg, 2009
- [VB92] VIVIAN, J. E.; BERKOWITZ, N. H.: Anticipated bias from an outgroup: An attributional analysis. *European Journal of Social Psychology*, (22)4, 1992, S. 415–424
- [VDC19-ol] VDC FELLBACH: Augmented Reality: 2030 werden 400.000 Menschen allein in Deutschland mit AR-Technologien arbeiten. Unter: <https://www.vdc-fellbach.de/nachrichten/2019/12/11/augmented-reality-2030-werden-400000-menschen-allein-in-deutschland-mit-ar-technologien-arbeiten/>, 12. Mai 2020
- [VDI/VDE3694:2014] VEREIN DEUTSCHER INGENIEURE: Lastenheft/Pflichtenheft für den Einsatz von Automatisierungssystemen – VDI/VDE 3694, 2014
- [VDI2206:2004] VEREIN DEUTSCHER INGENIEURE: Entwicklungsmethodik für mechatronische Systeme. VDI-Richtlinie 2206, Beuth Verlag, Berlin, 2004
- [VDI5610:2009] VEREIN DEUTSCHER INGENIEURE: Wissensmanagement im Ingenieurwesen – Grundlagen, Konzepte, Vorgehen. VDI 5610, Beuth Verlag GmbH, 2009
- [VHN09] VERLINDEN, J.; HORVÁTH, I.; NAM, T.-J.: Recording augmented reality experiences to capture design reviews. *International Journal on Interactive Design and Manufacturing (IJIDeM)*, (3)3, 2009, S. 189–200
- [Vis22-ol] VISOMETRY: VisionLib – Augmented Reality Tracking for Industries. Unter: <https://visionlib.com/>, 2. Februar 2022
- [Vog13] VOGEL-HEUSER, B.: Automation als Enabler für Industrie 4.0 in der Produktion auf Basis von Cyber Physical Systems. In: Vogel-Heuser, B. (Hrsg.): *Engineering von der Anforderung bis zum Betrieb. Embedded Systems*. 1, Tagungen und Berichte, Band 3, Kassel University Press, Kassel, 2013
- [VP21] VAKALIUK, T. A.; POCHTOVIUK, S. I.: Analysis of tools for the development of augmented reality technologies: 4th International Workshop on Augmented Reality in Education. AREdu, 11. Mai, Ukraine, 2021
- [vuz22-ol] VUZIX: Vuzix Blade Upgraded Smart Glasses. Unter: <https://www.vuzix.com/products/vuzix-blade-smart-glasses-upgraded>, 18. Januar 2022
- [vVG+20] VAN GOETHEM, S.; VERLINDEN, J.; GARCIA, I.; DEGRYSE, R.; PERDIEUS, S.; WATTS, R.; VERWULGEN, S.: The Use of Immersive Technologies for Design Reviews. In: Ahram, T.; Falcão, C. (Hrsg.): *Advances in Usability, User Experience, Wearable and Assistive Technology. Advances in Intelligent Systems and Computing*, Springer International Publishing, Cham, 2020, S. 656–662
- [W3C08-ol] W3C: Punning - OWL. Unter: <https://www.w3.org/2007/OWL/wiki/Punning>, 27. Juni 2024
- [Wan05] WANDKE, H.: Assistance in human–machine interaction: a conceptual framework and a proposal for a taxonomy. *Theoretical Issues in Ergonomics Science*, (6)2, 2005, S. 129–155

- [Web22-ol] WEBGL FUNDAMENTALS: Scene Graph. Unter: <https://webglfundamentals.org/webgl/lessons/webgl-scene-graph.html>, 31. Januar 2022
- [Wei06] WEILKIENS, T.: Systems engineering mit SysML/UML: modellierung, analyse, design. Dpunkt-Verlag, 2006
- [Wei11] WEINMANN, A.: Anwendungsbezogenes Tracking – Proseminar Augmented Reality in der Anwendung, Institut für Computervisualistik, Universität Koblenz, 2011
- [WEL07-ol] WELT: Growian: Größter Misserfolg und Helfer der Windparks. Unter: <https://www.welt.de/wirtschaft/article1129577/Groesster-Misserfolg-und-Helfer-der-Windparks.html>, 27. Juni 2024
- [WEM+13] WIEBEL, M.; EIFLER, T.; MATHIAS, J.; KLOBERDANZ, H.; BOHN, A.; BIRKHOFER, H.: Modellierung von Unsicherheit in der Produktentwicklung: Exploring Uncertainty. Springer Gabler, Wiesbaden, 2013, S. 245–269
- [Wer20-ol] WERKZEUGMASCHINENLABOR WZL DER RWTH AACHEN: Kompetenzen der Agilen Entwicklung am RWTH Aachen Campus. Unter: <https://www.agile-produktentwicklung.com/>, 15. April 2020
- [Wes17] WESTERMANN, T.: Systematik zur Reifegradmodell-basierten Planung von Cyber-Physical Systems des Maschinen-und Anlagenbaus. Dissertation, Heinz Nixdorf Institut, Universität Paderborn, 2017
- [WK14] WENDE, J.; KIRADJIEV, P.: Eine Implementierung von Losgröße 1 nach Industrie-4.0-Prinzipien. e & i Elektrotechnik und Informationstechnik, (131)7, 2014, S. 202–206
- [WLB12] WONG, W.; LIU, W.; BENNAMOUN, M.: Ontology learning from text. ACM Computing Surveys, (44)4, 2012, S. 1–36
- [Wol19] WOLFARTSBERGER, J.: Analyzing the potential of Virtual Reality for engineering design review. Automation in Construction, (104), 2019, S. 27–37
- [Wol21-ol] WOLFF, C.: Xiaomi Smart Glasses: Brille mit integriertem MicroLED-Display vorgestellt. Unter: <https://www.notebookcheck.com/Xiaomi-Smart-Glasses-Brille-mit-integriertem-MicroLED-Display-vorgestellt.561184.0.html>, 18. Januar 2022
- [WZS+17] WOLFARTSBERGER, J.; ZENISEK, J.; SIEVI, C.; SILMBROTH, M.: A virtual reality supported 3D environment for engineering design review: 23rd International Conference on Virtual System & Multimedia (VSMM). IEEE, 2017
- [XRG21-ol] XRG - WE CONNECT THE INDUSTRY WITH X-REALITY: AR 3S - Augmented Reality Engineering Space | XRG Business Software. Unter: <https://xrgo.io/en/product/ar-3s/>, 4. Februar 2022
- [Yu14] YU, L.: A developer's guide to the semantic web. 2nd Edition, Springer, Berlin, 2014
- [ZC18] ZAKER, R.; COLOMA, E.: Virtual reality-integrated workflow in BIM-enabled projects collaboration and design review: a case study. Visualization in Engineering, (6)1, 2018, S. 1–15
- [Zed13] ZEDLITZ, J.: Konzeptuelle Modellierung mit UML und OWL – Untersuchung der Gemeinsamkeiten und Unterschiede mit Hilfe von Modelltransformationen. Dissertation, Christian-Albrechts-Universität zu Kiel, 2013

Anhang

Inhaltsverzeichnis	Seite
A1 Ergebnisse der Unternehmensbefragung	A-1
A2 Ergänzungen zum Stand der Technik.....	A-7
A2.1 Systematische Planung AR-basierter Validierung	A-7
A2.1.1 Hauptmerkmale eines Produkts nach PAHL ET AL.	A-7
A2.1.2 Wiedergabetreue von Prototypen	A-9
A2.2 Wissensbasierte AR-System-Konfiguration	A-10
A2.2.1 Die Struktur wissensbasierter Systeme	A-10
A2.2.2 Systematische Wissensmodell-Erstellung nach KOHN.....	A-12
A2.2.3 Technische Details zur Web Ontology Language (OWL)	A-14
A2.2.4 Technische und funktionale Aspekte eines AR-Systems.....	A-16
A2.2.5 AR-Endgeräte	A-18
A2.2.6 AR-Anwendungs- und Erfahrungswissen	A-20
A2.2.7 Einflussfaktoren auf AR-Systeme aus der Einsatzumgebung	A-21
A2.2.8 Das Ontologie-Modellierungswerkzeug Protégé.....	A-24
A2.3 Aufwandsarme Entwicklung von AR-Validierungssoftware	A-26
A2.3.1 Entwicklungsumgebung Unity	A-26
A2.3.2 Datenvorbereitung und -integration	A-29
A2.3.3 Ergänzungen zu AR-SDKs	A-31
A2.3.4 Studie zu AR-basierter Produktvalidierung	A-33
A2.3.5 Usability-Untersuchungen von PAES UND IRIZARRY.....	A-34
A3 Ergänzungen zum Instrumentarium	A-37
A4 Ergänzungen zur Anwendung und Bewertung	A-40

A1 Ergebnisse der Unternehmensbefragung

Die Unternehmensbefragung wurde im Februar 2021 über das Online-Werkzeug *SurveyMonkey*¹⁸ durchgeführt. Die Umfrage wurde über verschiedene Mailverteiler (bspw. über das Netzwerk OWL ViProSim) sowie über direkte Kommunikation mit Projektpartnern kommuniziert. 23 Vertreter von verschiedenen Unternehmen haben an der Umfrage teilgenommen. Die folgenden Screenshots zeigen die acht Fragen und die entsprechenden Antworten bzw. Rückmeldungen, wie sie von *SurveyMonkey* bereitgestellt werden.

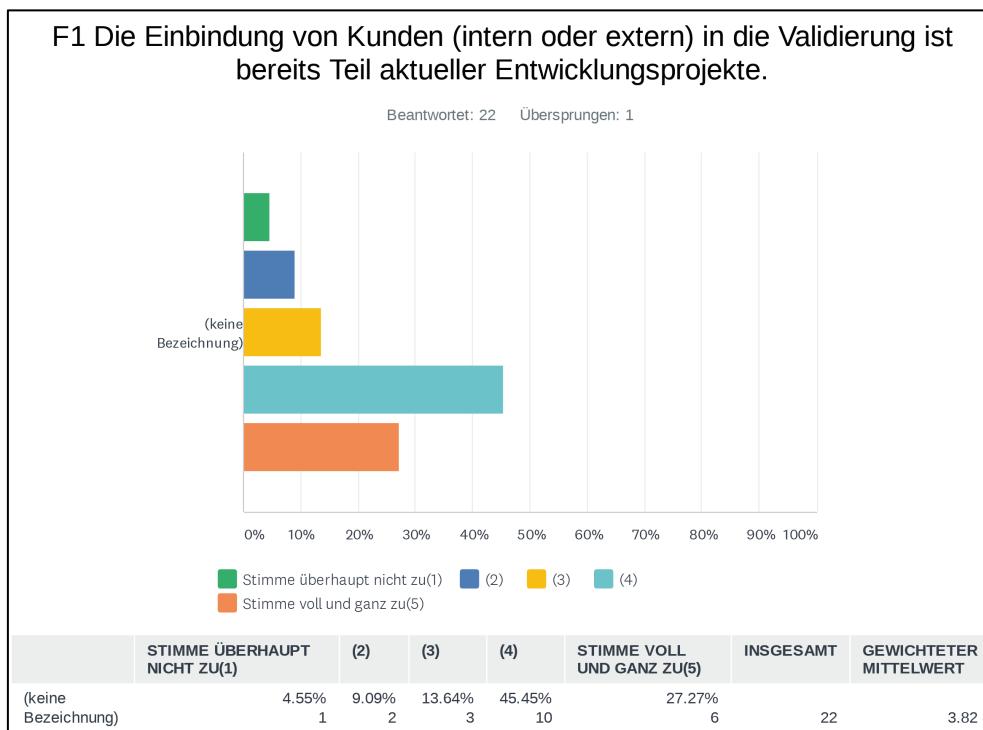


Abbildung A-1: Frage 1

¹⁸ <https://www.surveymonkey.de/>

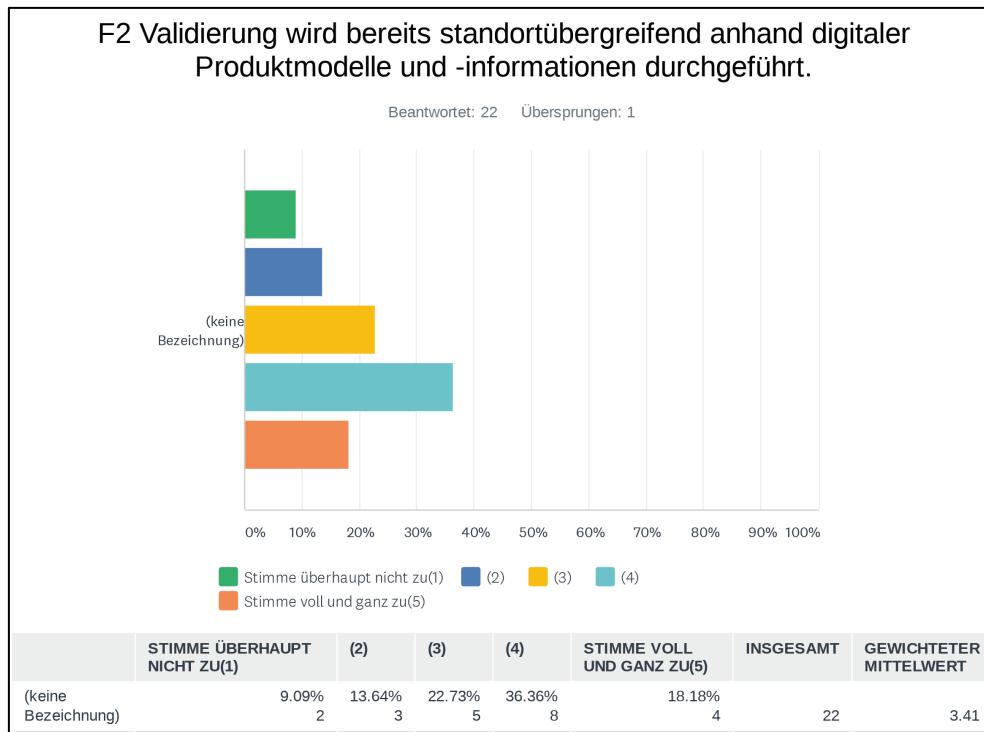


Abbildung A-2: Frage 2

F3 Zum besseren Verständnis der zuvor abgegebenen Antworten, beschreiben Sie bitte kurz, wie Validierung in aktuellen Entwicklungsprojekten durchgeführt wird.

Beantwortet: 14 Übersprungen: 9

#	BEANTWORTUNGEN	DATE
1	Wir nutzen seit ca. 9 Monaten VR-/AR- für Akquisition und auch bei der internen Produktentwicklung. Im frühen Entwicklungsstadium werden anhand der VR-/AR-Modelle Potentiale, Risiken, etc. im Team besprochen. Kundenspezifische Lösungen können mittels VR unseren Kunden präsentiert werden. AR stellt sich teilweise noch etwas schwieriger dar, da die Aufwände teilweise noch zu hoch sind. Es gibt viele Ideen diese Themen voranzutreiben und als Mehrwert unseren Kunden anzubieten. Wir zwar mehrere Firmen (Produktions-Firma, Vertriebs- und Service-Firma) und nutzen gerne für interne Abstimmungen die Tools, sind aber alle an einem Standort!	2/10/2021 10:09 AM
2	.. nein Betriebsgeheimnis	2/10/2021 7:47 AM
3	z.B. entwicklungsübergreifende Prüfungen, Typprüfungen, Prototypen, digitale Produktmodelle und Produktdaten, Simulation verschiedener Verhaltensaspekte, Pilotanwendungen	2/9/2021 8:11 PM
4	Produkteigenschaften werden durch Lasten-/Pflichtenheft durch Produktmanager als Sprachrohr des externen Kunden festgelegt. Deren Überprüfung erfolgt während der Entwicklungsphase überwiegend virtuell, durch Simulation.	2/9/2021 3:13 PM
5	Präsentation der CAD-Modelle durch den Entwickler. Diskussion der Zusammenhänge. Beurteilung der Infos der Fachbereiche anhand von Checklisten	2/9/2021 1:08 PM
6	DesignThinking, Kundenbefragungen, Kundentest, Testmarkt, QFD	2/9/2021 9:16 AM
7	Simulation, 3D CAD Modelle, 3D Rendering	2/9/2021 8:42 AM
8	Anhand Austausch von Anforderungen / projektmappen und 3 D Modellen	2/8/2021 4:39 PM
9	Austausch von Informationen und DAten	2/8/2021 4:19 PM
10	Validierungsplan --> Validierungsarbeitspaket --> Reifegrad des Prüflings --> Validierung --> Kommunikation/Vermittlung der Ergebnisse --> Dokumentation	2/8/2021 3:26 PM
11	Größtenteils durch interne Prüfungen und Tests	2/8/2021 1:01 PM
12	2 und 3D CAD Modelle	2/8/2021 12:46 PM
13	Auf Basis von Prototypen und 3D Modellen	2/8/2021 12:21 PM
14	3D CAD oder mittels JT-Modellen in virtuellen Meetings	2/8/2021 12:07 PM

Abbildung A-3: Frage 3

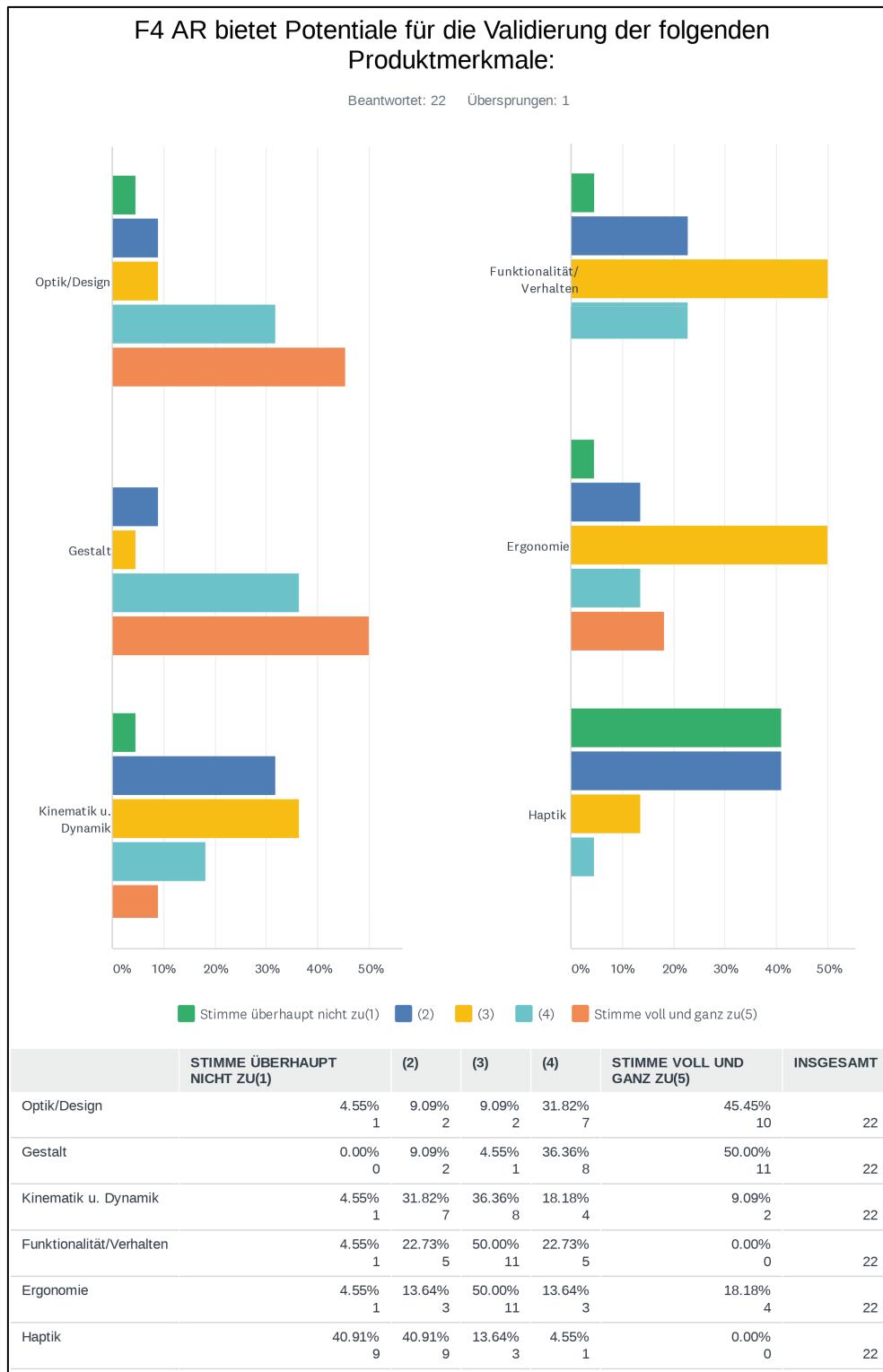


Abbildung A-4: Frage 4

#	WELCHE WEITEREN PRODUKTMERKMALE KÖNNEN AUS IHRER SICHT MIT AR VALIDIERT WERDEN?	DATE
1	Maße in Räumlichkeiten, Einbausituationen	2/12/2021 5:15 PM
2	Überprüfung Montageaufnahmen/Lehren Überprüfung Saugerposition an Spinnen und Zuführgeräte bei Pressen	2/10/2021 8:11 AM
3	Form / Fit	2/9/2021 1:11 PM
4	Die oben genannten Potenziale können in der Regel nicht gesamtheitlich durch ein AR-Modell erreicht werden. Häufig müssen spezifische AR-Modelle für die einzelnen Produktmerkmale eingesetzt werden. Weiteres Potenzial sehe ich im Laborbereich in der Kombination von AR-Modellen mit realen Prototypen zum Vergleichen, Verstehen und Ideen generieren.	2/9/2021 9:22 AM
5	Einbauverhalten , Simulation digitaler Zwilling/ , Raumgestaltung; Lastenheft ERstellung	2/8/2021 4:42 PM
6	AR ist aus meiner Sicht nur eine Anzeige, ein Vermittlungswerkzeug. Die eigentliche Validierung passiert nicht in/mit der VR. Nur da wo die Vermittlung von Eigenschaften Kern der Validierung ist, macht VR Sinn.	2/8/2021 3:31 PM
7	Einpassen in die Kundenumgebung	2/8/2021 12:09 PM

Abbildung A-5: Ergänzungen zu Frage 4

F5 In welchen Phasen der Entwicklung sehen Sie AR als hilfreich zur Unterstützung der Validierung (einzelner Produktaspekte) an?		
Beantwortet: 18 Übersprungen: 5		
#	BEANTWORTUNGEN	DATE
1	Frühe Überprüfung erster Konzepte, schnelle Überprüfung von optischen und gestalterischen Aspekten auch in späteren Phasen, Erreichbarkeitstests bevor physische Prototypen verfügbar sind	2/12/2021 5:15 PM
2	von frühen Phasen bis hin zu späteren Phasen inkl. Markteinführung	2/12/2021 10:33 AM
3	In der Vor- und Serienentwicklung	2/11/2021 9:55 AM
4	Wir sind bemüht nach einen PEP (Produktentstehungsprozess) zu arbeiten. In den Phasen "Innovation" - hier speziell bei der Konzeptfindung und anschließend nach Freigabe in der Phase der Entwicklung sehen wir Potentiale.	2/10/2021 10:23 AM
5	in der frühen Phase der Vorentwicklung und bei späteren Änderungen	2/10/2021 8:11 AM
6	..bis Prototyping	2/10/2021 7:49 AM
7	Prinzipiell in allen Phasen der Produktentstehung, in erster Linie in den frühen Phasen (Konzeptentwicklung / Design), weniger in der Phase der Konstruktion / Fertigungsvorbereitung	2/9/2021 8:14 PM
8	Insbesondere in sehr frühen Stadien der Konzeptionierung. Je detaillierter der Entwurf wird, desto geringer schätze ich die Möglichkeiten von AR ein, diese Details noch vermitteln zu können.	2/9/2021 3:16 PM
9	Diskussion von Mock-ups mit Kunden Entwicklungsphase Sobald ein physikalischer Prototyp zur Verfügung steht, ist dieser sicher besser geeignet.	2/9/2021 1:11 PM
10	In allen Phasen des PEP können spezifische AR-Modelle sinnvoll eingesetzt werden.	2/9/2021 9:22 AM
11	Durchgehend von der Ideenfindung bis hin zu Phasen mit mittleren bis hohen Reifegraden der betrachteten Produkte / Komponenten.	2/9/2021 8:46 AM
12	Produktentstehung / Vermarktung - Lastenheft/ Einbau / REalisierung / Betriebsmittel Beschaffung / Vermarktung / Aqoise	2/8/2021 4:42 PM
13	Handhabung	2/8/2021 4:20 PM
14	Ergebnisvermittlung	2/8/2021 3:31 PM
15	Zur Prüfung von zusammenhängenden Funktionalitäten	2/8/2021 1:03 PM
16	Konzeptfindung	2/8/2021 12:47 PM
17	Planung, Inbetriebnahme, service	2/8/2021 12:22 PM
18	Angebotsphase beim Kunden Validierung mit dem Kunden	2/8/2021 12:09 PM

Abbildung A-6: Frage 5

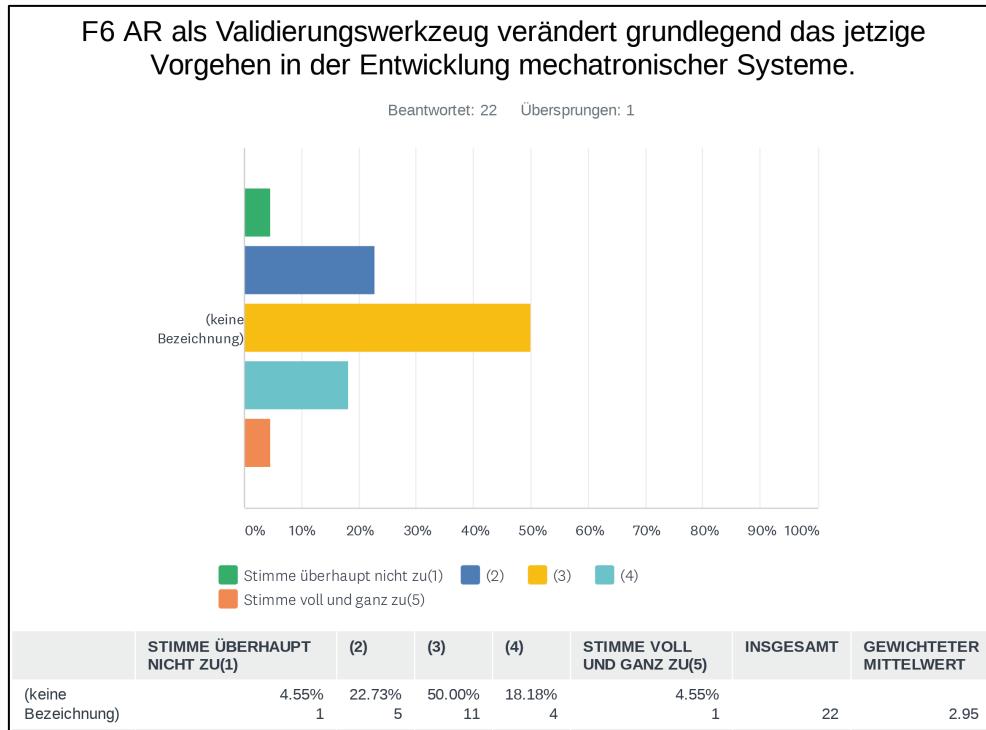


Abbildung A-7: Frage 6

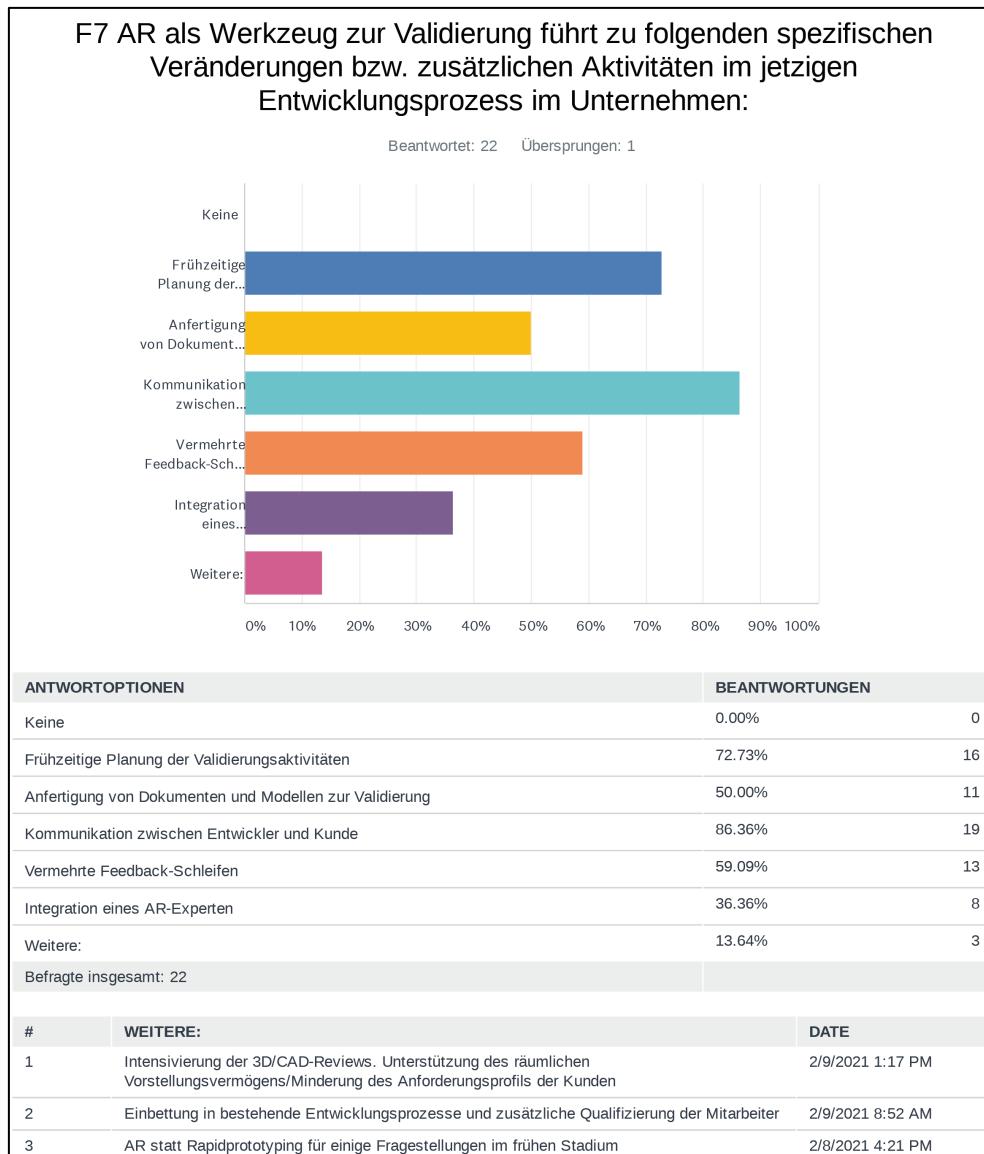


Abbildung A-8: Frage 7

F8 Welche weiteren Herausforderungen sehen Sie bei der Etablierung von AR zur Produktvalidierung als Standardwerkzeug in Ihrer Produktentwicklung?		
Beantwortet: 12 Übersprungen: 11		
#	BEANTWORTUNGEN	DATE
1	Die Entwicklung der Anwendungen und die Beschaffung der Geräte	2/12/2021 5:16 PM
2	- notwendige Investitionen und Schulungen - Änderung von Prozessen - Akzeptanz bei Mitarbeitern	2/11/2021 9:57 AM
3	Teilweise noch zu aufwendig für uns. Wir bauen teilweise große Anlagen, die in bestehende Räumlichkeiten eingebaut werden. Diese bestehenden Räumlichkeiten sind meist durch andere Maschinen, Ausrüstungen, etc. belegt und sollen durch eine neue Anlage umgerüstet werden. Der Aufwand zur Bereinigung kann unter Umständen den Mehrwert in Frage stellen.	2/10/2021 10:27 AM
4	Akzeptanz der Methode Erforderliche Investitionen Anwendungsgebiete, Anwender-Rollen sind nicht definiert.	2/10/2021 8:14 AM
5	größere Datenmengen, mehr oder z.T. zu viele Infos	2/10/2021 7:51 AM
6	Kosten, bzw. Kosten-Nutzen-Verhältnis. Es muss einfach ersichtlich werden, dass die Vorteile der AR über "man kann sich etwas in 3D anschauen" hinaus gehen, um die Hürde "Anschaffung" zu überspringen.	2/9/2021 3:18 PM
7	Standartisierung der Datenformate Verfügbarkeit der Hardware	2/9/2021 1:17 PM
8	Schnelle Verfügbarkeit der spezifischen AR-Modelle aufgrund der vielen, schnellen Veränderungen in den frühen Produktentwicklungsphasen. Die AR-Modelle müssen nahezu auf Knopfdruck zur Verfügung stehen.	2/9/2021 9:25 AM
9	Nutzen und Aufwand darstellen. Wird häufig als Spielzeug abgetan. Amortisation des Invest	2/8/2021 4:44 PM
10	Geeignete use cases zu finden, bei denen der erreichbare Mehrwert den Aufwand übersteigt. Wo ist die Vermittlung von Ergebnissen so dominant?	2/8/2021 3:34 PM
11	Akzeptanz in der Produktentwicklung	2/8/2021 12:23 PM
12	Ausstattung mit der spezifischen Hardware. Finden des geeigneten Personals, Eingewöhnung des Personals.	2/8/2021 12:11 PM

Abbildung A-9: Frage 8

A2 Ergänzungen zum Stand der Technik

In Kapitel 4 wurde Stand der Technik bzgl. der Handlungsfelder dieser Arbeit untersucht. Dieser Anhang ergänzt die Untersuchungen um weitere Ansätze und Informationen, die für die Erarbeitung der Lösungen in den Handlungsfeldern relevant sind.

A2.1 Systematische Planung AR-basierter Validierung

Im Folgenden werden hilfreiche Informationen zur systematischen Planung AR-basierter Validierung ergänzt.

A2.1.1 Hauptmerkmale eines Produkts nach PAHL ET AL.

Zur Unterstützung der Aufstellung von Anforderungslisten für Produkte werden oft Leitfäden oder Checklisten verwendet. Dabei handelt es sich meist um Sammlungen von typischen Anforderungsarten [PL11, S. 42]. Entsprechende Checklisten ermöglichen eine systematische Identifikation von Anforderungen und verhindern, dass wichtige Anforderungen vergessen werden. Ein Beispiel einer solchen Checkliste ist die **Leitlinie basierend auf Hauptmerkmalen eines Produkts** nach PAHL ET AL. [PBF+07]. Die in Abbil-

dung A-10 dargestellten sogenannten Hauptmerkmale decken alle relevanten Aspekte eines Produkts ab. Dazu zählen neben technisch-funktionalen Aspekten wie der Geometrie oder der Kinematik auch Aspekte, die die Produktion und spätere Lebensphasen des Produkts betreffen. Für jedes Hauptmerkmal werden Beispiele genannt. Für das Hauptmerkmal Geometrie können Anforderungen bspw. die Größe, Höhe oder auch den Durchmesser betreffen. Das Hauptmerkmal Ergonomie adressiert vor allem die Mensch-Maschine-Interaktion, unter anderem die Bedienung oder die Übersichtlichkeit.

Hauptmerkmal	Beispiele
Geometrie	Größe, Höhe, Breite, Länge, Durchmesser, Raumbedarf, Anzahl, Anordnung, Anschluss, Ausbau und Erweiterung
Kinematik	Bewegungsart, Bewegungsrichtung, Geschwindigkeit, Beschleunigung
Kräfte	Kraftgröße, Krafrichtung, Krafthäufigkeit, Gewicht, Last, Verformung, Steifigkeit, Federeigenschaften, Stabilität, Resonanzen
Energie	Leistung, Wirkungsgrad, Verlust, Reibung, Ventilation, Zustandsgrößen wie Druck, Temperatur, Feuchtigkeit, Erwärmung, Kühlung, Anschlussenergie, Speicherung, Arbeitsaufnahme, Energieumformung
Stoff	Physikalische und chemische Eigenschaften des Eingangs- und Ausgangsprodukts, Hilfsstoffe, vorgeschriebene Werkstoffe (Nahrungsmittelgesetz u. ä.), Materialfluss und -transport
Signal	Eingangs- und Ausgangssignale, Anzeigeart, Betriebs- und Überwachungsgeräte, Signalform
Sicherheit	Unmittelbare Sicherheitstechnik, Schutzsysteme, Betriebs-, Arbeits- und Umweltsicherheit
Ergonomie	Mensch-Maschine-Beziehung: Bedienung, Bedienungsart, Übersichtlichkeit, Beleuchtung, Formgestaltung
Fertigung	Einschränkung durch Produktionsstätte, größte herstellbare Abmessung, bevorzugtes Fertigungsverfahren, Fertigungsmittel, mögliche Qualität und Toleranzen
Kontrolle	Mess- und Prüfmöglichkeit, besondere Vorschriften (TÜV, ASME, DIN, ISO, AD-Merkblätter)
Montage	Besondere Montagevorschriften, Zusammenbau, Einbau, Baustellenmontage, Fundamentierung
Transport	Begrenzung durch Hebezeuge, Bahnprofil, Transportwege nach Größe und Gewicht, Versandart und -bedingungen
Gebrauch	Geräuscharmut, Verschleißrate, Anwendung und Absatzgebiet, Einsatzort (z. B. schwefelige Atmosphäre, Tropen,...)
Instandhaltung	Wartungsfreiheit bzw. Anzahl und Zeitbedarf der Wartung, Inspektion, Austausch und Instandsetzung, Anstrich, Säuberung
Recycling	Wiederverwendung, Wiederverwertung, Entsorgung,endlagerung, Beseitigung
Kosten	Max. zulässige Herstellkosten, Werkzeugkosten, Investition und Amortisation
Termin	Ende der Entwicklung, Netzplan für Zwischenschritte, Lieferzeit

Abbildung A-10: Hauptmerkmale eines Produkts und zugehörige Beispiel-Aspekte [PBF+07]

Beim Einsatz der Leitlinie wird davon ausgegangen, dass eine Menge an Anforderungen bereits definiert wurde. Dies können grundlegende Anforderungen sein, die vom Kunden

als selbstverständlich angesehen werden, oder explizit und spezifisch vom Kunden genannte technische Anforderungen. Anhand der Liste der Hauptmerkmale und der zugehörigen Beispiele lassen sich ausgehend von den bereits definierten Anforderungen weitere Erkenntnisse und damit verbundene zusätzliche Anforderungen für die vorliegende Entwicklungsaufgabe herleiten. Dazu werden für jedes Hauptmerkmale entsprechende Leitfragen bereitgestellt. Die Leitlinie wurde bereits weiterentwickelt und die Hauptmerkmale um organisatorische und marktrelevante Aspekte ergänzt [BG21, S. 195].

Bewertung: Die Hauptmerkmale eines Produkts nach PAHL ET AL. stellen eine ganzheitliche und generalisierte Übersicht der möglichen Merkmale eines Produkts dar. Anforderungen an ein Produkt können lediglich diese Merkmale betreffen. Die Hauptmerkmale bilden daher eine sehr gute Grundlage zur Klassifikation und Generalisierung von Produktanforderungen im Rahmen der Potentialbewertung für AR.

A2.1.2 Wiedergabetreue von Prototypen

Prototypen dienen der Repräsentation eines Produkts. Allerdings ist das Ziel von Prototypen meist nicht die Repräsentation des Produkts als Ganzes. Vielmehr werden lediglich einzelne für eine Validierungsaufgabe relevante Aspekte eines Produkts abgebildet. Dieser Ansatz spart zum einen Aufwände bei der Erstellung eines Prototyps. Zum anderen sorgen andere für die Validierung irrelevante Aspekte nicht für Ablenkung. Prototypen haben entsprechend eine Art Filterfunktion für ein Produkt. Zur Bewertung von Prototypen dient die sogenannte Wiedergabetreue (engl.: Fidelity). Sie ist definiert als der Grad, zu dem ein Prototyp dem finalen Produkt entspricht [TLM14]. Eine hohe Wiedergabetreue bedeutet entsprechend, dass der Prototyp sehr realitätsnah ist und dem Produkt sehr nah kommt. Entsprechend der Filterfunktion von Prototypen wird die Wiedergabetreue bezüglich einzelner Aspekte definiert. Ein Prototyp hat meist eine hohe Wiedergabetreue für die relevanten Aspekte und gleichzeitig eine niedrige Wiedergabetreue für andere Aspekte. Um eine einheitliche Beschreibung der Wiedergabetreue zu ermöglichen, haben LIM ET AL. fünf Dimensionen der Wiedergabetreue und entsprechende Beispielvariablen definiert (Abbildung A-11) [LST08]. Die Dimensionen umfassen die Erscheinung, Daten, die Funktionalität, die Interaktivität sowie die räumliche Struktur.

Basierend auf den fünf Dimensionen der Wiedergabetreue wurde von KOHLER ET AL. das sogenannte **Filter-Fidelity-Profil** entwickelt [KHD+13]. Dabei wird die Wiedergabetreue von Prototypen jeweils bzgl. der Variablen der verschiedenen Dimensionen von 0 bis 5 bewertet und die Bewertung anschließend grafisch dargestellt. 0 bedeutet, die Variable ist in der vorliegenden Situation nicht relevant. Ein Wert von 1 bedeutet, dass die Variable im Prototyp nicht ausgestaltet ist. Realitätsnah abgebildete Variablen erhalten den Wert 5. Durch die gemeinsame Bewertung und Darstellung mehrerer Prototypen im Filter-Fidelity-Profil können diese gut miteinander verglichen werden.

Bewertung: Die Dimensionen der Wiedergabetreue nach LIM ET AL. ähneln in ihrer Art den Hauptmerkmalen eines Produkts nach PAHL ET AL. Bei beiden Ansätzen werden

ganzheitlich und generisch Merkmale beschrieben, die ein Produkt haben kann bzw. die durch einen Produktprototyp abgebildet werden können. REINEMANN leitet daraus 22 Dimensionen der Wiedergabetreue AR-basierter Prototypen ab (vgl. Abschnitt 4.1.5). Diese sind für die vorliegende Arbeit hochrelevant, da es sich um einen bereits auf den Kontext AR zugeschnittenen Katalog an Produktmerkmalen handelt.

Filtering dimension	Example variables
<i>Appearance</i>	size; color; shape; margin; form; weight; texture; proportion; hardness; transparency; gradation; haptic; sound
<i>Data</i>	data size; data type (e.g. number; string; media); data use; privacy type; hierarchy; organization
<i>Functionality</i>	system function; users' functionality need
<i>Interactivity</i>	input behavior; output behavior; feedback behavior; information behavior
<i>Spatial structure</i>	arrangement of interface or information elements; relationship among interface or information elements—which can be either two- or three-dimensional, intangible or tangible, or mixed

Abbildung A-11: Die fünf Dimensionen und jeweilige Beispiel-Variablen von Prototypen nach LIM ET AL. [LST08]

A2.2 Wissensbasierte AR-System-Konfiguration

In diesem Abschnitt werden Ergänzungen zu wissensbasierten Systemen sowie Details zu verschiedenen für die AR-System-Konfiguration relevanten Wissensaspekten beschrieben.

A2.2.1 Die Struktur wissensbasierter Systeme

Ein besonderes Merkmal wissensbasierter Systeme ist die strikte Trennung des Wissens von der Logik und Funktionalität des Software-Systems. Wie bereits kurz in Abschnitt 4.2 beschrieben, bestehen wissensbasierte Systeme daher im Wesentlichen aus zwei Komponenten [Kur89]:

- Wissensbasis
- Problemlösungskomponente

Die klare Trennung der Wissensbasis von der Problemlösungskomponente bildet den Unterschied zu konventionellen Softwarelösungen, in denen Wissen typischerweise starr im Quellcode verankert ist [Kur89].

In der **Wissensbasis** wird das für die Problemlösung relevante Wissen in einer geeigneten Form gespeichert und formalisiert (Abbildung A-12). Unter anderem wird das Wissen direkt von einem Knowledge Engineer erstellt. Die **Problemlösungskomponente** ist Teil

des Steuerungssystems. Das Steuerungssystem enthält zudem eine **Wissensakquisitionskomponente**, über die Fachexperten in Zusammenarbeit mit dem Knowledge Engineer weiteres Wissen in der Wissensbasis hinzufügen können. Je nach Art der Wissensquellen kann Wissen in der Wissensbasis direkt angelegt oder durch intelligente Mechanismen automatisiert aus Daten generiert werden. Beim Einsatz des wissensbasierten Systems interagiert der Endbenutzer mit einer **Dialogkomponente** im Steuerungssystem. Über die Dialogkomponente wird fallspezifisches Wissen definiert und so das zu lösende Problem spezifiziert. Das fallspezifische Wissen dient als Input für die Problemlösungskomponente. Diese greift entsprechend des vorliegenden Problems gezielt auf die Wissensbasis zu, verarbeitet und analysiert das Wissen und zieht entsprechende Schlüsse zur Lösung des Problems. Über eine **Erklärungskomponente** werden die Ergebnisse zurück an die Dialogkomponente gespielt und dem Endbenutzer in geeigneter und nachvollziehbarer Form ausgegeben. Die Wissensbasis und die Problemlösungskomponente stellen zwar einzelne Komponenten dar, sie sind jedoch eng miteinander verknüpft. Je nach Problemstellung und Wissensverarbeitung muss die Wissensbasis das Wissen in geeigneter Form abbilden. Umgekehrt beeinflusst die Repräsentation des Wissens in der Wissensbasis stark die Möglichkeiten der Wissensverarbeitung in der Problemlösungskomponente [Kur89]. Beide Komponenten müssen daher gut aufeinander abgestimmt sein.

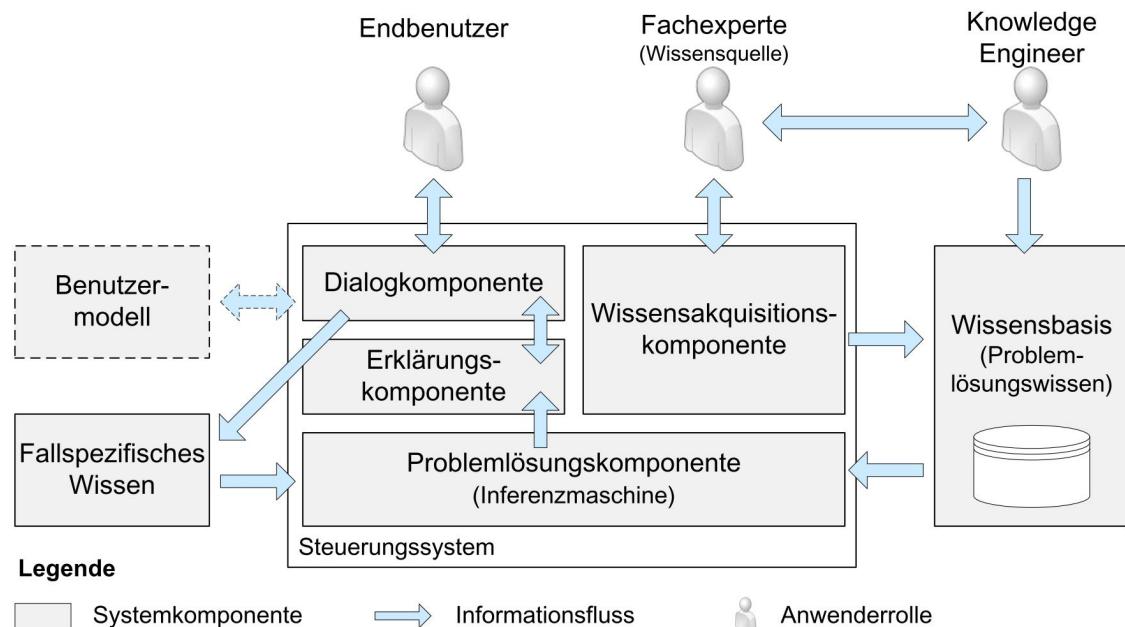


Abbildung A-12: Grundsätzliche Struktur wissensbasierter Systeme nach [Kur89], zitiert aus [Bau15]

Bewertung: Die durch KURBEL beschriebene Struktur stellt eine wichtige Grundlage für die Entwicklung wissensbasierter Systeme dar. Die Struktur ermöglicht eine flexiblere Modellierung und kontinuierliche Erweiterung des Wissens, ohne den Quellcode der Konfigurations-Software zu ändern. Im Kontext der AR-System-Konfiguration, im dem

vielfältige Wissensaspekte berücksichtigt werden müssen, ist diese Art der Wissensverarbeitung vorteilhaft. Eine direkte Integration des Wissens in ein Softwaresystem wäre nicht praktikabel.

A2.2.2 Systematische Wissensmodell-Erstellung nach KOHN

Wissensbasierte Systeme sollen die Differenz des für eine Problemlösung notwendigen Wissens und des vorhandenen Wissens ausgleichen und das fehlende Wissen situationspezifisch bereitstellen. Dazu gibt es laut KOHN zwei Möglichkeiten der **Verwendung von Wissensquellen als Wissensbasis** (Abbildung A-13) [Koh14].

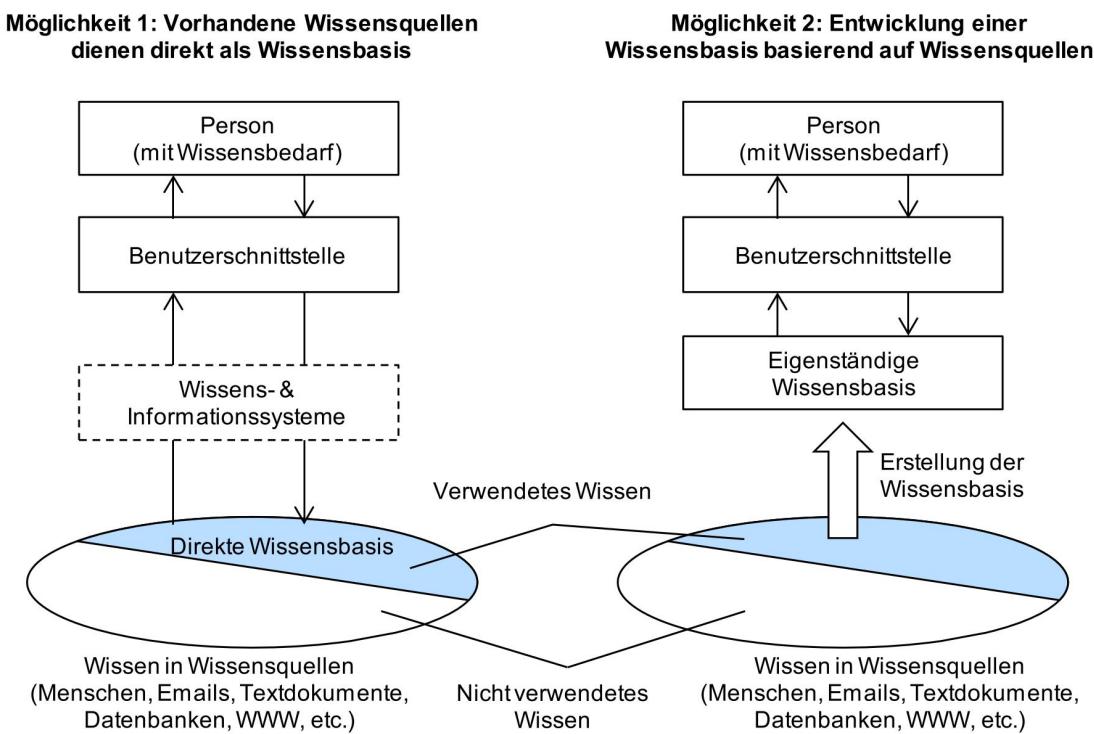


Abbildung A-13: Zwei Möglichkeiten der Verwendung von Wissensquellen als Wissensbasis [Koh14]

Bei Möglichkeit 1 werden die vorhandenen Wissensquellen direkt verwendet. Dies entspricht der Funktionalität von Wissens- und Informationssystemen. Über geeignete Benutzerschnittstellen wird direkt auf das Wissen, das beispielsweise in Textdokumenten oder Datenbanken oder auch im Internet vorliegt, zugegriffen. Such- oder Strukturierungsmechanismen können den Zugriff unterstützen. Schwierig ist bei Möglichkeit 1 vor allem die geeignete Aufbereitung des Wissens sowie die Auswahl der relevanten Wissensaspekte innerhalb der Wissensquellen. Möglichkeit 2 ist die Erstellung einer eigenständigen Wissensbasis basierend auf dem in den Wissensquellen vorhandenen für eine Aufgabe relevanten Wissen. In einer solchen Wissensbasis wird das Wissen für einen speziellen Einsatzzweck aufbereitet und formalisiert. Entsprechende Wissensbasen können als Grundlage für wissensbasierte Systeme zur Lösung von spezifischen Problemstellungen

verwendet werden. Die Erstellung einer Wissensbasis sowie zugehöriger Benutzerschnittstellen erfolgt im Rahmen des **Knowledge Engineerings**.

Zur **Erstellung von Wissensmodellen** im Rahmen des Knowledge Engineerings beschreibt KOHN die in Abbildung A-14 dargestellte systematische Vorgehensweise. Ausgangspunkt ist eine Menge an ausgewählten Wissensquellen. Dabei kann es sich um Artefakte handeln oder um Experten. Die Wissensmodelle stellen die Grundlage eines wissensbasierten Systems dar. Dieses System dient einem bestimmten Zweck bzw. soll eine spezifische Problemstellung unterstützen. Eine wichtige Aufgabe der Wissensakquise als Teil des Knowledge Engineerings ist daher die Identifikation des relevanten Wissens aus den ausgewählten Wissensquellen sowie eine geeignete Beschreibung des Wissens [SSA+00]. Zur Unterstützung der Wissensakquise und später für die Implementierung des wissensbasierten Systems existieren verschiedene Methodiken. Grundsätzlich sollten bei beiden Schritten fünf ursprünglich von MILTON beschriebene Prinzipien beachtet werden [Mil07]:

- **Systematik:** Methoden sollen systematisch eingesetzt werden, um die Wissensakquisition effizient und nachvollziehbar zu gestalten.
- **Vorhandenes Wissen:** Bei der Erstellung von Wissensmodellen soll wenn möglich auf vorhandenes Wissen und bereits aufbereitete Wissensmodelle zurückgegriffen werden, um die Aufwände zu reduzieren.
- **Techniken:** Es sollten geeignete Techniken für die Erhebung, Analyse und Modellierung des Wissens verwendet werden.
- **Software:** Spezielle Software kann die Wissensakquisition unterstützen und die Durchführung vereinfachen.
- **Lernen:** Ein Erfahrungstransfer und das Lernen aus anderen Wissensakquisitions-Aktivitäten sind sinnvoll und hilfreich.

Bewertung: KOHN beschreibt zwei Möglichkeiten der Verwendung von Wissensquellen als Wissensbasis. Für die vorliegende Arbeit ist die Entwicklung einer eigenständigen Wissensbasis, die das für einen spezifischen Einsatzzweck relevante Wissen beinhaltet, interessant. Die von KOHN vorgeschlagene systematische Vorgehensweise zur Erstellung von Wissensmodellen kann grundsätzlich auf die Erstellung einer solchen Wissensbasis für die AR-System-Konfiguration übertragen werden. Aus einer Menge an ausgewählten Wissensquellen gilt es somit, das relevante Wissen zu identifizieren und geeignet zu beschreiben. Dieser notwendige Schritt wurde bereits in Kapitel 3.5 identifiziert. Mit den fünf Prinzipien nach MILTON liefert der Ansatz von KOHN zudem wichtige bei der Erarbeitung der Wissensbasis zu berücksichtigende Aspekte. Sie verdeutlichen unter anderem die Notwendigkeit geeigneter Techniken für die Modellierung des Wissens sowie von Software-Lösungen zur Unterstützung der Wissensakquise. Diese Aspekte werden im Abschnitt 4.2.1 genauer analysiert.

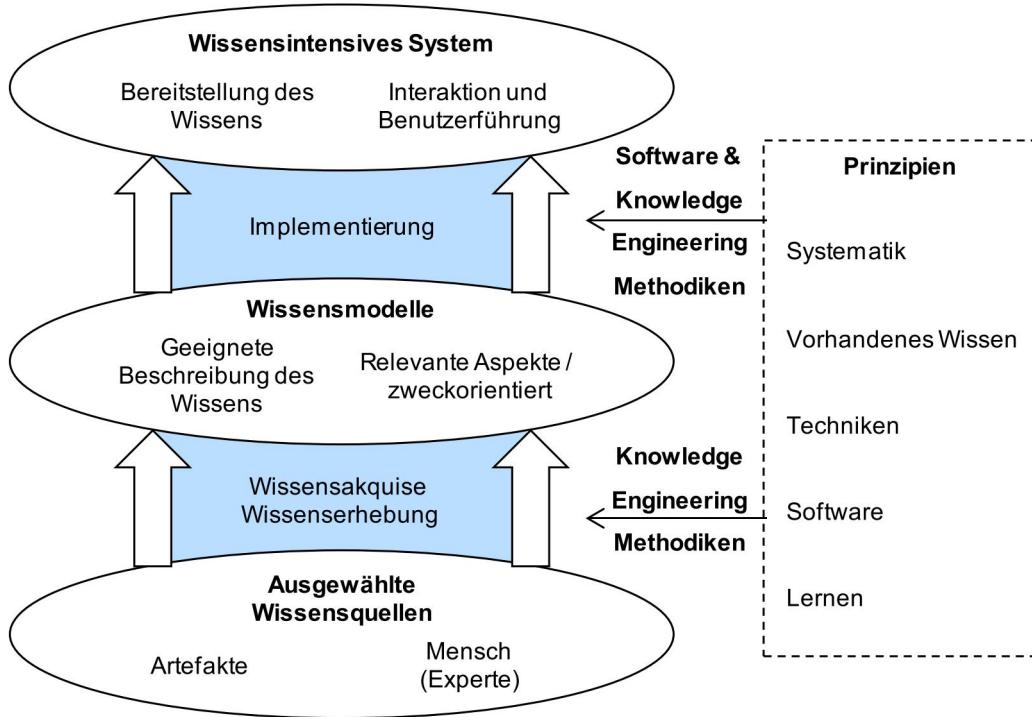


Abbildung A-14: Systematische Vorgehensweise zur Erstellung von Wissensmodellen im Rahmen des Knowledge Engineerings [Koh14]

A2.2.3 Technische Details zur Web Ontology Language (OWL)

Die Formalisierung von Wissen wird in der OWL über verschiedene **Grundbausteine** realisiert. Klassen werden über den Grundbaustein *owl:Class* definiert. Beziehungen werden als *Properties* bezeichnet. Mit *owl:ObjectProperties* lassen sich Beziehungen zwischen Individuen und Klassen definieren. *owl:DatatypeProperties* dienen der Zuweisung von Attributen zu einer Klasse oder Instanz. Attribute können beispielsweise Zahlenwerte (*Integer* oder *Double*) oder auch Zeichenfolgen (*Strings*) sein. Grundsätzlich baut die OWL auf dem RDFS-Vokabular auf. Die OWL erweitert dieses jedoch um Ausdrucksmöglichkeiten zur Beschreibung komplexer Zusammenhänge zwischen Klassen. So werden unter anderem Kardinalitäten, Disjunktionen und Restriktionen ermöglicht. Zudem können Beziehungen spezielle Charakteristika zugewiesen werden. So kann eine Beziehung *transitiv*, *symmetrisch*, *invers* oder *funktional* sein [Har11].

Eine OWL-basierte Wissensbasis besteht in der Regel aus einer **Terminological Box (TBox)** und einer **Assertional Box (ABox)** [Fel14]. Die *TBox* wird auch „Schema“ genannt. Sie beschreibt die Struktur des Wissens und enthält entsprechend Informationen über die enthaltenen Klassen, deren Klassenhierarchien sowie über grundlegende Zusammenhänge und Attributtypen. Ein Beispiel ist die Information „Alle Kameras sind Sensoren“. Die *ABox* enthält konkretes Wissen über Individuen der Klassen und konkrete Beziehungen zwischen Individuen. Die *ABox* wird auch „Daten“ genannt. Ein Beispiel für

eine Information der *ABox* ist, dass eine „Sony Alpha6000“ eine Instanz der Klasse „Kamera“ ist. Jede einzelne Information wird in der OWL als **Axiom** gespeichert. Axiome entsprechen den Tripeln des RDF bestehend aus Subjekt, Prädikat und Objekt. Prädikate können vielseitig sein und unter anderem Klassenhierarchien (z.B. *subclassOf*), Beziehungen (z.B. *isPartOf*) oder Attributzuweisungen (z.B. *hasPrice*) repräsentieren [MPP12-ol]. Jede mittels OWL definierte Ontologie entspricht einem gültigen RDF-Dokument und somit einem durch entsprechende Software-Werkzeuge lesbaren wohlgeformten XML-Dokument [Hef16-ol].

Zur Definition von Regeln in der OWL wurde die **Semantic Web Rule Language (SWRL)** entwickelt [MLB+16]. SWRL-Regeln stellen ein wichtiges Werkzeug zur Steigerung der Expressivität einer Ontologie dar. Eine Regel besteht aus einer *Bedingung* und einer *Aktion*, die jeweils über Atome beschrieben werden. Mehrere Atome können mit einem logischen *UND* verbunden werden. Ein Atom setzt sich aus den Klassen und Beziehungen der Ontologie zusammen und bezieht sich auf Individuen oder Wertebereiche. So können über Atome Klassenzugehörigkeiten oder vorhandene Beziehungen abgefragt werden. Zusätzlich ermöglicht die SWRL durch Built-Ins (*swrlb*) mathematische Vergleiche, Berechnungen oder auch Datumsabfragen in der *Bedingung*. Variablen in den Atomen werden mit einem Fragezeichen markiert. Sie dienen als Platzhalter, die bei der Anwendung der Regel durch mögliche Werte der Variablen ersetzt werden. Wenn alle Atome der *Bedingung* erfüllt sind, müssen auch die Atome der *Aktion* erfüllt sein [Bau15]. So kann über die *Aktion* einer Regel neues Wissen generiert werden. Die folgende Regel überprüft beispielsweise, ob ein Element ein Buch ist, das Buch einen Preis hat und dieser Preis größer als 200 ist. Ist dies der Fall, wird das Buch automatisch der Klasse *ExpensiveBook* zugeordnet.

$$Book(?b) \wedge hasPrice(?b, ?p) \wedge swrlb:greaterThan(?p, 200) \rightarrow ExpensiveBook(?b)$$

Neben Regeln dienen **Restriktionen** als weiteres Instrument zur Ableitung von Wissen. Restriktionen entsprechen Einschränkungen von Klassen, die von jedem Individuum der Klasse erfüllt werden müssen. Beispielsweise gilt für Individuen der Klasse Fahrrad, dass sie genau (*exactly*) zwei Räder haben. Zudem kann über Restriktionen definiert werden, dass ein Individuum einer Klasse eine Beziehung zu mindestens einem Individuum einer anderen Klasse hat (*some*) oder ausschließlich zu Individuen einer spezifischen Klasse oder einer Menge an Klassen (*only*). Ist eine Restriktion als *necessary condition* definiert, müssen alle Individuen der Klasse die Restriktion erfüllen. Die Restriktion bedeutet umgekehrt jedoch nicht, dass ein Individuum, das die Restriktion erfüllt, automatisch der Klasse zugeordnet wird. Dies gilt dagegen bei *necessary and sufficient conditions*. In diesem Fall würde beispielsweise ein Individuum mit genau (*exactly*) zwei Rädern automatisch der Klasse Fahrrad zugeordnet. [Hor11]

Die Abfrage von Informationen aus RDF-Datenmodellen erfolgt über die Abfragesprache SPARQL (SPARQL Protocol And RDF Query Language) [Yu14]. Aufgrund der erweiterten

terten Semantik der OWL gegenüber RDF wurde speziell für OWL-Ontologien die **Semantic Query-enhanced Web Rule Language (SQWRL)** entwickelt [OD09]. SQWRL basiert auf der SWRL. Der Aufbau von SQWRL-Suchanfragen ähnelt daher stark dem von SWRL-Regeln. Anstatt einer *Aktion* enthalten Suchanfragen neben der *Bedingung* jedoch eine Aufforderung zur Ausgabe der Anfrage-Ergebnisse. Diese Aufforderung wird über den Operator *sqwrl:select* definiert. Eine der wichtigsten Besonderheiten von SQWRL gegenüber SPARQL ist die Möglichkeit, Sammlungen von Elementen zwischenspeichern. Dazu dient der Operator *sqwrl:makeSet*. Die beispielhafte Suchanfrage

$$\text{Person}(\textit{?p}) \wedge \text{hasAge}(\textit{?p}, \textit{?a}) \wedge \text{swrlb: lessThan}(\textit{?a}, 9) \rightarrow \text{sqwrl:select}(\textit{?p}, \textit{?a})$$

gibt alle in der Ontologie enthaltenen Personen, die ein Alter kleiner als Neun haben, zusammen mit ihrem Alter aus. Die Ausgabe erfolgt in Tabellenform.

Ein wichtiger bei der Arbeit mit OWL-Ontologien zu berücksichtigender Aspekt ist der **Open World Ansatz (OWA)** [OD09]. Beim OWA wird davon ausgegangen, dass das repräsentierte Wissen unvollständig ist [DS06]. Informationen, die nicht in der Ontologie angelegt wurden, sind nicht automatisch falsch. Sie können vergessen worden sein oder die Ontologie ist noch nicht vollständig angelegt. Der OWA führt dazu, dass manche Anfragen nicht möglich sind. Beispielsweise liefern die Anfragen „*Liste alle Personen, die mehr als zwei Fahrzeuge besitzen*“ und „*Liste alle Personen, die kein Fahrzeug besitzen*“ keine Ergebnisse. Relationen zu Fahrzeugen könnten vergessen worden sein oder sie wurden noch nicht angelegt.

Es existieren verschiedene **Software-Werkzeuge**, die die Überführung von UML-Modellen in OWL-Ontologien unterstützen. Unter anderem sind *UML2OWL* [Lei20-ol], *um-ITUowl* [GM12] und *UMLtoOWL* [Gas07-ol] bekannt. Keines der Werkzeuge setzt alle laut ZEDLITZ möglichen Transformationsregeln um. Zudem legen die Werkzeuge individuelle Richtlinien für die UML-Modelle zu Grunde. Mit dem **Ontology UML Profile** wurde ein Profil zur Erstellung von UML-Modellen entwickelt, das eine direkte Übertragung in eine OWL-Ontologie mittels eines spezifischen Software-Werkzeugs ermöglicht [DGD+05]. Dieses Profil ist jedoch sehr komplex und UML-Modelle werden umfangreich und unübersichtlich. Unter anderem müssen fast alle Klassen mit Stereotypen gekennzeichnet werden, beispielsweise als „*OntClass*“ [Zed13].

Bewertung: Bei der Entscheidung für den Einsatz semantischer Technologien und speziell der OWL, sind die hier beschriebenen Informationen sowie weitere Details aus den genannten Quellen hochrelevant für die technische Realisierung einer entsprechenden Lösung.

A2.2.4 Technische und funktionale Aspekte eines AR-Systems

Im Rahmen dieser Arbeit wird unter dem AR-System die für AR notwendige Hardware sowie die darauf laufende Software verstanden. AR-System-Architekturen werden in einer Vielzahl an Publikationen in unterschiedlichen Detaillierungsgraden beschrieben.

Erste AR-Systemkonzepte wurden von AZUMA vorgestellt (Abbildung A-15). Die Konzepte Video-See-Through, Monitor-basiert sowie Optical-See-Through beinhalten jeweils einen Monitor, der entweder direkt oder über optische Combiner wahrgenommen wird. Zudem verfügt ein AR-System nach AZUMA unter anderem über einen Tracker, einen Szenengenerator sowie Kameras. Ähnlich dazu wurde unter anderem das Augmented and Mixed Reality (MAR) Referenz-Modell in der ISO/IEC18039 definiert [ISO/IEC18039:2019]. Es verfügt über eine zentrale Ausführungseinheit, die basierend auf Sensorik virtuelle Inhalte über eine Display-Einheit an den Benutzer ausgibt. Zudem existieren weitere Publikationen, die ähnliche übergeordnete System-Strukturen beschreiben.

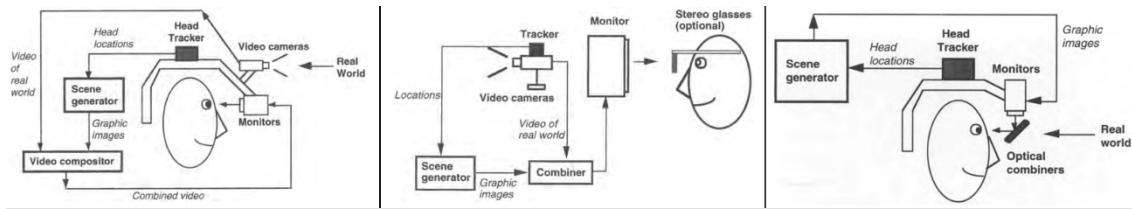


Abbildung A-15: AR-Systemkonzepte nach Azuma [Azu97]: Video-See-Through (links), Monitor-basiert (Mitte), Optical-See-Through (rechts)

Grundsätzlich existiert eine enorme Menge an Publikationen und Berichten zu **technischen und funktionalen Aspekten eines AR-Systems**. Eine Analyse der Beiträge von 2008 bis 2017 zur ISMAR als eine der wichtigen AR/VR-Konferenzen zeigt, dass jährlich zahlreiche Publikationen zu Themen wie Tracking, Displays, Interaktion, Visualisierung und AR-Apps veröffentlicht werden [KBB+18]. All diese Publikationen enthalten Wissen, dass die AR-System-Konfiguration unterstützen kann. Einzelne Komponenten wie die Display-Einheit, das Tracking oder auch die Interaktion werden beispielsweise von BILLINGHURST ET AL. genauer beschrieben und dabei verschiedene technische und funktionale Ausprägungen verglichen [BCL15]. Unter anderem wird genauer auf Prozessoren eingegangen, die für verschiedene Klassen von AR notwendig oder ausreichend sind. Eine ähnlich umfangreiche Behandlung der Technologie AR erfolgt beispielsweise durch BROLL [Bro19] und durch SCHMALSTIEG UND HOLLERER [SH16]. Des Weiteren existiert eine Vielzahl an Publikationen und technischen Dokumenten, die einzelne Komponenten im Detail behandeln. Verschiedene Varianten von See-Through-Displays werden inklusive der technischen Herausforderungen sowie der Vor- und Nachteile von LEE UND WU beschrieben [LZW19]. GRIMM ET AL. beschäftigen sich gezielt mit verschiedenen Eingabe-Geräten und Tracking-Ansätzen [GBH+19]. ALT unterscheidet bei Tracking zwischen inertialen, magnetischen, akustischen, optischen und elektromagnetischen Ansätzen [Alt02]. CHATZOPOULOS ET AL. analysieren die Ansätze ausführlich bezüglich der benötigten Sensorik sowie ihrer Genauigkeit und möglicher Einsatzszenarien [CBH+17]. Verschiedene Marker-lose und sequentielle optische Trackingansätze werden von WEINMANN im Detail beschrieben und verglichen [Wei11]. Technische Details zu Sensorik werden in verschiedenen spezifischen Publikationen behandelt, beispielsweise zu Timo-of-Flight Kameras, die einen 3D-Scan der Umgebung liefern und in vielen AR-Geräten

verbaut sind [Li14]. Bei der Interaktion als wichtigem Bestandteil von AR wird beispielsweise zwischen Gesten-, Sprach-, Touchscreen-, Touchpad-, Button- und Controller-Steuerung unterschieden [BCL15]. Zudem sind für viele AR-Einsatzszenarien Aspekte wie die Netzwerkverbindung, bspw. mittels LAN oder WLAN, oder Kommunikation über Bluetooth oder NFC relevant [PSM08]. Neue auf der Netzwerk-Verbindung aufbauende Datenverarbeitungs-Ansätze in Richtung Cloud-Computing werden beispielsweise von NAQVI ET AL. in [NMR+15] beschrieben.

Bewertung: Für die AR-System-Konfiguration relevantes Wissen über AR-Systeme, deren Struktur sowie technischen und funktionalen Komponenten liegt bislang in einer Vielzahl unterschiedlicher und größtenteils unstrukturierter Publikationen vor. Aus dieser Vielzahl an Quellen muss relevantes Wissen in einem geeigneten Detailgrad gesammelt und aufbereitet werden. Eine Analyse aller verfügbarer Quellen wird nicht möglich sein. Ein Ansatz zur Erarbeitung des Wissens wäre daher, mit den übergreifenden Arbeiten zu starten und die weiteren spezifischen Quellen zu nutzen, um notwendiges detailliertes Wissen zu einzelnen Aspekten gezielt zu ergänzen. Bei einem geeigneten Konzept kann Wissen zudem nachträglich und kontinuierlich entsprechend der an die Arbeit gestellten Anforderung A5 erweitert und ergänzt werden (s. Abschnitt 3.6).

A2.2.5 AR-Endgeräte

AR-Hardware ist schon lange Zeit Teil der Forschung. FELLMANN ET AL. beschreiben eine Auswahl früher AR-Hardware, die im Rahmen von Forschungsprojekten entstanden sind [FHM+15]. Dazu zählt unter anderem auch das System MARS, das bereits ab 1996 an der Columbia University entwickelt wurde. MARS steht für Mobile Augmented Reality System [Col12-ol]. Es handelt sich um mobile, aber große AR-Hardware, die in einem Rucksack getragen wird und Visualisierungen auf dem Campus der Universität ermöglicht. Ähnlich wie MARS waren AR-Systeme zunächst groß und schwer und nicht alltagstauglich.

Mittlerweile gibt es eine Vielzahl unterschiedlicher AR-Geräte am Markt. Erste praktikable AR-fähige Geräte waren Dank Software Development Kits (SDK) von der metaio GmbH [met11-ol] oder PTC (Vuforia) [PTC22a-ol] **Smartphones und Tablets**. Durch eine stetige Weiterentwicklung der Hardware und auch der Software sind Smartphones und Tablets weiterhin beliebte AR-Endgeräte. Heutzutage besitzt fast jeder ein Smartphone. Somit ist die Benutzung der Geräte aus dem Alltag bekannt und die Akzeptanz entsprechend hoch. Apple bietet mit dem *iPhone* und dem *iPad* dank der Integration von LIDAR-Sensoren die innovativsten handgehaltenen AR-Geräte [App22b-ol]. 2012 startete Google mit der *Google Glass* als erstes Unternehmen den Versuch, eine **kopfgetragene Datenbrille** am Markt zu etablieren [Goo21-ol]. Bei *Google Glass* handelt es sich um eine monokulare Brille mit einem Display (Abbildung A-16, links). Über das Display wird im Sichtfeld eines Auges ein virtuelles bzw. digitales Bild dargestellt. Google war

zunächst nicht sehr erfolgreich, da die Technik neu und die Akzeptanz bei den potenziellen Benutzern eher gering war. Mittlerweile gibt es eine sehr große Menge weiterer monokularer Brillen am Markt, die sich im Design, in ihren Spezifikationen sowie in ihren Einsatzzwecken unterscheiden. Hersteller solcher Brillen mit industriellem Fokus sind beispielsweise Vuzix [vuz22-ol], realwear [Rea21-ol] oder auch Epson [Eps22-ol]. Zudem erhalten Datenbrillen verstärkt Einzug im Alltag. So arbeiten Unternehmen wie Meta (ehemals Facebook), Huawei und auch Xiaomi [Wol21-ol] an alltagstauglichen Brillen für den privaten Gebrauch. Neben den monokularen Datenbrillen sind vor allem die **binokularen AR-Brillen** von besonderer Bedeutung im Kontext der Validierung. Diese Brillen ermöglichen durch die Visualisierung über zwei Displays, eins für jedes Auge, eine dreidimensionale Darstellung. Als erstes Unternehmen brachte Microsoft 2016 die *HoloLens* als binokulare AR-Brille an den Markt [Mic22e-ol]. Mittlerweile gibt es bereits seit 2019 den Nachfolger *HoloLens 2* (Abbildung A-16, rechts). Auch im Bereich binokularer Brillen gibt es mittlerweile eine große Auswahl. Geräte wie die *magic leap One* [mag22-ol] oder die *ThinkReality A3* von Lenovo [Len22-ol] stellen Alternativen zur *HoloLens* dar. Zudem kommen immer mehr Anbieter mit binokularen AR-Brillen auch für den privaten Nutzer an den Markt. Ein Beispiel ist die Firma nreal mit der AR-Brille *Light* [nre22-ol].



Abbildung A-16: Beispiele verfügbarer AR-Geräte: Google Glass Enterprise Edition 2 (links) [Goo21-ol], Microsoft HoloLens 2 (rechts) [Mic22e-ol]

Jedes einzelne AR-Gerät hat individuelle Eigenschaften und Attribute. Grundlegende Infos sind jeweils der **technischen Spezifikation** zu entnehmen. Diese stellen eine wichtige Wissensquelle für die vorliegende Arbeit dar. Die Spezifikation der HoloLens 2 beispielsweise gibt ausführlich Auskunft über das Display, die verbaute Sensorik, Audio-Input- und -Output-Möglichkeiten, die Konnektivität, Interaktionskonzepte, das Gewicht sowie die Stromversorgung [Mic22a-ol]. Weitere wichtige Aspekte sind zudem die Information, dass eine Korrektur-Brille unter der AR-Brille getragen werden kann, und selbstverständlich der Preis des Geräts. Entsprechende Spezifikationen stehen bei allen Anbietern Online zur Verfügung. Zudem hat RÖLTGEN im Rahmen seiner Dissertation Steckbriefe zu einer Auswahl an AR-Geräten erstellt, in denen die Spezifikation der Geräte einheitlich aufbereitet ist (Abbildung A-17, vgl. Abschnitt 4.2.2.2) [Röl21].

Bewertung: Aufgrund der stetigen Weiterentwicklung der Geräte sowie der Veröffentlichung neuer Geräte zu den verschiedenen Varianten von AR wird die Auswahl und die

Varianz an AR-Geräten immer größer und die AR-System-Konfiguration immer komplexer. Die aktuell verfügbaren Informationen stellen nur eine Momentaufnahme dar. Im Rahmen dieser Arbeit können daher zunächst nur die aktuell relevanten Geräte analysiert und in die wissensbasierte AR-System-Konfiguration integriert werden. Die jeweiligen Informationen zu AR-Geräten sind frei im Internet bei den Herstellern verfügbar und können entsprechend verwendet werden.

Epson Moverio Pro BT-2200 1.999 US-\$			
Magic Leap One 2.295 US-\$			
Microsoft Hololens 3.000 US-\$			
	<i>Am Markt verfügbar</i>		
			
Datenbrillenspezifikation			
Ausgabe	Stereoskopisch, durchsichtig 1268 x 720 Pixel (pro Auge) 34° Sichtfeld (diagonal) 3D-Lautsprecher		
Sensoren	Inertiale Messeinheit Vier Lichtkameras (Kopftracking) Time-of-Flight-Tiefensensor 2,4-MP-Foto-/Videokamera		
Leistung			
Anforderung	Ausprägungen		
Auflösung	<720p	720p-1080p	>1080p
Sichtfeld	<30°	30-60°	>60°
Farb-darstellung	Monochrom	Bichrom	Polychrom
Sprach-steuerung	Keine	Vordefin. Kommandos	Sem. Sprach-erkennung
Gesten-steuerung	Keine	Vordefin. Gesten	Hand-/Fin-gertracking
Bluetooth	Kein Bluetooth erforderlich	Bluetooth 4.2 oder älter	Bluetooth 5 oder neuer
WiFi	Kein WiFi erforderlich	IEEE 802.11n oder älter	IEEE 802.11 ac oder neuer
Kamera-Auflösung	Keine	<720p	720p oder höher
	Nicht	Mono/	Spatial Sound

Abbildung A-17: Spezifikation von AR-Endgeräten anhand von Steckbriefen [RöL21]

A2.2.6 AR-Anwendungs- und Erfahrungswissen

Zu dem für die AR-System-Konfiguration relevanten Wissen gehört Anwendungs- und Erfahrungswissen. Dieses Wissen umfasst Erkenntnisse und Erfahrungen, die von AR-Experten und einzelnen Unternehmen durch die Anwendung der Technologie AR gemacht wurden. Da viele Unternehmen nicht über entsprechendes Wissen verfügen, muss es gesammelt und zusammen mit den weiteren Wissensaspekten in der Wissensbasis bereitgestellt werden.

Anwendungs- und Erfahrungswissen liegt vor allem in Form von Erfahrungsberichten und wissenschaftlichen Publikationen zu konkreten **AR-Anwendungsfällen** vor. ELIA ET AL. beschreiben beispielsweise den Einsatz von AR in der Produktion [EGL16]. SYBERFELDT ET AL. befassen sich mit AR zur Montage-Unterstützung [SDH+15]. Ein Beispiel zur Remote Maintenance liefern MOURTZIS ET AL. [MSA20]. Zudem beschreiben viele generelle Publikationen zu AR beispielhafte Anwendungsfälle, unter anderem [SCB17], [Bro19] und [SH16]. Über die genannten Quellen hinaus existiert eine Vielzahl an Erfahrungsberichten und Publikationen, die spezifische Anwendungsfälle und entsprechende Erfahrungen zu den eingesetzten AR-Systemen und Rahmenbedingungen beschreiben.

Entsprechende Beschreibungen in den spezifischen und auch übergreifenden Publikationen enthalten meist auch Informationen zur Anwendungssoftware und darin inbegriffenen Software-Funktionalitäten.

Weitere Informationen können Studien und Marktreports entnommen werden. Eine Studie der IDG RESEARCH SERVICES beschreibt unter anderem die Nutzung verschiedener AR-Funktionen [IDG19]. Der Buyer's Guide von PTC stellt verschiedene Software-Lösungen und -funktionalitäten, deren Differenzierungsmerkmale sowie mögliche Einsatzbereiche vor [PTC19]. Erfahrungen mit AR im Kontext der Fertigung werden beispielsweise von der ARC Advisory Group beschrieben [ARC18]. Aus entsprechenden Quellen lässt sich entsprechend für die AR-System-konfiguration relevantes Wissen ableiten.

Bewertung: Erfahrungs- und Anwendungswissen liegt wie auch ein Großteil der weiteren Wissensaspekte in einer Vielzahl unterschiedlicher Quellen vor und ist größtenteils unstrukturiert. Es lässt sich jedoch für die AR-System-Konfiguration wertvolles Wissen aus diesen Quellen ableiten.

A2.2.7 Einflussfaktoren auf AR-Systeme aus der Einsatzumgebung

Bei der Konfiguration des AR-Systems müssen stets auch externe Einflussfaktoren aus der Einsatzumgebung berücksichtigt werden. Es ist sicherzustellen, dass das AR-System inklusive aller funktionaler und technischer Komponenten in der geplanten Einsatzumgebung funktioniert. Es existiert eine Vielzahl an Quellen, in denen einzelne Einflussfaktoren genauer analysiert werden. Dazu zählen sowohl wissenschaftliche Publikationen als auch praktische Erfahrungsberichte. Unter anderem wurden **Einflussfaktoren auf AR** von SCHILLING im Rahmen seiner Dissertation untersucht (s. Abschnitt 4.2.2.5).

Ein sehr wichtiger Aspekt, den es zu berücksichtigen gilt, sind **Umwelteinflüsse**. Die *Lichtverhältnisse* können beispielsweise Einfluss auf optische Tracking-Ansätze haben [Cra13]. Schatten können zu Irritationen führen und eine sich ändernde Beleuchtung für die Lokalisierung verwendete Merkmale in der Umgebung optisch verändern und somit zu Problemen beim Tracking führen [SAM09]. Eine helle Beleuchtung, beispielsweise durch Sonneneinstrahlung, ist vor allem bei der Wahl der Display-Technologie und der Darstellungsart zu berücksichtigen. Über Optical See-Through-Displays beispielsweise sind Inhalte in hellen Umgebungen selbst bei hohem Kontrast meist nicht erkennbar [LZW19]. Einen Vergleich verschiedener Display-Technologien liefert unter anderem BROLL (Tabelle A-9-1). Die Darstellung von hellen Inhalten auf hellem Hintergrund ist mit Optical See-Through-geräten nicht möglich, da dunkle Inhalte fast vollständig transparent sind auf entsprechenden Displays [Bro19]. Neben den Displays hat starkes Sonnenlicht zudem Einfluss auf Infrarot-basierte Time-of-Flight-Kameras [Li14, KPR05]. Umwelteinflüsse wie die *Lufttemperatur*, der *Luftdruck* oder die *Luftfeuchtigkeit* sind ebenfalls zu berücksichtigen. Sie haben beispielsweise Einfluss auf spezielle Tracking-Ansätze wie dem Ultraschall-Tracking [Teg06]. Zudem sind diese Aspekte in Bezug auf Schutzklassen der AR-Geräte zu berücksichtigen. Die über den Ingress Protection (IP)

Code definierten Schutzklassen beschreiben den Gehäuseschutz von elektrischen Geräten bezüglich *Fremdkörpern* und Wasser bzw. *Feuchtigkeit* [IEC60529:2013].

Tabelle A-9-1: Darstellung verschiedener virtueller Inhalte auf unterschiedlichen Anzeige-Systemen abhängig vom Hintergrund nach [Bro19]

Darstellung heller virt. Inhalte auf hellem Hintergrund	... dunklem Hintergrund
Optical See-Through	Bedingt, hohe Transparenz	Gut, geringe Transparenz
Video See-Through	Gut	Gut
Projektion	Bedingt	Gut
Darstellung dunkler virt. Inhalte auf hellem Hintergrund	... dunklem Hintergrund
Optical See-Through	Nicht möglich, fast vollständige Transparenz	Bedingt, hohe Transparenz
Video See-Through	Gut	Gut
Projektion	Nicht möglich	Bedingt
Darstellung virtueller Schatten auf hellem Hintergrund	... dunklem Hintergrund
Optical See-Through	Nicht möglich	Möglich
Video See-Through	Möglich	Möglich
Projektion	Nicht möglich	Nicht möglich

Fremdkörper wie Staub und Schmutz sind Beispiele für **Störquellen** in der Einsatzumgebung. Sie können die Zuverlässigkeit von Tracking-Ansätzen, beispielsweise eines Marker-Trackings beeinflussen [ME19]. Weitere Störquellen, die speziell beim Einsatz elektromagnetischer Tracking-Ansätze zu berücksichtigen sind, sind *metallische und magnetische Gegenstände* und mögliche *elektromagnetische Felder* [Sch19]. Umgekehrt können AR-Geräte durch elektromagnetische Strahlung und Interferenzen negativen Einfluss auf andere Systeme in der Einsatzumgebung haben. Dies ist beispielsweise in Krankenhäusern von besonderer Bedeutung [Cra13]. Projektions-basierte Systeme können unter anderem durch *Oberflächentexturen* beeinflusst werden. Speziell reflektierende Oberflächen beeinträchtigen zudem optische Tracking-Verfahren [HGB01]. Hohe *Mauern* und *Häuserschluchten* schränken das Tracking per GPS stark ein [Str08]. Grundsätzlich sind Tracking-Ansätze oftmals mit dem „Line-of-sight“-Problem konfrontiert. Tracking funktioniert in diesen Fällen nur, wenn eine direkte Sichtverbindung zwischen Transmitter und Receiver besteht [Sch08a]. Dies betrifft vor allem optische Tracking-Ansätze. Weitere Quellen beschreiben Störquellen für AR-Systeme bzw. deren Komponenten auf beliebig detailliertem Niveau. TEGTMEIER analysiert beispielsweise den Einfluss *hochfrequenter Geräuschquellen* oder Echos auf Ultraschall-basiertes Tracking [Teg06]. LEIBSON befasst sich sehr genau mit Inertial Measurement Units (IMU), die in fast jedem AR-Gerät verbaut sind, und geht dabei auf Einflüsse durch *Hart- und Weicheisen* ein [Lei19-ol].

Eine besondere Berücksichtigung sollte zudem **Einsatzbereich-spezifischen Faktoren** gewidmet werden. Dazu zählt die vorhandene IT-Infrastruktur wie beispielsweise das *Netzwerk* oder die *Stromversorgung*, die Einfluss auf die Geräte-Auswahl haben können [PSM08]. Die *Umgebungslautstärke* beeinflusst unter anderem die Wiedergabe von Audio und die Wahl der Audio-Output-Komponente. So verbietet der Einsatz von AR in *Ruheräumen* die Wiedergabe von Ton und die Interaktion über Sprach-Steuerung [Cra13]. Der *Einsatzort* ist auch für die Wahl des Trackings relevant. Marker-Tracking benötigt eine präparierte Einsatzumgebung [KP10]. Je nach Einsatzort ist die Anbringung oder Platzierung eines Markers jedoch nicht möglich. Ebenso ist GPS-Tracking in Innenräumen schwierig [Sch08a]. Bzgl. des Einsatzorts ist auch die Transportierbarkeit des AR-Geräts zu berücksichtigen. FELLMANN ET AL. unterscheiden hier zwischen *sehr transportierfähig, eingeschränkt transportierbar* und *stationär einsetzbar* [FHM+15]. Wird AR in einer *Werkstatt* oder Produktionshalle eingesetzt, ist davon auszugehen, dass das AR-Gerät mit Schmutz und Staub in Berührung kommt. Daher sind entsprechende Schutzklassen zu berücksichtigen. Zudem sind Arbeitsschutzrichtlinien wie eine Helmfplicht oder bzgl. des Tragens einer Schutzbrille in die AR-System-Konfiguration einzubeziehen. Speziell bei der Durchführung händischer Tätigkeiten, beispielsweise zur Validierung von Interaktionskonzepten, ist der Grad der Handlichkeit von AR-Systemen ausschlaggebend [EGL16]. CARICATO ET AL. unterscheiden hier zwischen *nur eine Hand frei, beide Hände frei, eine Hand frei mit einer Einschränkung* und *beide Hände frei mit Einschränkung* [CCG+14]. Weitere Limitationen und Faktoren bezüglich des Nutzers werden von VAN KREVELEN UND POELMAN untersucht [KP10]. Relevant ist unter anderem, ob der Nutzer eine Brille trägt oder körperliche Einschränkungen hat. Körperliche Einschränkungen können vor allem die Interaktion beeinflussen. Abhängig von der *Einsatzdauer* hat auch der *Tragekomfort* eines AR-Geräts, speziell von AR-Brillen, eine besondere Bedeutung für den Nutzer [ARF+19].

Bewertung: Es existiert eine Vielzahl an Quellen, die mögliche Einflussfaktoren aus der Einsatzumgebung beschreiben. Einzelne Quellen adressieren speziell auf AR-Systeme bezogene Einflüsse. Andere dagegen beschäftigen sich mit einzelnen technischen Komponenten wie Sensoren oder Tracking-Ansätzen. Diese Quellen gehen teilweise technisch sehr tief ins Detail. Eine umfangreiche Nutzung der Quellen ist sinnvoll, um das für die AR-System-Konfiguration relevante Wissen möglichst vollständig abzubilden. Dabei ist jedoch ein geeignetes Detail-Level zu finden, um die AR-System-Konfiguration nicht unnötig kompliziert zu gestalten und die Anwendbarkeit zu gewährleisten. Eine gute Aufbereitung einer Auswahl an zu berücksichtigenden Aspekten wird bereits durch RÖLTGEN bereitgestellt. Es handelt sich dabei jedoch um eine informative Beschreibung relevanter Aspekte. Für die vorliegende Arbeit ist eine konkretere Definition der Einflüsse notwendig, um das Wissen später teilautomatisiert auswerten zu können. Der Leitfragenkatalog dient daher lediglich als eine Quelle bei der Erarbeitung und Formalisierung des Wissens.

A2.2.8 Das Ontologie-Modellierungswerkzeug Protégé

Bei Protégé handelt es sich um einen **Open-Source Ontologie-Editor**, der am *Stanford Center for Biomedical Informatics Research* entwickelt wurde [Sta22-01]. Erste Entwicklungsaktivitäten wurden bereits 1987 von Mark Musen im Rahmen eines Forschungsprojekts gestartet [BCT07]. Seitdem wird der Editor ständig weiterentwickelt und optimiert. Protégé gilt als das am meisten genutzte Werkzeug zur Erstellung und Verwaltung von Ontologien und wird sowohl in der Wissenschaft als auch von Unternehmen eingesetzt [Mus15]. Dank einer konfigurierbaren grafischen Benutzeroberfläche (Abbildung A-18) ist Protégé sowohl für Anfänger als auch Experten geeignet. Für den Import- und Export von Ontologien werden alle gängigen Daten-Formate unterstützt. Der Editor wurde basierend auf Java entwickelt und bietet die Möglichkeit zur Integration von Plugins [O'C09]. Verschiedene frei verfügbare Plugins werden vom Protégé-Entwicklerteam sowie weiteren Entwicklern aus der ganzen Welt bereitgestellt. Diese Plugins ermöglichen beispielsweise alternative Visualisierungsmöglichkeiten oder das Management und die Zusammenführung mehrerer Ontologien und stellen Problemlösungskomponenten und Reasoner bereit [NCF+03].

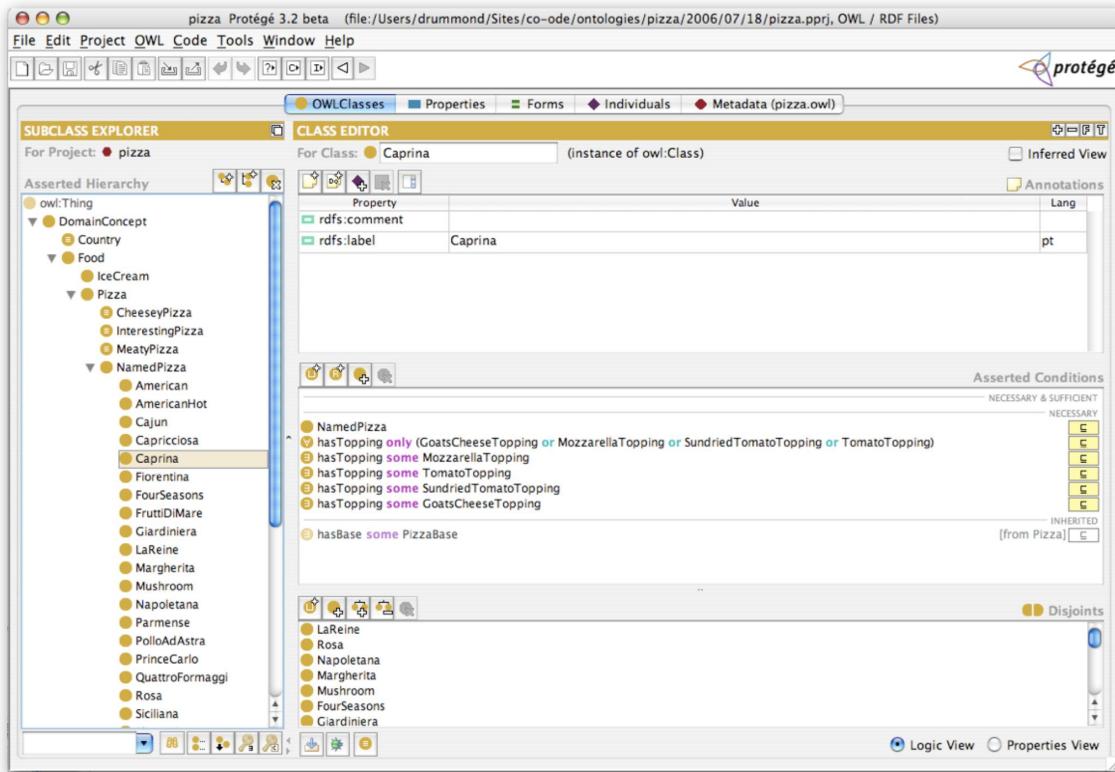


Abbildung A-18: Grafische Benutzeroberfläche von Protégé [DHD+07]

Über das **Plugin Cellfie** wird das automatisierte Einlesen von Informationen aus Kalkulationstabellen ermöglicht [HO18-01]. Cellfie basiert auf der von O'CONNOR entwickelten Tabellen-Mapping-Technik *M² Language* und setzt Transformationsregeln zur Überführung der Informationen in eine OWL-Ontologie softwaretechnisch um. Cellfie bietet eine

grafische Benutzeroberfläche, über die die Transformationsregeln angelegt und Axiome automatisiert in einer Ontologie angelegt werden können (Abbildung A-19). Die genaue Vorgehensweise zum Einsatz von Cellfie wird von HARDI in einem Online-Tutorial beschrieben [Har17-ol]. Ein praktisches Beispiel zum Einsatz von Protégé und Cellfie wird von KHAN ET AL. in [KSS19] gegeben.

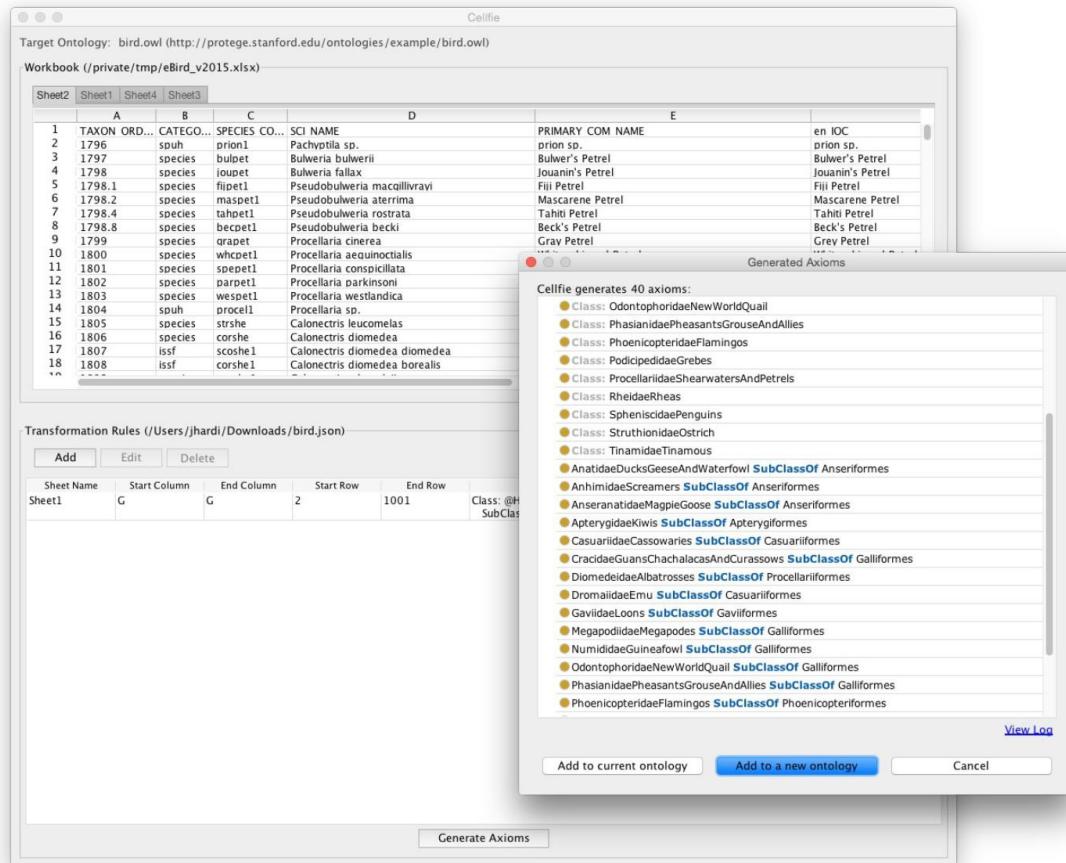


Abbildung A-19: Grafische Benutzeroberfläche des Cellfie-Plugins für Protégé zum Überführen von Informationen aus Kalkulationstabellen in OWL-Ontologien [HO18-ol]

Bewertung: Protégé bietet eine grafische Benutzeroberfläche zur Arbeit mit OWL-Ontologien. Das Plugin Cellfie ermöglicht zudem eine aufwandsarme Überführung von Informationen aus Kalkulationstabellen in eine OWL-Ontologie. Dazu müssen lediglich einmalig Transformationsregeln definiert werden. Entsprechend stellt Protégé eine sehr gute Lösung zur softwaretechnischen Realisierung der in Abschnitt 4.2.1.2 beschriebenen Überführung von Wissen aus Excel-Tabellen in eine OWL-Ontologie dar.

A2.3 Aufwandsarme Entwicklung von AR-Validierungssoftware

Im Folgenden werden ergänzend zum Stand der Technik Informationen und Ansätze beschrieben, die für die aufwandsarme Entwicklung von AR-Anwendungssoftware relevant sein können.

A2.3.1 Entwicklungsumgebung Unity

Unity ist eine **Laufzeit- und Entwicklungsumgebung für 3D-Anwendungen** des Unternehmens Unity Technologies, das 2004 in Dänemark gegründet wurde [Haa14]. Unity bietet eine grafische Benutzeroberfläche zur intuitiven Entwicklung. Anwendungsgebiete sind unter anderem die Spieleentwicklung, industrielle 3D-Visualisierungen sowie die Entwicklung von virtuellen Erlebnissen mittels AR und VR [Uni22q-ol]. Dabei unterstützt Unity mehr als 20 unterschiedliche Plattformen, darunter iOS, Android, Windows, PS4, Linux und WebGL [Uni22d-ol]. Viele Funktionalitäten werden über Drag-and-Drop und grafische Schaltflächen ermöglicht. Entsprechend reduziert Unity den Programmieraufwände für Entwickler. Umfangreiche Beschreibungen von Unity und der Unity-basierten AR-Anwendungsentwicklung finden sich in [LB17] und [GL19].

Das wichtigste Konzept der Entwicklung von 3D-Anwendungen in Unity ist das **GameObject** [Uni22e-ol]. Jedes Objekt in einer 3D-Szene ist ein *GameObject*, angefangen von 3D-Modellen über Lichter, Kameras, Interaktionselementen bis hin zu Spezialeffekten. *GameObjects* werden in einem hierarchischen Szenengraphen organisiert und logisch sowie funktional in Verbindung gebracht. Szenengraphen stellen ein gängiges Konzept von 3D-Anwendungen dar [Web22-ol]. Über Szenengraphen können Eltern-Kind-Beziehungen von Elementen bzw. im Fall von Unity von *GameObjects* definiert werden. Beispielsweise können Transformationen somit hierarchisch aufgebaut und eine Menge an *GameObjects* unter einem Eltern-Knoten zu einer Gruppe kombiniert werden. *GameObjects* können in Unity mit Komponenten versehen werden. Über Komponenten lassen sich Funktionalitäten und Eigenschaften von *GameObjects* definieren und konfigurieren. Zum einen stellt Unity eine große Auswahl an vordefinierten Komponenten bereit, unter anderem zu physikalischen Aspekten, Animationen, Künstlicher Intelligenz und Benutzeroberflächen [Uni22f-ol]. Individuelle Komponenten können zudem durch den Entwickler in Form von Software-Skripten selbst programmiert werden. Als Programmiersprachen unterstützt Unity C#, JavaScript und Boo [Haa14]. *GameObjects* inklusive ihrer Komponenten können als wiederverwendbare Bausteine in Form von **Prefabs** in einem Projekt gespeichert werden [Uni22a-ol]. Zudem können ganze Szenen als **Projekt-Templates** gespeichert werden. Ein Template enthält beispielsweise grundlegende Funktionalitäten und Objekte für ein Anwendungsszenario. Beim Anlegen einer neuen Szene für das Anwendungsszenario kann das Template als Basis verwendet und mit individuellen Inhalten versehen werden.

Alle in einem Projekt verwendeten Daten werden als **Assets** bezeichnet. Assets umfassen visuelle, auditive und funktionale Daten. Sie können zudem abstrakte Elemente wie Animationsmasken, Text oder numerische Daten beinhalten [Uni22c-ol]. Zum einen können Assets innerhalb von Unity erstellt werden. Das gilt beispielsweise für primitive Geometrien oder Animationen. Zum anderen können externe Dateien als Assets importiert werden. Dies wird beispielsweise bei 3D-Modellen, Audio-Dateien oder Grafiken genutzt. Zudem können seit November 2010 Dritt-Anbieter-Assets über den **Unity Asset Store** erworben und direkt in einem Projekt verwendet werden [Haa14]. Entwickler können über den Store selbst entwickelte Assets mit speziellen Funktionalitäten offiziell für andere Entwickler kostenlos oder gegen einen Kaufpreis zur Verfügung stellen. Unity importiert Assets automatisch und führt bei Bedarf im Hintergrund eine automatisierte Konvertierung in andere Formate durch. Diese Konvertierung ist unter anderem notwendig, damit eine Anwendung später auf einer spezifischen Plattform ausgeführt werden kann. Die sogenannte **Asset-Pipeline** wird fortlaufend weiterentwickelt und optimiert, um den Import-Prozess auch bei umfangreichen Projekten möglichst effizient und schnell zu ermöglichen [Hou21-ol].

Abbildung A-20 zeigt die grafische Benutzeroberfläche von Unity. Neben der Werkzeugleiste (A) und der Statusleiste (H) beinhaltet diese die folgenden wesentlichen Fenster (vgl. Buchstaben in Abbildung A-20):



Abbildung A-20: Benutzeroberfläche von Unity [Uni22b-ol]

B) Hierarchy Window: Das Hierarchy-Fenster repräsentiert den Szenengraphen einer Unity-Szene in Textform. Es werden alle in der Szene enthaltenen *GameObjects* entsprechend ihrer Hierarchie gelistet.

C) Game View / Simulation: Das Game-View-Fenster stellt die Anwendung so dar, wie sie aus Sicht der virtuellen Kameras bei Ausführung aussehen und funktionieren wird. Es handelt sich entsprechend um eine Art Simulation der Anwendung. Sie kann über einen

Play-Button gestartet werden. Bei AR-Anwendungen funktioniert die Simulation einer Anwendung nur bedingt im Editor, da beispielsweise spezielle Sensorik nur auf dem AR-Endgerät verfügbar ist.

D) Scene View: Im Scene View wird die virtuelle 3D-Szene zur Bearbeitung dargestellt. Der Entwickler hat freie Kontrolle über die Perspektive unabhängig der virtuellen Kameras für die Ausführung. Im Scene View können Konfigurationen und Transformationen direkt an den visuell dargestellten *GameObjects* vorgenommen werden.

E) Overlays: Über Overlays können zusätzliche Werkzeugeisten genutzt werden. Es können auch eigene Werkzeuge entwickelt und bereitgestellt werden.

F) Inspector: Im Inspector werden die Eigenschaften und Komponenten des aktuell ausgewählten *GameObjects* dargestellt. Die Auswahl erfolgt über das Hierarchie-Fenster oder den Scene View. Im Inspector können Konfigurationen vorgenommen und Komponenten hinzugefügt und bearbeitet werden.

G) Project Window: Im Projekt-Fenster werden alle im Projekt enthaltenen Assets in der entsprechenden Ordner-Struktur dargestellt. Neu hinzugefügte Assets tauchen hier auf und können in der Szene per Drag-and-Drop verwendet werden.

Unity bietet zudem die Möglichkeit der **Individualisierung der Benutzeroberfläche**. Zusätzlich zu den Standard-Fenstern können über *ScriptableWizards* eigene Fenster erstellt und darüber selbst entwickelte Funktionalitäten für den Entwickler bereitgestellt werden [Uni22p-ol]. Zudem können Editor-Funktionalitäten bedarfsgerecht über Kontext-Menüs (*ContextMenu*) ausgelöst werden [Uni22k-ol]. Eine hilfreiche Übersicht und Erklärung aller von Unity bereitgestellter Funktionalitäten und Möglichkeiten liefert das **User Manual** [Uni22l-ol]. Weitere Möglichkeiten zur Anpassung und Vereinfachung der Benutzeroberfläche bieten Plugins aus dem Asset Store. Mit dem Plugin *Odin Inspector* kann beispielsweise die Konfigurationsoberfläche im Inspektor für Komponenten individuell strukturiert und interaktiv gestaltet werden [Odi22-ol].

Unity kann kostenlos heruntergeladen und installiert werden [Uni22m-ol]. Für den kommerziellen und professionellen Einsatz sind jedoch Lizenzgebühren zu bezahlen. Mit circa 1.800€ pro Jahr pro Lizenz sind die Lizenzkosten im Vergleich zu anderen industriellen Software-Werkzeugen jedoch eher gering.

Bewertung: Bei Unity handelt es sich um eine weit verbreitete Entwicklungsumgebung für AR-Anwendungen. Unity unterstützt alle gängigen Plattformen und bietet Entwicklern eine sehr gute grafische Benutzeroberfläche. Unity unterstützt alle wichtigen Software Development Kits (s. Abschnitt 4.3.1.2) und bietet über den Asset Store die Möglichkeit der Integration vielfältiger Plugins. So können beliebige Funktionalitäten und Visualisierungen, beispielsweise auch komplexe Animationen, in Unity realisiert werden. Über Prefabs können Funktionalitäten vorbereitet und modular bereitgestellt werden. Zudem ermöglichen Szenen-Templates die Vorbereitung und Wiederverwendung grundlegender Funktionalitäten und Konfigurationen für einzelne Anwendungsszenarien. Über

integrierte sowie Dritt-Anbieter-Funktionalitäten kann die Benutzeroberfläche individuell strukturiert, gestaltet und vereinfacht werden. All dies bietet grundsätzlich sehr gute Möglichkeiten, um die Entwicklung von AR-Anwendungen für das spezifische Anwendungsszenario Produktvalidierung zu vereinfachen und zu unterstützen.

A2.3.2 Datenvorbereitung und -integration

In AR-Anwendungen lassen sich grundsätzlich verschiedenste Arten von Daten und Informationen integrieren und darstellen. Am wichtigsten für 3D-Anwendungen sind **3D-Modelle**. 3D-Modelle können über CAD-Konstruktionssoftware oder 3D-Modellierungssoftware erstellt werden. In der Produktentwicklung wird vor allem CAD-Konstruktionssoftware eingesetzt. Bei CAD-Daten handelt es sich um parametrische Modelle, die Geometrien mathematisch präzise beschreiben. 3D-Modellierungswerkzeuge dagegen liefern visuelle Polygon-Modelle. Polygon-Modelle werden aus vielen kleinen Flächen, meist Dreiecken, dreidimensional zusammengesetzt. Im Kontext der Validierung können Animationen oder auch die Darstellung von Zusatzinformationen am 3D-Modell die Bewertung von Produktmerkmalen unterstützen. Die Darstellung von **Animationen und Zusatzinformationen** muss entsprechend vorbereitet werden. Zum einen kann dies sowohl in CAD- als auch Modellierungswerkzeugen erfolgen. Zum anderen können Animationen und Zusatzinformationen auch während der Anwendungsentwicklung direkt in Unity erstellt werden.

Grundsätzlich sind Datenformate bei der Integration von 3D-Modellen in Unity ein wichtiger Aspekt. Am Markt existiert eine Vielzahl an Software-Werkzeugen zur Erstellung von 3D-Modellen. Sowohl für CAD- als auch für Polygon-Modelle bestehen vielfältige Datenformate. Viele Software-Hersteller haben proprietäre Formate entwickelt. McHENRY UND BAJSCY beschreiben 16 beispielhafte Software-Werkzeuge und über 100 verschiedene 3D-Datenformate [MB08]. Unity kann die gängigsten der Polygon-Formate direkt importieren [Uni22j-ol]. Zu den **Standard-Formaten** zählen FBX, DAE und OBJ. FBX ist ein Objekt-basiertes Format, das sehr gut zu der in Unity verwendeten Objekt-basierten Szenenstruktur passt. Seit 2006 wird das Format von Autodesk® verwaltet. DAE ist das Format des COLLADA-Standards, der als Austausch-Format zwischen verschiedenen Modellierungs-Werkzeugen definiert wurde [Khr11-ol]. Sowohl FBX als auch DAE können neben den Geometrien weitere Szenen-Informationen speichern, darunter auch Animationen, Zusatzinformationen oder Lichteinstellungen. Bei OBJ handelt es sich um ein einfaches Datenformat, das lediglich die Geometrien repräsentiert [LCN+19]. Unity empfiehlt den Einsatz von FBX als Standard-Datenformat. Wenn möglich sollten Modelle vor der Nutzung im FBX-Format exportiert werden [Uni22j-ol]. Unity kann zudem auch **proprietäre Formate von 3D-Modellierungswerkzeugen** importieren. Dies gilt für die Werkzeuge Autodesk® Maya®, Autodesk® 3ds Max®, Modo®, Blender und SketchUp. Die Formate werden von Unity automatisiert in FBX konvertiert. Notwendig ist dazu, dass die Modellierungs-Werkzeuge auf dem System in-

stalliert sind. Unity öffnet diese im Hintergrund und kommuniziert mit ihnen zur Konvertierung der Formate. Die Integration von **CAD-Modellen** gestaltet sich komplizierter. Wie bereits in Abschnitt 3.4.3 beschrieben, stellt der Unterschied zwischen CAD- und Polygon-Modellen eine Herausforderung bei der Datenintegration dar. Der einfachste Weg zur Nutzung von Modellen, die mit CAD-Werkzeugen erstellt wurden, ist der Export im FBX-Format. Dieser wird von verschiedenen Werkzeugen ermöglicht. Ist ein FBX-Export nicht möglich, müssen die Modelle außerhalb des CAD-Tools konvertiert werden. Anhand einer sogenannten **Tesselierung** müssen parametrische Modelle in eine Polygon-Darstellung überführt werden [ERZ14, S. 178]. Dabei werden die mathematisch beschriebenen Geometrien über Polygone in einem gewünschten Detailgrad über Polygone beschrieben. Zum einen ist dies über 3D-Modellierungswerkzeuge wie bspw. dem kostenlosen Blender möglich [WZS+17]. CAD-Formate können in Blender importiert, bearbeitet und als FBX wieder exportiert werden. Der Bearbeitungsprozess kann je nach Modell jedoch aufwendig und zeitintensiv sein. Eine semi- oder komplett automatisierte Tesselierung wird mittels verschiedener **Konvertierungs-Werkzeuge** ermöglicht. Das bekannteste Beispiel ist die Software Pixyz [Uni22h-ol]. Pixyz wurde von Unity gekauft und ins Software-Portfolio integriert. Pixyz führt entsprechend einer vom Anwender festzulegenden Konfiguration eine automatisierte Tesselierung durch, entweder lokal oder auch Cloud-basiert. Die Nutzung von Pixyz ist jedoch mit zusätzlichen Lizenzkosten von bis zu 2.500\$ pro Jahr und Arbeitsplatz verbunden. Mit JT (Jupiter Tessellation) besteht ein ISO-standardisiertes Format, das als Container für verschiedene 3D-Formate verwendet werden kann [ISO14306:2017]. JT kann entsprechend parametrische CAD- und visuelle Polygon-Daten in einer Datei beinhalten [KHV13]. Zudem bietet JT die Möglichkeit, tesselierte Polygon-Modelle in verschiedenen Detailgraden zu speichern [ERZ14, S. 335]. So kann während einer Visualisierung beispielsweise beim Zoomen direkt das passend detaillierte Modell verwendet werden. JT wird von vielen Unternehmen im Rahmen von PLM-Werkzeugen (Product Lifecycle Management) verwendet. Die Nutzung von JT in Unity bedarf allerdings einer kostenpflichtigen Lizenz für das JT Open Toolkit von Siemens [Sie20-ol].

Bewertung: Die Integration von 3D-Modellen in eine AR-Anwendung ist wichtiger Aspekt speziell im Kontext der Produktvalidierung. Produkte werden während der Entwicklung meist über CAD-Daten repräsentiert. CAD-Formate können jedoch nicht ohne weiteres in Unity importiert werden. Unity empfiehlt daher den Export im FBX-Format. Dieser wird von vielen Software-Werkzeugen unterstützt. FBX-Modelle haben den Vorteil, dass sie zusätzliche Informationen wie Animationen speichern können. So lassen sich Animationen und weitere Visualisierungen direkt in den CAD-Tools erstellen und mit den Geometrien exportieren. Dies ist vorteilhaft für die Entwickler und Konstrukteure, die sich mit den CAD-Werkzeugen auskennen. Somit stellt FBX ein vielversprechendes Format zur Verwendung in Handlungsfeld 3 dar.

A2.3.3 Ergänzungen zu AR-SDKs

Dieser Abschnitt liefert ergänzende Details zu den in Abschnitt 4.3.1.2 beschriebenen AR-SDKs.

Im Folgenden werden zunächst weitere Details zu den Bausteinen des ***Mixed Reality Toolkit (MRTK)*** von Microsoft beschrieben. Das *MRTK* unterstützt verschiedene *Eingabesysteme* und -funktionalitäten. Große Teile der Interaktion basieren beim *MRTK* auf Hand- und Finger-Gesten. Unter anderem wird eine virtuelle Tastatur bereitgestellt (s. Abbildung A-21 a). Durch ein sehr präzises *Hand-Tracking* können flexibel eigene Funktionalitäten durch Entwickler realisiert werden. Auf der HoloLens 2 wird zudem *Eye-Tracking* über vordefinierte Bausteine ermöglicht. *UI-Steuerelemente* entsprechen aufwandsarm verwendbaren Prefabs und Komponenten zur Erstellung von Menüs und Interaktionselementen. So können beispielsweise im Raum vor dem Benutzer schwebende Menüs erstellt werden (Abbildung A-21 b). Über spezielle Visualisierungen der Fingerspitzen wird das Gefühl der Interaktion mit Elementen verstärkt (Abbildung A-21 c). Und zur einfacheren Interaktion mit 3D-Elementen können automatisiert Begrenzungssteuerelemente verwendet werden (Abbildung A-21 d). QuickInfo-Elemente ermöglichen die einfache Integration von Informations-Labels an 3D-Objekten (Abbildung A-21 e). Über das *Begrenzungssystem* kann der für AR-Anwendung nutzbare Bereich der realen Umgebung definiert und entsprechend visuell abgegrenzt werden. Dies kann beispielsweise bei Kollisionsgefahren eingesetzt werden. Über den *Multi-Szenen-Manager* lassen sich einfach AR-Anwendungen bestehend aus mehreren Szenen erstellen. Zudem unterstützt das *MRTK* verschiedene Ansätze der *räumlichen Wahrnehmung*. So kann bei entsprechender Sensorik eine Umgebungsrekonstruktion stattfinden. Diese ermöglicht eine bessere Kombination der realen Umgebung mit den virtuellen Inhalten. Abbildung A-21 f) zeigt eine exemplarische Visualisierung der Umgebungsrekonstruktion. Vorgefertigte *Shader* des *MRTK* stellen für AR-optimierte Darstellungsformen verschiedener Elemente bereit.

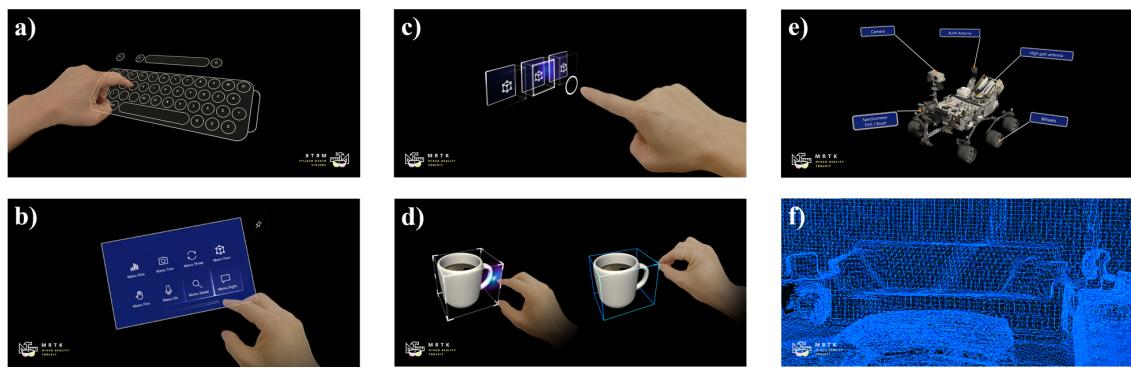


Abbildung A-21: Beispiele von Interaktionselementen und Funktionalitäten des *MRTK*
(Die Grafiken wurden aus [SCP+21-ol] verwendet)

In Ergänzung zu den zuvor erwähnten Interaktionsmöglichkeiten stellt das *MRTK* *Sprachsteuerung* und eine *Diktat-Funktion* bereit. So kann Sprache während der Anwendung direkt in Text umgewandelt und verwendet werden. Über *Solver* können zudem

spezielle Positionierungsalgorithmen verwendet werden. Beispielsweise ermöglichen *Solver* die räumliche Positionierung und Mitführung von Interaktionselementen um den Benutzer herum (vgl. Abbildung A-21 a-c). Im *MRTK* enthalten sind verschiedene Beispielszenen, die den Einsatz der verschiedenen Bausteine anschaulich und leicht verständlich zeigen. Microsoft ermöglicht zudem die Einbindung von *Azure Cloud-Services* in *MRTK*-basierte Anwendungen. Unter anderem können *Spatial Anchors* zur geräteübergreifenden Lokalisierung und Positionierung sowie Speech und Bildverarbeitungsdienste. In einer ausführlichen Dokumentation des *MRTK* sind alle Funktionalitäten und Möglichkeiten im Detail beschrieben und spezifiziert [Mic22c-ol].

AR Foundation unterstützt eine Reihe grundlegender AR-Funktionalitäten mit einheitlicher Programmierschnittstelle (API) auf verschiedenen Plattformen. Unter anderem ermöglicht *AR Foundation* eine einheitliche Programmierung des Trackings. *Device Tracking* wird auf allen Plattformen unterstützt. *2D-Image-Tracking* läuft mit *AR Foundation* nicht auf der HoloLens und *3D-Objekt-Tracking* ist nur über das ARKit auf iOS-Systemen möglich. Die Tabelle verdeutlicht, dass die unterstützten Funktionalitäten abhängig sind von den gerätespezifischen SDKs und es daher große Unterschiede für die Plattformen gibt. Für iOS-Geräte werden aufgrund des umfangreichen *ARKits* alle Funktionalitäten unterstützt. Für die HoloLens dagegen werden nur das *Device Tracking*, Raumanker (*Anchors*), Umgebungsrekonstruktion (*Meshing*) sowie das *Session-Management* zur Steuerung der Anwendung unterstützt. Alle weiteren Funktionalitäten sind über das *MRTK* zu realisieren.

Speziell im Kontext der kollaborativen Validierung sind **Netzwerkfunktionalitäten** wichtig. *Unity* hat mit *UNet* lange Zeit eine eigene Multiplayer- und Netzwerklösung bereitgestellt [Uni22i-ol]. Aktuell arbeitet *Unity* an der neuen Open-Source Lösung *MLAPI*. Auf *MLAPI* aufbauend wird unter anderem *Netcode* als Netzwerk-Bibliothek bereitgestellt. *Netcode* befindet sich jedoch noch in der Entwicklung [Uni22g-ol]. Abbildung A-22 zeigt eine Auswahl verfügbarer Netzwerk- und Kollaborations-SDKs, die im Rahmen einer Studie bewertet wurden. Neben der *MLAPI* stehen mit *DarkRift 2*, zwei Varianten des *Photon SDK* sowie *Mirror* weitere SDKs von Dritt-Anbietern zur Verfügung. Bei *DarkRift 2* handelt es sich um eine hoch performante und umfangreiche Lösung [Dar21-ol]. Das SDK unterstützt auf sehr technischem und somit Low-Level-Niveau. Dies bietet umfangreiche Möglichkeiten und hohe Flexibilität. Anwendungsnahe Netzwerk-Funktionalitäten müssen jedoch weiterhin vom Entwickler realisiert werden. *Mirror* ist ein weit verbreitetes kostenloses SDK, das umfangreiche Anwendungsfunktionalitäten auf Mid-Level-Niveau bietet [Mir22-ol]. Eine Schwachstelle liegt bei *Mirror* im Speichermanagement sowie im Mangel an High-Level-Funktionalitäten wie bspw. Kompressions- oder Kompensationsfunktionen. *Photon Quantum* ist speziell für Multiplayer-Spiele entwickelt worden. *Photon PUN* (*Photon Unity Networking*) dagegen kann flexibel für alle Arten von Anwendungen in *Unity* eingesetzt werden [Pho22-ol]. *Photon PUN* bietet alle relevanten Netzwerk- und Kollaborationsfunktionalitäten und kann bis zu einer gewissen Anzahl an Nutzern kostenlos verwendet werden. Ein Vorteil von *Photon PUN*

ist zudem, dass Microsoft ein spezielles Tutorial zum Einsatz des SDK in Mixed Reality-Anwendungen basierend auf dem *MRTK* bereitstellt [Mic22d-ol].

	Stability/ support	Ease-of- use	Perfor- mance	Scalability	Feature breadth	Cost*	Customers recommend for
MLAPI	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	Free	Most client-server games for up to ~64 players that want a stable breadth of mid-level features
DarkRift 2	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	\$100 for source	Games with high perf/scale requirements that want to build on a stable LL layer
Photon PUN	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	\$0.30/PCU	Simple and small (2–8 players) mesh-topology games
Photon Quantum 2.0	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	\$1000/mo + \$0.50/PCU	Games desiring deterministic roll-back, like MOBA games, for up to 32 players
Mirror	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★	Free	Stable and proven client-server solution, loved best for its community and ease-of-use

* Note that Photon pricing provides access to the networking libraries and services, whereas other solutions are standalone networking libraries, and the cost of services is separate.

Abbildung A-22: Bewertung einer Auswahl an Netzwerk- und Kollaborations-SDKs durch eine Umfrage mit mehr als 200 Unity-Nutzern [Hou20]

A2.3.4 Studie zu AR-basierter Produktvalidierung

VAN GOETHEM ET AL. haben eine Studie mit 12 Master-Studierenden durchgeführt, um herauszufinden, wie AR effizient zur Validierung von Produkten eingesetzt werden kann [vVG+20]. Als exemplarisches Produkt wurde unter anderem ein Akkuschrauber verwendet. Für den Akkuschrauber wurden bewusst sechs Konstruktionsfehler gemacht, die nicht den Anforderungen an das Produkt entsprechen (Abbildung A-23, 1-6): 1) Der Griff ist zu dünn, 2) Die Beule an der Unterseite ist unnötig, 3) Der Trigger-Knopf ist schwer zu erreichen, 4) Es fehlt ein Schalter, 5) Es fehlt ein Knopf zum Herauslösen des Akkus, 6) Ein Schieberegler ist nicht beweglich. Diese sechs Konstruktionsfehler werden über das 3D-Visualisierungsmodell entsprechend implizit repräsentiert. Als Validierungswerkzeug wurde eine **HoloLens-Anwendung** entwickelt. Das Modell des Akkubohrers wird den Testpersonen über diese Anwendung dreidimensional präsentiert (Abbildung A-23, 7). Durch den Einsatz der eigenen Hände durch Interaktion mit dem Modell können somit unter anderem auch ergonomische Aspekte überprüft werden. Zur Evaluierung der Tauglichkeit von AR zur Unterstützung der Validierung erfolgte die Validierung zusätzlich mittels einer Standard-Visualisierungslösung am Bildschirm. Als Ergebnis der Studie kam unter anderem heraus, dass die AR-basierte Validierung trotz der eher einfachen

Funktionalität eine bessere Validierung räumlicher sowie kontextueller Aspekte ermöglicht. Zudem hatten zehn von zwölf Studierenden mehr Spaß mit AR, was entsprechend förderlich ist für die Bereitschaft und die Motivation zur Validierung.

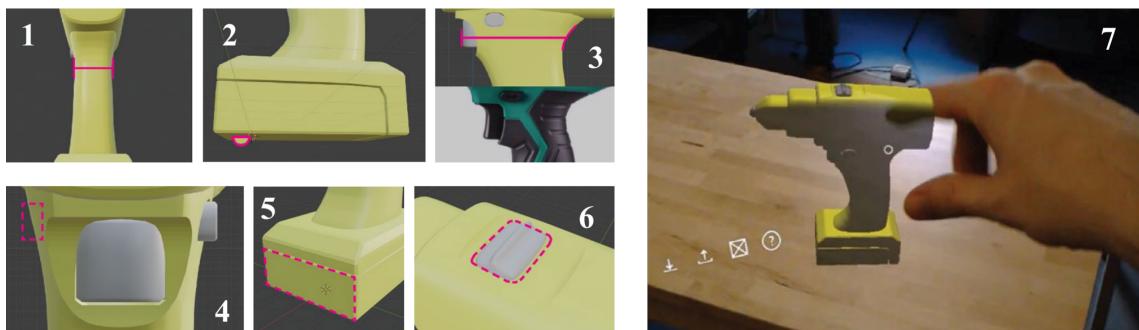


Abbildung A-23: Über einen AR-basierten Prototypen dargestellte Mängel an einem Akubohrer (1-6) und die Live-Ansicht über eine HoloLens (7)

Bewertung: VAN GOETHEM ET AL. beschreiben eine einfache HoloLens-AR-Anwendung zur immersiven Präsentation von virtuellen Prototypen. Die Anwendung bietet auch ohne komplexe Funktionalitäten bessere Validierungsmöglichkeiten bzgl. räumlicher und kontextueller Aspekte als klassische Visualisierungsansätze. Dies bestätigt den Nutzen der vorliegenden Arbeit. Zudem zeigen VAN GOETHEM ET AL., dass AR-Validierungssoftware nicht zwangsläufig komplexe Funktionalitäten bieten muss. Wichtig ist eine individuell auf das jeweilige Validierungsszenario zugeschnittene Funktionalität. Dieses Wissen kann bei der Erarbeitung der Lösung für Handlungsfeld 3 berücksichtigt werden.

A2.3.5 Usability-Untersuchungen von PAES UND IRIZARRY

Um AR sinnvoll als Validierungswerkzeug einzusetzen, ist eine gute Usability der AR-Anwendung notwendig. Usability wird im Deutschen als Gebrauchstauglichkeit bezeichnet. Laut der ISO9241 entspricht Usability dem Ausmaß, mit dem ein System oder ein Produkt vom Benutzer in einem gewissen Kontext genutzt und bedient werden kann [ISO9241:2006]. Eine hohe Usability ermöglicht eine intuitive, effiziente und zufriedenstellende Benutzung. Ein Vergleich der Usability zwischen einer klassischen CAD-Visualisierung mit Maus- und Tastatur-Steuerung und einer virtuellen Visualisierung mit Gesten- und Sprachsteuerung bestätigt die grundsätzlichen Vorteile AR-basierter Validierung [SB15]. Im Rahmen einer systematischen Usability-Studie haben PAES UND IRIZARRY zudem drei Aspekte einer VR-basierten Validierung genauer untersucht [PI18]: Die **kognitive Leistung** des Benutzers, die **Interaktionsleistung** des Benutzers, die **Eignung des eingesetzten virtuellen Systems** für die Validierungsaktivitäten. Mehrere professionelle Designer haben eine VR-Anwendung verwendet, um Architekturmodelle zu begutachten und zu überprüfen. Anschließend haben sie subjektive und objektive Fragen zur Darstellung, zur Benutzerschnittstelle sowie zur Gesamterfahrung der virtuellen Validierung beantwortet. Zum einen bestätigt die Auswertung der Studie, dass eine immersive Visuali-

sierungen die Vermittlung von Designlösungen und die Qualität der Kommunikation verbessern kann. Zum anderen zeigt die Studie Mängel bei der Gestaltung der Benutzeroberfläche auf. Folgende Aspekte wurden als verbesserungswürdig identifiziert:

- Beim **Teleportieren** in der virtuellen Welt eignet sich ein Fuß-Symbol besser als ein Kreis. Ein Fuß entspricht eher dem Verständnis von Bewegung.
- Eine Funktionalität zum **Wechsel zwischen verschiedenen Darstellungsformen** wäre hilfreich (beispielsweise Röntgen-Ansicht, Schwarz-Weiß-Darstellung, texturiertes Modell, realistische Darstellung etc.).
- Eine Option zum **Stoppen oder Pausieren der Anwendung** in einer Notsituation wird gewünscht.
- Eine **virtuelle Uhr**, die Laufzeit der Anwendung anzeigt, wäre hilfreich.
- **Animationen**, beispielsweise das Öffnen einer Tür oder sich bewegende Vorhänge, würden die Validierung realistischer machen.
- Eine **Multi-User-Funktionalität** zur kollaborativen Validierung wird gewünscht.
- Eine **natürlichsprachliche Benutzeroberfläche** wäre hilfreich, beispielsweise unterstützt durch Sprachassistenten wie Apple Siri oder Amazon Alexa.

Die identifizierten Aspekte bilden die Grundlage zur Optimierung und Weiterentwicklung der VR-Anwendung.



Abbildung A-24: VR-basierte Validierung von Architektur-Modellen [PI18]

Bewertung: Die genannten Arbeiten verdeutlichen die Wichtigkeit einer guten Usability von virtuellen Anwendungen. Die Technologien sind für viele Nutzer noch neu. Die Nutzer sind entsprechend unerfahren mit den innovativen Interaktionslösungen und Benutzerschnittstellen. Eine hohe Usability ist vor allem für eine effizient durchzuführende Validierung von enormer Bedeutung. Nutzer müssen in der Lage sein, sich auf die Validierung zu konzentrieren. Komplexe nicht intuitiv verständliche Benutzerschnittstellen sind zu vermeiden. Die von PAES UND IRIZARRY identifizierten Aspekte entsprechen wichtigen Hinweise, die bei der Entwicklung von AR-Anwendungen für die Validierung berücksichtigt werden sollten. Der Wechsel zwischen Darstellungsformen, interaktive Animati-

onen sowie natürlichsprachliche Benutzerschnittstellen entsprechen wichtigen Funktionalitäten, die durch die angestrebte Lösung zur Unterstützung der Entwicklung adressiert werden sollten.

A3 Ergänzungen zum Instrumentarium

Abhängigkeit der Produktmerkmale von den Datenspekten (0 oder 1)	Datenspekte	Bewertungs- und Klassifikationsstabelle										AR-Potentiale (0-5)	AR-Potentiale zur Validierung der Produktmerkmale (0-5)	Nutzen der AR-Potentiale zur Validierung der Produktmerkmale (0-5)			
		Produktmerkmale					Interaktion										
3D-Modelldaten		Animationen		Kombination mit realer Umgebung	3D-Darstellung		Darstellung		Interaktion		Max. Nutzwert:		Nutzen		Nutzenbewertung		
2,7	1	3	2,3	0,7	0,3	Pauschaler Aufbereitungs- aufwand der Datenspekte (0-3)	Allinstellungs faktor der AR-Potentiale (1-5)	2	5	1	1	1	1	2	2	1	1
2	2	1	1	1	1	Materialien	Gestalt	5	4	1	1	1	1	1	1	7	13
2	2	1	1	1	1	Simulationsdaten	Proportionen und Form	4	4	1	1	1	1	1	1	10	31
2	2	1	1	1	1	Funktionalelementen	Räumliche Dimension, Maße und Abstände	5	5	1	1	1	1	1	1	10	35
2	2	1	1	1	1	3D-Darstellung	Lage und Wirkung in realer Umgebung	2	1	1	1	1	1	1	1	10	30
2	2	1	1	1	1	Optik	Farb-/Lichtwirkung	4	2	2	2	2	2	2	2	10	24
2	2	1	1	1	1	Transparenz	Oberflächenstruktur	3	3	1	1	1	1	1	1	8	24
2	2	1	1	1	1	Funktionen	Funktionalität	2	1	3	2	1	1	1	1	3	6
2	2	1	1	1	1	Materialien	Intensiv Systemverhalten	2	1	4	1	1	1	1	1	10	12
2	2	1	1	1	1	3D-Modelldaten	Endnutzer-Funktionalität	2	1	4	1	1	1	1	1	10	17
2	2	1	1	1	1	Animationen	Kinematik	3	4	1	1	1	1	1	1	9	12
2	1	1	1	1	1	3D-Modelldaten	Interaktion	2	1	2	2	2	2	2	2	10	18
2	2	1	1	1	1	Materialien	Benutzerschnittstelle	3	3	1	1	1	1	1	1	10	29
2	2	1	1	1	1	3D-Modelldaten	Einschrankbarkeit	3	1	1	1	1	1	1	1	10	19
2	1	1	1	1	1	Materialien	Erreichbarkeit	2	1	2	2	2	2	2	2	8	12
2	1	1	1	1	1	3D-Modelldaten	Struktur	4	2	1	1	1	1	1	1	3	9
1	1	2	1	1	1	Materialien	Systemstruktur	1	2	1	1	1	1	1	1	3	0,8
2	2	1	1	1	1	3D-Modelldaten	Konnektivität	2	3	1	1	1	1	1	1	3	9
2	2	1	1	1	1	Materialien	Anschlüsse / Externe Wirkbeziehungen	2	1	1	1	1	1	1	1	10	24
2	1	1	1	1	1	Materialien	Nonvisuelle Erscheinung	2	1	1	1	1	1	1	1	5	1
2	1	1	1	1	1	Materialien	Haptik	2	1	1	1	1	1	1	1	5	1
2	2	1	1	1	1	Materialien	Weitere	2	1	1	1	1	1	1	1	2	4
2	2	1	1	1	1	Materialien	Montage/Zusammenbau	3	2	1	1	1	1	1	1	10	17
2	2	1	1	1	1	Materialien	Einbau/Installation	3	1	1	1	1	1	1	1	10	20
2	2	1	1	1	1	Materialien		2	1	1	1	1	1	1	1	10	27
2	2	1	1	1	1	Materialien		2	1	1	1	1	1	1	1	10	23

Abbildung A-25: Generische Experten-Definitionen und -Bewertungen

Produktmerkmale						
Kategorie	Merkmal	Beschreibung	Nutzen von AR	Relevante Daten		
Gestalt	Proportionen und Form	Proportionen betreffen das Größenverhältnis der Elemente des Produkts untereinander sowie deren grundsätzliche Form und Positionierung zueinander. Eine 3D-Visualisierung ist auch ohne AR sinnvoll.	1,1			Es sind lediglich 3D-Modelle im Detailgrad 2 „Geometrie“ notwendig.
	Räumliche Dimension, Maße und Abstände	Das Merkmal betrifft die Größe des Produkts sowie Maße und Abstände im Bezug zur realen Umgebung. Eine dreidimensionale Einbettung in die reale Umgebung mittels AR bringt daher einen großen Nutzen.	2,7			Es sind lediglich 3D-Modelle im Detailgrad 2 „Geometrie“ notwendig.
	Lage und Wirkung in realer Umgebung	AR bietet sehr gute Möglichkeiten, um das Produkt dreidimensional in seiner realen Umgebung zu positionieren. So können die Wirkung des Produkts, die optische Erscheinung oder auch der Platzbedarf visualisiert werden.	3,0			Es sind 3D-Modelle notwendig, bei Bedarf mit optischen Details (Detailgrad 3)
Optik	Farb-/Lichtwirkung	Die Farbe und möglicherweise Muster sowie die grundsätzliche optische Wirkung des Produkts sind stets abhängig von der Umgebung. Daher bietet sich eine Validierung mit AR an. So kann der Benutzer zudem verschiedene Varianten aus freien Perspektiven betrachten.	2,1			Es sind 3D-Modelle mit gewissen optischen Details notwendig (vermutl. Detailgrad 3)
	Transparenz	Transparenz betrifft die Lichtdurchlässigkeit und -undurchlässigkeit des Produkts. Dabei kann die reale Umgebung eine entscheidende Rolle einnehmen. Der Einsatz AR ist grundsätzlich vielsprechend.	2,1			Es sind 3D-Modelle mit entsprechenden Transparenzeigenschaften notwendig.
	Oberflächenstruktur	Bei der Oberflächenstruktur geht es um die plastische Struktur der Oberfläche des Produkts. Diese kann optische und auch haptische Eigenschaften betreffen. Die Vorteile von AR sind eher gering.	0,5			Zusätzlich zu 3D-Modellen mit Texturdetails können Materialproben genutzt werden.
Funktionalität	Internes Systemverhalten	Verhalten, das im Inneren des Produkts stattfindet. Animationen bieten sich zur Veranschaulichung der Funktionalität an. Alternativ kann die originale Produktfunktionalität abgebildet werden. Metainformationen sowie individuelle Ansichten (bspw. Elektrik und Mechanik) können die Validierung unterstützen. Grundsätzlich ist jedoch kein großer Vorteil durch AR gegeben.	1,0			3D-Modelle sowie Animationen sollen vorhanden sein. Zudem können Originalfunktionalitäten integriert und Simulationsdaten dargestellt werden.
	Endnutzer-Funktionalität	Funktionalitäten, die bei einer Black-Box-Betrachtung relevant sind und äußere Aktionen und Reaktionen betreffen. Eine 3D-Darstellung unterstützt durch Animationen sowie unter Einbeziehung des Nutzers ist vielsprechend. Eine Einbettung in die Realität mittels AR bringt nur in Einzelfällen Vorteile.	1,5			3D-Modelle inkl. Animationen sind notwendig. Unterstützend können Funktionalitäten direkt eingebunden werden.
	Kinematik	Kinematik betrifft bewegliche Teile sowie Änderungen der Gestalt bzw. Form. Bspw. können sich Klappen öffnen, Kolben ausfahren oder das Produkt sich als Ganzes bewegen. Eine animierte 3D-Darstellung ist sehr gut geeignet zur Validierung. Der Einsatz von AR ist nur bedingt sinnvoll.	1,0			3D-Modelle inkl. Animationen sind notwendig. Unterstützend können Funktionalitäten und Simulationsdaten eingebunden werden.
Interaktion	Benutzerschnittstelle	Betrifft Ein- und Ausgabeverhalten sowie Interaktionselemente wie Knöpfe, Displays oder sonstige interaktive Elemente. Eine animierte 3D-Darstellung unter Einbeziehung des Benutzers ist hilfreich. Natürliche Benutzereingaben ermöglichen eine realitätsnahe Absicherung. Eine Kombination mit der Realität ist nur in Einzelfällen relevant, bspw. wenn physische Elemente einzubringen werden können oder bereits ein Produkt vorhanden ist, an dem die Benutzerschnittstelle angepasst werden soll.	1,5			3D-Modelle werden benötigt. Animationen und Funktionalitäten können im Einzelfall eingesetzt werden.
	Einsehbarkeit	Einsehbarkeit betrifft das Produkt an seinem Einsatzort. Das Produkt kann bspw. die Sicht auf andere Umgebungslemente versperren oder dessen Wahrnehmung durch die Umgebung beeinträchtigt werden. Entsprechend ist eine 3D-Darstellung in der realen Umgebung unter Einbeziehung des Nutzers sehr sinnvoll und nutzendifferenz.	2,5			Es werden lediglich 3D-Modelle benötigt, je nach individuellen Anforderungen mit unterschiedlichem Detailgrad.
	Erreichbarkeit	Erreichbarkeit betrifft vor allem ergonomische Aspekte. Unter Einbeziehung des Benutzers ist bspw. zu überprüfen, ob relevante Elemente am Produkt erreichbar sind. Eine 3D-Darstellung ist wichtig, die Einbettung in die Realität nur in Einzelfällen.	1,6			Es werden 3D-Modelle mit korrekter Struktur und Geometrie benötigt (Detailgrad 2).
Struktur	Systemstruktur	Die Systemstruktur betrifft die physische Zusammensetzung des Produkts aus einzelnen Komponenten und Teilsystemen sowie deren Wirkzusammenhänge und Wechselwirkungen untereinander. Eine 3D-Darstellung bietet eine gute Validierungsgrundlage. Animationen können die Verständnis erhöhen.	1,0			3D-Modelle sind notwendig, allerdings keine Details. Animationen sind optional zur Veranschaulichung einzelner Aspekte.
	Konnektivität	Konnektivität beinhaltet vor allem die Kommunikation und den Informationsaustausch mit anderen Systemen und Produkten über verschiedene Schnittstellen. Da vor allem die Funktionalität sowie Software-Schnittstellen betroffen sind, ist eine interaktive 3D-Darstellung eher wenig nutzendifferenz.	0,8			Funktionalitäten sind notwendig. 3D-Modelle, Animationen und Simulationsdaten können zur Veranschaulichung eingesetzt werden.
	Anschlüsse/Ext. Wirkbeziehungen	Anschlüsse betreffen vor allem am Produkt positionierte Anschlussstellen. Zudem sind Wirkbeziehungen zu anderen Systemen relevant, beispielsweise Stoff- oder Energieflüsse oder auch kinematische Abhängigkeiten. Eine 3D-Darstellung mittels AR in der realen Umgebung ist vielsprechend.	2,1			Geometrische 3D-Modelle werden in Kombination mit Animationen benötigt. Funktionalitäten können verwendet werden.
Nonvisuelle Erscheinung	Haptik	Haptik beinhaltet die fühlbare Beschaffenheit des Produkts. Das betrifft vor allem die Oberfläche und Form. Für die Validierung ist eine Visualisierung nicht ausreichend. Es bedarf der Integration von haptischen Interaktionen oder der Integration der physischen Materialien.	1,7			Lediglich Materialien sind notwendig.
	Akustik	Die Akustik beinhaltet Betriebsgeräusche oder auch akustisches Feedback an den Benutzer. Über eine akustische Ausgabe können die Geräusche wiedergegeben werden. In Kombination mit einer 3D-Visualisierung, u. a. auch in AR, kann der Benutzer die Geräusche im 3D-Raum wahrnehmen.	1,7			3D-Modelle sowie Audio-Signale werden benötigt. Animationen sind optional zur Veranschaulichung.
Weitere	Montage/Zusammenbau	Montage betrifft den Zusammenbau des Produkts an sich aus seinen Komponenten und Teilsystemen. Die Einsatzumgebung ist dabei zunächst nicht relevant. Über Animationen an 3D-Modellen kann eine anschauliche und leicht verständliche Erklärung und Absicherung der notwendigen Schritte erfolgen.	1,5			3D-Modelle inkl. Animationen sind notwendig.
	Einbau/Installation	Der Einbau betrifft die Installation am geplanten Einsatzort. Entsprechend ist eine AR-basierte Visualisierung in der realen Umgebung sinnvoll. Diese kann durch Animationen unterstützt werden. Dem Benutzer wird mittels AR eine realitätsnahe Validierung ermöglicht.	2,3			3D-Modelle inkl. Animationen sind notwendig.

Abbildung A-26: Produktmerkmal-Steckbrief

Datenaspekte						
Aspekt	Beschreibung	Detailgrad 1	Detailgrad 2	Detailgrad 3	Peuschaler Aufwand	
3D-Modelldaten	3D-Modelldaten stellen die Grundlage der AR-Visualisierung dar. Sie werden während der Entwicklung als CAD-Konstruktionsdaten erarbeitet. Für die Nutzung mittels AR müssen sie je nach eingesetzten CAD-Tool in verschiedene Formate (bspw. FBX) übertragen werden. Alternativ können 3D-Visualisierungsmodelle mittels verschiedener Modellierung-Tools erarbeitet werden. Je nach Anwendung kommen verschiedene Detailgrade notwendig sein.	Große Form – Für eine erste Abschätzung der Größe und der Proportionen reichen oft grobe Modelle. Sie können aufwandsarm erstellt werden.	Genaue Geometrie – Genaue Geometrien sind meist in den CAD-Modellen vorhanden. Sie können mit gewissen Aufwandsaufwänden verfeinert werden.	Texturierte Detail-Modelle – Für die Validierung optischer Aspekte sind realitätsnahe 3D-Modelle notwendig. Sie bedürfen aufwändiger Vorbereitungen.	2,7	
Animationen	Animationen sind eine sehr gute Möglichkeiten, um Funktionalitäten, Interaktionsmöglichkeiten und vor allem komplexe Sachverhalte zu vermitteln und zu veranschaulichen. Mit Animationen können Produktmerkmale und -eigenschaften dynamisch präsentiert und vom Kunden besser verstanden werden. Animationen lassen sich in Modellierungs- oder teilweise auch CAD-Tools erstellen. Alternativ können Animationen durch Software-Funktionalitäten live in der Anwendung erstellt werden. Im Vergleich zu 3D-Modelldaten sind die Aufwände eher gering.	Einfache Hinweise – Beispielsweise Hinweise auf die Verstärkung einer Stahlstütze oder die Hervorhebung eines Stabbaus aus dem Stahlbaus.	Transformationen – Durch Verschieben, Drehen und Rotation einzelner virtuellen Elemente können Funktionsdaten, kinematische Beziehungen und Zusammenhänge verdeutlicht werden.	Komplexe Animationen – Komplexe Animationen erfordern eine detaillierte Anpassung von 3D-Pfaden oder die Kombination mehrerer Transformationen entsprechend wie Änderungen des Designs etc.	1,0	
Funktionalitäten	Funktionalitäten entsprechen der reellen Funktionalität des Produkts oder stellen diese zumindest angenähert nach. Im Gegensatz zu Animationen handelt es sich daher nicht um speziell erstellte Hinweise zur Veranschaulichung, sondern um das richtige Verhalten. Funktionalitäten betreffen bei heutigen Systemen vorwiegend die Software. Beispieleweise ist die autonome Navigation eines Staubsauger-Roboters eine Funktionalität, für deren Überprüfung die entsprechende Software in die Validierung eingebunden werden müsste.	Kontinuierlich – Eine kontinuierliche Funktionalität kann bspw. zufälliges Herumfahren eines Staubsaugers sein, um das Design in verschiedenen Perspektiven zu validieren.	Reaktiv/interaktiv – Bei reaktivem oder interaktivem Funktionsintegration passiert sich das Verhalten der realen oder virtuellen Umgebung sowie Eingriffen des Benutzers an und verändert sich entsprechend.	Analytisch – Zusätzlich zur Reaktion auf die Umwelt findet eine Analyse der Einflüsse oder des Wahrnehmungen zur Anpassung des Verhaltens statt. Diese Analyse muss in die Anwendung integriert werden.	3,0	
Simulationsdaten	Simulationsdaten entstehen im Laufe der Entwicklung durch die digitale Simulation von Datenaspekten. Dies kann Materialflusssimulationen, Strömungen, Kinetik und Dynamik oder auch Stofftransportdaten betreffen. Unter Umständen ist eine Visualisierung der Ergebnisse mittels AR hilfreich. Dazu müssen die Simulationsdaten entsprechend vorbereitet und in die AR-Anwendung integriert werden.	Werte und Graphen – Die Werte und Graphen müssen sich als Werte und Graphen darstellen und somit relativ leicht in die AR-Anwendung integrieren.	Darstellung am Modell – Die Darstellung erfordert die Positionierung von Simulationsergebnissen am 3D-Modell zur Veranschaulichung.	Komplexe Visualisierungen – Komplexe Anwendungen erfordern Beispiele wie Strömungsvisualisierung oder farbcodeierte FEM-Analyseergebnisse müssen dargestellt werden.	2,3	
Materialien	Materialien werden vor allem zur Überprüfung von haptischen oder des Produktmaterial an sich betreffenden Aspekten benötigt. Das Material muss entsprechend physisch vorliegen. Je nach Detailgrad ist die Erarbeitung der Materialien als Grundlage der Validierung mit unterschiedlichen Beschaffungs- oder Fertigungsaufwänden verbunden.	Materialprobe – Eine Materialprobe kann ausreichend sein, um bspw. die Haptik der Oberfläche zu bewerten oder die Robustheit.	Form-Prototypen – Bspw. für ergonomische Aspekte ist die korrekte Form einer Komponente relevant. Das Material an sich ist in diesem Fall nicht relevant.	Material in Form – Sowohl die Form als auch das Material sind relevant. Entsprechend wird ein formgerechter Prototyp aus dem vorgesehenen Material benötigt.	0,7	
Audio/Sound	Akustische Signale oder Töne können beispielsweise eingesetzt werden, um die Läufgeräte und Betriebsgeräusche eines Produkts oder auch akustisches Feedback an den Benutzer zu übertragen. Durch die Validierung kann der Benutzer die akustische Wiedergabe mittels AR für die Validierung zu einem Eindruck der Gesamtheit in der realen Umgebung bekommen. Die Erarbeitung und Integration entsprechender Daten ist verhältnismäßig aufwandsarm.	Einfache Töne – Einfache Standardtöne sind für die Validierung ausreichend, bspw. ein Piepen oder Klingeln oder ein sonstiger Signaltönen.	Samples – Es werden vordefinierte Audio-Clips benötigt, bspw. ein Motorrauschen oder das Betriebsgeräusch einer Spülmaschine.	Reale Aufnahmen – Für die Validierung sind reale Aufnahmen bzw. Geräusche notwendig. Diese müssen individuell erarbeitet werden.	0,3	

Abbildung A-27: Datenaspekt-Steckbrief

AR-Potentiale					
Kategorie	Potential	Technik	Beschreibung	Aufwände	
Darstellung	Kombination mit realer Umgebung	AR-spezifisch	Mittels AR können virtuelle Prototypen in die reale Umgebung eingebettet und mit ihr kombiniert werden. Dies ermöglicht eine Validierung im realen Kontext. Dabei können Raumlichkeiten relevant sein oder auch die Einbindung vorhandener physischer Teilsysteme oder haptischer Elemente.	0,3	
	Dreidimensionale Darstellung	AR oder VR	AR ermöglicht eine interaktive dreidimensionale Darstellung digitaler Daten und Modelle. Benutzer können sich frei um die Modelle herumbewegen. Bei einer technischen Umsetzung mit binokularen AR-Brillen werden die Inhalte zudem dreidimensional und immersiv vom Benutzer wahrgenommen.	0,2	
	Animation der Inhalte	Digitale 3D-Visualisierung ausreichend	Die dargestellten Inhalte können durch Animationen angereichert werden. So können Vorgänge oder komplexe Sachverhalte anschaulich dargestellt und veranschaulicht werden.	1,3	
	Funktionale Interaktivität		In AR-Anwendungen können die originalen Funktionalitäten integriert oder angebunden werden. Beispielsweise können Software-Algorithmen in visuellen virtuellen Prototypen kombiniert werden und die Funktionalität realitätsnah abilden und entsprechend darstellen.	3,0	
	Flexible Inhaltsbearbeitung		Dargestellte Inhalte können flexibel ausgetauscht, verändert oder ausgetauscht werden. So lassen sich beispielsweise aufwandsarm verschiedene Varianten validieren oder Einblicke ins Innere eines Produkts ermöglichen.	1,6	
	Darstellung von Meta-Informationen		Meta-Informationen können als zusätzliche Visualisierungen direkt am 3D-Modell ergänzt werden. Das können bspw. Maßangaben, Sensorwerte oder auch eine Preisangabe etc. sein, die für die Validierung an den relevanten Stellen bei Bedarf eingeblendet werden können.	1,6	
	Personalisierte Ansichten		Für einzelne Personen können irrelevanten Informationen ausgetauscht bzw. relevante Informationen eingeblendet werden. Einem Elektriker kann bspw. eine andere Visualisierung bereitgestellt werden als einem Mechaniker, selbst bei gleichzeitiger Betrachtung des Prototypen über zwei einzelne Geräte.	2,4	
Interaktion	Einbeziehung des Benutzers	AR	AR ermöglicht Immersion. Entsprechend nimmt der Benutzer virtuelle Inhalte als Teil seiner Umgebung wahr. Er kann zudem interaktiv in die Visualisierung mit einbezogen werden und mit dem virtuellen Prototypen realitätsnah räumlich interagieren.	1,2	
	Natürliche Benutzereingaben	AR oder VR	AR bietet die Möglichkeit natürlicher Benutzereingaben. Bei entsprechender Sensorik stellen bspw. Gesten eine sehr natürlich Art der Interaktion mit virtuellen Inhalten dar. Zudem kann Interaktion über Sprache stattfinden, was bei heutigen technischen Systemen immer wichtiger wird.	1,6	
	Tangible User Interface	Digitale 3D-Visualisierung ausreichend	Es existieren verschiedene Interaktionslösungen, die haptische Feedback ermöglichen. Zudem kann der Einsatz von physischen Objekten als Interaktionselemente die Validierung unterstützen.	1,6	
	Akustische Ausgabe		Über AR-Geräte kann eine akustische Ausgabe erfolgen. Dabei kann eine dreidimensionale Verortung der Audioquellen im virtuellen Raum und entsprechend eine dreidimensionale räumliche Wiedergabe erfolgen, um bspw. Betriebsgeräusche und Lärm in einer realen Umgebung zu validieren.	1,1	
Anbindung	Software-Schnittstellen	Digitale 3D-Visualisierung ausreichend	Mittels Software-Schnittstellen können AR-Anwendungen bzw. virtuelle AR-Prototypen an andere Systeme oder Komponenten oder auch Software-Lösungen angebunden werden. So kann bspw. eine Kommunikation mit vernetzten Systemen ermöglicht und entsprechend validiert werden.	2,6	
Übergeordnet	Integriertes virtuelles Feedback	AR	Mit AR ist es möglich, Feedback direkt dreidimensional am virtuellen Prototypen zu geben. Feedback kann in Form von Text oder Audio-Statements erfolgen oder durch 3D-Icons oder Freihandzeichnungen, die dreidimensional am Modell verortet werden. So kann das Feedback besser zugeordnet und verstanden werden.	2,1	
	Digitale Telemetrie und Dokumentation	AR oder VR	Mit AR lassen sich verschiedene Telemetrie-Daten dokumentieren und auswerten. Das können die Blickrichtung, Kopfbewegung oder auch das Bewegungsverhalten des Benutzers sein, dass für die Validierung relevant ist. Die digitale Durchführung ermöglicht grundsätzlich eine einfache Dokumentation der Validierungsergebnisse.	3,0	
	Automatisierte Benutzerführung	Digitale 3D-Visualisierung ausreichend	Benutzer können mit AR automatisiert durch die Validierung geführt werden. Dabei kann die Visualisierung bspw. Schritt-für-Schritt erfolgen oder der Benutzer durch Visualisierungen gezielt auf gewisse Aspekte aufmerksam gemacht werden. Zudem können Anforderungen oder Fragen der Reihe nach eingeblendet werden.	2,7	
	Ortsunabhängigkeit		Beim Einsatz mobiler AR-Geräte ist eine Validierung an beliebigen Orten möglich. Virtuelle Prototypen können im Büro, in der Maschinenhalle, im Wohnzimmer oder auch draußen auf der Straße und somit in einem beliebigen Kontext erlebt und validiert werden.	0,2	
	Kollaboration		Ar ermöglicht neben der Visualisierung auch die Kommunikation und Vernetzung von Akteuren. So können Produkte auch standortübergreifend gemeinsam mittels AR erlebt werden. Bspw. kann ein Kunde einen Prototypen betrachten während der Entwickler live aus der Ferne zugeschaltet ist und in die Perspektive des Kunden eintaucht.	1,8	

Abbildung A-28: AR-Potential-Steckbrief

A4 Ergänzungen zur Anwendung und Bewertung

The table structure is as follows:

- Essentielle Komponenten:**
 - Anzeigesystem: Projektions-basiert, Optical-See-Through, Video-See-Through
 - Bewegungsverfolgung: Rekonstruktives Tracking SLAM, GPS-Tracking, Elektromagnetisches Tracking, Mechanisches Tracking, Infrarot-Tracking, WiFi-Tracking, Sequenzielles Tracking, Ultraschall-Tracking, UWB-Tracking, Inertiales Tracking, Bluetooth-Tracking
 - Datenspeicher: Cloudspeicher, MemoryCard, Festplatte, Interner Speicher
 - Interaktionssystem: Tasten-Steuerung, Touchscreen-Steuerung, Touchpad-Steuerung, Controller-Steuerung, Sprachsteuerung, Computerperipherie-Steuerung
- Optionale Komponenten:**
 - Anschluss: DisplayPort, Apple-Lightning, USB, HDMI
 - Audio-Input: Mikrofon, KopfhörerMikrofon, BluetoothMikrofon
 - Audio-Output: Kopfhörer, Lautsprecher, BluetoothKopfhörer
 - Daten-Verbindung: ZigBee, Bluetooth, FIR-IrDA, NFC
 - Fremdkörperschutz: IP5x, IP4x, IP6x, IP1x, IP0x, IP3x, IP2x
 - Kühlung: PassiveKühlung, AktiveKühlung
- Umgebungs faktoren:**
 - Arbeitsschutzzvorschrift: Schutzhelm-Vorschrift, Schutzbrillen-Vorschrift, Ohropax-Vorschrift
 - Arbeitsumgebung: Büro, Werkstatt, Reinraum, Sterilraum
 - Einsatzort: Outdoor, Indoor
 - Einsatzvorbereitung: NichtPräparierteUmgebung, PräparierteUmgebung
 - Fremdkörper: Größer50mm, Kleiner1mm, Kleiner50mm, Kleiner12mm, utogen1:mm
 - Handhabung: KeineHandFreiMitFreiraum, EineHandFreiMitFreiraum, BeideHändeFrei, KeineHandFrei, EineHandFrei
 - IT-Infrastruktur: UMTS-Netz, Cloud, Steckdose, FünfG-Netz, WLAN-Router
 - Lautstärke: NormaleLautstärke, HoheLautstärke, GeringeLautstärke, Ruhebereich
 - Lichtverhältnis: SchwacheBeleuchtung, StarkeBeleuchtung, DirekteBeleuchtung, SchwankendeBeleuchtung, GemäßigteBeleuchtung

Abbildung A-29: Geforderte und ausgeschlossene System-Komponenten und Umfeldfaktoren im Rahmen der AR-System-Konfiguration inkl. vom Anwender vorgenommener zusätzlicher Konfigurationen (rote Kreise) (Ausschnitt 1/2).

Gestensteuerung	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Primäre Stromversorgung	<input checked="" type="checkbox"/>	
Kabelgebunden	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Akku	<input type="checkbox"/>	<input type="checkbox"/>
Prozessor	<input checked="" type="checkbox"/>	
HighPerformance	<input type="checkbox"/>	<input type="checkbox"/>
LowPerformance	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Exkludiert	<input type="checkbox"/>	<input type="checkbox"/>
MediumPerformance	<input type="checkbox"/>	<input type="checkbox"/>
Sekundäre Stromversorgung	<input checked="" type="checkbox"/>	
PowerBank	<input type="checkbox"/>	<input type="checkbox"/>
Systemtyp	<input checked="" type="checkbox"/>	
DesktopPC	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Laptop	<input type="checkbox"/>	<input checked="" type="checkbox"/>
HMPD	<input type="checkbox"/>	<input checked="" type="checkbox"/>
HMD-Monokular	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Tablet	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Projektor	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Smartphone	<input type="checkbox"/>	<input checked="" type="checkbox"/>
HMD-Binokular	<input type="checkbox"/>	<input type="checkbox"/>
Netzwerk-Verbindung		
UMTS	<input checked="" type="checkbox"/>	<input type="checkbox"/>
LAN	<input type="checkbox"/>	<input type="checkbox"/>
WLAN	<input type="checkbox"/>	<input type="checkbox"/>
FünfG	<input type="checkbox"/>	<input type="checkbox"/>
Luftfeuchtigkeit		
NormaleLuftfeuchtigkeit	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Hoheluftfeuchtigkeit	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Geringeluftfeuchtigkeit	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Lufttemperatur		
Geringelufttemperatur	<input type="checkbox"/>	<input checked="" type="checkbox"/>
NormaleLufttemperatur	<input type="checkbox"/>	<input checked="" type="checkbox"/>
HoheLufttemperatur	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Sensorik		
IR-Kamera	<input type="checkbox"/>	<input type="checkbox"/>
Feuchtigkeitssensor	<input type="checkbox"/>	<input type="checkbox"/>
RGB-Kamera	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Umgebungslicht-Sensor	<input type="checkbox"/>	<input type="checkbox"/>
Drucksensor	<input type="checkbox"/>	<input type="checkbox"/>
Ultraschall-Sensor	<input type="checkbox"/>	<input type="checkbox"/>
Beschleunigungssensor	<input type="checkbox"/>	<input type="checkbox"/>
Gyroskop	<input type="checkbox"/>	<input type="checkbox"/>
Magnetfeld-Sensor	<input type="checkbox"/>	<input type="checkbox"/>
GPS-Sensor	<input type="checkbox"/>	<input type="checkbox"/>
RGB-D-Kamera	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Mobilität		
Stationär	<input type="checkbox"/>	<input type="checkbox"/>
Mobil	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Nutzer		
Brillenträger	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Elektrisch-betriebeneImplantate	<input type="checkbox"/>	<input type="checkbox"/>
Insulinpumpe	<input type="checkbox"/>	<input type="checkbox"/>
Schwerhörig	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Herzschrittmacher	<input type="checkbox"/>	<input type="checkbox"/>
Nässe		
Tropfwasser	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Sprühwasser	<input type="checkbox"/>	<input checked="" type="checkbox"/>
ZeitweiligesUntertauchen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
TropfwasserUnter15Grad	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Spritzwasser	<input type="checkbox"/>	<input type="checkbox"/>
SchweresStrahlwasser	<input type="checkbox"/>	<input type="checkbox"/>
Trocken	<input type="checkbox"/>	<input type="checkbox"/>
DauerhaftesUntertauchen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Strahlwasser	<input type="checkbox"/>	<input type="checkbox"/>
Trage-Option		
Alltags-Brille	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Schutzhelm	<input type="checkbox"/>	<input type="checkbox"/>
Schutzbrille	<input type="checkbox"/>	<input type="checkbox"/>
Oberflächentextur		
WeicheOberflächen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
AbsorbierendeOberflächen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
SpiegelndeOberflächen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Störquelle		
BewegendeReferenzsysteme	<input type="checkbox"/>	<input checked="" type="checkbox"/>
MagnetischeGegenstände	<input type="checkbox"/>	<input checked="" type="checkbox"/>
SichtBlockierendeGegenstände	<input type="checkbox"/>	<input checked="" type="checkbox"/>
HochspannungsSchaltvorgänge	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Häuserschlucht	<input type="checkbox"/>	<input checked="" type="checkbox"/>
hochfrequenteGeräuschquellen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Funkeninduktoren	<input type="checkbox"/>	<input checked="" type="checkbox"/>
elektromagnetischeFelder	<input type="checkbox"/>	<input checked="" type="checkbox"/>
MetallischeGegenstände	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Proceed with configuration

Abbildung A-30: Geforderte und ausgeschlossene System-Komponenten und Umfeldfaktoren im Rahmen der AR-System-Konfiguration inkl. vom Anwender vorgenommener zusätzlicher Konfigurationen (rote Kreise) (Ausschnitt 2/2).