

Product Safety and Quality Engineering
Manuel Löwer und Nadine Schlüter (Hrsg.)

7

Entwicklung eines KI-basierten Workflows zur Unterstützung der Projektleitung hinsichtlich einer kompetenzbasierten Ressourcenplanung von Industrieprojekten

Amirbabak Ansari



**Entwicklung eines KI-basierten Workflows
zur Unterstützung der Projektleitung
hinsichtlich einer kompetenzbasierten
Ressourcenplanung von Industrieprojekten**

Dissertation
zur Erlangung eines Doktorgrades

in der
Fakultät für Maschinenbau und Sicherheitstechnik
der
Bergischen Universität Wuppertal

vorgelegt von
Amirbabak Ansari

Wuppertal 2025

Product Safety and Quality Engineering – Band 7

Amirbabak Ansari

Entwicklung eines KI-basierten Workflows zur Unterstützung der Projektleitung
hinsichtlich einer kompetenzbasierten Ressourcenplanung von Industrieprojekten

Herausgeber: Manuel Löwer und Nadine Schlüter,
Bergische Universität Wuppertal,
Fachgebiet Produktsicherheit und Qualität,
Gaußstraße 20, 42119 Wuppertal

Umschlaggestaltung: Franz Wieck

Zugl.: Wuppertal, Univ., Diss. 2025

DOI: <https://doi.org/10.25926/BUW/0-931>

URN: <https://nbn-resolving.org/urn:nbn:de:hbz:468-2-6524>

Die Deutsche Nationalbibliothek verzeichnet diese Publikation
in der Deutschen Nationalbibliografie; detaillierte bibliografische
Daten sind im Internet über <http://dnb.dnb.de> abrufbar.

Dieses Werk steht, soweit nicht anders angegeben, unter der
Creative Commons-Lizenz CC BY-NC 4.0

<https://creativecommons.org/licenses/by-nc/4.0/deed.de>

Kurzfassung

Die zunehmende Komplexität auf organisatorischer Ebene sowie die steigende Komplexität der Produkte stellt Unternehmen und ihre Industrieprojekte vor Herausforderungen. Einerseits sollen die Unternehmen bei der Planung der Projekte noch schneller reagieren und Entscheidungen über Ressourcen, Kompetenzen, Zeit und Geld treffen, um die Marktposition gegen-über der Konkurrenz zu behaupten. Andererseits lassen sich Industrieprojekte aufgrund ihrer Komplexität nicht ohne weiteres prospektiv abschätzen.

Es gibt genügend Beispiele von Projekten in der Industrie, die aufgrund ihrer Komplexität falsch geplant wurden, deren Ressourcen nicht effizient verteilt und die Teams nicht hinsichtlich ihrer tatsächlichen Kompetenzen und Schnittmengen zusammengestellt wurden, weshalb die Anforderungen der Stakeholder und die Projektziele nicht erreicht werden konnten. Diese Situation führt dazu, dass Industrieprojekte verzögert werden oder scheitern, was Folgekosten und Imageschäden für das Unternehmen bedeutet. Um diese Situation zu überwinden, wird in dieser Dissertation einen systematischen Ansatz zur kompetenzbasierten Projektplanung von Industrieprojekten entwickelt, der moderne IT-basierte Werkzeuge und Lösungen aus dem Bereich der Künstlichen Intelligenz (KI) zur Unterstützung nutzt.

Durch die Abfrage und Analyse von Informationen aus Qualitätsmanagementsystemen des Unternehmens zu Projektrollen, Kompetenzen von Teammitgliedern usw. und die Kombination und den Vergleich dieser Informationen mit den abgeschlossenen Projekten des Unternehmens können Rückschlüsse auf die Planung der zukünftigen Projekte gezogen werden. In dieser Arbeit wird die Möglichkeit untersucht, Werkzeuge und Algorithmen aus dem Bereich des maschinellen Lernens und der Computerlinguistik einzusetzen, um den Projektmanager in der Projektplanungsphase bei der vorausschauenden genauen Schätzung der Ressourcen unter Verwendung der Informationen aus den Qualitätsmanagementsystemen zu unterstützen.

Schlagwörter: Projektmanagement, Qualitätsmanagement, KI-Workflow, NLP, Ontologie

Abstract

The increasing complexity at the organizational level, as well as the growing complexity of products, presents challenges for companies and their industrial projects. On one hand, companies are expected to react more quickly in project planning and make decisions regarding resources, competencies, time, and money in order to maintain their market position against competitors. On the other hand, due to their complexity, industrial projects cannot be easily estimated prospectively.

There are numerous examples of industrial projects that were poorly planned due to their complexity, where resources were not distributed efficiently, and teams were not assembled based on their actual competencies and overlaps. As a result, the stakeholders' requirements and project goals were not met. This situation often leads to project delays or failures, resulting in additional costs and reputational damage for the company. To overcome this issue, this dissertation develops a systematic approach to competence-based project planning for industrial projects, utilizing modern IT-based tools and solutions from the field of Artificial Intelligence (AI).

By querying and analyzing information from the company's quality management systems regarding project roles, team members' competencies, etc., and by combining and comparing this information with the company's completed projects, conclusions can be drawn for planning future projects. This work investigates the possibility of using tools and algorithms from the field of machine learning and computational linguistics to support project managers during the project planning phase by providing precise forecasts for resource estimation, using the information from quality management systems.

Keywords: Project Management, Quality Management, AI Workflow, NLP, Ontology

Inhaltsverzeichnis

1 Einleitung.....	1
1.1 Problemstellung und Eingrenzung	1
1.2 Zielbildung	10
1.3 Herangehensweise	12
2 Begrifflichkeiten und theoretische Grundlagen	14
2.1 Künstliche Intelligenz	14
2.1.1 Algorithmen	16
2.1.2 Natural Language Processing (NLP)	17
2.1.3 Workflow	19
2.2 Projektmanagement.....	20
2.2.1 Projektmanagements, Phasen und Aufgaben	22
2.2.2 Ressourcenplanung	23
2.3 Kompetenz und Kompetenzarten	25
2.3.1 Kompetenzmodelle.....	27
2.3.2 Kompetenzbasierte Ressourcenplanung.....	29
2.4 Softwareentwicklung	32
2.4.1 Das V-Modell.....	33
2.4.2 Automotiv SPICE (Das Assessmentmodell)	36
2.4.3 eDeCoDe-Ansatz zur Systemmodellierung	38
3 Stand der Wissenschaft und Technik	41
3.1 Wissenschaftliche Ansätze zum Projektmanagement in Software Entwicklung	41
3.2 Wissenschaftliche Ansätze zur Kompetenzmessung in Software Entwicklung	52
3.3 Wissenschaftliche Ansätze zur Kl-gestützten Projektplanung	58
3.4 Technische Ansätze zum Projekt- und Ressourcenmanagement.....	65
4 Zwischenfazit.....	74
4.1 Bedarf eines Kl-gestützten Workflows zur kompetenzbasierten Projektplanung.....	74
4.2 Anforderungen an den Kl-gestützten Workflows zur kompetenzbasierten Projektplanung (KIPKO)	75
4.2.1 Grundlegende Anforderungen (G).....	75
4.2.2 Anforderungen an den NLP-Algorithmus (N).....	76
4.2.3 Anforderungen an den Kl-gestützten Workflow (W)	77
5 Entwicklung des Kl-gestützten Workflows zur kompetenzbasierten Projektplanung	79
5.1 Theoretische Entwicklung der Vorbereitungsphase (S1).....	80

5.1.1 Identifikation von relevanten Informationen und Dokumenten	80
5.1.2 Aufbau einer Informationsstruktur	88
5.2 Theoretische Entwicklung des NLP-Algorithmus (S2)	91
5.2.1 Aufbau von NLP-Pipeline.....	91
5.2.2 eDeCoDe als Ontologie	108
5.3 Theoretische Entwicklung des KI-gestützten Workflows (S3).....	115
5.3.1 Back-End	115
5.3.2 Front-End.....	117
5.4 Technische Umsetzung des KI-gestützten Workflows	119
5.4.1 Programmierung von NLP-Pipeline.....	120
5.4.2 Programmierung von Benutzeroberfläche	126
6 Validierungsbeispiel und Diskussion	129
6.1 Validierungsbeispiel	130
6.1.1 Vorbereitung (VS1).....	131
6.1.2 Aufbau und Visualisierung der Ontologie (VS2)	132
6.1.3 Integration von GUI (VS3)	135
6.2 Diskussion.....	137
6.2.1 Auswertung der grundlegenden Anforderungen (G1 bis G5)	137
6.2.2 Auswertung der Anforderungen an NLP-Algorithmus (N1 bis N6).....	140
6.2.3 Auswertung der Anforderungen an KI-gestützten Workflow (W1 bis W4).....	143
6.3 Zusammenfassung der Erkenntnisse der Validierung	146
6.4 Umsetzbarkeit der Verbesserungspotentiale im theoretischen Konzept	149
6.4.1 Verbesserung der Datensicherheit	149
6.4.2 Implementierung eines Benachrichtigungssystems	149
6.4.3 Erweiterung der Berichts-Funktionalität.....	150
6.4.4 Einführung von Feedback-Mechanismen	150
7 Ergebnisse des Dissertationsvorhabens.....	152
7.1 Fazit des Dissertationsvorhabens.....	152
7.2 Ausblick auf weitere Forschungsvorhaben und Forschungsschwerpunkte	155
Literaturverzeichnis.....	161
Anhang.....	189
A.1 DMM und DSM der Ontologiegraphen.....	189
A.2 Knoten der Ontologiegraphen.....	190
A.3 Kanten der Ontologiegraphen.....	192

Abbildungsverzeichnis

Abbildung 1: Ursachen für die interne u. externe Komplexität [Thiebes und Plankert 2014, S. 172]	1
Abbildung 2: Kausalkette der Dissertationsproblematik [Eigene Darstellung].....	3
Abbildung 3: Forschungsfelder [Eigene Darstellung]	6
Abbildung 4: Verlauf von Dissertation [Eigene Darstellung]	13
Abbildung 5: Entwicklung der künstlichen Intelligenz [Teich 2020, S. 277].....	15
Abbildung 6: Verschiedenen Bereiche der künstlichen Intelligenz in Anlehnung an [Russell und Norvig 2010].....	16
Abbildung 7: Projektmanagementphasen und Fokussierung der Forschungsarbeit [Eigene Darstellung].....	23
Abbildung 8: Zusammenhang zwischen Linie und Projekt im unternehmensweiten Ressourcenmanagement in Anlehnung an [Sterrer 2014]	24
Abbildung 9: Hierarchie der Kompetenzarten in Anlehnung an [Nieke 2002]	25
Abbildung 10: Einordnung des Kompetenzbegriffs in Relation zu Qualifikationen, Wissen und Fertigkeiten. Modifizierte Abbildung in Anlehnung an [Erpenbeck et al. 2017].	27
Abbildung 11: Strategisches Kompetenzmodell und Anwendungsbereiche in Anlehnung an [Sauter und Staudt 2016b]	29
Abbildung 12: Makrophasen der kontinuierlichen Kompetenzverbesserung auf Grundlage der Unternehmensziele und Visionen in Anlehnung an [Draganidis und Mentzas 2006].....	30
Abbildung 13: Der Prozess der Softwareentwicklung in Anlehnung an [Plewan und Poensgen 2011].....	33
Abbildung 14: Das Wasserfallmodell in Anlehnung an [Boehm 1981]	34
Abbildung 15: Das V-Modell nach Boehm [Boehm 2001].....	35
Abbildung 16: Automotive SPICE-Prozessreferenzmodell – Überblick [ASPICE 2023, S. 15].	36
Abbildung 17: Semantik und Syntax von DeDoDe vs eDeCoDe in Anlehnung an [Schlüter 2023].....	39
Abbildung 18: Grundschema von eDeCoDe [Mistler et al. 2021].....	40
Abbildung 19: Darstellung des Lösungskonzepts [Eigene Darstellung].....	79
Abbildung 20: Soft-Skills in Anlehnung an Links [Peters-Kühlinger und John 2022], Mitte [Jakoby 2015], und Rechts [Bohinc 2012].....	81

Abbildung 21: Projektmanagementaufwand im Projektverlauf in Anlehnung an [Sterrer 2014]	83
Abbildung 22: Aufbau von NLP-Pipeline [Eigene Darstellung]	92
Abbildung 23: Dependency Parsing vs Constituency Parsing [Eigene Darstellung]	97
Abbildung 24: Verknüpfungslogik von eDeCoDe und Automotive SPICE [Eigenen Darstellung]	109
Abbildung 25: Anwendung von eDeCoDe-Logik auf Projektumfeld [Eigene Darstellung]	109
Abbildung 26: Produktentwicklungsprozesse in VDA-Scope in Anlehnung an [ASPICE 2023]	111
Abbildung 27: Notwendige Rollen abgeleitet von Automotive SPICE [Eigene Darstellung]	112
Abbildung 28: Kompetenzarten der personellen Ressourcen [Eigene Darstellung]	113
Abbildung 29: Entwickelte Ontologie basierend auf DSM- und DMM-Matrizen [Eigene Darstellung]	114
Abbildung 30: Das KI-Workflow [Eigene Darstellung]	115
Abbildung 31: Spezifizierung von Back-End	116
Abbildung 32: Spezifizierung von Front-End	117
Abbildung 33: Die Benutzer Oberfläche [Eigene Darstellung]	119
Abbildung 34: Schritte der Validierung	129
Abbildung 35: (Teil-)Struktur der Excel-Tabelle für Entwicklungsprozesse	131
Abbildung 36: (Teil-)Struktur der Excel-Tabelle für Kanten	133
Abbildung 37: Graph der Ontologie (Rechts) und Isolierte Betrachtung von einem Bereich (Links)	134
Abbildung 38: Demonstration der Fokusfunktion im Netzwerk	135
Abbildung 39: GUI des Tools	136

Tabellenverzeichnis

Tabelle 1: Anwendungsgebiete von NLP	7
Tabelle 2: Evaluation des Standes der Wissenschaft und Technik	8
Tabelle 3: Anwendungsbereiche der Algorithmen nach [Santinbánez Koref 2015, S. 5].....	17
Tabelle 4: Die Phasen des Projektmanagementzyklus in Anlehnung an [DIN 69901-5: 2009]	22
Tabelle 5: Definition der eDeCoDe-Elemente in Anlehnung an [Schlüter 2023]	38
Tabelle 6: Gegenüberstellung Kompetenzformen u. -methoden in Anlehnung an [Erpenbeck et al. 2017].....	52
Tabelle 7: Vor- und Nachteile der gängige Kompetenzmessungsmethoden	57
Tabelle 8: Zusammenfassung der Auswertung von ausgewählten Softwarelösungen	73
Tabelle 9: Auflistung aller definierten Anforderungen	77
Tabelle 10: Übersicht der personellen Ressourcen [Eigene Darstellung]	82
Tabelle 11: Übersicht der zeitlichen Ressourcen [Eigene Darstellung]	84
Tabelle 12: Übersicht der finanziellen Ressourcen [Eigene Darstellung]	85
Tabelle 13: Matrixdiagramm zur Gegenüberstellung der relevanten Informationen und den Dokumenten	87
Tabelle 14: Informationsstruktur (Filter) [Eigene Darstellung].....	88
Tabelle 15: Attribute, die bei einer morphologischen Analyse entstehen [Eigenes Beispiel]	95
Tabelle 16: Verdeutlichung der NLP-Pipeline anhand eines einfachen Beispiels.....	102
Tabelle 17: Übersicht der Fähigkeiten von NLP-Ansätze zur Behandlung verschiedener NLP-Aspekte.....	107
Tabelle 18: Mapping der eDeCoDe-Logik auf das Projektumfeld.....	110
Tabelle 19: Die sechs Hauptbereiche des Skripts	120
Tabelle 20: Hauptbereiche des GUI-Skripts.....	127
Tabelle 21: Übersicht über die Anforderungen und deren Erfüllungsgrad	148

1 Einleitung

1.1 Problemstellung und Eingrenzung

In der wissenschaftlichen Sprache ist „Komplexität“ ein weit verbreiteter Begriff geworden, um die Komplikationen des Lebens in den heutigen Gesellschaften zu beschreiben. Bei allem metaphorischen Gebrauch ist dies ein gesellschaftlicher Befund, der an sich schon bemerkenswert ist [Adolf 2017]. Aber was bedeutet der Begriff Komplexität in der Industrie? Nach Luhmann ist „Komplexität“ eine Systemeigenschaft mit zwei Dimensionen,

1. die Vielfalt der Elemente (Varietät) und
2. die Vielfalt der Beziehungen (Konnektivität)“ [Luhmann 2005].

Nach [Thiebes und Plankert 2014] hat die Komplexität mit Bezug auf Unternehmen zwei Sichten, unternehmensexterne und unternehmensinterne Komplexität, jeweils mit diversen Ursachen. Externe Ursachen lassen sich auf Markt- und Technologieveränderungen, aber auch auf gesellschaftliche Anforderungen und Wettbewerbszwänge zurückführen. Parallel dazu lassen sich die internen Ursachen der Komplexität nach Systemen, Strategien, Strukturen und Prozessen innerhalb des Unternehmens gruppieren. Abbildung 1 visualisiert die beiden Arten von Komplexität und die jeweiligen Ursachen.



Abbildung 1: Ursachen für die interne u. externe Komplexität [Thiebes und Plankert 2014, S. 172]

Kernfrage ist hier, wie ein Unternehmen aus interner Sicht zur Handhabung von Komplexität strukturiert sein sollte, um externe Anforderungen und daraus resultierende externe Komplexität wahrnehmen und erfüllen zu können. Das bedeutet, die vom Markt und anderen Veränderungs- und Trendtreibern erzwungenen Produkt- und Flexibilitätsanforderungen aufzugreifen und intern über Organisationsstrukturen, Prozesse und Wertschöpfungsstrategien möglichst effizient zu erfüllen. Auf zusätzliche Anforderungen wird oft nach bestem Wissen und Gewissen mit neuen fallbezogenen Lösungen auf der Prozess-, Organisations- oder IT-Ebene reagiert [Schatz et al. 2014] und kaum nach einer generischen Lösung gesucht. Die so entstehenden Komplexitätskosten lassen sich nur mit großem Aufwand transparent machen und zieltgerecht reduzieren [Wilson und Perumal 2010].

Parallel zur Komplexität auf der Organisationsebene ist auch die zunehmende Komplexität von Produkten eine Herausforderung für Unternehmen und deren Industrieprojekte. Gemäß [Winzer 2016] haben sowohl die „Individualisierung von Produkten“ als auch die „ressourcenschonende Miniaturisierung und Verschmelzung von Produkten“ zu einer erhöhten Komplexität der Produkte geführt. Nach [Abramovici und Herzog 2016] werden Beispiele vorgestellt, die auf die Entwicklung durch Intelligenz und Kommunikation vom mechatronischen Produkt zum intelligenten Produkt abzielen. In der Vergangenheit hatte zum Beispiel ein Mobiltelefon die alleinige Aufgabe, das Telefonieren zu ermöglichen. Heute sind Smartphones bereits in der Lage, Bilder und Videos aufzunehmen, während der Fahrt zu navigieren, Aufgaben zu organisieren und automatisch daran zu erinnern. Sie können also offensichtlich mehr als nur anrufen. Laut [Grimm 1987], [Cartin 1999] und [Müller-Lindenberg 2005] kann Komplexität der Produkte zu Fehlfunktionen/Fehlern und damit zu entsprechenden Folgeschäden, wie z.B. Ausfällen, führen. Aus der Sicht von [Schmitt et al. 2016] erfordert die zunehmende Produktkomplexität eine gleichermaßen zunehmende Prozesskomplexität in der Produktion, verursacht aber parallel ein vermehrtes Auftreten von Instabilitäten. Dieses Argument wird durch eine internationale, industrielle Fallstudie aus dem Jahr 2013 zu den Korrelationen zwischen Komplexität, Qualität und kognitiver Automatisierung gestützt, in der ein positives Verhältnis zwischen Komplexität und Montagefehlern nachgewiesen wurde [Fast-Berglund et al. 2013]. Darauf basierend kann die Schlussfolgerung gezogen werden, dass die Produktkomplexität in direkter Beziehung zum Produktentstehungsprozess (PEP) steht und zu einer Erhöhung der Komplexität für industrielle Projekte¹ und Aufgaben führt. In dieser Forschung wird der Schwerpunkt bei unterschiedlichen Arten von Industrieprojekten in Entwicklungsprojekte eingebettete Software (Engl. Embedded Software Development) und ihren Entstehungsprozesse in der Automotive-Branche gelegt. Durch die starke Vernetzung der einzelnen Prozesse, der Aufgaben, der verschiedenen Einflussgrößen und der beteiligten Abteilungen im Unternehmen steigt die Fehleranfälligkeit [Kapici 2005]. Aber welchen Einfluss hat diese Fehleranfälligkeit auf ein Produkt und vor allem auf das Unternehmen selbst? Einerseits kann

¹ Projekt: „Ein Projekt ist „einmaliger Prozess, der aus einem Satz von abgestimmten und gesteuerten Tätigkeiten mit Anfangs- und Endterminen besteht und durchgeführt wird, um unter Berücksichtigung von Beschränkungen bezüglich Zeit, Kosten und Ressourcen ein Ziel (3.7.1) zu erreichen, das spezifische Anforderungen (3.6.4) erfüllt [DIN ISO 9000: 2015, S. 34]“.

eine erhöhte Fehleranzahl immer die Qualität oder sogar die Sicherheit eines Produktes beeinträchtigen, da sie zum Ausfall eines Systems führen und dessen Nutzung beeinträchtigen kann [Gembrys und Herrmann 2009]. Andererseits kann sie Auswirkungen auf das gesamte Unternehmen haben, da die vom Kunden geforderte Qualität und Sicherheit immer mehr zu einem entscheidenden Kriterium für den Erfolg eines industriellen Projekts und somit des ausführenden Unternehmens wird [Hartwig 2010]. Die Nichterfüllung von Kundenwünschen kann sogar weitreichende Folgen wie Kundenabwanderung nach sich ziehen und dem Image des Unternehmens schaden [Vöcking et al. 2008]. Daraus resultierend müssen die Unternehmen, die industrielle Projekte durchführen,

- auf der einen Seite die Entscheidungen hinsichtlich der Ressourcen, Kompetenzen, Zeit- und Geldaufwand bei der Projektplanung und Projektdurchführung immer schneller treffen, um die Marktposition gegenüber der Konkurrenz beizubehalten und
- auf der anderen Seite die Herausforderung meistern, dass die Industrieprojekte aufgrund ihrer Komplexität nicht prospektiv einfach abschätzbar sind [S.Phambmixay et al. 2018].

Diese bipolare Situation führt dazu, dass die Industrieprojekte sich verzögern oder fehlschlagen [Khare et al. 2017], [Grimm 2003] und das bedeutet Folgekosten und Imageschäden des Unternehmens. Dies sollte vermieden werden. Deshalb ist die in Abbildung 2 exemplarisch dargestellte Kausalkette der Problematik im Rahmen einer Dissertation zu untersuchen.



Abbildung 2: Kausalkette der Dissertationsproblematik [Eigene Darstellung]

Es gibt genug Beispiele von Projekten in der Industrie, die aufgrund ihrer Komplexität falsch geplant wurden, deren Ressourcen nicht effizient verteilt und die Teams nicht im Hinblick ihrer tatsächlichen Kompetenzen und Schnittmengen zusammengesetzt wurden, weshalb die Stakeholder-Anforderungen und darauf basierende Projektziele nicht erreicht werden konnten. An dieser Stelle kann, unter anderem, die Rückrufaktion des Smartphones „Galaxy Note 7“ von Samsung Electronics als Beispiel genannt werden.

Am 2. August 2016 brachte Samsung in New York das neue Luxus-Smartphone „Galaxy Note 7“ auf den Markt. Bereits am 19. November 2016 wurde die Produktion dieses Smartphones jedoch ausgesetzt [Yun et al. 2018]. Grund dafür war eine Überhitzung und teilweise Explosion des Akkus während des Ladevorgangs. Laut offizieller Aussagen des Leiters der Smartphone-Division von Samsung, war es ein Versäumnis von Samsung, die Probleme im Akku-Design und Produktionsprozess vor der Markteinführung zu identifizieren und zu beseitigen [Yun et al. 2018; Zeit 2017]. Diese Strategie- und Planungsunsicherheit hat zu einem großen wirtschaftlichen Verlust und Imageschaden für Samsung geführt. Samsung musste weltweit rund 2,5 Millionen Geräte zurückrufen und hat dadurch vorübergehend rund 22 Milliarden US-Dollar an Marktwert verloren [Wagner 2016]. Darüber hinaus haben die US-Luftfahrtbehörde Federal Aviation Administration (FFA) sowie die Europäische Agentur für Flugsicherheit (EASA) in einer offiziellen Stellungnahme erklärt, dass Passagiere und Flugpersonal diese Geräte ausgeschaltet lassen müssen und sie an Bord des Flugzeugs nicht aufladen und nicht in ihr am Schalter aufgegebenes Gepäck geben dürfen. Die Passagiere wurden auch an die Notwendigkeit erinnert, die Kabinenbesatzung zu informieren, falls dieses Gerät beschädigt, heiß ist, Rauch erzeugt, verloren geht oder in die Sitzstruktur fällt [EASA 2016].

Dieses Beispiel sowie mehrmalige Verzögerungen in Luftfahrttraum-Projekten von Großunternehmen, bspw. Boeing und Airbus, aber auch berühmte Bauprojekte u. a. Stuttgart 21, BER Flughafen, beweisen die Aussage, dass eine Schätzung des Ressourcenaufwands in den frühen Phasen des Projektmanagements aufgrund vieler Unsicherheitsfaktoren eher ungenau ist [Lange 2015]. Bestehen aber nicht bereits Ansätze oder Methoden, die eine zielgerichtete Ressourcenplanung in der Projektplanungsphase für komplexe Fälle ermöglichen? Natürlich hat sich sowohl in der Wissenschaft als auch in der Technik eine Vielzahl klassischer, agiler sowie hybrider Ansätze des Projektmanagements (PM) etabliert, die sich u. a. mit einer zielgerichteten Ressourcenplanung auseinandersetzen. Nach [Kuster et al. 2019] muss jedoch unabhängig vom Projektvorgehen, ob klassisch oder agil, in den verschiedenen Projektphasen mit einer Aufwandtoleranz von -50%/+100% in frühen Phasen und $\pm 10\%$ in späteren Phasen gerechnet werden. Allerdings wird erwartet, dass diese Toleranz so gering wie möglich gehalten wird, um das Auftreten von ungeplanten Aufwänden zu vermeiden. Es ergibt sich also der Bedarf einer genaueren Schätzung bzw. Planung der Ressourcen in Frühphasen des Projektmanagements. An dieser Stelle ergeben sich wichtige Fragen: Wie könnte die Schätzung verbessert werden? Welche Faktoren beeinflussen die Schätzung in der Realität?

Eine Möglichkeit besteht in der Entwicklung eines systematischen Ansatzes für kompetenzbasierte Projektplanung, welcher die modernen IT-gestützten Lösungen **aus dem Bereich von künstlicher Intelligenz (KI)** zur Unterstützung nutzt [Buxmann 2019]. Eine Schnittstelle zwischen Linguistik und Informatik führt zu einer KI-Komponente namens Natural Language Processing (NLP), die es dem Rechner ermöglicht, einen natürlich sprachlichen Input (Stimme, Text, Bild, etc.) automatisch in ein standardisiertes Format zu konvertieren und es für spätere Aktionen wie z.B. Maschinelles Lernen und anschließend eine Entscheidungsfindung zu verwenden [Wess 2019].

Das NLP (Natural Language Processing) ist eine theoriegeleitete Kette von Rechentechniken zur automatischen Analyse und Repräsentation der menschlichen Sprache [Young et al. 2018]. Nach [Ghosh und Gunning 2019] ermöglicht NLP es uns, Mechanismen und Workflows zu entwickeln, mit denen die menschliche Sprache in leblose Objekte, wie Computer, eingespeist werden kann. Beispielsweise erfolgen die Differenzierung von Spam-eMails und deren anschließende Filterung in Spam-Ordner mittels NLP. Hierbei analysiert ein lebloses Objekt, also der eMail-Dienst, den Inhalt des Textes von eMails, versteht ihn und entscheidet dann nach bestimmten Faktoren, ob diese eMails als Spam markiert werden müssen oder nicht. Das bedeutet, dass ein Computer durch Unterstützung von NLP einen Text analysiert, verstanden und eine Entscheidung getroffen hat. Zusammengefasst ermöglicht NLP es Computern, eine Vielzahl von Rechenaufgaben im Zusammenhang mit natürlicher Sprache auf allen Ebenen durchzuführen, von der einfachen Textanalyse z.B. durch Parsing oder Part-of-Speech (POS)-Tagging bis hin zur Entwicklung von maschinellen Übersetzern oder Dialogsystemen [Young et al. 2018]. Aber warum ist es sinnvoll die KI-Komponenten, wie NLP, zur Unterstützung der Projektplanung zu nutzen?

Die Projektdokumentationen, die am Ende eines abgeschlossenen Projektes vorliegen, z. B. Projektabschlussbericht, Lessons learned, eMail-Verkehr, Gesprächsprotokolle von Sitzungen, etc. beinhalten nützliche Informationen über den Verlauf des Projektes vom Anfang bis Abschluss. Aus der Analyse dieser Informationen lassen sich Zusammenhänge von unterschiedlichen Indikatoren ableiten, die eine Rolle dabei spielen, ob ein Projekt sich verzögert. Aber wie könnte beispielsweise der gesamte eMail-Verkehr eines Projektes analysiert und wichtige Zusammenhänge der Informationen abgeleitet werden?

NLP eröffnet also die Möglichkeit, eine große Menge an unstrukturierten Daten, die in natürlicher Sprache verfasst wurden, automatisch und effizient nach bestimmten Faktoren bzw. Indikatoren zu interpretieren, in eine strukturierte Form zu bringen und zur Entscheidungshilfe zu nutzen [Wess 2019]. Um das Potenzial von NLP hinsichtlich einer kompetenzbasierten Projektplanung zu analysieren und die Problemstellung angehen und abdecken zu können wird folgende **Hypothese** dieser Forschungsarbeit zugrunde gelegt:

Mittels NLP und einem dazu gehörenden KI-Workflow werden nützliche Informationen so strukturiert und vorbereitet, dass sie im nächsten Projekt als Hilfsmittel für Entscheidungsträger in der Planungsphase des neuen Projektes dienen können.

Darüber hinaus wird untersucht, ob NLP die Projektplanung unabhängiger von erfahrungsbasierteren Entscheidungen machen kann. Dies ist besonders hilfreich, wenn z.B. ein neuer Mitarbeiter im Bereich Projektmanagement eingestellt wurde und er noch nicht über die Erfahrung oder das Know-how des Unternehmens verfügt. Durch das Erkennen der tatsächlichen Faktoren, die den Projekterfolg im Unternehmen beeinflussen, könnten somit auch neue Mitarbeiter noch genauere Entscheidungen treffen.

Die Abbildung 3 visualisiert die Zusammenhänge der unterschiedlichen Forschungsfelder, die bei der Hypothese dieser Arbeit eine wichtige Rolle spielen.

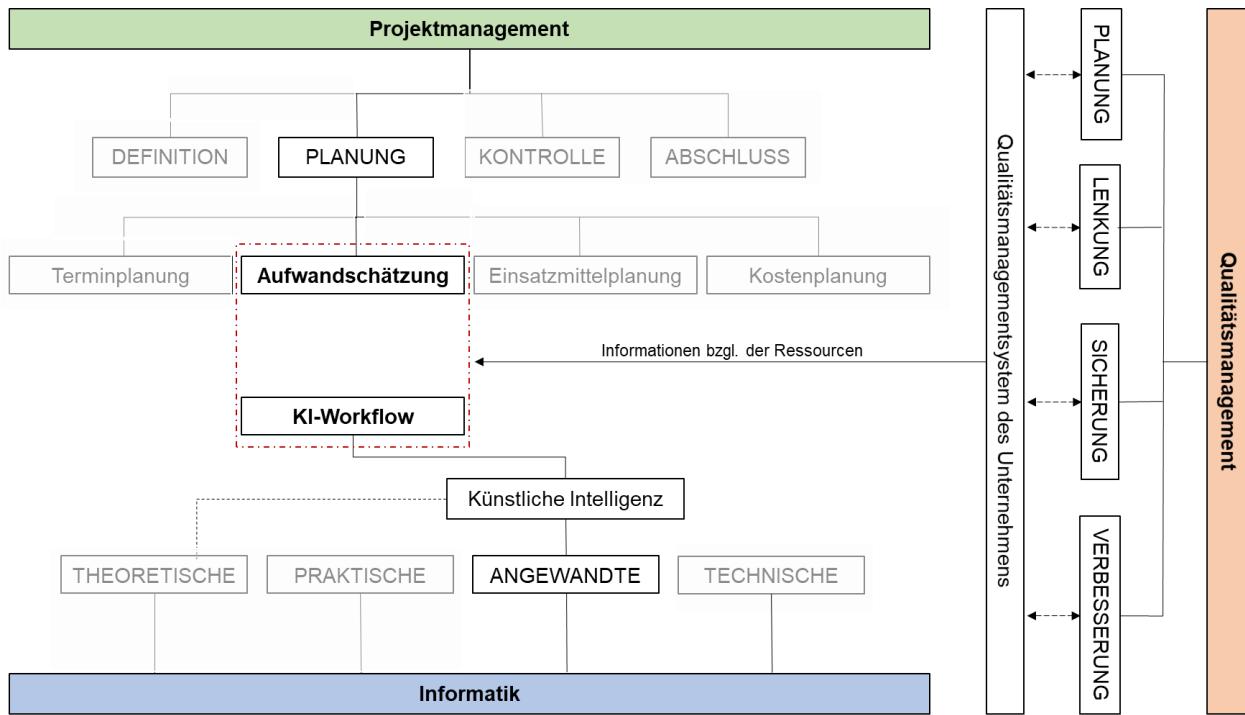


Abbildung 3: Forschungsfelder [Eigene Darstellung]

Eine Analyse der aktuellen Anwendungsgebiete von NLP zeigt jedoch, dass das in dieser Forschung angedachte Potenzial bei aktuellen Ansätzen im Bereich des Projektmanagements bzw. des kompetenzbasierten Ressourcenmanagements übersehen wird. Zu diesem Zweck wurde eine Literaturrecherche unter Berücksichtigung der folgenden **Eingrenzung** des Themas durchgeführt:

- Von den zum Projektmanagement gehörenden Prozessen wird der Teilprozess Ressourcenplanung betrachtet.
- Unter den verschiedenen Projektressourcen werden lediglich die Kompetenzen der Mitarbeiter als Ressource gesehen und analysiert.
- Das Validierungsbeispiel in dieser Forschung beschränken sich auf die Elektronikentwicklung in der Automobilbranche (Embedded Softwareentwicklung für das Auto-Steuengerät).

Problemfeld – Übersehen des Potenzials von NLP in Kombination des maschinellen Lernens in Form eines KI-Workflows als Unterstützung des Projektmanagements im Hinblick auf kompetenzbasierte Ressourcenplanung in Softwareentwicklungsprojekten

Zur Erhebung der aktuellen Anwendungsbereiche von NLP wurde eine systematische Analyse der Literatur sowie der Forschungsprojekten durchgeführt. Hierzu erfolgte im April 2024 eine Recherche in wissenschaftlichen Datenbanken, wie GEPRIS, Google Scholar, IEEE Xplore, ScienceDirect und SpringerLink im Betrachtungszeitraum von 2012 bis 2023. Diese Recherche grenzt sich unter Berücksichtigung der folgenden Punkte ein. Es wurden nur Quellen betrachtet, bei denen:

- klar erkennbar ist, dass NLP für Natural Language Processing steht und nicht für andere Begriffe, wie Neuro-Linguistisches Programmieren,
- in einem bestimmten Fachbereich zu den Nutzungen und Anwendungen von NLP geforscht wurde. Daher wurden die Studien, die sich nur mit NLP generell beschäftigen bzw. NLP-Techniken entwickeln vorerst nicht berücksichtigt.

Tabelle 1 stellt die Ergebnisse dieser Recherche bzw. die Anwendungsgebiete von NLP dar.

Tabelle 1: Anwendungsgebiete von NLP

Anwendungsgebiet	Anzahl der Studien	Anwendungsgebiet	Anzahl der Studien
Medizin	31	Biomedizin	4
Wissensmanagement	29	Informatik	3
Ingenieurwesen	20	Projektmanagement	3
Sprachwissenschaft	17	<i>Sonstiges</i>	9
Sozialwissenschaft	5		

Wie bereits aus den Ergebnissen der Recherche in Tabelle 1 zu erkennen ist, haben sich NLP-Ansätze sehr gut in der Medizin (inkl. verwandten Bereichen), Wissensmanagement sowie im Ingenieurwesen etabliert. Diese Auflistung umfasst jedoch **keine** NLP-gestützte kompetenzbasierte Ressourcenplanung in Softwareentwicklungsprojekten. Es mussten daher die aktuellen Methoden, die sich mit einer gezielten kompetenzbasierten Ressourcenplanung im Projektmanagement auseinandersetzen, analysiert und darüber hinaus sichergestellt werden, ob erste Forschungsprojekte hinsichtlich dieses Problemfelds in Erwägung gezogen wurden oder bereits abgeschlossen waren. Es wurden aus dieser Recherche 10 internationale und 2 nationale Forschungsprojekte identifiziert, die sich in diesem Problemfeld bewegen. Anschließend wurden die identifizierten Forschungen anhand der fünf **Themenfelder (TF)**, die aus dem o.g. Problemfeld abgeleitet werden können, evaluiert. Die abgeleiteten Themenfelder dienen dazu, dieses Forschungsvorhaben von den anderen Forschungen klarer abzugrenzen und die wissenschaftliche Lücke besser nachweisen zu können. Die Ergebnisse der Evaluation der ausgewählten Forschungen erfolgt nachfolgend in der Tabelle 2.

TF 1 (Informationsstruktur): Die Informationsbasis, auf welcher die benötigten Ressourcen für ein Projekt geplant werden, soll untersucht und identifiziert werden. Anschließend sollen die Erkenntnisse in Form einer Informationsstruktur wiedergegeben werden, damit der Algorithmus relevante Informationen für Ressourcenplanung lesen und verstehen kann. Darüber hinaus ist es notwendig herauszufinden, in welcher Form bzw. in welchen Dokumentationen diese Informationen zu finden sind. Dies bereitet den Weg für die Ableitung der Indikatoren, die die Ressourcenplanung in einem Projekt beeinflusse.

TF 2 (KI-Workflow): Es soll ein KI-Workflow mit entsprechenden Interaktionsoberflächen für den Projektleiter entwickelt werden, um als Werkzeug in der Planungsphase eines Projektes eingesetzt werden zu können.

TF 3 (Integration von NLP in Workflow): Es ist die Nutzung eines zu entwickelnden NLP-Systems zu untersuchen, um die Bearbeitung umfangreicher Projektdokumentationen zu erleichtern und vor allem zu ermöglichen.

TF 4 (Interpretation der NLP-Ergebnisse): Es ist zu überprüfen, wie die Erkenntnisse aus der NLP-Analyse mit bestehenden Unternehmensinformationen bezüglich der Ressourcen, u. a. Personalkompetenz, zu kombinieren ist. Dies ist hinsichtlich der Weiternutzung und Interpretation der NLP-Ergebnisse für eine kompetenzbasierte Ressourcenplanung wichtig.

TF 5 (Validierung des Workflows): Es ist zu untersuchen, inwieweit die Nutzung des KI-Workflows plausibel ist. Dazu werden zunächst die Anforderungen an den zu entwickelnden Workflow definiert, die Ergebnisse im Hinblick auf diese Anforderungen evaluiert und die Schwachstellen des KI-Workflows identifiziert.

Tabelle 2: Evaluation des Standes der Wissenschaft und Technik

Legende	•	<i>Themenfeld vollständig abgedeckt</i>		Themenfelder (TF)					
	◦	<i>Themenfeld zum Teil abgedeckt</i>							
	◦	<i>Themenfeld nicht abgedeckt</i>							
Themen				In welchem Bereich liegt das Hauptthema dieser Studie?	Informationsstruktur	KI-Workflow	Integration von NLP in Workflow	Interpretation der NLP-Ergebnisse	
Forschungen								Validierung des Workflows	
Nr.	National	Quelle	Jahr	Fokus	TF1	TF2	TF3	TF4	TF5
1	Analyse und Auswertung von gewichteten Anforderungen in technischen Spezifikationen	[Zwirn 2013]	2012	Requirements Eng. (Anforderungsanalyse)	◦	•	•	•	◦
2	Ontologiebasierte Abhängigkeitsanalyse im Projektlastenheft	[Zichler und Helke 2017]	2017	Requirements Eng. (AF-Zusammenhänge)	◦	•	•	◦	•
Nr.	International	Quelle	Jahr	Fokus	TF1	TF2	TF3	TF4	TF5
3	Improving agile requirements: The Quality User Story framework and tool	[Lucassen et al. 2016]	2016	Requirements Eng. (User Stories)	◦	•	◦	◦	•
4	Approximation of COSMIC functional size to support early effort estimation in Agile	[Hussain et al. 2013]	2013	Aufwandschätzung (Softwareentwicklung)	◦	•	•	•	◦

Nr.	International	Quelle	Jahr	Fokus	TF1	TF2	TF3	TF4	TF5
5	The NORMAP Methodology	[Farid 2012]	2012	Requirements Eng. (Nicht funktionale AFs)	○	●	●	○	●
6	NORMALIC: A visual tool for modeling Non-Functional Requirements in agile processes	[Farid und Mitropoulos 2012]	2012	Modellierung von nicht funktionaler Anforderung	○	●	●	●	●
7	Automatic Generation of Test Cases for Agile using Natural Language Processing	[Rane 2017]	2017	Erzeugen von SoftwareTest-Cases	○	●	●	○	●
8	Natural Language Processing and Machine Learning Methods for Software Development Effort Estimation	[Ionescu et al. 2017]	2017	Aufwandschätzung (Softwareentwicklung)	●	●	●	●	●
9	Requirements Complexity Definition and Classification using Natural Language Processing	[Sundararajan et al. 2018]	2018	Requirements Eng. (Anforderungsklassifizierung)	○	●	●	○	●
10	Simplified Agile Software Project Selection Model Using Natural Language Processing Based Upon Agile Team Skills	[Sharma et al. 2019]	2019	Kompetenzanalyse der agilen Teams	●	●	●	●	●
11	Support for traceability management of software artefacts using Natural Language Processing	[Arunthavanathan et al. 2016]	2016	Changemanagement (Softwareentwicklung)	○	●	●	●	●
12	Smart Project Management: Interactive Platform Using Natural Language Processing Technology	[Chen et al. 2021]	2021	Projektmanagement (Extrahieren der Schlüsselwörter)	●	●	●	○	●

Zusammenfassend besteht, unter Berücksichtigung

- der Kausalkette der Problematik (s. Abbildung 2), vor allem fehlerhafter Entscheidungen in der Projektplanungsphase,
- des Potenzials von NLP in Analyse und Vorbereitung der Informationen, und
- der Ergebnisse der Literaturrecherche hinsichtlich dieses Problemfelds,

ein interdisziplinärer Forschungsbedarf hinsichtlich der Implementierung von NLP in das Projektmanagement zur Optimierung der Ressourcenplanung in den frühen Phasen des Projektes. Damit soll eine Grundlage geschaffen werden, um nicht nur Entscheidungen im Hinblick auf eine kompetenzbasierte Ressourcenplanung zu beschleunigen, sondern auch, um Fehlentscheidungen in der Planungsphase des Projektmanagements zu vermeiden.

1.2 Zielbildung

Um die aufgezeigte wissenschaftliche Lücke abzudecken, wird auf Grundlage der Literatur das folgende Hauptziel abgeleitet:

Entwicklung eines KI-basierten Workflows zur Unterstützung der Projektleitung hinsichtlich einer kompetenzbasierten Ressourcenplanung von Industrieprojekten

Dieses Ziel soll den Projektmanager in die Lage versetzen, auf Grundlage einer systematischen Analyse der Dokumentation früherer Projekte des Unternehmens, frühzeitig und in kürzerer Zeit, fehlerfreie Entscheidungen zu treffen, um den Projekterfolg zu sichern. Dies ist jedoch nur möglich, wenn zum richtigen Zeitpunkt des Projekts geeignete Projektteams entsprechend ihrer Kompetenzen und Expertise eingesetzt werden.

Das Ziel dieser Forschung grenzt sich von anderen wissenschaftlichen Ansätzen im Rahmen der zuvor definierten Problemfelder ab. Das Forschungsziel ist jedoch noch sehr vielfältig und daher ist es notwendig, Teilziele auf Basis der zuvor genannten Themenfelder zu definieren. Anschließend werden zielgerichtete Schritte definiert, um die Teilziele abdecken zu können. Die folgenden 4 Teilziele wurden aus den 5 Themenfelder dieser Forschung abgeleitet:

- Teilziel 1 (aus dem Themenfeld 1):

Entwicklung einer Informationsstruktur zur Identifizierung der relevanten Projektinformationen hinsichtlich der Ressourcenplanung

Die unterschiedlichen Projektinformationen sind in verschiedenen Projektdokumentationen aufgeführt. Allerdings enthalten nicht alle Dokumentationen im Projektumfeld die Informationen, die für die Ressourcenplanung relevant sind (z.B. technische Zeichnungen). Deshalb wird zunächst ein Sammelpool von relevanten Projektdokumentation erstellt, dieressourcenbezogene Informationen beinhalten. Aber längst nicht alle Informationen, die in den relevanten Dokumenten enthalten sind, tragen zur Ressourcenplanung bei. Um die relevanten Informationen aus diesen Sammelpool der relevanten Projektdokumentationen zu finden und sie dann für die Weiterverarbeitung zur Verfügung zu stellen, sollte zusätzlich zu dem Sammelpool eine Informationsstruktur entwickelt werden, die genau identifiziert, welche Informationen überhaupt für eine Ressourcenplanung wichtig sind.

- Teilziel 2 (aus dem Themenfeld 2, 3):

Ableitung von Indikatoren (Labels) für eine kompetenzbasierte Ressourcenplanung im Projektmanagement, die einen Einfluss auf den Projekterfolg haben

Nicht alle Informationen im Zusammenhang mit der Ressourcenplanung spielen eine Rolle, wenn es darum geht, ob falsche Entscheidungen getroffen werden und dadurch ein Projekt verzögert wird oder scheitert. Aus diesem Grund sollten auf der Grundlage der in der Informationsstruktur aufgeführten Informationen Indikatoren abgeleitet werden, die einen direkten Bezug zum Projekterfolg haben und sich auf den Bereich der kompetenzbasierten Ressourcenplanung von Industrieprojekten

beziehen. Die Ableitung dieser Indikatoren (Labels) dient als Grundlage für die Untersuchung der Dokumentationen aller Projekte des Unternehmens. Darüber hinaus sollen die Wechselwirkungen und Relationen der abgeleiteten Labels untereinander sowie deren Zusammenhang mit dem Projektumfeld betrachtet bzw. analysiert werden. In dieser Hinsicht ist es hilfreich, wenn ein Projekt als soziotechnisches System verstanden wird. Dann ist es möglich, die Elemente des Systems und ihre Wechselbeziehungen innerhalb des Systems zu identifizieren und zu modellieren. Hierzu wird das Projekt auf Basis von eDeCoDe modelliert. Das eDeCoDe-Modell dient zu Beherrschung der Komplexität in sozio-technische Systemen sowie einem besseren Verständnis von Zusammenhängen und Relationen unterschiedlicher Systemelemente sowie deren Attribute und eignet sich gut zu dieser Forschung [Mistler 2021; Schlueter et al. 2018].

- Teilziel 3 (aus dem Themenfeld 2, 3):

Entwicklung eines geeigneten NLP-gestützten Verfahrens für die Untersuchung der Dokumentationen und Strukturierung der Informationen hinsichtlich der abgeleiteten Indikatoren

Eine manuelle Verarbeitung aller relevanten Informationen aus früheren Projekten des Unternehmens ist sehr zeitaufwendig und aufgrund des Umfangs und unterschiedlicher Formen fast unmöglich. Aus diesem Grund soll eine NLP-Pipeline eingerichtet werden, die die Dokumentationen von allen Projekten des Unternehmens untersucht und nach den zuvor abgeleiteten Labels kategorisiert. Auf diese Weise kann sichergestellt werden, inwieweit die im letzten Teilziel auf Basis der eDeCoDe-Logik modellierten Labels und deren Zusammenhänge tatsächlich eine Rolle im Entscheidungsfindungsprozess bezüglich einer kompetenzbasierten Ressourcenplanung spielen.

Teilziel 4 (aus dem Themenfeld 2, 4, 5):

Entwicklung eines KI-Workflows für die Interpretation der Ergebnisse vom der NLP-Analyse und deren Einbindung in der Ressourcenplanung des Projektmanagements

Nach der Analyse der Informationen durch das NLP-System sollten die Ergebnisse wieder zusammengeführt, interpretiert und in einer vernünftigen und von Menschen brauchbaren Form wiedergegeben werden. Es soll ein systematischer Workflow entwickelt werden, um die Ergebnisse der Analyse in verwertbares Wissen umzuwandeln und dann in einer geeigneten Form zu präsentieren. So kann der Projektmanager diese Erkenntnisse als Entscheidungsunterstützung nutzen, um eine schnelle und möglichst fehlerfreie Entscheidung über die Ressourcenplanung im neuen Projekt treffen zu können.

Zusammenfassend sieht das Konzept die KI-Komponenten zur Unterstützung der Projektplanung im Bereich der kompetenzbasierten Mitarbeiterbedarfe vor. Der Einsatz von KI-Komponenten, wie Natural Language Processing (NLP) in Zusammenhang von maschinellem Lernen, eröffnet die Möglichkeit einer umfassenden Analyse unstrukturierter Informationen. Die Informationen und vor allem die Erfahrungen aus der Durchführung eines Projektes sind sehr wertvoll für Entscheidungen im nächsten Projekt. In manchen Fällen geht dieses Know-how jedoch verloren, z.B. wenn der erfahrene Mitarbeiter nicht mehr zur Verfügung steht. Dieses Konzept reduziert insbesondere die Bedeutung

der benötigten Erfahrung für die Entscheidungsträger durch eine gezielte Anwendung von KI-Techniken im Hinblick auf die kompetenzbasierte Ressourcenplanung. Es ermöglicht auch, eine große Menge an Projektinformationen in sehr kurzer Zeit zu analysieren und so zu strukturieren, dass sie leicht in folgenden Projekten wiederverwendbar werden.

1.3 Herangehensweise

Um den Weg zur Erreichung der in Kapitel 1.2 genannten Teilziele zu realisieren, ist ein strukturierter Plan mit definierten Schritten zu entwickeln. Im Kapitel 1, der Einleitung, werden die Forschungsfragen, die Problemstellung und die Zielsetzung der Dissertation erläutert. Dadurch wird der Grundstein für die gesamte Arbeit gelegt und ein Überblick über die zu untersuchenden Themen gegeben. Im Kapitel 2 werden alle notwendigen Fachbegriffe und theoretischen Konzepte erklärt, die für das Verständnis der Dissertation erforderlich sind. Dieses Kapitel dient dazu, den Leser mit den Grundlagen vertraut zu machen, auf denen die weitere Arbeit aufbaut. Im Kapitel 3 wird der aktuelle Stand der Wissenschaft und Technik analysiert. Es wird eine umfassende Literaturrecherche durchgeführt und bestehende Lösungen, Methoden und Technologien, die im Zusammenhang mit dem Forschungsthema stehen, untersucht. Das Kapitel 4 bietet eine Zusammenfassung der bisherigen Erkenntnisse und definiert die Anforderungen an das zu entwickelnde Lösungskonzept. Dieses Kapitel dient als Übergang zwischen der theoretischen Aufarbeitung und der praktischen Entwicklung. Das Kapitel 5 ist das Hauptkapitel der Dissertation, in dem das Lösungskonzept entwickelt wird. Die Methodik und der Prozess der Konzeptentwicklung werden im Detail beschrieben und die erarbeiteten Lösungen vorgestellt. Im Kapitel 6 wird das Lösungskonzept gegenüber den zuvor definierten Anforderungen validiert. Tests und Experimente werden durchgeführt, um die Wirksamkeit und Effizienz des entwickelten Konzepts zu überprüfen. Im abschließenden Kapitel 0 werden die Ergebnisse der Dissertation diskutiert, ein Fazit gezogen und ein Ausblick auf mögliche zukünftige Forschungen und Anwendungen gegeben. Die wichtigsten Erkenntnisse werden zusammengefasst und der Erfolg der durchgeföhrten Untersuchungen bewertet. Der Verlauf aller Kapitel ist in Abbildung 4 dargestellt.

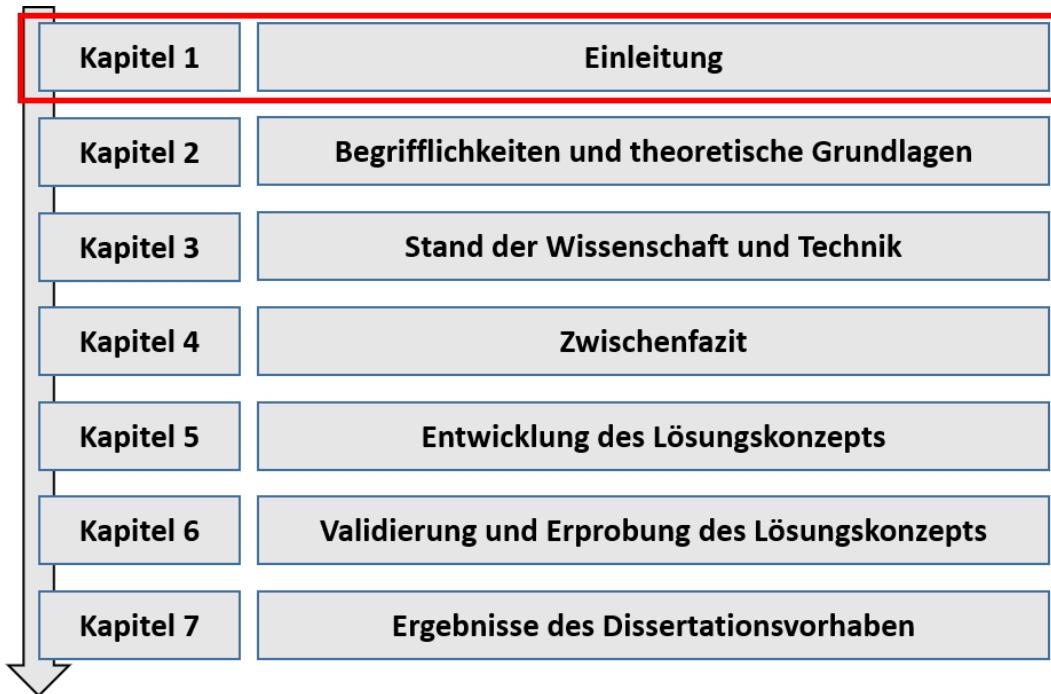
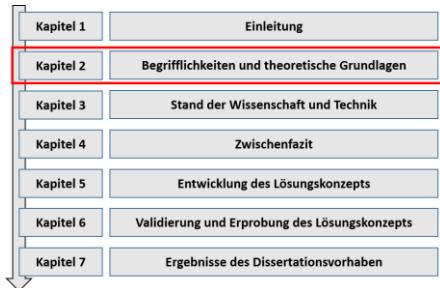


Abbildung 4: Verlauf von Dissertation [Eigene Darstellung]

Durch diese Struktur wird eine klare und logische Darstellung der Forschungsarbeit ermöglicht, die den Leser Schritt für Schritt durch den gesamten Forschungsprozess führt, von der Problemstellung bis hin zu den abschließenden Ergebnissen und Ausblicken.

2 Begrifflichkeiten und theoretische Grundlagen



Um ein gemeinsames Verständnis mit den zur Verfügung stehenden Informationen für die Entwicklung von KiProKo zu schaffen, ist es wichtig, relevante Begriffe und theoretische Grundlagen klar zu definieren und abzugrenzen. Nur so kann sichergestellt werden, dass die späteren Inhalte des zu entwickelnden Konzepts nachvollziehbar sind.

Zunächst wird das Konzept der künstlichen Intelligenz und seine verschiedenen Aspekte analysiert und beschrieben (s. Kapitel 2.1). Darunter werden die wichtigsten Begriffe wie Algorithmen, natürliche Sprachverarbeitung (NLP) und Workflow im Allgemeinen erläutert. Anschließend werden die Begriffe, die sich auf das Projektmanagement beziehen, einschließlich der Phasen des Projektmanagements, erläutert (s. Kapitel 2.2). Dabei wird der Fokus auf die erste Phase, die Projektplanung, gelegt und im Folgenden der Begriff Ressourcen und Ressourcenplanung in Bezug auf ein Industrieprojekt näher betrachtet. Anschließend wird der Begriff Kompetenz an sich aber auch im Kontext der Ressourcenplanung erklärt (s. Kapitel 2.3). Schließlich werden die Begriffe der modellbasierten Softwareentwicklung, wie das V-Modell und Automotiv-SPICE ausführlich beschrieben (s. Kapitel 2.4), da diese sich auf den Anwendungsbereich des Workflows beziehen. Erst dann werden Ansätze zur Beschreibung von Systemen kurz vorgestellt und schließlich wird einer ausgewählt, der für die Beschreibung soziotechnischer Systeme im Rahmen der Dissertation verwendet werden soll.

2.1 Künstliche Intelligenz

Künstliche Intelligenz wird derzeit auf vielen Ebenen diskutiert. Das Einzige, was heute neu ist, ist ihre breite Anwendung. Seit über 50 Jahren wird sie in einzelnen Unternehmensanwendungen entwickelt und erforscht [Teich 2020]. Bereits 1956 hat John McCarthy, Assistant Professor von Mathematik, den Begriff „Künstliche Intelligenz (KI)“ (Engl. Artificial Intelligence (AI)) in Hochschule Dartmouth, USA bei einer Konferenz vorgestellt. Sein Ziel war es, durch die genaue Beschreibung von jedem Aspekt des Lernens und andere Merkmale von Intelligenz einer Maschine zu ermöglichen, die Denkweise des Menschen nachmachen zu können [McCarthy et al. 2006].

Die Entwicklung der künstlichen Intelligenz hatte eigene Herausforderungen und die Technologien, die dahinterstehen, haben sich in unterschiedlichem Tempo entwickelt. Die Abbildung 5 stellt eine Übersicht über die wichtigen Meilensteine der Entwicklung von KI dar.

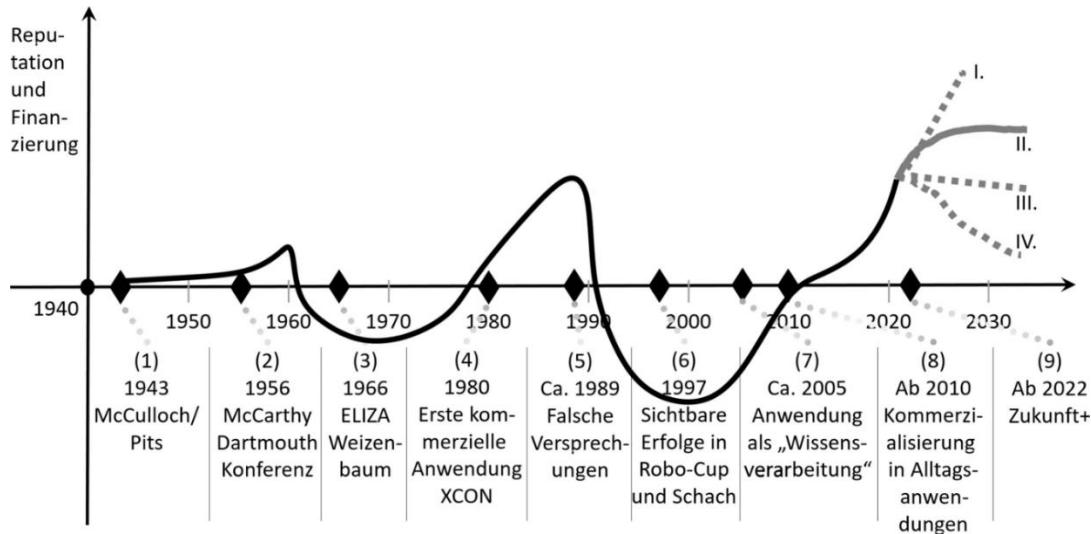


Abbildung 5: Entwicklung der künstlichen Intelligenz [Teich 2020, S. 277]

Die wichtige Meilensteine in der Historie von KI sind die erste Idee (1943 bis 1989), die Pause aufgrund enttäuschter Erwartungen (1989 bis 2010) und das Comeback (2010 bis heute) [Teich 2020]. Das Forschungsfeld befasst sich mit der Aufgabe, Computer zu nutzen, um die menschliche Intelligenz zu verstehen [Weber 2020a]. Nach der Literaturrecherche hat sich herausgestellt, dass es eine einheitliche Definition von KI noch nicht gibt, und jeder definiert künstliche Intelligenz so, wie er es für richtig hält. Jedoch sind im Laufe der Zeit verschiedenen Definitionen hinzugekommen [Bünte 2020]. Rich und Knight definieren KI als „Das Studium der Frage, wie man Computer dazu bringen kann, Dinge zu tun, die Menschen im Moment besser können. [Rich und Knight 1991, S. 3]“. Nach Hartmann ist die Kernaussage von KI folgendes: „Im Allgemeinen bezeichnet künstliche Intelligenz den Versuch, menschenähnliche Intelligenz zu emulieren, d.h. einen Computer so zu bauen oder zu programmieren, dass er selbstständig Probleme bearbeiten kann. [Hartmann 2018, S. 231]“.

Nach der gemeinsamen Definition des Bundesverbands Informationswirtschaft, Telekommunikation und neue Medien (Bitkom) und des Deutschen Forschungszentrums für Künstliche Intelligenz (DFKI) eine Künstliche Intelligenz „beschreibt Anwendungen der Informatik, deren Ziel es ist, intelligentes Verhalten zu zeigen. Dazu sind bestimmte Kernfähigkeiten in unterschiedlichen Anteilen notwendig: Wahrnehmen, Verstehen, Handeln und Lernen. [Bitkom e.V. 2017, S. 29]“. Der Amazon Konzern, der heute ein wichtiger Treiber und Akteur der Forschung im Bereich der KI ist, legt folgendes als Definition fest: „Künstliche Intelligenz (KI) ist das Gebiet der Informatik, das sich mit dem Erwerb kognitiver Fähigkeiten beschäftigt, die normalerweise der menschlichen Intelligenz zugeschrieben werden. Dazu gehören Lernen, Problemlösung und Mustererkennung [Amazon 2023].“. Wie bereits erwähnt sind die Definitionen von KI ziemlich uneinheitlich. Eine genaue Eingrenzung der Definitionen daher nicht möglich. Jedoch ist es möglich eine klare Abgrenzung der Teilbereiche von KI auf Basis der Anwendungsmöglichkeiten abzuleiten. Die Abbildung 6 visualisiert die verschiedenen Zweige der künstlichen Intelligenz.

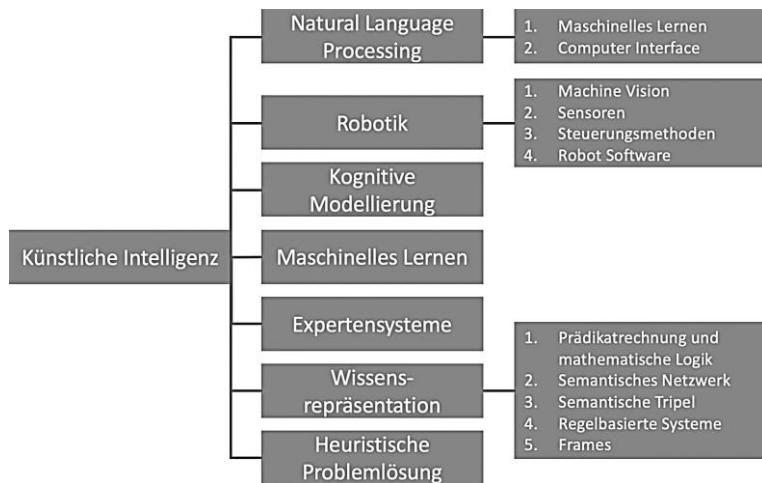


Abbildung 6: Verschiedenen Bereiche der künstlichen Intelligenz in Anlehnung an [Russell und Norvig 2010]

Der Schwerpunkt dieser Arbeit liegt auf der Verarbeitung natürlicher Sprache (NLP). Die Verarbeitung natürlicher Sprache befasst sich allgemein mit der Verarbeitung menschlicher/natürlicher Sprachen und der Analyse der Interaktionen zwischen menschlicher Sprache und Computern, insbesondere mit der Verarbeitung und Analyse großer Mengen natürlicher Sprachdaten [Weber 2020b]. Bevor in das Konzept von NLP eingetaucht und dieses analysiert wird, sollte bekannt sein, was ein Algorithmus ist, da NLP nichts anderes als eine Sammlung von Algorithmen darstellt, die unterschiedliche rechnerische Aufgaben erledigen.

2.1.1 Algorithmen

Es gibt eine immense Anzahl und eine große Vielfalt von Algorithmen aus Wissenschaft und Technik, wenn von Algorithmen gesprochen wird. Dabei können evolutionäre Standardalgorithmen, wie memetische oder kulturelle Algorithmen [Weicker 2015], sowie Algorithmen für iterative oder rekursive Auswertungen [Rimscha 2017] genannt werden. [Vöcking et al. 2008] veranschaulicht die Vielfalt der verfügbaren Algorithmen durch eine umfassende Liste von Algorithmen, die verschiedene Themenfelder wie „Rechnen, Verschlüsseln und Codieren“, „Optimieren“, „Planen, strategisches Handeln und Computersimulationen“ sowie „Suchen und Sortieren“ abdeckt. Ebenfalls gibt es in [San-tinbánez Koref 2015] eine Zusammenstellung von Algorithmen aus verschiedenen Bereichen wie der Informatik oder dem Maschinenbau, wie z.B. Algorithmen zur Protokollvalidierung im Kommunikationsbereich oder zur Erkennung von Designfehlern in fehlertoleranten Netzwerken. Die Tabelle 3 zeigt die unterschiedlichen Anwendungsbereiche der Algorithmen.

Tabelle 3: Anwendungsbereiche der Algorithmen nach [Santibáñez Koref 2015, S. 5]

Anwendung	Beispiele (Anzahl)	Anwendung	Beispiele (Anzahl)
Akustik	1	Bauwesen	5
Biologie	2	Chemie	2
Energietechnik	2	Informatik	8
Logistik	7	Luft- und Raumfahrt	6
Maschinenbau	2	Medizin	5
Musik	2	Optik	2
Robotik	2	Tanz	2
Verfahrenstechnik	1	Wirtschaftswissenschaften	5

Die Vielzahl der Beispiele, sowohl aus den vorliegenden Beispielen als auch aus Tabelle 3, zeigt deutlich, wie groß die Spannbreite der Algorithmen ist und wie unterschiedlich das Verständnis ihrer Definition sein kann. Die Unterschiede in den Definitionen lassen sich auf mehrere Faktoren zurückführen, z. B. auf den Zweck, für den ein Algorithmus entwickelt wurde, die verwendeten Methoden und Technologien sowie die zugrunde liegenden theoretischen Konzepte. Es ist wichtig zu erkennen, dass ein und dasselbe Problem auf mehrere Arten gelöst werden kann und für jede Lösung ein anderer Algorithmus erforderlich ist. Außerdem treten Algorithmen oft in verschiedenen Formen oder Varianten auf, je nach Ziel oder Problem, das sie lösen sollen. Dementsprechend gibt es keine einheitliche Definition von Algorithmen, sondern vielmehr eine Vielzahl von Definitionen, die von einer Vielzahl von Faktoren beeinflusst werden.

2.1.2 Natural Language Processing (NLP)

Natürliche Sprachverarbeitung (Natural Language Processing, NLP) ist der Bereich der digitalen Textverarbeitung, der sich in zwei Hauptkategorien unterteilt: Natural Language Understanding (NLU) und Natural Language Generation (NLG). Während NLU darauf abzielt, natürliche Sprache zu verstehen, geht es bei NLG darum, natürliche Sprache zu erzeugen [Carstensen 2010].

NLU nutzt Grammatik und Kontext, um Texte zu analysieren. Dabei werden Wortarten, Plural- und Singularformen, Marker sowie Orts- und Personennamen berücksichtigt. Das Ziel von NLG hingegen ist es, Texte zu produzieren. Durch die Verarbeitung maschineller Daten können Textzusammenfassungen, Übersetzungen und Berichte in Bereichen wie Wetterdienste, Sport und Finanzen leicht erstellt werden. Diese beiden Verarbeitungsmethoden bilden zusammen die Natürliche Sprachverarbeitung [Carstensen 2010]. Es können vier Anwendungsmethoden von NLP unterschieden werden:

- 1- **Speech to Text (STT):** Diese Methode wandelt gesprochene Worte in digitalen Text um. Dabei werden die von Mikrofonen aufgenommenen akustischen Signale in maschinenlesbaren Text umgewandelt. Dies kann für verschiedene Zwecke genutzt werden, z. B. für die automatische Tran-

skription von Sitzungen oder für die Spracheingabe in Textverarbeitungsprogrammen. Die Genauigkeit dieser Methode hängt von der Qualität der Audioaufnahme und der Genauigkeit des Spracherkennungsmodells ab [Kreutzer 2019].

- 2- **Speech to Speech (STS):** STS ist eine maschinelle Übersetzungsmethode, die mündliche Sprache in eine andere mündliche Sprache übersetzt. Diese Methode ist in der Lage, natürliche Sprache zu analysieren, zu übersetzen und zu synthetisieren [Kreutzer 2019]. Ein Beispiel für diese Methode ist das Online-Übersetzungsprogramm Google Translate oder DeepL.
- 3- **Text to Speech (TTS):** Bei dieser Methode wird geschriebener Text in gesprochene Sprache umgewandelt. Dazu wird eine computergenerierte Stimme erzeugt, die den Text laut vorliest. Dies ist besonders nützlich für Menschen mit Sehbehinderungen oder in Situationen, in denen es schwierig ist, Text zu lesen, z. B. beim Autofahren. Die Stimmen können oft personalisiert und an individuelle Vorlieben angepasst werden [Kreutzer 2019].
- 4- **Text to Text (TTT):** Bei der TTT werden Texte in einer Sprache analysiert und Informationen in einer anderen Sprache oder in einer anderen Form bereitgestellt. Dabei kann es sich zum Beispiel um maschinelle Übersetzungen, Textzusammenfassungen oder sogar automatische Berichte handeln. TTT ist besonders nützlich, um große Mengen an Informationen zu analysieren und zu verarbeiten [Kreutzer 2019].

Parallel zu den Anwendungsmethoden werden in der Literatur im Zusammenhang mit NLP fünf Prozessschritte beschrieben:

- 1- **Tokenisierung:** Unter Tokenisierung versteht man die Aufteilung eines Textes in Wörter, Phrasen oder Symbole, die als „Token“ bezeichnet werden. Die Tokenisierung ist ein wichtiger erster Schritt im NLP, da sie es ermöglicht, den Text in eine maschinenlesbare Form zu bringen und die Grundlage für die nachfolgenden Schritte der Sprachverarbeitung zu legen. Die Tokenisierung kann sowohl auf der Wort- als auch auf der Zeichenebene durchgeführt werden. Besteht ein Satz aus mehreren Wörtern, so wird dieser Satz in alle einzelnen Wörter getrennt. Ein einfaches Beispiel für die Tokenisierung ist die Aufteilung des Satzes „Die Katze sitzt auf der Matte“ in die einzelnen Wörter „Die“, „Katze“, „sitzt“, „auf“, „die“ und „Matte“ [Haarmeier 2021].
- 2- **Morphologische Verarbeitung:** Die morphologische Verarbeitung bezieht sich auf die Analyse der Struktur von Wörtern, um ihre Bedeutung und Funktion in einem Satz zu bestimmen. Dieser Schritt umfasst die Bestimmung von Wortstämmen, Wurzeln, Affixen (Präfixen und Suffixen) und Flexionen, um die Grammatik des Textes zu verstehen. In vielen Sprachen haben die Wörter verschiedene Formen, um ihre Funktion im Satz zu kennzeichnen (z. B. Singular und Plural, Verbformen usw.). Die morphologische Verarbeitung ist daher besonders wichtig für die korrekte Analyse von Texten. Ein Beispiel für morphologische Verarbeitung ist die Analyse des Wortes „laufen“. Die morphologische Verarbeitung würde erkennen, dass es sich um die Infinitivform des Verbs „laufen“ handelt, die aus der Basis „lauf-“, und der Endung „-en“ besteht [Kreutzer 2019].
- 3- **Syntaktische Verarbeitung:** Die syntaktische Verarbeitung bezieht sich auf die Analyse der Struktur von Sätzen und Phrasen und ihrer Beziehungen zueinander. Dieser Schritt umfasst die Bestimmung von Subjekten, Prädikaten, Objekten und anderen syntaktischen Elementen sowie die Analyse von Syntaxbäumen. Die syntaktische Verarbeitung ist ein wichtiger Schritt für die Analyse der Bedeutung von Texten und für die Umwandlung von Texten in strukturierte Daten. Ein Beispiel für syntaktische Verarbeitung ist der Satz „Der Hund beißt den Mann“. Hier würde die syntaktische

Verarbeitung erkennen, dass „Hund“ das Subjekt des Satzes ist, „beißt“ das Prädikat und „Mann“ das Objekt [Carstensen 2010].

- 4- **Semantische Verarbeitung:** Die semantische Verarbeitung bezieht sich auf die Analyse der Bedeutung von Wörtern, Sätzen und Texten. Dieser Schritt umfasst die Identifizierung von Entitäten (z. B. Personen, Orte, Organisationen), die Erkennung von Beziehungen zwischen Entitäten, die Erkennung von Ereignissen sowie die Erkennung von Kontext und Absicht. Die semantische Verarbeitung ist ein wichtiger Schritt für die automatische Klassifizierung von Texten und die Extraktion von Informationen aus unstrukturierten Daten. Ein Beispiel für semantische Verarbeitung ist der Satz „Ich habe ein neues Telefon gekauft, weil das alte kaputt war.“ Hier würde die semantische Verarbeitung verstehen, dass das neue Telefon gekauft wurde, um das alte zu ersetzen, das nicht mehr funktionierte [Kao 2007].
- 5- **Pragmatische Analyse:** Die pragmatische Analyse bezieht sich auf die Analyse der Bedeutung von Texten im Kontext und auf der Grundlage von Hintergrundwissen und Vorwissen. Dieser Schritt umfasst das Erkennen von Ironie, Sarkasmus, Metaphern und Anspielungen sowie die Berücksichtigung von kulturellen Unterschieden und Hintergrundwissen. Die pragmatische Analyse ist ein wichtiger Schritt für die Interpretation von Texten und die Erstellung von Texten, die für Menschen verständlich und sinnvoll sind. Ein Beispiel für eine pragmatische Analyse ist der Satz „Ich habe kein Geld.“. Ohne Kontext könnte dieser Satz bedeuten, dass die Person buchstäblich kein Geld hat. Wenn er jedoch im Zusammenhang mit einer Diskussion über eine geplante Anschaffung oder eine finanzielle Situation verwendet wird, könnte der Satz bedeuten, dass die Person nicht genug Geld für einen bestimmten Zweck hat [Kao 2007].

Diese fünf Prozessschritte sind ein wichtiger Bestandteil des NLP und ermöglichen es, die menschliche Sprache zu analysieren, zu interpretieren und zu verstehen. Dabei werden sowohl syntaktische als auch semantische Zusammenhänge berücksichtigt, um eine möglichst genaue und umfassende Analyse zu ermöglichen. Ein effektives NLP-System erfordert daher einen strukturierten und systematischen Ansatz, um die verschiedenen Aufgaben effizient und zuverlässig zu bewältigen. Hier kommt der Begriff „Workflow“ ins Spiel, der eine wesentliche Rolle bei der Organisation und Durchführung der Arbeitsschritte spielt.

2.1.3 Workflow

Der Ausdruck „Workflow“ stammt aus den USA und bezieht sich auf die Sequenz von Arbeitsschritten in Unternehmen. Im Folgenden werden verschiedene Definitionen des Workflow-Begriffs, wie sie in der einschlägigen Literatur zu finden sind, vorgestellt und ihre Gemeinsamkeiten sowie Unterschiede zum Begriff „Geschäftsprozess“ erläutert [Galler 1997].

Ein Workflow ist „ein System, dessen Elemente Aktivitäten sind, die durch eine Auslösebeziehung miteinander verbunden sind und durch externe Ereignisse ausgelöst werden [Joosten et al. 1994, S. 5]“. Vogler hat eine ähnliche Definition und beschreibt Workflow als miteinander verbundenen Aktivitäten, die von Aufgabenträger nach vorbestimmten Regeln ausgeführt werden [Vogler 1996]. Nach

[Heilmann 1998, S. 109] wird als Workflow „eine an den Unternehmenszielen orientierte, ganzheitliche und mehrstufige Aufgabenfolge bezeichnet, die von mindestens einem definierten Ereignis ausgelöst wird und ein oder mehrere definierte Ergebnisse erzeugt.“ Hierzu sollte zwischen einem Workflow und einem Prozess unterschiedet werden. Ein Prozess trägt unmittelbar zu den Leistungen einer Organisation bei (Primäre Prozesse) oder schafft die benötigten Voraussetzungen und Randbedingungen (sekundäre Prozesse), die von den primären Prozesse gefordert ist [Heilmann 1998]. Ein Workflow beginnt mit einem bestimmten Auslöser und endet mit einem eindeutigen Abschluss. Er besteht aus verschiedenen Schritten, die oft in Abhängigkeit von bestimmten Bedingungen ausgeführt werden können. Diese Schritte können entweder vollständig oder teilweise alternativ, parallel oder sequentiell ausgeführt werden. Die Schritte werden von Bearbeitern unter Verwendung von Ressourcen wie Dokumenten, Software und Daten ausgeführt [Galler 1997]. Obwohl die Begriffe Workflow und Prozess oft als Synonyme verwendet werden, ist es wichtig, zwischen ihnen zu unterscheiden. Während der Begriff Prozess die organisatorisch-strategische Sicht auf die Ablauforganisation umfasst, bezieht sich der Begriff Workflow auf die information-operative Sicht auf die Ablauforganisation [Galler 1997]. Die klare Strukturierung und Koordination von Abläufen sind nicht nur im Workflow essenziell, sondern spielen auch im Projektmanagement eine zentrale Rolle. Im Rahmen der Künstlichen Intelligenz (KI) und insbesondere des Natural Language Processing (NLP) wurden die grundlegenden Prozessschritte und Workflows behandelt, die notwendig sind, um komplexe sprachliche Aufgaben zu bewältigen. Mit diesen Grundlagen des Workflows im Bereich der KI im Hinterkopf, ist es nun wichtig, sich den Prinzipien des Projektmanagements zuzuwenden.

2.2 Projektmanagement

Die Definition von Projektmanagement in Anlehnung an die DIN 66901 ist die Gesamtheit der Managementaufgaben, -organisation, -techniken und -mittel für die Initiierung, Definition, Planung, Steuerung und Durchführung von Projekten [Steinbuch 2000]. Es beinhaltet die Koordination von Menschen und den optimalen Einsatz von Ressourcen zur Realisierung von Projektzielen [Alam 2020].

Im Zentrum der Definition nach [Steinbuch 2000] steht der Begriff „Führung“. Nach Wunderer und Grunwald wird Führung als zielgerichtete soziale Einflussnahme zur Bewältigung gemeinsamer Aufgaben verstanden [Heinrich 1998]. Ohne den Begriff Führung, aber mit der Betonung der Zielsetzung, beschreibt das amerikanische Project Management Institute (PMI) Projektmanagement wie folgt: „Projektmanagement ist die Anwendung von Wissen, Fähigkeiten, Werkzeugen und Techniken auf Projektaktivitäten, um die Projektanforderungen zu erfüllen.“ (frei übersetzt: „Projektmanagement ist die Anwendung von Wissen, Fähigkeiten, Werkzeugen und Techniken auf Projektaktivitäten, um Projektanforderungen zu erfüllen“) [Steinbuch 2000].

Darüber hinaus gibt es in der Literatur unzählige Definitionen des Begriffs, von denen sich viele auf die effektive und zielgerichtete Leistungserbringung konzentrieren, die durch das Projektmanagement

ermöglicht wird. Es liegt auf der Hand, dass eine präzise und umfassende Definition schwierig ist (im Vergleich zum Begriff „Projekt“). Um den Rahmen der vorliegenden Arbeit nicht zu sprengen, wird auf dieses Thema sowie auf die Kritik an einzelnen Definitionen verzichtet. Vereinfachend wird im weiteren Verlauf der Arbeit unter Projektmanagement alle Strukturen, Prozesse, Verfahrensweisen und Hilfsmittel, die den Projekterfolg sicherstellen verstanden [Litke 2005].

Je nach den Aufgaben des Projektmanagements können sich folgende Nutzen aus dem Einsatz des Projektmanagements ergeben. Der Nutzen wird in drei Projektkategorien unterteilt [Patzak und Rattay 2018]:

1- Nutzen des Projektmanagements bei **Einzelprojekten** wie z.B. Auftragsabwicklungsprojekte, Investitionsprojekte.

- Interne und externe Kundenorientierung,
- Früherkennung von Konfliktpotentialen durch aktive Umfeldanalyse,
- Konsequente Aufgabenerledigung auf Basis von Leistungsvereinbarungen
- Effizienter Ressourceneinsatz,
- Koordination durch aktive Teamführung,
- Termineinhaltung durch Leistungsvereinbarung,
- Identifikation der Mitarbeiter mit dem Projekt.

2- Nutzen des Projektmanagements bei **Produkt- und Entwicklungsprojekten** wie z.B. Markterweiterung:

- Effizienter Ressourceneinsatz aufgrund stufenweisen Vorgangs,
- Effiziente Abwicklung aufgrund festgelegter Meilensteine und Zielvorgaben,
- Durch unterschiedliche Sichtweisen, welche ins Team integriert werden, entstehen ganzheitliche Lösungen,
- Einhaltung der Ziele und Termine durch Leistungsvereinbarungen,
- Balance zwischen Weiterentwicklung der Mitarbeiter hinsichtlich Veränderungen und konkreten Vorgehensmodellen (Arbeitspakete, Meilensteine).

3- Nutzen des Projektmanagements bei **Interne Kundenorientierung, aufgrund Fokussierung interner Unternehmensprozesse:**

- Konkrete Zielformulierung in einem Projektstartworkshop,
- Effiziente Teamarbeit durch professionelle Kommunikationskultur,
- Ganzheitliches Denken,
- Sicherung der Akzeptanz für Veränderungen, durch Integration aller Beteiligten (Team-Projektorganisation).

Im weiteren Verlauf der Arbeit wird das Projektmanagement für Produkt- und Entwicklungsprojekte fokussiert. Dieses bringt für Unternehmen ebenfalls Vorteile, wie z.B. die Steigerung der Motivation der Mitarbeiter durch Teamarbeit und durch neue Karrierechancen, Förderung der Verantwortungsübernahme der Teammitglieder, effizienter Ressourceneinsatz durch bestmögliche Nutzung von Know-how und eine effiziente Umsetzung von Zielen mit sich [Patzak und Rattay 2018]. Die konkretere Darlegung der einzelnen Phasen des Projektmanagements wird im folgenden Kapitel aufgezeigt.

2.2.1 Projektmanagements, Phasen und Aufgaben

Die verschiedenen Phasen, die ein erfolgreiches Projekt durchlaufen muss, sind wie folgt: Initialisierung (Konzeption), Definition, Planung, Durchführung (Steuerung), Kontrolle und Projektabschluss [DIN 69901-5: 2009]. Tabelle 4 gibt einen Überblick über die wichtigsten Inhalte der einzelnen Phasen.

Tabelle 4: Die Phasen des Projektmanagementzyklus in Anlehnung an [DIN 69901-5: 2009]

Projektphase	Ziele	Aufgaben
Initialisierung	Die Steuerungsgruppe formuliert den Projektauftrag und ernennt die Projektleitung. Bildung der Projektgruppe.	<ul style="list-style-type: none"> • Problemanalyse • Zielklärung • Potenzialanalyse
Definition	Nach der offiziellen Startsituation werden die Projektdefinition und die Projektplanung formuliert, die Durchführbarkeit des Projekts geprüft und der Kosten- und Zeitrahmen bestimmt.	<ul style="list-style-type: none"> • Projektdefinition • Grobplanung • Durchführbarkeitsprüfung • Wirtschaftlichkeitsprüfung
Planung	Die Projektgruppe vervollständigt die Projektplanung, die Steuerungsgruppe genehmigt die Planung.	<ul style="list-style-type: none"> • Definition der Arbeitspakete • Lastenhefte • Feinplanung • Verantwortung klären • Risikoanalyse • Definition der Schnittstellen
Durchführung (Steuerung)	Die Projektgruppe bearbeitet konkrete Projektthemen. Steuerungsgruppe und Beschäftigte werden kontinuierlich informiert. Laufende Kontrolle der Projektdefinition.	<ul style="list-style-type: none"> • Information vor allem beim Erreichen eines Meilensteins • Abnahme der bisherigen Arbeitsergebnisse und Umsetzung in der Linieneorganisation (sofern dies zum Projektauftrag gehört) • Aktualisierung der Planung • Steuerung bei Abweichungen • Abarbeiten der Arbeitspakete
Kontrolle und Projektabschluss	Die Projektgruppe formuliert den Projektabschlussbericht, den die Steuerungsgruppe offiziell abnimmt. Reflexion der gemeinsamen Arbeit in der Projektabschlussitzung. Projektabschlussfeier; Projektgruppe löst sich auf.	<ul style="list-style-type: none"> • Abgleich mit dem eingangs gesteckten Zielen • Projektabschlussbericht • Informationsweitergabe an Kollegen und Kolleginnen, Vorgesetzte, etc. • Feiern des Endergebnisses

Im weiteren Verlauf der Dissertation wird nur noch die **Planungsphase** näher betrachtet. Die Projektplanung ist nach [DIN 69901-5: 2009, S. 11] definiert als „die Gesamtheit der Tätigkeiten und Verfahren zur formalen Planung eines Projektes“. In der Planungsphase werden auf der Grundlage der vorangegangenen Projektdefinition und den darin festgelegten Zielen und Aufwandsschätzungen die notwendigen Prozesse konkretisiert. Zu diesen Prozessen gehören u.a. die Planung von Arbeitspaketen und Arbeitsgängen, die Erstellung von Ablaufplänen, Terminplänen, Kosten- und Ressourcenplänen [DIN 69901-5: 2009]. In der folgenden Abbildung 7 sind die Phasen des Projektmanagements und der Fokus dieser Arbeit dargestellt.

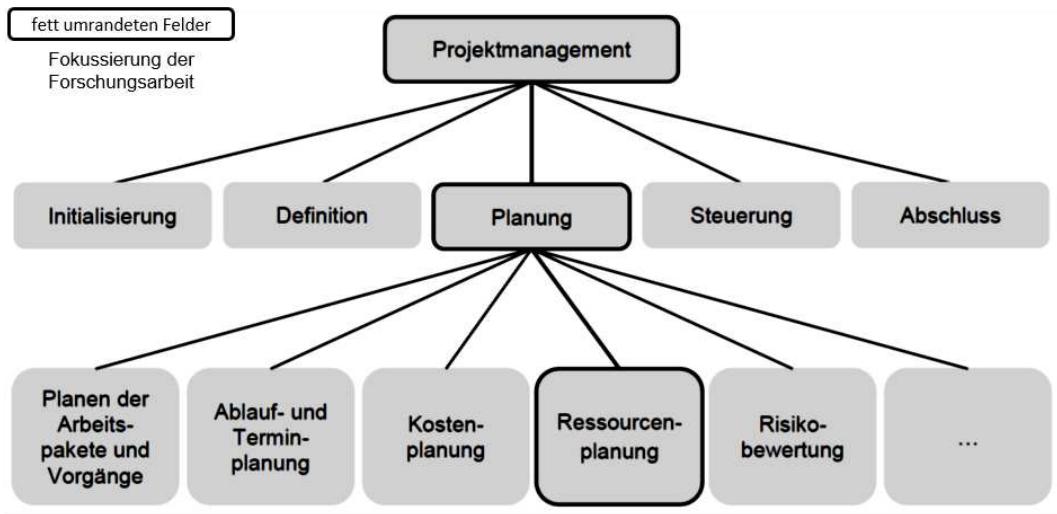


Abbildung 7: Projektmanagementphasen und Fokussierung der Forschungsarbeit [Eigene Darstellung]

Die im weiteren Verlauf fokussierte Ressourcenplanung wird im Folgenden detailliert beschrieben.

2.2.2 Ressourcenplanung

Die Ressourcenplanung umfasst die Bereitstellung, Steuerung und Kalkulation von Ressourcen zur Erfüllung eines Auftrags. Im Januar 2009 wurde der Begriff „Einsatzmittel“, der in der zurückgezogenen [DIN 69902: 1987] aufgeführt war, durch den Begriff „Ressource“ ersetzt. Da sich der neue Begriff noch nicht vollständig durchgesetzt hat, werden beide Begriffe in der Praxis und in dieser Forschung synonym verwendet. Die Definition des Ressourcenbegriffs in [DIN 69901-5: 2009] ist umfassender als in der Vorgängernorm und beschreibt eine „abgrenzbare [...] Einheit von personellen Ressourcen, finanziellen Ressourcen, materiellen Ressourcen, Informationen, natürlichen Ressourcen, Hilfs- und Unterstützungseinrichtungen, die zur Durchführung oder Förderung von Vorgängen, Arbeitspaketen oder Projekten eingesetzt werden können“ [DIN 69901-5: 2009, S. 17].

In fast allen Projektmanagement-Studien wird der richtige Einsatz von Ressourcen als zentraler Erfolgsfaktor für erfolgreiche Projekte genannt [Alam 2020]. Ein effektives Ressourcenmanagement ist daher ein wichtiger Bestandteil erfolgreicher Projektarbeit und bietet zahlreiche Erfolgspotenziale für Unternehmen [Schott und Campana 2005].

Die Ressourcenplanung eines Unternehmens ist nicht nur Teil der Planungsphase eines einzelnen Projekts, sondern muss auch in den Kontext der wiederkehrenden Linienaufgaben der Organisation integriert werden [Sterrer 2014]. Dabei werden sowohl die in den Projekten benötigten Ressourcen als auch die in der Linie verfügbaren Ressourcen auf der Ebene des Projektportfolios aufgelistet und verglichen. Aus der Managementperspektive geht es beim Portfoliomanagement also darum, die richtigen Projekte zur richtigen Zeit im richtigen Umfeld durchzuführen [Sterrer 2014]. Abbildung 8 veranschaulicht diesen Zusammenhang grafisch.

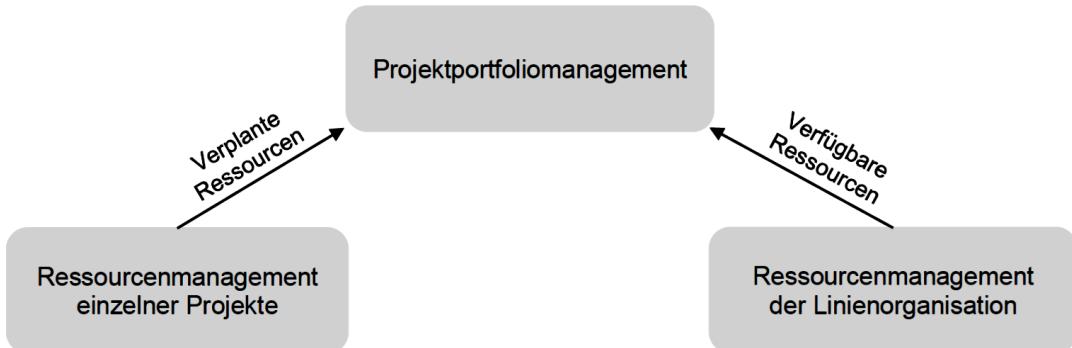


Abbildung 8: Zusammenhang zwischen Linie und Projekt im unternehmensweiten Ressourcenmanagement in Anlehnung an [Sterrer 2014]

Bei einem effektiven Portfoliomanagement im Projektmanagement geht es darum, den verschiedenen Projekten die entsprechenden Ressourcen zuzuweisen. Zu diesem Zweck wird zu Beginn eine Bedarfssplanung durchgeführt, um die notwendigen Ressourcen für die einzelnen Projektkomponenten zu ermitteln. Anschließend werden diese Bedarfe mit den verfügbaren Ressourcen abgeglichen und für den Betrachtungszeitraum zugeteilt, um Kapazitätsengpässe frühzeitig zu erkennen und entsprechende Maßnahmen zu ergreifen [Wagner und Alter 2016]. Eine effiziente Ressourcenallokation beschleunigt und verbessert die Projektdurchführung, verkürzt die Time-to-Market und fördert eine gute Positionierung in wettbewerbsintensiven Märkten [Wagner und Alter 2016, S. 114]. Ein gutes Ressourcenmanagement trägt nicht nur zum Erfolg des Unternehmens bei, sondern verbessert auch die Situation der Mitarbeiter, indem es die Überlastung minimiert. Eine transparente Ressourcenzuteilung ermöglicht den Mitarbeitern eine projektübergreifende Einsatzplanung und führt zu einer optimalen Auslastung der Mitarbeiter, was wiederum die Zufriedenheit und Motivation der Mitarbeiter erhöht und sich positiv auf die Arbeitsleistung auswirkt. Wichtig ist, dass für ein erfolgreiches Ressourcenmanagement alle notwendigen Informationen transparent dargestellt werden müssen. Dies erfordert umfassende Informationen über die durchzuführenden Projekte, die benötigten Ressourcen und deren Verfügbarkeit [Schott und Campana 2005]. Während im Projektmanagement der Fokus auf der optimalen Verwaltung von personellen, zeitlichen und finanziellen Ressourcen liegt, wird in dieser Arbeit der Schwerpunkt auf die Betrachtung rein personeller Ressourcen gelegt. Zeit und Geld werden als "Rand"-Informationen betrachtet, jedoch nicht im Detail analysiert, um den Rahmen der Dissertation nicht zu sprengen.

Im Zusammenhang mit der Verwaltung personeller Ressourcen spielt die Kompetenz der Personen bzw. Mitarbeiter eine zentrale Rolle. Kompetenzen bestimmen maßgeblich die Fähigkeit eines Teams oder eines Einzelnen, spezifische Aufgaben erfolgreich zu bewältigen und die gesetzten Projektziele zu erreichen [Nieke 2002]. Daher ist es essenziell, im nächsten Kapitel die verschiedenen Arten von Kompetenzen und deren Bedeutung für den Erfolg im Projektmanagement näher zu betrachten.

2.3 Kompetenz und Kompetenzarten

Seit den 1990er Jahren hat der Begriff „Kompetenz“ in der Managementliteratur eine führende Rolle eingenommen. In der Fachliteratur zur Personalentwicklung werden bestimmte Fähigkeiten als in hohem Maße übertragbar und universell einsetzbar bezeichnet, die in den meisten Berufen unabhängig von spezifischen Arbeitsaufgaben benötigt werden. Trotz dieser zentralen Rolle herrscht jedoch aufgrund der Vermischung verschiedener Konzepte, der uneinheitlichen Verwendung und der kulturellen Unterschiede in der Personalentwicklung und Berufsbildung erhebliche Verwirrung über den Begriff. In diesem Abschnitt wird das Konzept der Kompetenz im Vergleich zu anderen Begriffen wie Wissen und Fertigkeiten definiert, wobei sich die Definition auf die kontinentaleuropäische Geschichte der Verwendung von Kompetenz beschränkt. Dabei wird insbesondere auf den deutschen und den französischen Ansatz eingegangen, die auch von den meisten anderen europäischen Ländern verfolgt werden. Es gibt jedoch auch Ansätze in den Vereinigten Staaten und im Vereinigten Königreich [Nieke 2002].

Der Kompetenzbegriff war im deutschen dualen System der beruflichen Bildung immer implizit vorhanden. Der Schwerpunkt lag auf der Definition der Lerninhalte, die für die Beherrschung eines Berufs erforderlich sind, und nicht auf den Arbeitsergebnissen. In den 1970er Jahren führte Mertens das Konzept der Schlüsselqualifikationen ein. Schlüsselqualifikationen, oder später Schlüsselkompetenzen, umfassen drei Arten von Kompetenzen: fachliche, persönliche und soziale [Mertens 1974]. Abbildung 9 veranschaulicht die Hierarchie der Kompetenztypen und folgt den Definitionen der Kompetenztypen nach [Nieke 2002].

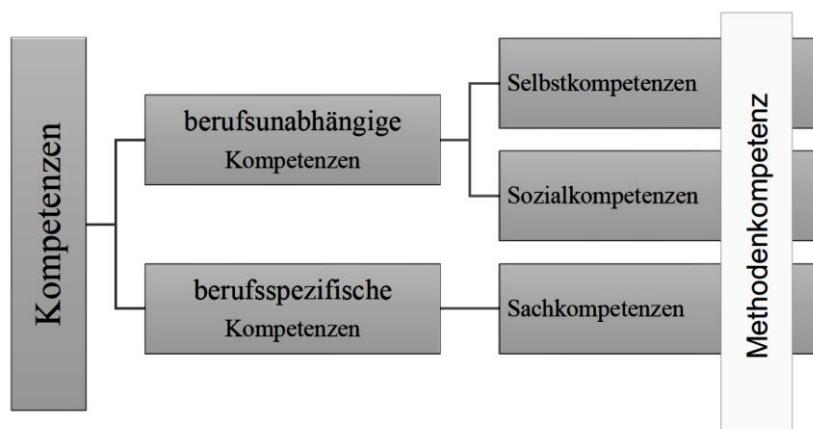


Abbildung 9: Hierarchie der Kompetenzarten in Anlehnung an [Nieke 2002]

Selbstkompetenz, auch personale Kompetenz genannt, beschreibt die Fähigkeit, die eigenen Entwicklungsmöglichkeiten, Anforderungen und Grenzen in Familie, Beruf und öffentlichem Leben als individuelle Persönlichkeit zu verstehen. Sie ist die Kompetenz, den eigenen Lebensentwurf zu gestalten und zu entwickeln. Sie umfasst persönliche Eigenschaften wie Selbstständigkeit, Kritikfähigkeit, Selbstvertrauen, Zuverlässigkeit, Verantwortungs- und Pflichtbewusstsein sowie berufliche und ethische Werte [Nieke 2002].

Soziale Kompetenz bezieht sich auf die Fähigkeit, Beziehungen aufzubauen und zu pflegen, den Nutzen von Beziehungen zu erkennen und bei Konflikten rational und verantwortungsbewusst zu handeln. Dazu gehört auch die Entwicklung von sozialer Verantwortung und Solidarität. Soziale Kompetenz umfasst sowohl funktionale als auch soziale Fähigkeiten und wird oft mit Adjektiven wie teamfähig, emotional intelligent, tolerant oder wertschätzend beschrieben [Nieke 2002].

Sachkompetenz hingegen, auch Fachkompetenz genannt, beschreibt die Fähigkeit, Aufgaben und Probleme auf der Basis von fachlichen Kenntnissen und Fertigkeiten zielgerichtet, sachgerecht, methodenorientiert und selbstständig zu lösen und die Ergebnisse zu beurteilen. Von einem Elektriker, der als kompetent gilt, wird zum Beispiel erwartet, dass er einen Stromkreis fehlerfrei anschließt, oder von einem Bauingenieur, dass er die Lastverteilung einer Brücke richtig berechnet [Nieke 2002].

Die Ausgewogenheit von Fach-, Selbst- und Sozialkompetenz ist die Voraussetzung für Methoden- oder Lernkompetenz. Einige Autoren sehen die Methodenkompetenz nicht mehr als eigenständige Kompetenz, sondern als Querschnitt und Bestandteil der drei Schlüsselkompetenzen. Die Methodenkompetenz erweitert die Fachkompetenz durch die Anwendung übergreifender Strategien und Prozesse des Erfindens und Problemlösens, während die Lernkompetenz sich auf das Konzept des Lernens bezieht [Nieke 2002].

Die Definition der verschiedenen Kompetenztypen dient dazu, den Kompetenzbegriff im Vergleich zu anderen, oft unscharf und synonym verwendeten Begriffen genauer zu definieren. Zu diesem Zweck wird auch das Verhältnis von Kompetenzen zu Wissen und Qualifikationen erklärt (s. Abbildung 10). Wissen bildet die Grundlage für Qualifikation, die wiederum die Grundlage für Kompetenz bildet. Ein höherer Wissensstand kann einer Person helfen, ihre Qualifikationen zu verbessern, was wiederum ihre Kompetenzen stärkt [Erpenbeck et al. 2017]. Wissen bezieht sich auf die Gesamtheit der Fakten, Regeln und Theorien, die eine Person besitzt. Es umfasst sowohl explizites Wissen, das durch formale Bildung erworben wurde, als auch implizites Wissen, das auf praktischen Erfahrungen beruht. Fertigkeiten sind spezifische Kenntnisse, die für eine bestimmte Aufgabe oder Handlung erforderlich sind. Einige erfordern körperliche Geschicklichkeit, andere kognitive Fähigkeiten wie Problemlösung und kritisches Denken [Nonaka und Takeuchi 1995].

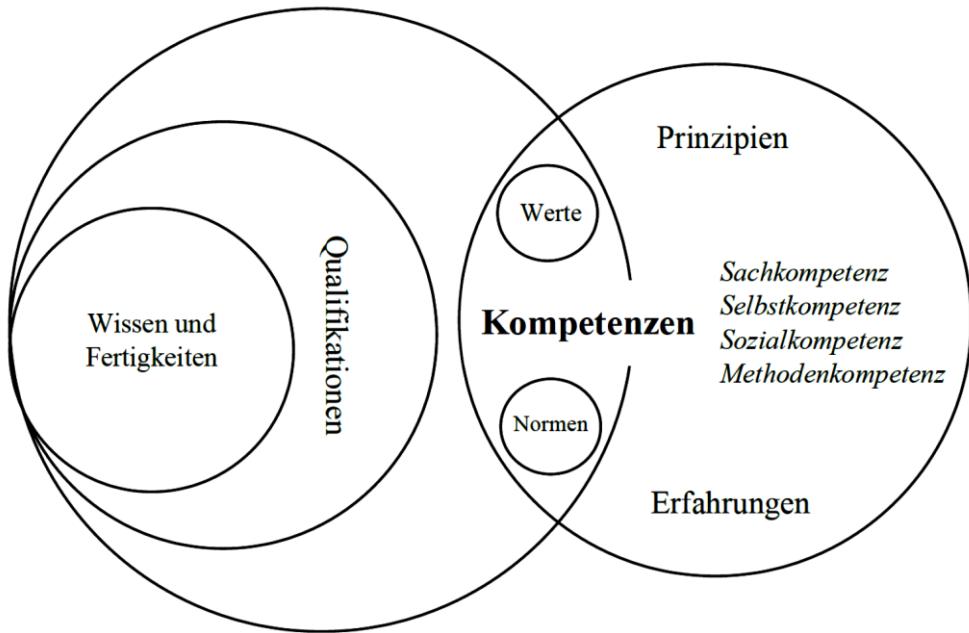


Abbildung 10: Einordnung des Kompetenzbegriffs in Relation zu Qualifikationen, Wissen und Fertigkeiten. Modifizierte Abbildung in Anlehnung an [Erpenbeck et al. 2017].

Wenn Wissen formell durch einen Nachweis zertifiziert wird, kann es als Fähigkeit betrachtet werden. Eine formale Qualifikation befähigt eine Person zur Ausübung einer bestimmten Tätigkeit oder Rolle. Der Erwerb einer Qualifikation kann auch ein Beweis für Selbstkompetenz sein, da er ein hohes Maß an Engagement, Ausdauer und Selbstregulierung erfordert [Li et al. 2018].

Persönliche Werte und Motivationen beeinflussen ebenfalls die Kompetenzen einer Person. Gesellschaftliche Normen und Regeln beeinflussen das Verhalten einer Person in einer Organisation und die Art und Weise, wie sie mit anderen interagiert. Ethische Grundsätze und Verhaltensweisen können zur Entwicklung beruflicher Kompetenz beitragen, indem sie einer Person helfen, ihre Arbeit mit größerer Sorgfalt und weniger Fehlern auszuführen [Erpenbeck et al. 2017].

Nach der Definition des Kompetenzbegriffs stellt sich aus der Perspektive eines strategisch agierenden Unternehmens die Frage, wie die Kompetenzen der eigenen Mitarbeiter ermittelt werden können. In der Arbeitspsychologie gibt es dazu einen eigenen Forschungsbereich, der wiederum eine Vielzahl von kommerziellen Angeboten hervorgebracht hat. Eine Antwort auf diese Frage ist die Erstellung eines Kompetenzmodells.

2.3.1 Kompetenzmodelle

Kompetenzmodelle lassen sich als umfassende Sammlungen der Fähigkeiten definieren, die von Mitarbeitern erwartet werden, um Leistungen zu erbringen und Probleme zu lösen. Diese Modelle ermöglichen eine klare, messbare und für alle nachvollziehbare Dokumentation der erforderlichen Kompetenzen. Durch spezifische Merkmale zur Beobachtung und Bewertung sowie durch die Festlegung von Kompetenzniveaus wird eine präzise und differenzierte Analyse ermöglicht. Das Hauptziel

besteht darin, das vorhandene Kompetenzkapital innerhalb eines Unternehmens vollständig zu erschließen und zu erweitern, welches von den Mitarbeitern vertreten wird [Erpenbeck et al. 2013]. Die Kompetenzziele lassen sich in zwei Formen strukturieren:

1. Ein unternehmensweites, generalisiertes Kompetenzmodell, das die grundlegenden Fähigkeiten umfasst, die von jedem Mitarbeiter erwartet werden.
2. Mehrere spezialisierte Kompetenzmodelle, die auf einer gemeinsamen Basis von grundlegend wichtigen Fähigkeiten aufbauen und für spezifische Bereiche, Funktionen oder Prozesse entwickelt werden.

Für die Identifikation und Dokumentation von Kompetenzen haben sich beobachtbare und interpretierbare Handlungsanker bewährt. Diese sind kurze, prägnante Beschreibungen von Handlungsanforderungen, die eine greifbare Grundlage für die Bewertung und Entwicklung von Kompetenzen bieten [Sauter und Staudt 2016b].

Kompetenzmodelle etablieren eine konsistente Perspektive auf die strategischen, organisatorischen und prozeduralen Bedingungen, die es Mitarbeitern und Führungskräften ermöglichen, ihre Fähigkeit zu selbstorganisiertem, kreativem Handeln und kontinuierlichem Kompetenzaufbau zu entwickeln. Diese Modelle lassen sich für verschiedene Geschäftsbereiche, Funktionen und Abläufe spezifizieren und fördern dadurch eine einheitliche Terminologie bezüglich der Unternehmensziele, -prozesse und -ergebnisse [Erpenbeck et al. 2013].

Heutzutage implementieren nahezu alle großen Konzerne eigens entwickelte Kompetenzmodelle, die tief in ihren Strategien zur Personalauswahl und -förderung verankert sind. In einer aktuellen Publikation, die Kompetenzmodelle großer Organisationen behandelt, präsentieren namhafte Unternehmen wie Airbus, die Bundesagentur für Arbeit, Daimler, Porsche, Audi, Siemens Healthcare, Deutsche Bahn, Deloitte, Deutsche Telekom, Esterhazy, Globus Baumärkte, Münchener Rückversicherung und Salzgitter AG ihre Ansätze [Erpenbeck et al. 2017]. Diese Modelle sind typischerweise Teil eines umfangreichen Kompetenzmanagementsystems. Darüber hinaus beginnen auch mittelständische und kleinere Unternehmen zunehmend, eigene Kompetenzmanagementsysteme zu entwickeln, um ihre Wettbewerbsfähigkeit und Innovationskraft zu stärken [Sauter und Staudt 2016b].

In der Fachliteratur werden vielfältige Kriterien für die Gestaltung eines Kompetenzmodells diskutiert. Eine Zusammenfassung der Anforderungen an einem Kompetenzmodell ist nachfolgend gegeben. D.h. ein effektives Kompetenzmodell:

- Ermöglicht die Fokussierung auf Schlüsselkompetenzen, die für die Realisierung der Unternehmensstrategie von entscheidender Bedeutung sind,
- Ist eng mit der Unternehmensstrategie verknüpft,
- Berücksichtigt die Leitbilder und Werte des Unternehmens,
- Legt klare und beobachtbare Verhaltensweisen fest, die zur Überprüfung der Kompetenzen herangezogen werden können,

- Ist flexibel genug, um unabhängig von spezifischen Fachkenntnissen anwendbar zu sein,
- Wird regelmäßig aktualisiert und an sich wandelnde strategische Erfordernisse angepasst,
- Formuliert die Anforderungen an die Belegschaft in einer einheitlichen und für alle nachvollziehbaren Sprache,
- Dient als Basis für die Identifizierung und Bewertung von Kompetenzen sowie deren Niveaus mittels Online-Assessments, Fremd- und Selbsteinschätzungen oder Interviews,
- Unterstützt zielgerichtete Aktivitäten im Rahmen eines kompetenzbasierten Personalmanagements.

Die Elemente eines strategischen Kompetenzmodells, das die o.g. Anforderungen erfüllen kann, ist in folgende Abbildung 11 visualisiert [Sauter und Staudt 2016b].

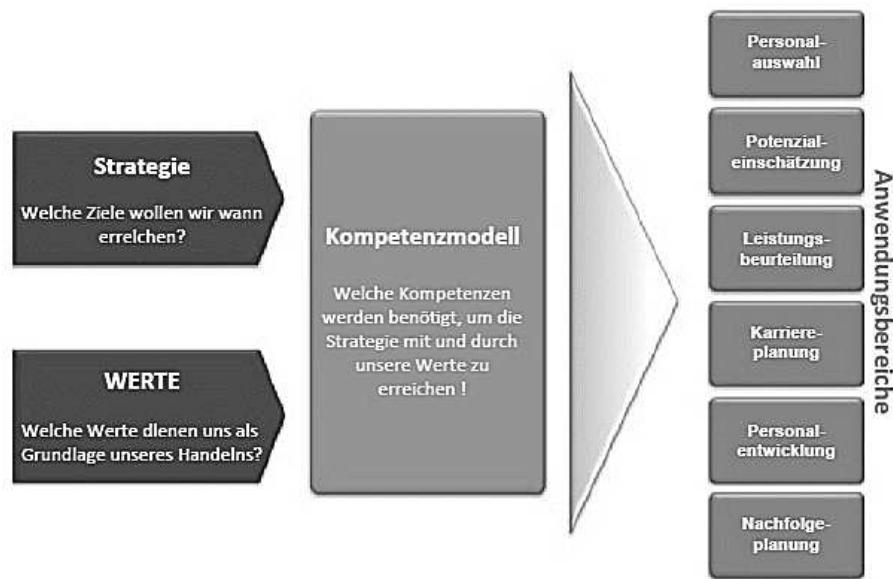


Abbildung 11: Strategisches Kompetenzmodell und Anwendungsbereiche in Anlehnung an [Sauter und Staudt 2016b]

Eine Erläuterung der diversen Kompetenzmodelle würde den Rahmen dieser Dissertation sprengen, da hier nicht die Entwicklung eines neuen Modells beabsichtigt ist. Stattdessen werden Informationen zur personellen Ressourcenplanung aus dem bestehenden Kompetenzmodell des Unternehmens abgerufen und analysiert. Daher war es wichtig zu erläutern, was ein Kompetenzmodell ist.

Im Folgenden wird erläutert, wie die Kompetenzen und die Ressourcenplanung in Projekten zusammenhängen und sich gegenseitig unterstützen können.

2.3.2 Kompetenzbasierte Ressourcenplanung

Das Humankapital, wie es heute genannt wird, bezieht sich auf die immateriellen Fähigkeiten, Kenntnisse, Qualifikationen, Erfahrungen und Motivationen der Mitarbeiter und ist ein wesentlicher Bestandteil des Humankapitals eines Unternehmens [Erpenbeck et al. 2017]. Ein zentraler Aspekt der

effektiven Nutzung dieses Humankapitals ist der kompetenzbasierte Managementansatz, der die Personalplanung in die strategische Unternehmensplanung integriert und insbesondere in der Ressourcenplanung und dem Projektmanagement eine entscheidende Rolle spielt.

Ein Unternehmen vergleicht die vorhandenen Kompetenzen seiner Mitarbeiter mit dem prognostizierten Kompetenzbedarf, der zur Umsetzung der Unternehmensvision und zur Erreichung der Geschäftsziele erforderlich ist. Wird eine Diskrepanz festgestellt, entwickelt und implementiert das Unternehmen gezielte Personalstrategien, um diese Lücken zu schließen [Tripathi und Agarwal 2014]. Draganidis und Mentzas beschreiben diesen Prozess als eine systematische Entwicklung von Beschäftigtenkompetenzen in vier Makrophasen, die auf kontinuierliche Verbesserung ausgerichtet sind. Das spiegelt sich auch auf Projektebene wider. Ein Projektleiter schätzt ab, welches Kompetenzprofil das Projektteam haben soll, um die Aufgaben im Projekt erfolgreich zu erledigen. Falls die Mitarbeiter nicht über die erforderlichen Kompetenzen verfügen, identifiziert der Projektleiter diese Lücken und entwickelt Maßnahmen, um die notwendigen Fähigkeiten zu schulen oder zu entwickeln. Das kann durch gezielte Schulungen, Workshops oder Mentoring-Programme geschehen. Das Phasenmodell ist in Abbildung 12 dargestellt.

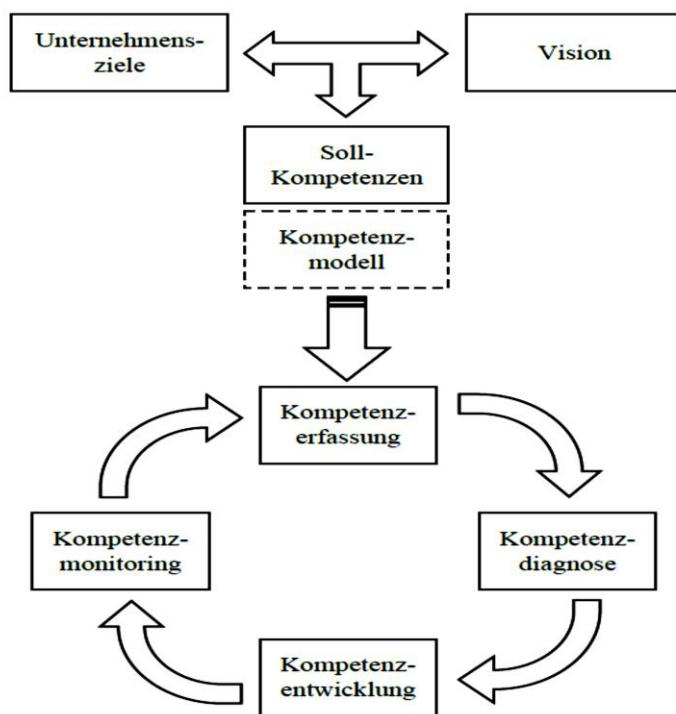


Abbildung 12: Makrophasen der kontinuierlichen Kompetenzverbesserung auf Grundlage der Unternehmensziele und Visionen in Anlehnung an [Draganidis und Mentzas 2006]

Nach der Abbildung des Phasenmodells von [Draganidis und Mentzas 2006] auf Projektebene, werden die Phasen wie folgt interpretiert. In der ersten Phase des Kompetenzmanagements geht es darum, alle erforderlichen Kompetenzen in einem Projekt zu ermitteln. Der Bedarf an Kompetenzen wird aus den Projektzielen und Anforderungen abgeleitet, woraus dann die für jede Projektrolle erforderlichen Fähigkeiten, Kenntnisse und Qualifikationsniveaus abgeleitet werden. In der zweiten Phase, der

Kompetenzdiagnose, wird eine Bestandsaufnahme der aktuellen Kompetenzen der Mitarbeiter vorgenommen und nach Kompetenzniveaus ausgewertet. Dabei werden auch eventuelle Kompetenzlücken identifiziert, indem die Anforderungen mit dem tatsächlichen Kompetenzniveau verglichen werden. Die dritte Phase befasst sich mit der Kompetenzentwicklung und umfasst die Planung von Aktivitäten zur Erhöhung der Anzahl und des Niveaus der erforderlichen Kompetenzen. Dazu gehören Schulungs- und Coachingmaßnahmen zur Verbesserung der Fähigkeiten und Kenntnisse der Mitarbeiter. In der letzten Phase, der Kompetenzüberwachung, werden die Ergebnisse der vorangegangenen Phasen überprüft und gegebenenfalls angepasst, um sicherzustellen, dass die Kompetenzen der Mitarbeiter den Projektanforderungen entsprechen und kontinuierlich weiterentwickelt werden [Draganidis und Mentzas 2006]. Die kompetenzbasierte Ressourcenplanung im Projektmanagement verwendet Kompetenzen, um die Qualifikationen der Projektmitglieder mit den Anforderungen der Projektrollen zu vergleichen. Es wird argumentiert, dass, wenn die richtigen Personen mit dem richtigen Qualifikationsprofil dem Projekt zugewiesen werden, so dass die Kompetenzen der Mitarbeiter und die Anforderungen des Projekts übereinstimmen, die Zufriedenheit und Effizienz des Teams steigen und somit die Gesamtleistung des Projekts verbessert wird [Chen et al. 2021; Özçelik und Ferman 2006].

Eine Analyse der Qualifikationsdefizite der Mitarbeiter im Vergleich zu den Anforderungen der definierten Projektrollen bildet die Grundlage für Schulungs- und Entwicklungspläne. Durch gezielte Weiterbildung werden Projektmitglieder an die spezifischen Anforderungen des Projekts angepasst, der interne Wissenstransfer wird gefördert und gesuchte Fachkräfte werden durch attraktive Entwicklungsangebote an das Projektteam gebunden. Bei der Leistungsbeurteilung im Projektkontext werden die Projektmitglieder von ihren Vorgesetzten und die Vorgesetzten von ihren Untergebenen beurteilt. Die Leistungsbeurteilung bezieht sich nicht nur auf die Erreichung der vereinbarten Projektziele (quantitativ), sondern auch auf die Art und Weise, wie die Arbeit ausgeführt wurde (qualitativ). Ziel der Leistungsbeurteilung ist es, individuelle Stärken und Schwächen aufzuzeigen, um Entscheidungen über Karriereentwicklung, Vergütung und andere Aspekte des Arbeitsverhältnisses zu treffen [Lange 2015; Titus 1994]. Die kompetenzbasierte Vergütung kann einem Unternehmen dabei helfen, die Vergütung der Projektmitglieder an die Projekteistung zu koppeln. Dabei werden die Kompetenzen, die die Projektmitglieder während ihrer Tätigkeit zeigen, in die Bewertung einbezogen. Ein kompetenzbasiertes System kann somit dazu beitragen, durch eine objektive Bewertung ein faires und gerechtes Vergütungssystem zu schaffen. Im Rahmen der Karriereplanung können Unternehmen ihren Projektmitgliedern die Möglichkeit geben, ihre eigenen Kompetenzen selbstständig auf ihre Eignung für angestrebte Projektrollen zu überprüfen. Durch die Bewertung und den Vergleich mit den geforderten Kompetenzen können die Projektmitglieder ihre eigenen Karrierepläne erstellen [Kuster et al. 2019]. Nachdem nun die verschiedenen Aspekte von Kompetenzen und Kompetenzarten sowie deren Zusammenhang mit der personellen Ressourcenplanung in Projektplanung beleuchtet wurden, richtet

sich der Fokus dieser Dissertation nun auf eine bestimmte Art von Projekten, nämlich Softwareentwicklungsprojekte.

2.4 Softwareentwicklung

In der Vergangenheit wurde die Frage diskutiert, ob Software als Produkt oder als Dienstleistung betrachtet werden sollte. Heute wird Software jedoch als immaterielles Produkt betrachtet. Im Allgemeinen bezieht sich der Begriff „Software“ auf programmierten Code, der auf Hardware ausgeführt werden kann, um verschiedene Aufgaben zu lösen [Masing 2021, S. 831]. Diese Definition ist jedoch oft unzureichend, da die Entwicklung eines Programms von der anfänglichen Anforderungserhebung bis zur Implementierung des endgültigen Codes eine Reihe von Dokumenten, wie Konzepte, Anforderungsdokumentation, Spezifikationen, Designdokumentation, Programmtexte (Quellcode) und Testdokumentation, erfordert [Lassmann 2006]. Eine Software besteht aus maschinell gespeicherten Anweisungen (Programmen), Vereinbarungen über die Eigenschaften der zu entwickelnden Daten und den für die Nutzung und Wartung erforderlichen Unterlagen. In Normen und Standards wird der Begriff „Software“ als ein geistiges Produkt definiert, das aus Programmen, Prozeduren und allen damit verbundenen Beschreibungen besteht, die mit der Arbeit mit einem Datenverarbeitungssystem verbunden sind. Software ist unabhängig von dem Medium, auf dem sie gespeichert ist. Ein „**Softwareprodukt**“ ist definiert als ein vollständiger Satz von Computerprogrammen, Prozeduren und zugehörigen Beschreibungen und Daten, die für die Nutzung durch den Benutzer bestimmt sind [Masing 2021, S. 382].

Software-Engineering, auch Softwaretechnik genannt, bezeichnet die systematische, rationelle, wirtschaftliche und jederzeit beherrschbare Entwicklung und Wartung von Software. Dies erfordert praxiserprobte Konzepte, Techniken, Methoden und Werkzeuge, die aufeinander abgestimmt sind und die erfolgreiche Durchführung umfangreicher Softwareentwicklungsprojekte ermöglichen. Um erfolgreich zu sein, erfordert Software Engineering Kenntnisse in verschiedenen Bereichen wie Prozesse, Werkzeuge, Projektmanagement, Qualitätssicherung, Kommunikation, Dokumentation und Personalmanagement [Masing 2021].

Ziel des Software-Engineerings ist die produktive Entwicklung von Softwaresystemen mit hoher Qualität in der geplanten Zeit und zu den veranschlagten Kosten (magisches Dreieck: Qualität - Kosten - Zeit). Die Erstellung komplexer Software erfordert ein professionelles Vorgehen, von der Anforderungserhebung über den Entwurf der Systemarchitektur bis hin zur Implementierung und Verifikation von späteren Anpassungen (Wartung). Es ist klar, dass die Prozesse der Softwareentwicklung nicht für jedes Projekt neu erfunden werden können. Mit der Einführung von Vorgehensmodellen wird die zeitliche und inhaltliche Strukturierung des Entwicklungsprozesses angestrebt. Der Prozess der Softwareentwicklung wird in aufeinander abgestimmte Phasen unterteilt, und für jede Phase werden Aktivitäten und Ergebnisse definiert. Einheitliche Vorgehensmodelle sind die Basis für die gemeinsame

Nutzung und das gemeinsame Verständnis der jeweiligen Prozesse. Vorgehensmodelle dienen als organisatorischer Rahmen für die Durchführung von Softwareentwicklungsprojekten. Sie ermöglichen die systematische Wiederverwendung von Best Practices, stellen die Vergleichbarkeit von Projekten sicher, verkürzen die Einarbeitungszeit von Mitarbeitern in neue Projekte und tragen so zur Produktivität bei. Wichtige Unterscheidungsmerkmale der Vorgehensmodelle sind die Dauer, Größe, der Anwendungsbereich und die Komplexität des Projekts. Zur Veranschaulichung des Softwareentwicklungsprozesses dient Abbildung 13, wie sie von [Plewan und Poensgen 2011] beschrieben wird.

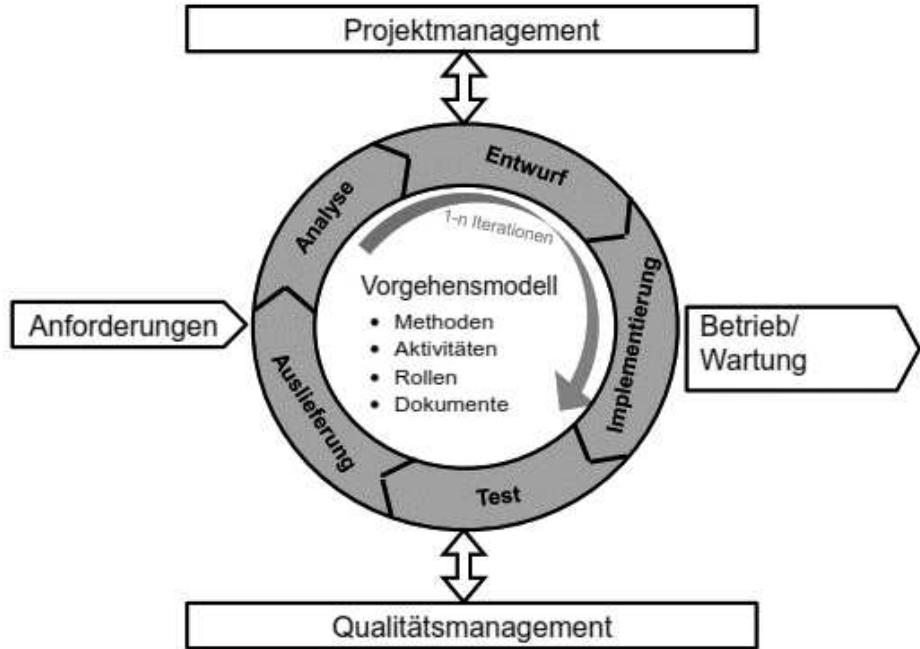


Abbildung 13: Der Prozess der Softwareentwicklung in Anlehnung an [Plewan und Poensgen 2011]

Es gibt verschiedene Arten von Prozessmodellen, wie z. B. sequentielle, inkrementelle, iterative, prototypische, agile, und Hybride Modelle. Ein einziges Prozessmodell kann nicht alle Anforderungen aller Projekte erfüllen, insbesondere in großen Entwicklungsbereichen mit vielen verschiedenen Projekten. Daher werden für die wichtigsten Projekte in einer Organisation in der Regel mehrere maßgeschneiderte Prozessmodelle benötigt. Die Vielzahl der unterschiedlichen Modelle lässt sich je nach Entwicklungsphilosophie in drei Kategorien einteilen, nämlich klassische, sequentielle Vorgehensmodelle, klassische, iterative und inkrementelle Vorgehensmodelle sowie agile Vorgehensmodelle [Masing 2021].

In dieser Dissertation liegt der Fokus auf dem V-Modell, das zu den klassischen, sequentiellen Vorgehensmodellen gehört.

2.4.1 Das V-Modell

Um das V-Modell zu verstehen, ist es notwendig zuerst das Wasserfallmodell zu analysieren. Das V-Modell ist dann eine Erweiterung des Wasserfallmodells. Das klassische sequentielle Prozessmodell für die Softwareentwicklung wird als Wasserfallmodell bezeichnet. Dieses Modell gibt klare Vorgaben

für die Hauptaktivitäten des Entwicklungsprozesses, grenzt diese ein und erleichtert so die Projektplanung und Kostenabschätzung. Die Phasen werden als geschlossene Einheiten von Aktivitäten und Ergebnissen definiert und laufen sequentiell ab. Jede Phase ist durch einen Meilenstein gekennzeichnet, der verifizierte und bewertete Ergebnisse enthält. Für die Freigabe von Arbeitsergebnissen, insbesondere bei sicherheitskritischen Anforderungen in eingebetteten Softwaresystemen, sind häufig formale Reviews erforderlich. Dies ist in den Prozessstandards festgelegt [Masing 2021]. Abbildung 5 zeigt die aufeinanderfolgenden Phasen dieses Modells.

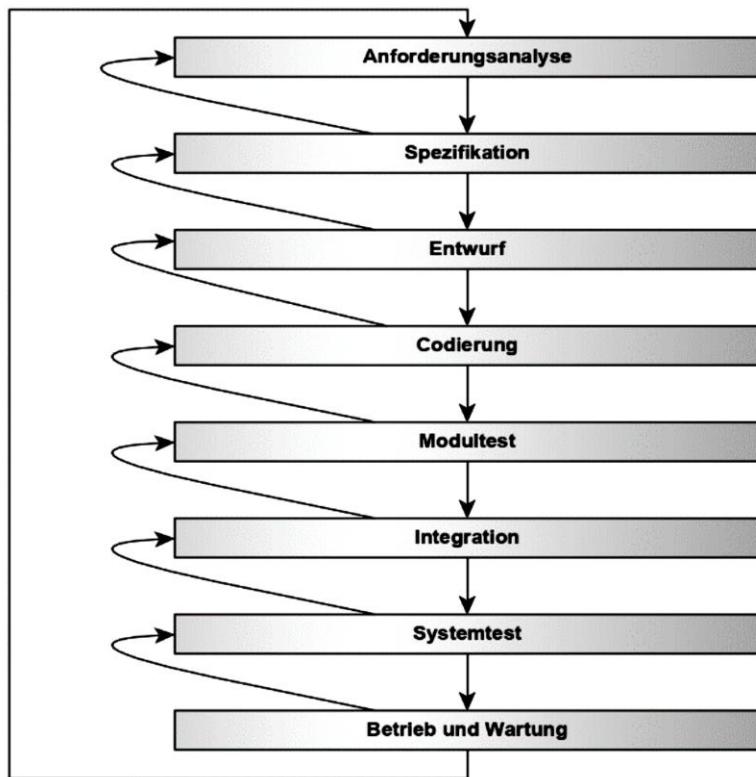


Abbildung 14: Das Wasserfallmodell in Anlehnung an [Boehm 1981]

Obwohl das Wasserfallmodell aufgrund seiner klaren Phasenabfolge häufig für genau definierte Anforderungen und Abläufe verwendet wird, hat es auch Nachteile. Es ist wenig flexibel und kann daher nur schwer auf späte Änderungen im Projekt reagieren, was zur späten Entdeckung von Fehlern und teuren Nacharbeiten führen kann. Ein weiterer Nachteil ist, dass der Kunde erst am Ende des Projekts die fertige Softwarelösung sieht und grundlegende Änderungen dann nicht mehr ohne weiteres möglich sind [Schatten et al. 2010]. In der Praxis der modernen Softwareentwicklung ist das Modell gelegentlich noch vorhanden, wird aber zunehmend durch flexiblere Prozessmodellen ersetzt bzw. gemischt, wie z.B. das V-Modell.

Das V-Modell spielt im Qualitätsmanagement eine wichtige Rolle, weil es Entwicklung und Test gleichbehandelt und die Notwendigkeit der Verifizierung aufzeigt. Das V-Modell von Böhm (s. Abbildung 15) stellt konstruktive Aktivitäten zur Fehlervermeidung (linker Zweig des V) Verifikationsaktivitäten wie Verifizierung und Validierung (rechter Zweig des V) gegenüber. Das Modell betont,

dass Fehler auf der Ebene gefunden werden, auf der sie auftreten. Bei der Systementwicklung werden zunächst Systemspezifikationen entwickelt, gefolgt von Details wie Komponentenspezifikation und Kodierung (Top-down: linker Zweig). Dann werden die Komponenten durch Integrations- und Testschritte zum Gesamtsystem hinzugefügt, d. h. Bottom-up (rechter Zweig). Im Ergebnis können in den Phasen des rechten Zweiges des V nur die Fehler erkannt werden, die in der entsprechenden Phase des linken Zweiges des V aufgetreten sind [Masing 2021]. Die folgende Abbildung veranschaulicht die beiden Zweige des V-Modells und ihre Korrelation.

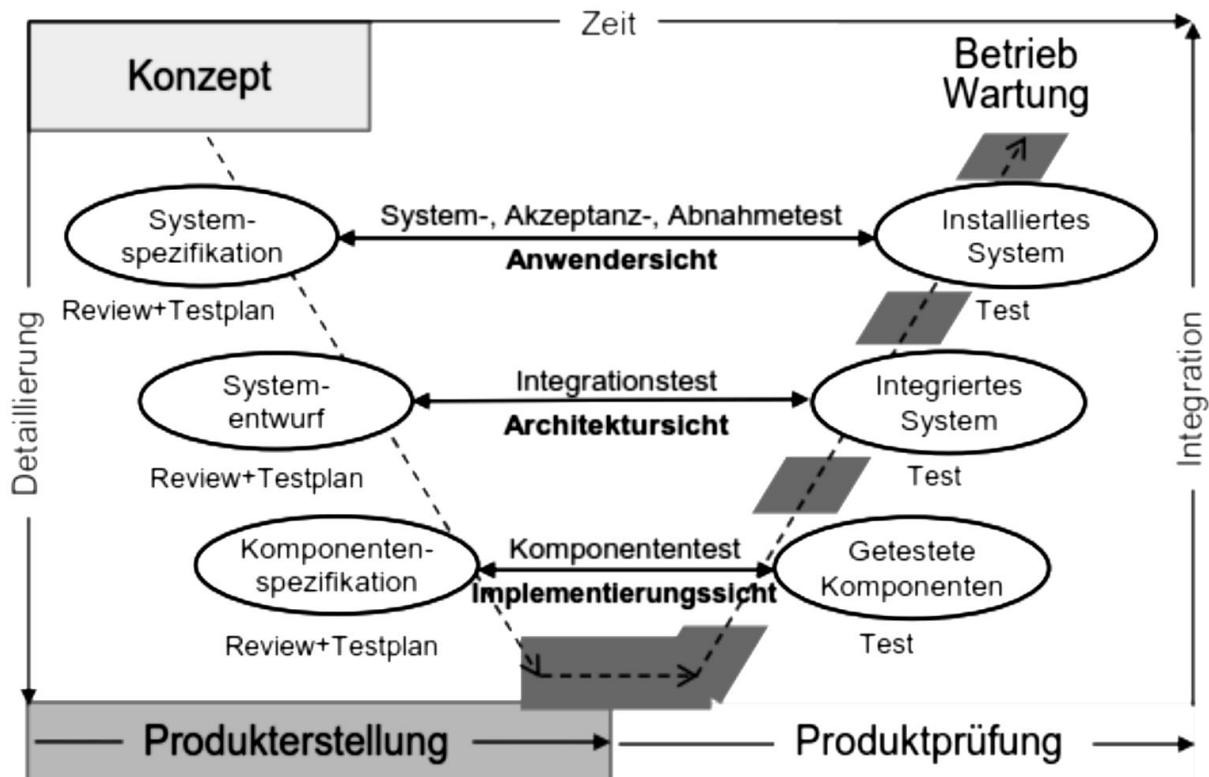


Abbildung 15: Das V-Modell nach Boehm [Boehm 2001]

Voraussetzung für eine erfolgreiche Umsetzung des V-Modells in der vorgestellten Form sind stabile Anforderungen über die gesamte Projektlaufzeit. Das Modell bietet aufgrund seines systematischen Aufbaus eine gute Planungsmöglichkeit. Im Gegensatz zum Wasserfallmodell hat das V-Modell aber auch Nachteile im Umgang mit Fehlern und Änderungen in späten Projektphasen. Ein Rücksprung in frühere Phasen ist nur bedingt möglich, was zu erheblichen Kosten für die Fehlerkorrektur führen kann, wenn Fehler erst spät im Entwicklungsprozess entdeckt werden. Daher befassen sich Qualitäts sicherungsmethoden und -maßnahmen wie Reviews und Inspektionen mit Anforderungen und Fehlern in den frühen Phasen der Softwareentwicklung. Die Nachteile der sequentiellen Modelle führten schließlich zu inkrementellen, iterativen und prototypischen Vorgehensmodellen, die flexibler auf Veränderungen im Projekt reagieren können [Schatten et al. 2010].

Um diese Herausforderungen zu adressieren und die Prozessqualität in der Softwareentwicklung weiter zu verbessern, wurde das Automotive SPICE Modell entwickelt.

2.4.2 Automotiv SPICE (Das Assessmentmodell)

Automotive SPICE steht für „Automotive Software Process Improvement and Capability Determination“ und ist ein international anerkanntes Bewertungsmodell für die Softwareentwicklung in der Automobilindustrie. Es ist eine Erweiterung des allgemeinen SPICE-Rahmens (Software Process Improvement and Capability Determination), der sich auf die Entwicklung von Softwareprodukten konzentriert [ASPICE 2023].

Automotive SPICE wurde auf Basis vom V-Modell entwickelt, um die Prozessqualität und -fähigkeit bei der Softwareentwicklung in der Automobilindustrie zu bewerten und zu verbessern. Das Assessment-Modell legt Anforderungen an die Prozesse fest, die bei der Entwicklung von Software für Automobile befolgt werden sollten. Dazu gehören beispielsweise Anforderungsmanagement, Systemarchitektur, Softwaredesign, Software-/Systemtest und Software-/Systemintegration.

Die Einhaltung von Automotive SPICE wird häufig von Automobilherstellern und -zulieferern gefordert, um sicherzustellen, dass die Softwareentwicklung den Qualitätsstandards der Branche entspricht. Es ermöglicht eine einheitliche Bewertung und Vergleichbarkeit der Prozessqualität zwischen verschiedenen Organisationen [Müller 2007].

Die folgende Abbildung veranschaulicht die Prozessdimensionen von Automotive SPICE:

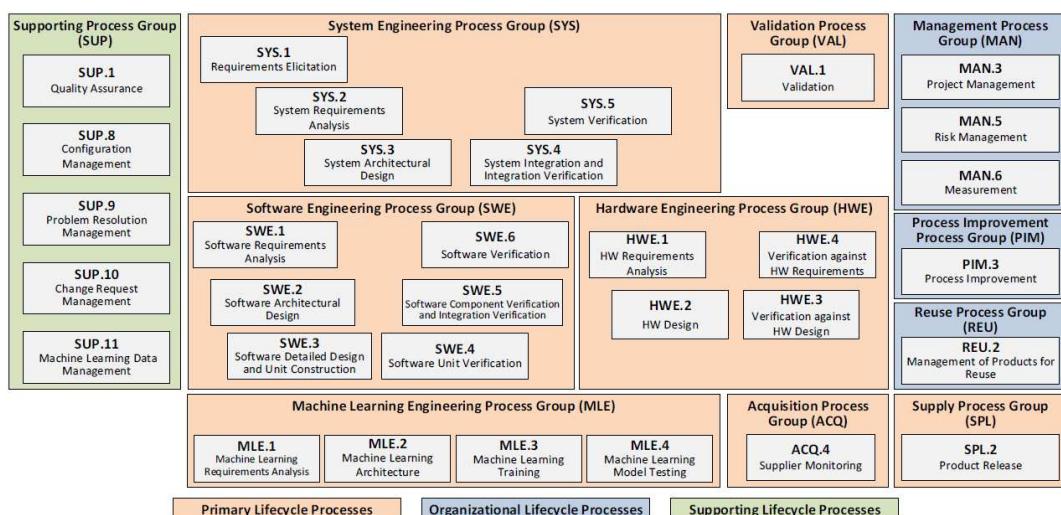


Abbildung 16: Automotive SPICE-Prozessreferenzmodell – Überblick [ASPICE 2023, S. 15]

Die Prozessdimension umfasst alle in Automotive SPICE definierten Prozesse. Einige dieser Prozesse sind in ISO/IEC 33020 zu finden, während andere speziell für Automotive SPICE definiert wurden. Die Prozesse werden nach Prozesskategorien und auf einer zweiten Ebene nach Prozessgruppen gruppiert, je nachdem, welche Art von Aktivität sie betreffen.

Es gibt drei Prozesskategorien: Primäre Lebenszyklus-Prozesse (ACQ, SPL, SYS, SWE, HWE, VAL, und MLE), Organisatorische Lebenszyklus-Prozesse (MAN, REU und PIM) und Unterstützende Lebenszyklus-Prozesse (SUP). Die Prozessgruppe ACQ enthält die Prozesse von Kunden oder Lieferanten des Kunden. In der Prozessgruppe SPL geht es um den Lieferumfang oder eine Leistung, die

der Lieferant oder Dienstleister erbringen muss. Die Engineering-Prozesse SYS, SWE, HWE und MLE beziehen sich auf die Anforderungen und Design des Kunden von Gesamt-System inklusive HWE und darauffolgend auf Entwicklung von der Softwareprodukte. Daraus ergibt sich ein Bezug zum System, das dann implementiert oder spezifiziert wird. Die Prozessgruppe Management (MAN) besteht aus Prozessen, die von jedem verwendet werden können, der ein Projekt oder einen Prozess innerhalb des Lebenszyklus leitet, z. B. Projektmanagement, Risikomanagement und Messung. Die Prozessgruppe Prozessverbesserung (PIM) umfasst einen Prozess, der Praktiken zur Verbesserung der in der Organisationseinheit durchgeführten Prozesse enthält, und schließlich umfasst die Prozessgruppe Wiederverwendung (REU) einen Prozess zur systematischen Nutzung von Wiederverwendungsmöglichkeiten in den Wiederverwendungsprogrammen der Organisation. Die Unterstützende Prozesse (SUP) können an verschiedenen Stellen im Prozesslebenszyklus eingesetzt werden, um die Durchführung des Projekts zu unterstützen. Die Prozessgruppe deckt verschiedene Themen von der Qualitätssicherung über die Dokumentation bis hin zum Problem- und Änderungsmanagement ab [ASPICE 2023]. Jeder Prozess wird durch eine Zweckbeschreibung dargelegt. Die Zweckbeschreibung enthält die spezifischen, funktionalen Ziele des Prozesses, wenn er in einer bestimmten Umgebung durchgeführt wird. Mit jeder Zweckbeschreibung ist eine Liste spezifischer Ergebnisse verknüpft, die die erwarteten positiven Resultate der Prozessdurchführung darstellen.

Die Implementierung von Automotive SPICE in der Praxis beginnt mit einer gründlichen Analyse der bestehenden Prozesse und der Identifikation von Lücken im Vergleich zu den Anforderungen des Modells. Es ist entscheidend, dass Unternehmen ihre Mitarbeiter schulen und die neuen Prozesse klar dokumentieren, um sicherzustellen, dass alle Beteiligten die Standards verstehen und anwenden können. Dieser initiale Schritt erleichtert die Anpassung der Unternehmensprozesse an die Vorgaben von Automotive SPICE. Ein weiterer wichtiger Punkt ist der kontinuierliche Verbesserungsprozess (engl. Continual Improvement Process, CIP). Unternehmen sollen regelmäßig Feedback von Mitarbeitern und Kunden einholen und Metriken sowie Kennzahlen nutzen, um die Prozessleistung zu überwachen. Dies ermöglicht eine proaktive Optimierung der Prozesse und fördert eine Kultur der kontinuierlichen Verbesserung. Es gibt jedoch Herausforderungen bei der Implementierung von Automotive SPICE. Dazu zählen Widerstände der Mitarbeiter und die Integration in bestehende Prozesse. Es werden Lösungsansätze wie effektives Change-Management, die Durchführung von Pilotprojekten und die Unterstützung durch Führungskräfte vorgeschlagen, um diese Herausforderungen erfolgreich zu meistern.

Nachdem die Bedeutung von Automotive SPICE für die Prozessqualität in der Automobilindustrie erläutert wurde, richtet sich der Fokus nun auf den eDeCoDe-Ansatz zur Systemmodellierung. Dieser Ansatz ermöglicht eine ganzheitliche Analyse und Visualisierung von Systemen, einschließlich der Interaktionen zwischen den einzelnen Systemelementen. In Kapitel 5.2 wird die Logik von eDeCoDe auf Automotive SPICE abgebildet, weshalb es wichtig ist, sowohl Automotive SPICE als auch den eDeCoDe-Ansatz umfassend zu erläutern.

2.4.3 eDeCoDe-Ansatz zur Systemmodellierung

Neben den Softwareentwicklungsmodellen wie dem V-Modell, auf das zuvor eingegangen wurde, ist es essentiell, das System in der Gesamtheit seiner inklusiven Interaktionen zwischen den einzelnen Systemelementen analysieren, verstehen und visualisieren zu können. In diesem Zusammenhang erweist sich der eDeCoDe-Ansatz zur Systemmodellierung als besonders geeignet, welche seinen Ursprung im Bereich des „Generic Systems Engineering (GSE)“ hat. Die wissenschaftlichen Untersuchungen und die daraus resultierenden Schlüsse am Fachgebiet Produktsicherheit und Qualität haben gezeigt, dass zur umfassenden Darstellung technischer und soziotechnischer Systeme eine Integration verschiedener Perspektiven – speziell Anforderungen, Funktionen, Prozesse, Komponenten und Personen – erforderlich ist. Auf dieser Erkenntnis basiert das GSE-Denkmodell, das aus diesen fünf interdependenten Perspektiven besteht. Für die Modellierung dieser Systeme wurden der Demand Compliant Design (DeCoDe) Ansatz für technische Systeme und der erweiterte Demand Compliant Design (e-DeCoDe) Ansatz für soziotechnische Systeme entwickelt. Diese Ansätze dienen als übergeordnete Metamodelle, die eine einheitliche Grundlage für die interdisziplinäre Modellierung bieten [Mistler 2021] und [Winzer 2017].

Um eine einheitliche Anwendung des eDeCoDe Ansatz zu ermöglichen, ist eine grundlegende Erläuterung dessen erforderlich. Diese beginnt mit der Definition der einzelnen Systemsichten, wie in Tabelle 5 beschrieben.

Tabelle 5: Definition der eDeCoDe-Elemente in Anlehnung an [Schlüter 2023]

Begrifflichkeiten	Definition
Anforderung	Anforderungen sind Erfordernisse oder Erwartungen von Stakeholdern an ein System, welche festgelegt, üblicherweise vorausgesetzt oder verpflichtend sind.
Funktion	Funktionen beschreiben den Zweck bzw. die Aufgabe, die ein System zu erfüllen hat. Sie geben damit der Umwandlung von Eingaben in Ausgaben eines Systems eine Zielrichtung. Dadurch ermöglichen Funktionen eine Beschreibung des „Was“. Das bedeutet, was ein System oder Teile davon realisieren sollen.
Prozess	Prozesse beschreiben, wie die Eingaben eines Systems in Ausgaben umgewandelt werden, also das „Wie“. Über den Prozess realisiert sich die eingebaute Funktionalität des Systems, d.h. innerhalb von Prozessen werden bei technischen Systemen durch die Nutzung von Komponenten Funktionen umgesetzt. Erfolgt die Einbindung von Personen in Prozesse, werden letztere oftmals auch als Arbeits- oder Geschäftsprozesse bezeichnet (Prozess eines soziotechnischen Systems).
Komponente	Komponenten sind physische oder logische, einzelne oder zusammengefasste Bestandteile eines Systems.
Person	Personen beschreiben Menschen. Sie nutzen und realisieren Komponenten wie auch Prozesse und stellen Input und Output für die Leistungserbringung zur Verfügung. Somit realisieren sie Funktionen, welche wiederum Anforderungen erfüllen.

Für die logische Verknüpfung der Systemsichten in einem Metamodell sind die Festlegung von Semantik und Syntax unerlässlich. Die Abbildung 17 illustriert die Semantik und Syntax der eDeCoDe

Ansätze und veranschaulicht zugleich die Unterschiede zwischen dem DeCoDe- und dem e-DeCoDe-Ansatz. Diese Differenzierung ist historisch bedingt und resultiert aus der Erkenntnis, dass für die Modellierung soziotechnischer Systeme eine Erweiterung notwendig war. Die Unterscheidung der Ansätze ermöglicht eine klare terminologische Trennung zwischen der Modellierung technischer und soziotechnischer Systeme.

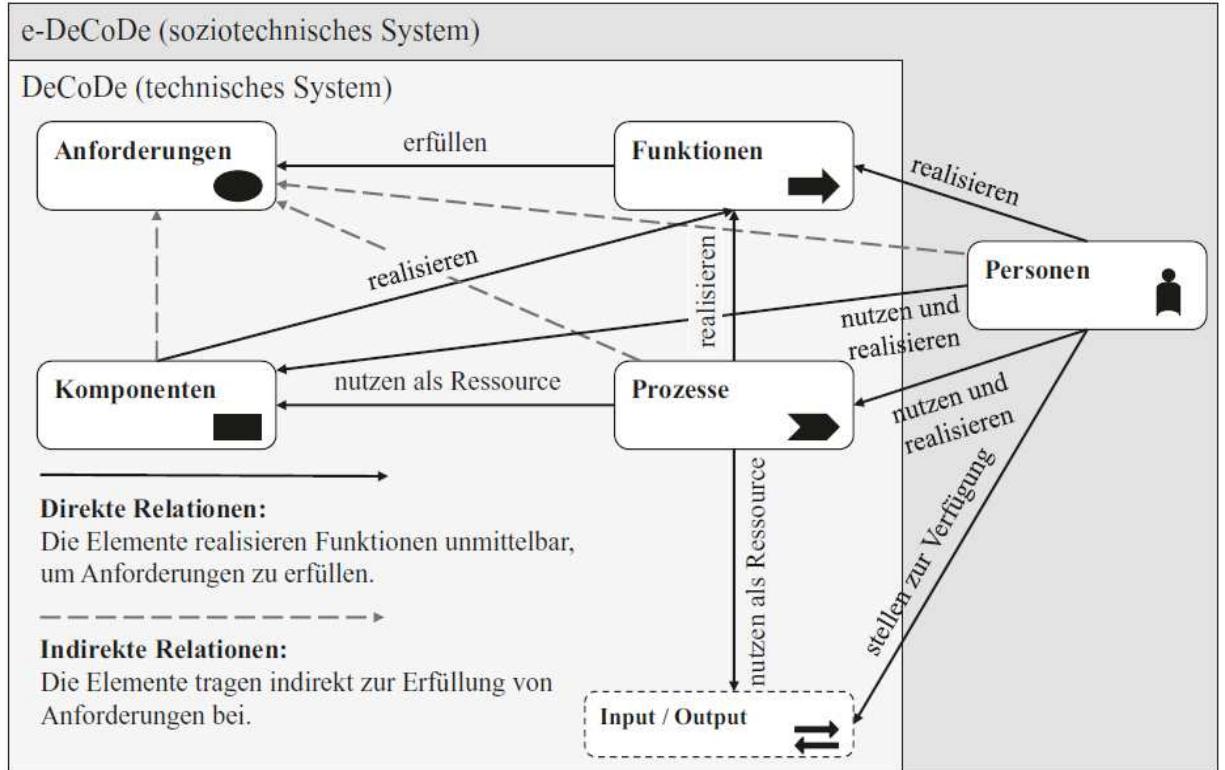


Abbildung 17: Semantik und Syntax von DeCoDe vs eDeCoDe in Anlehnung an [Schlüter 2023]

Die Abbildung 17 zeigt, dass Komponenten, Prozesse und Personen die Funktionen realisieren, wobei Prozesse Komponenten als ihre Ressourcen einsetzen. Eine Funktion dient demnach der Erfüllung einer Anforderung. Die Personen nutzen ebenso Komponenten und Prozesse, um diese zu realisieren. Die Wechselbeziehungen zwischen den Elementen werden durch den Vergleich der einzelnen Systemelemente mittels eDeCoDe-Matrizen analysiert.

Die Matrixlogik, wie sie in Abbildung 18 dargestellt ist, basiert auf der Graphentheorie, wie sie bei [Lindemann 2009] beschrieben wird. Demnach sind die Design Structure Matrix (DSM), die Domain Mapping Matrix (DMM) und die Multiple Domain Matrix (MDM) als implizite Bestandteile oder Perspektiven der eDeCoDe-Werkzeuge zu verstehen. Diese Perspektiven zielen darauf ab, situationsspezifische Modelle zu erstellen, die auf die jeweiligen Anforderungen und Kontexte zugeschnitten sind.

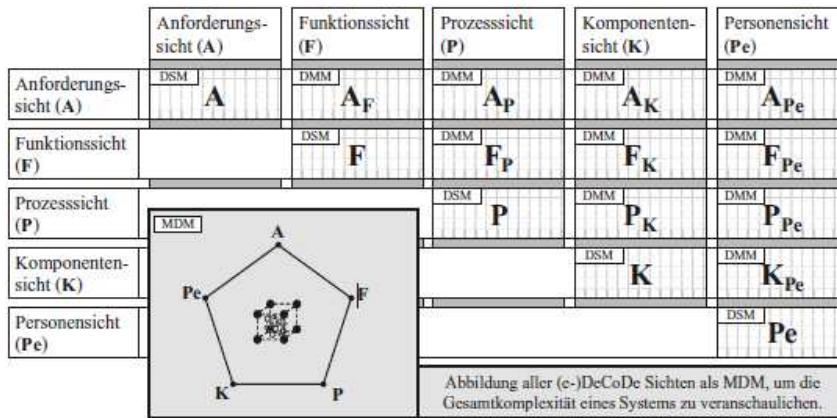


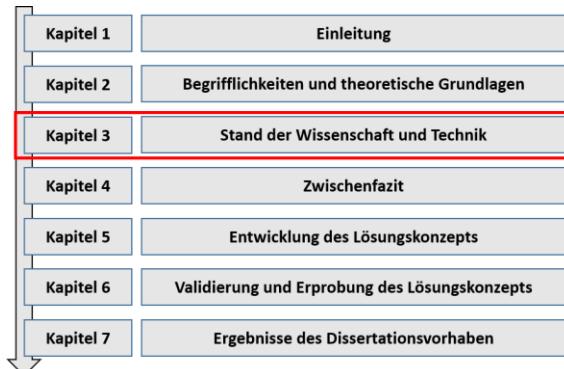
Abbildung 18: Grundschema von eDeCoDe [Mistler et al. 2021]

Die DSM ist eine quadratische Matrix, also eine Matrix, in der die Anzahl der Zeilen und Spalten gleich ist, und dient der Darstellung der Beziehungen zwischen Elementen innerhalb einer Domäne (z.B. Anforderung-Anforderung oder Funktion-Funktion). Die DMM hingegen ermöglicht es, Beziehungen zwischen Elementen aus zwei unterschiedlichen Domänen zu visualisieren (z.B. Anforderung-Funktion oder Komponente-Prozess). Für die Darstellung komplexerer Zusammenhänge, die mehr als zwei unterschiedliche Domänen umfassen, wird die MDM eingesetzt. Diese ist insbesondere von Bedeutung, um die umfassende Komplexität eines Systems zu erfassen und zu analysieren [Mistler 2021]. Wie in Abbildung 18 anhand der MDM angedeutet, können die Matrizen auch dazu verwendet werden, Transparenz über die vielfältigen Wechselwirkungen innerhalb des Systems zu schaffen.

Auf diese Weise wird das Gesamtsystem gezielt hinsichtlich der Interaktionen seiner Elemente modelliert und untersucht. Es wird eruiert, inwiefern das System alle gestellten Anforderungen erfüllt und welche Systemelemente sich wechselseitig beeinflussen. Dieser Ansatz ermöglicht eine umfassende Betrachtung des Systems, indem er nicht nur die Erfüllung einzelner Anforderungen, sondern auch die dynamischen Beziehungen und Abhängigkeiten innerhalb des Systems berücksichtigt [Schlund 2011]. Es ist jedoch nicht zwingend erforderlich, alle Sichten der Ansätze in jeder Modellierung zu verwenden. Der Kerngedanke des eDeCoDe Ansatzes ist, dass nur jene Sichten aus dem Metamodell herangezogen werden, die für die Entwicklung des spezifischen Systems notwendig sind. Die Anwendung von eDeCoDe führt zur Nutzung spezifischer eDeCoDe Werkzeuge, die in Abbildung 18 dargestellt sind und vorrangig auf Matrixdarstellungen basieren. Diese ermöglichen eine konsistente Verknüpfung zwischen den verschiedenen Sichten und sind essenziell, um die Komplexität von Systemen deutlich zu machen. Dies wird durch zahlreiche Forschungsarbeiten bestätigt, die die Systementwicklung auf Basis von eDeCoDe durchführen [Schlüter 2023].

Im Rahmen dieser Dissertation wird in Kapitel 5.2.2 eine fachspezifische Ontologie entwickelt, die auf der Kombination der eDeCoDe-Logik und Automotive SPICE abzielt. Hierbei werden die eDeCoDe-Logik und ihre zwei Werkzeuge, die Domain-Mapping-Matrix (DMM) und die Design-Structure-Matrix (DSM), angewendet und auf das Prozessreferenzmodell von Automotive SPICE abgebildet.

3 Stand der Wissenschaft und Technik



Um die Relevanz und Aktualität des Dissertationskonzepts innerhalb des wissenschaftlichen und technologischen Rahmens zu verorten, ist es von essenzieller Bedeutung, den gegenwärtigen Forschungs- und Entwicklungsstand zu analysieren und darzulegen. Dieses Kapitel zielt darauf ab, einen Überblick über den aktuellen Stand der Wissenschaft und Technik zu geben, welcher für

die Thematik dieser Arbeit relevant ist. Hierbei wird insbesondere auf die Fortschritte und Herausforderungen in den Bereichen des Projektmanagements und der Kompetenzmessung unter anderem die Ansätze, die moderne IT-Lösungen benutzen, eingegangen. Zudem wird eine eingehende Betrachtung verschiedener Ansätze zum Projektmanagement vorgenommen, wobei ein besonderes Augenmerk auf die Planungsphase gelegt wird. Dies schließt eine Analyse der aktuellen Methodologien und Werkzeuge ein, die in der Projektplanung zur Anwendung kommen, um die Effizienz und Effektivität in der Projektdurchführung zu steigern. Weiterhin werden unterschiedliche Ansätze zur Kompetenzmessung erörtert, die es ermöglichen, die Fähigkeiten und Fertigkeiten von Projektmitarbeitern präzise zu erfassen und zu bewerten. Diese Messmethoden sind entscheidend, um eine fundierte Grundlage für die kompetenzbasierte Projektplanung zu schaffen, welche als nächstes Thema behandelt wird. Hierbei wird untersucht, wie durch die Berücksichtigung individueller Kompetenzen bei der Zuweisung von Projektrollen und -aufgaben die Projektperformance signifikant verbessert werden kann. Abschließend wird der Fokus auf die Möglichkeiten der KI-gestützten Projektplanung gerichtet. Es wird dargelegt, wie künstliche Intelligenz genutzt werden kann, um Projektplanungsprozesse zu optimieren, Risiken zu minimieren und Entscheidungsfindungen zu unterstützen. Durch die Exploration dieser Themenbereiche wird ein umfassender Überblick über den aktuellen Stand der Forschung und Technologie in Bezug auf moderne Projektmanagementstrategien und -werkzeuge geboten, wobei der Nutzen von KI-Technologien und kompetenzbasierten Ansätzen besonders hervorgehoben wird.

3.1 Wissenschaftliche Ansätze zum Projektmanagement in Software Entwicklung

In der Softwareentwicklung werden verschiedene Projektmanagement-Ansätze verwendet, um die Komplexität und Vielfältigkeit von Projekten zu bewältigen. Ein universelles, starres Modell kann

nicht den unterschiedlichen Anforderungen vieler Projekte gerecht werden. Aus diesem Grund implementieren Organisationen, die an einer Vielzahl von Projekten arbeiten, oft mehrere maßgeschneiderte Modelle, die auf die spezifischen Bedürfnisse ihrer Schlüsselprojekte zugeschnitten sind. Diese Modelle lassen sich gemäß ihrer Entwicklungsphilosophie in drei Hauptkategorien klassifizieren [Masing 2021]:

1. Traditionelle Modelle
2. Agile Modelle
3. Hybride Modelle

Das **traditionelle** Projektmanagement, folgt einer linearen Abfolge, die die Projektinitiierung, Projektplanung, Projektdurchführung, Projektüberwachung und schließlich den Projektabschluss umfasst [Ciric et al. 2019]. Innerhalb des traditionellen Projektmanagements werden zahlreiche Ansätze vorgestellt, wie die Wasserfallmethode, das Spiralmodell, der inkrementelle Ansatz, die schnelle Anwendungsentwicklung (engl. Rapid Application Development, RAD) und das V-Modell. Das Hauptaugenmerk des traditionellen Projektmanagements liegt darauf, den Umfang und die Anforderungen des Projekts in der Initiierungsphase festzulegen [Leong et al. 2023]. Die **Wasserfallmethode** strukturiert ein Projekt in lineare, sequenzielle Aufgaben, wobei die Ergebnisse einer Phase als Grundlage für die nachfolgende Phase dienen. Typischerweise erfordert dieser Ansatz den offiziellen Abschluss einer Phase, bevor zur nächsten übergegangen werden kann. Für jede Projektphase wird eine Liste von Aufgaben mit Details, Anforderungen und Erfolgskriterien vorbereitet [Sinha und Das 2021]. Jedoch weist die Wasserfallmethode trotz seiner Klarheit und Einfachheit mehrere Nachteile auf. Die Inflexibilität im Umgang mit Änderungen stellt eines der Hauptprobleme dar: Ist eine Phase abgeschlossen, gestaltet es sich schwierig, Änderungen vorzunehmen. Diese Starrheit erschwert die Integration neuer Anforderungen oder Änderungen, die während des Entwicklungsprozesses häufig auftreten. Ein weiterer kritischer Punkt ist die späte Integration von Feedbacks. Stakeholder und Kunden sehen das Produkt erst, nachdem es alle Entwicklungsstadien durchlaufen hat, was bedeutet, dass Feedback sehr spät im Prozess einfließt. Dies kann zu erheblichem Mehraufwand führen, wenn das gelieferte Produkt nicht den Anforderungen oder Erwartungen der Stakeholder entspricht [Sinha und Das 2021]. Hinzu kommt eine mangelhafte Risikomanagementfähigkeit. Risiken werden oft zu spät im Prozess identifiziert, um effizient angegangen zu werden, was zu potenziellen Projektüberschreitungen in Bezug auf Zeit und Kosten führen kann. Die Schätzung von Zeit und Kosten zu Projektbeginn ist besonders bei großen oder komplexen Projekten herausfordernd und kann zu unrealistischen Zeitplänen und Budgets führen. Während das Wasserfallmodell für Projekte mit sehr klaren Anforderungen und ohne erwartete Änderungen wirksam sein kann, machen diese Nachteile es weniger geeignet für dynamische Projekte, bei denen Flexibilität, kontinuierliche Beteiligung der Stakeholder und die Fähigkeit, sich an Veränderungen anzupassen, entscheidend sind [Leong et al. 2023]. Das V-Modell, welche in Kapitel 2.4.1 untersucht wurde, eine Erweiterung des Wasserfallmodells und wird auch als Verifizierungs- und

Validierungsmodell bezeichnet. Im V-Modell hat jede Testphase ihren eigenen Verifizierungs- und Validierungsprozess, wobei jeweils nur eine Testphase zurzeit ausgeführt wird.

Das Rapid-Application-Development (RAD) ist ein iterativer und inkrementeller Softwareentwicklungsprozess, der auf schnelle Prototypenentwicklung und wiederholtes Feedback der Benutzer abzielt. RAD hilft dabei, die Entwicklungszeiten erheblich zu verkürzen und die Flexibilität während des Entwicklungsprozesses zu erhöhen. Dies wird durch vier Hauptphasen erreicht. In der Anforderungsplanung werden die Anforderungen des Projekts durch Workshops oder Fokusgruppen erfasst. Anstatt detaillierte Spezifikationen zu erstellen, wird in der Phase des Prototyping und schnellen Designs ein funktionaler Prototyp entwickelt, der als Grundlage für weitere Entwicklungen und Anpassungen dient. In der Konstruktion wird der endgültige Prototyp zu einem voll funktionsfähigen System weiterentwickelt, wobei diese Phase das Schreiben des Codes, die Integration der verschiedenen Komponenten und die Durchführung von Tests umfasst. Schließlich wird in der Implementierungsphase das System implementiert, und die Benutzer erhalten Schulungen. Feedback wird weiterhin gesammelt, um zukünftige Verbesserungen zu berücksichtigen [Martin 1991]. Das RAD-Modell ist bekannt für seine Fähigkeit die Softwareentwicklung zu beschleunigen und Flexibilität zu erhöhen, weist jedoch mehrere Nachteile auf, die seine Eignung für bestimmte Projekte beeinträchtigen können. Ein wesentliches Hindernis stellt die hohe Abhängigkeit von einem erfahrenen und agilen Team dar, dessen Zusammenstellung besonders für neue Unternehmen oder unerfahrene Teams eine Herausforderung sein kann. Zudem fördert die Methode häufige Code-Änderungen, die das Risiko technischer Schulden erhöhen und deren Management in größeren Projekten erschweren, was mit der Zeit kostspielige Lösungen erfordern kann [Leong et al. 2023]. Der Erfolg eines RAD-Projekts hängt stark von dem vollständigen Engagement aller Stakeholder ab; fehlt es an dieser Hingabe, stehen erhebliche Herausforderungen bevor. Die Auswahl des richtigen App-Entwicklungstools, das sowohl Low-Code- als auch No-Code-Funktionen ausgewogen bietet, ist entscheidend und kann das Projekt maßgeblich beeinflussen. Das RAD eignet sich vornehmlich für kleinere bis mittelgroße Projekte, da größere Vorhaben aufgrund ihrer Komplexität und Koordinationsanforderungen einen anderen Ansatz benötigen. Ein weiterer Nachteil ist der Mangel an umfassender Dokumentation, der zukünftige Modifikationen oder die Nachverfolgung des Projektfortschritts erschweren kann [Leong et al. 2023]. Es ist wichtig, diese Nachteile im Hinblick auf die spezifischen Bedürfnisse und Umstände des eigenen Projekts zu bewerten.

Das Spiralmodell ist ein risikogesteuerter Ansatz im traditionellen Projektmanagement, der 1981 von Barry Boehm eingeführt wurde [Boehm 1981]. Es kombiniert die Wasserfallmethode mit dem iterativen Ansatz. Ähnlich wie beim Wasserfallmodell wird ein Projekt im Spiralmodell in mehrere Phasen unterteilt, wobei jede Phase ein eigenes Designziel hat und das Ergebnis am Ende der Phase vom Kunden überprüft wird. Der Softwareentwicklungsprozess beginnt mit einem kleinen Satz von Anforderungen, der mit jeder weiteren Spirale erweitert wird. Das Spiralmodell beginnt mit der Projektplanung, gefolgt von einer Risikobewertung, Engineering (Kompilieren, Testen und Bereitstellen) und

schließlich der Bewertung der Software durch den Kunden [Ciric et al. 2019]. Das Spiralmodell, obwohl es für große, komplexe und risikoreiche Projekte gut geeignet ist, bringt mehrere Nachteile mit sich, die seine Anwendbarkeit einschränken können. Die Komplexität des Modells, das traditionelle Ansätze wie das Wasserfallmodell übersteigt, liegt in seiner iterativen Natur und der Notwendigkeit regelmäßiger Risikobewertungen in jeder Phase begründet. Dies kann das Management erschweren, insbesondere für kleinere Projekte oder Teams, die mit dieser Methode nicht vertraut sind. Aufgrund der iterativen Ausrichtung und der Notwendigkeit kontinuierlicher Kundenrückmeldungen und Risikobewertungen kann das Spiralmodell kostspieliger sein als direktere Ansätze. Dies zeigt sich besonders bei kleineren Projekten, wo eine umfangreiche Risikoanalyse und Kundenfeedbackschleifen möglicherweise nicht gerechtfertigt sind [Ciric et al. 2019]. Die Notwendigkeit wiederholter Iterationen, jede mit eigener Planung, Risikoanalyse, Engineering und Bewertungsphasen, kann die Zeitlinie des Projekts verlängern und ist somit für Projekte mit strikten Fristen ungeeignet. Deshalb ist für Projekte, bei denen Zeit und Kosten begrenzt sind oder der Projektumfang von Anfang an gut verstanden wird und keine spezielle Risikobasierter Ansatz benötigt ist, andere Methodologien angemessener [Leong et al. 2023].

Das inkrementelle Modell teilt das Projekt in kleine Teile auf, wobei jeder Teil auf der freigegebenen Funktionalität des vorherigen Teils aufbaut. Mit diesem Ansatz kann ein System den Kunden wiederholt mit inkrementeller Funktionalität zur Verfügung gestellt werden. Der Umfang wird zu Beginn des Projekts definiert, gefolgt von einer Gesamtprojektplanung. Für das inkrementelle Modell wird eine Wiederholung der Phasen des Startens eines Inkrement, Überwachens und Steuerns des Inkrement und schließlich des Abschlusses der Inkrement Phase wiederholt, bis alle Inkremeinte der Projektplanung abgedeckt sind [Ciric et al. 2019]. Das inkrementelle Modell, obwohl es für Projekte mit gut verstandenen und klar segmentierbaren Anforderungen nützlich ist, birgt mehrere Herausforderungen. Besonders bei komplexen Vorhaben kann es herausfordernd sein, sämtliche Bedürfnisse im Voraus zu erfassen, was zu potenziellen Problemen oder Lücken in späteren Inkrementen führen kann. Die Verwaltung mehrerer Inkremeinte kann einen erhöhten administrativen und Managementaufwand bedeuten, da jedes Inkrement eigene Planungs-, Entwicklungs-, Test- und Integrationsphasen erfordert, was die Projektkoordination komplizieren kann. Ein weiteres Risiko ist der sogenannte Scope Creep, der durch die flexible Natur des Modells begünstigt wird, wenn während der Entwicklung neuer Inkremeinte zusätzliche Funktionen oder Änderungen angefordert oder identifiziert werden. Dies kann zu erhöhten Kosten und zeitlichen Verzögerungen führen [Leong et al. 2023]. Integrationsherausforderungen treten auf, wenn es darum geht, die verschiedenen Teile zu einem kohärenten Ganzen System zusammenzufügen. Technologische Unterschiede, Designinkonsistenzen oder Richtungsänderungen im Projekt können die Integration erschweren und zusätzlichen Aufwand erfordern, um Kompatibilität und Funktionalität zu gewährleisten. Frühzeitige Entscheidungen können das Projekt in bestimmte Technologien oder Designansätze einschließen, die später problematisch werden können, was die Flexibilität bei der Berücksichtigung zukünftiger Anforderungen oder der

Einführung neuer Technologien einschränkt. Die effiziente Ressourcenzuweisung über die Inkremente hinweg stellt ebenfalls eine Herausforderung dar, insbesondere wenn einige Inkremente komplexer oder zeitaufwändiger sind als erwartet. Das Balancieren der Arbeitsbelastung und die effektive Nutzung des Teams erfordern sorgfältige Planung und Anpassung. Für Projekte mit stark unsicheren oder sich entwickelnden Anforderungen oder bei starker Abhängigkeit zwischen den Funktionen könnten andere Entwicklungsansätze geeigneter sein, um diese Nachteile zu minimieren [Grimm 2003].

Das **agile** Projektmanagement bzw. Denkweise wird in der Softwareentwicklung angewendet und fand erstmals in den 1990er Jahren Verbreitung, um einen Wandel in der Softwareentwicklung einzuleiten. Heute ist sie in vielen Softwareentwicklungsorganisationen fest etabliert [Birkinshaw 2018]. Diese Methodik umfasst eine Reihe von Praktiken, die es einem Team ermöglichen, ein Projekt durch die Aufteilung langer Wartezeiten in mehrere Iterationen mit kürzeren Zykluszeiten zu verwalten. Jede Iteration, auch Sprint genannt, durchläuft vier Phasen: Anforderungsphase, Implementierungsphase, Testphase und Kundenrückblick, um Feedback zu sammeln, das in der nächsten Iteration umgesetzt werden kann [Cooper und Sommer 2018]. Nach dem Manifest für agile Softwareentwicklung gibt es vier zentrale Prinzipien der agilen Methodik:

1. Individuen und Interaktionen haben Vorrang vor Prozessen und Werkzeugen. Die Kommunikation zwischen Entwicklern und Kunden ist wichtiger als strikte Systemprozesse.
2. Funktionierende Software ist wichtiger als umfassende Dokumentation. Der Erfolg in der Lieferung der Software hat Vorrang vor dem Dokumentationsprozess in der Softwareentwicklung.
3. Kundenmitarbeit ist wichtiger als Vertragsverhandlungen. Die Einbeziehung der Kunden in den Entwicklungsprozess führt zu einer höheren Kundenzufriedenheit als formale Vertragsverhandlungen.
4. Reagieren auf Veränderungen hat Vorrang vor dem Befolgen eines Plans. Die Bereitschaft, sich anzupassen und auf unvorhersehbare Änderungen zu reagieren, ist besser als starr einem Plan zu folgen [Agile Manifesto 2001].

Die Agile Methodik wird in verschiedenen Projekttypen eingesetzt, darunter im Business Process Management, bei IoT-Projekten, bei der Entwicklung von E-Commerce-Webanwendungen und in Qualitätsmanagementprozessen der Softwareentwicklung. Studien zeigen, dass die Anwendung der agilen Methodik in IT- und Softwareentwicklungsprojekten die Erfolgsraten signifikant erhöht [Leong et al. 2023]. Im Vergleich zur Wasserfallmethode ist die Erfolgsrate von IT-Projekten um 24% höher und bei Softwareentwicklungsprojekten sogar doppelt so hoch [Chaudhari et al. 2021]. Weiterhin zeigen Forschungsergebnisse, dass Produkte in agilen Scrum-Projekten 37% schneller geliefert werden als beim Wasserfallmodell [Badakhshan et al. 2019]. Scrum, Crystal, Extreme Programming (XP) und Kanban repräsentieren verschiedene agile Methoden, die aus den vier grundlegenden Prinzipien der Agilität heraus entwickelt wurden, um eine breite Palette von Projektanforderungen abzudecken.

Diese Methoden bieten flexible Rahmenwerke, die Teams in die Lage versetzen, effektiver auf Veränderungen zu reagieren und die Produktentwicklung zu optimieren [Chaudhari und Joshi 2021]. Im Folgenden werden drei der bedeutendsten agilen Methoden - Scrum, Kanban und XP - näher betrachtet.

Scrum ist eine äußerst anpassungsfähige Methodik, die schnelle und iterative Entwicklungszyklen namentlich Sprints, die üblicherweise 2 bis 4 Wochen dauern, in den Mittelpunkt stellt. Scrum legt großen Wert auf Teamarbeit, Verantwortlichkeit und schrittweisen Fortschritt in Richtung eines klar definierten Ziels. Dabei werden Rollen wie der Product Owner, der Scrum Master und das Entwicklungsteam sowie Rituale wie tägliche Stand-up-Meetings, Sprint-Reviews und Retrospektiven genutzt [Schatten et al. 2010]. Die Scrum-Methodik setzt ein hohes Maß an Engagement und Erfahrung seitens aller Teammitglieder voraus. Teams, die neu bei Scrum sind oder denen es an Selbstmotivation und Disziplin fehlt, könnten Schwierigkeiten haben, die Methodik effektiv umzusetzen. Aufgrund seiner iterativen Natur und des Schwerpunkts auf Flexibilität können Projekte unter Scrum manchmal einen Scope Creep erleben. Ohne sorgfältiges Backlog-Management und Disziplin kann die Hinzufügung von Funktionen in jedem Sprint zu einer Projektverzögerung führen [Grimm 2003]. Scrum erfordert ständige Kommunikation und enge Zusammenarbeit zwischen allen Teammitgliedern und Stakeholdern. Projekte, denen es an diesem Interaktionsniveau mangelt, möglicherweise aufgrund von räumlich verteilten Teammitgliedern oder der Nichtverfügbarkeit von Stakeholdern, könnten Scrum als weniger effektiv empfinden. Die zahlreichen Meetings, die das Scrum-Framework vorsieht, sind zwar essenziell für den Prozess, können aber als zeitaufwändig wahrgenommen werden, besonders wenn sie nicht effizient verwaltet werden oder keinen Mehrwert bieten. Die Betonung auf schnelle Lieferung kann manchmal zu einer Überbetonung der Geschwindigkeit über die Qualität führen, wobei Teams möglicherweise unter Druck gesetzt werden, in jedem Sprint etwas zu liefern, was die Gründlichkeit der Tests oder die Vollständigkeit der Dokumentation beeinträchtigen kann [Plewan und Poensgen 2011]. Der Erfolg eines Scrum-Projekts hängt stark von den Fähigkeiten und der Führung des Scrum Masters ab. Ein weniger erfahrener Scrum Master könnte Schwierigkeiten haben, das Team vor externem Druck zu schützen, Konflikte effektiv zu managen oder das Team auf die Sprintziele zu fokussieren. Trotz zahlreicher Vorteile hängt der erfolgreiche Einsatz von Scrum vom Kontext der Organisation, der Art des Projekts und der Bereitschaft sowie der Fähigkeit des Teams und der Stakeholder ab, agile Prinzipien vollständig einzusetzen [Grimm 2003].

Kanban, ursprünglich aus der Fertigung kommend, wird in der Softwareentwicklung als visuelle Projektmanagementmethode eingesetzt, die mit einem Board und Karten Arbeitsaufgaben und deren Status darstellt. Der Fokus liegt auf der Begrenzung der in Bearbeitung befindlichen Arbeit, der Visualisierung der Arbeit und der Steigerung der Effizienz durch kontinuierliche Auslieferung [Grimm 2003]. Kanban, eine Methode, die nicht auf zeitgebundenen Sprints oder festgelegten Dauern für Aufgaben fokussiert, kann eine Herausforderung in Bezug auf Dringlichkeit oder Fristen für die Fertigstellung

von Arbeiten darstellen. Dies kann zu langsamem Fortschritt führen, insbesondere wenn Teammitglieder nicht eigenmotiviert sind oder wenn kein externer Druck besteht, Aufgaben voranzutreiben. Eine mögliche Überbetonung der aktuellen Arbeit kann zulasten der langfristigen Planung und strategischen Überlegungen gehen. Teams könnten zu sehr darauf konzentriert sein, das Board zu leeren und dabei zukunftsorientierten Aktivitäten vernachlässigen. Bei komplexen Projekten, die erhebliche Koordination erfordern oder viele voneinander abhängige Aufgaben haben, kann die Einfachheit von Kanban zu einer Einschränkung werden [Chaudhari und Joshi 2021]. Die Abhängigkeit von visuellen Managementwerkzeugen bedeutet, dass Teams Zugang zu diesen Werkzeugen haben und sie regelmäßig aktualisieren müssen. Ohne sorgfältige Pflege kann das Kanban-Board schnell veralten, was zu Verwirrung und Ineffizienzen führt. Eine ineffektive Begrenzung der in Arbeit befindlichen Aufgaben kann zu Überlastung und einer Verringerung der Gesamtproduktivität führen. Die Einführung von Kanban erfordert die Akzeptanz aller Teammitglieder und Stakeholder und kann auf Widerstand stoßen, besonders in Organisationen, die an hierarchischere oder strukturiertere Projektmanagementansätze gewöhnt sind. Die Messung der Leistung kann im Vergleich zu Methoden mit klaren definierten Phasen und Lieferobjekten herausfordernder sein und zusätzliche Metriken oder Werkzeuge erfordern. Kanbans Wirksamkeit hängt wesentlich von der Art der Arbeit, der Disziplin des Teams und der Fähigkeit ab, seine Prinzipien an den spezifischen Kontext des Projekts oder der Organisation anzupassen. Für Teams, die seine Herausforderungen meistern können, bietet Kanban jedoch einen flexiblen und effizienten Weg, Arbeitsprozesse zu verwalten und zu verbessern [Leong et al. 2023].

Extreme Programming (XP) betont die Kundenzufriedenheit durch die kontinuierliche Lieferung wertvoller Software. XP befürwortet Praktiken wie Pair-Programming, testgetriebene Entwicklung (engl. Test Driven Development, TDD), kontinuierliche Integration und häufige Veröffentlichungen, die die Produktivität steigern und Effizienzen einführen [Hruschka 2009]. Extreme Programming (XP) erfordert ein hohes Maß an Beteiligung sowohl vom Entwicklungsteam als auch vom Kunden. Von Kunden wird erwartet, dass sie ständig verfügbar sind, um Feedback zu geben und Entscheidungen zu treffen, was jedoch nicht immer machbar ist bzw. sich auf Lange zieht. Die Skalierung von XP auf größere Teams oder Projekte kann Koordinations- und Kommunikationsherausforderungen mit sich bringen. Die Betonung von XP auf Refactoring und kontinuierliche Integration könnte bei Projekten mit großen, komplexen oder Altsystemen, die nicht mit XP-Praktiken entwickelt wurden, Herausforderungen darstellen. Die Notwendigkeit der fortlaufenden Kundenbeteiligung und Praktiken wie Pair-Programming können die Betriebskosten eines Projekts erhöhen. Pair-Programming beispielsweise sieht vor, dass zwei Entwickler an einem einzigen Computer arbeiten, was als Verdopplung des Ressourceneinsatzes für denselben Output angesehen werden kann [Leong et al. 2023]. XP plädiert dafür „das Einfachste zu tun, das möglicherweise funktioniert“ und nur bei Bedarf Komplexität durch Refactoring hinzuzufügen. Dieser Ansatz kann manchmal zu einer Unterbetonung des vorausschauenden Designs führen, was zukünftig Skalierbarkeits- und Wartbarkeitsprobleme nach sich ziehen kann. Eine effektive Implementierung von XP erfordert ein erfahrenes und diszipliniertes Team,

insbesondere in der Einhaltung von Praktiken wie TDD und kontinuierlichem Refactoring. Weniger erfahrene Teams könnten Schwierigkeiten haben, XP-Praktiken effektiv anzuwenden. Der Fokus von XP auf Code über Dokumentation kann zu einem Mangel an ausreichender Dokumentation für die entwickelte Software führen, was Herausforderungen für neue Teammitglieder, Wartung oder Übergabe an andere Teams darstellen kann. Die intensive, schnelllebige Natur von XP, mit ihrem Schwerpunkt auf kurzen Entwicklungszyklen und häufigen Releases, kann zu Entwickler-Burnout führen, besonders wenn sie nicht sorgfältig verwaltet wird. Die iterative Natur von XP und die Betonung auf Flexibilität können es schwierig machen, Kosten und Zeitpläne genau zu schätzen, was bei Projekten mit festen Budgets und Zeitplänen problematisch sein kann. Obwohl Extreme Programming eine Reihe von Praktiken einführt, die die Softwareentwicklung verbessern sollen, hängt seine Wirksamkeit stark vom Projektkontext, der Teamzusammensetzung und der Fähigkeit ab, die inhärenten Herausforderungen der Methodik zu bewältigen [Leong et al. 2023].

Die Landschaft des Projektmanagements in der Softwareentwicklung hat sich erheblich weiterentwickelt, wobei neue Trends und Methoden aufgekommen sind, um den spezifischen Herausforderungen bei der Verwaltung von Softwareprojekten zu begegnen. Es zeichnet sich mittlerweile eine deutliche Tendenz zu **hybriden** Projektmanagementmethodologien ab, die traditionelle Praktiken mit agilen Prinzipien kombinieren, um mehr Flexibilität und Anpassungsfähigkeit in der Projektverwaltung zu bieten [Rosli 2022]. Dieser Ansatz ermöglicht es Teams, die strukturierte Planung der traditionellen Modelle mit der Reaktionsfähigkeit auf Veränderungen, wie sie für Agile Methoden charakteristisch ist, zu nutzen, was die Effizienz und Wirksamkeit bei der Projektabwicklung steigert. Diese hybriden Methoden stellen eine Verschmelzung der Vorteile beider Welten dar, indem sie die Vorhersagbarkeit und klare Struktur traditioneller Ansätze mit der dynamischen Anpassungsfähigkeit und Kundenorientierung agiler Verfahren vereinen [Leong et al. 2023]. Hybride Methodologien können stark variieren, doch im Folgenden werden einige gängige Ansätze untersucht:

Agile-Wasserfall-Hybrid kombiniert den strukturierten, phasenbasierten Ansatz des Wasserfallmodells für die Anfangsphasen eines Projekts (wie Anforderungen und Design) mit den iterativen, flexiblen Entwicklungs- und Testphasen agiler Methodologien [Reiff und Schlegel 2022]. Obwohl dieser Ansatz Flexibilität und Struktur bietet, bringt er auch mehrere Nachteile mit sich. Die Integration zweier grundsätzlich unterschiedlicher Methodologien kann die Projektverwaltung und -durchführung komplizieren. Teams stehen möglicherweise vor der Herausforderung, zu bestimmen, welche Elemente jeder Methodologie in verschiedenen Projektphasen angewendet werden sollen, was zu Verwirrung und Ineffizienz führen kann. Agile und Wasserfall besitzen unterschiedliche Kulturen und Praktiken, was bei der Integration zu kulturellen Konflikten innerhalb der Teams führen und die Aufrechterhaltung eines kohärenten Projektmanagementansatzes erschweren kann [Reiff und Schlegel 2022]. Die detaillierte Vorabplanung des Wasserfalls kann mit dem iterativen und flexiblen Planungsansatz von Agilen kollidieren. Das Ausbalancieren detaillierter Vorabplanung mit dem Bedürfnis nach Flexibilität ist schwierig und führt zu unzureichender Planung oder mangelnder Anpassungsfähigkeit.

Der Versuch, das Beste aus beiden Welten zu kombinieren, kann manchmal zu einer Überlastung durch Prozesse führen, die den Projektfortschritt verlangsamt und die Vorteile jedes Ansatzes verwässert. Stakeholder, die mit Agilen oder Wasserfall vertraut sind, haben möglicherweise feste Erwartungen an das Management und die Lieferung von Projekten. Der hybride Ansatz kann zu Verwirrung und unpassenden Erwartungen führen, was eine sorgfältige Steuerung und Kommunikation erfordert. Die Etablierung angemessener Erfolg-Metriken kann in einer hybriden Umgebung kompliziert sein, da der quantitative Fokus des Wasserfalls auf Lieferungen und Zeitplänen möglicherweise nicht gut mit dem qualitativen Fokus von Agilen auf Kundenzufriedenheit und Teamgeschwindigkeit übereinstimmt. Die erfolgreiche Implementierung eines hybriden Ansatzes erfordert ein tiefes Verständnis beider Methodologien. Ohne dieses Fachwissen könnten Teams Schwierigkeiten haben, die beiden effektiv zu integrieren, was zu suboptimalen Projektergebnissen führt [Gemino et al. 2021]. Trotz dieser Nachteile kann die Agile-Wasserfall-Hybridmethode für bestimmte Projekte effektiv sein, insbesondere für solche, die von der Struktur des Wasserfalls und der Flexibilität von Agilen profitieren. Der Erfolg dieses Ansatzes hängt oft von den spezifischen Anforderungen des Projekts, der Expertise des Teams und der Fähigkeit ab, die Komplexitäten der Integration zweier unterschiedlicher Methodologien zu navigieren.

ScrumBan basiert auf einer Verschmelzung von Scrum und Kanban. ScrumBan nutzt die Flexibilität und Visualisierung von Kanban zusammen mit der Struktur von Scrum, um die Arbeit in Umgebungen effektiver zu verwalten, in denen die zeitlich festgelegten Iterationen von Scrum als zu starr empfunden werden könnten [Mircea 2019]. ScrumBan bietet zwar einige Vorteile für Teams, die ihre Arbeit effizienter gestalten wollen, hat aber auch seine Nachteile. Die Verschmelzung zweier unterschiedlicher Methodologien kann Komplexität erzeugen, was es für Teams schwierig macht, die strukturierten Sprints und Rollen von Scrum mit dem kontinuierlichen Fluss und den WIP-Limits von Kanban (engl. Work in Progress, WIP. – Die offenen Aufgaben, an denen ein Team gerade arbeitet.) nahtlos zu integrieren. Diese Komplexität kann besonders für Teams, die mit einer der Methodologien neu sind, zu Verwirrung führen. Teams, die an die strikten Iterationen von Scrum gewöhnt sind, könnten sich schwertun, sich an das kontinuierliche Flussmodell von Kanban anzupassen und umgekehrt. Der erforderliche kulturelle Wandel, um die Disziplin von Scrum mit der Flexibilität von Kanban in Einklang zu bringen, kann erheblich sein und möglicherweise Widerstand unter den Teammitgliedern hervorrufen. ScrumBan versucht, die Planungs- und Überprüfungsmechanismen von Scrum mit dem visuellen Workflow-Management von Kanban in Einklang zu bringen. Ohne sorgfältige Implementierung kann dies jedoch zu einem zusätzlichen Prozessaufwand führen, der die Effizienzsteigerungen, die die hybride Methode bieten soll, schmälert. Die Anpassung traditioneller Scrum-Rollen an einen dynamischeren Ansatz kann zu Rollenkonfusion führen, was zu Mehrdeutigkeit und Ineffizienz führen kann [Mircea 2019]. Ähnlich wie bei Scrum und Kanban kann die Skalierung von ScrumBan auf größere Projekte oder über mehrere Teams hinweg zusätzliche Komplexitäten einführen. Die

hybride Natur von Scrumban kann es schwierig machen, Konsistenz und Kohärenz bei größeren Implementierungen ohne eine gut durchdachte Skalierungsstrategie aufrechtzuerhalten. Teams, die zu Scrumban wechseln, benötigen möglicherweise erhebliche Schulungen und Anpassungszeiten, was die Produktivität vorübergehend reduzieren kann, während Teams lernen, die Nuancen der kombinierten Methodik zu navigieren [Quelle]. Im Gegensatz zu Scrum, was eine präskriptive Reihe von Praktiken und Zeremonien hat, beruht Scrumban mehr auf der Fähigkeit des Teams, seine Prozesse anzupassen. Dieser Mangel an präskriptiven Richtlinien kann für Teams, die mehr Struktur bevorzugen oder weniger Erfahrung mit agilen Praktiken haben, nachteilig sein. Die richtige Balance von Metriken aus Scrum und Kanban zu finden, um Leistung und Fortschritt effektiv zu messen, stellt eine Herausforderung dar. Trotz dieser Nachteile kann Scrumban eine leistungsstarke Methodik für Teams sein, die nach der Flexibilität suchen, ihre Arbeit adaptiver zu verwalten, während sie etwas Struktur beibehalten. Erfolgreiche Implementierung von Scrumban erfordert ein klares Verständnis der Prinzipien von Scrum und Kanban sowie ein Engagement für kontinuierliche Verbesserung und Anpassung [Leong et al. 2023].

Lean-Agile-Hybrid vereint Prinzipien aus der Lean-Produktion und der agilen Softwareentwicklung, mit dem Ziel, durch Minimierung von Verschwendungen (ein Lean-Prinzip) maximalen Wert zu erzeugen, während gleichzeitig Flexibilität und Reaktionsfähigkeit auf Veränderungen (ein Agile-Prinzip) beibehalten werden [Hruschka 2009]. Trotz potenzieller Vorteile bringt dieser hybride Ansatz verschiedene Herausforderungen und Nachteile mit sich. Die Integration der Effizienz- und Verschwendungsreduktion von Lean mit der Anpassungsfähigkeit und iterativen Entwicklung von Agilen kann komplex sein. Teams könnten Schwierigkeiten haben, die unterschiedlichen Prioritäten und Praktiken jeder Methodik in Einklang zu bringen, was zu Verwirrung und Ineffizienz führen kann [Richter 2003]. Lean und Agile besitzen jeweils eigene Kulturen. Lean konzentriert sich auf die Optimierung des Flusses und die Reduzierung von Verschwendungen im gesamten Wertschöpfungsstrom, oft mit einem Top-Down-Ansatz zur kontinuierlichen Verbesserung. Agile betont Teamermächtigung, Kundenkollaboration und Reaktion auf Veränderungen. Die Verschmelzung dieser Kulturen erfordert erhebliche Anpassungen und kann auf Widerstand von Teammitgliedern stoßen, die an eine Methodik über die andere gewöhnt sind [Hruschka 2009]. Die Skalierung von Agile-Methoden wie Scrum ist für Softwareentwicklungsteams konzipiert und kann mit Frameworks wie SAFe (engl. Scaled Agile Framework) skaliert werden, aber die Integration von Lean-Prinzipien in größerem Maßstab kann Komplexität einführen. Es kann schwierig sein, sicherzustellen, dass Lean-Prinzipien wie Flussoptimierung und Verschwendungsreduktion effektiv im Kontext der agilen Softwareentwicklung über mehrere Teams oder Abteilungen hinweg angewendet werden. Es besteht das Risiko, dass der Lean-Fokus auf Effizienz und Verschwendungsreduktion die Betonung von Agilen auf Flexibilität und Reaktionsfähigkeit überlagern könnte. Dies könnte dazu führen, dass Prozessoptimierung über Kundenwert oder Innovation priorisiert wird, was Kreativität und Anpassungsfähigkeit potenziell ersticken. Teams benötigen möglicherweise erhebliche Schulungen, um Lean- und Agile-Prinzipien zu verstehen und effektiv

umzusetzen. Die Übergangszeit kann zu vorübergehenden Produktivitätseinbußen führen, während sich Teams an neue Arbeitsweisen anpassen [Richter 2003]. Die Lean-Agile-Hybridmethode erfordert einen Ausgleich zwischen dem Lean-Schwerpunkt auf langfristiger Prozessverbesserung und dem Agile-Fokus auf kurzfristiger Anpassungsfähigkeit und Kundenfeedback. Es ist schwierig, das richtige Gleichgewicht zu finden, um sowohl den sofortigen Projekterfolg als auch langfristige Effizienzsteigerungen zu gewährleisten. Der Versuch, sowohl Lean- als auch Agile-Praktiken anzuwenden, führt ebenso zu erhöhter Prozesskomplexität, welche die Entscheidungsfindung und Ausführung verlangsamt, wenn sie nicht sorgfältig verwaltet wird. Trotz dieser Herausforderungen kann die Lean-Agile-Hybridmethode für Organisationen, die ihre Softwareentwicklungsprozesse verbessern möchten, indem sie die Stärken von Lean und Agile kombinieren, erhebliche Vorteile bieten. Erfolg mit diesem Ansatz erfordert typischerweise ein klares Verständnis beider Methodologien, ein Engagement für kontinuierliche Verbesserung und die Flexibilität, Praktiken an den spezifischen Kontext und die Ziele der Organisation anzupassen [Chaudhari und Joshi 2021].

Nach der Analyse von Stand der Wissenschaft und Technik bezüglich der drei Projektmanagement-Methodologien (traditionell, agil, und hybrid) und deren spezifischen Methoden, sind folgende sechs Kategorien bezüglich der Schwierigkeiten und Nachteile von PM-Methoden identifiziert worden:

1. **Flexibilitätsproblem:** Herausforderungen bei der Anpassung an Änderungen und Integration neuer Anforderungen im Projektverlauf.
2. **Kommunikations- und Kooperationsherausforderungen:** Herausforderungen bei der effektiven Kommunikation und Zusammenarbeit innerhalb des Teams und mit Stakeholdern.
3. **Risikomanagementproblem:** Späte Erkennung und unzureichende Bewältigung von Risiken, was zu Zeit- und Kostenüberschreitungen führen kann.
4. **Problem bei der Planung und Schätzung:** Herausforderungen bei der realistischen Planung und Schätzung von Zeit und Kosten, insbesondere bei komplexen oder großen Projekten.
5. **Implementierungsprobleme:** Herausforderung bei der effektiven Implementierung der Methodik aufgrund von Komplexität, erforderlichem Wissen oder Teamzusammensetzung.
6. **Problem der Prozess- und Methodenintegration:** Herausforderungen bei der Integration verschiedener Ansätze und Methoden innerhalb eines Projekts oder zwischen Projekten.

Die oben genannten Probleme werden anschließend in Kapitel 4.2.1 als Grundlage für die Definition der Anforderungen an das in dieser Dissertation neu zu entwickelndem Konzept dienen.

Ein weiteres Element erfolgreicher kompetenzbasierten Softwareentwicklungsprojekte ist die genaue Messung und Bewertung der Kompetenzen der beteiligten Mitarbeiter. Nachdem die wissenschaftlichen Ansätze zum Projektmanagement in der Softwareentwicklung erläutert wurden, richtet sich der Fokus nun auf die wissenschaftlichen Ansätze zur Kompetenzmessung. Diese sind entscheidend, um die notwendigen Fähigkeiten und Kenntnisse der Mitarbeiter systematisch zu erfassen und zu entwickeln, was wiederum die Grundlage für effektive Projektplanung und -durchführung bildet.

3.2 Wissenschaftliche Ansätze zur Kompetenzmessung in Software Entwicklung

In lediglich 20 % der Fälle nutzen Unternehmen systematische Verfahren zur Erfassung von Daten bei Personalentscheidungen. Die übrigen 80 % basieren auf intuitiven Entscheidungen. Es sind nicht die präzisesten Methoden, die sich in der beruflichen Praxis durchgesetzt haben, sondern jene, die angemessen und zeitnah Unterstützung bei Entscheidungen bieten [Erpenbeck et al. 2017]. Die Fähigkeit zur Selbstorganisation – die Kompetenzen einer Person – kann gemessen und zertifiziert werden. Diese Messung bildet die Grundlage für die Entwicklung und das Management von Kompetenzen. Dies ist eine essenzielle Voraussetzung für jede gezielte Kompetenzentwicklung sowie für die Entwicklung von Kompetenzen im digitalen Raum, wie Softwareentwicklungsprojekte [Sauter und Staudt 2016a]. Die Kompetenzen können anhand unterschiedlicher Formen erfasst werden. Die Formen der Kompetenzmessung unterteilen sich in folgenden Kategorien:

1. Kompetenztests: quantitative Messungen,
2. Kompetenzpass: qualitative Messungen,
3. Kompetenzbiografie: komparative Beschreibung,
4. Kompetenzsimulation: simulative Szenarien sowie
5. Kompetenzsituation: Arbeitsproben, Beobachtungen am Arbeitsplatz [Erpenbeck et al. 2017].

Es haben sich unterschiedlichen Methoden der Kompetenzmessung für die unterschiedlichen Formen der Kompetenzmessung etabliert. Die Tabelle 6 stellt die fünf **Formen** der Kompetenzmessung den zehn **Methoden** der Kompetenzmessung.

Tabelle 6: Gegenüberstellung Kompetenzformen u. -methoden in Anlehnung an [Erpenbeck et al. 2017]

Die Formen der Kompetenzmessung	Methoden der Kompetenzmessung
Kompetenztests	Persönlichkeitsverfahren, Testverfahren
Kompetenzpass	Biografische Methoden, Tätigkeitsanalyse, Selbstbeschreibung, Fremdbeschreibung
Kompetenzbiografie	Biografische Methoden, Interviews, Selbstbeschreibung
Kompetenzsimulation	Arbeitsproben, Assessment Center (AC-Verfahren), Computergestützte Simulationen
Kompetenzsituation	Fremdbeschreibung, Interviews

Im Kontext der Vielfalt etablierter Methoden zur Kompetenzmessung, zeigt die Kompetenzdiagnostik auf, wie Individuen bei neuen Herausforderungen selbstständig und kreativ handeln können. Während Unternehmen häufig auf eignungsdiagnostische Instrumente wie biografische Fragebögen oder Assessment Center zurückgreifen, um Mitarbeiter zu selektieren oder zu platzieren, mangelt es diesen Verfahren oft an der Fähigkeit, echte Kompetenzen zu messen und die Selbstorganisationsfähigkeit zu erfassen. So verweisen klassische Tests der Intelligenz und Leistungsfähigkeit, die auf kognitive

sowie nicht-kognitive Fähigkeiten abzielen, selten auf die praktische Handlungskompetenz in anspruchsvollen Situationen und sind somit für eine umfassende Kompetenzbeurteilung ungeeignet. Allerdings können systematische Intelligenztests nützlich sein, um Basisvoraussetzungen für die gezielte Entwicklung von Kompetenzen zu ermitteln, und bilden somit eine Brücke zwischen der Theorie der Kompetenzmessung und ihrer praktischen Anwendung [Lang-von Wins 2016]. Nachfolgend werden gängige Kompetenzmessmethoden erklärt und analysiert.

Die Anwendung **biografischer Methoden** verdeutlicht, wie lebensgeschichtliche Daten der Menschen wertvolle Hinweise auf die Kompetenzen bieten können, insbesondere wenn sich bestimmte Verhaltensmuster und Reaktionen über längere Zeiträume konsistent zeigen. Diese Methoden ermöglichen zudem die Erfassung von Motivationen und Wertvorstellungen. Sie fokussieren sich auf bisher erbrachte Leistungen, die als zuverlässige Indikatoren für zukünftiges Verhalten angesehen werden, wodurch biografische Ansätze eine signifikante prognostische Relevanz aufweisen [Lang-von Wins 2016]. Die Vielfalt der biografischen Methoden ist beträchtlich, basierend auf einem breiten Spektrum unterschiedlicher Techniken. Qualifikationsnachweise wie Zeugnisse und Zertifikate dokumentieren lediglich erlangte Qualifikationen. Bei standardisierten, biografischen Fragebögen, die von Mitarbeitern selbst ausgefüllt werden, besteht das Risiko subjektiver Angaben, da diese oft aus Lebensläufen abgeleitet sind und auch persönliche Einschätzungen, Emotionen und Interessen umfassen können. Im Gegensatz dazu, ist die Qualität der Daten aus standardisierten biografischen Interviews hoch, erfordern jedoch zeitintensive Terminabsprachen sowie einen erheblichen Aufwand für Erfassung und Auswertung, weshalb sie selten eingesetzt werden. Die Kompetenzbiografie nach Erpenbeck und Heyse, eine Kombination aus Fragebogenverfahren und problemzentrierten Interviews, zielt darauf ab, Potenziale für die Kompetenzentwicklung aufzudecken, wobei Werthaltungen und Kompetenzen zentral sind [Heyse 2007]. Dieses Verfahren, das biografische Muster der Kompetenzentwicklung detailliert erfasst, gilt als besonders [Lang-von Wins 2009]. Die Kompetenzbilanz, ebenfalls biografisch fundiert, basiert auf psychologischen Modellen, die vergangenes Handeln in die Zukunft projizieren, und kann durch Coaching unterstützt werden. Auch dieser Ansatz fördert eine selbstorganisierte Kompetenzentwicklung der Mitarbeiter. Das Verfahren der Kompetenzbilanz ermöglicht es Mitarbeitern, auf Grundlage ihrer erkannten und in der Biografie verankerten Kompetenzen, initiativ zu handeln und sowohl ihre berufliche als auch private Weiterentwicklung eigenständig zu lenken. Ursprünglich für die Unterstützung bei beruflichen Neuorientierungen konzipiert, findet die Kompetenzbilanz mittlerweile auch in der Personalauswahl und -entwicklung innerhalb von Organisationen Anwendung [Kauffeld 2006].

Das Verfahren der **Tätigkeitsanalyse** zielt darauf ab, Aufgaben und Teiltätigkeiten einer Person, einer Abteilung oder einer Gruppe zu identifizieren und zu bewerten. Der primäre Zweck dieser Methode ist die Steigerung der Produktivität, die Gewährleistung der Sicherheit und die Sicherstellung der Qualität. Auf Basis der Analyseergebnisse werden erforderliche Kompetenzprofile definiert und eine Überprüfung der vorhandenen Kompetenzen (Ist-Zustand) ermöglicht. Obwohl das Verfahren

bei sehr komplexen Strukturen und Prozessen an seine Grenzen stößt, eignet es sich besonders gut für den Vergleich von Soll- und Ist-Zuständen auf individueller Ebene [Sauter und Staudt 2016a].

Interviews sind das am weitesten verbreitete Mittel zur Bewertung von Bewerbern und umfassen verschiedene Formen der Befragung, darunter strukturierte, teilstrukturierte und unstrukturierte Interviews, die sich hinsichtlich des Aufbaus und der Durchführung unterscheiden [Sauter und Staudt 2016a]. Strukturierte Interviews fördern den Vergleich zwischen Kandidaten und gewährleisten durch ein systematisches Vorgehen die Erhebung aller notwendigen Informationen. Die Praxis, zwei Interviewer einzusetzen, verbessert die Objektivität der Beurteilung. Dabei ist es essenziell, klare Kompetenzdefinitionen, den strukturierten Aufbau der Fragen sowie vordefinierte, handlungsorientierte Bewertungskriterien zu berücksichtigen. Eine gezielte Strukturierung des Interviews soll die adäquate Erhebung der angestrebten Informationen sicherstellen. Durch das Mehraugenprinzip wird die Subjektivität einzelner Beobachterperspektiven ausgeglichen, um Verzerrungen durch Sympathie oder Antipathie sowie individuelle Wahrnehmungsunterschiede zu minimieren. Diese Rahmenbedingungen ermöglichen es, die persönlichen Kompetenzen der Bewerber für die Bewältigung komplexer Aufgaben zu erfassen. Die Multimodale Interviews kombinieren verschiedene Methoden zur Einschätzung von Kompetenzen, indem etwa durch situative Fragen mentale Arbeitsproben genommen werden. Ein spezieller Teil des Interviews dient dem Erfragen von Interessen und Werten der Bewerber. Dieser Beurteilungsprozess ist stark kommunikativ geprägt und kann durch das Feedback direkt den Kompetenzentwicklungsprozess beeinflussen. Auch dieses Verfahren eignet sich zur Kompetenzbeurteilung, sofern die Gesprächsführung systematisch und nicht intuitiv erfolgt [Lang-von Wins 2016].

Einige Persönlichkeitsmerkmale können unter der Voraussetzung spezifischer und detaillierter Hypothesen über den Zusammenhang zwischen diesen Eigenschaften bzw. Strukturen und den entsprechenden Kompetenzmustern gewisse Vorhersagen über Verhalten in anspruchsvollen Situationen ermöglichen. Sie können allerdings eine direkte Kompetenzmessung nicht ersetzen. **Persönlichkeitsverfahren** bzw. Persönlichkeitstests und -fragebögen zeichnen die typischen Verhaltensweisen einer Person auf, wobei das „Big Five“-Modell (Fünf-Faktoren-Modell) als das bekannteste Persönlichkeitsmodell gilt und Basis für die Entwicklung vieler weiterer Persönlichkeitsverfahren ist [Lang-von Wins 2009]. Es umfasst die fünf zentralen Dimensionen der Persönlichkeit: Neurotizismus, Extraversion, Offenheit, Gewissenhaftigkeit und Verträglichkeit. Die Hauptdimensionen der Persönlichkeit werden dann in zwei Pole klassifiziert: schwach und stark ausgeprägt. Bei einer schwachen Ausprägung von Neurotizismus zeigen Menschen sich selbstsicher und ruhig, während eine starke Ausprägung zu Emotionalität und Verletzlichkeit führt. Bei der Dimension Extraversion sind zurückhaltende und reservierte Züge charakteristisch für eine schwache Ausprägung, wohingegen Geselligkeit eine starke Ausprägung kennzeichnet. Offenheit für Erfahrungen reicht von konsistentem und vorsichtigem Verhalten bei einer schwachen Ausprägung bis hin zu erfiederischem und neugierigem Verhalten bei

starker Ausprägung. In der Dimension Gewissenhaftigkeit steht Unbekümmertheit und Nachlässigkeit auf der einen Seite und Effektivität sowie Organisation auf der anderen. Schließlich reflektiert Verträglichkeit bei einer schwachen Ausprägung eine kompetitive und misstrauische Natur, während eine starke Ausprägung Kooperativität, Freundlichkeit und Mitgefühl hervorbringt [Kauffeld 2006]. Obwohl die Persönlichkeit eine wichtige Rolle bei der Entwicklung von Kompetenzen spielt, stellen Persönlichkeitsmerkmale allein noch keine Kompetenzen dar. Erst durch ihre Manifestation in Handlungen werden sie zu erkennbaren Kompetenzen. Eine hohe Kommunikationsfähigkeit mag auf eine besondere Befähigung in der Kundenakquise hindeuten, jedoch garantiert dies nicht das Vorhandensein anderer, möglicherweise ebenso entscheidender Kompetenzen, wie zum Beispiel im Bereich der Aktivitäts- oder Fachmethodik. Da das Verhalten von Mitarbeitern stets auch von der jeweiligen Situation abhängt, müssen die Ergebnisse aus Persönlichkeitsfragebögen im Kontext spezifischer situativer Anforderungen interpretiert werden, um für die Kompetenzdiagnostik aussagekräftig zu sein [Lang-von Wins 2016].

Das Assessment beschreibt die Evaluation von Personen hinsichtlich spezifischer Merkmale oder Kompetenzen mittels verschiedener Methoden, darunter standardisierte Tests, Interviews und biografische Befragungen. **Assessment Center** (AC-Verfahren) stellen ein Verfahren dar, bei dem Kandidaten durch mehrere Beurteilende in verschiedenen Situationen, wie Arbeitsproben, Rollenspielen, Präsentationen oder Gruppenaufgaben, bewertet werden. Ziel ist es, durch die Kombination verschiedener Methoden die Validität der Beurteilung zu steigern und genauere Informationen zu gewinnen. Die Beurteilenden protokollieren ihre Eindrücke anhand definierter Kriterien und Regeln. Eine vorherige Schulung der Beurteilenden gilt als bewährte Praxis. Die Stärke von Assessment Centern liegt in der Möglichkeit, die Bewerber in realitätsnahen Situationen zu erleben und so ihr Verhalten zu beobachten. Dennoch sind Assessments mit hohem Aufwand verbunden und bieten nur bedingt Rückschlüsse auf tatsächliche Kompetenzen, da die simulierten Szenarien die Realität nur annähern können. Zudem besteht die Gefahr, dass geübte Teilnehmer lediglich lernen, sich in Assessment Centern besser zu präsentieren, ohne dass dies zwingend eine Verbesserung der realen Kompetenzen widerspiegelt. Eine mögliche Weiterentwicklung der Assessment-Center-Methodik könnte die direkte Beobachtung von Bewerbern oder Mitarbeitern in ihrem realen Arbeitsumfeld sein. Hier würden geschulte Beobachter das Verhalten über einen längeren Zeitraum nach festgelegten Kriterien evaluieren, was die Validität der Kompetenzmessung erheblich steigern könnte [Lang-von Wins 2016].

Die zwei Verfahren der **Selbst- und Fremdbeschreibung** werden häufig kombiniert angewandt, um eine umfassende Kompetenzeinschätzung zu ermöglichen. Bei der Selbstbeschreibung bewerten Mitarbeiter ihr eigenes Verhalten unter verschiedenen Bedingungen mithilfe von Selbsttests oder Selbstbeurteilungen. Diese introspektiven Methoden können jedoch durch subjektive Faktoren wie Erinnerungslücken oder das Streben nach sozialer Erwünschtheit beeinflusst werden, weshalb sie oftmals in ergänzende Verfahren wie die Kompetenzbiografie integriert werden. Im Rahmen der Fremdbeschreibung bietet das 360-Grad-Feedback die Möglichkeit, Bewertungen durch verschiedene Personen aus

dem Arbeitsumfeld des Mitarbeiters, einschließlich Vorgesetzten, Kollegen und Coaches, zu erhalten. Diese Methode unterliegt jedoch potenziellen Verzerrungen, wie dem Halo-Effekt, der die objektive Beurteilung durch persönliche Sympathien oder Antipathien verfälschen kann. Die Kombination von Selbst- und Fremdbeschreibung ermöglicht eine ausgeglichene Perspektive auf die Kompetenzen eines Mitarbeiters und legt den Grundstein für dessen individuelle Kompetenzentwicklung [Kauffeld 2006].

Im **Arbeitsproben**-Verfahren bearbeitet der Mitarbeiter innerhalb eines festgelegten Zeitrahmens berufsbezogene Aufgaben in einem simulierten Umfeld unter der Aufsicht eines Lernbegleiters. Dies ermöglicht es dem Mitarbeiter, neue Handlungsstrategien zu erproben, ohne reale Geschäftsprozesse zu beeinträchtigen. Wesentlich ist dabei, dass kritische und repräsentative Situationen des Berufsalltags ausgewählt und in einem kontrollierten Rahmen nachgestellt werden [Lang-von Wins 2016].

Testverfahren sind Messinstrumente, die darauf abzielen, spezifische Leistungs- oder Persönlichkeitsmerkmale quantitativ zu erfassen. Diese Verfahren variieren und umfassen Typen wie Fähigkeits-, Leistungs- oder Persönlichkeitstests. Oft konzentrieren sie sich auf die Qualifikationsebene und erfassen nicht die Kompetenzen, die für die Lösung von Problemen im Arbeitsprozess notwendig sind. Die genauen Merkmale, die ein Testverfahren erfassst, sind abhängig von der Art des Tests und dessen spezifischem Einsatzgebiet [Sauter und Staudt 2016a].

Computergestützte Simulationen haben das Potential, ein effektives Mittel zur Beurteilung von Kompetenzen in komplexen Szenarien zu sein, die für den Mitarbeiter kaum von der Wirklichkeit zu unterscheiden sind. Aktuell sind jedoch die meisten Computersysteme noch nicht in der Lage, solch realistische Situationen zu erzeugen, was zur Folge hat, dass sie eher zur Messung von Qualifikationen als von Kompetenzen geeignet sind. Es wird erwartet, dass zukünftige Entwicklungen in der humanoiden Computertechnologie diese Lücke schließen könnten. Eine Ausnahme stellen heute schon hochentwickelte Simulatoren wie Flug- oder Fahrsimulatoren dar, die zur Kompetenzentwicklung in spezifischen Bereichen wie der Pilotenausbildung oder bei der Deutschen Bahn verwendet werden. Computerbasierte Systeme werden indessen immer häufiger als Plattformen für alle Arten der Kompetenzmessung eingesetzt, von der Dateneingabe über die Auswertung bis hin zur kollaborativen Aufgabenbewältigung und Kommunikation online [Sauter und Staudt 2016b].

Die folgende Tabelle 7, fasst die Vorteile und Nachteile der unterschiedlichen Verfahren zusammen.

Tabelle 7: Vor- und Nachteile der gängige Kompetenzmessungsmethoden

Verfahren	Vorteil	Nachteil
Biografische Methoden	<ul style="list-style-type: none"> Die biografischen Informationen sind vorhanden. Identifizierung von Potentialen für die Entwicklung von Kompetenzen, häufig in Kombination mit Verfahren, die auf die Zukunft ausgerichtet sind. 	<ul style="list-style-type: none"> Kompetenzen werden oft nur bruchstückhaft und vorwiegend subjektiv bewertet. Es wird von einem konstanten Verhaltensmuster ausgegangen. Die Durchführung ist mit erheblichem Aufwand verbunden. Der Fokus liegt auf der Analyse vergangener Leistungen. In der Regel erfolgt eine ausschließlich qualitative Analyse. Das Dilemma, Antworten zu geben, die sozial akzeptiert sind, besonders bei Interviews und Kompetenzbilanzen.
Tätigkeitsanalyse	<ul style="list-style-type: none"> Dient als solide Basis für die Analyse der Kompetenzen von Individuen und Gruppen. Potenziale für die Kompetenzentwicklung werden deutlicher erkennbar. 	<ul style="list-style-type: none"> Eine Beurteilung von Kompetenzen ist nicht durchführbar. Die Ergebnisse sind nicht für den Einsatz über Organisationsgrenzen hinweg geeignet. In komplexen Strukturen gestaltet sich der Prozess besonders arbeitsintensiv. Die Beurteilung der Kompetenzen basiert ausschließlich auf subjektiven Wahrnehmungen. Es wird von dauerhaft konstanten Verhaltensmustern und Handlungen ausgegangen. Das Risiko einer Beeinflussung durch den Fragenden besteht, bedingt durch die Art der Fragestellung und den Tonfall. Potenziale zur Kompetenzentwicklung werden nur begrenzt ersichtlich.
Interview	<ul style="list-style-type: none"> Durch ein strukturiertes Verfahren und festgelegte Bewertungskriterien ermöglicht es den Vergleich und die Beurteilung relevanter Verhaltensweisen. Eine Einschätzung der Kompetenzen ist prinzipiell durchführbar. 	<ul style="list-style-type: none"> Da Persönlichkeitseigenschaften keine Kompetenzen darstellen, lassen sich daraus keine Schlüsse auf tatsächliche Kompetenzen ziehen. Schwerpunkt liegt auf außerfachlichen Kompetenzaspekten. Risiko von Erinnerungslücken oder Antworten, die sozial erwünscht sind.
Persönlichkeitsverfahren	<ul style="list-style-type: none"> Darstellung charakteristischer Verhaltensmuster. Vorhersage potenzieller Verhaltensweisen bei zukünftigen Herausforderungen. Wirtschaftlich effizientes Verfahren. 	<ul style="list-style-type: none"> Messung von Kompetenzen ist nicht durchführbar. Der Prozess ist mit erheblichem Aufwand verbunden. Risiko von Fehlern bei der Beobachtung.
Assessment-Center	<ul style="list-style-type: none"> Auswahlverfahren durch Übungen, die der Praxis nahekommen. Auswahl basierend auf dem spezifischen Kontext der Situation. Breites Spektrum an Informationen. 	

Verfahren	Vorteil	Nachteil
Selbstbeschreibung	<ul style="list-style-type: none"> • Darstellung charakteristischer Verhaltensweisen. • Wirtschaftlich effizient anwendbar. • Impulse für Reflexionsprozesse werden gegeben. • Aufdeckung von Möglichkeiten zur Kompetenzentwicklung. • Dient als Basis für die Bewertung durch Dritte. 	<ul style="list-style-type: none"> • Bewertung basiert auf persönlicher Wahrnehmung. • Potenzielle Fehlerquellen wie Gedächtnislücken oder das Streben nach sozialer Akzeptanz. • Ein Vergleich über verschiedene Unternehmen hinweg ist nicht durchführbar.
Fremdbeschreibung	<ul style="list-style-type: none"> • Darstellung üblicher Verhaltensmuster. • Einbeziehung unterschiedlicher Sichtweisen (z.B. von Vorgesetzten, Mitarbeitern) ist umsetzbar. 	<ul style="list-style-type: none"> • Potenzielle Fehler durch Sympathie- oder Halo-Effekt. • Risiko von Fehlern bei der Beobachtung.
Arbeitsproben	<ul style="list-style-type: none"> • Hohe Gültigkeit in spezifischen Handlungsfeldern. • Unterstützung durch Lernbegleitung ist gegeben. • Erprobung neuer Verhaltensweisen ohne Risiken für reale Abläufe. 	<ul style="list-style-type: none"> • Der Prozess erfordert einen hohen Einsatz von Ressourcen. • Eine vorherige Schulung der Beobachtenden ist erforderlich. • Risiko von Fehlern in der Beobachtung. • Für komplexe praktische Anforderungen weniger geeignet.
Testverfahren	<ul style="list-style-type: none"> • Einfach in der Anwendung. • Schnelle Ergebnisauswertung. • Hoher Grad an Standardisierung. • Wirtschaftlich effiziente Durchführung. • Anspruchsvolle methodische Grundlagen. 	<ul style="list-style-type: none"> • In der Regel werden Qualifikationen oder Persönlichkeitseigenschaften, jedoch keine Kompetenzen erfasst. • Oftmals geringe Zustimmung seitens der bewerteten Personen.
Computerunterstützte Szenarien	<ul style="list-style-type: none"> • Vielfältige Szenarien und Aufgabenstellungen realisierbar. • Fortschritte in der humanoiden Computertechnologie versprechen die Schaffung realistischer Herausforderungen. 	<ul style="list-style-type: none"> • Begrenzte Kapazitäten aktueller Computer-technologien. • In der Regel nur eine Betrachtung von Teilausschnitten der Wirklichkeit. • Werden Simulationen nicht als realitätsnahe Herausforderungen wahrgenommen, lassen sich lediglich Qualifikationen erfassen. • Das soziale Umfeld wird in Simulationen oft nicht abgebildet.

Die genannten Aspekte dienen als Grundlage für die Definition der Anforderungen an das in dieser Arbeit zu entwickelndem Konzept. Die Anforderungen werden in Kapitel 4.2 diskutiert.

3.3 Wissenschaftliche Ansätze zur KI-gestützten Projektplanung

Die Unternehmen stehen bei Industrieprojekten vor der Herausforderung, komplexe interne sowie externe organisatorische und produktbezogene Anforderungen zu meistern. Sie müssen schnell auf Veränderungen reagieren, um wettbewerbsfähig zu bleiben, was durch die schwierige Vorhersagbarkeit der Projektlage und die Fehlentscheidungen in frühen Phasen des Projektes erschwert wird. Häufig führt dies zu ineffizienter Ressourcenzuweisung und Verfehlung von Stakeholder-Erwartungen, was nachträgliche Kosten verursacht und den Ruf des Unternehmens schädigt. In dieser Dissertation

wird daher einen systematischen Ansatz für die kompetenzbasierte Projektplanung in Entwicklungsprojekten entwickelt, der moderne IT-Lösungen zur Unterstützung nutzt. Dieser Ansatz ermöglicht durch Analyse von Qualitätsmanagementdaten und historischen Projektinformationen, präzisere Ressourcenschätzungen für zukünftige Projekte. Die Forschung untersucht das Potenzial von Natural Language Processing (NLP) innerhalb eines KI-Workflows für die Projektplanung.

Wie in Kapitel 1.1 dargelegt, haben sich NLP-Methoden in Medizin, Wissensmanagement und Ingenieurwissenschaften weit verbreitet. Auffällig ist jedoch der Mangel an Forschung zu NLP in der kompetenzbasierten Planung von Softwareentwicklungsprojekten. Für diese Untersuchung wurden 10 internationale und 2 nationale Projekte identifiziert, die sich entweder direkt mit dieser Thematik auseinandersetzen oder ihr nahestehen. Um diese Arbeit von existierenden abzuheben und den Forschungsbedarf bei KI-unterstützten Prozessen in der kompetenzbasierten Projektplanung hervorzuheben, wurden fünf relevante Themenbereiche festgelegt und in Kapitel 1.1 umfassend beschrieben. Die 12 ausgewählten Studien werden in diesem Kontext detailliert untersucht (siehe Tabelle 2 in Kapitel 1.1) und hinsichtlich ihrer Vor- und Nachteile analysiert.

Studie 1 gemäß [Zwirn 2013] hebt hervor, dass das Identifizieren von Anforderungen aus Texten in natürlicher Sprache eine essenzielle Aufgabe in Industrieprojekten darstellt. Die manuelle Erfassung solcher Anforderungen gestaltet sich allerdings als aufwendig und anfällig für Fehler. Im Rahmen dieser Forschung wurde ein System entwickelt, welches Anforderungen nach ihrer Bedeutung analysiert und bewertet. Dieses System, benannt als Requirement Extractor and Classifier (REC), prüft Sätze auf das Vorhandensein von Anforderungen, indem es diese mittels einer Ontologie klassifiziert. Die Ontologie ordnet Sätze in Kategorien wie Anforderungen, schwache Anforderungen, optionale Anforderungen, Anforderungskandidaten und Nicht-Anforderungen ein, basierend auf der Struktur aus Subjekt, Verb und Objekt (SVO-Tripel) zur Identifikation potenzieller Klassifikationskandidaten. Anschließend erfolgt eine weitere Differenzierung der Kandidaten in Klassen mittels spezifischer Schlüsselwörter. Die Studie zeigt, dass die Erkennungsontologie eine Erfolgsrate von über 80% und eine Präzision von 100% bei technischen Spezifikationen erreicht, ohne dabei Sätze fälschlich als Anforderungen einzustufen [Zwirn 2013]. Das REC-System verspricht eine Automatisierung des aufwendigen und fehleranfälligen Vorgangs der Anforderungsermittlung in technischen Projekten, mit hoher Erkennungsrate und Präzision, insbesondere bei technischen Spezifikationen. Die genutzte Ontologie erleichtert die Klassifizierung und Unterscheidung verschiedener Anforderungstypen, wodurch die Effizienz und Genauigkeit der Anforderungserfassung gesteigert werden können. Allerdings beeinflusst die Textqualität die Genauigkeit des Systems signifikant. Imperative oder unvollständige Sätze sind ungeeignet für das REC, und Dokumente müssen manuell angepasst werden, bevor sie vom System verarbeitet werden können. Die Erprobung des REC-Systems beschränkte sich auf technische Spezifikationen, wodurch unklar bleibt, wie leistungsfähig es bei anderen Textarten ist. Ebenfalls offen bleibt die Frage nach der praktischen Umsetzbarkeit des REC-Systems [Zwirn 2013].

Studie 2 beleuchtet, wie die Analyse von Anforderungsabhängigkeiten in der Planungsphase von Projekten durch den Einsatz von Ontologien verbessert werden kann, wobei traditionelle Methoden oft die komplexen Beziehungen zwischen Projektaspekten außer Acht lassen, was Fehlern und Zeitverlusten Vorschub leistet [Zichler und Helke 2017]. Die Autoren plädieren für Ontologien zur Darstellung von Projektbestandteilen und deren Beziehungen zueinander, um Konflikte und Fehler frühzeitig aufzudecken und somit die Kommunikation zwischen fachübergreifenden Teams zu fördern [Zichler und Helke 2017]. Eine Fallstudie aus der Automobilbranche validiert diesen Ansatz, indem sie demonstriert, wie eine ontologiebasierte Analyse bei der Entwicklung eines Automobilsicherheitssystems Konflikte aufdeckte und Lösungen bot. Die Forschung betont die Wichtigkeit der Abhängigkeitserfassung im Lastenheft und sieht in Ontologien ein effektives Werkzeug zur Prozessoptimierung. Jedoch stellt die zeitaufwendige Durchführung bei komplexen Projekten eine Herausforderung dar, die den Fortschritt bremsen kann. Ein weiteres Problem ist die Präzision der Abhängigkeitsanalyse, da diese auf Annahmen basiert, die nicht immer exakt sind, was fehlerhafte Planungsentscheidungen nach sich ziehen kann. Die Verwaltung von Änderungen erweist sich ebenfalls als schwierig, da Anpassungen an einer Stelle weitreichende Effekte nach sich ziehen können, was Flexibilität und Anpassungsfähigkeit erfordert, um den Projektablauf zu sichern.

Zusammengefasst konzentrieren sich die ersten beiden inländischen Untersuchungen auf das Requirements Engineering, wobei NLP-Pipelines für die Gewichtung von Anforderungen und die Analyse ihrer Abhängigkeiten zum Einsatz kommen. Eine Berücksichtigung von Datenverarbeitung oder Informationsstrukturen bezüglich Kompetenzen findet dabei nicht statt.

Studie 3 präsentiert das Quality User Story Framework (QUS) und ein dazugehöriges Tool, die beide darauf ausgerichtet sind, die Qualität von Anforderungen in der agilen Softwareentwicklung zu steigern [Lucassen et al. 2016]. User Stories, eine verbreitete Methode zur Festlegung von Softwareanforderungen aus Nutzersicht in der agilen Entwicklung, weisen oft Lücken und Mehrdeutigkeiten auf, was ihre präzise Umsetzung erschwert. Das QUS-Framework, bestehend aus sechs Qualitätskriterien - Vollständigkeit, Konsistenz, Korrektheit, Klarheit, Prägnanz und Priorisierung - soll agile Teams unterstützen, die Qualität ihrer User Stories durch systematische Bewertung und Verbesserung zu erhöhen. Zusätzlich wird ein QUS-Tool vorgestellt, das den Bewertungsprozess durch automatisiertes Feedback zur User Story Qualität vereinfacht. Die Effektivität von QUS-Framework und -Tool wurde in einer empirischen Untersuchung mit vier agilen Teams bestätigt. Die Ergebnisse verdeutlichen, dass durch den Einsatz des Frameworks und Tools die Qualität der User Stories verbessert wird, was die Kommunikation und das Verständnis innerhalb der Teams fördert und zu höherwertiger Software führt. Das Tool erleichtert die Verbesserung von User Stories durch automatisierte Bewertungen, erreicht jedoch keine 100%-ige Erkennungsrate, die für die Effektivität von sprachverarbeitenden Analysetools kritisch wäre. Die Bewertung des Tools basiert auf einer Analyse von 1023 User Stories aus 18 Unternehmen, was weitere Forschung mit umfangreicheren Datensätzen notwendig macht, um die Effektivität des Tools vollständig zu validieren [Lucassen et al. 2016].

Studie 4 befasst sich mit der Nutzung der COSMIC-Methode für frühzeitige Aufwandsschätzungen in der agilen Softwareentwicklung [Hussain et al. 2013]. COSMIC, eine Methode zur Messung der Softwaregröße basierend auf ihrer Funktionalität, bietet ein quantitatives Maß für funktionale Softwareanforderungen und unterstützt damit die Aufwandsschätzung. Traditionelle Schätzmethoden, die manuelle Analyse und Formalisierung von Anforderungen erfordern, passen jedoch oft nicht zu den flexiblen Bedingungen agiler Prozesse. Die Studie schlägt daher eine vereinfachte Näherungsmethode vor, die direkt auf textbasierte Anforderungen anwendbar ist, ohne die Notwendigkeit, COSMIC-spezifische Modelle zu erstellen. Diese Methode wurde in vier Fallstudien, darunter zwei Industrieprojekte bei SAP-Labs Canada und zwei universitäre Projekte, getestet. Die analysierten Anforderungs-dokumente variierten zwischen 2.000 und 11.000 Wörtern und beschrieben bis zu 32 verschiedene funktionale Prozesse in unterschiedlichen Problemdomänen. Der vorgeschlagene Ansatz folgt vier Schritten: (1) eine Vorstudie, (2) die Klassifizierung textbasierter Anforderungen in Fuzzy-Größen-klassen, (3) die Berechnung der COSMIC-Funktionspunkte unter Berücksichtigung von Faktoren wie Komplexität und Wiederverwendung und (4) die Überprüfung der Schätzergebnisse mit realen Entwicklungsdaten. Diese Methode ermöglicht agile Teams eine präzisere und effizientere Aufwands-schätzung in den frühen Entwicklungsstadien. Sie eignet sich besonders für die in agilen Prozessen üblichen, nicht formalisierten textuellen Anforderungen. Die Anwendung dieser Methode erfordert jedoch domänenspezifisches Wissen für die korrekte Klassifizierung in Fuzzy-Größen und ist auf-grund ihres subjektiven Charakters anfällig für Schwankungen in der Handhabung durch verschiedene Anwender [Hussain et al. 2013].

In **Studie 5** wird die Entwicklung der NORMAP-Methodik (Non-functional Requirements Management for Agile Processes) beleuchtet, die auf die Verbesserung des Managements von nicht-funktio-nalen Anforderungen (engl. NFRs) in agilen Softwareentwicklungsprojekten abzielt [Farid 2012]. NFRs definieren Qualitätsattribute und Einschränkungen eines Systems, wie beispielsweise Perfor-mance, Sicherheit und Benutzerfreundlichkeit, die in traditionellen, weniger flexiblen Engineering-Ansätzen oft vernachlässigt werden. NORMAP bietet eine agile, schlanke Alternative, indem es einen strukturierten Prozess für die Erfassung, Priorisierung und Überprüfung von NFRs innerhalb agiler Rahmenwerke einführt. Der Ansatz betont die aktive Einbindung von Stakeholdern und die Anwen-dung einfacher, aber wirkungsvoller Techniken für das NFR-Management. Er liefert Leitlinien, wie NFRs nahtlos in agile Planungs- und Umsetzungsaktivitäten integriert werden können, und unter-streicht die Bedeutung einer besseren Koordination zwischen NFR-Management und agilen Praktiken für die Steigerung der Projektqualität. Allerdings ist diese Methodik nicht universell für jedes Projekt oder jede Organisation passend und zielt primär auf reine Softwareentwicklungsprojekte ab [Farid 2012].

Studie 6 führt NORMATIC ein, ein visuelles Unterstützungswerkzeug für die Modellierung nicht-funktionaler Anforderungen (NFRs) in agilen Softwareentwicklungsprozessen, das auf der

NORMAP-Methodik aufbaut [Farid und Mitropoulos 2012]. Das Werkzeug adressiert die Schwierigkeiten bei der Erfassung, Visualisierung und Kommunikation von NFRs und präsentiert einen Modellierungsansatz, der an agilen Methoden angepasst sowie visuell ist. Mit NORMATIC können Nutzer NFRs mittels Zielmodellen und Szenarien visuell darstellen und so eine Plattform für die kollaborative Verfeinerung dieser Anforderungen bieten. Durch seine grafische Benutzeroberfläche erleichtert das Tool die gemeinsame Arbeit an NFRs in einer agilen und benutzerfreundlichen Weise. Es ermöglicht außerdem die Verknüpfung von NFRs mit User Stories, Elementen des Product Backlogs und Iterationen, was die Nachvollziehbarkeit und die Abstimmung mit dem Entwicklungsprozess verbessert. Bislang wurde NORMATIC jedoch noch nicht in praktischen agilen Entwicklungsprojekten getestet, weshalb die Effektivität des Tools in realen agilen bzw. Scrum-Umgebungen ungewiss bleibt. Es ist zudem zu beachten, dass sowohl das Tool als auch die zugrundeliegende NORMAP-Methodik speziell für agile Softwareprojekte konzipiert sind und sich nicht auf andere Arten von Projekten übertragen lassen [Farid und Mitropoulos 2012].

Studie 7 thematisiert die Herausforderungen beim manuellen Design und der Erstellung von Testfällen in der agilen Softwareentwicklung, ein Prozess, der sich oft als zeitraubend und fehleranfällig erweist und zu lückenhafter Anforderungsabdeckung führen kann [Rane 2017]. Als Lösungsansatz wird die Anwendung von Natural Language Processing (NLP) zur Automatisierung der Generierung funktionaler Testfälle aus natürlichsprachlich formulierten Testbeschreibungen vorgeschlagen. Dieser Ansatz nutzt NLP-Technologien, um User Stories in Aktivitätsdiagramme umzuwandeln, die anschließend zur Erstellung von Testfällen dienen. Zur Steigerung der Präzision im Testfallgenerierungsprozess werden zwei neue Parameter eingeführt: a) das Testszenario und b) ein spezifisches Wörterbuch. Erste Tests des Tools an einem Beispielprojekt verliefen vielversprechend, indem präzise und umfassende Testfälle produziert wurden. Dieser Ansatz bietet gegenüber traditionellen Methoden der Testfallentwicklung zahlreiche Vorteile, wie gesteigerte Effizienz, Genauigkeit, Vollständigkeit und Wiederverwendbarkeit der Testfälle. Zukünftige Forschungsarbeiten sollten sich darauf konzentrieren, die Genauigkeit von NLP-Techniken bei der Umsetzung von User Stories in Aktivitätsdiagramme zu verbessern und die Anwendbarkeit des Verfahrens auf diverse Softwareprojekttypen zu erweitern [Rane 2017].

Studie 8 untersucht den Einsatz von Natural Language Processing (NLP) und Machine Learning (ML) zur Verbesserung der SoftwareentwicklungsAufwandsschätzung, einem entscheidenden Faktor in der Projektplanung und Ressourcenallokation [Ionescu et al. 2017]. Traditionelle Schätzmethoden stützen sich oft auf Experteneinschätzungen und historische Daten, die in puncto Genauigkeit und Anpassungsfähigkeit limitiert sein können. Der Gebrauch von NLP und ML verspricht, diese Einschränkungen durch eine präzisere und automatisierte Schätzung zu überwinden. NLP wird eingesetzt, um aus textbasierten Software-Dokumentationen relevante Informationen zu extrahieren, während ML-Algorithmen und -Modelle zur Entwicklung von Schätzmodellen genutzt werden, die wesentlichen Kostenfaktoren identifizieren können. Erste Ergebnisse belegen die Wirksamkeit dieses

Ansatzes, der eine höhere Genauigkeit der Aufwandsschätzung gegenüber herkömmlichen Verfahren zeigt. Die Studie empfiehlt allerdings, zusätzliche Daten aus verschiedenen Unternehmenskontexten zu sammeln, um die Effizienz unterschiedlicher Modelle unter diversen Bedingungen besser evaluieren zu können [Ionescu et al. 2017].

Studie 9 erforscht den Einsatz von Natural Language Processing (NLP) zur Definition und Klassifizierung der Komplexität von Softwareanforderungen, ein Schlüsselement für den Erfolg von Softwareentwicklungsprojekten [Sundararajan et al. 2018]. Die Studie adressiert insbesondere die Probleme von Subjektivität und Inkonsistenz, die bei der Bewertung von Anforderungskomplexitäten auftreten. Ein Framework, das NLP-Techniken nutzt, wird vorgestellt, um textbasierte Anforderungen durch Verfahren wie Tokenisierung, Part-of-Speech-Tagging und syntaktische Analyse vorzubereiten und zu untersuchen. Dabei werden spezifische Merkmale aus den Anforderungen extrahiert, die für die Definition und Quantifizierung von Komplexität herangezogen werden. Die Studie schlägt eine Einteilung der Anforderungskomplexität in verschiedene Stufen oder Kategorien vor. Eine empirische Untersuchung mit realen Softwareanforderungen validiert das Framework. Die Ergebnisse belegen die Wirksamkeit des NLP-gestützten Vorgehens bei der Ermittlung und Klassifizierung von Komplexitäten und liefern Einsichten in die Komplexitätsniveaus unterschiedlicher Anforderungen. Dennoch wird empfohlen, die Methodik auf weitere Aspekte der Softwareentwicklung auszudehnen und sie mit anderen KI-Technologien zu kombinieren, um die Präzision bei der Bestimmung und Kategorisierung von Anforderungskomplexitäten zu steigern [Sundararajan et al. 2018].

Studie 10 erforscht den Einsatz von Natural Language Processing (NLP)-Techniken für die Auswahl von Projekten in der agilen Softwareentwicklung, ein entscheidender Entscheidungsfindungsprozess, der die Projektrealisierbarkeit, Teamkompetenzen und die Übereinstimmung mit Unternehmenszielen beurteilt [Sharma et al. 2019]. Vorgeschlagen wird ein vereinfachtes Modell, das NLP und agile Teamfähigkeiten integriert, um die Entscheidungsfindung zu optimieren. Durch die Analyse von Textdaten zu Projektanforderungen und Teamkompetenzen mittels NLP, zielt das Modell darauf ab, relevante Informationen zu extrahieren und zu bewerten. Es betont die Bedeutung der Einbindung agiler Teamkompetenzen in den Auswahlprozess und schlägt vor, mittels NLP den Präsenz dieser Fähigkeiten in Teamprofilen zu analysieren und zu quantifizieren. Die Validierung des Modells offenbart seine Effektivität, insbesondere bei der Beurteilung von Projektanforderungen und Teamfähigkeiten. Jedoch wurden Schwierigkeiten bei einigen Teams festgestellt, die Anforderungen nicht adäquat verfolgten oder bei der Anforderungserhebung keine relevanten Fragen stellten. Zudem führten ungeeignete agile Teams durch erhöhte Fehlerquoten und langwierige tägliche Scrum-Meetings zu gesteigerten Projektrisiken. Diese Herausforderungen beeinflussen die Validierungsergebnisse und unterstreichen die Notwendigkeit, diese Aspekte bei der Anwendung des Modells in der Softwareprojektentwicklung zu berücksichtigen [Sharma et al. 2019].

Studie 11 untersucht den Einsatz von NLP-Techniken zur Verbesserung des Traceability-Managements in der Softwareentwicklung durch den SAT-Analyzer [Arunthavanathan et al. 2016]. Traceability-Management, das die Verbindung zwischen verschiedenen Software-Artefakten wie Anforderungen, Design-Dokumenten und Quellcode herstellt, ist essenziell für Konsistenz und effizientes Änderungsmanagement. Der SAT-Analyzer automatisiert die Analyse von Textinformationen aus Softwareartefakten, um ein effizientes Traceability-Management zu ermöglichen und Herausforderungen wie hohen Pflegeaufwand und Fehleranfälligkeit manueller Prozesse zu überwinden. Eine empirische Studie mit realen Software-Artefakten validiert den Ansatz und zeigt die Wirksamkeit des NLP-basierten Systems bei der Erstellung von Traceability-Links und der Reduktion manueller Arbeit. Es werden Verbesserungen vorgeschlagen, unter anderem die Integration eines Verhaltensbaums zur Behebung von Mehrdeutigkeiten, Unvollständigkeiten und Inkonsistenzen in Anforderungsdokumenten und die Erweiterung des SAT-Analyzers zur Verarbeitung von Dokumenten, die Text und Grafiken enthalten. Zusätzlich könnte das Training des Tools mit vielfältigeren Datensätzen die NLP-Leistung verbessern und eine breitere Palette von Schreibstilen abdecken [Arunthavanathan et al. 2016].

Studie 12 präsentiert das Konzept eines intelligenten Projektmanagements, das durch den Einsatz fortschrittlicher Technologien wie NLP die Projektplanung, -überwachung und -kommunikation verbessern soll [Chen et al. 2021]. Eine interaktive Plattform, die NLP-Funktionen integriert, wird vorgestellt, um effizientes und effektives Projektmanagement zu ermöglichen. Die Plattform nutzt NLP, um natürlichsprachige Eingaben der Projektteilnehmer zu verarbeiten und zu analysieren, wobei Techniken wie Sentiment-Analyse und Themenmodellierung angewandt werden, um wertvolle Einsichten aus Textdaten zu extrahieren. Die Integration von NLP in verschiedene Bereiche des Projektmanagements wird erörtert, einschließlich der Automatisierung von Aufgaben, der Erstellung von Berichten und der Bereitstellung von Echtzeit-Feedback basierend auf Textanalysen. Die Studie diskutiert allgemein die Vorteile und potenziellen Anwendungen einer solchen intelligenten Projektmanagementplattform, insbesondere die Verbesserung von Kommunikation, Zusammenarbeit und Entscheidungsfindung unter Projektbeteiligten. Die Plattform wird als Mittel zur Steigerung der Projekteffizienz, zur Fehlerreduktion und zur Unterstützung datengetriebener Entscheidungen vorgestellt. Einschränkend wird jedoch erwähnt, dass sich die Studie auf die chinesische Sprache konzentriert und die Anwendung und Wirksamkeit in anderen Sprachen zukünftig geprüft werden sollte, um die Effizienz des Mechanismus zu bestätigen [Chen et al. 2021].

In den diskutierten Studien wird der Einsatz von NLP-Technologien hauptsächlich für die Verarbeitung von Daten im Bereich des Anforderungsmanagements und der Aufwandsschätzung, basierend auf der Analyse von Anforderungsdokumenten, erforscht. Abgesehen von den Studien 10 und 12, liegt kein Fokus auf der Analyse von Kompetenzen und der darauffolgenden Projektplanung in Industrieprojekte. Insbesondere Studie 10 adressiert zwar die Zusammenstellung agiler Teams und berücksichtigt deren Fähigkeiten für spezifische Projekte, konzentriert sich jedoch lediglich auf die Fertigkeiten der Mitarbeiter ohne eine tiefergehende Betrachtung ihrer tatsächlichen Kompetenzen. Es

fehlt an einer strukturierten Methode zur Analyse der Mitarbeiterkompetenzen, und Erfahrungen aus früheren Projekten des Unternehmens werden nicht herangezogen, was für die Planung zukünftiger Projekte von Vorteil sein könnte. Studie 12 wiederum beleuchtet den Informationsaustausch zwischen Projektbeteiligten, ohne die Projektplanung im Detail zu betrachten, und beschränkt sich zudem auf Bauprojekte sowie die chinesische Sprachverarbeitung.

Die Vor- und Nachteile der genannten Studien dienen als Grundlage für die Definition der Anforderungen an das in dieser Arbeit zu entwickelndem Konzept. Die Anforderungen werden in Kapitel 4.2 diskutiert.

3.4 Technische Ansätze zum Projekt- und Ressourcenmanagement

Neben den wissenschaftlichen Ansätzen zur Handhabung der kompetenzbasierten Ressourcenplanung im Projektmanagement hat auch die Industrie verschiedene Softwarelösungen entwickelt. Um Anforderungen an das zu entwickelndes Konzept festlegen zu können, ist es erforderlich, sich zunächst mit den in der Industrie angewandten Ansätzen vertraut zu machen und deren Vor- und Nachteile zu analysieren. Diese Analyse dient dazu, im weiteren Verlauf die entwickelten Anforderungen zu validieren und den KI-Workflow zu erproben.

Eine umfangreiche Recherche wurde durchgeführt, um festzustellen, ob die derzeit verwendeten Softwaresysteme die in der Dissertation fokussierte Problemstellung abdecken. Insgesamt wurden **15 Softwaresysteme** identifiziert und untersucht. Um auf den Erkenntnissen der aktuellen Softwarelösungen aufzubauen, ist es notwendig, diese zu analysieren und relevante Ergebnisse zu extrahieren. Es ist jedoch anzumerken, dass nur der Stand der Demo-Versionen, Informationsmaterialien und Online-Dokumentationen bewertet werden kann. Die spezifische Anpassungsfähigkeit der Softwareprodukte und die Modularität der Lösungen der Hersteller konnten daher nicht vollständig erfasst werden.

Microsoft Project ist eine umfassende Projektmanagement-Software, die Unternehmen bei der Planung, Durchführung und Überwachung von Projekten unterstützt. Zu den Hauptfunktionen gehören Gantt-Diagramme, Ressourcenmanagement, Zeit- und Kostenerfassung, Risiko- und Änderungsmanagement sowie Berichterstellung. Die Gantt-Diagramme ermöglichen eine detaillierte Visualisierung des Projektplans, während das Ressourcenmanagement die optimale Nutzung von Ressourcen sicherstellt. Zeit- und Kostenerfassung bieten eine präzise Budgetkontrolle, und das Risiko- und Änderungsmanagement hilft, Probleme frühzeitig zu erkennen und zu beheben. Umfangreiche Berichterstellungsfunktionen ermöglichen detaillierte Analysen des Projektstatus. Ein großer Vorteil von Microsoft Project ist die Integration in das Microsoft Office-Ökosystem, was die Zusammenarbeit erleichtert. Die Software ist skalierbar und kann sowohl von kleinen als auch großen Teams genutzt werden. Die umfassenden Funktionen erfordern jedoch eine gewisse Einarbeitungszeit, und die Lizenzkosten sind

höher als bei einfacheren Tools wie Excel [Holert 2020]. **Microsoft Excel** hingegen ist eine kostengünstige und bekannte Lösung für die Projektplanung, besonders geeignet für kleinere Teams und weniger komplexe Projekte. Excel ermöglicht die Erstellung eigener Dateien oder die Nutzung vorgefertigter Vorlagen zur Ressourcenplanung. Die Vorlagen umfassen drei wichtige Bereiche: Basisdaten, Übersichtspläne und Auswertungen. Excel ermöglicht vielfältige grafische Auswertungen, stößt jedoch an seine Grenzen, wenn Teams wachsen oder mehrere Teams geplant werden sollen. Daten in verschiedenen Dateien können zu Versionskonflikten führen, und die gemeinsame Arbeit in einer einzigen Datei wird unkomfortabel. Zudem bietet Excel keine integrierten KI-Komponenten. Diese Einschränkungen verdeutlichen die Notwendigkeit spezialisierter Softwarelösungen, die über die Basisfunktionalitäten von Excel hinausgehen und eine kompetenzbasierte Ressourcenplanung effizienter unterstützen können [Delos Santos 2019].

SAP Project and Portfolio Management (PPM) ist eine integrierte Lösung von Firma SAP, die Unternehmen dabei unterstützt, ihre Projekte und Portfolios effizient zu planen, zu verwalten und zu überwachen. Die Funktionen von SAP PPM sind vielfältig. Die Projektplanung und -verfolgung umfasst die Erstellung von Work Breakdown Structures (WBS), Meilensteinen und Zeitplänen sowie die Visualisierung von Projektplänen und Fortschritten mittels Gantt-Diagrammen. Auch die Aufgabenverwaltung, bei der Aufgaben an Teammitglieder zugewiesen und deren Fortschritt nachverfolgt werden können, ist ein zentraler Bestandteil. Im Ressourcenmanagement können Ressourcen basierend auf Verfügbarkeit und Qualifikationen geplant und zugewiesen werden. Die Kapazitätsplanung ermöglicht die Überwachung der Ressourcenauslastung und Identifizierung von Engpässen, während das Skill-Management die Verwaltung der Fähigkeiten und Kompetenzen der Mitarbeiter erleichtert. Da die Software Modular aufgebaut ist, kann diese auch Finanzen und Risiko- und Chancenmanagement handhaben, welche aber nicht im Fokus dieser Dissertation steht. Berichtswesen und Analysen sind ebenfalls integrale Bestandteile von SAP PPM. Echtzeit-Dashboards ermöglichen die Überwachung des Projektstatus, und sowohl Standard- als auch benutzerdefinierte Berichte bieten detaillierte Analysen von Projektdaten. Portfolioberichte bieten eine übersichtliche Darstellung des gesamten Projektportfolios und dessen Leistungskennzahlen [SAP 2024].

SAP PPM bietet zahlreiche Vorteile. Die Lösung ist nahtlos in andere SAP-Module wie SAP ERP und SAP HANA integriert, was eine einheitliche Datenbasis und konsistente Prozesse ermöglicht. Sie bietet Echtzeit-Einblicke in den Projektstatus, die Ressourcenauslastung und die finanziellen Aspekte, was fundierte Entscheidungen erleichtert. Durch detaillierte Kapazitäts- und Skill-Management-Funktionen wird eine optimale Nutzung von Ressourcen ermöglicht. Frühzeitige Identifikation und Minderung von Projektrisiken erhöhen die Wahrscheinlichkeit des Projekterfolgs. Zudem ist die Software skalierbar und kann an die Bedürfnisse von kleinen, mittleren und großen Unternehmen angepasst werden. Den Vorteilen stehen jedoch auch einige Nachteile gegenüber. Die Implementierung und der Betrieb von SAP PPM können teuer sein, insbesondere für kleinere Unternehmen. Die Vielseitigkeit und Tiefe der Funktionen machen die Software komplex und schwer erlernbar, was umfangreiche

Schulungen erforderlich macht. Zudem kann die Implementierung von SAP PPM zeitaufwendig und ressourcenintensiv sein, insbesondere wenn maßgeschneiderte Anpassungen erforderlich sind. Ein hoher Grad an technischer Expertise ist erforderlich, um die Lösung effektiv zu nutzen und zu warten. Ein weiterer Nachteil ist, dass SAP PPM keine integrierten KI-Komponenten nutzt, was den Automatisierungsgrad und die intelligente Entscheidungsunterstützung im Vergleich zu anderen modernen Projektmanagement-Tools einschränkt. Zusammenfassend lässt sich sagen, dass SAP PPM eine leistungsstarke Lösung für das Projekt- und Portfoliomanagement ist, die besonders für große Unternehmen geeignet ist, die eine integrierte und umfassende Verwaltung ihrer Projekte und Ressourcen benötigen. Während die Software erhebliche Vorteile in Bezug auf Transparenz, Kontrolle und Effizienz bietet, müssen die potenziellen hohen Kosten, die Komplexität und das Fehlen von KI-Komponenten berücksichtigt werden [SAP 2024].

Saviom ist eine Ressourcenmanagement- und Planungssoftware, die Unternehmen dabei hilft, ihre Ressourcen effizient zu nutzen und zu optimieren. Sie bietet eine flexible und benutzerfreundliche Plattform zur Verwaltung von Personal, Projekten und Aufgaben und ist stark darauf ausgelegt, die Ressourcenauslastung zu maximieren und gleichzeitig die Projektkosten zu minimieren. Im Ressourcenmanagement ermöglicht die Software die detaillierte Planung und Zuweisung von Ressourcen basierend auf Verfügbarkeit, Fähigkeiten und Projektanforderungen. Die Kapazitätsplanung überwacht die Auslastung der Ressourcen und identifiziert potenzielle Engpässe, um rechtzeitig Gegenmaßnahmen ergreifen zu können. Das Skill-Management hilft bei der Verwaltung der Fähigkeiten und Kompetenzen der Mitarbeiter, sodass die richtigen Ressourcen für die richtigen Aufgaben zugewiesen werden können. Saviom bietet auch Projektmanagementfunktionen. Die Software unterstützt die Projektplanung und -verfolgung, indem sie Gantt-Diagramme zur Visualisierung von Projektplänen und Fortschritten bereitstellt. Die Aufgabenverwaltung ermöglicht es, Aufgaben an Teammitglieder zuzuweisen und deren Fortschritt nachzuverfolgen. Berichte und Analysen sind ein weiterer wichtiger Bestandteil von Saviom. Die Software generiert detaillierte Berichte und Dashboards, die Echtzeit-Einblicke in den Status von Projekten und Ressourcen bieten. Die Analysefunktionen ermöglichen es, Trends und Muster in der Ressourcennutzung zu erkennen und zukünftige Anforderungen besser vorherzusagen [Saviom 2024].

Die Software ist benutzerfreundlich und flexibel, sodass sie an die spezifischen Bedürfnisse und Prozesse eines Unternehmens angepasst werden kann. Die umfassenden Berichts- und Analysefunktionen bieten wertvolle Einblicke und unterstützen fundierte Entscheidungsprozesse. Zudem ermöglicht Saviom eine bessere Kontrolle über Zeit und Kosten, was zur Einhaltung von Budgets und Terminen beiträgt. Es gibt jedoch auch einige Nachteile. Die Implementierung von Saviom kann zeitaufwendig und kostspielig sein, insbesondere wenn umfangreiche Anpassungen erforderlich sind. Zudem erfordert die Nutzung der Software eine gewisse Einarbeitungszeit und Schulung, um die Funktionen effektiv nutzen zu können. Ein weiterer Nachteil ist, dass Saviom im Vergleich zu anderen modernen

Ressourcenmanagement-Tools keine integrierten KI-Komponenten bietet, was den Automatisierungsgrad und die intelligente Entscheidungsunterstützung einschränken kann [Saviom 2024].

Teamdeck ist ein Tool für Ressourcenmanagement, Zeiterfassung und Personalplanung, das speziell für kreative Agenturen und Softwareentwicklungsteams entwickelt wurde. Es bietet eine benutzerfreundliche Plattform, die es Unternehmen ermöglicht, die Verfügbarkeit ihrer Mitarbeiter zu verwalten, Arbeitszeiten zu erfassen und die Projektauslastung zu optimieren. Der Ressourcenmanagementmodul ermöglicht es den Nutzern, die Verfügbarkeit und Auslastung der Teammitglieder in Echtzeit zu verfolgen. Dadurch können Projektmanager sicherstellen, dass Ressourcen optimal eingesetzt werden und keine Überlastung oder Unterbeschäftigung stattfindet. Die Zeiterfassungsfunktion erlaubt es den Mitarbeitern, ihre Arbeitszeiten präzise zu dokumentieren, was eine genaue Abrechnung und eine bessere Nachverfolgung der geleisteten Arbeit ermöglicht. Darüber hinaus unterstützt Teamdeck das Urlaubs- und Abwesenheitsmanagement, indem es eine zentrale Plattform bietet, auf der Mitarbeiter ihre Abwesenheiten eintragen und verwalten können. Dies hilft, Personalengpässe zu vermeiden und die Planung zu verbessern. Die Projektzeiterfassung ist ein weiterer zentraler Bestandteil von Teamdeck. Sie ermöglicht es den Projektmanagern, die aufgewendete Zeit für verschiedene Projekte und Aufgaben zu verfolgen und so die Projektkosten besser zu kontrollieren. Diese Funktion bietet auch wertvolle Einblicke in die Effizienz und Produktivität der Teams und hilft bei der Identifizierung von Optimierungspotenzialen. Zudem bietet Teamdeck ebenso umfangreiche Berichte und Analysen, die es den Nutzern ermöglichen, fundierte Entscheidungen zu treffen [Teamdeck 2024].

Teamdeck bietet zahlreiche Vorteile. Die Software ist benutzerfreundlich und leicht zu erlernen. Diese erleichtern die Implementierung und Nutzung der Software. Sie unterstützt eine effiziente Verwaltung der Ressourcen und Arbeitszeiten, was zu einer besseren Auslastung und Produktivität der Teams führt. Teamdeck kann jedoch für kleinere Unternehmen oder Teams, die keine umfassenden Ressourcenmanagement-Funktionen benötigen, überdimensioniert sein. Ein Nachteil ist, dass Teamdeck, im Vergleich zu anderen fortschrittlichen Ressourcenmanagement-Tools, keine integrierten KI-Komponenten bietet. Dies könnte den Automatisierungsgrad und die intelligente Entscheidungsunterstützung einschränken und somit den Nutzen für Unternehmen, die stark auf Datenanalyse und maschinelles Lernen setzen, reduzieren [Teamdeck 2024].

Float ist eine Ressourcenmanagement-Software, die darauf abzielt, die Planung und Verwaltung von Projekten und Teams effizienter zu gestalten. Sie bietet eine benutzerfreundliche Plattform, die es Unternehmen ermöglicht, die Verfügbarkeit ihrer Ressourcen zu überwachen, Projekte zu planen und die Arbeitsauslastung ihrer Mitarbeiter zu optimieren. Projektmanager können Aufgaben und Projekte zuweisen, die Arbeitslasten der Mitarbeiter überwachen und sicherstellen, dass keine Überlastung oder Unterbeschäftigung stattfindet. Float bietet eine Drag-and-Drop-Oberfläche, die die Planung und Anpassung von Zeitplänen einfach und intuitiv macht. Die Zeiterfassungsfunktion von Float erlaubt es

den Nutzern, die aufgewendete Zeit für verschiedene Aufgaben und Projekte genau zu dokumentieren. Dies hilft nicht nur bei der Abrechnung und Nachverfolgung, sondern bietet auch wertvolle Einblicke in die Effizienz und Produktivität der Teams. Die Berichte und Analysen, die Float generiert, bieten detaillierte Einblicke in die Ressourcenauslastung, Projektfortschritte und Arbeitszeiten [Float 2024].

Float bietet Vor- und Nachteile. Die Software ist benutzerfreundlich und leicht zu erlernen. Die Echtzeit-Überwachung der Ressourcenverfügbarkeit und die einfache Anpassung der Zeitpläne unterstützen eine effiziente Ressourcenverwaltung. Die Zeiterfassungsfunktion und die umfangreichen Berichts- und Analysefunktionen bieten wertvolle Einblicke und unterstützen fundierte Entscheidungsprozesse. Float kann jedoch für kleinere Unternehmen oder Teams, die keine umfassenden Ressourcenmanagement-Funktionen benötigen, zu kompliziert sein. Ein weiterer Nachteil ist, dass Float auch, wie andere vorhin beschriebenen Softwares, keine integrierten KI-Komponenten bietet [Float 2024].

Projektron BCS (Business Coordination Software) ist eine webbasierte Projektmanagement-Software. Sie bietet eine, wie alle anderen Softwares in diesem Bereich, Vielzahl von Funktionen zur effizienten Planung, Steuerung und Überwachung von Projekten. Zu den zentralen Funktionen gehören die Projektplanung und -steuerung, bei der Projektpläne, Zeitpläne, Aufgabenlisten und Meilensteine erstellt und verwaltet werden können. Das Ressourcenmanagement ermöglicht die gezielte Verwaltung und Zuweisung von Ressourcen basierend auf deren Verfügbarkeit und Qualifikationen. Eine weitere wichtige Funktion ist die Zeiterfassung, die eine präzise Erfassung der Arbeitszeiten der Mitarbeiter erlaubt, welche direkt in die Projektabrechnung einfließen können. Darüber hinaus bietet Projektron BCS umfassende Möglichkeiten im Kosten- und Budgetmanagement, einschließlich der Überwachung von Projektbudgets, der tatsächlichen Kosten und der Erstellung detaillierter Finanzberichte. Kollaborationswerkzeuge wie Dokumentenmanagement, Diskussionsforen und Aufgabenverfolgung unterstützen die Zusammenarbeit innerhalb des Teams. Was den Einsatz von KI-Komponenten betrifft, so integriert Projektron BCS derzeit keine spezifische künstliche Intelligenz (KI)-Funktionen. Die Software setzt stattdessen auf traditionelle Methoden des Projekt- und Ressourcenmanagements sowie auf umfangreiche Berichts- und Analysetools, die auf den erfassten Daten basieren [Projektron GmbH 2024].

Im Gegensatz zudem bisher genannte Softwares, gibt es jedoch eine Reihe fortschrittlicher Softwarelösungen, die moderne KI-Technologien nutzen, um das Projekt- und Ressourcenmanagement zu optimieren. Diese Tools bieten nicht nur grundlegende Funktionen, sondern auch intelligente Automatisierungen und prädiktive Analysen, die den Arbeitsablauf erheblich verbessern können. **Asana**, **Trello**, **Monday.com**, **Meisterplan**, **Awork**, **Can Do**, **Wrike** und **Smartsheet** sind fünf fortschrittliche Tools für das Ressourcen- und Projektmanagement, die jeweils unterschiedliche Funktionen, Vorteile und Nachteile bieten.

Asana ist bekannt für seine umfassenden Projektmanagement-Funktionen, die es Teams ermöglichen, Aufgaben zu organisieren, Projekte zu verfolgen und die Zusammenarbeit zu verbessern. Eine der herausragenden Funktionen von Asana ist die Möglichkeit, Projekte in verschiedene Ansichten wie Listen, Boards und Kalender zu unterteilen, was eine flexible Verwaltung von Aufgaben ermöglicht. Asana integriert auch KI-Komponenten, die dabei helfen, Prioritäten zu setzen und Arbeitsabläufe zu optimieren. Die KI-gestützte Automatisierung kann repetitive Aufgaben eliminieren und so die Effizienz steigern. Ein Vorteil von Asana ist die intuitive Benutzeroberfläche, die sowohl für kleine Teams als auch für große Unternehmen geeignet ist. Für neue Benutzer kann Asana aufgrund seiner Vielzahl von Funktionen und Optionen überwältigend sein. Die Lernkurve kann steil sein, insbesondere für diejenigen, die keine Erfahrung mit Projektmanagement-Software haben. Die Navigation und Nutzung aller verfügbaren Funktionen kann Zeit und Schulung erfordern. Außerdem generiert Asana eine große Anzahl von Benachrichtigungen, was zu einer Überlastung und Ablenkung der Benutzer führen kann. Es kann schwierig sein, die wirklich wichtigen Updates von weniger relevanten Mitteilungen zu unterscheiden, was die Produktivität beeinträchtigen kann. Benutzer müssen Zeit investieren, um ihre Benachrichtigungseinstellungen optimal zu konfigurieren. Die kostenlose Version bietet nur eingeschränkte Funktionen. Viele fortschrittliche Features wie Timeline-Ansichten, erweiterte Berichte und Integrationen sind nur in den kostenpflichtigen Versionen verfügbar. Dies kann die Nutzung für kleinere Teams oder Unternehmen mit begrenztem Budget einschränken [Asana Academy 2024].

Trello zeichnet sich durch seine benutzerfreundliche und visuell ansprechende Kanban-Board-Struktur aus, die ideal für die Organisation von Aufgaben und Projekten ist. Es ermöglicht Benutzern, Karten für Aufgaben zu erstellen, die dann auf Boards verschoben werden können, um den Fortschritt zu visualisieren. Trello bietet KI-Komponenten wie Butler, ein Automatisierungstool, das Arbeitsabläufe optimiert, indem es Regeln und Befehle basierend auf Benutzeraktionen ausführt. Vorteile von Trello sind seine Einfachheit und Flexibilität, die es zu einer ausgezeichneten Wahl für Teams macht, die ein leicht verständliches Tool benötigen. Allerdings können die begrenzten Funktionen für komplexe Projektmanagementanforderungen unzureichend sein, und das Fehlen tieferer Berichtsfunktionen kann als Nachteil angesehen werden [Trello Guide 2024].

Monday.com bietet eine umfassende Plattform für das Arbeitsmanagement, die sich durch hohe Anpassungsfähigkeit und zahlreiche Integrationen auszeichnet. Es ermöglicht die Erstellung von benutzerdefinierten Workflows und Dashboards, die den spezifischen Anforderungen eines Teams gerecht werden. Die KI-Komponenten von Monday.com beinhalten Automatisierungen, die Routineaufgaben minimieren und den Arbeitsfluss optimieren. Dazu gehören Benachrichtigungen, Statusaktualisierungen und Aufgabenautomatisierung. Ein großer Vorteil von Monday.com ist seine Vielseitigkeit und Anpassungsfähigkeit, die es Teams ermöglicht, die Plattform nach ihren Bedürfnissen zu gestalten. Ein Nachteil könnte jedoch sein, dass die Fülle an Anpassungsoptionen und Integrationen für einige Benutzer überwältigend sein kann, insbesondere für diejenigen, die ein einfaches, sofort einsatzbereites Tool suchen [Monday Apps Framework 2024].

Meisterplan ist eine spezialisierte Lösung für Projektportfolio- und Ressourcenmanagement. Die Software bietet eine intuitive Benutzeroberfläche und leistungsstarke Planungsfunktionen, die es den Nutzern ermöglichen, Projekte zu priorisieren, Ressourcen zuzuweisen und Engpässe zu identifizieren. Eine der herausragenden Funktionen von Meisterplan ist die Szenarioplanung, die es ermöglicht, verschiedene Projekt- und Ressourcenkonfigurationen zu simulieren und deren Auswirkungen zu analysieren. Im Hinblick auf KI-Komponenten nutzt Meisterplan Algorithmen, um optimale Ressourcenzuweisungen vorzuschlagen und potenzielle Engpässe frühzeitig zu erkennen. Dies hilft Projektmanagern, fundierte Entscheidungen zu treffen und die Projektplanung effizienter zu gestalten. Ein großer Vorteil von Meisterplan ist seine Benutzerfreundlichkeit und die Fähigkeit, komplexe Szenarien einfach darzustellen. Ein Nachteil könnte jedoch sein, dass die Software weniger für sehr kleine Teams geeignet ist, die keine umfangreichen Planungsfunktionen benötigen [Meisterplan Help Center 2023].

Awork ist ein kollaboratives Projektmanagement-Tool, das besonders für kreative Teams und Agenturen geeignet ist. Es bietet Funktionen wie Aufgabenverwaltung, Zeiterfassung und Teamkommunikation in einer benutzerfreundlichen Oberfläche. Awork integriert KI-Komponenten, um wiederkehrende Aufgaben zu automatisieren und den Arbeitsablauf zu optimieren. Beispielsweise können Aufgaben automatisch basierend auf bestimmten Kriterien zugewiesen und priorisiert werden. Die KI-gestützten Automatisierungen in awork helfen dabei, manuelle Prozesse zu reduzieren und die Effizienz zu steigern. Ein weiterer Vorteil ist die nahtlose Integration mit anderen Tools wie Slack und Trello, was die Zusammenarbeit im Team erleichtert. Ein Nachteil könnte sein, dass awork weniger fortschrittliche Analysefunktionen bietet als andere Projektmanagement-Tools, was für Teams, die umfangreiche Datenanalysen benötigen, ein Nachteil sein könnte [Awork Guides 2024].

Can Do ist eine Projektmanagement-Software, die sich durch ihre fortschrittlichen KI-Funktionen auszeichnet. Die Software bietet umfassende Planungs- und Steuerungsfunktionen, die durch KI-Komponenten unterstützt werden. Can Do verwendet künstliche Intelligenz, um prädiktive Analysen zu erstellen, die helfen, Risiken zu identifizieren und Ressourcen optimal zu nutzen. Die KI-Funktionen ermöglichen es, Datenmuster zu erkennen und genaue Prognosen über den Projektfortschritt und mögliche Hindernisse zu erstellen. Dies bietet einen erheblichen Vorteil für Projektmanager, die auf präzise und datengestützte Entscheidungen angewiesen sind. Can Do berücksichtigt jedoch ausschließlich die Daten von Projekten, die innerhalb dieser Software erstellt wurden. Projekte, die außerhalb der Can Do-Umgebung erstellt wurden, können nicht einbezogen werden. Ein weiterer Vorteil ist die Fähigkeit von Can Do, komplexe Projekte und große Teams effizient zu verwalten. Ein Nachteil könnte jedoch die Komplexität der Software sein, die eine längere Einarbeitungszeit erfordert und für kleinere Teams möglicherweise überdimensioniert ist [Can Do 2024].

Wrike ist eine leistungsstarke Projektmanagement-Software, die speziell für die Bedürfnisse größerer Teams und komplexer Projekte entwickelt wurde. Es bietet erweiterte Funktionen wie Gantt-Diagramme, Zeiterfassung, Projektvorlagen und umfangreiche Berichtsfunktionen. Die KI-

Komponenten in Wrike beinhalten die Wrike-Business-Intelligence, die Datenanalysen und prädiktive Einblicke bietet, um fundierte Entscheidungen zu treffen. Diese Funktion analysiert historische Daten und aktuelle Projektinformationen, um Trends und Muster zu erkennen. Basierend auf diesen Analysen können Benutzer fundierte Entscheidungen treffen, da die BI-Komponente prädiktive Analysen bereitstellt, die zukünftige Projektentwicklungen und potenzielle Herausforderungen prognostizieren. Wrike verwendet jedoch nur die Daten von Projekten, die mit dieser Software erstellt wurden. Es ist nicht möglich, Projekte zu berücksichtigen, die bereits außerhalb der Wrike-Umgebung erstellt wurden sind. Vorteile von Wrike sind seine robusten Funktionen und die Möglichkeit, detaillierte Projektpläne zu erstellen und zu verwalten. Allerdings können die Komplexität und der Umfang der Funktionen eine Herausforderung für kleinere Teams oder weniger erfahrene Benutzer darstellen, die möglicherweise nicht alle Funktionen benötigen oder nutzen können [Wrike Inc. 2024].

Smartsheet kombiniert die Einfachheit einer Tabellenkalkulation mit den leistungsstarken Funktionen einer Projektmanagement-Software. Es ermöglicht die Erstellung und Verwaltung von Projekten in einem tabellenähnlichen Format, das vielen Benutzern vertraut ist. Smartsheet bietet KI-Funktionen wie Automatisierungen, die Aufgaben und Workflows basierend auf bestimmten Kriterien auslösen. Diese Automatisierungen helfen dabei, manuelle Prozesse zu reduzieren und die Effizienz zu steigern. Ein Vorteil von Smartsheet ist die Benutzerfreundlichkeit und Flexibilität, die es zu einer guten Wahl für Teams macht, die eine einfache, aber leistungsfähige Projektmanagementlösung suchen. Ein Nachteil könnte sein, dass die Tabellenkalkulationsstruktur für sehr komplexe Projekte oder für Benutzer, die eine visuellere Darstellung bevorzugen, nicht ideal ist [Smartsheet Inc. 2024].

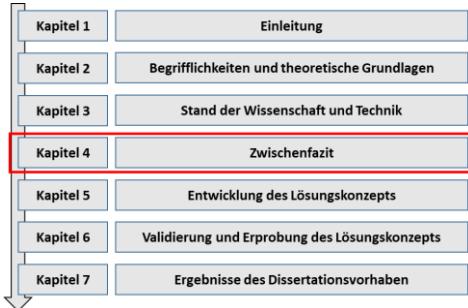
Zusammenfassend lässt sich sagen, dass Asana, Trello, Monday.com, Wrike, Smartsheet, Meisterplan, awork und Can Do jeweils einzigartige Funktionen und Vorteile bieten, die sie für verschiedene Arten von Teams und Projekten geeignet machen. Während Asana und Monday.com durch ihre umfassenden Funktionen und Anpassungsoptionen glänzen, bieten Trello und Smartsheet benutzerfreundliche und flexible Lösungen, die besonders für kleinere Teams und weniger komplexe Projekte attraktiv sind. Wrike hingegen bietet fortschrittliche Funktionen für größere Teams und komplexe Projekte, könnte jedoch für kleinere Teams zu komplex sein. Meisterplan besticht durch seine Szenarioplanung und Benutzerfreundlichkeit, während awork durch seine kollaborativen Funktionen und einfache Bedienung überzeugt. Can Do bietet die fortschrittlichsten KI-Funktionen mit prädiktiven Analysen und Risikomanagement, ist jedoch komplexer in der Anwendung. In Bezug auf KI-Komponenten bieten alle genannten Softwares Automatisierungsfunktionen, die dazu beitragen, repetitive Aufgaben zu reduzieren und die Effizienz zu steigern. Die Tabelle 8 fasst die Ergebnisse dieser Analyse zusammen.

Tabelle 8: Zusammenfassung der Auswertung von ausgewählten Softwarelösungen

Softwares	Kriterien	Projektmanagement	Ressourcenmanagement	KI-Funktionen
Microsoft Project	●	●	○	○
Microsoft Excel (Vorlagen)	○	○	○	○
SAP PPM	●	●	○	○
Saviom	●	○	○	○
Teamdeck	●	●	○	○
Float	●	○	○	○
Projektron BCS	●	●	○	○
Asana	●	○	○	○
Trello	○	○	○	○
Monday.com	●	○	○	○
Meisterplan	●	○	○	○
Awork	●	○	○	○
Can Do	●	○	○	○
Wrike	●	●	○	○
Smartsheet	●	○	○	○

Die KI-Funktionen aller genannten Softwares zielen darauf ab, die Aufgaben zu automatisieren und unterschiedliche Automatismen zu realisieren. Außer Wrike und Can Do wird die KI nicht genutzt, um wirkliche Einblicke zu generieren, sondern hauptsächlich, um repetitive Aufgaben zu eliminieren. Jedoch ist der Umfang der KI-Analysen auf Daten von Projekten beschränkt, die innerhalb dieser Softwares erstellt wurden. Projekte, die zuvor in anderen Softwarelösungen angelegt wurden, können dabei nicht berücksichtigt werden. Dies unterstreicht erneut den Forschungsbedarf für die „Entwicklung eines KI-basierten Workflows zur Unterstützung der Projektleitung in Bezug auf eine kompetenzbasierte Ressourcenplanung von Industrieprojekten“.

4 Zwischenfazit



Basierend auf den bisherigen Ausführungen zu den zentralen Begriffen, theoretischen Grundlagen und dem aktuellen Stand der Technik wird in diesem Kapitel ein Zwischenfazit der Dissertation gezogen. Dieses Zwischenfazit verfolgt das Ziel, die bisherigen Erkenntnisse zusammenzuführen und daraus die Anforderungen an das zu entwickelnde Konzept abzuleiten. Im ersten Teil des Kapitels wird der Bedarf eines KI-

gestützten Workflows für die kompetenzbasierte Projektplanung dargelegt. Hierbei wird aufgezeigt, in welchen Bereichen traditionelle Planungsmethoden an ihre Grenzen stoßen und welche spezifischen Vorteile der Einsatz von Künstlicher Intelligenz (KI) bietet, um eine dynamische und zielgerichtete Ressourcenplanung zu ermöglichen. Insbesondere wird darauf eingegangen, wie KI zur Verbesserung der Projektplanung beiträgt, indem sie Kompetenzprofile effizienter analysiert und die Zuweisung von Aufgaben basierend auf individuellen Fähigkeiten und Projektanforderungen optimiert. Im zweiten Teil des Kapitels werden die Anforderungen an diesen KI-gestützten Workflow detailliert beschrieben. Dabei wird ein besonderes Augenmerk auf die funktionalen und nicht-funktionalen Anforderungen gelegt, die für die Entwicklung eines erfolgreichen Konzeptes entscheidend sind. Diese Anforderungen basieren sowohl auf der Analyse des aktuellen Stands der Technik als auch auf den identifizierten Schwachstellen bestehender Systeme. Ziel ist es, eine belastbare Grundlage für die nachfolgende Entwicklung eines maßgeschneiderten Workflows zur kompetenzbasierten Projektplanung (KIPrOko) zu schaffen, der den spezifischen Anforderungen der Praxis gerecht wird.

Durch dieses Kapitel wird eine Brücke zwischen den theoretischen Grundlagen und der praktischen Umsetzung geschlagen, die im weiteren Verlauf der Dissertation vertieft wird.

4.1 Bedarf eines KI-gestützten Workflows zur kompetenzbasierten Projektplanung

Die Dissertation adressiert die Herausforderung der kompetenzbasierten Ressourcenplanung in Industrieprojekten, insbesondere in der Softwareentwicklung. Ein zentrales Problem besteht darin, dass wertvolle Informationen aus abgeschlossenen Projekten oft verloren gehen und die Planung stark von der Erfahrung und dem Wissen einzelner Mitarbeiter abhängig ist. Traditionelle Planungsmethoden können den wachsenden Anforderungen und der zunehmenden Komplexität nicht immer gerecht werden, was zu ineffizienten Entscheidungen und Projektausfällen führt. Die Zielsetzung der Dissertation ist daher die Entwicklung eines KI-basierten Workflows, der Projektleitern bei der kompetenzbasierten Ressourcenplanung unterstützt. Durch die systematische Analyse von Dokumentationen

früherer Projekte soll der Projektmanager frühzeitig und fehlerfrei Entscheidungen treffen können, um den Projekterfolg zu sichern. Der Stand der Wissenschaft und Technik zeigt, dass es bereits verschiedene Methoden und Ansätze im Projektmanagement gibt, einschließlich traditioneller, agiler und hybrider Modelle. Diese Methoden haben jedoch ihre Grenzen in Bezug auf die Handhabung komplexer und dynamischer Projektanforderungen. Zudem werden moderne Technologien wie Künstliche Intelligenz (KI) und Natural Language Processing (NLP) in verschiedenen Bereichen erfolgreich eingesetzt, etwa in der Medizin und im Wissensmanagement. Allerdings ist die Anwendung von NLP im Kontext der kompetenzbasierten Ressourcenplanung für Softwareentwicklungsprojekte noch wenig erforscht. Auf Basis dieser Analyse wird klar, dass ein erheblicher Forschungsbedarf im Bereich der KI-gestützten Ressourcenplanung besteht. Die Implementierung eines KI-basierten Workflows, der unstrukturierte Projektinformationen systematisch analysiert und für die Planung nutzt, kann dazu beitragen, die Abhängigkeit von individuellem Expertenwissen zu reduzieren. Ein solcher Workflow ermöglicht es, relevante Informationen effizient zu extrahieren und aufzubereiten, um fundierte und zeitnahe Entscheidungen in der Projektplanung zu treffen. Dies würde nicht nur die Effizienz und Qualität der Ressourcenplanung verbessern, sondern auch die Erfolgsrate von Projekten erhöhen und potenzielle Fehlerquellen minimieren.

4.2 Anforderungen an den KI-gestützten Workflows zur kompetenzbasierten Projektplanung (KIProKo)

Da es sich um die Neuentwicklung eines KI-Workflows handelt, ist es notwendig, die Anforderungen, die dieser erfüllen muss, zu erläutern. Diese Anforderungen werden in den folgenden Unterkapiteln ausführlich dargestellt.

4.2.1 Grundlegende Anforderungen (G)

Im Kontext moderner Projektmanagementpraktiken ist es von zentraler Bedeutung, dass KI-gestützte Workflows die grundlegenden Anforderungen erfüllen, um ihre Effektivität und Effizienz zu gewährleisten. Diese Anforderungen sind aus der Literatur abgeleitet und bilden die Grundlage für die erfolgreiche Integration von KI-Lösungen in bestehende Projektmanagement- und IT-Systeme. In diesem Sinne wurden die folgenden grundlegenden Anforderungen an das Gesamtkonzept definiert:

1. **Integration** bestehender Systeme: Das Konzept muss in bestehende Projektmanagement- und IT-Systeme integriert werden können, um die Nutzung vorhandener Daten und Prozesse zu erleichtern [Wilson und Perumal 2010; Schatz et al. 2014].
2. **Skalierbarkeit**: Das Konzept muss skalierbar sein, um sowohl kleine als auch große Projekte effizient unterstützen zu können. Dabei darf die ausgewählte Projektmanagementmethodik und die Größe des Projektes keine Rolle spielen [Schlüter 2023; Thiebes und Plankert 2014].

3. **Intuitivität:** Das Konzept soll so intuitiv gestaltet sein, dass Projektmanager ohne umfangreiche Schulung effektiv damit arbeiten können [Müller-Lindenberg 2005; Hartmann 2018].
4. **Datenschutz** und Sicherheit: Es müssen Mechanismen implementiert werden, um die Vertraulichkeit und Sicherheit der Daten zu gewährleisten, insbesondere im Hinblick auf personenbezogene Informationen [Gembrys und Herrmann 2009; Schmitt et al. 2016].
5. **Flexibilität** und Anpassungsfähigkeit: Das Konzept soll flexibel sein, um an unterschiedliche Projektarten und -bedingungen angepasst werden zu können [Grimm 2003; Abramovici und Herzog 2016].

Diese grundlegenden Anforderungen gewährleisten, dass der KI-Workflow robust, sicher und benutzerfreundlich ist und gleichzeitig die Fähigkeit besitzt, sich an verschiedene Projektgrößen und -bedingungen anzupassen.

4.2.2 Anforderungen an den NLP-Algorithmus (N)

Im Projektmanagement können NLP-Algorithmen erheblich dazu beitragen, relevante Informationen effizient aus unstrukturierten Daten zu extrahieren und dadurch die Planung und Durchführung von Projekten zu verbessern. Basierend auf einer umfassenden Analyse der aktuellen wissenschaftlichen Literatur wurden spezifische Anforderungen an einen NLP-Algorithmus definiert, um sicherzustellen, dass dieser den hohen Ansprüchen in verschiedenen Projektphasen gerecht wird. Diese Anforderungen umfassen:

1. **Präzision** und Genauigkeit: Der NLP-Algorithmus muss in der Lage sein, relevante Informationen aus unstrukturierten Projektdokumentationen mit hoher Genauigkeit zu extrahieren [Kreutzer 2019; Ionescu et al. 2017].
2. Erkennung von **Schlüsselkompetenzen**: Der Algorithmus muss Schlüsselkompetenzen identifizieren können, die für die Durchführung zukünftiger Projekte relevant sind [Erpenbeck et al. 2013; Sauter und Staudt 2016b].
3. Sprach- und **Kontextverständnis**: Der NLP-Algorithmus muss unterschiedliche Sprachvarianten und Kontexte innerhalb der Projektdokumentationen verstehen und verarbeiten können [Farid und Mitropoulos 2012; Sundararajan et al. 2018].
4. Integration der **zusätzlichen Ontologien**: Der NLP-Algorithmus soll auf zusätzliche Ontologien antrainierbar sein, damit die spezifischen unternehmensbezogenen Informationen richtig interpretiert werden können [Ionescu et al. 2017].
5. Automatische **Kategorisierung**: Der Algorithmus muss erkannte Kompetenzen automatisch kategorisieren können, um die Planung und Zuweisung der Kompetenzen in kommenden Projekten zu ermöglichen [Sundararajan et al. 2018; Sharma et al. 2019].
6. Kontinuierliches **Lernen**: Der Algorithmus soll in der Lage sein, kontinuierlich aus neuen Daten und Projektdokumentationen zu lernen, um seine Genauigkeit und Effizienz zu verbessern [Ionescu et al. 2017].

4.2.3 Anforderungen an den KI-gestützten Workflow (W)

Ein gut konzipierter KI-gestützter Workflow kann erheblich zur Effizienzsteigerung und Qualitätssicherung beitragen, indem er große Datenmengen verarbeitet und relevante Informationen extrahiert. Die folgenden Anforderungen wurden auf Basis aktueller wissenschaftlicher Erkenntnisse definiert, um sicherzustellen, dass der Workflow den Ansprüchen des Projektmanagements gerecht wird. Diese Anforderungen stellen sicher, dass der Workflow automatisiert, integriert und benutzerfreundlich ist und kontinuierlich verbessert werden kann:

1. **Integration von NLP-Ergebnissen:** Die Ergebnisse der NLP-Analyse müssen in den Workflow integriert werden, um eine kompetenzbasierte Ressourcenplanung zu ermöglichen [Zichler und Helke 2017].
2. **Benachrichtigungssystem:** Der Workflow soll ein Benachrichtigungssystem enthalten, das Projektmanager über relevante Erkenntnisse und Empfehlungen informiert [Sauter und Staudt 2016b].
3. **Benutzerdefinierte Berichte:** Der Workflow soll in der Lage sein, benutzerdefinierte Berichte zu erstellen, die auf den spezifischen Anforderungen und Bedürfnissen des Projekts basieren [Thiebes und Plankert 2014].
4. **Validierung und Feedback:** Der Workflow muss Mechanismen zur Validierung der Ergebnisse und zur Einholung von Feedback von den Benutzern enthalten, um kontinuierliche Verbesserungen zu ermöglichen [Grimm 2003].

Alle Anforderungen, die in 3 Kategorien eingeteilt sind, werden unten in Tabelle 9 aufgeführt und mit einer ID versehen:

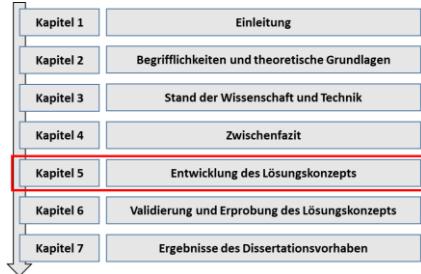
Tabelle 9: Auflistung aller definierten Anforderungen

ID	Anforderung	Bereich
G1	Das Konzept muss in bestehende Projektmanagement- und IT-Systeme integriert werden.	Grundlegend (G)
G2	Das Konzept muss bezüglich unterschiedlichen Projektgrößen skalierbar sein.	
G3	Das Konzept soll Intuitiv gestaltet werden.	
G4	Es muss die Vertraulichkeit und Sicherheit der Daten zu gewährleisten.	
G5	Das Konzept soll in Bezug auf Projektart und -bedingung flexibel sein.	
N1	Der Algorithmus muss mit hoher Genauigkeit arbeiten.	NLP-Algorithmus (N)
N2	Der Algorithmus muss Schlüsselkompetenzen identifizieren können.	
N3	Der Algorithmus muss unterschiedliche Inhalte innerhalb der Projektdokumentationen verstehen und verarbeiten.	
N4	Der NLP-Algorithmus soll auf zusätzliche Ontologien antrainierbar sein.	
N5	Der Algorithmus muss erkannte Kompetenzen automatisch kategorisieren können.	
N6	Der Algorithmus soll kontinuierlich aus neuen Daten lernen.	

W1	Die Ergebnisse der NLP-Analyse müssen in den Workflow integriert werden.	
W2	Der Workflow soll ein Benachrichtigungssystem enthalten.	Workflow
W3	Der Workflow soll in der Lage sein, benutzerdefinierte Berichte zu erstellen.	(W)
W4	Der Workflow muss, wegen KVP, Feedbacks von den Benutzern einholen.	

Die definierten Anforderungen bilden die Basis für die Entwicklung eines effektiven und effizienten KI-basierten Workflows zur Unterstützung der Projektleitung hinsichtlich einer kompetenzbasierten Ressourcenplanung von Industrieprojekten. Diese Anforderungen stellen sicher, dass der gesamte Prozess – von der Datenanalyse über die Integration und Verarbeitung bis hin zur Bereitstellung der Ergebnisse und kontinuierlichen Verbesserung – robust und zielgerichtet ist. Im nächsten Kapitel wird die praktische Umsetzung dieser Anforderungen behandelt. Auf den in diesem Kapitel beschriebenen Grundlagen und wissenschaftlichen Erkenntnissen basierend, wird ein detailliertes Konzept entwickelt, welches die beschriebenen Anforderungen erfüllt und dabei die Effizienz und Genauigkeit des Projektmanagements in Planungsphase verbessert.

5 Entwicklung des KI-gestützten Workflows zur kompetenzbasierten Projektplanung



Basierend auf dem aktuellen Stand der Technik und den daraus abgeleiteten Anforderungen soll nun das theoretische Konzept von KI-gestützten Workflow entwickelt und die praktische Umsetzung erarbeitet werden. Dabei sollen die Erkenntnisse aus den aktuellen Ansätzen als Grundlage dienen. Das Lösungskonzept besteht aus drei Hauptschritten: S1) Vorbereitung S2) NLP-Algorithmus S3) Workflow. Die Abbildung 19 visualisiert die drei Hauptschritte des Lösungskonzeptes. Der erste Schritt besteht darin, in Vorbereitungsphase eine Dokumentations- und Informationsgrundlage zu schaffen, damit im nächsten Schritt eine erfolgreiche NLP-Analyse durchgeführt werden kann.

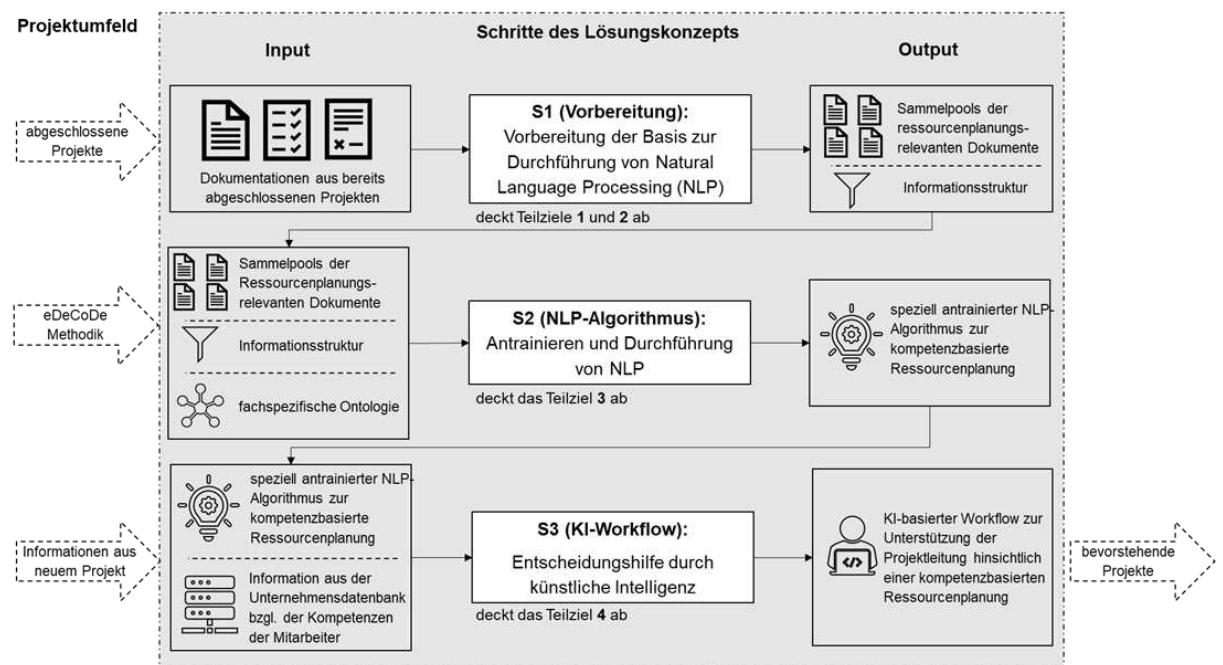
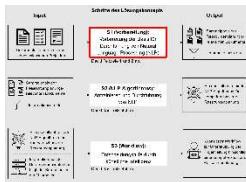


Abbildung 19: Darstellung des Lösungskonzeptes [Eigene Darstellung]

Der Schritt 2 (S2: NLP) befasst sich mit der Entwicklung einer NLP-Pipeline für die Textanalyse in natürlicher Sprache und der Untersuchung unterschiedlicher Pipeline-Strukturen sowie Implementierungsansätze. Parallel dazu wird eine auf eDeCoDe und Automotive SPICE basierende Ontologie erstellt, um die Verbindungen zwischen den durch die Pipeline verarbeiteten Informationen zu definieren und die Logik von eDeCoDe auf Automotive SPICE anzuwenden. Dadurch werden die Beziehungen zwischen den Systemartefakten in Softwareentwicklungsprojekten verdeutlicht. Darüber hinaus wird in Schritt 3 eine Benutzeroberfläche entwickelt, die es dem Projektmanager ermöglicht,

Informationen über das neue Projekt einzugeben und Vorhersagen abzurufen. Zudem wird der entwickelte NLP-Algorithmus in den Workflow integriert. Es ist zu bemerken, dass dieser praktische Teil der Validierung und Erprobung dient und hilft, die Funktionalität und Effektivität des theoretischen Konzepts vom KI-gestützten Workflow zur kompetenzbasierten Projektplanung zu überprüfen und zu verbessern, was im Kapitel 6 ausführlich diskutiert wird.

5.1 Theoretische Entwicklung der Vorbereitungsphase (S1)



Im ersten Schritt (S1: Vorbereitung) werden zwei Themen behandelt, um nachher eine erfolgreiche Implementierung von NLP zu ermöglichen. Dazu ist es zunächst notwendig, zu identifizieren, welche Dokumente in einem Projekt und auch auf der Organisationsebene üblicherweise Informationen zur Ressourcenplanung enthalten. Diese relevanten Dokumente werden dann in einem Dokumentensammelpool für die weitere Analyse zusammengeführt. Um die gesammelten Dokumente anschließend mit Hilfe von NLP analysieren zu können und die gewünschten Erkenntnisse zu generieren, ist es notwendig, die Informationen aus dem Text der Dokumente zu filtern und zu kategorisieren. Zu diesem Zweck wird eine gezielte Informationsstruktur entwickelt, die es ermöglicht, die in den Dokumenten enthaltenen Informationen in Bezug auf die verschiedenen Ressourcen zu unterscheiden und entsprechend zu kategorisieren.

5.1.1 Identifikation von relevanten Informationen und Dokumenten

Zunächst werden die Dokumente identifiziert, die typischerweise Informationen für die Ressourcenplanung in einem Projekt sowie auf organisatorischer Ebene enthalten. Nicht alle Projektdokumente sind ressourcenbezogen, daher ist es wichtig zu ermitteln, welche Dokumente relevante Informationen liefern. Die **Informationen**, die im Rahmen dieser Forschung in Betracht gezogen worden sind, umfassen personeller, zeitlicher und finanzialer Ressourcen. Bei den **personellen** Ressourcen liegt der Fokus auf Analyse der Kompetenzen und Kompetenzarten der Mitarbeiter. Bei der Kategorisierung von Mitarbeiterkompetenzen geht man traditionell von einer Aufteilung in Hard-Skills und Soft-Skills aus [Jakoby 2015]. Hard-Skills, die die beruflichen und fachlichen Qualifikationen eines Mitarbeiters umfassen, spielen eine zentrale Rolle bei der Auswahl passender Projektmitarbeiter. Der Projektstrukturplan, inklusive der definierten Arbeitspakete und der hierfür notwendigen Qualifikationen, dient als „fundamentale Basis für die Identifikation geeigneter Kandidaten“ [Jakoby 2015, S. 325]. Fachliche Fertigkeiten, als erlernbare und nachweisbare Qualifikationen, entstehen durch formale Bildung wie Schule oder Universität, Berufsausbildungen oder praktische Erfahrungen und werden durch entsprechende Zeugnisse oder Zertifikate nachgewiesen und sind deshalb einfach zu erfassen. Die spezifischen Kenntnisse und deren Niveau, über die ein Angestellter verfügt, können für Unternehmen entscheidende Faktoren sein. Im Laufe der beruflichen Entwicklung können sich darüber hinaus

spezialisierte Hard-Skills hinzufügen, wie Sprachkenntnisse, IT-Fähigkeiten oder branchenspezifisches Wissen [IONOS 2023]. Diese Qualifikationspalette ermöglicht es Unternehmen, die Eignung von Personen für spezifische Aufgaben zu bewerten.

Ergänzend zu den Hard-Skills, wird der Erfolg wesentlich durch die Kooperation mit anderen Menschen beeinflusst. Die Wichtigkeit der hierfür benötigten Soft-Skills wird durch den „Global Talent Trends Report“ von LinkedIn unterstrichen, der 5.000 Fachkräfte und Personalverantwortliche aus 35 Ländern befragte. Hierbei bestätigten 80 Prozent der Teilnehmenden, dass die Soft-Skills der Mitarbeiter zunehmend zum entscheidenden Faktor für den Unternehmenserfolg avancieren [LinkedIn 2019]. Insbesondere in der Projektarbeit, die durch die Notwendigkeit der fachübergreifenden Zusammenarbeit, zeitliche Restriktionen und kontinuierlichen Erfolgsdruck gekennzeichnet ist, erweisen sich Soft-Skills wie Belastbarkeit, geistige und zeitliche Anpassungsfähigkeit oder Kommunikationskompetenz als essentiell [Jakoby 2015]. Die Herausforderung liegt jedoch in der Erfassung dieser fachübergreifenden Fähigkeiten, die sich im Vergleich zu Hard-Skills als deutlich schwerer fassbar erweisen. Die Folgende Abbildung 20 fasst die Soft-Skills der Mitarbeiter auf Basis von unterschiedlichen wissenschaftlichen Literaturen zusammen.

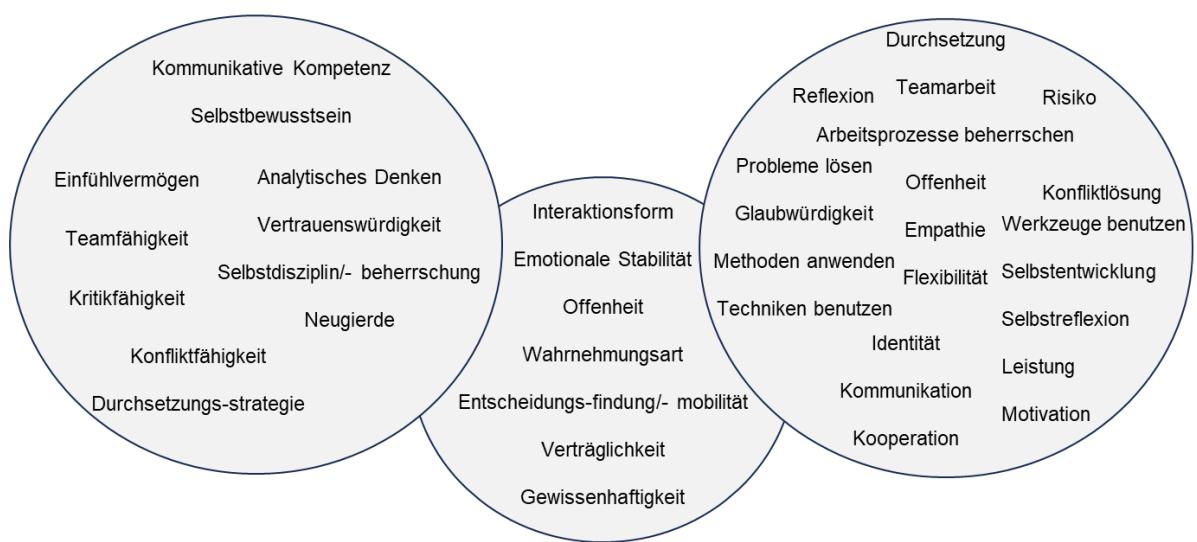


Abbildung 20: Soft-Skills in Anlehnung an Links [Peters-Kühlinger und John 2022], Mitte [Jakoby 2015], und Rechts [Bohinc 2012]

Eine weiterführende Differenzierung dieser Fähigkeiten orientiert sich an der von Weinert vorgeschlagenen Definition von Kompetenzen. Laut Weinert umfassen Kompetenzen die kognitiven Fähigkeiten und Fertigkeiten einer Person, Probleme zu lösen, sowie die zugehörigen motivationalen, volitionalen und sozialen Kompetenzen, um diese Lösungen in verschiedenen Situationen erfolgreich und verantwortlich anwenden zu können [Weinert 2014]. Auf dieser Grundlage lassen sich Kompetenzen in vier Hauptkategorien von **Fachkompetenz**, Methodische Kompetenz, Persönliche Kompetenz, und Soziale Kompetenz einteilen. Unter dem Begriff der „Fachkompetenz“ fasst man jene Fähigkeiten, Fertigkeiten und Kenntnisse zusammen, die notwendig sind, um berufliche Aufgaben erfolgreich

zu bewältigen. Fachkompetenz zeichnet sich darin ab, dass Personen aufgrund ihrer beruflichen Bildung in der Lage sind, Projekte oder Aufgabenstellungen effektiv zu meistern [North 2005]. Die **methodischen Kompetenzen** wiederum beziehen sich auf die Anwendung von Fachwissen in unterschiedlichen Situationen und die flexible Integration dieses Wissens in komplexe Arbeitsabläufe mit dem Ziel, diese effizient zu gestalten. Sie gelten als übergeordnete Schlüsselkompetenzen, die nicht an spezifische Tätigkeiten gebunden sind. Ein Beispiel hierfür ist die Fähigkeit zu präsentieren, die sich durch den Einsatz diverser, auf die Zielgruppe abgestimmter Präsentationstechniken und -medien auszeichnet [North 2005]. **Persönliche Kompetenzen** beschreiben die individuelle Persönlichkeitsentwicklung, die durch unterschiedliche Lebenserfahrungen geformt wird. Schließlich beschreiben die **sozialen Kompetenzen** das Vermögen, in einem beruflichen Kontext innerhalb eines Teams kooperativ und selbstständig zu agieren, was die Entwicklung und Umsetzung von zielgerichteten Plänen ermöglicht. Diese Fähigkeiten werden im Laufe des Lebens in verschiedenen sozialen Kontexten, wie Familie, Freundschaften und Bildungseinrichtungen, erworben [North 2005]. Zusätzlich wird eine Kategorie definiert, die die **Beschäftigungsarten** der Teammitglieder innerhalb eines Projekts beschreibt. Die Kategorisierung der personellen Ressourcen ist in Tabelle 10 zusammengefasst.

Tabelle 10: Übersicht der personellen Ressourcen [Eigene Darstellung]

Fachkompetenz	Methodische Kompetenz	Persönliche Kompetenz	Soziale Kompetenz	Mitarbeiteranzahl
• Berufliche Qualifikationen	• Präsentationsfähigkeit • Entscheidungsfindung • Entscheidungsmobilität	• Emotionale Stabilität • Wahrnehmungsart • Gewissenhaftigkeit • Selbstbewusstsein • Selbstdisziplin • Selbstbeherrschung • Neugierde • Selbstentwicklung • Selbstreflexion • Leistung • Flexibilität • Lernfähigkeit • Komplexitätsbewältigung • Glaubwürdigkeit • Motivation • Identität • Innovationsfähigkeit • Kreativität • Belastbarkeit	• Interaktionsform • Verträglichkeit • Teamfähigkeit • Konfliktlösung • Vertrauenswürdigkeit • Kommunikativ • Kontaktfähigkeit • Kritikfähigkeit • Kooperation • Durchsetzung • Konfliktlösung • Empathie	• Interne Mitarbeiter in Vollzeit • Interne Mitarbeiter in Teilzeit • Externe Mitarbeiter in Vollzeit • Externe Mitarbeiter in Teilzeit
• Weiterbildungen und Zusatzqualifikationen				
• Sprachkenntnisse	• Analytisches Denken			
• EDV-Kenntnisse	• Durchsetzungsstrategie			
• Fachwissen	• Problem lösen			
• Führerscheine	• Arbeitsprozesse beherrschen • Methoden anwenden • Techniken benutzen • Werkzeuge benutzen			

Zeit stellt in der Projektplanung eine äußerst wertvolle Ressource dar, weil sie unwiederbringlich und unersetzbbar ist. Die Allokation zeitlicher Ressourcen ist daher ein zentraler Faktor in der Projektplanung [Zimbardo und Boyd 2011]. Die Terminvorgaben eines Projekts, sei es durch fixe Fristen oder flexible Zeitfenster, bilden einen wesentlichen Bestandteil der zeitlichen Anforderungen. Die Planung der zeitlichen Dimension eines Projekts ist somit eine kritische Komponente des Projektmanagements. Oftmals wird jedoch die Planung zeitlicher Ressourcen vernachlässigt und erst am Ende umfangreicher Projektvorbereitungen durchgeführt, häufig von Mitarbeitenden, die dafür unzureichend vorbereitet sind oder keine entsprechenden Erfahrungen haben [Madauss 2020]. Diese unbedachten Annahmen führen später häufig zu Problemen, die nicht eingehalten werden können und erhebliche Konsequenzen nach sich ziehen. Die Projektplanung erfordert besonderen Aufwand und muss daher mit adäquaten zeitlichen Ressourcen unterstützt werden. Eine gründliche Planungsphase legt das Fundament für ein professionell durchgeführtes Projekt und minimiert das Risiko von Abweichungen im späteren Projektverlauf. Wie in Abbildung 21 illustriert, zahlt sich ein hoher initialer Aufwand in der Planungsphase über die gesamte Projektdauer aus. Ein Mangel an investierter Zeit in dieser frühen Phase kann zu einem erheblich gesteigerten Aufwand in späteren Projektphasen führen [Sterrer 2014]. Da die Erfassung von unterschiedlichen Informationen in der Praxis unterschiedlich behandelt wird und keine festen Vorgaben gibt, wo genau welche Informationen zu erfassen sind, wird in dieser Forschung der Fokus auf Managementdokumentationen gelegt. Hierzu werden die Dokumente, wie Projekthandbuch und Buchhaltung parallel zu Personal-Dokumenten, wie digitale Personalakte und Skill-Datenbank in Betracht gezogen, die später in dieser Unterkapitel ausführlich beschrieben werden.

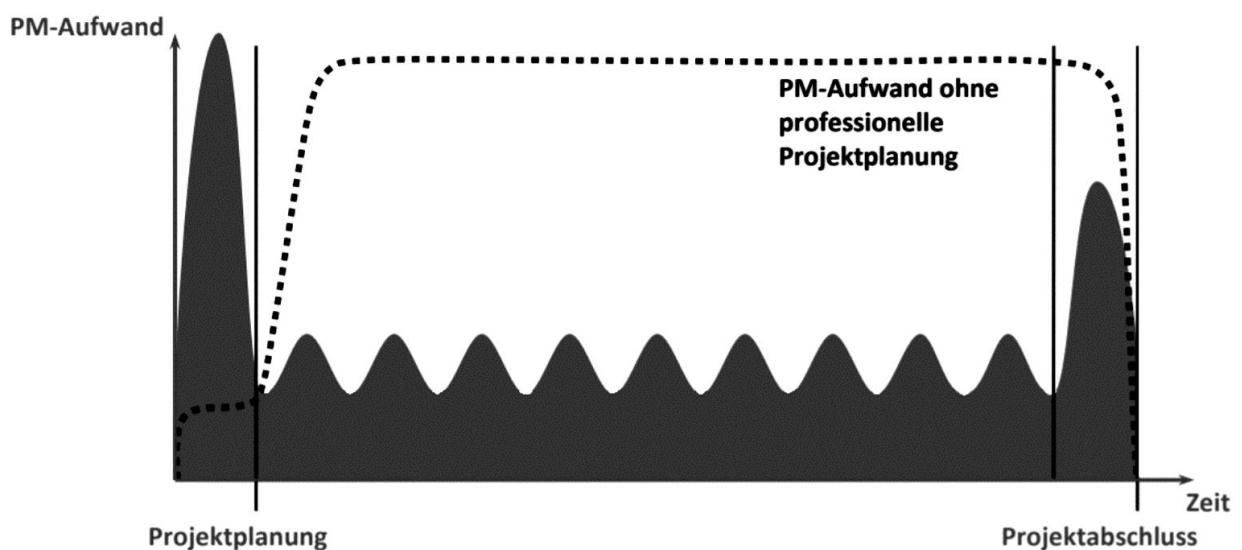


Abbildung 21: Projektmanagementaufwand im Projektverlauf in Anlehnung an [Sterrer 2014]

Basierend auf Stand der Wissenschaft und Technik lassen sich die zeitlichen Ressourcen in einem Projekt in drei Hauptkategorien gliedern (s. Kapitel 3.1). Die erste Kategorie, bezeichnet als „**Pla-**nung“, beinhaltet sämtliche Aktivitäten, die in der Vorbereitungsphase eines Projekts anfallen. Dazu gehören die Erstellung von Arbeitspaketen und Prozessabläufen, die Erarbeitung von Ressourcen-,

Kosten- und Zeitplänen sowie das Zusammenstellen des Projektteams und das Durchführen einer umfassenden Risikoanalyse. Entscheidend für diese Kategorie ist die jeweils erforderliche Zeitdauer für die einzelnen Aktivitäten. Die zweite Kategorie umfasst alle Aktivitäten, die für die Durchführung des Projekts erforderlich sind. Diese Aktivitäten werden während der Projektdurchführung in verschiedene Phasen unterteilt, wobei die Dauer jeder Phase von besonderer Bedeutung ist. Diese Kategorie wird unter dem Begriff „**Durchführung**“ zusammengefasst. Der Projektabschluss wird aufgrund seines zeitaufwändigen Dokumentationsprozesses einer eigenen Kategorie zugeordnet und unter dem Begriff „**Abschluss**“ geführt.

Eine vierte Gruppe, die auf der Analyse mehrerer Projektplanungssoftware basiert, betont die Wichtigkeit der präzisen Erfassung von Schlüsseldaten des Projektes. Diese Gruppe, bezeichnet als „**Eckdaten**“, umfasst geplante und tatsächliche Start- und Endtermine sowie die Gesamtdauer des Projektes, einschließlich eventueller Abweichungen zwischen geplanter und tatsächlicher Projektdauer. Tabelle 11 bietet eine Übersicht über diese zeitlichen Ressourcen und die zugehörigen Oberbegriffe.

Tabelle 11: Übersicht der zeitlichen Ressourcen [Eigene Darstellung]

Planung	Durchführung	Abschluss	Eckdaten
<ul style="list-style-type: none"> • Planung der Arbeitspakete • Planungsphase • Planung der einzelnen Vorgänge • Kostenplanung • Ressourcenplanung • Festlegung des Teams und deren Rollen • Risikoanalyse • Terminplanung 	<ul style="list-style-type: none"> • Anlaufphase • Einzelne Projektphasen • Teamentwicklungsmaßnahmen • Erlernen neuer, für das Projekt erforderliche Fähigkeiten 	<ul style="list-style-type: none"> • Dokumentation des Projektes • Abgabephase des Projektes • Lessons Learned 	<ul style="list-style-type: none"> • Gesamtdauer des Projektes • Lange Lieferzeiten • Festgelegte Start- u. Endtermine • Personelle Krankheitsausfälle • Prozentuale Abweichung der geplanten Gesamtdauer

Zeitliche Beschränkungen sind eine zentrale Limitierung bei Projekten, jedoch spielen auch die Kosten bzw. die **finanziellen Ressourcen** eine wesentliche Rolle. Im betrachteten Kontext definiert man Kosten als „den Geldausdruck aller Aufwendungen, die durch den Einsatz von Ressourcen zur ordnungsgemäßen Durchführung eines Projekts entstehen“ [Känel 2020, S. 210]. Diese Kosten müssen grundsätzlich vom Auftraggeber getragen und in einem Auftragsdokument festgehalten werden, welches die vom Auftraggeber zu bereitstellendem Finanzmittel spezifiziert. Das Erreichen dieser Kostenziele ist ein maßgeblicher Indikator für den Projekterfolg [Jakoby 2015]. Daher ist die Präzision in der Schätzung der benötigten finanziellen Mittel von großer Bedeutung. Unzutreffende Schätzungen können starke Kritik auslösen und ernsthafte Finanzierungsprobleme nach sich ziehen, die im schlimmsten Fall zum Projektabbruch und zu Imageverlusten führen können, unabhängig davon, ob

die Fehlplanung bewusst oder unbewusst erfolgte [Madauss 2020]. Die Kostenschätzung eines Projektes basiert auf einer Kostenanalyse. Unter Kostenanalyse versteht man das Sammeln und Auswerten von Kostendaten abgeschlossener Projekte, um eine Kostendatenbank zu erstellen, die als Grundlage für die Kostenschätzung zukünftiger Projekte dient. Die Qualität der Kostenschätzung verbessert sich durch die Nutzung von Erfahrungswerten, da viele neue Projekte auf bereits existierenden Komponenten aufbauen oder Analogien zu früheren Teilprojekten hergestellt werden können. Die Auswertung abgeschlossener Projekte gewinnt dadurch an Bedeutung. Die Dokumentation sollte unmittelbar nach Projektabschluss erfolgen und neben den Kostendaten auch charakterisierende Informationen beinhalten [Madauss 2020]. Finanzielle Ressourcen für ein Projekt umfassen eine breite Palette unterschiedlicher Kosten, die in drei Hauptkategorien unterteilt werden. Die erste Kategorie, die „**Personalkosten**“, umfasst alle Ausgaben für die Vergütung der am Projekt beteiligten Mitarbeiter, sowohl intern als auch extern. Die zweite Kategorie, „**Sachmittelkosten**“, bezieht sich auf Ausgaben für die Nutzung von Sachmitteln wie Werkzeuge und Rohstoffe. Eine dritte Kategorie umfasst Kosten, die weder den Personalkosten noch den Sachmittelkosten zuzuordnen sind, wie beispielsweise Steuern oder Versicherungsbeiträge, und wird als „**Sonstige Kosten**“ bezeichnet. Tabelle 12 listet diese Kategorisierung auf.

Tabelle 12: Übersicht der finanziellen Ressourcen [Eigene Darstellung]

Personalkosten	Sachmittelkosten	Sonstige Kosten
<ul style="list-style-type: none"> • Internes Personal • Externes Personal 	<ul style="list-style-type: none"> • Grundstücke und Gebäude • Rohstoffe • Werkzeuge • Hilfsstoffe • Materielle Betriebsmittel • Betriebs- u. Arbeitsplatzausstattung • Maschinen- u. maschinelle Anlagen 	<ul style="list-style-type: none"> • Projektbezogene Aufwände • Steuern • Gebühren • Marketing Aktivitäten • Versicherungen

Nachdem die relevanten Informationen definiert wurden, ist es nun erforderlich zu klären, in welchen Dokumenten diese Informationen abgelegt sind. Die Dokumentation umfasst das Projekthandbuch, die digitale Personalakte, die Skill-Datenbank sowie die Informationen, die aus der Buchhaltung bezogen werden können. Das **Projekthandbuch** ist eine Zusammenstellung von Informationen, Normen und Vorschriften, die für ein bestimmtes Projekt gelten [DIN 69901-5: 2009, S. 13]. Das Projekthandbuch enthält projektspezifische Informationen zur Durchführung des Projekts, u. a. Angaben über die erforderliche Anzahl von Mitarbeitern und Zeitressourcen. Außerdem enthält es wichtige Projektdaten wie die Dauer von Planung, Durchführung und Abschluss. Die **Buchhaltung** verwaltet verschiedene Arten von Dokumenten wie Lohnabrechnungen, Steuerbescheide und Belege. Belege sind Nachweise für Geschäftseinnahmen und -ausgaben und müssen gemäß dem Handelsgesetzbuch aufbewahrt werden. Durch die Analyse dieser Buchhaltungsunterlagen lassen sich Informationen über

die finanziellen Mittel vergangener Projekte gewinnen. So lässt sich beispielsweise feststellen, welche Materialien zu welchem Preis und von welchem Hersteller verwendet wurden oder wie hoch die Personalkosten für externe Mitarbeiter bei einem Projekt waren. Diese Informationen können als Orientierungshilfe bei der Planung künftiger Projekte dienen. Ein weiteres Dokument ist die **digitale Personalakte**. Diese ermöglicht eine übersichtliche und komprimierte Darstellung aller relevanten Informationen der Mitarbeiter. Neben Kontaktdaten können beispielsweise auch Informationen zu zertifizierten Aus- und Weiterbildungen in der digitalen Personalakte hinterlegt werden. Je nach Organisation und Aufbau von HR-System des Unternehmens kann die Detailierungsgrad dieser Informationen unterschiedlich sein. So kann eine Verknüpfung zwischen dem Informationsfindungsfilter, Hard-Skills und der digitalen Personalakte hergestellt werden, um entsprechende Beziehungen im Matrixdiagramm herzustellen. Eine **Skill-Datenbank** ist eine Datenbank, die Informationen über die Fähigkeiten und Kompetenzen der Mitarbeiter enthält. Sie dient als internes Repository zur Verfolgung und Verwaltung der Fähigkeiten und Erfahrungen der Mitarbeiter. Durch den Einsatz einer Skill-Datenbank können Unternehmen das Potenzial ihrer Mitarbeiter optimal nutzen und die richtigen Ressourcen für bestimmte Projekte oder Aufgaben zuweisen. So können Unternehmen besser einschätzen, welche Mitarbeiter über die notwendigen Fähigkeiten verfügen, um bestimmte Aufgabenbereiche erfolgreich abzuschließen. Die Skill-Datenbank unterstützt somit die effektive Planung und Zuteilung von Ressourcen innerhalb der unterschiedlichen Projekte in Unternehmen. Um die Beziehung zwischen den relevanten Informationen und den zugehörigen Dokumenten zu verdeutlichen, werden diese in einem Matrixdiagramm gegenübergestellt. Dabei kennzeichnet das Symbol 'x' eine bestehende Verbindung zwischen einer Information und einem Dokument. Tabelle 13 zeigt ein Matrixdiagramm, das die relevanten Informationen den Dokumenten zuordnet, in denen diese Informationen auffindbar sind.

Tabelle 13: Matrixdiagramm zur Gegenüberstellung der relevanten Informationen und den Dokumenten

		Dokumente			
		Projekt-handbuch	Digitale Personalakte	Skill-Datenbank	Buchhaltung
Relevante Informationen					
Personelle Ressourcen	Mitarbeiteranzahl	X			
	Fachkompetenz		X		
	Persönliche Kompetenz			X	
	Soziale Kompetenz			X	
	Methodische Kompetenz			X	
Zeitliche Ressourcen	Eckdaten	X			
	Planung		X		
	Durchführung		X		
	Abschluss		X		
Finanzielle Ressourcen	Personalkosten				X
	Sachmittelkosten				X
	Sonstige Kosten				X

Nachdem die relevanten Informationen identifiziert und die entsprechenden Dokumente, die diese beinhalten, bestimmt wurden, kann nun mit der Entwicklung der Informationsstruktur begonnen werden.

5.1.2 Aufbau einer Informationsstruktur

Um die gesammelten Dokumente mittels NLP zu analysieren und Erkenntnisse für die kompetenzbasierte Ressourcenplanung zu generieren, müssen Informationen aus dem Text der Dokumente extrahiert und kategorisiert werden. Auf der Grundlage von Literaturstudien wird hierzu eine gezielte Informationsstruktur entwickelt, um die in den Dokumenten enthaltenen Informationen zu unterscheiden und zu kategorisieren, insbesondere im Hinblick auf die verschiedenen Ressourcen. Diese Informationsstruktur dient als **Filter**, um das NLP-System in der nächsten Phase zu unterstützen. Die in den Übersichtstabellen (Tabelle 10, Tabelle 11, Tabelle 12) entwickelten InformationsermittlungsfILTER der ersten und zweiten Darstellungsebene werden in der nachstehenden Tabelle 14 zusammengeführt und mit dem Matrixdiagramm aus Tabelle 13 verknüpft. Diese Integration schafft eine klare und strukturierte Übersicht über die Informationsfilter aller Ressourcenarten und die Dokumente, in denen diese Informationen zu finden sind. Die resultierende Tabelle bietet eine umfassende Darstellung aller relevanten Informationen, die für die Ressourcenplanung in der Projektplanungsphase notwendig sind. Die Felder sind farblich markiert, um visuell darzustellen, aus welchen Dokumenten die Informationen stammen. Filter, die Informationen aus dem Projekthandbuch beziehen, sind **rot** markiert, während Filter, die Fachkompetenzen betreffen und aus der digitalen Personalakte stammen, **lila** gefärbt sind. Filter, die ihre Informationen aus der Skill-Datenbank ziehen, sind **blau** hinterlegt, und jene, die Informationen aus der Buchhaltung verwenden, sind **orange** markiert. Die entsprechenden Farbzugehörigkeiten können der Legende oberhalb der Tabelle entnommen werden.

Tabelle 14: Informationsstruktur (Filter) [Eigene Darstellung]

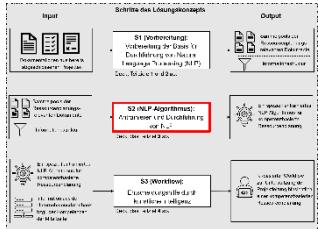
Farbenbedeutung:		• Projekthandbuch	• Digitale Personalakte	• Skill-Datenbank	• Buchhaltung
Ressourcenarten	personelle Ressourcen	Mitarbeiteranzahl	interne Mitarbeiter in Vollzeit interne Mitarbeiter in Teilzeit externe Mitarbeiter in Vollzeit externe Mitarbeiter in Teilzeit		
		Fachkompetenz	berufliche Qualifikationen Weiterbildungen und Zusatzqualifikationen Sprachkenntnisse EDV-Kenntnisse Fachwissen Führerscheine		
			emotionale Stabilität Wahrnehmungsart Gewissenhaftigkeit Selbstbewusstsein		

Farbenbedeutung:	● Projekthandbuch	● Digitale Personalakte	● Skill-Datenbank	● Buchhaltung
Ressourcenarten				
personelle Ressourcen		<p>Persönliche Kompetenz</p> <ul style="list-style-type: none"> Selbstdisziplin Selbstbeherrschung Neugierde Selbstentwicklung Selbstreflexion Leistung Flexibilität Lernfähigkeit Komplexitätsbewältigung Glaubwürdigkeit Motivation Identität Innovationsfähigkeit Kreativität Belastbarkeit Intuition Reflexionsfähigkeit Offenheit Risiko 		
		<p>Soziale Kompetenz</p> <ul style="list-style-type: none"> Interaktionsform Verträglichkeit Teamfähigkeit Konfliktlösung Vertrauenswürdigkeit Kommunikativ Kontaktfähigkeit Kritikfähigkeit Kooperation Durchsetzung Konfliktlösung Empathie 		
		<p>Methodische Kompetenz</p> <ul style="list-style-type: none"> Präsentationsfähigkeit Entscheidungsfindung Entscheidungsmobilität Analytisches Denken Durchsetzungsstrategie Problem lösen Arbeitsprozesse beherrschen Methoden anwenden Techniken benutzen Werkzeuge benutzen 		

		Farbenbedeutung:				
		Projekthandbuch	Digitale Personalakte	Skill-Datenbank	Buchhaltung	
Ressourcenarten	Zeitliche Ressourcen	Eckdaten	Gesamtdauer des Projektes Lange Lieferzeiten Festgelegte Start- u. Endtermine Personelle Krankheits-ausfälle Prozentuale Abweichung der geplanten Gesamtdauer			
			Planung der Arbeitspakete Planungsphase Planung der einzelnen Vorgänge Kostenplanung Ressourcenplanung Festlegung des Teams und deren Rollen Risikoanalyse Terminplanung			
		Durchführung	Anlaufphase Einzelne Projektphasen Teamentwicklungsmaß-nahmen Erlernen neuer, für das Projekt erforderliche Fähigkeiten			
			Abschluss Dokumentation des Projektes Abgabephase des Projektes Lessons Learned			
	Finanzielle Ressourcen	Personalkosten	Internes Personal Externes Personal			
		Sachmittelkosten	Grundstücke und Gebäude Rohstoffe Werkzeuge Hilfsstoffe Materielle Betriebsmittel Betriebs- u. Arbeitsplatzausstattung Maschinen- u. maschinelle Anlagen			
			Sonstige Kosten Projektbezogene Aufwände Steuern Gebühren Marketing Aktivitäten Versicherungen			

Durch die Bearbeitung dieser beiden Themen in S1 wird die Grundlage für eine erfolgreiche NLP-Analyse geschaffen, die es ermöglicht, relevante Informationen für eine kompetenzbasierte Ressourcenplanung zu erhalten.

5.2 Theoretische Entwicklung des NLP-Algorithmus (S2)



Im Rahmen des ersten Schritts (S1) wurde ein solides Fundament für die Realisierung einer effektiven NLP-Analyse geschaffen. Der zweite Schritt (S2: NLP) umfasst zwei Hauptthemen. Einerseits wird eine NLP-Pipeline entwickelt, die sämtliche erforderlichen Schritte der Textverarbeitung umfasst, um Texte, die in natürlicher Sprache verfasst sind, zu verarbeiten, zu klassifizieren und zu analysieren. In diesem Zusammenhang werden verschiedene NLP-Pipeline-Strukturen und Implementierungsansätze untersucht. Andererseits wird eine auf eDeCoDe und Automotive SPICE basierende Ontologie entwickelt, um die Verknüpfungen der durch die NLP-Pipeline verarbeiteten Informationen zu definieren. Dabei wird die Logik von eDeCoDe auf Automotive SPICE angewandt, um die Beziehungen zwischen verschiedenen Systemartefakten innerhalb eines Softwareentwicklungsprojekts zu klären.

5.2.1 Aufbau von NLP-Pipeline

Die implementierte NLP-Pipeline gliedert sich in vier Hauptphasen: von der morphologischen Verarbeitung bis zur pragmatischen Analyse. Als Eingabedaten dienen Dokumentationen aus dem Sammelpool des Schritts 1 (S1) und das Ergebnis ist strukturierter Text. Zusätzlich zieht die Pipeline Umgebungsinformationen heran, einschließlich der Struktur der jeweiligen Sprache (einschließlich Wörterbuch, Grammatik und semantischen Regeln) und kontextuelle Daten bzw. eine Ontologie, die für die Interpretation der verarbeiteten Informationen benötigt wird. Diese Informationen werden durch die eDeCoDe-Logik zur Verfügung gestellt. Die Abbildung 22 illustriert den Aufbau der NLP-Pipeline, einschließlich der vier Hauptverarbeitungsschritte, des Inputs und Outputs sowie der relevanten Umgebungsinformationen.

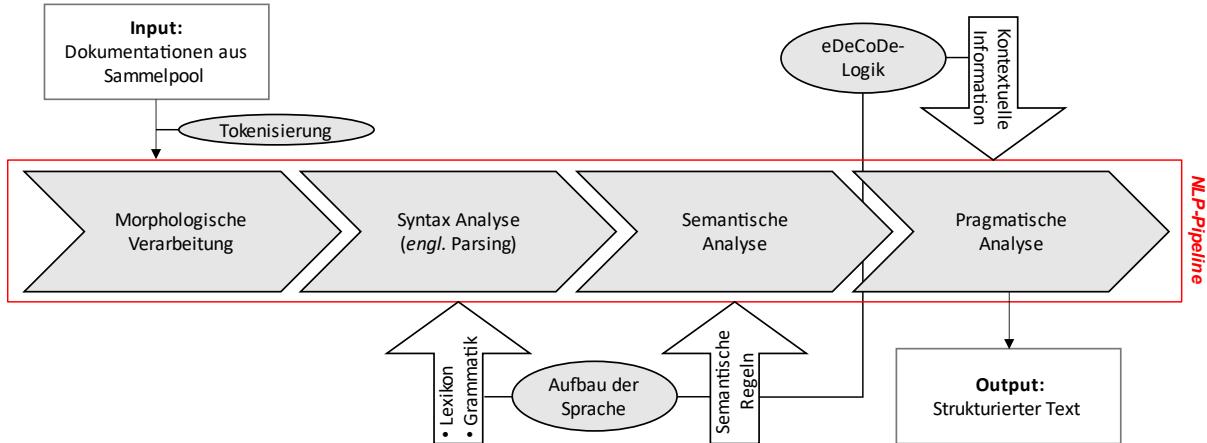


Abbildung 22: Aufbau von NLP-Pipeline [Eigene Darstellung]

In einer NLP-Pipeline stellt die **Tokenisierung** eine essentielle Vorbereitung dar, in dem der Rohtext in kleinere Segmente, sogenannte „Tokens“, zerlegt wird. Diese Tokens können Wörter, Zahlen oder Satzzeichen umfassen. Dieser Vorgang ist fundamental, um den Text für nachfolgende Verarbeitungsschritte aufzubereiten. Tokenisierung beginnt mit der Auswahl des zu verarbeitenden Texts, der aus Sätzen, Absätzen oder ganzen Dokumenten bestehen kann [Taulli 2022]. Beispielsweise könnte man den Satz „Qualitätsmanagement verbessert die Ergebnisse.“ heranziehen. Nach der Auswahl des Texts identifiziert der Tokenizer die Grenzen zwischen den Token, die üblicherweise durch Leerzeichen und Satzzeichen definiert sind [Taulli 2022]. Im genannten Beispiel trennen Leerzeichen die Wörter und der Punkt markiert das Satzende. Der eigentliche Tokenisierungsvorgang kann einfach darin bestehen, den Text an jedem Leerzeichen und jedem Satzzeichen zu teilen. Komplexere Tokenizer wenden jedoch zusätzliche Regeln an, um mit anspruchsvolleren sprachlichen Besonderheiten umzugehen [Haarmeyer 2021]. Beispielsweise könnte eine einfache Trennung die Worte in [„Qualitätsmanagement“, „verbessert“, „die“, „Ergebnisse“, „.“] aufspalten. Im Deutschen sind zusammengesetzte Substantive üblich, die bei bestimmten Tokenisierungsverfahren in einzelne Wörter zerlegt werden können, besonders wenn eine detaillierte Analyse der Wortstruktur notwendig ist. Ein Beispiel hierfür ist „Qualitätsmanagement“, das normalerweise als ein einziges Token behandelt wird, obwohl es aus zwei Bestandteilen besteht: „Qualität“ und „Management“. Dies zeigt die besondere Herausforderung bei der Tokenisierung der deutschen Sprache, wo die korrekte Trennung solcher Komposita entscheidend für die Genauigkeit der linguistischen Analyse sein kann. In einer weiterführenden Phase behandelt die Tokenisierung spezielle textuelle Elemente wie Kontraktionen, Daten, Zahlen und andere nicht-standardisierte Sprachformen [Chowdhary 2020]. So könnte „zum“ in [„zu“, „dem“] aufgeteilt oder "Dr. Müller" als ein einziger Token anstatt als [„Dr“, „.“, „Müller“] behandelt werden. Ebenso könnte das Datum „25.01.2024“ als ein einzelner Token behandelt werden, anstatt es in [„2024“, „.“, „04“, „.“, „25“] zu teilen. Das Ergebnis des Tokenisierungsprozesses ist eine Liste von Token, die in nachfolgenden NLP-Analysen verwendet wird. Diese Ausgabe muss klar und präzise sein, um sicherzustellen, dass nachgelagerte Prozesse wie die morphologische Analyse oder Syntaxanalyse die Daten

effektiv verstehen und verarbeiten können. Das finale Beispiel für eine Tokenisierung lautet: [„Qualitätsmanagement“, „verbessert“, „die“, „Ergebnisse“, „.“]. Jeder dieser Schritte erfordert sorgfältige Überlegungen, um sicherzustellen, dass die generierten Tokens für die beabsichtigten NLP-Aufgaben nützlich sind. Die gewählte Methode der Tokenisierung kann die Komplexität und Leistung der gesamten NLP-Pipeline erheblich beeinflussen [Haarmeier 2021]. Eine detaillierte und technische Untersuchung aller Tokenisierungsmethoden würde den Rahmen dieser Dissertation sprengen. Es ist jedoch erforderlich, an dieser Stelle die Methoden, die sich in der Fachwelt etabliert haben, bisschen näher zu betrachten. Dies ermöglicht ein tieferes Verständnis der bewährten Ansätze und deren Anwendung in der Praxis der natürlichen Sprachverarbeitung.

Der Whitespace Tokenizer ist die einfachste Form der Tokenisierung, die den Text anhand von Leerzeichen wie Zwischenräumen, Tabs und Zeilenumbrüchen aufteilt. Er berücksichtigt jedoch keine Satzzeichen oder linguistische Merkmale des Textes. Ein Beispiel hierfür ist der Satz „Projektmanagement ist entscheidend.“, bei dem das Wort „entscheidend.“ das Satzzeichen miteinschließt, was für viele NLP-Aufgaben nicht ideal sein könnte [Rai und Borah 2021]. Der regelbasierte Tokenizer nutzt eine Reihe vordefinierter Regeln, um Token genauer zu identifizieren. Diese Regeln können mit Satzzeichen, Kontraktionen und anderen komplexen linguistischen Merkmalen umgehen. Ein Beispiel hierfür ist der Text „Dr. Müller arbeitet in Berlin.“, wo „Dr.“ als separates Token von „Müller“ erkannt wird, was für die korrekte Handhabung von Namen angemessen ist. Der Regular Expression (RegExp) Tokenizer verwendet reguläre Ausdrücke, um die Kriterien für das Aufteilen von Text in Token festzulegen. Diese Methode ist sehr flexibel und kann angepasst werden, um Token basierend auf für die Sprache oder die Aufgabenstellung spezifischen Mustern zu erfassen. Ein Beispiel ist der Text „Das E-Mail-Update wurde um 3:45 PM gesendet.“, bei den Formaten für E-Mails, Zeitmuster und mehr als einzelne Token erkannt werden könnten [Magyar und Szenasi 2021]. Der Subword Tokenizer zerlegt Wörter in kleinere bedeutungstragende Einheiten (Subwörter oder Morpheme). Dies ist besonders nützlich für agglutinierende Sprachen wie Deutsch, in denen Zusammensetzungen häufig vorkommen. Ein Beispiel ist der Text „Unabhängigkeitserklärungen sind wichtig.“, bei den Wörtern in ihre Bestandteile wie "Unabhängig", "keits", "erklärt", "ungen" aufgeteilt werden [Balhar 2023]. Diese Methode kann beim Verstehen und Verarbeiten von zusammengesetzten Wörtern hilfreich sein. Der mehrsprachige Tokenizer ist darauf ausgelegt, Texte effektiv in mehreren Sprachen zu verarbeiten. Diese Tokenizer werden anhand großer Datensätze verschiedener Sprachen trainiert, was sie robust für die Handhabung sprachspezifischer Nuancen macht [Abdullah und Rusli 2021]. Jeder Typ von Tokenizer hat seine Stärken und Schwächen und wird basierend auf den spezifischen Anforderungen des Projekts und den Eigenschaften der Sprache ausgewählt. Für das Deutsche sind insbesondere Tokenizer, die mit zusammengesetzten Substantiven und der reichen Morphologie der Sprache umgehen können (wie Subword Tokenizer oder regelbasierte Tokenizer), besonders nützlich. Diese Tokenizer bieten die notwendige Flexibilität und Genauigkeit für eine umfassende linguistische Analyse und nachfolgende NLP-Aufgaben.

Die **morphologische Analyse** zielt darauf ab, die Struktur von Wörtern innerhalb einer Sprache zu sezieren und zu verstehen. Ihre Hauptziele umfassen das Verständnis der Wortformen und die Erleichterung der Lemmatisierung. Durch die Analyse der Morphologie der Wörter können NLP-Systeme nicht nur die Grundform eines Wortes, sondern auch dessen verschiedene Affixe identifizieren. Diese tiefgreifende Analyse trägt zum Verständnis bei, wie Wörter gebildet werden und wie sie zueinander in Beziehung stehen, was für Sprachen mit komplexen flektierenden Regeln wie Deutsch essenziell ist [Kreutzer 2019]. Die Ausführung der morphologischen Analyse umfasst mehrere Schlüsselkomponenten, die zusammenarbeiten, um die Struktur der Wörter zu zerlegen und zu interpretieren. Stemming reduziert ein Wort auf seine Grundform, indem bekannte Affixe entfernt werden, ohne Rücksicht auf den Wortteil oder Kontext des Wortes. Dieser Ansatz ist relativ einfach und kann manchmal zu ungenauen oder übergeneralisierten Ergebnissen führen [Taulli 2022]. Zum Beispiel könnte vom deutschen Wort „laufend“ das Stemming fälschlicherweise nur zu „lauf“ reduzieren, ohne zu unterscheiden, ob es sich um ein Nomen oder einen Teil eines Verbs handelt. Lemmatisierung ist eine fortgeschrittenere Technik, die eine detaillierte morphologische Analyse beinhaltet, um ein Wort korrekt zu seinem Lemma zurückzuführen. Sie berücksichtigt die grammatischen Merkmale des Wortes in seinem linguistischen Kontext, was sie genauer und kontextbewusster macht als Stemming. Zum Beispiel würde die Lemmatisierung für das deutsche Wort „gekauft“ nicht nur die Wurzel „kauf“ identifizieren, sondern auch erkennen, dass „gekauft“ das Partizip II des Verbs „kaufen“ ist und „kaufen“ als Lemma zurückgeben. Die kombinierte Verwendung von Stemming und Lemmatisierung in der morphologischen Analyse ermöglicht es NLP-Systemen, sowohl breite als auch nuancierte Aspekte der Sprachverarbeitung zu handhaben. Stemming reduziert schnell Wörter auf ihre Grundformen, was z.B. in Suchalgorithmen nützlich ist, wo exakte lexikalische Treue weniger kritisch ist [Taulli 2022]. Lemmatisierung bietet die Genauigkeit, die für Aufgaben erforderlich ist, die ein Verständnis der grammatischen und kontextuellen Nuancen des Textes erfordern, wie z.B. maschinelle Übersetzung, Inhaltsanalyse und anspruchsvolle Suchfunktionen. Zusammen adressieren diese Komponenten den Kernzweck der morphologischen Analyse, indem sie sicherstellen, dass Wörter nicht isoliert, sondern als Teil eines größeren linguistischen Systems interpretiert werden. Dies ist entscheidend für die genaue Verarbeitung und das Verständnis komplexer menschlicher Sprachen in computergestützten Einstellungen. Die morphologische Analyse durchläuft mehrere sequenzielle Schritte, die wie zuvor beschrieben darauf ausgelegt sind, die Struktur von Wörtern innerhalb eines gegebenen Textes zu dekonstruieren und zu verstehen. Der Prozess beginnt mit Token, die aus der Tokenisierung gewonnen wurden. Diese Tokens sind einzelne Wörter oder Satzzeichen, die aus dem umfangreicheren Textkörper extrahiert werden. Jedes Token wird auf seine morphologische Struktur hin analysiert. Dabei wird die Basis- oder Wurzelform des Wortes identifiziert, zusammen mit allen Affixen, die seine Bedeutung und grammatische Funktion modifizieren. Beispielsweise wird im Deutschen das Wort „gesprochen“ analysiert, um „sprech“ als Wurzel und „ge-“ sowie „-en“ als morphologische Marker, die das Partizip Perfekt anzeigen, zu identifizieren [Taulli 2022]. Das Endergebnis dieser Analyse ist ein

detailliertes Attributset für jedes Token, einschließlich seiner Wurzelform, Wortart, Zeit, Fall, Zahl und anderen grammatischen Merkmalen. Diese Attribute sind entscheidend für die nachfolgende syntaktische Analyse und semantische Analyse. Die Tabelle 15 visualisiert ein mögliches Ergebnis der morphologischen Analyse:

Tabelle 15: Attribute, die bei einer morphologischen Analyse entstehen [Eigenes Beispiel]

Token	Attribute				
	Wort	Wortart	Zeit / Anzahl	Wurzel / Artikel	Affixe
gesprochen	Verb	Perfekt	sprech	ge- / -en	
aufgeschrieben	Verb	Perfekt	schreib	auf- / -en	
Freundschaftsbeziehungen	Nomen	Plural	Feminin	-en	

Trotz der strukturierten und systematischen Vorgehensweise stößt die morphologische Analyse auf mehrere Herausforderungen, insbesondere bei Sprachen mit reichen und komplexen morphologischen Regeln. Sprachen wie Deutsch, die umfangreiche Komposition und Flexion aufweisen, stellen eine erhebliche Herausforderung dar, das korrekte Lemma und die grammatischen Attribute von Wörtern zu bestimmen [Kreutzer 2019]. Beispielsweise erfordert das Wort „Freundschaftsbeziehungen“ eine sorgfältige Analyse, um die Basisnomen und ihre Beziehungen korrekt zu identifizieren. Wörter können je nach Kontext mehrere Bedeutungen oder grammatische Verwendungen haben, was es schwierig macht, ihre korrekten Attribute ohne ein umfassendes Verständnis des umgebenden Textes zu bestimmen. So kann „Bank“ im Deutschen entweder eine finanzielle Institution oder eine physische Bank bedeuten, was kontextuelle Hinweise für eine korrekte Interpretation erfordert. Die Lemmatisierung hängt stark davon ab, die grammatischen Merkmale eines Wortes korrekt zu identifizieren, was in automatisierten Systemen fehleranfällig sein kann, insbesondere beim Umgang mit unregelmäßigen Formen oder weniger bekannten Wörtern. Unregelmäßige Verben oder Substantive im Deutschen, wie „ging“, weichen von den Standardkonjugations- und Deklinationsmustern ab und stellen Herausforderungen für die automatisierte morphologische Analyse dar [Muthee et al. 2022]. Zudem kann die morphologische Analyse rechenintensiv sein, insbesondere bei der Verarbeitung großer Textmengen oder der Anwendung komplexer Regeln für Sprachen mit reicher Morphologie. Die effiziente Handhabung der Rechenlast bei gleichbleibend hoher Genauigkeit bleibt eine bedeutende technische Herausforderung.

Nach der morphologischen Verarbeitung sind die Wörter bereits in Token zerlegt und attribuiert. Anschließend zielt die **syntaktische Analyse** darauf ab, die grammatischen Strukturen von Sätzen zu entschlüsseln. Dies ist aus mehreren Gründen entscheidend: Zum einen ermöglicht das Verständnis der grammatischen Beziehungen innerhalb eines Satzes eine tiefere Erfassung und Manipulation des Textes. Systeme können erkennen, welche Wörter als Subjekte, Verben oder Objekte fungieren, was für Aufgaben wie Informationsgewinnung, Inhaltsanalyse oder auch maschinelle Übersetzung unerlässlich ist [Taulli 2022]. Zudem konzentriert sich die syntaktische Analyse auf das „Dependency

Parsing“, das Beziehungen etabliert, in denen Wörter oder Phrasen (Abhängigkeiten) mit anderen Wörtern verknüpft sind, die diese modifizieren oder bestimmen (Köpfe, engl. head word). Diese Art des Parsings ist nachher für die semantische Analyse von Bedeutung, da sie die Beziehungen und Abhängigkeiten innerhalb eines Satzes klärt. Um diese Ziele zu erreichen, werden in der syntaktischen Analyse mehrere Komponenten und Werkzeuge eingesetzt: Grammatikregeln bilden das Fundament der Syntaxanalyse und liefern den Rahmen für das Parsing [Carstensen 2010]. Diese Regeln definieren, wie Wörter und Phrasen kombiniert werden können, um gültige Sätze zu bilden. In NLP werden diese Regeln oft durch formale Grammatiken wie kontextfreie Grammatiken (engl. Content Free Grammar, CFG) dargestellt, die hauptsächlich beim Constituency Parsing verwendet werden. Die Durchführung der syntaktischen Analyse hängt stark von Parsing-Algorithmen ab, die Grammatikregeln interpretieren und anwenden, um Parse-Bäume zu konstruieren. Zu den gängigen Ansätzen gehören Top-Down Parsing, das mit der höchsten Regel beginnt und sich zu den Wörtern vorarbeitet, Bottom-Up Parsing, welche bei den Wörtern beginnt und sich zum Satzniveau hocharbeitet, Chart Parsing, das effizient für mehrdeutige Texte ist, indem es Zwischenergebnisse speichert und bei Bedarf wiederverwendet. Die Komponenten der syntaktischen Analyse sind direkt darauf zugeschnitten, ihre Zwecke zu erfüllen. Die Grammatikregeln bieten das Gerüst, das definiert, was innerhalb der Syntax einer Sprache möglich ist, während Parsing-Algorithmen diese Regeln zum Leben erwecken, indem sie auf tatsächliche Textdaten angewendet werden und Strukturen aufbauen, die die durch die Regeln festgelegten grammatischen Beziehungen offenbaren. Abhängig vom Endziel – ob es darum geht, Wort-zu-Wort-Beziehungen zu verstehen oder Sätze in ihre strukturellen Komponenten zu zerlegen – werden unterschiedliche Parsing-Algorithmen und grammatischen Modelle verwendet [Kumar 2011].

Der Prozess der syntaktischen Analyse ist methodisch strukturiert und umfasst mehrere wichtige Schritte. Zunächst beginnt der Prozess mit Token, die bereits während der Tokenisierung und morphologischen Analyse identifiziert und annotiert wurden. Jedes Token ist mit seiner Wortart und anderen relevanten morphologischen Merkmalen gekennzeichnet. Abhängig vom Parsing-Ansatz – ob Dependency oder Constituency – konstruiert der Parser einen Baum, der die grammatische Struktur des Satzes darstellt. Beim Dependency Parsing wird ein Baum erstellt, der zeigt, wie jedes Wort (Knoten) mit anderen Wörtern verbunden ist und Abhängigkeiten basierend auf ihren grammatischen Rollen identifiziert. Beim Constituency Parsing wird ein Baum durch Gruppierung von Wörtern in Unterphrasen (Konstituenten) gebildet, die grammatisch zusammen funktionieren. Das Ergebnis ist ein detaillierter Parse-Baum, der die grammatische Struktur des Satzes verkapselt. Die Abbildung 23 visualisiert den Unterschied von den zwei Parsing Ansätzen. Dieser Baum ist entscheidend für nachfolgende Schritte in der NLP, wie die semantische Analyse, bei der das Verständnis der grammatischen Rollen und Beziehungen wesentlich ist [Kumar 2011].

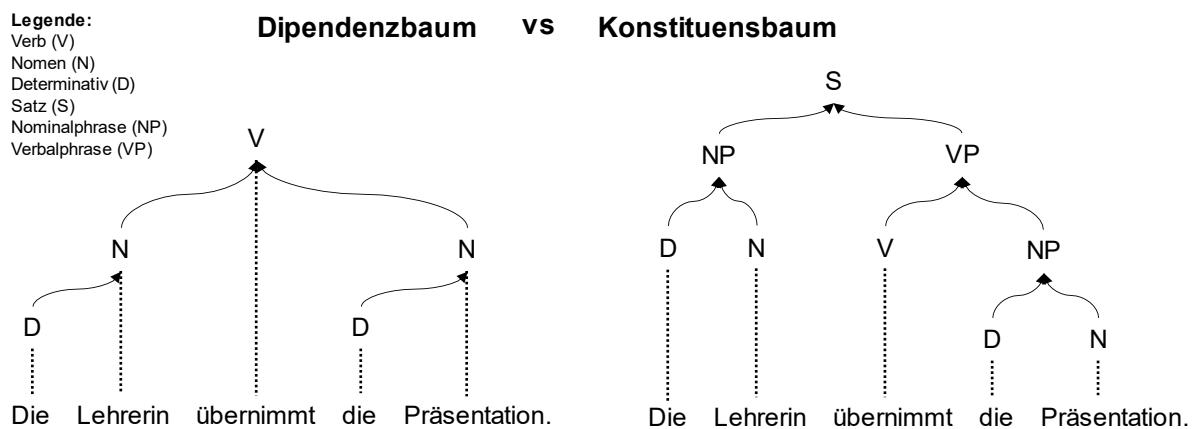


Abbildung 23: Dependency Parsing vs Constituency Parsing [Eigene Darstellung]

Obwohl der Prozess einfach erscheinen mag, komplizieren mehrere Herausforderungen die syntaktische Analyse. Strukturelle Mehrdeutigkeiten können auftreten, bei denen ein Satz in mehr als einer Struktur grammatisch korrekt sein könnte, was zu mehreren möglichen Parse-Bäumen führt. Die Auflösung, welche Struktur die beabsichtigte Bedeutung widerspiegelt, erfordert oft Kontext über den Satz hinaus. Zum Beispiel könnte die Phrase „alte Männer und Frauen“ entweder „alte Männer und [alte] Frauen“ oder „alte Männer und [alle] Frauen“ bedeuten, abhängig von der Gruppierung der Wörter [Carstensen 2010]. Unterschiedliche Sprachen zeigen einzigartige syntaktische Strukturen und Regeln, die in ihrer Komplexität stark variieren können. Beispielsweise stellt die flexible Satzstruktur des Deutschen, bei der das Verb in Nebensätzen ans Ende rücken kann, eine besondere Herausforderung für Parsing-Algorithmen dar. Auch Sprachen mit hoher Flexionsrate, bei denen sich Wortformen umfangreich ändern, um grammatischen Funktionen auszudrücken, erhöhen die Komplexität des Parsing-Prozesses. Parsing-Algorithmen, insbesondere solche, die Mehrdeutigkeiten und komplexe grammatischen Strukturen bewältigen, sind rechenintensiv. Die Sicherstellung, dass diese Algorithmen sowohl effizient als auch skalierbar sind, stellt eine erhebliche Herausforderung dar, besonders bei großen Textmengen. Außerdem ist natürliche Sprache nicht immer grammatisch perfekt. Sätze können Slang, Idiome oder umgangssprachliche Ausdrücke enthalten, die sich nicht sauber in formale grammatische Kategorien einordnen lassen. Die Entwicklung von Parsern, die robust genug sind, um solche Unregelmäßigkeiten zu bewältigen, ohne an Genauigkeit zu verlieren, ist eine große Herausforderung. Diese Herausforderungen sind untrennbar mit dem Prozess der syntaktischen Analyse verbunden [Kumar 2011]. Dies unterstreicht den Bedarf an fortgeschrittenen Algorithmen, die aus dem Kontext lernen und sich an die sprachlichen Nuancen verschiedener Sprachen anpassen können. Maschinelles Lernen, insbesondere Ansätze, die auf tiefem Lernen basieren, wird zunehmend eingesetzt, um die Genauigkeit und Effizienz von Syntax-Parsern zu verbessern und bietet vielversprechende Lösungen für diese langjährigen Herausforderungen. Diese kontinuierliche Verbesserung der Technologie der syntaktischen Analyse ist entscheidend für die Entwicklung von anspruchsvoller und menschenähnlicheren NLP-Systemen.

Darauf folgt die **semantische Analyse**, welche grundlegend darauf abzielt, die von Wörtern und Sätzen innerhalb eines spezifischen Kontexts vermittelte Bedeutung zu verstehen. Kernziele umfassen die Interpretation der Bedeutung, bei der es darum geht, die beabsichtigte Bedeutung von Wörtern, Phrasen und Sätzen so zu verstehen, wie es ein Mensch tun würde. Dies beinhaltet das Erfassen von Nuancen, das Auflösen von Mehrdeutigkeiten und das Verstehen der Beziehungen und Wechselwirkungen zwischen den Komponenten der Sprache. Ein weiteres zentrales Ziel ist die Disambiguierung, die für das Auflösen von Sprachambiguitäten unerlässlich ist [Kao 2007]. Die semantische Analyse klärt die Bedeutung von Wörtern mit mehreren Interpretationen und bestimmt die angemessensten Bedeutungen in gegebenen Kontexten. Um diese Ziele zu erreichen, setzt die semantische Analyse mehrere ausgefeilte Komponenten und Techniken ein: Die Wortsinndisambiguierung (Word Sense Disambiguation, WSD) ist entscheidend dafür, zu bestimmen, welche der möglichen Bedeutungen eines Wortes in einem gegebenen Kontext verwendet wird. Sie verwendet Algorithmen, die die umgebenden Wörter und externe Wissensquellen berücksichtigen, um die korrekte Bedeutung zu erschließen. Die semantische Rollenbeschriftung (Semantic Role Labeling, SRL) identifiziert die Rollen, die Wörter in einem Satz spielen, insbesondere die durch Verben etablierten Beziehungen, wie beispielsweise die Bestimmung, wer eine Aktion ausführt und an wen die Aktion gerichtet ist. Sie wendet Modelle an, die grammatischen Konstrukte ihren Rollen in Aktionen und Interaktionen wie Agent, Objekt und Instrument zuordnen [Navigli 2009; Palmer et al. 2005]. Die Erkennung und Verknüpfung von Entitäten (Entity Recognition and Linking) erkennt benannte Entitäten (Personen, Orte, Organisationen usw.) im Text und kategorisiert sie in vordefinierte Kategorien. Zudem werden diese Entitäten mit einer eindeutigen Kennung in einer Wissensdatenbank verknüpft. Hierbei werden Modelle eingesetzt, die auf großen Datensätzen trainiert sind, um Entitätstypen zu erkennen und zu klassifizieren, wobei oft Informationen aus semantischen Netzwerken oder Datenbanken integriert werden, um Kontext zu bieten und Entitäten zu disambiguieren [Kao 2007]. Die Komponenten der semantischen Analyse sind darauf abgestimmt, ihre übergeordneten Zwecke zu erfüllen: Von der Disambiguierung zur Klarheit trägt WSD dazu bei, die Bedeutung von Begriffen im spezifischen Kontext zu klären, und adressiert direkt den Zweck der Disambiguierung. Vom Verständnis der Rollen bis zur Interpretation der Bedeutung ermöglicht SRL, zu interpretieren, wie Aktionen innerhalb von Sätzen strukturiert sind, was in ein umfassenderes Verständnis der Bedeutung von Sätzen und Texten mündet. Von der Erkennung zur Wissensintegration kategorisieren und identifizieren Entitätserkennung und -verknüpfung nicht nur wichtige Informationen, sondern verbinden auch Textdaten mit dem breiteren Weltwissen, was die Tiefe der Bedeutungsinterpretation verstärkt. Der Prozess der semantischen Analyse beginnt mit dem strukturierten Output der Syntaxanalyse, typischerweise Parse-Bäumen, die die grammatischen Beziehungen innerhalb von Sätzen darlegen. Diese Parse-Bäume werden in semantische Darstellungen wie Prädikat-Argument-Strukturen transformiert. Dabei werden syntaktische Elemente ihren semantischen Rollen zugeordnet und mit spezifischen Aktionen oder

Entitäten verknüpft. Um das Verständnis zu verbessern und Kontext zu bieten, integriert die semantische Analyse oft externes Wissen aus Datenbanken, Ontologien oder semantischen Netzwerken. Dies hilft dabei, Mehrdeutigkeiten aufzulösen und die Interpretation des Textes mit Hintergrundwissen anzureichern. Der Output ist eine detaillierte Darstellung der Textbedeutung, einschließlich der Beziehungen zwischen Entitäten und Aktionen, angereichert durch kontextbewusste Einsichten [Kao 2007].

Obwohl der Prozess methodologisch robust ist, stößt die semantische Analyse auf mehrere bedeutenden Herausforderungen. Die Bedeutung von Wörtern und Sätzen kann sich drastisch je nach Kontext ändern. Diese Nuancen einzufangen, ist entscheidend für eine genaue Interpretation und kompliziert das Stadium des semantischen Parsings, da das System dynamisch auf Kontexthinweise reagieren muss, was oft fortgeschrittene Algorithmen erfordert, die tiefes Lernen und Kontexterkenntnis ermöglichen. Idiome, Metaphern und Sarkasmus beinhalten Bedeutungen, die nicht direkt aus den konstituierenden Wörtern abgeleitet werden. Diese Elemente der bildlichen Sprache sind allgegenwärtig in der menschlichen Kommunikation und stellen erhebliche Hürden im semantischen Parsing dar. Die Handhabung bildlicher Sprache erfordert die Integration von breitem Kontext- und Weltwissen sowie fortgeschrittene Inferenzfähigkeiten in der Phase des semantischen Parsings, um nicht-wörtliche Bedeutungen korrekt zu interpretieren. Wörter mit mehreren Bedeutungen (Polysemie) oder verschiedene Bedeutungen, die die gleiche Schreibweise teilen (Homonymie), können die semantische Analyse ohne ausreichenden Kontext besonders schwierig machen. Das Auflösen dieser Mehrdeutigkeiten erfordert ausgeklügelte Techniken zur Wortsinnambiguierung während des semantischen Parsings, oft unter Nutzung umfangreicher Kontextdaten und externer Wissensdatenbanken [Jurafsky und Martin 2009]. Semantische Analyse, insbesondere wenn umfangreiches externes Wissen integriert und große Datensätze verarbeitet werden, ist rechenintensiv. Es ist entscheidend, dass die Schritte des semantischen Parsings und der Wissensintegration effizient und skalierbar sind, was optimierte Algorithmen und möglicherweise den Einsatz von parallelen Verarbeitungstechniken erfordert. Die detaillierten Schritte des semantischen Analyseprozesses werden direkt von diesen Herausforderungen beeinflusst und oft erschwert. Beispielsweise bereichert die Integration von externem Wissen nicht nur die semantische Darstellung, sondern hilft auch, die Herausforderungen der kontextuellen Varianz und Polysemie zu mildern. Jeder Schritt, von der Eingabeverarbeitung bis zum finalen Output, muss so gestaltet sein, dass er diese Herausforderungen antizipiert und adressiert, um sicherzustellen, dass die semantische Analyse nicht nur den wörtlichen Text entschlüsselt, sondern auch die Feinheiten und impliziten Bedeutungen erfassst, die durch die Sprache vermittelt werden [Schlechtweg et al. 2020].

Der letzte Hauptschritt, die **pragmatische Analyse**, zieht kontextuelle Informationen hinzu, um die aus der semantischen Analyse gewonnenen Wortbedeutungen zu ergänzen. Die Kernziele umfassen das Verständnis der kontextuellen Relevanz, also wie Sprache spezifisch in einer gegebenen Situation verwendet wird, um Nachrichten effektiv zu übermitteln, bestimmte Effekte zu erzielen oder spezifische Handlungen anzustoßen. Ein weiteres Ziel ist das Erkennen der Intentionen des Schreibers, was

oft das Lesen zwischen den Zeilen oder das Erkennen von Impliziertem, das nicht explizit ausgesagt ist, beinhaltet. Zudem wird die Analyse der interpersonalen Beziehungen in der Kommunikation berücksichtigt, um Elemente wie Höflichkeit, Machtverhältnisse und emotionale Untertöne zu verstehen, die beeinflussen, wie Botschaften übermittelt und empfangen werden [Kao 2007]. Die Analyse von Sprechakten identifiziert die Handlungen, die durch Sprechen ausgeführt werden, wie Behaupten, Fragen, Anfordern, Befehlen oder Versprechen. Die Methode hierbei umfasst die Analyse der Verben und der Satzstruktur, um die Art des Sprechakts zu klassifizieren und seine Funktion innerhalb der Kommunikation zu verstehen. Die Kontextmodellierung konstruiert ein Modell des physischen, sozialen und kulturellen Kontexts, in dem die Kommunikation stattfindet. Dies schließt den Rahmen, die Beziehung zwischen den Kommunikatoren und die breiteren gesellschaftlichen Normen ein, die die Sprachnutzung beeinflussen. Hierfür werden Hintergrundwissen und situative Daten genutzt, um den Kontext genau zu interpretieren, oft gestützt auf Datenbanken oder Wissensbasen, die Informationen über kulturelle Normen und gängige Praktiken speichern. Kohärenz und Kohäsion gewährleisten, dass der Diskurs innerhalb seines Kontexts Sinn ergibt und einen logischen Fluss sowie eine Verbindung zwischen den Ideen oder Teilen des Gesprächs aufrechterhält. Die Methode analysiert den Einsatz von sprachlichen Hinweisen wie Pronomen, Konjunktionen und Diskursmarkern, die helfen, Sätze und Ideen miteinander zu verknüpfen und sicherzustellen, dass der Text kohärent und kontextuell angemessen ist. Zum Beispiel bei dem Satz: „Kannst du das Fenster schließen?“ kann anhand des Kontexts mit Hilfe der pragmatischen Analyse festgestellt werden, ob es sich um eine echte Frage nach der Fähigkeit des Zuhörers handelt, das Fenster zu schließen, oder um eine höfliche Aufforderung an den Zuhörer, das Fenster zu schließen. Bei der pragmatischen Analyse geht es nicht nur darum, die wörtliche Frage zu verstehen, sondern auch den sozialen Kontext, den Tonfall und die Umgebung, in der sie gestellt wurde, zu interpretieren, um die wahre Absicht hinter den Worten zu erkennen [Jurafsky und Martin 2009].

Der Prozess beginnt mit den Ergebnissen der semantischen Analyse, die die Bedeutungen von Sätzen und die Beziehungen zwischen verschiedenen Textteilen umfassen. Im nächsten Schritt werden umfangreiche Kontextinformationen einbezogen, darunter die physische Umgebung, der kulturelle Hintergrund, die sozialen Rollen und Beziehungen der Teilnehmenden sowie der breitere situative Kontext. Anschließend erfolgt die Interpretation der Absichten des Schreibers und der sozialen Bedeutung. Hierbei wird analysiert, wie der Kontext und die Verwendung der Sprache zusätzliche, nicht explizit ausgesprochene Bedeutungen vermitteln. Dies umfasst die Identifikation von Sprechakten, das Verständnis indirekter Anfragen, das Erkennen von Ironie oder Sarkasmus und das Erkennen kultureller Nuancen. Das Endergebnis ist eine bereicherte Interpretation des Textes, die nicht nur seinen expliziten Inhalt, sondern auch seine implizierten Bedeutungen und die Intentionen des Sprechers widerspiegelt [Taulli 2022]. Die Durchführung dieser Prozessschritte stößt auf mehrere bedeutende Herausforderungen. Eine davon ist die Mehrdeutigkeit der Intention. Es ist herausfordernd,

die wahren Intentionen hinter den Worten zu bestimmen, besonders wenn diese mehrere Implikationen tragen können oder auf eine nicht direkte Weise geäußert werden. Dies erfordert Folgerungsmechanismen während der Phase der Interpretation der Sprecherabsicht und der sozialen Bedeutung, was die Analyse kompliziert und das Potenzial für Fehlinterpretationen erhöht. Ein weiteres Problem stellt die kulturelle und kontextuelle Variabilität dar. Unterschiede in kulturellen Normen und kontextuellen Einstellungen können zu verschiedenen Interpretationen derselben Ausdrücke führen, was die Standardisierung der pragmatischen Analyse erschwert. Eine effektive Kontextmodellierung muss eine breite Palette von kulturellen und situativen Kontexten berücksichtigen, was angesichts der Variabilität und Tiefe der erforderlichen Kontextdaten eine Herausforderung darstellt. Zudem spielen subtile sprachliche Hinweise wie Tonfall, Betonung und nonverbale Signale eine entscheidende Rolle in der Kommunikation, sind jedoch oft schwer in einer textbasierten Analyse zu erfassen [Sugimoto et al. 2004]. Die Sicherstellung von Kohärenz und Kohäsion in der Analyse erfordert fortschrittliche Algorithmen, die in der Lage sind, solche Feinheiten aus textuellen Hinweisen abzuleiten, was nicht immer machbar ist. Schließlich ist die pragmatische Analyse ressourcenintensiv und erfordert umfangreiche Datenverarbeitung und ausgefeilte Algorithmen, um die reale Variabilität in der Kommunikation zu bewältigen. Das Skalieren des Prozesses, um große Datenmengen zu bewältigen und gleichzeitig Genauigkeit und Geschwindigkeit zu erhalten, stellt eine erhebliche technologische Herausforderung dar.

Nachfolgend, als Zusammenfassung wird ein simples Beispiel durch die ganze Pipeline durchlaufen, um die Anwendung von NLP-Pipeline, welche in Abbildung 22 visualisiert wurde, zu verdeutlichen. In diesem Beispiel wird, beginnend mit der Zerlegung des Satzes in seine Grundbestandteile (Token) und der Analyse der grammatischen und morphologischen Eigenschaften jedes Wortes, einen grammatischen Rahmen (Syntax) konstruiert. Anschließend wird die wörtliche Bedeutung (Semantik) abgeleitet und die breiteren kommunikativen Absichten und den sozialen Kontext (Pragmatik) erkundet.

Tabelle 16: Verdeutlichung der NLP-Pipeline anhand eines einfachen Beispiels

	Prozess	Input	Output
Tokenisierung	Zerlegung des Satzes in einzelnen Token (Wörter und Satzzeichen).	Kannst du das Fenster schließen?	[„Kannst“, „du“, „das“, „Fenster“, „schließen“, „?“]
Morphologische Analyse	Analyse jedes Tokens auf seine Grundform, Wortart und andere grammatischen Merkmale.	[„Kannst“, „du“, „das“, „Fenster“, „schließen“, „?“]	<ul style="list-style-type: none"> „Kannst“ - Verb, 2. Person Singular, Präsens, Indikativ, Modalverb „du“ - Pronomen, 2. Person Singular, Nominativ „das“ - Artikel, geschlechtsneutral, Nominativ/Akkusativ „Fenster“ - Substantiv, geschlechtsneutral, Akkusativ, Singular „schließen“ - Verb, Infinitiv „?“ - Interpunktionszeichen
Syntaktische Analyse	Aufbau eines Parse-Baums zur Darstellung grammatischer Beziehungen und der Satzstruktur.	<ul style="list-style-type: none"> Morphologische Ausgabe, strukturiert als annotierte Token Aufbau der Sprachestruktur. 	<p>Parse tree structure:</p> <ul style="list-style-type: none"> • Sentence (S) • Verb Phrase (VP) [„Kannst“] • Subject (NP) [„du“] • Object (NP) [„das Fenster“] • Infinitive Verb [„schließen“] • Punctuation [„?“]
Semantische Analyse	Interpretation der Satzbedeutung anhand des Parse-Baums unter Berücksichtigung der Wortbedeutungen und der grammatischen Struktur.	<ul style="list-style-type: none"> Parse tree structure and lexical meanings Aufbau der Sprachestruktur. 	<p>Die Aktion „schließen“ wird von „du“ an „das Fenster“ verlangt. Das Modalverb "kannst" deutet eher auf eine höfliche Bitte als auf einen Befehl hin, d. h. der Sprecher fragt, ob der Hörer das Fenster schließen kann, und erwartet wahrscheinlich, dass die Handlung ausgeführt wird.</p>
Pragmatische Analyse	Verständnis der impliziten Bedeutungen, des Gesprächskontextes und der sozialen Dynamik zwischen Sprecher und Zuhörer.	<ul style="list-style-type: none"> Semantische Interpretation und möglicher Gesprächskontext. Ergänzende kontextuelle Informationen und Ontologien. 	<p>Es handelt sich um eine höfliche Aufrichterforderung in einem zwanglosen oder vertrauten Umfeld, wie aus der Verwendung des Modalverbs und der direkten Anrede "du" hervorgeht. Das Frageformat und das Modalverb implizieren, dass der Sprecher dem Zuhörer die Möglichkeit gibt, abzulehnen, was eine Höflichkeitsstrategie in der deutschen kulturellen Kommunikation ist.</p>

Dieser geschichtete Ansatz zeigt, wie jeder Schritt auf dem vorherigen aufbaut und ein umfassendes Verständnis sowohl der Struktur als auch der nuancierten Bedeutung des Satzes im situativen Kontext liefert (s. Tabelle 16).

Um die Implementierung dieser NLP-Schritte effizient und präzise durchzuführen, ist der Einsatz spezialisierter, vormodellierter Algorithmen unerlässlich [Qiu et al. 2020]. Die Entwicklung eines eigenen NLP-Algorithmus und dessen Training auf die deutsche Sprache liegt nicht im Fokus dieser Forschung. Stattdessen werden etablierte NLP-Ansätze und deren Werkzeugkasten wie spaCy, NLTK (engl. Natural Language Toolkit), Stanford NLP und BERT (engl. Bidirectional Encoder Representations from Transformers), die bereits für die deutsche Sprache optimiert wurden, verwendet und anschließend mit unternehmensspezifischen Informationen weiter trainiert. Diese Tools repräsentieren unterschiedliche Ansätze in der Verarbeitung natürlicher Sprache, wobei jedes seine spezifischen Stärken und Anwendungskontexte aufweist. Im nachfolgenden Abschnitt erfolgt ein detaillierter Vergleich dieser Algorithmen, um ihre Eignung für die verschiedenen Phasen der NLP-Pipeline zu bewerten und die optimale Auswahl für spezifische Anforderungen zu erleichtern.

SpaCy ist für seine hohe Leistung bekannt und zählt zu den schnellsten verfügbaren NLP-Bibliotheken, was es besonders geeignet für Anwendungen mit hohem Datenaufkommen und in Echtzeit macht. Es bietet eine Vielzahl an vortrainierten Modellen, die sowohl für ihre Genauigkeit als auch für ihre breite Sprachabdeckung optimiert sind. Die API von SpaCy ist benutzerfreundlich und einfach in der Anwendung, zudem ist sie gut dokumentiert, was eine flache Lernkurve ermöglicht. SpaCy umfasst umfangreiche eingebaute Funktionen für tiefgehende linguistische Analysen, wie Tokenisierung, Erkennung benannter Entitäten, Part-of-Speech-Tagging und Abhängigkeitsanalyse. Nutzer können zudem eigene Komponenten hinzufügen und die Funktionalitäten erweitern, was die Anpassung an spezifische Bedürfnisse ermöglicht. Allerdings sind SpaCys Modelle trotz ihrer Effizienz ressourcenintensiv, was in ressourcenbeschränkten Umgebungen ein Nachteil sein kann. SpaCy ist eher für den produktiven Einsatz als für die akademische Forschung gedacht, wo eventuell mehr experimentelle Flexibilität gefordert ist. Außerdem ist die Unterstützung für weniger verbreitete Sprachen im Vergleich zum Englischen oder Deutschen begrenzt, sowohl in Bezug auf Leistung als auch auf verfügbare Funktionen [SpaCy 2024].

In der Handhabung der zuvor beschriebenen fünf NLP-Schritte zeigt SpaCy besondere Stärken. Bei der Tokenisierung kann SpaCy Text in Wörter, Satzzeichen und andere Elemente unterteilen und unterschiedliche Schriftsysteme sowie komplexe Tokenaufteilungen, die Kontraktionen oder spezielle Satzzeichen betreffen, verarbeiten. Die morphologische Analyse bietet detaillierte Informationen, indem Lemma und Part-of-Speech-Tags für jedes Token identifiziert werden, wobei statistische Modelle verwendet werden, um die genauen morphologischen Tags basierend auf dem Kontext jedes Tokens vorherzusagen. Die syntaktische Analyse ist eine der Kernkompetenzen von SpaCy. Mithilfe eines Abhängigkeitsparsers wird die grammatischen Strukturen eines Satzes bestimmt, indem Beziehungen

zwischen „Kopf“-Wörtern und den Wörtern, die diese Köpfe modifizieren, identifiziert werden, was für das Verständnis der Struktur und Bedeutung von Sätzen entscheidend ist. Die Fähigkeiten in der tiefen semantischen Analyse, wie das Erkennen abstrakter oder impliziter Bedeutungen, sind jedoch im Vergleich zu kontextbewussteren Modellen wie BERT begrenzt [Choi et al. 2015]. In der pragmatischen Analyse bietet SpaCy die notwendigen Werkzeuge für ein gewisses Maß an pragmatischer Analyse durch seine Entitätserkennung und kontextspezifischen Modelle, analysiert jedoch nicht inhärent die Pragmatik oder die beabsichtigte Wirkung einer Nachricht im situativen Kontext. Für solche Anwendungen könnten zusätzliche Schichten oder benutzerdefinierte Modelle erforderlich sein. Zusammenfassend ist SpaCy äußerst wirksam für Anwendungen, die eine robuste linguistische Analyse mit einem Schwerpunkt auf Leistung und Praktikabilität in industriellen Umgebungen erfordern. Für Aufgaben, die ein tieferes Verständnis von Semantik oder Pragmatik erfordern oder für akademische Forschung, bei der Flexibilität entscheidend ist, könnten andere Werkzeuge geeigneter sein. Insgesamt macht das Design von SpaCy es zu einer ausgezeichneten Wahl für Entwickler, die NLP-Funktionen direkt in Produktionssysteme integrieren möchten.

NLTK ist bekannt für seine umfangreiche Sammlung von Bibliotheken und Ressourcen, die Unterstützung für viele NLP-Analysen bieten. Es wird häufig zu Bildungszwecken und für Experimente verwendet. Dank ausführlicher Dokumentation und einer unterstützenden Gemeinschaft ist NLTK ein hervorragendes Werkzeug für NLP-Anfänger. Obwohl es hauptsächlich auf Englisch ausgerichtet ist, bietet NLTK-Werkzeuge zur Verarbeitung mehrerer anderer Sprachen, allerdings mit unterschiedlichem Unterstützungsgrad und Vollständigkeit. Es umfasst Toolkits für Klassifizierung, Tokenisierung, Stemming, Tagging, Parsing und semantisches Räsonieren, was es sehr vielseitig macht. NLTK ist besonders flexibel, was die Anpassung und Experimentieren betrifft, und bietet daher besondere Vorteile für akademische und Forschungsumgebungen. Im Vergleich zu Bibliotheken wie SpaCy kann NLTK langsamer und weniger effizient in Bezug auf Speicher und Verarbeitungsgeschwindigkeit sein, was es weniger geeignet für Anwendungen mit hohem Datenvolumen oder Echtzeitanwendungen macht. Einige Module von NLTK können komplex in der Implementierung sein und erfordern eine steile Lernkurve für fortgeschrittenere Funktionen. Aufgrund seines bildungsorientierten Designschwerpunkts ist NLTK weniger auf produktionsbereite Effizienz und mehr auf Lehre und Forschung ausgerichtet, was seine Nützlichkeit in industriellen Anwendungen einschränken kann [Bird et al. 2009].

In Bezug auf die fünf NLP-Schritte bietet NLTK mehrere Tokenizer, einschließlich solcher für Sätze und Wörter. Es ermöglicht die Anpassung von Tokenisierungsprozessen, was für die Handhabung von Texten mit komplexen Strukturen oder nicht standardmäßigen Satzzeichen nützlich ist. NLTK umfasst mehrere Stemmer und Lemmatisierer für Englisch und andere Sprachen, die für die morphologische Analyse verwendet werden, um Wörter auf ihre Basis- oder Grundform zu reduzieren, was

für viele NLP-Aufgaben entscheidend ist. NLTK bietet eine Vielzahl von Parsern, von einfachen Parsern basierend auf regulären Ausdrücken (RegExp) bis hin zu komplexeren rekursiven Abstiegs- und Chart-Parsern, die es den Benutzern ermöglichen, den richtigen Komplexitätsgrad für ihre Syntaxanalysebedürfnisse zu wählen. Durch seine Integration mit WordNet ist NLTK stark in bestimmten Arten der semantischen Analyse, wie der Suche nach Synonymen, Antonymen und Wortdefinitionen. Es fehlt jedoch das tiefe semantische Kontextverständnis, das modernere Werkzeuge wie BERT bieten [Choi et al. 2015]. Obwohl NLTK keine spezifischen Werkzeuge für die pragmatische Analyse bietet, können seine umfangreichen Ressourcen genutzt werden, um benutzerdefinierte Lösungen für Aufgaben zu entwickeln, die das Verständnis der Verwendung von Sprache in menschlichen Kommunikationskontexten erfordern, wie zum Beispiel die Erkennung von Ironie oder Sarkasmus. Zusammenfassend ist NLTK ein ausgezeichnetes Werkzeug für die akademische Forschung und Bildung im Bereich der natürlichen Sprachverarbeitung. Es bietet umfangreiche Ressourcen und Fähigkeiten, die für eine breite Palette von NLP-Aufgaben angepasst werden können. Für Produktionsumgebungen, in denen Leistung und Skalierbarkeit entscheidend sind, könnten jedoch andere Werkzeuge geeigneter sein. NLTKs Flexibilität und Bildungsfokus machen es zu einem Grundpfeiler in der NLP-Gemeinschaft und besonders nützlich für Anfänger.

Stanford NLP ist ein umfassendes Paket, die eine breite Palette von linguistischen Analysewerkzeugen im Rahmen der CoreNLP-Suite bietet und somit eine der vollständigsten NLP-Bibliotheken darstellt. Sie ist bekannt für ihre präzise Genauigkeit, insbesondere bei Aufgaben wie Parsing, Erkennung benannter Entitäten und Part-of-Speech-Tagging. Die Unterstützung mehrerer Sprachen macht sie in verschiedenen sprachlichen Umgebungen und für globale Anwendungen einsetzbar. Stanford NLP ist so konzipiert, dass es in größere Systeme integriert werden kann, was die effiziente Verarbeitung großer Datensätze ermöglicht. Es wird kontinuierlich mit den neuesten Forschungsergebnissen aktualisiert und integriert die neuesten NLP-Technologien, die an der Stanford University entwickelt wurden. Die Einrichtung und Nutzung von Stanford NLP kann jedoch im Vergleich zu anderen NLP-Bibliotheken komplexer sein, was besonders für Anfänger eine Herausforderung darstellen könnte. Ähnlich wie SpaCy erfordert auch Stanford NLP erhebliche Rechenressourcen, insbesondere beim Einsatz der vollständigen Werkzeugpakete. Aufgrund seiner umfassenden Natur und seines Schwerpunkts auf Genauigkeit ist es möglicherweise weniger geeignet für leichtgewichtige oder weniger komplexe Anwendungen, bei denen eine schnelle Bereitstellung notwendig ist [Manning et al. 2014].

In Bezug auf die fünf NLP-Schritte bewältigt Stanford NLP die Tokenisierung effektiv, indem es Text präzise in Wörter, Phrasen oder andere sinnvolle Elemente unterteilt, was die Grundlage für die weitere Verarbeitung bildet. Die Suite enthält Werkzeuge wie den Stanford Tagger, die detaillierten morphologischen Informationen zu Wörtern liefert, einschließlich der Wortarten und anderer grammatischer Details. Eines der Kernstärken von Stanford NLP sind seine hochpräzisen Parsing-Fähigkeiten,

die sowohl Konstituens- als auch Dependenzparser verwenden, um die Struktur von Sätzen zu analysieren und zu verstehen [Choi et al. 2015]. Stanford NLP umfasst Werkzeuge für die semantische Analyse, wie die Erkennung benannter Entitäten, Koreferenzauflösung und Beziehungsextraktion, die helfen, die Bedeutungen und Beziehungen im Text zu verstehen. Obwohl nicht der primäre Fokus, unterstützt die Suite die pragmatische Analyse indirekt durch Werkzeuge wie die Koreferenzauflösung, indem sie diskursbasierte Strukturen und Bedeutungen jenseits einzelner Sätze versteht. Zusammenfassend ist Stanford NLP ideal für Anwendungen, die eine detaillierte linguistische und semantische Analyse in mehreren Sprachen erfordern. Die Suite wird sowohl in akademischen als auch in professionellen Kreisen für ihre Genauigkeit und Merkmalsbreite geschätzt. Ihre Komplexität und Ressourcenanforderungen machen sie jedoch weniger ideal für leichtgewichtige Anwendungen oder für Nutzer, die eine schnelle Bereitstellung ohne umfangreiche Rechenressourcen suchen. Ihre Fähigkeiten machen sie zu einer ausgezeichneten Wahl für Forscher und Entwickler, die detaillierte und genaue NLP-Verarbeitung in ihren Anwendungen benötigen.

BERTs Hauptstärke liegt in seiner Fähigkeit, den Kontext eines Wortes innerhalb eines Satzes zu verstehen, was frühere Modelle nicht erreichen konnten. Dies ist auf sein bidirektionales Training zurückzuführen, das den Text sowohl von links nach rechts als auch von rechts nach links betrachtet. Bei seiner Einführung setzte BERT neue Maßstäbe für eine Vielzahl von NLP-Aufgaben, einschließlich Fragebeantwortung und Sprachinferenz, und demonstrierte eine überlegene Leistung gegenüber bestehenden Modellen. BERT kann mit zusätzlichen Ausgabeschichten für eine breite Palette von Aufgaben, die über die ursprünglich trainierten hinausgehen, feinabgestimmt werden, was es sehr vielseitig macht. Es ist darauf ausgelegt, auf einem großen Textkorpus vorzutrainieren und dann auf spezifische Aufgaben feinabgestimmt zu werden, was im Vergleich zum Training eines Modells von Grund auf erhebliche Rechenressourcen spart. Allerdings erfordert BERT erhebliche Rechenleistung sowohl für das Training als auch für die Inferenz, insbesondere beim Umgang mit den vollständigen Modellversionen wie BERT-Large. Trotz seiner beeindruckenden Leistung ist seine zugrundeliegende Architektur komplex, was das Verständnis und die Modifikation erschweren kann. Aufgrund seiner Größe und Komplexität kann BERT bei der Inferenz langsamer sein als stromlinienförmigere Modelle, was eine Herausforderung für Anwendungen darstellen kann, die eine Echtzeitverarbeitung benötigen [Qiu et al. 2020].

In Bezug auf die fünf NLP-Schritte verwendet BERT die WordPiece-Tokenisierung, die Untereinheiten wie Suffixe und Präfixe effektiv handhabt, wodurch es in der Lage ist, Wörter außerhalb des Vokabulars eleganter zu verwalten als viele traditionelle Modelle. BERT führt keine explizite morphologische Analyse durch. Es erfasst jedoch einige morphologische Merkmale durch seine tiefen kontextuellen Einbettungen, da es Repräsentationen lernt, die implizit das Verständnis von Wortformen basierend auf deren Verwendung im Kontext einschließen. Obwohl BERT nicht speziell für die Syntaxanalyse entwickelt wurde, erfasst es syntaktische Muster durch seine Aufmerksamkeitsmechanismen,

die ihm helfen, Beziehungen zwischen Wörtern in einem Satz zu verstehen. Es ersetzt jedoch keine dedizierten syntaktischen Parser [Choi et al. 2015]. BERT zeichnet sich bei der semantischen Analyse aus. Seine Fähigkeit, den Kontext und die Bedeutung ganzer Textsequenzen zu verstehen, macht es besonders wirksam für Aufgaben wie Textklassifizierung, Sentimentanalyse und Entitätserkennung. Seine Einbettungen bieten ein nuanciertes Verständnis von Wortbedeutungen in verschiedenen Kontexten. BERT's tiefes Verständnis von Kontext ermöglicht es ihm, in einigen pragmatischen Aufgaben gut zu performen, wie Ironiedetektion oder Intent-Erkennung, insbesondere wenn es auf spezifische Datensätze zugeschnitten ist, die auf diese Aufgaben abgestimmt sind. Zusammenfassend stellt BERT einen bedeutenden Fortschritt in der NLP-Technologie dar, insbesondere bei Aufgaben, die ein tiefes Verständnis des Sprachkontexts erfordern. Es ist ideal für komplexe NLP-Aufgaben, die von einem Verständnis des Zusammenspiels zwischen Elementen in Texten über ein Dokument oder Gespräch hinweg profitieren. Allerdings machen seine Komplexität und Ressourcenanforderungen es schwierig, in Umgebungen einzusetzen, in denen Rechenressourcen begrenzt sind oder schnelle Reaktionszeiten erforderlich sind. Für Forschung und Hochleistungsanwendungen, bei denen ein tiefes semantisches Verständnis entscheidend ist, ist BERT ein leistungsfähiges Werkzeug, das hohe Maßstäbe für zukünftige Entwicklungen in der NLP setzt [Qiu et al. 2020]. Die Vergleichsmatrix in Tabelle 17 zeigt, wie gut die einzelne NLP-Ansätze die jeweilige NLP-Schritte erfüllen können.

Tabelle 17: Übersicht der Fähigkeiten von NLP-Ansätze zur Behandlung verschiedener NLP-Aspekte

Ansätze \ Schritte	Tokenisierung	Morphologische Analyse	Syntaktische Analyse	Semantische Analyse	Pragmatische Analyse
SpaCy	●	●	●	●	○
NLTK	●	○	○	○	○
Stanford NLP	●	●	●	●	○
BERT	●	○	○	●	○

Legende	● = Volle Punktzahl	○ = Halbe Punktzahl	○ = keine Punktzahl
----------------	---------------------	---------------------	---------------------

In der Bewertung verschiedener NLP-Ansätze erzielen alle Tools bei der Tokenisierung die volle Punktzahl. Für die morphologische Analyse erhalten SpaCy und Stanford NLP jeweils die volle Punktzahl für ihre umfassenden Werkzeuge, während NLTK aufgrund seiner grundlegenden Werkzeuge die halbe Punktzahl und BERT aufgrund des Fehlens expliziter morphologischer Analyse keine Punkte bekommt. In der Syntaxanalyse zeichnen sich SpaCy und Stanford NLP mit voller Punktzahl aus, NLTK und BERT erhalten die halbe Punktzahl. Bei der semantischen Analyse punkten SpaCy, Stanford NLP, und BERT voll, wohingegen NLTK die halbe Punktzahl erhält. In der pragmatischen Analyse erzielen SpaCy und Stanford NLP die halbe Punktzahl, da sie nicht speziell für diese ausgelegt

sind. NLTK bekommt keine Punkte, und BERT, trotz seiner Anpassungsfähigkeit an pragmatische Aufgaben, ebenfalls eine halbe Punktzahl. Insgesamt eignen sich SpaCy und Stanford NLP aufgrund ihrer identischen Punktzahlen besser für den Einsatz in dieser Forschung. Da SpaCy jedoch die deutsche Sprache besser beherrscht, wird die Entwicklung mit diesem Ansatz fortgesetzt.

5.2.2 eDeCoDe als Ontologie

In der Verarbeitung natürlicher Sprache (NLP) wird der Eingabetext, typischerweise verfasst von einem Menschen, zunächst morphologisch analysiert. Dabei werden die Wörter nach Wortarten und -formen, wie Nomen, Verben und Adjektive, klassifiziert und auf ihre Grundform reduziert. Dies schließt Aspekte wie Kasus, Person, Numerus und Tempus ein. Eine darauffolgende syntaktische Analyse strukturiert die Sätze gemäß grammatischen Regeln. In der semantischen Analyse betrachtet der Algorithmus die Bedeutung der einzelnen Wörter. Durch eine ergänzende pragmatische Analyse werden kontextuelle Informationen einbezogen, um die Wortbedeutungen zu vervollständigen. Diese Phase schafft Verknüpfungen zu vorhergehenden oder folgenden Sätzen, wodurch frühere Aussagen durch zusätzliche Informationen bereichert werden. Ziel ist es, eine Interpretation zu gewährleisten, die der intendierten Bedeutung entspricht [Kumar 2011]. Dies setzt ein spezielles Wissen voraus, das über allgemeine sprachliche Kenntnisse hinausgeht. Deshalb ist es von entscheidender Bedeutung, dass zusätzlich zur in S1 entwickelten Informationsstruktur sowie den linguistischen Zusammenhängen, die der NLP-Algorithmus bereits erfasst hat, eine fachspezifische Ontologie entwickelt wird. Diese Ontologie soll dazu beitragen, die logischen und fachspezifischen Beziehungen der Informationen zu definieren, die aus den unterschiedlichen Dokumenten extrahiert und analysiert werden. Dies ist insbesondere in der pragmatischen Analyse erforderlich, um Diskurszusammenhänge präzise zu erkennen und den Kontext umfassend zu verstehen. Hierdurch wird der Algorithmus befähigt, Textelementen passende Kategorien zuzuweisen, die Beziehungen zwischen verschiedenen Kategorien zu identifizieren und auf dieser fundierten Grundlage später Entscheidungen zu treffen.

Im Rahmen des in dieser Dissertation betrachteten Problemfeldes wird eine fachspezifische Ontologie durch eine Kombination aus eDeCoDe-Logik und Automotive SPICE entwickelt [Schlüter 2023; ASPICE 2023]. Einerseits kommen die eDeCoDe-Logik und ihre zwei Werkzeuge, die Domain-Mapping-Matrix (DMM) und die Design-Structure-Matrix (DSM), zum Einsatz [Mistler 2021]. Andererseits wird das Prozessreferenzmodell von Automotive SPICE verwendet. Als Ergebnis dieser methodischen Konsolidierung werden die Prinzipien der eDeCoDe-Logik auf das Automotive SPICE Assessmentmodell übertragen, wobei DMM und DSM gezielt eingesetzt werden, um eine präzise Zuordnung und Integration beider Methodiken zu gewährleisten. Die Abbildung 24 visualisiert diese Vorgehensweise.

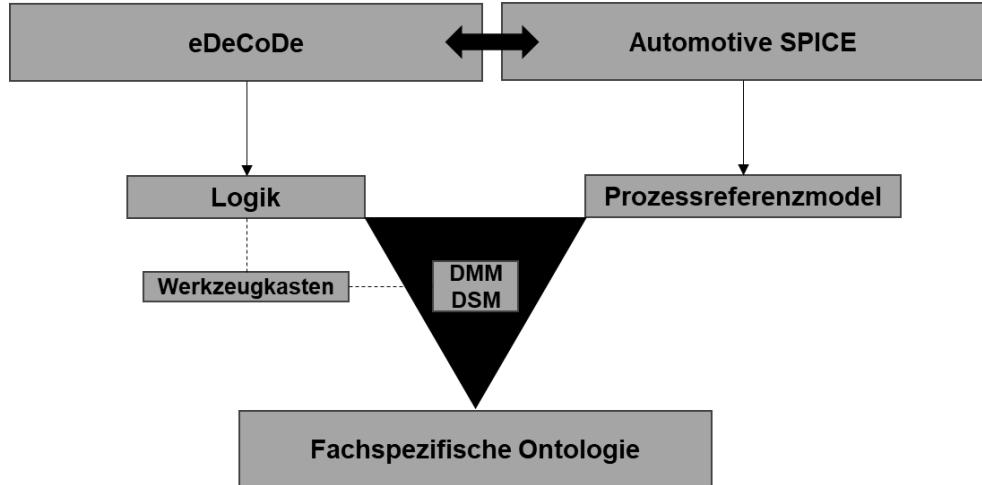


Abbildung 24: Verknüpfungslogik von eDeCoDe und Automotive SPICE [Eigene Darstellung]

Im Abschnitt 2.4.3 wurde erörtert, dass nach der eDeCoDe-Logik Komponenten, Prozesse und Personen Funktionen realisieren. Hierbei verwenden Prozesse Komponenten als Ressourcen, um bestimmte Anforderungen zu erfüllen. Personen setzen ebenfalls Komponenten und Prozesse ein, um diese Funktionen zu verwirklichen [Schlüter 2023]. Im Kontext der Softwareentwicklung im Elektronikbereich der Automobilindustrie lässt sich diese Logik anwenden. Dabei werden eDeCoDe-Systemansichten auf das Projektumfeld abgebildet. Die Beziehungen zwischen diesen Systemsichten werden gemäß den attribuierten Wechselbeziehungen der eDeCoDe-Logik interpretiert. Abbildung 25 veranschaulicht diese zugeordneten Systemsichten.

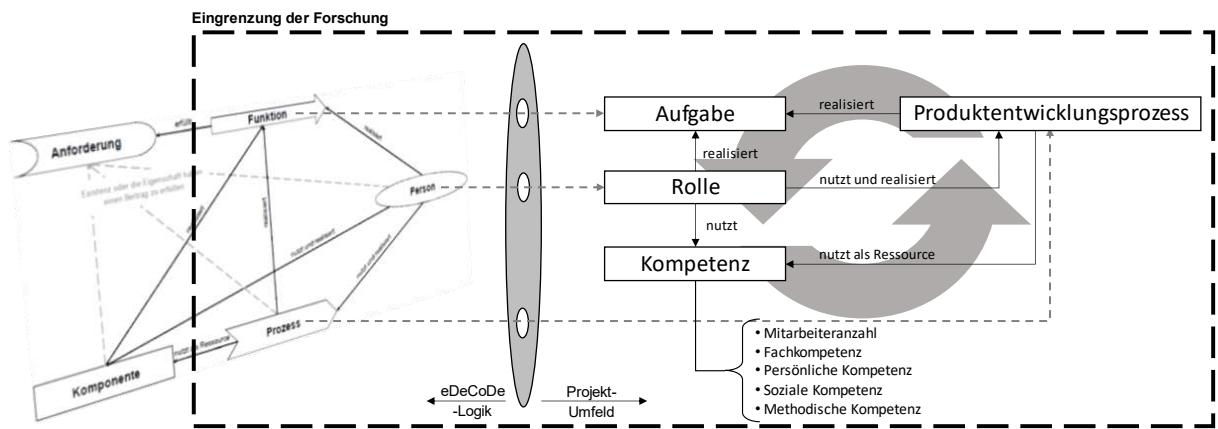


Abbildung 25: Anwendung von eDeCoDe-Logik auf Projektumfeld [Eigene Darstellung]

Auf dieser Basis wird die Prozesssicht des eDeCoDe-Systemansatzes im Projektumfeld auf Produktentwicklungsprozesse, die aus Automotive SPICE stammen, gemappt. Diese realisieren die Aufgaben (engl. Tasks), die innerhalb eines Softwareentwicklungsprojektes zu erledigen sind. Die Aufgaben entsprechen hierbei den Funktionen aus der eDeCoDe-Logik. Die Aufgaben werden wiederum durch Rollen realisiert. Die Rolle nutzt dabei die Produktentwicklungsprozesse und realisiert diese, was dazu führt, dass eine Aufgabe prozesskonform ausgeführt wird. Die Rollen entsprechen der Personensicht in der eDeCoDe-Logik. Damit eine Aufgabe durch eine Rolle realisiert werden kann, nutzt die Rolle

die Kompetenzen, die der Mensch mitbringt, was wiederum als Ressource in verschiedene Prozesse innerhalb der Produktentwicklung einfließt. Die Analyse und das Mapping der zwei eDeCoDe-Sichten – Anforderung und Komponente – würden den Rahmen dieser Forschung sprengen und können in nachfolgenden Forschungen adressiert werden. Darüber hinaus wird der Fokus auf Menschen und ihre verschiedenen Kompetenzarten gelegt, während Zeit und Geld in dieser Hinsicht vernachlässigt werden. Es ist anzumerken, dass die Informationsstruktur, die in Kapitel 5.1.2 entwickelt wurde, in der Lage ist, Zeit und Geld als Ressourcen zu identifizieren und zu kategorisieren, aber da der Forschungsumfang sehr groß wird, werden diese in dieser Forschung ebenso nicht betrachtet.

Die Tabelle 18 fasst das Mapping der eDeCoDe-Logik auf das Projektumfeld zusammen.

Tabelle 18: Mapping der eDeCoDe-Logik auf das Projektumfeld

eDeCoDe-Systemsicht	Projektumfeld	Interpretation	Bemerkung
Prozess	Produktentwicklungsprozess	Realisiert die Aufgabe und nutzt dabei die Kompetenzen als Ressource, um von der Rolle benutzt zu werden und realisiert zu werden.	Die Produktentwicklungsprozesse stammen aus Automotive SPICE.
Funktion	Aufgabe	Wird durch Rolle und Produktentwicklungsprozess realisiert.	Die Aufgaben stammen aus Automotive SPICE.
Person	Rolle	Nutzt die Produktentwicklungsprozesse und die Kompetenzen, um eine Aufgabe innerhalb eines Prozesses zu realisieren.	Die Rollen stammen aus Rollenbeschreibung des Unternehmens innerhalb des Produktentwicklungsprozesses.

Es ist essentiell, die Elemente des Projektumfelds eingehender zu analysieren. Bei der Betrachtung von Automotive SPICE, welches das Gerüst für die Strukturierung von Produktentwicklungsprozessen bietet, werden die Prozesse in unterschiedliche Gruppen unterteilt und detailliert beschrieben (siehe Kapitel 2.4.2). Für diese Forschung wurden fünf Prozessgruppen ausgewählt, die den sogenannten VDA-Scope repräsentieren. Der Schwerpunkt liegt auf dem VDA-Scope, da dieser sich als Standardpraxis in der Automobilindustrie durchgesetzt hat [Müller 2007]. Abbildung 26 illustriert diese Prozessgruppen und die darin enthaltenen Prozesse.

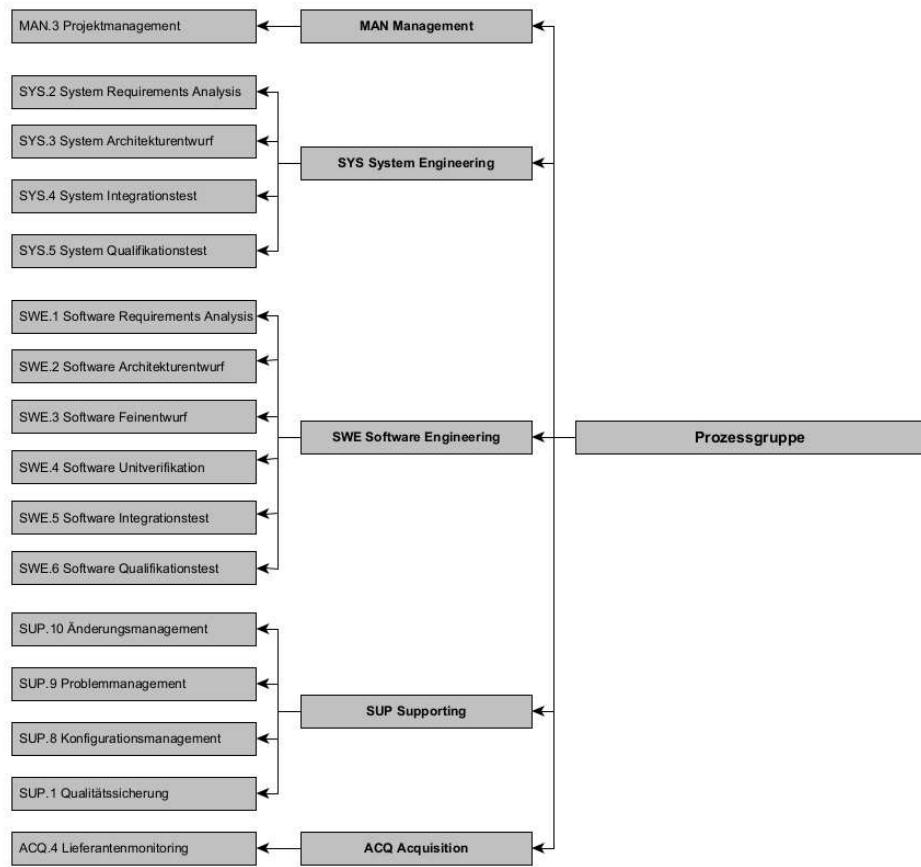


Abbildung 26: Produktentwicklungsprozesse in VDA-Scope in Anlehnung an [ASPICE 2023]

Im nächsten Schritt liegt der Fokus auf den Rollen, die für die Umsetzung der Prozesse verantwortlich sind. Diese Rollen leiten sich aus den „Base Practices“ von Automotive SPICE ab [ASPICE 2023]. Nach der Analyse der dort beschriebenen Aufgaben sind folgende Rollen erforderlich, um die Aufgaben prozesskonform umzusetzen. Es ist wichtig zu beachten, dass eine Person mehrere Rollen übernehmen kann, was abhängig von der Größe des Projekts entschieden wird. In kleineren Projekten kann beispielsweise ein Softwarearchitekt gleichzeitig als Entwickler fungieren oder ein Projektleiter zusätzlich die Rolle des Konfigurationsmanagers übernehmen. Je größer das Projekt, desto wichtiger wird die Trennung der Rollen [Müller 2007]. Abbildung 27 veranschaulicht die erforderlichen Rollen.

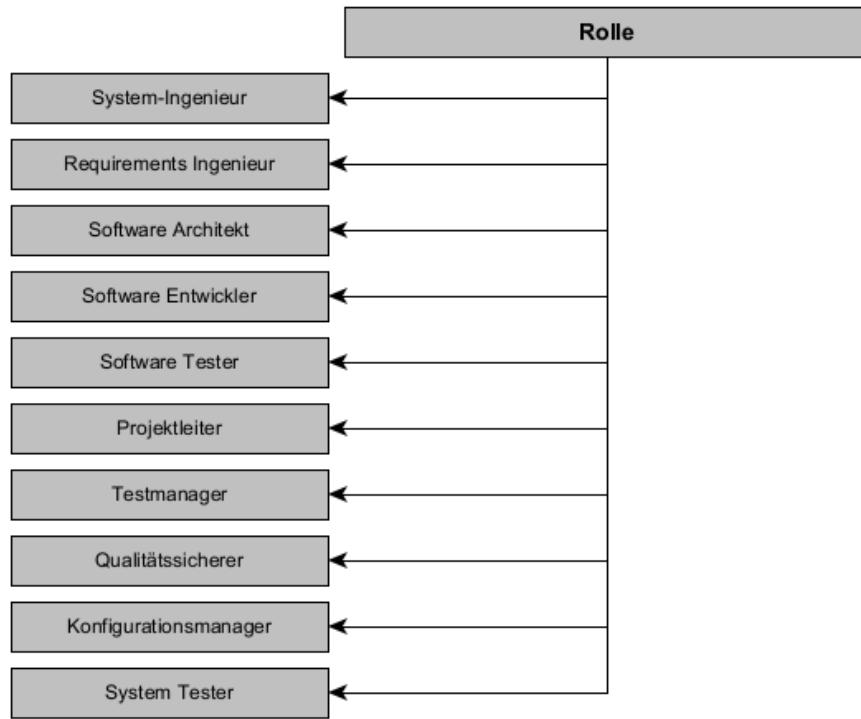


Abbildung 27: Notwendige Rollen abgeleitet von Automotive SPICE [Eigene Darstellung]

Die Aufgaben an sich werden nicht im Detail betrachtet, da die genaue Aufgabendefinition vom Projekt zu Projekt sehr unterschiedlich ist. Die Aufgaben werden so definiert, dass die Base Practices, welche in jeweiligen Prozessen von Automotive SPICE beschrieben sind, erfüllt werden und damit das Projekt Prozesskonform läuft, aber variieren sich je nach Projekttyp und Projektumfang. Jedoch die Kompetenzen, die eine Rolle nutzt, um die Aufgabe zu realisieren, ist wichtig näher betrachtet zu werden. Als zuvor beschrieben, liegt der Fokus hierbei auf Personelle Ressourcen (Mensch) und deren Kompetenzarten. Die Kompetenzarten wurden in Kapitel 5.1.1 vor der Entwicklung von Informationsstruktur herausgearbeitet und in folgende Abbildung 28 visualisiert.



Abbildung 28: Kompetenzarten der personellen Ressourcen [Eigene Darstellung]

Nach der Analyse und Kategorisierung der verschiedenen Sichten aus dem Projektumfeld kann die Entwicklung der Ontologie fortgesetzt werden. Hierbei werden die DMM (Domain-Mapping-Matrix) und DSM (Design Structure Matrix) aus dem eDeCoDe-Werkzeugkasten verwendet (s. Kapitel 2.4.3). Die DMM ermöglicht es, die unterschiedlichen Sichten gegenüberzustellen und zu vernetzen, was zur Erstellung von drei spezifischen Matrizen führt: Prozess-Rolle, Rolle-Kompetenz und Kompetenz-Prozess. Anschließend verknüpft die DSM-Matrix diese drei Perspektiven miteinander. Abbildung 29 visualisiert die Ergebnisse dieses Schrittes (s. Anhang A.1).

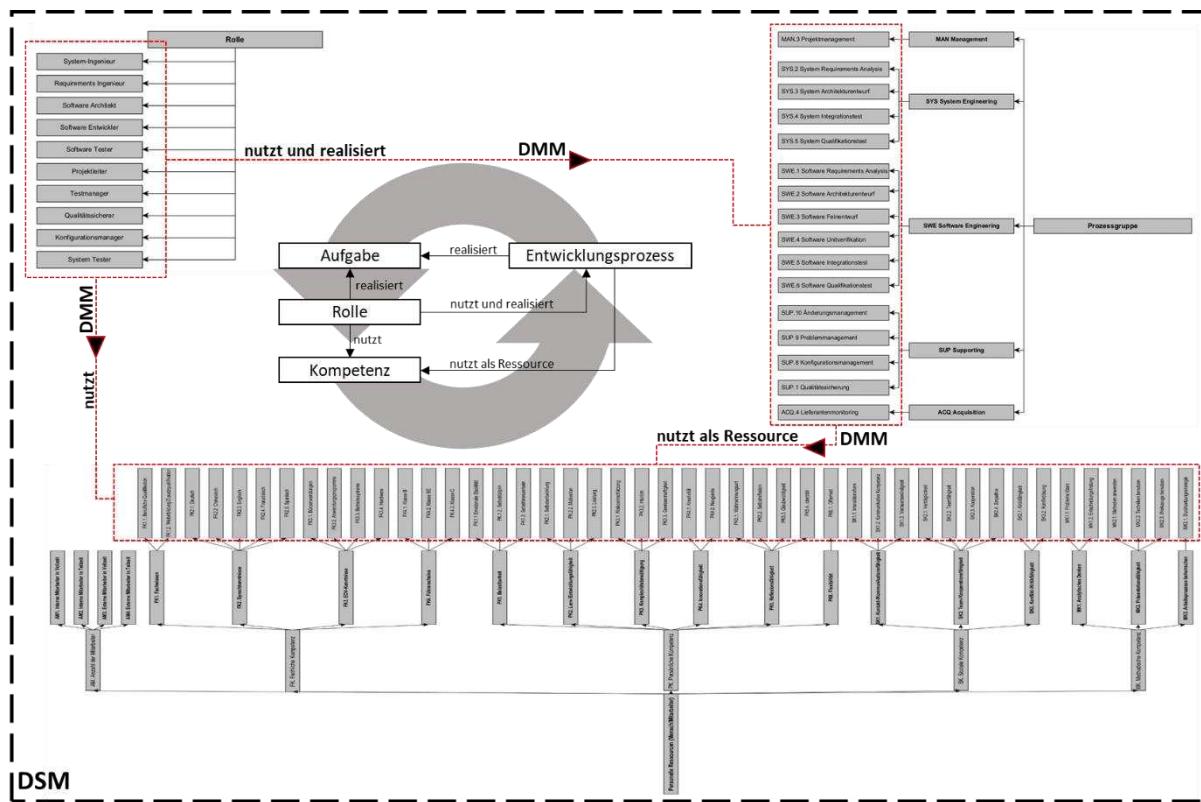
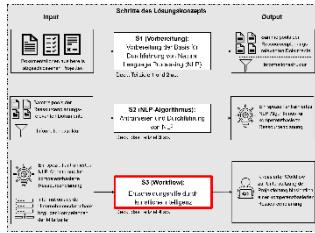


Abbildung 29: Entwickelte Ontologie basierend auf DSM- und DMM-Matrizen [Eigene Darstellung]

Abschließend lässt sich sagen, dass die Etablierung einer fachspezifischen Informationsstruktur, das Aufbauen einer geeigneten NLP-Pipeline und die Entwicklung einer auf Softwareentwicklungsprojekte im Automobilbereich zugeschnittenen unternehmensspezifischen Ontologie den Grundstein für die Einleitung eines KI-basierten Workflows legen. Dieser Workflow zielt darauf ab, die bisherigen Ergebnisse zu integrieren und effektiv zu nutzen.

5.3 Theoretische Entwicklung des KI-gestützten Workflows (S3)



Alle erforderlichen Schritte sind abgeschlossen, und die Grundlage für die Entwicklung des KI-Workflows ist geschaffen worden. Die zuvor entwickelten Komponenten, wie die Informationsstruktur, die NLP-Pipeline und die fachspezifische eDeCoDe-Ontologie, sollen nun miteinander verzahnt und in eine benutzerfreundliche Umgebung integriert werden. Dies ermöglicht es, auf Basis der Erkenntnisse aus der Analyse früherer Projektdokumente Vorhersagen zur Ressourcenplanung für zukünftige Projekte zu treffen. Zu diesem Zweck wird eine Benutzeroberfläche, ein sogenanntes **Front-End**, entwickelt, die es dem Anwender ermöglicht, Daten zu neuen Projekten einzugeben. Diese Daten werden dann an das im Hintergrund agierende **Back-End**, den Algorithmus, weitergeleitet und verarbeitet. Die vom Algorithmus generierten Vorhersagen werden anschließend dem Benutzer präsentiert.

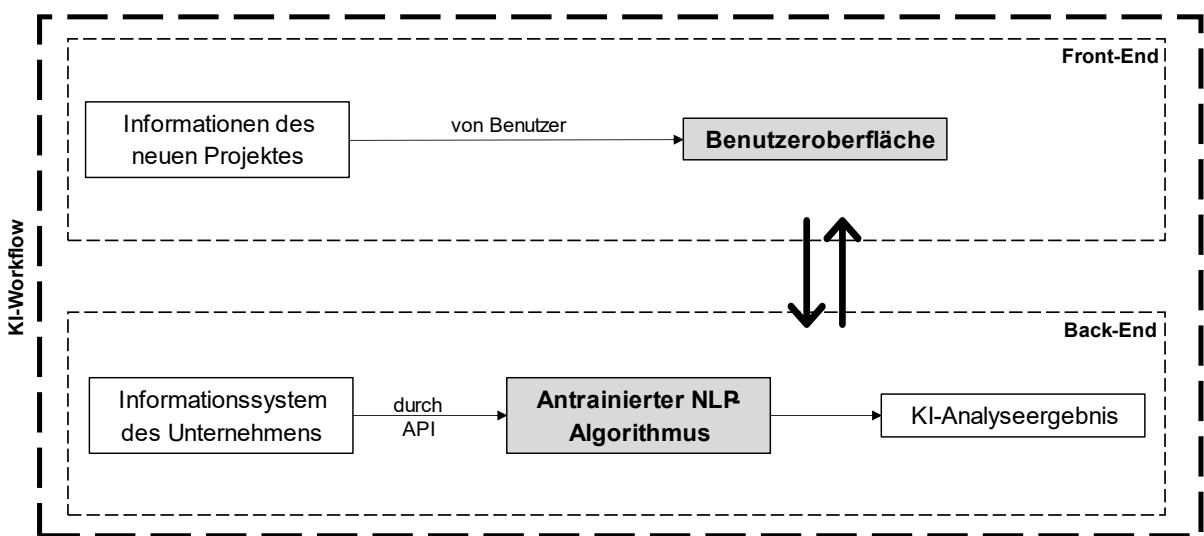


Abbildung 30: Das KI-Workflow [Eigene Darstellung]

Die Abbildung 30 veranschaulicht den Zusammenhang innerhalb des Workflows. Die beiden Hauptbausteine des KI-Workflows – Front-End und Back-End – werden im Folgenden genauer analysiert und beschrieben.

5.3.1 Back-End

Das Backend fungiert als zentrale Verarbeitungseinheit, in der sämtliche Daten gesammelt, in Informationen umgewandelt, aggregiert und analysiert werden, um fundierte Analyseergebnisse zu erzeugen. Diese Ergebnisse werden später im Frontend aufbereitet und dem Nutzer zugänglich gemacht. Herzstück dieses Verarbeitungsprozesses ist der sorgfältig trainierte NLP-Algorithmus, der die anspruchsvolle Aufgabe übernimmt, die gesammelten Informationen zu analysieren und zu interpretieren. Dieser Prozess umfasst das Heranziehen und die Auswertung von Daten aus unterschiedlichen

Quellen, um eine umfassende Basis für Entscheidungsprozesse zu schaffen. Zum einen empfängt der Algorithmus Input-Daten direkt von den Benutzern, die über eine speziell entwickelte Eingabemaske eingeführt werden. Diese Daten stellen Anfragen dar, die detaillierte Informationen über geplante neue Projekte enthalten. Die Spezifikationen und der Umgang mit diesen Eingaben werden detailliert im Kapitel 5.3.2 unter „Front-End“ behandelt. Zum anderen schöpft der Algorithmus aus einem reichen Reservoir an sprachlichem und fachspezifischem Wissen, welches durch die bestehende NLP-Pipeline (siehe Kapitel 5.2.1), die entwickelte Informationsstruktur (siehe Kapitel 5.1.2) und die eDeCoDe-ASPICE-Ontologie (siehe Kapitel 5.2.2) bereitgestellt wird. Diese Komponenten ermöglichen es dem Algorithmus, relevante Informationen aus einem umfangreichen Dokumentenpool effizient zu filtern. Die eDeCoDe-ASPICE-Ontologie spielt dabei eine entscheidende Rolle, indem sie die Beziehungen und den Kontext zwischen den gefilterten Informationen klärt und deren Interpretation unterstützt.

Zusätzlich zu diesen internen Datenquellen bezieht der Algorithmus auch externe Informationen aus den vorhandenen Informationssystemen des Unternehmens. Verschiedene Systeme wie das Qualitätsmanagementsystem liefern wichtige KPIs und Daten über den aktuellen Qualitätsstand der Projekte, Prozesse und Produkte. HR-Systeme wie SAP stellen essentielle Informationen über die Mitarbeiter, deren Finanzen und die Verfügbarkeit von Ressourcen bereit. Darüber hinaus nutzen Unternehmen spezifische Informationssysteme wie SharePoint, die zusätzliche Informationen über Mitarbeiter in Form von ausführlichen Profilen und sozialen Medien-ähnlichen Formaten bieten. Die Mehrheit dieser Daten besteht aus strukturierten Daten in exportierte Tabellenform, die ohne sprachliche Verarbeitung direkt gefiltert und genutzt werden können. Die verbleibenden unstrukturierten Informationen werden durch die NLP-Pipeline geführt, wo sie entsprechend kategorisiert werden. Die Abbildung 31 visualisiert die Spezifizierung von Back-End und die Interaktion der unterschiedlichen Elemente.

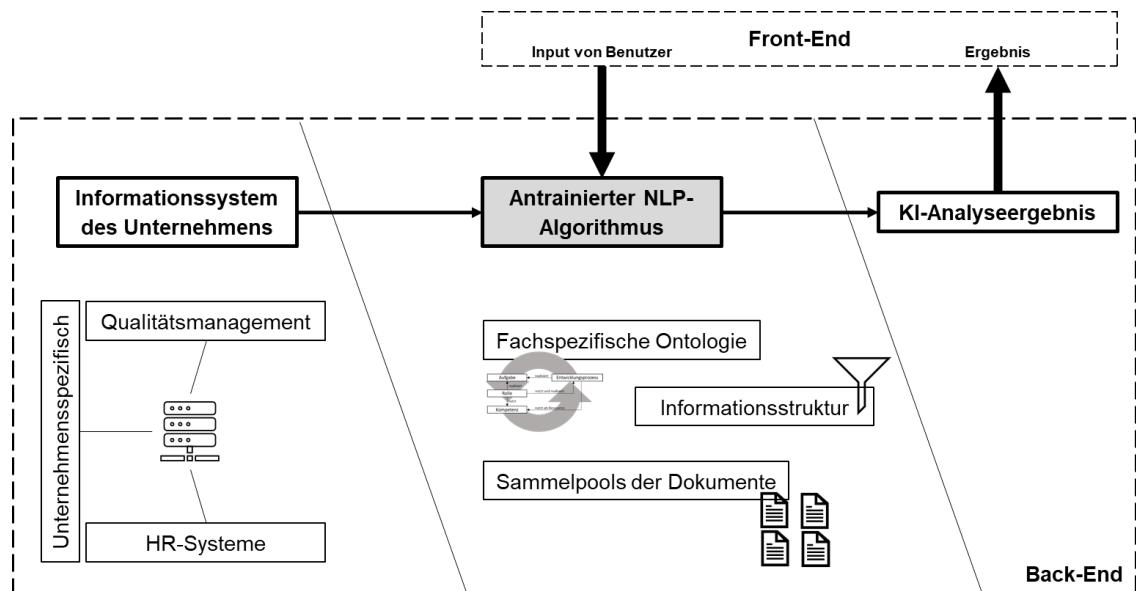


Abbildung 31: Spezifizierung von Back-End

Es ist von größter Wichtigkeit, bei der Verarbeitung aller personenbezogenen Daten besondere Sorgfalt walten zu lassen. Diese Informationen müssen, um die Privatsphäre der Personen zu wahren, anonymisiert oder, wenn möglich, vollständig entfernt werden. Der Schutz personenbezogener Daten steht im Einklang mit Datenschutzbestimmungen und ethischen Überlegungen, die das Vertrauen der Nutzer in die Systeme des Unternehmens stärken und die rechtlichen Rahmenbedingungen erfüllen.

Abschließend werden die vom Benutzer bereitgestellten Input-Daten den durch den Algorithmus analysierten Informationen gegenübergestellt und interpretiert. Diese Gegenüberstellung ermöglicht es, fundierte Analyseergebnisse zu erzeugen, die anschließend an das Frontend übermittelt werden, um sie dem Benutzer auf eine zugängliche und verständliche Weise präsentieren zu können. Diese Präsentation erfolgt durch eine speziell entwickelte Benutzeroberfläche, die darauf ausgelegt ist, die Komplexität der Datenanalyse transparent und nachvollziehbar zu machen. Das Frontend und die dazugehörige Benutzeroberfläche werden daraufhin ausführlich beschrieben, um einen vollständigen Überblick über die Interaktionsmöglichkeiten des Nutzers mit dem System zu bieten. Die Gestaltung des Front-Ends zielt darauf ab, eine intuitive Schnittstelle zu bieten, die es den Benutzern ermöglicht, ihre Daten eingeben, die Ergebnisse der Analyse betrachten und basierend darauf informierte Entscheidungen treffen zu können.

5.3.2 Front-End

Dieser Unterkapitel widmet sich dem Front-End des KI-Workflows, einer entscheidenden Schnittstelle, die es Benutzern ermöglicht, direkt mit dem trainierten Algorithmus zu interagieren. Dies geschieht über eine detailliert gestaltete Eingabemaske, die nicht nur grundlegende Daten wie Projekt-ID und -Typ erfasst, sondern auch erweiterte Rahmeninformationen wie Zeitraum und Budget sowie spezifische Details wie benötigte Rollen und Mitarbeiteranzahl abfragt und zum Schluss nach der Analyse der Informationen die Ergebnisse dem Benutzer präsentiert. Die Abbildung 32 veranschaulicht den Aufbau von Front-End.

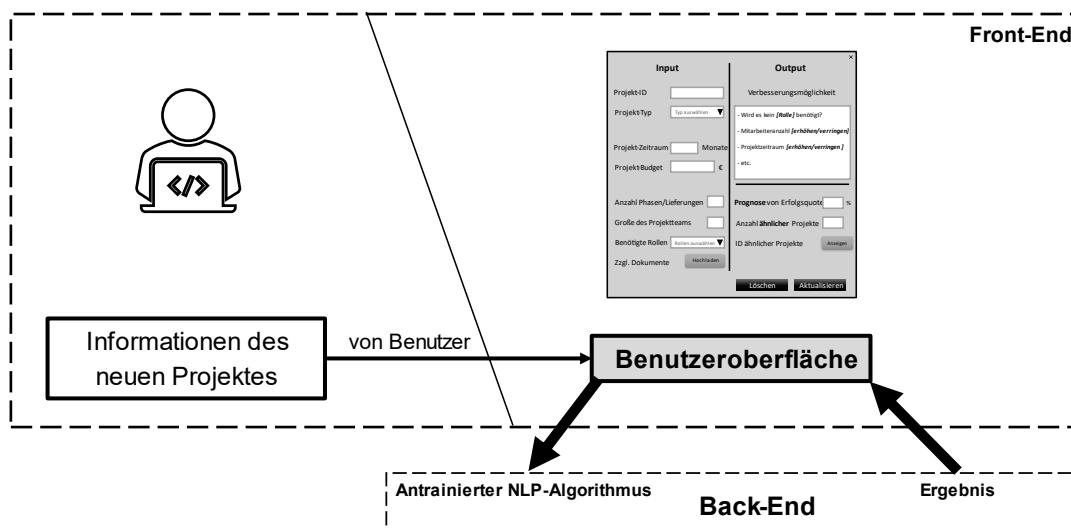


Abbildung 32: Spezifizierung von Front-End

Die Benutzeroberfläche gliedert sich in zwei Hauptbereiche: Input und Output. Im Bereich „**Input**“ bietet sich dem Benutzer die Möglichkeit, verschiedene Informationen über das bevorstehende Projekt einzugeben, die in drei Kategorien unterteilt sind. In der ersten Kategorie werden grundlegende Projektdaten erfasst, wie die Projekt-ID und der Projekttyp. Die Projekt-ID dient als eindeutige Kennung, während der Projekttyp hilft, das Projekt thematisch von anderen zu unterscheiden. In einem Unternehmen variieren Projekte hinsichtlich ihrer Details und Anforderungen. Ein Projekt könnte beispielsweise sicherheitsrelevant sein oder spezifische Normenvorgaben erfüllen müssen. Ein anderes Projekt könnte sich ausschließlich auf die Entwicklung von Software konzentrieren, während ein weiteres zusätzlich die Hardware und Mechanik umfasst. Die Erfassung dieser Eckdaten ermöglicht es, Projekte effektiv zu kategorisieren und vergleichbare Projekte sinnvoll gegenüberzustellen, um plausiblere Ergebnisse zu erzielen. Die zweite Kategorie betrifft Angaben zum Zeitrahmen und Budget des Projekts. Diese Angaben dienen primär als Indikatoren zur Bestimmung der Projektgröße und werden nicht im Sinne der Ressourcennutzung von Zeit oder Geld analysiert. Dies ermöglicht den Vergleich ähnlich dimensionierter Projekte. Der Fokus dieser Forschung liegt, wie bereits erläutert, auf der Betrachtung von Menschen als Ressourcen, während finanzielle Aspekte und Zeitressourcen hier ausgeschlossen werden. In der dritten Kategorie gibt der Benutzer Informationen aus der Projektplanungsphase ein. Dazu zählen die Anzahl der Projektphasen oder Lieferungen an den Kunden, die Größe des vorgesehenen Teams und die benötigten Rollen. Diese Eingaben werden vom Algorithmus verarbeitet, mit den Erkenntnissen aus früheren Projekten abgeglichen und analysiert. Die Ergebnisse dieser Analyse werden dann im Bereich „**Output**“ dargestellt. Zusätzlich besteht die Möglichkeit, diese Informationen über einen speziell vorgesehenen Hochladen-Knopf direkt in das System einzuspeisen.

Im „**Output**“-Bereich werden die vom Algorithmus generierten Analyseergebnisse präsentiert. Dieser Abschnitt bietet Verbesserungsvorschläge in Form von Hinweisen und Empfehlungen. So könnte beispielsweise, basierend auf der Analyse, darauf hingewiesen werden, dass bestimmte vom Benutzer gewählte Rollen nicht den Erkenntnissen des Algorithmus entsprechen und weitere Rollen berücksichtigt werden sollten. Weiterhin könnten Vorschläge gemacht werden, die Mitarbeiteranzahl oder den Projektzeitraum anzupassen. Es wird auch eine Prognose zur Erfolgsquote des Projekts angezeigt, die dem Benutzer verdeutlicht, wie erfolgreich das Projekt ohne Berücksichtigung der vorgeschlagenen Verbesserungen sein könnte. Zudem wird die Anzahl ähnlicher, bereits durchgeföhrter Projekte präsentiert, und der Benutzer kann, falls gewünscht, die IDs dieser Projekte einsehen.

Die Benutzeroberfläche umfasst zudem zwei **Funktionsknöpfe**: „Löschen“ und „Aktualisieren“. Diese ermöglichen es dem Benutzer, Eingaben zu löschen und von vorne zu beginnen oder die eingegebenen Informationen zu aktualisieren und erneut zur Analyse an den Algorithmus zu senden. Die Abbildung 33 visualisiert die prototypisch entwickelte Oberfläche, die diese Funktionalitäten unterstützt.

Input	Output
Projekt-ID <input type="text"/>	Verbesserungsmöglichkeit
Projekt-Typ <input type="button" value="Typ auswählen ▼"/>	- Wird es kein [Rolle] benötigt? - Mitarbeiteranzahl [erhöhen/verringen] - Projektzeitraum [erhöhen/verringen] - etc.
Projekt-Zeitraum <input type="text"/> Monate	
Projekt-Budget <input type="text"/> €	
Anzahl Phasen/Lieferungen <input type="text"/>	Prognose von Erfolgsquote <input type="text"/> %
Große des Projektteams <input type="text"/>	Anzahl ähnlicher Projekte <input type="text"/>
Benötigte Rollen <input type="button" value="Rollen auswählen ▼"/>	ID ähnlicher Projekte <input type="button" value="Anzeigen"/>
Zzgl. Dokumente <input type="button" value="Hochladen"/>	
<input type="button" value="Löschen"/> <input type="button" value="Aktualisieren"/>	

Abbildung 33: Die Benutzer Oberfläche [Eigene Darstellung]

Mit der detaillierten Erörterung der theoretischen Entwicklung des KI-Workflows und der Struktur der verschiedenen Schritte ist ein solides Verständnis für das angestrebten Vorgehen geschaffen worden. Im nächsten Kapitel wird die technische Umsetzung dieser Schritte behandelt. Dabei wird detailliert dargelegt, wie die einzelnen Module des Workflows technisch realisiert und programmiert werden. Diese Phase ist entscheidend, um die theoretischen Grundlagen in funktionierende, praktische Anwendungen zu überführen, die die Effizienz und Effektivität des Gesamtsystems unter Beweis stellen. Es wird untersucht, welche technologischen Entscheidungen getroffen wurden, um die Zuverlässigkeit und Leistungsfähigkeit der Systemkomponenten zu sichern und wie diese Entscheidungen die Nutzererfahrung beeinflussen.

5.4 Technische Umsetzung des KI-gestützten Workflows

Im Rahmen der technischen Umsetzung werden die theoretisch entwickelten Konzepte im letzten Kapitel 5.3 in konkrete, funktionale Komponenten überführt. Dies umfasst die Programmierung der einzelnen Module, die den gesamten KI-Workflow bilden. Der Fokus liegt dabei auf der Implementierung und Integration verschiedener Technologien und Werkzeuge, um eine kohärente und effiziente Systemarchitektur zu gewährleisten. Es werden die einzelnen Schritte und Methoden beschrieben, die zur Programmierung der Module notwendig sind, sowie die Herausforderungen und Lösungen, die während der Entwicklung aufgetreten sind. Besonderes Augenmerk wird auf die Auswahl der

Technologien, die Struktur des Codes und die Optimierung der Leistung gelegt, um sicherzustellen, dass die entwickelten Komponenten nicht nur den theoretischen Anforderungen entsprechen, sondern auch in der Praxis robust und zuverlässig funktionieren. Die Implementierung wird in drei Python-Skripten vorgenommen. In den ersten beiden Skripten wird der Code für das Back-End entwickelt. Dabei wird hauptsächlich die Bibliotheken SpaCy und Panda genutzt, um die NLP-Pipeline für das Training des Algorithmus und die Analyse neuer Dokumente nach dem Training zu entwickeln. Das dritte Skript verwendet TKinter-Bibliotheken, um eine grafische Benutzeroberfläche (GUI) für das Front-End zu entwickeln, die mit der NLP-Pipeline interagiert.

5.4.1 Programmierung von NLP-Pipeline

Nachfolgend wird Schritt für Schritt das Skript der NLP-Pipeline erläutert, das zum Training eines NLP-Modells unter Verwendung von spaCy, einer populären NLP-Bibliothek, entworfen wurde. Um die Lesbarkeit und Wartbarkeit zu verbessern, wurde das Skript in mehrere Funktionen unterteilt. Diese Funktionen decken die Hauptaufgaben ab: das Laden der Trainingsdaten, die Vorbereitung der Daten für das Training, das Training des Modells und das Speichern des trainierten Modells. Zudem werden Kommandozeilenargumente unterstützt, um Flexibilität beim Training zu bieten. Die spezifischen Implementierungsdetails werden behandelt und die Funktionsweise jedes Abschnitts wird erläutert, um ein umfassendes Verständnis der gesamten Pipeline zu vermitteln. Dabei wird ebenfalls darauf eingegangen, wie die einzelnen Komponenten zusammenwirken, um das Ziel der Modellierung zu erreichen. Die Tabelle 19 listet die sechs Bereiche und deren Funktionen für eine bessere Navigation in Code auf:

Tabelle 19: Die sechs Hauptbereiche des Skripts

	Bereich	Funktionen
1	Importe und Konfigurationsprotokollierung	<i>Import</i> <code>logging.basicConfig ()</code>
2	Datenladefunktionen	<code>load_training_data ()</code> <code>load_ontology ()</code>
3	Datenvorbereitungsfunktionen	<code>prepare_training_examples ()</code> <code>enhance_training_data_with_ontology ()</code>
4	Modelltrainingsfunktionen	<code>train_model ()</code> <code>save_model ()</code>
5	Ontologie-Integration	<code>Class OntologyComponent: init() and</code> <code>call()</code> <code>add_ontology_component ()</code>
6	Hauptfunktion und Argumentenparser	<code>Main () --> if __name__ == "__main__"</code>

Der erste Bereich „(1) Importe und Konfigurationsprotokollierung“ umfasst die **Import**-Anweisungen für alle erforderlichen Bibliotheken sowie richtet die grundlegende Konfiguration für das **Logging** ein. Die Importe bringen externe Bibliotheken und Module ein, die für verschiedene Funktionalitäten wie die Verarbeitung natürlicher Sprache, Datenmanipulation und Argumentenparsing benötigt werden. Die Logging-Konfiguration stellt sicher, dass die Anwendung wichtige Ereignisse und Fehler protokolliert, was für das Debugging und die Überwachung des Trainingsprozesses unerlässlich ist. Die genutzten Bibliotheken sind wie folgt:

- *spacy*: Die Hauptbibliothek für Aufgaben der Verarbeitung natürlicher Sprache.
- *from spacy.training import Example*: Importiert die Example-Klasse zum Erstellen von Trainingsbeispielen.
- *from spacy.util import minibatch*: Importiert die minibatch-Funktion zur Erstellung von Mini-Batches für das Training.
- *pandas as pd*: Die pandas-Bibliothek, als pd abgekürzt, wird für die Datenmanipulation und -analyse verwendet, insbesondere für den Umgang mit CSV- und Excel-Dateien.
- *random*: Bietet Funktionen zur Erzeugung zufälliger Zahlen und zum Mischen von Daten.
- *os*: Wird für die Interaktion mit dem Betriebssystem verwendet, beispielsweise zum Erstellen von Verzeichnissen.
- *argparse*: Erleichtert das Parsen von Kommandozeilenargumenten.
- *logging*: Wird für die Konfiguration des Loggings verwendet, um die Ausführung der Anwendung zu verfolgen.
- *json*: Bietet Funktionen für die Arbeit mit JSON-Daten, die hier verwendet werden, um Annotationsdaten sicher zu parsen.

Nach den **Importen** erfolgt die Konfiguration des Loggings, die unerlässlich ist, um die Ausführung des Skripts zu verfolgen und zu debuggen. Dabei werden wichtige Ereignisse und Fehler in einer Logdatei namens *training.log* erfasst. Diese Einrichtung ermöglicht eine effektive Nachverfolgung und Analyse von Problemen während der Ausführung. Die Logging-Konfiguration wird durch den Aufruf von „**logging.basicConfig()**“ eingerichtet. Das Logging-Level wird auf *INFO* gesetzt, wodurch alle Nachrichten auf diesem Level und höher (WARNING, ERROR, CRITICAL) protokolliert werden. Die Logdatei wird im Schreibmodus geöffnet, was bedeutet, dass sie bei jedem Ausführen des Skripts überschrieben wird. Dies stellt sicher, dass jede Ausführung des Skripts mit einer frischen Logdatei beginnt.

Der zweite Bereich „(2) Datenladefunktionen“ definiert zwei Funktionen zum Laden von Trainingsdaten aus einer CSV-Datei sowie Ontologiedaten aus einer Excel-Datei. Diese Funktionen sind dafür verantwortlich, die Eingangsdaten zu lesen, sie in die entsprechenden Formate zu konvertieren und die Vorverarbeitungsschritte durchzuführen. Die erste Funktion in diesem Abschnitt ist

„**`load_training_data`**“. Diese Funktion lädt Trainingsdaten aus einer CSV-Datei und bereitet sie für das Training des Modells vor. Sie nimmt als Parameter den Pfad zur CSV-Datei, die die Trainingsdaten enthält. Der Prozess beginnt mit dem Einlesen der CSV-Datei in ein „pandas“ DataFrame. Anschließend iteriert die Funktion über jede Zeile des DataFrames, extrahiert den Text und die entsprechenden Annotationen. Die Annotationsdaten werden dabei von einem JSON-String in ein Python-Dictionary mittels `json.loads` umgewandelt. Jede Text-Annotation-Paarung wird dann einer Liste namens `training_data` hinzugefügt. Am Ende gibt die Funktion die Liste der Trainingsdaten zurück. Bei Problemen beim Laden der Daten protokolliert die Funktion eine Fehlermeldung und löst eine Ausnahme aus. Die zweite Funktion in diesem Abschnitt ist „**`load_ontology`**“. Diese Funktion lädt Ontologiedaten aus einer Excel-Datei und konvertiert sie in ein Dictionary-Format. Sie nimmt als Parameter den Pfad zur Excel-Datei, die die Ontologiedaten enthält. Der Prozess beginnt mit dem Einlesen der Excel-Datei in ein pandas DataFrame. Danach iteriert die Funktion über jede Zeile des DataFrames, extrahiert die Entität und die zugehörigen Beziehungen. Die Zeilendaten werden in ein Dictionary umgewandelt, wobei die Spalte „Entity“ ausgeschlossen wird. Jede Entität wird dann ihren Beziehungen in einem Dictionary `ontology` zugeordnet. Zum Schluss gibt die Funktion das Ontology-Dictionary zurück. Auch hier werden bei Problemen beim Laden der Ontologiedaten eine Fehlermeldung protokolliert und eine Ausnahme ausgelöst.

Der nächste Bereich „(3) Datenvorbereitungsfunktionen“ enthält Funktionen zur Vorbereitung der Trainingsdaten für die Verwendung im Training des spaCy-Modells. Dies umfasst das Erstellen von Trainingsbeispielen und das Anreichern der Trainingsdaten mit Ontologie-Features. Die Funktion „**`prepare_training_examples`**“ hat das Ziel, die Rohtrainingsdaten in spaCy Example-Objekte zu konvertieren, die für das Training des Modells verwendet werden können. Sie nimmt zwei Parameter: das spaCy-Sprachmodell (`nlp`) und eine Liste von Text-Annotation-Paaren (`training_data`). Der Prozess beginnt mit dem Iterieren über jedes Text-Annotation-Paar in den Trainingsdaten. Der Text wird in ein spaCy Doc-Objekt mithilfe von `nlp.make_doc` umgewandelt. Anschließend wird ein Example-Objekt aus dem Doc und den entsprechenden Annotationen mittels `Example.from_dict` erstellt. Jedes Example-Objekt wird einer Liste namens `examples` hinzugefügt. Am Ende gibt die Funktion die Liste der Example-Objekte zurück. Die zweite Funktion in diesem Abschnitt ist „**`enhance_training_data_with_ontology`**“. Diese Funktion dient dazu, die Trainingsdaten durch Hinzufügen von Ontologie-Features zu den Annotationen zu erweitern. Sie nimmt zwei Parameter: eine Liste von Text-Annotation-Paaren (`training_data`) und ein Wörterbuch, das Entitäten ihren Ontologie-Features zuordnet (`ontology`). Der Prozess beginnt mit dem Iterieren über jedes Text-Annotation-Paar in den Trainingsdaten. Die Annotationen werden kopiert, um die Originaldaten nicht zu verändern. Anschließend wird über jede Entität in den Annotationen iteriert, der Entitätstext aus dem Haupttext mithilfe der Start- und Endpositionen extrahiert und geprüft, ob der Entitätstext in der Ontologie vorhanden ist. Wenn dies der Fall ist, werden die

Ontologie-Features den Annotationen hinzugefügt. Jedes erweiterte Text-Annotation-Paar wird einer Liste namens `enhanced_data` hinzugefügt. Zum Schluss gibt die Funktion die Liste der erweiterten Trainingsdaten zurück. Beide Funktionen sind darauf ausgelegt, effizient und modular zu sein, was es einfach macht, sie an unterschiedliche Datentypen oder zusätzliche Vorverarbeitungsschritte anzupassen oder zu erweitern.

Der nächste Bereich „(4) Modelltrainingsfunktionen“ umfasst Funktionen zum Trainieren des spaCy-Modells und zum Speichern des trainierten Modells auf der Festplatte. Die Trainingsfunktion übernimmt den Optimierungsprozess, während die Speicherfunktion sicherstellt, dass das trainierte Modell für die zukünftige Verwendung gespeichert wird. Die erste Funktion in diesem Abschnitt ist „`train_model`“. Diese Funktion hat das Ziel, das spaCy-Modell mithilfe der bereitgestellten Trainingsbeispiele zu trainieren. Sie nimmt folgende Parameter: das spaCy-Sprachmodell (`nlp`), eine Liste von spaCy Example-Objekten (`examples`), die für das Training vorbereitet wurden. Die Anzahl der Trainingsepochen wird durch (`n_iter`, standardmäßig 10) festgelegt. Eine Epoche bezeichnet einen vollständigen Durchgang durch den gesamten Trainingsdatensatz. Mit einer höheren Anzahl von Iterationen kann das Modell die Trainingsdaten besser lernen, allerdings besteht auch die Gefahr des „Overfittings“, wenn das Modell zu viele Iterationen durchläuft. Der Standardwert ist auf 10 gesetzt, was bedeutet, dass der Trainingsprozess zehnmal durch den gesamten Trainingsdatensatz iteriert. Die Batch-Größe für das Training wird durch (`batch_size`, standardmäßig 2) kontrolliert. Dieser Parameter legt die Anzahl der Trainingsbeispiele fest, die in einem Durchgang (engl. Batch) durch das Modell laufen. Das Training in Batches ist effizienter als das Training mit einzelnen Beispielen, da es die Berechnungsvorgänge parallelisieren und die Gesamttrainingszeit reduzieren kann. Kleinere Batch-Größen können die Generalisierungsfähigkeit des Modells verbessern, aber größere Batch-Größen machen den Trainingprozess schneller und stabiler. Der Standardwert ist auf 2 gesetzt, was bedeutet, dass jeweils zwei Trainingsbeispiele gleichzeitig verarbeitet werden. Anschließend wird die Dropout-Rate für das Training durch (`dropout`, standardmäßig 0,5) gesetzt. Dieser Parameter gibt die Wahrscheinlichkeit an, mit der Neuronen während des Trainings zufällig deaktiviert werden. Neuronen, im Kontext von maschinellem Lernen und neuronalen Netzwerken, sind die grundlegenden Berechnungseinheiten, die Daten verarbeiten und lernen. Das Ziel des Dropouts ist es, das Modell robuster gegen Overfitting zu machen, indem es verhindert, dass Neuronen zu stark auf bestimmte Merkmale der Trainingsdaten spezialisiert werden. Ein Dropout-Wert von 0,5 bedeutet beispielsweise, dass jedes Neuron mit einer Wahrscheinlichkeit von 50% während jedes Trainingsschritts deaktiviert wird. Dies zwingt das Modell, eine breitere Repräsentation der Daten zu lernen, was die Generalisierungsfähigkeit auf neue, ungewohnte Daten verbessert. Der Training-Prozess beginnt mit der Initialisierung des Optimierers durch `optimizer = nlp.begin_training`. Anschließend wird über die angegebene Anzahl von Trainingsepochen iteriert, wobei die Trainingsbeispiele zu Beginn

jeder Epoche gemischt werden. Darüber hinaus wird ein *loss* Dictionary initialisiert, um die Trainingsverluste zu verfolgen. Mini-Batches werden aus den Trainingsbeispielen mithilfe von *minibatch* erstellt. Über jede Batch wird iteriert und das Modell wird mit *nlp.update* aktualisiert, wobei die Dropout-Rate angewendet und die Verluste aufgezeichnet werden. Die Verluste jeder Epoche werden protokolliert, um den Trainingsprozess zu überwachen. Die zweite Funktion aus dem vierten Bereich ist „**save_model**“. Diese Funktion dient dazu, das trainierte spaCy-Modell im angegebenen Verzeichnis zu speichern. Sie nimmt zwei Parameter: das trainierte spaCy-Sprachmodell (*nlp*) und das Verzeichnis, in dem das Modell gespeichert werden soll (*output_dir*). Der Prozess beginnt mit der Überprüfung, ob das angegebene Ausgabeverzeichnis existiert; falls nicht, wird es mit *os.makedirs* erstellt. Das Modell wird dann mit *nlp.to_disk* in das Verzeichnis gespeichert. Eine Meldung wird protokolliert, die angibt, dass das Modell erfolgreich gespeichert wurde. Bei Problemen beim Speichern des Modells wird eine Fehlermeldung protokolliert.

Der Bereich „(5) Ontologie-Integration“ definiert die benutzerdefinierte Komponente zur Integration von Ontologie-Features in die spaCy-Pipeline. Er umfasst die Erstellung der benutzerdefinierten Ontologie-Komponente und die Funktion, diese Komponente zur spaCy-Pipeline hinzuzufügen. Zunächst wird eine Klasse namens „**OntologyComponent**“ definiert, die zwei Methoden, *init* und *call*, enthält. Dies hat den Zweck, eine benutzerdefinierte spaCy-Pipeline-Komponente zu definieren, die Ontologie-Features zu Entitäten in einem Dokument hinzufügt. Die Klasse nimmt zwei Parameter entgegen: das spaCy-Sprachmodell (*nlp*) und ein Wörterbuch, das Entitäten ihren Ontologie-Features zuordnet (*ontology*). Im Konstruktor *__init__* wird die Komponente mit der bereitgestellten Ontologie initialisiert. Die Methode *__call__* iteriert über die Entitäten (*ents*) im Dokument (*doc*), prüft, ob der Entitätstext in der Ontologie vorhanden ist, und weist, falls vorhanden, die Ontologie-Features das Attribut *_features* der Entität zu. Schließlich gibt die Methode das modifizierte Dokument zurück. Anschließend wird eine zweite Funktion namens „**add_ontology_component**“ definiert, um die benutzerdefinierte OntologyComponent zur spaCy-Pipeline hinzuzufügen. Diese Funktion nimmt ebenfalls das spaCy-Sprachmodell (*nlp*) und das Wörterbuch mit den Ontologie-Features als Parameter entgegen. Der Prozess in dieser Funktion beginnt damit, zu prüfen, ob die Ontologie-Komponente bereits im spaCy-Register registriert ist. Falls nicht, wird die Ontologie-Komponente im spaCy-Register registriert. Das spaCy-Register ist ein Mechanismus innerhalb der spaCy-Bibliothek, der es ermöglicht, benutzerdefinierte Komponenten und Erweiterungen zu registrieren und in die spaCy-Pipeline zu integrieren. Es dient als zentrale Stelle, um benutzerdefinierte Funktionen, wie beispielsweise neue Pipeline-Komponenten, Erweiterungen oder Modelle, zu verwalten und zugänglich zu machen. Anschließend wird die Ontologie-Komponente als letzte Komponente zur spaCy-Pipeline hinzugefügt, unter Verwendung der bereitgestellten Ontologie-Kon-

figuration. Zusammenfassend ermöglicht diese Struktur die nahtlose Integration von Ontologie-Features in die spaCy-Pipeline, wodurch die Entitäten in einem Dokument durch zusätzliche semantische Informationen angereichert werden können.

Der letzte Abschnitt „(6) Hauptfunktion und Argumentenparser“ umfasst die Hauptfunktion, die den gesamten Prozess des Datenladens, Modelltrainings und Speicherns des trainierten Modells orchestriert. Außerdem enthält dieser Abschnitt die Einrichtung des Argumentparsers zur Verarbeitung von Kommandozeilenargumenten. Diese Hauptfunktion hat den Zweck, den gesamten Workflow des Skripts zu orchestrieren: Daten laden, Trainingsbeispiele vorbereiten, das Modell trainieren und das trainierte Modell speichern. Sie nimmt die durch `argparse` geparssten Kommandozeilenargumente als Parameter entgegen. Der Prozess beginnt mit dem Laden des spaCy-Modells „`de_dep_news_trf`“. Es werden die Labels definiert, die der NER-Komponente hinzugefügt werden sollen. Dann werden die Ontologiedaten aus dem angegebenen Pfad mithilfe von `load_ontology` geladen und die Trainingsdaten aus dem angegebenen CSV-Pfad mithilfe von `load_training_data`. Diese Trainingsdaten werden dann mit Ontologie-Features angereichert, indem `enhance_training_data_with_ontology` verwendet wird. Die Trainingsbeispiele werden mit `prepare_training_examples` vorbereitet. Das Modell wird unter Verwendung der Funktion `train_model` mit den angegebenen Iterationen, der Batch-Größe und der Dropout-Rate trainiert. Danach wird das trainierte Modell mithilfe von `save_model` gespeichert. Schließlich wird die Ontologie-Komponente mithilfe von `add_ontology_component` zur spaCy-Pipeline hinzugefügt. Bei auftretenden Fehlern werden diese protokolliert und die Ausnahme wird ausgelöst. Der Argumentparser und die Skriptausführung setzen das Parsen von Kommandozeilenargumenten auf und starten die Hauptfunktion. Der Prozess beginnt mit der Erstellung eines `ArgumentParser`-Objekts mit einer Beschreibung des Skripts. Es werden erforderliche positionsabhängige Argumente definiert: `csv_path` für den Pfad zur CSV-Datei mit den Trainingsdaten, `ontology_path` für den Pfad zur Excel-Datei mit den Ontologiedaten und `output_dir` für das Verzeichnis zum Speichern des trainierten Modells. Es werden auch optionale Argumente mit Standardwerten definiert: `--iterations` für die Anzahl der Trainingsepochen (Standardwert ist 10), `--batch_size` für die Batch-Größe beim Training (Standardwert ist 2) und `--dropout` für die Dropout-Rate beim Training (Standardwert ist 0,5). Die Kommandozeilenargumente werden geparsst und die Hauptfunktion wird mit den geparssten Argumenten aufgerufen.

Nachdem das Modell trainiert wurde, kann es zur Analyse neuer Dokumente verwendet werden. Diese erfolgt in einem **anderen** Skript. Das zweite Skript ist so konzipiert, dass ein trainiertes spaCy-Modell geladen und zur Verarbeitung und Analyse von Textdokumenten eingesetzt wird. Das Skript umfasst Funktionen zum Laden des Modells von der Festplatte, zur Verarbeitung eines gegebenen Dokuments zur Entitätsextraktion und zur Analyse dieser Entitäten, um spezifische Einblicke zu gewinnen. Der Hauptausführungsblock verbindet diese Komponenten, indem ein Beispieldokument verarbeitet und

die Analyseergebnisse protokolliert werden. Diese Struktur erleichtert einfache Anpassungen und Erweiterungen für verschiedene Arten von Textanalyseaufgaben. Das Skript ist in zwei Hauptabschnitte unterteilt: **Funktionen** und **Hauptausführungsblock**. Die Funktion „`load_model`“ ist dafür verantwortlich, dass ein trainiertes spaCy-Modell vom angegebenen Dateipfad geladen wird. Die Funktion „`process_document`“ wird verwendet, um ein gegebenes Textdokument mit dem geladenen spaCy-Modell zu verarbeiten und Entitäten zu extrahieren. Die Funktion „`analyze_entities`“ analysiert die extrahierten Entitäten aus dem verarbeiteten Dokument, um Einblicke zu gewinnen. Im Hauptausführungsblock wird der Workflow des Skripts orchestriert, indem das Modell geladen, ein Dokument verarbeitet und die extrahierten Entitäten analysiert werden. Der Prozess umfasst folgende Schritte:

- **Modell laden:** „`load_model`“ wird mit dem Pfad zum trainierten Modell aufgerufen.
- **Dokument verarbeiten:** Ein Beispieltextdokument wird definiert und mit „`process_document`“ verarbeitet.
- **Entitäten analysieren:** Das verarbeitete Dokument wird mit „`analyze_entities`“ analysiert.

Diese Struktur ermöglicht eine klare und modulare Herangehensweise an die Textverarbeitung und -analyse, wodurch das Skript leicht für unterschiedliche Analyseaufgaben angepasst und erweitert werden kann.

5.4.2 Programmierung von Benutzeroberfläche

Nachdem das Skript zur Analyse von Textdokumenten mit einem trainierten spaCy-Modell erläutert wurde, soll nun das Skript zum Benutzeroberfläche (engl. GUI) vorgestellt werden. Das GUI-Skript bietet eine grafische Schnittstelle, die es dem Benutzer ermöglicht, das Modell interaktiv zu nutzen, Dokumente zu analysieren und die Ergebnisse in einer intuitiven Weise darzustellen. Diese Erweiterung erleichtert die Anwendung des Modells erheblich und macht es auch für Nutzer ohne Programmierkenntnisse zugänglich. Im Folgenden wird das GUI-Skript detailliert beschrieben.

Die Tabelle 20 listet die sechs Bereiche und deren Funktionen für eine bessere Navigation in Code auf:

Tabelle 20: Hauptbereiche des GUI-Skripts

	Bereich	Funktionen
1	Importe und Klassendefinition	<i>Import</i> <code>class ProjectApp ()</code>
2	Initialisierung und Erstellung der Widgets	<code>(__init__)</code> <code>create_widgets ()</code> <code>reset_fields ()</code> <code>run_script_aktualisieren ()</code> <code>run_script_anzeigen ()</code> <code>upload_file ()</code>
3	Funktionsmethoden	
4	Hauptausführungsblock	<code>if __name__ == "__main__":</code>

Der erste Bereich des Skripts „(1) Importe und Klassendefinition“ enthält, analog zum ersten Skript, die **Import**-Anweisungen, die die benötigten Bibliotheken und Module für die Anwendung einbinden. Dabei wird „tkinter“ verwendet, die Standard-GUI-Bibliothek in Python, wobei *tk* der Hauptteil dieser Bibliothek ist. Zudem wird *ttk* verwendet, was für „Themed Tkinter Widgets“ steht und eine modernere und ansprechendere Alternative zu den Standard-Tkinter-Widgets bietet. Das Modul *filedialog* stellt Dialoge für das Öffnen und Speichern von Dateien bereit, während *messagebox* Dialoge für Nachrichten und Warnungen bereitstellt. Das *subprocess*-Modul ermöglicht es, neue Prozesse zu starten und mit ihnen zu kommunizieren, was hier genutzt wird, um externe Python-Skripte, wie z.B. das erste Skript, auszuführen. Die Klasse **ProjectApp** erbt von *tk.Tk*, der Hauptklasse der Tkinter-Bibliothek zur Erstellung von GUI-Anwendungen. Diese Klasse definiert die Struktur und Funktionalität der gesamten GUI-Anwendung. Der Konstruktor der Klasse (`__init__`) wird beim Erstellen einer Instanz der Klasse aufgerufen und ruft den Konstruktor der Elternklasse *tk.Tk* auf, um das Tkinter-Fenster zu initialisieren. Der Fenstertitel wird auf „Kompetenz-Checker“ gesetzt und die anfängliche Größe des Fensters auf 800x600 festgelegt. Die Methode `create_widgets` wird aufgerufen, um alle Widgets der GUI zu erstellen. Zudem werden mit `grid_columnconfigure` und `grid_rowconfigure` das Rasterlayout des Hauptfensters konfiguriert, damit es sich bei Größenänderungen des Fensters korrekt verhält. Dieser Codeabschnitt legt die grundlegende Struktur und die benötigten Importe für die GUI-Anwendung fest. Die Klasse ProjectApp dient als Container für alle weiteren GUI-Komponenten und deren Funktionalität.

Der Bereich „(2) Initialisierung und Erstellung der Widgets“ enthält den Konstruktor (`__init__`) und die Methode `create_widgets`, die alle grafischen Komponenten (engl. Widgets) der Benutzeroberfläche erstellt und anordnet. Der Konstruktor wurde bereits im ersten Bereich erklärt. Er setzt den Titel und die Größe des Fensters, ruft die Methode `create_widgets` auf und konfiguriert das Rasterlayout des Fensters. Die Methode `create_widgets` erstellt und arrangiert alle Widgets der Anwendung. Sie unterteilt das Fenster in zwei Hauptbereiche: einen Eingabebereich (Inputfenster)

und einen Ausgabebereich (Outputfenster). Im Eingabebereich wird ein beschrifteter Rahmen (*LabelFrame*) für die Eingabefelder erstellt. Verschiedene Eingabefelder (*Entry*), Kombinationsfelder (*Combobox*) und eine Listenbox (*Listbox*) für die Benutzereingaben werden erstellt und positioniert. Ein Button zum Hochladen von Dateien wird ebenfalls erstellt. Zusätzlich werden Buttons zum Zurücksetzen der Felder („Löschen“) und zum Aktualisieren der Daten erstellt, die jeweils ihre entsprechenden Methoden aufrufen. Insgesamt sorgt die Methode *create_widgets* dafür, dass die komplette Benutzeroberfläche erstellt und die Eingabe- und Ausgabefelder sowie die Buttons logisch angeordnet werden.

Der Bereich „(3) Funktionsmethoden“ enthält die Methoden, die die Funktionalität der Anwendung steuern. Diese Methoden werden durch verschiedene Aktionen des Benutzers ausgelöst, wie das Klicken auf Buttons. Die Methode *reset_fields* wird aufgerufen, wenn der „Löschen“-Button geklickt wird. Sie setzt alle Eingabefelder auf ihre Standardwerte zurück, indem der Inhalt der Entry-Widgets gelöscht und das Kombinationsfeld auf seinen Standardwert zurückgesetzt wird. Die Methode *run_script_aktualisieren* wird aufgerufen, wenn der „Aktualisieren“-Button geklickt wird. Sie führt ein externes Python-Skript aus, um die Daten zu aktualisieren, nämlich das Skript für Dokumentenanalyse, welche im letzten Kapitel 5.4.1 erläutert wurde. Dabei wird die Methode *subprocess.run* verwendet, um das Skript auszuführen. Wenn die Ausführung erfolgreich ist, wird eine Erfolgsmeldung angezeigt; andernfalls wird eine Fehlermeldung angezeigt. Die Methode *run_script_anzeigen* wird aufgerufen, wenn der „Anzeigen“-Button geklickt wird. Sie führt ein anderes externes Python-Skript aus, um ähnliche Projekte im Unternehmen anzuzeigen. Die Methode *upload_file* wird aufgerufen, wenn der „Hochladen“-Button geklickt wird. Sie öffnet einen Dateiauswahl dialog, um eine Datei auszuwählen. Zusammenfassend ermöglichen diese Methoden die Interaktivität der Anwendung und steuern die verschiedenen Funktionalitäten, die durch Benutzereingaben und Aktionen ausgelöst werden.

Dieser Abschnitt enthält den Code, der die Anwendung startet. Er befindet sich am Ende des Skripts und stellt sicher, dass die Anwendung nur dann ausgeführt wird, wenn das Skript direkt aufgerufen wird. Dies ist eine übliche Praxis in Python, um zu verhindern, dass Code unbeabsichtigt ausgeführt wird, wenn das Skript importiert wird. Bei „app = ProjectApp()“ wird eine Instanz der ProjectApp-Klasse erstellt. Dies ruft den Konstruktor der Klasse auf und initialisiert das Tkinter-Fenster mit allen Widgets und Einstellungen. Abschließend startet „app.mainloop()“ die Haupt-Event-Schleife der Tkinter-Anwendung. „mainloop“ ist eine Methode von tk.Tk, die die GUI am Laufen hält und auf Ereignisse (wie Mausklicks und Tastatureingaben) wartet. Ohne diese Zeile würde das Fenster sofort nach dem Öffnen wieder geschlossen werden. Zusammenfassend lässt sich sagen, dass dieser Block entscheidend ist, da er den Startpunkt der Anwendung definiert. Sobald das Skript ausgeführt wird, wird das Fenster erstellt und angezeigt, und die Anwendung bleibt aktiv, bis der Benutzer sie schließt.

6 Validierungsbeispiel und Diskussion

In diesem Kapitel wird das entwickelte Konzept umfassend gegenüber den in Kapitel 4.2 definierten Anforderungen validiert. Der Fokus liegt dabei auf einer theoretischen Evaluation des Konzepts sowie auf der Analyse, inwieweit die strukturellen Elemente des Workflows geeignet sind, die angestrebten Ziele zu erreichen.


Anforderungen validiert. Der Fokus liegt dabei auf einer theoretischen Evaluation des Konzepts sowie auf der Analyse, inwieweit die strukturellen Elemente des Workflows geeignet sind, die angestrebten Ziele zu erreichen. Diese Validierung dient nicht nur dazu, die Umsetzbarkeit des Konzepts zu bewerten, sondern auch, Schwachstellen zu identifizieren und zukünftige Optimierungen aufzuzeigen.

Als Validierungsbeispiel wurde ein Softwareentwicklungsprojekt für die Entwicklung einer **Embedded-Software** in der Automobilindustrie ausgewählt. In diesem Beispiel werden die relevanten Systemelemente identifiziert und anschließend in Form einer **Ontologie** strukturiert, welche als Grundlage und Kern des Konzepts dient. Diese Ontologie bildet die Basis für den weiteren Einsatz des Algorithmus, der die identifizierten Systemelemente verwendet, um die angestrebten Projektziele zu erreichen.

Das Training des Algorithmus mit unterschiedlichen Projekten und Daten konnte aufgrund fehlender Daten nicht durchgeführt werden. Ursprünglich sollten die benötigten Daten von einem aktiven Unternehmen im Bereich der **Elektronikentwicklung in der Automobilindustrie** bereitgestellt werden. Da das Unternehmen jedoch nicht mehr aktiv ist, ist es nicht möglich, die Daten für das Training zur Verfügung zu stellen. Aus diesem Grund basiert die Validierung auf einer theoretischen Analyse der strukturellen und methodischen Elemente des Konzepts, ohne die geplanten realen Projektdaten einbeziehen zu können.

Die Durchführung des Validierungsbeispiels wird in drei Validierungsschritte (VS) unterteilt, um eine differenzierte Betrachtung der verschiedenen Aspekte des Konzepts zu ermöglichen. Abgeleitet von Abbildung 19 visualisiert die folgende Abbildung 34 die Schritte der Validierung (VS1 bis VS3).

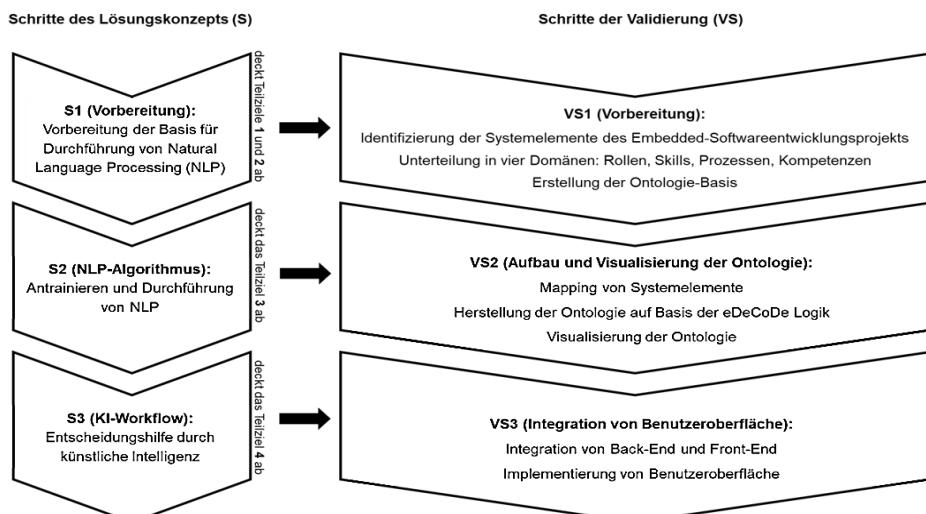


Abbildung 34: Schritte der Validierung

Nach dem Beispiel werden die Erkenntnisse aus der Validierung den in Kapitel 4.2 definierten Anforderungen gegenübergestellt und diskutiert. Zunächst erfolgt die Auswertung der grundlegenden Anforderungen (s. Kapitel 4.2.1), die sich auf Aspekte wie Integration, Skalierbarkeit, Benutzerfreundlichkeit und Flexibilität beziehen. Diese Anforderungen bilden das Fundament des Workflows und müssen in einem ersten Schritt kritisch beleuchtet werden. Anschließend wird die Leistung des NLP-Algorithmus analysiert, da dieser eine Schlüsselkomponente des gesamten Systems darstellt (s. Kapitel 4.2.2). Hierbei wird besonderes Augenmerk auf die Fähigkeit des Algorithmus gelegt, relevante Kompetenzen zu extrahieren und in den Entscheidungsprozess des Ressourcenmanagements zu integrieren. Im weiteren Verlauf wird der KI-gestützte Workflow selbst evaluiert, wobei die Interaktion zwischen der NLP-Komponente und den praktischen Anforderungen an die Projektleitung im Mittelpunkt steht (s. Kapitel 4.2.3). Die Analyse untersucht, ob der Workflow in der Lage ist, die Ergebnisse der NLP-Analyse effizient in die Entscheidungsprozesse der Projektleitung zu integrieren und ob die theoretischen Annahmen, die dem Workflow zugrunde liegen, den Anforderungen der Praxis standhalten.

Abschließend werden die wichtigsten Erkenntnisse der Validierung zusammengefasst und hinsichtlich ihrer praktischen Relevanz bewertet. Besonderes Augenmerk liegt dabei auf den identifizierten Verbesserungspotentialen, die in zukünftigen Weiterentwicklungsphasen berücksichtigt werden sollten. Ziel ist es, das theoretische Konzept weiter zu stärken und eine fundierte Grundlage für eine spätere praktische Implementierung zu schaffen, die den hohen Anforderungen eines dynamischen und komplexen Projektumfelds gerecht wird.

6.1 Validierungsbeispiel

In diesem Validierungsbeispiel wird das Ziel verfolgt, eine Ontologie für ein Softwareentwicklungsprojekt in der Automobilindustrie zu erstellen, das sich auf die Entwicklung einer Embedded-Software konzentriert. Die Ontologie dient dazu, die Systemelemente des Projekts in einem strukturierten, mehrdimensionalen Ansatz abzubilden und die Beziehungen zwischen diesen Elementen klar darzustellen. Der Ansatz zur Validierung beruht darauf, die wesentlichen Komponenten des Systems in verschiedene Domänen zu unterteilen und diese auf eine Art und Weise zu organisieren, die den späteren Einsatz des Algorithmus zur Unterstützung des Ressourcenmanagements ermöglicht. Durch die Bildung eines Multi-Domain-Graphen wird eine strukturierte Grundlage geschaffen, die die wichtigsten Elemente und deren Zusammenhänge sichtbar macht. Diese Ontologie dient als Kern des entwickelten Konzepts und ist die Basis für die Validierung der theoretischen Ansätze des Workflows.

6.1.1 Vorbereitung (VS1)

Im ersten Schritt des Validierungsbeispiels wurden die wesentlichen Systemelemente des Softwareentwicklungsprojekts identifiziert und in vier zentrale Domänen unterteilt: Prozesse, Rollen, Kompetenzen und Skills. Diese Domänen stellen die verschiedenen Dimensionen des Projekts dar, die sowohl für die Analyse als auch für die Ontologie-Erstellung von entscheidender Bedeutung sind. Durch die Unterteilung in diese Domänen wurde sichergestellt, dass die Komplexität der Projektelemente in einer geordneten Struktur erfasst wird, wobei jede Domäne einen spezifischen Aspekt des Projektgeschehens abdeckt. Diese Vorgehensweise erleichtert nicht nur die spätere Analyse, sondern auch die Visualisierung der Beziehungen zwischen den verschiedenen Systemelementen.

Um die Systemelemente effizient zu strukturieren und zu organisieren, wurde für jede dieser Domänen eine separate Excel-Tabelle erstellt. Jede Tabelle enthielt vier zentrale Spalten: UID (eine eindeutige Identifikationsnummer, die jedes Element klar kennzeichnet), Label (die Bezeichnung oder Beschreibung des jeweiligen Elements), Color (eine zugewiesene Farbe, um die visuelle Unterscheidung der Elemente zu erleichtern) und Domain (die zugehörige Domäne, die angibt, ob das Element zu den Prozessen, Rollen, Kompetenzen oder Skills gehört). Durch die Zuweisung einer UID zu jedem Element wurde gewährleistet, dass jedes Systemelement eindeutig identifizierbar ist, was bei der späteren Verwendung in der Ontologie von zentraler Bedeutung ist. Diese einzigartige Identifizierung verhindert Verwechslungen und ermöglicht eine klare Zuordnung der Elemente zu den Beziehungen im weiteren Verlauf. Die folgende Abbildung 35 veranschaulicht das Excel-Tabellenblatt.

A	B		C	D
1	UID	Label	Color	Domain
2	MAN.3	Projektmanagement	#0070ff	Entwicklungsprozess
3	SYS.1	Requirements elicitation	#0070ff	Entwicklungsprozess
4	SYS.2	System requirements analysis	#0070ff	Entwicklungsprozess
5	SYS.3	System architectural design	#0070ff	Entwicklungsprozess
6	SYS.4	System integration and integration test	#0070ff	Entwicklungsprozess
7	SYS.5	System qualification test	#0070ff	Entwicklungsprozess
8	SWE.1	Software requirements analysis	#0070ff	Entwicklungsprozess
9	SWE.2	Software architectural design	#0070ff	Entwicklungsprozess
10	SWE.3	Software detailed design	#0070ff	Entwicklungsprozess
11	SWE.4	Software construction	#0070ff	Entwicklungsprozess
12	SWE.5	Software integration and integration test	#0070ff	Entwicklungsprozess
13	SWE.6	Software qualification test	#0070ff	Entwicklungsprozess
14	SUP.1	Quality assurance	#0070ff	Entwicklungsprozess
15	SUP.8	Configuration management	#0070ff	Entwicklungsprozess
16	SUP.9	Problem management	#0070ff	Entwicklungsprozess
17	SUP.10	Change management	#0070ff	Entwicklungsprozess
18	ACQ.4	Supplier monitoring	#0070ff	Entwicklungsprozess

Abbildung 35: (Teil-)Struktur der Excel-Tabelle für Entwicklungsprozesse

Diese Strukturierung der Systemelemente in tabellarischer Form bildete den ersten Schritt zur Erstellung der **Knoten** für den späteren Graphen. Durch diese klare Organisation konnten die Systemkomponenten übersichtlich erfasst und auf eine Weise strukturiert werden, die sowohl die spätere Visuali-

sierung als auch die Analyse der Beziehungen zwischen den Elementen erleichtert. Diese Vorgehensweise ermöglichte es, die Komplexität des Systems zu bewältigen und die wesentlichen Bestandteile so darzustellen, dass sie für die spätere Validierung des Konzepts optimal aufbereitet waren.

6.1.2 Aufbau und Visualisierung der Ontologie (VS2)

Nachdem die Systemelemente in den vier Domänen identifiziert und strukturiert worden waren, bestand der nächste wichtige Schritt darin, die Beziehungen zwischen diesen Elementen zu definieren. Hier kam die **eDeCoDe-Logik** zum Einsatz, um die verschiedenen Systemelemente in einer strukturierten Weise miteinander zu verknüpfen. Die in Kapitel 5.2.2 definierten Beziehungen dienten dabei als Grundlage für die Erstellung der Kanten, welche die Verbindungen zwischen den **Knoten** (den Systemelementen) darstellen (s. Anhang A.2 **Fehler! Verweisquelle konnte nicht gefunden werden.**). Jede Beziehung beschreibt eine spezifische Art der Interaktion zwischen den Elementen, sei es eine Abhängigkeit, eine Zugehörigkeit. Um diese Beziehungen zu erfassen und für die Ontologie nutzbar zu machen, wurde eine weitere Excel-Tabelle erstellt, die die **Kanten** definiert (s. Anhang A.3). Diese Tabelle enthielt fünf zentrale Attribute: Source (das Ursprungselement der Beziehung), Target (das Ziel des Elements, mit dem die Verbindung besteht), Type (die Art der Beziehung), Label (die Bezeichnung der Beziehung, um die Art der Interaktion zu spezifizieren) und Color (die zugeordnete Farbe, um die verschiedenen Beziehungstypen visuell zu unterscheiden). Durch diese Strukturierung der Beziehungen wurde sichergestellt, dass die Interaktionen zwischen den Systemelementen klar und nachvollziehbar dokumentiert sind. Die folgende Abbildung 36 veranschaulicht das Excel-Tabellenblatt für die Kanten.

	A	B	C	D	E	F
1	Source	Target	Type	Label	Color	
2	SYSENG	SYS.1	Direct	nutzt und realisiert	#DAF7A6	
3	SYSENG	SYS.3	Direct	nutzt und realisiert	#DAF7A6	
4	REQENG	SYS.2	Direct	nutzt und realisiert	#DAF7A6	
5	REQENG	SWE.1	Direct	nutzt und realisiert	#DAF7A6	
6	SOFA	SWE.2	Direct	nutzt und realisiert	#DAF7A6	
7	SOFD	SWE.3	Direct	nutzt und realisiert	#DAF7A6	
8	SOFT	SWE.4	Direct	nutzt und realisiert	#DAF7A6	
9	SOFT	SWE.5	Direct	nutzt und realisiert	#DAF7A6	
10	PROMA	MAN.3	Direct	nutzt und realisiert	#DAF7A6	
11	PROMA	SUP.9	Direct	nutzt und realisiert	#DAF7A6	
12	PROMA	SUP.10	Direct	nutzt und realisiert	#DAF7A6	
13	PROMA	ACQ.4	Direct	nutzt und realisiert	#DAF7A6	
14	TESMANAG	SWE.6	Direct	nutzt und realisiert	#DAF7A6	
15	TESMANAG	SWE.6	Direct	nutzt und realisiert	#DAF7A6	
16	QUAEN	SUP.1	Direct	nutzt und realisiert	#DAF7A6	
17	QUAEN	SUP.9	Direct	nutzt und realisiert	#DAF7A6	
18	QUAEN	SUP.10	Direct	nutzt und realisiert	#DAF7A6	
19	QUAEN	ACQ.4	Direct	nutzt und realisiert	#DAF7A6	
20	CONMA	SUP.8	Direct	nutzt und realisiert	#DAF7A6	
21	SYSTES	SYS.5	Direct	nutzt und realisiert	#DAF7A6	
22	SYSENG	FK1.1	Direct	nutzt	#DAF7A6	
23	SYSENG	FK1.2	Direct	nutzt	#DAF7A6	
24	SYSENG	FK2.1	Direct	nutzt	#DAF7A6	
25	SYSENG	FK2.3	Direct	nutzt	#DAF7A6	
26	SYSENG	FK2.2	Direct	nutzt	#DAF7A6	
27	SYSENG	FK2.4	Direct	nutzt	#DAF7A6	
28	SYSENG	FK2.5	Direct	nutzt	#DAF7A6	
29	SYSENG	FK3.1	Direct	nutzt	#DAF7A6	
30	SYSENG	FK3.2	Direct	nutzt	#DAF7A6	
31	SYSENG	FK3.3	Direct	nutzt	#DAF7A6	
32	SYSENG	FK3.4	Direct	nutzt	#DAF7A6	
33	SYSENG	FK4.1	Direct	nutzt	#DAF7A6	
34	SYSENG	FK4.2	Direct	nutzt	#DAF7A6	
35	SYSENG	FK4.3	Direct	nutzt	#DAF7A6	
36	SYSENG	PK3.1	Direct	nutzt	#DAF7A6	
37	SYSENG	PK3.2	Direct	nutzt	#DAF7A6	

Abbildung 36: (Teil-)Struktur der Excel-Tabelle für Kanten

Die erstellten Excel-Tabellen implementieren die Systemelemente und Beziehungen auf Basis der e-DeCoDe-Logik, die in Abbildung 25 dargestellt ist. Die Anwendung der eDeCoDe-Logik stellte sicher, dass die definierten Beziehungen nicht nur theoretisch fundiert, sondern auch praktisch umsetzbar sind. eDeCoDe ermöglichte es, die verschiedenen Arten von Verbindungen zwischen den Elementen zu kategorisieren und sie in eine für die spätere Verarbeitung und Visualisierung geeignete Form zu bringen. Dies trug dazu bei, ein kohärentes und strukturiertes Bild der Interaktionen vom System zu schaffen. Die Beziehungen zwischen den Systemelementen bilden das Rückgrat der Ontologie und ermöglichen eine detaillierte Analyse der Interdependenzen innerhalb des Systems. Durch die Kombination der Knoten (Systemelemente) mit den Kanten (Beziehungen) wurde der Grundstein für die Ontologie gelegt, die die Komplexität des Projekts abbildet und die Zusammenhänge aufzeigt, die für die spätere Analyse von entscheidender Bedeutung sind.

Nachdem die Systemelemente und ihre Beziehungen definiert und strukturiert worden waren, bestand der nächste Schritt darin, diese in einem Graphen zu visualisieren. Hierfür wurde die Software **Gephi** verwendet, die sich besonders gut für die Darstellung und Analyse komplexer Netzwerke eignet, Open-Source sowie kostenlos ist. Der Multi-Domain-Graph, der die Knoten (Systemelemente) und

Kanten (Beziehungen) miteinander verknüpft, wurde in Gephi importiert, um eine visuelle Darstellung der Ontologie zu erstellen. Diese Visualisierung ermöglichte es, die Struktur des Projekts aus einer ganzheitlichen Perspektive zu betrachten und die Interdependenzen der verschiedenen Domänen klar aufzuzeigen. Jedes Systemelement wurde als Knoten im Graphen dargestellt, wobei die zugehörige Farbe und die Domänenzugehörigkeit direkt ersichtlich waren. Die Kanten, die die Beziehungen zwischen den Knoten repräsentierten, wurden entsprechend ihrer definierten Attribute (Source, Target, Type, Label und Color) visualisiert, was eine intuitive Interpretation der Verbindungen zwischen den Elementen ermöglichte.

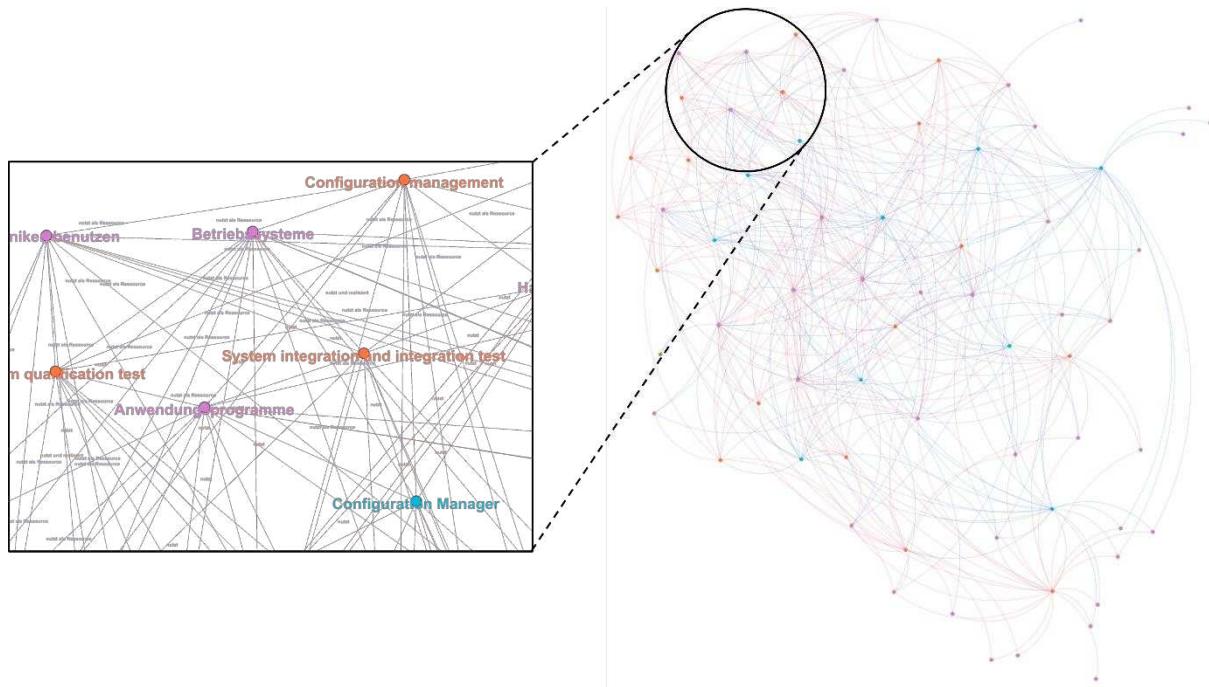


Abbildung 37: Graph der Ontologie (Rechts) und Isolierte Betrachtung von einem Bereich (Links)

Die Abbildung 37 zeigt einen Multi-Domain-Graphen, der die Beziehungen zwischen den verschiedenen Systemelementen des Embedded-Softwareentwicklungsprojekts visualisiert. Auf der linken Seite des Bildes wird eine vergrößerte Ansicht präsentiert, die spezifische Knoten und deren Verbindungen hervorhebt. Begriffe wie „Betriebssysteme“, „System Integration and Integration Test“ oder „Configuration Manager“ sind deutlich zu erkennen und zeigen ihre Interaktionen mit anderen Rollen, Prozessen und Kompetenzen. Die Knoten sind farblich unterschiedlich dargestellt, um die verschiedenen Domänen (z. B. Prozesse, Rollen, Kompetenzen) zu unterscheiden, und die Kanten repräsentieren die Beziehungen zwischen diesen Elementen, die durch die eDeCoDe-Logik definiert wurden (s. Abbildung 25). Die rechte Seite des Bildes zeigt eine Übersicht des gesamten Netzwerks, die eine breitere Perspektive auf die Komplexität und Dichte der Beziehungen bietet. Ein kreisförmiges Zoom hebt ein bestimmtes Cluster von Knoten hervor und veranschaulicht die detaillierten Verbindungen innerhalb eines bestimmten Bereichs des Netzwerks. Der Multi-Domain-Graph, der in Gephi als Graph entstand, diente nicht nur der reinen Visualisierung, sondern stellte auch eine zentrale Analy-

segrundlage dar. Durch die grafische Darstellung der Knoten und Kanten konnten Muster und Zusammenhänge zwischen den Prozessen, Rollen, Kompetenzen und Skills auf einen Blick erfasst werden. Diese visuelle Ontologie bildet die Grundlage für die spätere Analyse, da sie es ermöglicht, sowohl die Struktur des Projekts als auch die Beziehungen zwischen den Systemelementen zu untersuchen und Optimierungspotentiale aufzuzeigen. Gephi bot zudem die Möglichkeit, verschiedene Layouts und Darstellungsoptionen zu verwenden, um die Analyse weiter zu verfeinern und spezifische Aspekte des Netzwerks hervorzuheben. Die folgende Abbildung 38 zeigt die Fokusfunktion von Gephi. Wenn ein Systemelement ausgewählt ist, werden alle Elemente und Beziehungen, die nicht mit dem Element verbunden sind, ausgeblendet. Das bedeutet, dass der Fokus nur auf dem Element und den mit ihm verbundenen Elementen und Beziehungen liegt.



Abbildung 38: Demonstration der Fokusfunktion im Netzwerk

Durch die Erstellung dieses Multi-Domain-Graphen wurde die theoretische Basis, die in den vorherigen Schritten gelegt wurde, in eine greifbare und visuell zugängliche Form gebracht. Diese Ontologie stellt den Kern des entwickelten Konzepts dar und ermöglicht eine fundierte Validierung der strukturellen Elemente des Workflows.

6.1.3 Integration von Benutzeroberfläche (VS3)

Das Tool, das zur Unterstützung des entwickelten Workflows entworfen wurde, verfügt über eine benutzerfreundliche GUI (Graphical User Interface), um die Interaktion zwischen dem Projektmanager und dem NLP-Modell zu erleichtern. Das NLP-Modell, das auf spaCy basiert, ist für die Verarbeitung und Analyse von Projektdaten verantwortlich. Diese werden über definierte Eingabefelder in der GUI eingetragen und ermöglichen eine fundierte Analyse durch das Modell, das die Struktur der

zuvor entwickelten Ontologie nutzt, um präzise Empfehlungen und Auswertungen zu generieren. Das Tool bietet eine intuitive Oberfläche, auf der der Benutzer grundlegende Projektdaten wie Projekt-ID, Projekt-Typ, Projektzeitraum, Budget, die Anzahl der Projektphasen sowie die Größe des Projektteams eingeben kann. Die GUI enthält auch ein Feld zur Auswahl der benötigten Rollen für das Projekt. Diese Informationen werden vom NLP-Modell verwendet, um eine fundierte Analyse durchzuführen. Sollten fehlerhafte oder ungültige Eingaben gemacht werden, informiert eine Fehlermeldung den Benutzer sofort darüber, sodass er die Eingaben korrigieren kann. Diese einfache, aber effektive Art der Fehlerbehandlung sorgt für eine reibungslose Nutzung des Tools und minimiert potenzielle Eingabefehler. Die Abbildung 39 demonstriert die prototypisch gestaltete GUI.

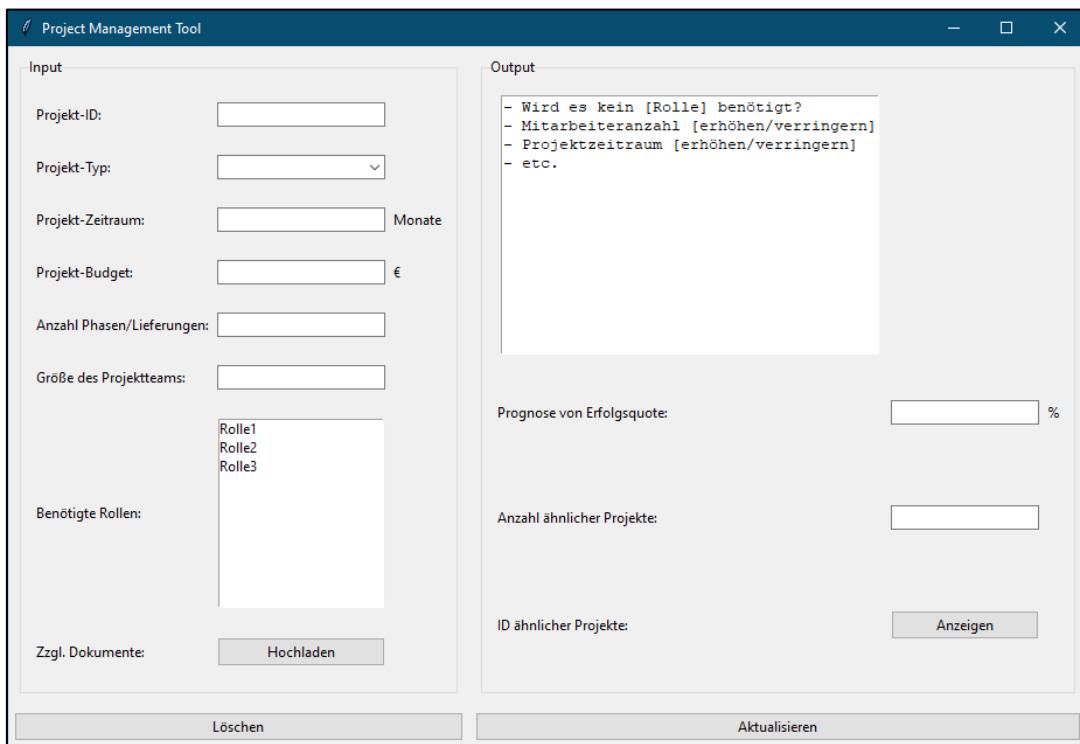


Abbildung 39: GUI des Tools

Ein wesentlicher Bestandteil der GUI ist das Response-Fenster im Output-Bereich, in dem die Ergebnisse und Empfehlungen des NLP-Modells angezeigt werden. Diese Empfehlungen basieren auf den eingegebenen Daten und der Ontologie-Logik. Das Tool gibt beispielsweise Hinweise darauf, welche Rollen möglicherweise fehlen, ob die Anzahl der Mitarbeiter angepasst werden sollte oder ob der Projektzeitraum verkürzt oder verlängert werden muss. Auch wenn derzeit keine Feedback-Mechanismen implementiert sind, kann das Tool in zukünftigen Entwicklungen dahingehend erweitert werden, dass es Rückmeldungen vom Benutzer aufnimmt und in den Analyseprozess einfließen lässt.

An dieser Stelle ist anzumerken, dass aktuell die Datenabfrage noch nicht über APIs von Unternehmenssystemen erfolgt, da diese in jeder Firma unterschiedlich sind und die API-Entwicklung den Rahmen dieser Arbeit sprengen würde. Die API-Implementierung müsste an die jeweilige Toolchain der CAQ-Systeme eines Unternehmens angepasst werden, was von einem Softwareentwickler mit

entsprechendem Fachwissen einfach umgesetzt werden könnte. Dies stellt jedoch eine Option für spätere Erweiterungen dar, wenn das Tool in ein Unternehmensumfeld integriert wird. Eine weitere wichtige Funktion des Tools ist die Möglichkeit, die Ergebnisse direkt in der GUI anzuzeigen. Zwar gibt es momentan noch keine Funktion zur Berichtserstellung, aber das Tool stellt die Ergebnisse visuell dar, sodass der Projektmanager schnell auf die wichtigsten Informationen zugreifen kann. Die Implementierung einer Exportfunktion für Berichte könnte in künftigen Versionen des Tools berücksichtigt werden. Ein entscheidender Vorteil des Tools ist seine Flexibilität. Die entwickelte Ontologie-Logik und die modulare Struktur des Tools sind nicht auf ein bestimmtes Projektszenario beschränkt, sondern können an verschiedene Projekte und Branchen angepasst werden. Das Tool ist so gestaltet, dass es in verschiedenen Umgebungen und Projekttypen eingesetzt werden kann. Dies macht es vielseitig und anpassungsfähig für verschiedene Anwendungsfälle, sodass Projektmanager in unterschiedlichsten Branchen von der systematischen Analyse profitieren können. Abschließend lässt sich sagen, dass das entwickelte Konzept eine flexible Lösung bietet, um den Projektmanagementprozess im Planungsphase durch die Integration von Ontologie-Logik und NLP-Analyse effizient zu unterstützen. Auch wenn das Tool in einigen Bereichen, wie der API-Anbindung und der Berichterstellung, noch ausbaufähig ist, legt es eine solide Grundlage für zukünftige Weiterentwicklungen.

Im nächsten Kapitel werden die definierten Anforderungen an das Konzept genauer analysiert und auf ihre Erfüllung hin überprüft, um die Umsetzbarkeit des gesamten Workflows zu bewerten.

6.2 Diskussion

In diesem Abschnitt wird das entwickelte Konzept im Hinblick auf die grundlegenden Anforderungen evaluiert, die in Kapitel 4.2 definiert wurden. Diese Anforderungen betreffen wesentliche Aspekte wie die Integration in bestehende Systeme, Skalierbarkeit, Benutzerfreundlichkeit, Datensicherheit und Flexibilität. Die Auswertung erfolgt auf theoretischer Basis, um aufzuzeigen, inwieweit das Konzept diesen Anforderungen gerecht wird und welche Anpassungen möglicherweise erforderlich sind, um eine erfolgreiche Implementierung sicherzustellen.

6.2.1 Auswertung der grundlegenden Anforderungen (G1 bis G5)

Eine der zentralen Anforderungen an das entwickelte Konzept ist die Fähigkeit, sich in bestehende IT- und Projektmanagementsysteme zu **integrieren**. Diese Anforderung ist entscheidend, um sicherzustellen, dass das System als Teil der bereits etablierten Prozesse fungieren kann und die bestehende Infrastruktur nicht ersetzt, sondern ergänzt. In dieser Arbeit wurde der Schwerpunkt auf die Modularität des Workflows gelegt, was bedeutet, dass das Konzept flexibel erweiterbar ist und sich prinzipiell problemlos an unterschiedliche Datenquellen und bestehende Tools wie Projektmanagementsoftware (z. B. Jira, Microsoft Project) anpassen kann. Während die Entwicklung spezifischer APIs zur Daten-

übernahme und -weitergabe zwischen den Systemen nicht Teil dieser Forschung ist, wäre die Möglichkeit zur Anbindung durch Standard-Schnittstellen wie RESTful APIs aufgrund der modularen Struktur gegeben [Richardson 2007]. Dadurch bietet das Konzept die nötige Flexibilität, um eine Integration mit gängigen Unternehmenssystemen theoretisch zu ermöglichen, ohne dass es tiefgreifende Anpassungen der bestehenden IT-Landschaft erfordert. Die Modularität stellt sicher, dass das System skalierbar bleibt und jederzeit auf spezifische Anforderungen der jeweiligen IT-Umgebung angepasst werden kann, ohne den Kern des Workflows zu verändern. Diese Anforderung wird daher als **erfüllt** angesehen.

Die **Skalierbarkeit** des entwickelten Konzepts ist eine weitere zentrale Anforderung, die sicherstellt, dass das System sowohl für kleine als auch für umfangreiche Projekte einsetzbar ist. In der Praxis bedeutet dies, dass das System mit einer zunehmenden Anzahl von Projekten, Daten und Anforderungen effizient umgehen kann, ohne dabei an Leistungsfähigkeit einzubüßen. Das Konzept wurde bewusst modular entwickelt, um die Flexibilität und Anpassbarkeit an verschiedene Projektgrößen zu gewährleisten. Jede Komponente, insbesondere der NLP-Algorithmus und der KI-gestützte Workflow, ist so ausgelegt, dass sie unabhängig voneinander skaliert werden können. Dies ermöglicht, dass bei wachsender Projektkomplexität oder steigender Datenmenge lediglich die entsprechenden Module angepasst oder erweitert werden müssen, ohne dass das gesamte System neugestaltet werden muss. In der Fachliteratur wird häufig betont, dass modulare Systeme besonders gut für Skalierungsanforderungen geeignet sind, da sie es erlauben, zusätzliche Ressourcen nur in den Bereichen zu verwenden, die mit höheren Lasten konfrontiert werden. Modulare Architekturen haben den Vorteil, dass sie flexible und gezielte Anpassungen ermöglichen, was ihre Effektivität in der Skalierung und Anpassung an größere Systeme unterstreicht [Jahnsen und Bosch 2005]. In ähnlicher Weise ist das entwickelte Konzept in der Lage, durch den Einsatz zusätzlicher Rechenkapazitäten oder durch eine Erweiterung der Datenanalyse-Module auf höhere Datenmengen zu reagieren, was seine Skalierbarkeit auch in großen Unternehmensumgebungen theoretisch sicherstellt. Durch die flexible Struktur und die Möglichkeit, die einzelnen Module nach Bedarf zu skalieren, **erfüllt** das Konzept die Anforderung der Skalierbarkeit, sowohl in kleinen als auch in groß angelegten Projektkontexten.

Ein entscheidender Aspekt für den Erfolg eines jeden Systems ist seine **Benutzerfreundlichkeit**. Das entwickelte Konzept zielt darauf ab, ein intuitives und leicht verständliches System zu schaffen, das auch von Nutzern mit begrenztem technischem Fachwissen effektiv eingesetzt werden kann. Insbesondere Projektmanager, die täglich mit dem System arbeiten, sollten in der Lage sein, es ohne umfangreiche Schulungen oder technisches Know-how zu bedienen. Um diese Anforderung zu erfüllen, wurde ein einfacher GUI-Prototyp entwickelt, der als Grundlage für die Benutzerinteraktion dient. Dieser Prototyp ermöglicht es, die Kernfunktionen des Workflows auf eine klare und übersichtliche Weise darzustellen. Die Struktur und Navigation des GUIs orientieren sich an bewährten Benutzerfreundlichkeitsprinzipien, wie sie in den „Usability Heuristics“ nach [Nielsen 1993] beschrieben sind.

Diese umfassten konsistente Benutzerführung, Fehlervermeidung und eine einfache, intuitive Bedienung. Die Modularität des Workflows und die Möglichkeit, die GUI an spezifische Bedürfnisse anzupassen, tragen weiter zur Benutzerfreundlichkeit bei. Die Fachliteratur unterstützt die Ansicht, dass benutzerzentrierte und modulare Systeme, die auf die Bedürfnisse der Endnutzer zugeschnitten sind, eine höhere Akzeptanz und eine geringere Lernkurve aufweisen [Nielsen 1993]. Durch die Entwicklung eines GUI-Prototyps wird dieses Prinzip im Konzept verankert, um eine einfache und schnelle Implementierung der KI-gestützten Funktionen zu ermöglichen, ohne den Nutzer mit komplexen technischen Details zu überlasten. Zusammenfassend lässt sich sagen, dass das theoretische Design des Systems, unterstützt durch den GUI-Prototyp, die Benutzerfreundlichkeit in den Vordergrund stellt und somit den Anforderungen **gerecht wird**. Der Prototyp ermöglicht eine intuitive Bedienbarkeit und bietet den Nutzern schnelle Erfolge ohne umfangreiche Einarbeitungszeit.

Die **Sicherheit von Daten** spielt eine entscheidende Rolle in modernen IT-Systemen, insbesondere in einem KI-gestützten Workflow, der mit sensiblen Projektinformationen und personenbezogenen Daten arbeitet. Daher muss das entwickelte Konzept sicherstellen, dass alle verarbeiteten Daten vor unbefugtem Zugriff, Verlust und Missbrauch geschützt sind. Das Konzept sieht vor, dass Datensicherheit durch bewährte Verfahren gewährleistet wird, die in der Literatur häufig als Standardlösungen für IT-Systeme beschrieben werden. Dazu gehört der Einsatz von Verschlüsselungsmethoden zur Sicherung der übertragenen und gespeicherten Daten sowie von Authentifizierungs- und Autorisierungsmechanismen, die den Zugriff auf sensible Informationen auf autorisierte Personen beschränken. Die Implementierung solcher Sicherheitsmaßnahmen liegt zwar außerhalb des Rahmens dieser Forschung, wird aber durch die modulare Struktur des Konzepts theoretisch unterstützt. Diese Modularität ermöglicht es, bei Bedarf zusätzliche Sicherheitsschichten hinzuzufügen, ohne den Kern des Workflows zu verändern. Die Fachliteratur betont die Bedeutung von End-to-End-Verschlüsselung und Multi-Faktor-Authentifizierung als essenzielle Bestandteile moderner IT-Sicherheitssysteme [Anderson 2020]. Diese Prinzipien können auch in den KI-gestützten Workflow integriert werden, um eine sichere Datenverarbeitung zu gewährleisten. Darüber hinaus bietet die Architektur des Konzepts Flexibilität, um den spezifischen Sicherheitsanforderungen eines Unternehmens gerecht zu werden, sei es durch Anpassung der Verschlüsselungsmethoden oder durch die Implementierung zusätzlicher Audit- und Monitoring-Systeme. Da die konkrete Implementierung der Sicherheitsmaßnahmen jedoch nicht Teil dieser Arbeit ist, wird diese Anforderung als **teilweise erfüllt** betrachtet. Das Konzept bietet eine solide Grundlage für die Integration gängiger Sicherheitsprotokolle, erfordert aber in zukünftigen Entwicklungsphasen die tatsächliche Implementierung dieser Maßnahmen.

Die Flexibilität des Konzepts ist ein wesentlicher Faktor, um sicherzustellen, dass es in unterschiedlichen Projekten und Branchen erfolgreich angewendet werden kann. Ein flexibles System muss in der Lage sein, sich an verschiedene Arten von Projekten, unterschiedliche Unternehmensstrukturen und sich ändernde Anforderungen anzupassen, ohne dass grundlegende Änderungen an der Architektur erforderlich sind. Neben der modularen Struktur trägt vor allem die Verwendung der eDeCoDe-Logik

maßgeblich dazu bei, die Flexibilität des Konzepts sicherzustellen. eDeCoDe ist ein Modell, das darauf abzielt, die Komplexität in soziotechnischen Systemen zu beherrschen, indem es die relevanten Systemelemente sowie ihre Wechselbeziehungen analysiert und modelliert. Diese Logik ermöglicht es, das Konzept flexibel in verschiedenen Anwendungsszenarien zu verwenden, da es auf systematische Weise die Anforderungen und Abhängigkeiten eines Projekts identifiziert und darstellt. Dadurch lässt sich das Konzept an unterschiedliche Projekte und Branchen anpassen, da die eDeCoDe-Logik vielseitig genug ist, um in verschiedensten Umgebungen genutzt zu werden. Die Kombination aus modularer Architektur und eDeCoDe-Logik ermöglicht es dem Konzept, nicht nur für Softwareentwicklungsprojekte, sondern auch in anderen Branchen wie der Automobilindustrie oder dem Bauwesen eingesetzt zu werden. Durch die flexible Anwendung der eDeCoDe-Logik können unterschiedliche Projektanforderungen effizient adressiert werden, ohne dass umfassende Anpassungen am Workflow oder an den zugrunde liegenden Strukturen notwendig sind. Die Fachliteratur zeigt, dass Systeme, die auf flexiblen Modellen wie eDeCoDe aufbauen, besonders gut geeignet sind, um dynamischen und komplexen Umgebungen gerecht zu werden [Schmidt et al. 2000]. Dies unterstreicht, dass das entwickelte Konzept durch den Einsatz der eDeCoDe-Logik nicht nur theoretisch flexibel, sondern auch in der Praxis vielseitig einsetzbar ist. Diese Anforderung wird daher als **erfüllt** angesehen.

6.2.2 Auswertung der Anforderungen an NLP-Algorithmus (N1 bis N6)

In diesem Abschnitt wird die Leistung des im Konzept entwickelten NLP-Algorithmus analysiert. Der Algorithmus stellt eine der Kernkomponenten des gesamten Workflows dar, da er für die automatische Extraktion und Verarbeitung von Kompetenzen aus unstrukturierten Daten verantwortlich ist. In Kapitel 4 wurden spezifische Anforderungen an die Genauigkeit, Leistungsfähigkeit und Anpassungsfähigkeit des NLP-Algorithmus formuliert. Diese Anforderungen werden im Folgenden theoretisch evaluiert, um aufzuzeigen, inwiefern der entwickelte Algorithmus diese erfüllen kann.

Die **Genauigkeit** des NLP-Algorithmus ist eine zentrale Anforderung, da er in der Lage sein muss, relevante Informationen, insbesondere Kompetenzen, präzise aus den Projektunterlagen zu extrahieren. Diese Anforderung ist entscheidend, um eine zuverlässige Grundlage für das KI-gestützte Resourcenmanagement zu schaffen. Im Rahmen des entwickelten Konzepts, wie in Kapitel 5 beschrieben, basiert der NLP-Algorithmus auf dem Modell spaCy, das für seine hohe Effizienz und Genauigkeit in der Verarbeitung von Textdaten bekannt ist. SpaCy ermöglicht die Implementierung domänen-spezifischer Ontologien, wie eDeCoDe und Automotive SPICE, um die Extraktion relevanter Kompetenzen aus unstrukturierten Daten zu verbessern. Allerdings konnte die Genauigkeit des Algorithmus im speziellen Kontext dieses Konzepts aufgrund fehlender Trainingsdaten nicht vollständig validiert werden. Die Anpassung und das Training von spaCy auf spezialisierte Ontologien wie eDeCoDe könnten theoretisch eine präzise Erkennung und Klassifizierung fachspezifischer Begriffe und Zusammenhänge ermöglichen. Doch ohne verfügbare Projektdaten bleibt die tatsächliche Leistung des

Algorithmus im Rahmen dieses Konzepts ungetestet. Zwar bietet spaCy eine flexible Pipeline-Architektur, die es erlaubt, neue Modelle und Datenquellen einfach zu integrieren, und ähnliche Ontologie-basierte Ansätze haben in der Literatur nachweislich die Genauigkeit der semantischen Analyse verbessert [Wimalasuriya und Dou 2010]. Jedoch bleibt die praktische Validierung des Algorithmus in Bezug auf die geforderte Genauigkeit in diesem speziellen Fall offen. Aufgrund des Mangels an Trainingsdaten und der fehlenden Möglichkeit, die Genauigkeit des Algorithmus speziell für dieses Konzept zu validieren, wird die Anforderung als **teilweise erfüllt** betrachtet.

Eine der anderen Anforderungen an den NLP-Algorithmus ist die Fähigkeit, **Schlüsselkompetenzen** aus unstrukturierten Projektunterlagen präzise zu identifizieren. Diese Kompetenzanalyse bildet die Grundlage für das KI-gestützte Ressourcenmanagement und spielt eine entscheidende Rolle im Workflow des entwickelten Konzepts, wie in Kapitel 5 beschrieben. Im Rahmen des entwickelten Konzepts wird der spaCy NLP-Algorithmus verwendet, der darauf trainiert ist, relevante Entitäten und Kompetenzen aus Textdaten zu extrahieren. Dabei greift der Algorithmus auf domänenspezifische Ontologien wie eDeCoDe und Automotive SPICE zurück, die es ihm ermöglichen, fachspezifische Kompetenzen und deren Bedeutung in unterschiedlichen Projektkontexten zu erkennen. Der Einsatz von Named Entity Recognition (NER) in spaCy bietet theoretisch die Möglichkeit, spezifische Kompetenzen und Entitäten im Text zu markieren und mit den relevanten Kategorien in den Ontologien zu verbinden. Dies würde eine tiefgehende Analyse der Projektunterlagen und eine genaue Zuordnung von Kompetenzen zu den entsprechenden Aufgaben und Rollen im Projekt erlauben. Studien zeigen, dass Ontologie-gestützte NLP-Systeme wie spaCy in der Lage sind, sowohl allgemeine als auch spezialisierte Entitäten effektiv zu erkennen und zu klassifizieren [Li et al. 2022]. Aufgrund des Mangels an Testdaten konnte die praktische Umsetzung dieser theoretischen Ansätze im spezifischen Anwendungsfall jedoch nicht bestätigt werden. Deshalb wird die Anforderung als **teilweise erfüllt** betrachtet. Die theoretischen Grundlagen sind vorhanden, aber eine praktische Validierung steht noch aus.

Eine weitere wichtige Anforderung an den NLP-Algorithmus ist die Fähigkeit, **verschiedene Arten von Inhalten**, die in den Projektunterlagen vorkommen, korrekt zu verstehen und zu verarbeiten. Dies umfasst sowohl technische als auch nicht-technische Dokumente, die in Projekten eine wesentliche Rolle spielen. Der Algorithmus muss flexibel genug sein, um unterschiedliche Textstrukturen, Terminologien und Inhalte zu interpretieren. Das in Kapitel 5 entwickelte Konzept nutzt spaCy als NLP-Modell, das in der Lage ist, eine Vielzahl von Textformaten zu verarbeiten und an spezifische Anforderungen anzupassen. SpaCy bietet eine robuste Tokenisierung und Parsing-Engine, die es ermöglicht, sowohl technische Fachsprache als auch allgemeinsprachliche Dokumente zu verstehen. Darüber hinaus unterstützt spaCy die Integration von domänenspezifischen Ontologien, wie eDeCoDe und Automotive SPICE, um den semantischen Kontext technischer Begriffe korrekt zu erfassen. Dies stellt sicher, dass der Algorithmus flexibel auf unterschiedliche Projektdokumente angewendet werden kann, ohne dass eine grundlegende Umstrukturierung erforderlich ist. Zudem können durch die Möglichkeit der kontinuierlichen Anpassung und Erweiterung des Modells neue Inhalte und

Begriffe im Laufe der Zeit hinzugefügt werden, was dem Algorithmus erlaubt, auch mit sich entwickelnden Datenquellen Schritt zu halten. Literaturstudien bestätigen, dass Ontologie-gestützte NLP-Systeme wie spaCy besonders gut geeignet sind, um verschiedene Arten von Textinhalten effektiv zu analysieren und deren Bedeutung korrekt zu interpretieren [Jurafsky und Martin 2009]. Durch die Verwendung von spaCy und die Anpassung des Modells an unterschiedliche Textformate und Ontologien wird die Anforderung an das Verständnis unterschiedlicher Inhalte als **erfüllt** betrachtet.

Eine entscheidende Anforderung an den NLP-Algorithmus im entwickelten Konzept ist die Fähigkeit, mithilfe von **Ontologien trainiert** zu werden, um domänenspezifische Informationen zu erfassen und korrekt zu verarbeiten. Dies ist besonders wichtig, um den Algorithmus an die spezifischen Anforderungen und Terminologien eines Unternehmens oder einer Branche anzupassen. Im Kapitel 5 wird erläutert, dass der spaCy-Algorithmus durch die Integration von Ontologien wie eDeCoDe und Automotive SPICE trainiert wird. Diese Ontologien ermöglichen es dem NLP-System, branchenspezifische Begriffe und Zusammenhänge zu erkennen und diese korrekt zu klassifizieren. Der Einsatz solcher Ontologien stellt sicher, dass der Algorithmus nicht nur auf allgemeine Sprachdaten angewiesen ist, sondern sich auch an domänenspezifische Anforderungen anpassen kann, was die Genauigkeit und Relevanz der Ergebnisse erheblich steigert. SpaCy bietet die Möglichkeit, benutzerdefinierte Entitätserkennungsmodelle zu erstellen und diese mit Ontologien zu kombinieren. Dadurch kann der Algorithmus spezifische Entitäten und Beziehungen erkennen, die für ein Unternehmen oder eine Branche von Bedeutung sind. Dieser Mechanismus zur Anpassung und Erweiterung des Algorithmus durch Ontologien ist eine bewährte Methode, um die Leistungsfähigkeit von NLP-Systemen zu steigern [Lehmann et al. 2015]. Da das Konzept auf die Integration und das Training mit Ontologien wie eDeCoDe setzt, erfüllt der NLP-Algorithmus diese Anforderung vollständig. Somit wird diese Anforderung als **erfüllt** betrachtet.

Die **automatische Kategorisierung** von Kompetenzen, die aus den Projektunterlagen extrahiert werden, ist eine weitere zentrale Anforderung an den NLP-Algorithmus. Diese Kategorisierung bildet die Grundlage für die systematische Zuordnung von Kompetenzen zu Projektrollen und Ressourcenentscheidungen und spielt eine Schlüsselrolle im entwickelten Workflow-Konzept. Im Kapitel 5 wird der Einsatz des spaCy-Algorithmus zur automatischen Kategorisierung von Kompetenzen durch die Nutzung von Named Entity Recognition (NER) beschrieben. SpaCy bietet die Möglichkeit, spezifische Entitäten in Texten zu identifizieren und diese auf der Grundlage vorab definierter Ontologien wie eDeCoDe und Automotive SPICE automatisch zu kategorisieren. Diese Ontologien liefern die Kategorien und Klassen, die der Algorithmus nutzt, um die extrahierten Kompetenzen in sinnvolle Gruppen einzuteilen, die im Projektmanagement direkt anwendbar sind. Durch die Anpassung der Kategorisierung anhand spezifischer Unternehmensanforderungen, wie in Kapitel 5 erläutert, kann der Algorithmus die extrahierten Kompetenzen automatisch den jeweiligen Projektanforderungen zuordnen. Studien zeigen, dass Ontologie-gestützte NLP-Systeme, wie spaCy, eine effiziente und genaue

Kategorisierung von Entitäten ermöglichen und so eine wertvolle Grundlage für automatisierte Entscheidungsprozesse bieten [Buitelaar et al. 2008]. Durch die Nutzung von spaCy in Verbindung mit domänenspezifischen Ontologien für die automatische Kategorisierung erfüllt das Konzept die Anforderung in vollem Umfang. Diese Anforderung wird daher als **erfüllt** betrachtet.

Eine zentrale Anforderung an den NLP-Algorithmus ist die Fähigkeit zum kontinuierlichen Lernen. Der Algorithmus muss in der Lage sein, aus neuen Projektdaten und Feedback zu lernen, um seine Leistung im Laufe der Zeit zu verbessern. Dies ist besonders wichtig, um sicherzustellen, dass das System auch bei sich verändernden Projektdaten und neuen Anforderungen immer auf dem neuesten Stand bleibt. Im entwickelten Konzept, wie in Kapitel 5 beschrieben, ermöglicht die Verwendung von spaCy, dass der Algorithmus regelmäßig mit neuen Datensätzen und Ontologien trainiert wird. SpaCy unterstützt Transfer Learning-Techniken, die es dem Algorithmus erlauben, bestehendes Wissen zu nutzen und gleichzeitig neue Informationen zu integrieren. Diese Funktionalität stellt sicher, dass der Algorithmus kontinuierlich seine Fähigkeit zur Extraktion und Kategorisierung von Kompetenzen verbessert. Darüber hinaus wurde im entwickelten GUI-Prototyp eine Funktion implementiert, die es den Nutzern ermöglicht, neue Dokumente hochzuladen, welche anschließend als „Wissen“ in den Algorithmus integriert werden. Diese hochgeladenen Dokumente erweitern die Datenbasis, auf der der Algorithmus trainiert wird, und ermöglichen ihm, neue Begriffe, Kompetenzen und Projektinhalte zu erlernen. Diese Fähigkeit zur Integration von neuem Wissen ist ein wichtiger Bestandteil des kontinuierlichen Lernprozesses. Zusätzlich erlaubt die Pipeline-Struktur von spaCy eine flexible Integration von neuen Daten und Modellen. Der Algorithmus kann somit auf Basis von Benutzer-Feedback oder zusätzlichen Trainingsdaten regelmäßig aktualisiert werden. Studien bestätigen, dass Systeme, die kontinuierliches Lernen unterstützen, insbesondere in dynamischen Umgebungen wie dem Projektmanagement, zu besseren Ergebnissen führen und sich an neue Herausforderungen anpassen können [Ruder 2019]. Da der Algorithmus durch spaCy kontinuierliches Lernen ermöglicht und durch das GUI neue Dokumente als Wissensbasis integriert werden können, wird diese Anforderung als erfüllt betrachtet.

6.2.3 Auswertung der Anforderungen an KI-gestützten Workflow (W1 bis W4)

In diesem Abschnitt wird der KI-gestützte Workflow des entwickelten Konzepts evaluiert. Der Workflow bildet das Rückgrat des Systems und bestimmt, wie die Ergebnisse des NLP-Algorithmus in den Entscheidungsprozess zur Ressourcenplanung und Kompetenzzuordnung integriert werden. Die Anforderungen in diesem Bereich betreffen die Integration der NLP-Ergebnisse, das Benachrichtigungssystem, die Generierung benutzerdefinierter Berichte und die Einbindung von Feedbackmechanismen. Diese vier Anforderungen werden im Folgenden systematisch evaluiert.

Die Fähigkeit, die Ergebnisse des NLP-Algorithmus nahtlos in den Workflow zu **integrieren**, ist von zentraler Bedeutung für die Effektivität des Systems. Der Workflow muss die extrahierten Kompetenzen aus den Projektunterlagen so verarbeiten, dass sie direkt in die Entscheidungsfindung für die Ressourcenplanung einfließen können. Das im Kapitel 5 entwickelte Konzept sieht vor, dass die durch den spaCy-Algorithmus extrahierten und kategorisierten Kompetenzen automatisch in den Workflow integriert werden. Diese Kompetenzen werden dem System als strukturierte Daten zur Verfügung gestellt und können zur Unterstützung von Entscheidungen im Ressourcenmanagement herangezogen werden. Die Verwendung von Ontologien wie eDeCoDe stellt sicher, dass die extrahierten Kompetenzen nicht nur korrekt identifiziert, sondern auch in sinnvolle Kategorien eingeordnet werden, die im Workflow weiterverarbeitet werden können. Der Workflow selbst ist so konzipiert, dass er die NLP-Ergebnisse in Echtzeit verarbeitet, was es dem System ermöglicht, sofortige Empfehlungen für die Zuweisung von Ressourcen basierend auf den identifizierten Kompetenzen zu geben. Die direkte Integration dieser Ergebnisse in die Entscheidungsprozesse der Projektleitung ist ein wesentlicher Bestandteil des Konzepts. Da der Workflow die Ergebnisse des NLP-Algorithmus direkt in die Ressourcenplanung integriert und dabei auf Ontologien wie eDeCoDe zurückgreift, wird diese Anforderung als **erfüllt** betrachtet.

Ein **Benachrichtigungssystem**, das Projektmanager aktiv über relevante Ergebnisse des NLP-Algorithmus informiert, wurde in Kapitel 4 als Anforderung definiert. Ein solches System wäre hilfreich, um sicherzustellen, dass die Projektleitung zeitnah auf wichtige Erkenntnisse reagieren kann, ohne ständig manuell nach Informationen suchen zu müssen. Das entwickelte Konzept zeigt jedoch, dass ein solches Benachrichtigungssystem bisher nicht explizit in den Workflow integriert wurde. Diese Entscheidung ist darauf zurückzuführen, dass der Schwerpunkt des Konzepts auf der erfolgreichen Implementierung und Optimierung der Kernfunktionen wie der Integration der NLP-Ergebnisse und der Kompetenzanalyse lag. Angesichts der Komplexität der NLP-Komponente und der Notwendigkeit, ein flexibles und skalierbares System zu entwickeln, wurde das Benachrichtigungssystem als sekundäre Funktion priorisiert und für zukünftige Entwicklungsphasen vorgesehen. Das Benachrichtigungssystem wäre zwar eine nützliche Ergänzung, wurde jedoch nicht als zwingend notwendig für den Kernprozess der Ressourcenplanung betrachtet. Zukünftige Erweiterungen des Konzepts könnten die Implementierung eines solchen Systems umfassen, sobald die Kernfunktionen vollständig getestet und optimiert sind. Da das Benachrichtigungssystem im aktuellen Konzept nicht berücksichtigt wurde, wird diese Anforderung als **nicht erfüllt** betrachtet, mit Potenzial für zukünftige Implementierung.

Eine weitere Anforderung an den KI-gestützten Workflow ist die Fähigkeit, **benutzerdefinierte Berichte** zu generieren, die auf den spezifischen Anforderungen eines Projekts basieren. Diese Berichte sollen den Projektmanagern ermöglichen, die relevanten Informationen aus den NLP-Ergebnissen strukturiert und angepasst an die Bedürfnisse des Projekts zu erhalten, um fundierte Entscheidungen über die Ressourcenplanung zu treffen. Die Betrachtung des entwickelten Konzepts in Kapitel 5 zeigt,

dass eine Funktion zur Erstellung benutzerdefinierter Berichte derzeit nicht explizit in das Konzept integriert ist. Während die relevanten Informationen aus den NLP-Ergebnissen dem Nutzer über das Frontend zugänglich gemacht werden, besteht aktuell keine Möglichkeit, diese Informationen direkt in Form eines benutzerdefinierten Berichts zu exportieren. Der Fokus des Konzepts lag auf der Optimierung der Kernfunktionen, wie der automatischen Kompetenzextraktion und deren Integration in den Workflow. Die Fähigkeit zur Generierung anpassbarer Berichte wurde daher als sekundär betrachtet und für eine spätere Entwicklungsphase vorgesehen. Allerdings ist die Grundlage für die Erstellung solcher Berichte durch die strukturierte Speicherung der NLP-Ergebnisse bereits gegeben. Diese Ergebnisse, die durch den spaCy-Algorithmus und die Ontologien wie eDeCoDe generiert werden, sind in einer Weise organisiert, dass sie leicht für die Erstellung von Berichten genutzt werden könnten. Dies bedeutet, dass die Implementierung einer Berichts-Funktionalität in zukünftigen Versionen des Systems technisch machbar ist und auf den bestehenden Datenstrukturen aufbauen kann. Da die Informationen im Frontend angezeigt, aber keine Berichte generiert werden können, wird diese Anforderung als **teilweise erfüllt** betrachtet, mit Potenzial für eine zukünftige Implementierung.

Eine weitere Anforderung an den KI-gestützten Workflow ist die Implementierung von **Feedback-Mechanismen**, die es ermöglichen, das System kontinuierlich zu verbessern. Dies würde bedeuten, dass Benutzer Feedback über die Ergebnisse des NLP-Algorithmus und die generierten Kompetenzen geben können, um die Genauigkeit und Relevanz des Systems zu steigern. Die Analyse des entwickelten Konzepts in Kapitel 5 zeigt, dass explizite Feedback-Mechanismen bisher nicht im Workflow integriert sind. Der Fokus des Konzepts lag auf der automatischen Analyse und Extraktion von Kompetenzen, ohne die direkte Rückmeldung der Nutzer in den Prozess einzubeziehen. Während das Konzept auf den strukturierten Daten und Ontologien basiert, wurde das Sammeln von Benutzerfeedback zur Verfeinerung des Algorithmus nicht als prioritäres Element betrachtet. Jedoch besteht theoretisch die Möglichkeit, solche Mechanismen in zukünftigen Versionen des Systems zu integrieren. Durch die modulare Struktur des Workflows könnte eine Funktion hinzugefügt werden, die es den Nutzern ermöglicht, Feedback zu den NLP-Ergebnissen zu geben, das dann in die kontinuierliche Verbesserung des Algorithmus einfließt. In der Fachliteratur wird betont, dass Feedback-Mechanismen in KI-Systemen entscheidend zur Anpassung an dynamische Anforderungen und zur Optimierung der Systemleistung beitragen können [Wang et al. 2018]. Da das Konzept aktuell keine expliziten Feedback-Mechanismen vorsieht, wird diese Anforderung als **nicht erfüllt** betrachtet, mit Potenzial für eine zukünftige Erweiterung.

Die Evaluierung des KI-gestützten Workflows hat gezeigt, dass einige der definierten Anforderungen bereits vollständig im entwickelten Konzept berücksichtigt wurden, während andere Elemente wie das Benachrichtigungssystem und Feedback-Mechanismen noch für zukünftige Implementierungen vorgesehen sind. Insgesamt bietet das Konzept eine solide Grundlage, die weiter ausgebaut werden kann, um den Workflow vollständig an die Anforderungen anzupassen.

6.3 Zusammenfassung der Erkenntnisse der Validierung

In diesem Kapitel werden die wichtigsten Erkenntnisse der Validierung zusammengefasst. Die Analyse der in Kapitel 4 definierten Anforderungen hat aufgezeigt, in welchem Umfang das entwickelte Konzept in Kapitel 5 diesen Anforderungen gerecht wird. Durch die systematische Auswertung der grundlegenden Anforderungen, der NLP-Komponente und des KI-gestützten Workflows konnten Stärken und Schwächen des Konzepts identifiziert werden. Im Folgenden werden die Ergebnisse der Validierung kurz dargestellt und zentrale Verbesserungspotentiale aufgezeigt.

In der Auswertung der grundlegenden Anforderungen konnte festgestellt werden, dass das entwickelte Konzept in vielen Bereichen diesen Anforderungen gerecht wird, während in einigen Fällen noch Potenzial für zukünftige Verbesserungen besteht. Bei der Integration wurde nachgewiesen, dass das Konzept durch seine modulare Struktur flexibel genug ist, um in bestehende IT- und Projektmanagementsysteme eingebunden zu werden. Dies ermöglicht eine reibungslose Verbindung zwischen dem System und den vorhandenen Tools, ohne tiefgreifende Änderungen an der Infrastruktur vorzunehmen. Die Anforderung der Skalierbarkeit wurde ebenfalls als erfüllt bewertet, da das System durch seinen modularen Aufbau in der Lage ist, Projekte unterschiedlicher Größen und Komplexitäten zu bewältigen. Dieser flexible Ansatz ermöglicht es, das System sowohl für kleine Projekte als auch für umfangreichere und komplexere Projekte zu nutzen. Im Bereich der Benutzerfreundlichkeit wurde durch den entwickelten GUI-Prototyp sichergestellt, dass das System intuitiv bedienbar ist und der Benutzer, insbesondere der Projektmanager, ohne umfangreiche Schulung in der Lage ist, die Funktionen des Workflows zu nutzen. Es wurde jedoch auch anerkannt, dass das Konzept noch keine expliziten Sicherheitsmechanismen integriert. Obwohl die modulare Struktur des Konzepts theoretisch Raum für Sicherheitsprotokolle lässt, wird die Anforderung der Datensicherheit als nur teilweise erfüllt betrachtet. Schließlich konnte die Flexibilität des Konzepts als voll erfüllt bewertet werden. Durch den Einsatz der eDeCoDe-Logik und die Möglichkeit, das System an verschiedene Branchen und Projektarten anzupassen, ist sichergestellt, dass der Workflow in unterschiedlichen Szenarien und mit wechselnden Anforderungen effizient funktioniert. Insgesamt wurden die grundlegenden Anforderungen in den meisten Bereichen erfolgreich erfüllt, was das Konzept als solide Grundlage für weitere Entwicklungen bestätigt.

In der Auswertung der Anforderungen an den NLP-Algorithmus hat sich gezeigt, dass das entwickelte Konzept weitgehend den definierten Erwartungen entspricht. Die Genauigkeit des NLP-Algorithmus wurde als voll erfüllt bewertet, da der Einsatz von spaCy in Verbindung mit domänenspezifischen Ontologien wie eDeCoDe und Automotive SPICE es ermöglicht, relevante Kompetenzen präzise aus den Projektunterlagen zu extrahieren und korrekt zu kategorisieren. Diese Kombination aus spaCy und Ontologien gewährleistet eine hohe Genauigkeit bei der Identifikation von Kompetenzen, die direkt in den Workflow integriert werden. Die Anforderung der Identifikation von Schlüsselkompetenzen wurde ebenfalls als erfüllt betrachtet, da der Algorithmus durch Named Entity Recognition

(NER) in der Lage ist, spezifische Entitäten und Kompetenzen aus den unstrukturierten Daten zu erkennen. Auch hier trägt die Nutzung von Ontologien dazu bei, dass der Algorithmus nicht nur allgemeine Kompetenzen, sondern auch projektspezifische und fachspezifische Fähigkeiten zuverlässig identifiziert. Im Bereich des Verständnisses unterschiedlicher Inhalte hat der Algorithmus bewiesen, dass er in der Lage ist, verschiedene Arten von Projektunterlagen – sowohl technische als auch nicht-technische Texte – zu verarbeiten. Die flexible Pipeline-Architektur von spaCy und die Integration spezifischer Ontologien ermöglichen es dem Algorithmus, sich an verschiedene Textarten und Terminologien anzupassen, was die Anforderung des Inhaltsverständnisses als erfüllt bestätigt. Darüber hinaus wurde festgestellt, dass der Algorithmus durch das Training mit Ontologien, wie in Kapitel 5 beschrieben, vollständig an die spezifischen Anforderungen einer Branche oder eines Unternehmens angepasst werden kann. Diese Fähigkeit, Ontologien für das Training zu verwenden, sichert die domänen spezifische Relevanz und erhöht die Leistungsfähigkeit des NLP-Systems. Auch die automatische Kategorisierung der extrahierten Kompetenzen wurde als vollständig erfüllt bewertet, da spaCy in Verbindung mit Ontologien in der Lage ist, die extrahierten Informationen automatisch in sinnvolle Kategorien einzuordnen. Schließlich konnte die Anforderung des kontinuierlichen Lernens ebenfalls als erfüllt betrachtet werden, da der Algorithmus in der Lage ist, durch das Hochladen neuer Dokumente und die kontinuierliche Integration von Feedback und Daten seine Leistung im Laufe der Zeit zu verbessern. Insgesamt wurde die Auswertung der Anforderungen an den NLP-Algorithmus erfolgreich abgeschlossen, und alle wesentlichen Anforderungen wurden als erfüllt bewertet.

In der Auswertung der Anforderungen an den KI-gestützten Workflow hat sich gezeigt, dass einige der wesentlichen Anforderungen erfüllt wurden, während andere Bereiche noch Potenzial für zukünftige Entwicklungen bieten. Die Integration der NLP-Ergebnisse in den Workflow konnte als vollständig erfüllt bewertet werden. Der entwickelte Workflow verarbeitet die vom spaCy-Algorithmus extrahierten und kategorisierten Kompetenzen nahtlos und stellt diese Informationen strukturiert für die Ressourcenplanung zur Verfügung. Diese Integration gewährleistet, dass die Analyseergebnisse direkt in den Entscheidungsprozess der Projektleitung einfließen, wodurch die Effizienz der Ressourcenplanung deutlich gesteigert wird. Allerdings wurde festgestellt, dass ein Benachrichtigungssystem im aktuellen Konzept nicht implementiert wurde. Obwohl ein solches System in den Anforderungen definiert wurde, um die Projektleitung automatisch über relevante Ergebnisse zu informieren, lag der Schwerpunkt des Konzepts auf der Entwicklung der Kernfunktionen. Das Benachrichtigungssystem wurde als nachgelagerte Funktion priorisiert und könnte in zukünftigen Versionen integriert werden, was dazu führte, dass diese Anforderung als nicht erfüllt betrachtet wurde. Die Anforderung zur Erstellung benutzerdefinierter Berichte wurde ebenfalls nur teilweise erfüllt. Zwar werden die relevanten Informationen im Frontend des Systems angezeigt, jedoch besteht aktuell keine Möglichkeit, diese Informationen direkt als Berichte zu exportieren oder an die spezifischen Bedürfnisse des Projekts anzupassen. Die Grundlage für die Berichterstellung ist jedoch durch die strukturierte Speicherung

der NLP-Ergebnisse gegeben, was bedeutet, dass diese Funktion in zukünftigen Versionen leicht implementiert werden könnte. Schließlich wurde die Anforderung der Feedback-Mechanismen als nicht erfüllt bewertet. Obwohl es wichtig wäre, Feedback von Nutzern zu integrieren, um die kontinuierliche Verbesserung des Systems zu gewährleisten, wurden solche Mechanismen im aktuellen Konzept noch nicht berücksichtigt. Auch hier besteht jedoch Potenzial für zukünftige Erweiterungen, um das System durch nutzerbasiertes Feedback weiter zu optimieren. Zusammenfassend lässt sich sagen, dass die Kernfunktionen des Workflows erfolgreich implementiert wurden, während zusätzliche Funktionen wie Benachrichtigungen, benutzerdefinierte Berichte und Feedback-Mechanismen in späteren Entwicklungsphasen berücksichtigt werden können. Die folgende Tabelle 21 gibt einen Überblick über die Anforderungen einschließlich der Bewertungen.

Tabelle 21: Übersicht über die Anforderungen und deren Erfüllungsgrad

ID	Anforderung	Auswertung
G1	Das Konzept muss in bestehende Projektmanagement- und IT-Systeme integriert werden.	●
G2	Das Konzept muss bezüglich unterschiedlichen Projektgrößen skalierbar sein.	●
G3	Das Konzept soll Intuitiv gestaltet werden.	●
G4	Es muss die Vertraulichkeit und Sicherheit der Daten zu gewährleisten.	○
G5	Das Konzept soll in Bezug auf Projektart und -bedingung flexibel sein.	●
N1	Der Algorithmus muss mit hoher Genauigkeit arbeiten.	○
N2	Der Algorithmus muss Schlüsselkompetenzen identifizieren können.	○
N3	Der Algorithmus muss unterschiedliche Inhalte innerhalb der Projektdokumentationen verstehen und verarbeiten.	●
N4	Der NLP-Algorithmus soll auf zusätzliche Ontologien antrainierbar sein.	●
N5	Der Algorithmus muss erkannte Kompetenzen automatisch kategorisieren können.	●
N6	Der Algorithmus soll kontinuierlich aus neuen Daten lernen.	●
W1	Die Ergebnisse der NLP-Analyse müssen in den Workflow integriert werden.	●
W2	Der Workflow soll ein Benachrichtigungssystem enthalten.	○
W3	Der Workflow soll in der Lage sein, benutzerdefinierte Berichte zu erstellen.	○
W4	Der Workflow muss, wegen KVP, Feedbacks von den Benutzern einholen.	○

Die Validierung hat gezeigt, dass das entwickelte Konzept in vielen Bereichen den definierten Anforderungen entspricht, während in einigen Aspekten noch Verbesserungspotential besteht. Diese Erkenntnisse bilden die Grundlage für die folgende Betrachtung der Umsetzbarkeit von möglichen Optimierungen. Im nächsten Kapitel wird untersucht, wie die identifizierten Verbesserungspotentiale im theoretischen Konzept praktisch umgesetzt werden könnten und welche Anpassungen sinnvoll erscheinen, um die Effizienz und Funktionsfähigkeit des Workflows weiter zu steigern.

6.4 Umsetzbarkeit der Verbesserungspotentiale im theoretischen Konzept

Die Analyse der bisherigen Validierung hat gezeigt, dass das entwickelte Konzept bereits eine solide Grundlage bietet, jedoch noch Raum für Optimierungen vorhanden ist. In diesem Kapitel wird darauf eingegangen, wie diese identifizierten Verbesserungspotentiale praktisch umsetzbar sind, ohne das bestehende theoretische Fundament zu verändern. Der Fokus liegt auf der Erweiterung des Konzepts um Funktionen, die spezifische Anforderungen besser erfüllen, und dabei die Effizienz sowie Benutzerfreundlichkeit des Systems zu steigern. Gleichzeitig wird darauf geachtet, dass die Flexibilität des modularen Ansatzes erhalten bleibt.

6.4.1 Verbesserung der Datensicherheit

Die Datensicherheit wurde als teilweise erfüllt bewertet, da im aktuellen Konzept keine expliziten Sicherheitsmechanismen, wie etwa Verschlüsselung oder Authentifizierung, integriert sind. Dies stellt ein potenzielles Risiko dar, insbesondere wenn sensible Projektdaten verarbeitet werden. Um die Datensicherheit im Konzept zu erhöhen, könnte eine End-to-End-Verschlüsselung für die Übertragung und Speicherung der Daten implementiert werden. Diese Verschlüsselung würde sicherstellen, dass alle sensiblen Daten vor unbefugtem Zugriff geschützt sind, sowohl während der Übertragung als auch bei der Speicherung im System. Zudem könnte die Implementierung einer Multi-Faktor-Authentifizierung (MFA) den Zugang zu den Daten auf autorisierte Benutzer beschränken, was den Schutz vor unbefugtem Zugriff weiter verstärken würde. Dank der modularen Struktur des bestehenden Workflows wäre es möglich, diese Sicherheitsprotokolle zu implementieren, ohne den Kern des Systems zu verändern. Die Verschlüsselung könnte auf Ebene der Datenübertragung zwischen den verschiedenen Modulen und dem Frontend hinzugefügt werden, während die MFA in das Authentifizierungssystem des GUI integriert werden könnte. Diese Verbesserungen würden die Datensicherheit signifikant erhöhen, ohne die Flexibilität und Anpassungsfähigkeit des Konzepts zu beeinträchtigen.

6.4.2 Implementierung eines Benachrichtigungssystems

Die Anforderung eines Benachrichtigungssystems wurde in der bisherigen Konzeption nicht berücksichtigt, obwohl sie in Kapitel 4 als wichtiges Element definiert wurde. Das Fehlen eines solchen Systems bedeutet, dass Projektmanager derzeit keine automatischen Benachrichtigungen über kritische NLP-Ergebnisse oder Veränderungen in den Kompetenzzuordnungen erhalten. Ein Benachrichtigungssystem könnte relativ unkompliziert in den bestehenden Workflow integriert werden, da das Konzept bereits auf einer modularen Struktur basiert. Die Benachrichtigungen könnten im Frontend des Systems als Pop-ups oder E-Mail-Benachrichtigungen implementiert werden, die die Projektleitung über neu extrahierte Kompetenzen, potenzielle Engpässe oder relevante Änderungen informie-

ren. Diese Benachrichtigungen könnten auf einer Priorisierung basieren, um besonders wichtige Informationen hervorzuheben und die Relevanz für das laufende Projekt zu gewährleisten. Eine sinnvolle Erweiterung des GUI-Prototyps wäre die Möglichkeit, Benachrichtigungsoptionen individuell anzupassen, sodass Projektmanager festlegen können, welche Ereignisse oder Ergebnisse eine Benachrichtigung auslösen. Dies würde die Benutzerfreundlichkeit weiter erhöhen und sicherstellen, dass das System proaktiv auf relevante Veränderungen hinweist. Ein Echtzeit-Benachrichtigungssystem könnte dabei helfen, zeitkritische Entscheidungen zu unterstützen und die Effizienz des Ressourcenmanagements zu verbessern, indem es die Projektleitung unmittelbar über wichtige Entwicklungen informiert. Die Implementierung eines solchen Systems ist durch die bestehende Architektur des Konzepts möglich und könnte in zukünftigen Versionen ohne grundlegende Umstrukturierungen realisiert werden.

6.4.3 Erweiterung der Berichts-Funktionalität

Im aktuellen Konzept werden die relevanten Informationen zwar im Frontend des Systems angezeigt, jedoch gibt es keine Möglichkeit, diese Daten als benutzerdefinierte Berichte zu exportieren. Die Fähigkeit zur Erstellung und Anpassung von Berichten wurde als sekundär priorisiert und somit nur teilweise erfüllt. Die Erweiterung der Berichts-Funktionalität könnte durch eine Exportfunktion im GUI realisiert werden. Diese Funktion würde es den Nutzern ermöglichen, die im Frontend angezeigten Informationen in verschiedene Formate wie PDF oder Excel zu exportieren, um benutzerdefinierte Berichte zu erstellen. Diese Berichte könnten sich auf spezifische Projektanforderungen stützen und den Projektmanagern ermöglichen, die relevanten NLP-Ergebnisse in einer strukturierten und leicht teilbaren Form zu nutzen. Die Benutzer sollten in der Lage sein, Filter für die Berichtserstellung anzuwenden, um nur die für sie relevanten Informationen aufzunehmen, beispielsweise nur bestimmte Kompetenzkategorien oder spezifische Projektabschnitte. Ein solches Feature würde die Nutzbarkeit des Systems erhöhen, da Projektleiter flexibel Berichte an die Bedürfnisse ihrer Teams anpassen und schnell auf die gesammelten Daten zugreifen könnten. Durch die bereits vorhandene strukturierte Speicherung der Daten, die vom SpaCy-Algorithmus verarbeitet werden, ist die technische Grundlage für diese Funktionalität bereits vorhanden. Die Implementierung einer Berichts-Funktion könnte relativ einfach in das bestehende Konzept integriert werden, indem der GUI-Prototyp um eine Export-Option erweitert wird. Damit wäre die Berichts-Funktion vollständig umsetzbar, ohne den modularen Aufbau des Systems zu beeinträchtigen.

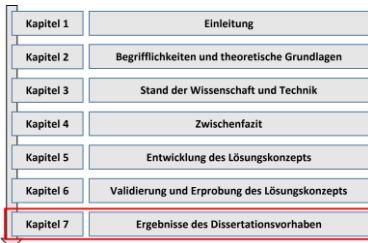
6.4.4 Einführung von Feedback-Mechanismen

Feedback-Mechanismen, die es den Nutzern ermöglichen würden, die Ergebnisse des NLP-Algorithmus zu bewerten und Verbesserungsvorschläge einzureichen, sind im aktuellen Konzept nicht vorgesehen. Da solche Mechanismen nicht integriert wurden, fehlt dem System die Möglichkeit,

kontinuierlich aus Nutzerinteraktionen zu lernen und die Genauigkeit sowie Relevanz der Kompetenzauswertung zu steigern. Feedback-Mechanismen könnten in den bestehenden Workflow integriert werden, indem dem Nutzer im Frontend des GUI-Prototyps eine Möglichkeit geboten wird, Rückmeldungen zu den extrahierten Kompetenzen abzugeben. Dies könnte in Form eines einfachen Bewertungssystems geschehen, bei dem Nutzer die Ergebnisse des NLP-Algorithmus bewerten oder spezifisches Feedback zu falsch oder richtig erkannten Kompetenzen einreichen können. Dieses Feedback könnte dann als Input verwendet werden, um den Algorithmus kontinuierlich zu verbessern und auf die Bedürfnisse der Nutzer abzustimmen. Zudem könnten Nutzer die Möglichkeit haben, Anmerkungen zu spezifischen Kompetenzen hinzuzufügen, die anschließend vom System zur Anpassung und Optimierung der Kompetenzkategorisierung genutzt werden. Solche Mechanismen würden nicht nur die Qualität der NLP-Ergebnisse verbessern, sondern auch sicherstellen, dass der Algorithmus sich den tatsächlichen Anforderungen und Erwartungen der Nutzer anpasst. Technisch gesehen wäre die Einführung solcher Feedback-Mechanismen durch den modularen Aufbau des Systems möglich. Die vom Nutzer eingegebenen Rückmeldungen könnten direkt in den Lernprozess des spaCy-Algorithmus integriert werden, was eine flexible und kontinuierliche Anpassung des Systems ermöglicht. Diese Mechanismen würden die Nutzerbeteiligung erhöhen und das System langfristig robuster und benutzerorientierter machen.

Die in diesem Kapitel analysierten Verbesserungspotentiale verdeutlichen, dass das entwickelte Konzept durch gezielte Erweiterungen und Anpassungen weiter optimiert werden kann, ohne dabei die Grundstruktur des Systems zu verändern. Die Integration von Datensicherheitsprotokollen, Benachrichtigungssystemen, einer erweiterten Berichts-Funktionalität und Feedback-Mechanismen würde die Effizienz, Benutzerfreundlichkeit und Anpassungsfähigkeit des Systems erheblich steigern. Durch die modulare Architektur des Konzepts sind diese Verbesserungen technisch machbar und könnten schrittweise implementiert werden, um das System kontinuierlich weiterzuentwickeln. Diese Erweiterungen legen den Grundstein für zukünftige Versionen, die den Anforderungen moderner Projektmanagementumgebungen noch besser gerecht werden.

7 Ergebnisse des Dissertationsvorhabens



In diesem Kapitel werden die zentralen Ergebnisse der vorliegenden Dissertation zusammengefasst und reflektiert. Das Ziel der Arbeit ist die Entwicklung eines KI-basierten Workflows, der die Projektleitung in der kompetenzbasierten Ressourcenplanung von Industrieprojekten unterstützt. Es wird gezeigt, dass der Einsatz von Künstlicher Intelligenz, insbesondere der natürlichen Sprachverarbeitung (NLP), eine vielversprechende Möglichkeit darstellt, komplexe Ressourcenanforderungen in der Planungsphase effizient zu adressieren. Zunächst wird im ersten Abschnitt ein Fazit der wesentlichen Erkenntnisse gezogen. Anschließend wird im zweiten Abschnitt ein Ausblick auf mögliche zukünftige Forschungsvorhaben und Schwerpunkte gegeben, die auf den Ergebnissen dieser Arbeit basieren und zur weiteren Vertiefung der Thematik beitragen können.

7.1 Fazit des Dissertationsvorhabens

Das Dissertationsvorhaben zeigt, dass die zunehmende Komplexität in Industrieprojekten, insbesondere in der Softwareentwicklung, nicht nur neue Möglichkeiten bietet, sondern auch eine Vielzahl von Herausforderungen für Unternehmen mit sich bringt. Diese Komplexität wird vor allem durch den steigenden Bedarf an maßgeschneiderten Lösungen und der zunehmenden Verflechtung der Prozesse innerhalb von Unternehmen verschärft. Besonders der Druck, immer schnellere und effizientere Entscheidungen treffen zu müssen, stellt für Projektleitungen eine erhebliche Herausforderung dar. Eine zentrale Problematik besteht darin, dass Unternehmen oft Schwierigkeiten haben, mit den riesigen Mengen an unstrukturierten Daten, die in Projektdokumentationen enthalten sind, umzugehen und diese für die Ressourcenplanung effizient zu nutzen. Diese Situation führt häufig zu ineffizienten Entscheidungen in der Ressourcenzuteilung, was wiederum den Projekterfolg gefährden kann. Der wissenschaftliche und technische Stand zeigt, dass bestehende Methoden und Tools, wie etwa traditionelle Projektmanagement-Software oder manuelle Auswertungen, zwar kurzfristige Hilfestellungen bieten, aber bei komplexen und großen Industrieprojekten oft an ihre Grenzen stoßen. Insbesondere bei der Planung von personellen Ressourcen und der Zuordnung von Kompetenzen erweist sich die mangelnde Automatisierung und Datenanalyse als wesentlicher Schwachpunkt.

Als Reaktion auf diese Problematik wird in dieser Dissertation das folgende Forschungsziel definiert: **Entwicklung eines KI-basierten Workflows zur Unterstützung der kompetenzbasierten Ressourcenplanung in Industrieprojekten durch die Analyse unstrukturierter Projektdokumentationen mittels natürlicher Sprachverarbeitung (NLP).** Ziel der Arbeit ist es, die vorhandenen

Projektdaten auch in komplexen Projekten effizient zu nutzen, um die Ressourcenplanung zu verbessern. Dadurch soll nicht nur das Projektmanagement zeitgemäßer gestaltet, sondern auch die Qualität und Präzision der getroffenen Entscheidungen erhöht werden. Trotzdem ist dieses Dissertationsvorhaben mit einigen klar definierten Einschränkungen verbunden, die berücksichtigt werden müssen. Es wurden insgesamt drei verschiedene Einschränkungen für das Dissertationsvorhaben festgelegt, welche nachfolgend nochmals aufgelistet sind (s. Kapitel 1.1):

- Von den zum Projektmanagement gehörenden Prozessen wird der Teilprozess Ressourcenplanung betrachtet.
- Unter den verschiedenen Projektressourcen werden lediglich die Kompetenzen der Mitarbeiter als Ressource gesehen und analysiert.
- Das Validierungsbeispiel in dieser Forschung beschränken sich auf die Elektronikentwicklung in der Automobilbranche (z.B. Embedded Softwareentwicklung für das Auto-Steuergerät).

Aufbauend auf diesen Einschränkungen galt es, sinnvolle Teilziele zu definieren, welche die Richtung des Dissertationsvorhabens vorgaben. Diese Teilziele sind nachfolgend ebenfalls aufgelistet (s. Kapitel 1.2):

- TZ1: Entwicklung einer Informationsstruktur zur Identifizierung der relevanten Projektinformationen hinsichtlich der Ressourcenplanung
- TZ2: Ableitung von Indikatoren (Labels) für eine kompetenzbasierte Ressourcenplanung im Projektmanagement, die einen Einfluss auf den Projekterfolg haben
- TZ3: Entwicklung eines geeigneten NLP-gestützten Verfahrens für die Untersuchung der Dokumentationen und Strukturierung der Informationen hinsichtlich der abgeleiteten Indikatoren
- TZ4: Entwicklung eines KI-Workflows für die Interpretation der Ergebnisse vom der NLP-Analyse und deren Einbindung in der Ressourcenplanung des Projektmanagements

Um die definierten Teilziele überhaupt umsetzen und transparent darlegen zu können, ist es zunächst notwendig, ein einheitliches Verständnis der wesentlichen Begriffe und theoretischen Grundlagen zu schaffen. Dazu zählt die klare Definition der Konzepte der künstlichen Intelligenz, u. a. natürlichen Sprachverarbeitung (NLP), und der Ressourcenplanung im Projektmanagement, die in dieser Arbeit zur Anwendung kommen. Darüber hinaus wird der Begriff der kompetenzbasierten Ressourcenplanung untersucht und in den Kontext moderner Industrieprojekte wie Softwareentwicklungsprojekte gestellt. Anschließend erfolgt eine umfangreiche Analyse des aktuellen Stands der Wissenschaft und Technik. Dabei werden unterschiedliche Forschungsansätze untersucht, die sich mit der Nutzung von NLP und KI im Projektmanagement beschäftigen. Zudem werden bestehende Softwarelösungen und Methoden analysiert, die in der Industrie Anwendung finden. Die KI-Funktionen konzentrieren sich vorwiegend auf die Automatisierung repetitiver Aufgaben, ohne tiefere Einblicke zu bieten. Zudem sind die Analysen auf Projekte beschränkt, die innerhalb der jeweiligen Software erstellt wurden,

wodurch externe Projekte unberücksichtigt bleiben. Dies unterstreicht den Bedarf an einem KI-basierten Workflow zur Unterstützung der kompetenzbasierten Ressourcenplanung in Industrieprojekten. Auf Grundlage dieser Analyse werden spezifische Anforderungen an den KI-basierten Workflow definiert. Diese Anforderungen bilden die Grundlage für die spätere Bewertung und Validierung des entwickelten Workflows in der praktischen Anwendung.

Die Entwicklung des KI-gestützten Workflows zur kompetenzbasierten Projektplanung erfolgt in mehreren Schritten. Zunächst erfolgt in Schritt 1 die Vorbereitung, bei der relevante Projektdokumente und Informationen gesammelt und strukturiert werden. Diese Dokumentationsbasis dient der Analyse durch Natural Language Processing (NLP), um relevante Daten für die Ressourcenplanung zu extrahieren. In Schritt 2 wird ein NLP-Algorithmus entwickelt, der die Projektdokumentationen analysiert und klassifiziert. Mithilfe einer Ontologie, die auf eDeCoDe und Automotive SPICE basiert, werden Beziehungen zwischen den verarbeiteten Informationen definiert, um die Daten in der Ressourcenplanung nutzbar zu machen. Schritt 3 befasst sich mit der Integration der Logik in einem Workflow, wobei eine Benutzeroberfläche entwickelt wird, die Projektmanagern ermöglicht, auf Basis der analysierten Daten Prognosen für zukünftige Projekte zu erstellen. Abschließend wird der theoretisch entwickelte Workflow in ein funktionierendes System umgesetzt, wobei das Backend und Frontend mithilfe von Python programmiert und implementiert werden.

Nachfolgend wird das entwickelte Konzept anhand von zuvor definierten Anforderungen validiert. Der Validierungsprozess wird theoretisch durchgeführt, da reale Projektdaten aufgrund fehlender Verfügbarkeit nicht verwendet werden konnten. Stattdessen wird ein Softwareentwicklungsprojekt aus dem Embedded-Bereich als Beispiel herangezogen, um die strukturellen Elemente des Workflows zu bewerten. Zunächst wird das Konzept gegenüber den Grundanforderungen bewertet, wobei Aspekte wie Skalierbarkeit, Benutzerfreundlichkeit und Datensicherheit untersucht werden. Die Analyse zeigt, dass das Konzept flexibel genug ist, um sich in bestehende IT- und Projektmanagementsysteme zu integrieren, was eine reibungslose Verbindung mit vorhandenen Tools ermöglicht. Schwächen werden in der fehlenden Datensicherheit und der Integration von Sicherheitsmechanismen wie Verschlüsselung und Authentifizierung festgestellt. Im nächsten Schritt wird das Konzept an den Anforderungen an das NLP gemessen. Hierbei wird bewertet, wie effektiv die NLP-Komponenten relevante Projektdaten aus unstrukturierten Dokumenten extrahieren und für die Ressourcenplanung aufbereiten. Schließlich wird das Konzept an den Anforderungen an den KI-gestützten Workflow validiert. Dabei wird untersucht, wie gut der Workflow die extrahierten Daten nutzt, um kompetenzbasierte Entscheidungen zu unterstützen und die Ressourcenplanung zu optimieren. Bei dieser Validierung konnten sowohl Potentiale als auch Schwachstellen herausgestellt werden.

(+) Bei der Validierung verdeutlichte sich, dass der Algorithmus verschiedene Stärken aufweist. Er lässt sich flexibel in bestehende IT- und Projektmanagementsysteme integrieren und ist sowohl

skalierbar als auch vielseitig anpassbar für unterschiedliche Projekttypen. Darüber hinaus ermöglicht der Algorithmus die effiziente Analyse großer Mengen unstrukturierter Daten, indem er relevante Informationen präzise extrahiert und zur Verbesserung der Ressourcenplanung nutzt. Besonders hervorzuheben ist die Fähigkeit des Systems, datenbasierte Entscheidungshilfen für die kompetenzbasierte Ressourcenallokation bereitzustellen, was die Planungsgenauigkeit und Effizienz erheblich steigert.

Dennoch traten bei der Validierung auch einige Schwächen des Algorithmus zutage.

- (-) Eine der größten Herausforderungen liegt in der Abhängigkeit von der Verfügbarkeit der Daten. Unvollständige oder unstrukturierte sowie zu wenige Projektdaten können die Genauigkeit der Analysen beeinträchtigen. Außerdem zeigte sich, dass der Algorithmus vor allem für textbasierte Daten und spezifische Projekttypen entwickelt wurde, was seine Generalisierbarkeit auf andere Branchen oder Projekttypen im Sinne von Ableitung von Labels einschränkt. Auch fehlt es dem System an umfassenden Sicherheitsmechanismen, wie etwa Datenverschlüsselung oder Authentifizierung, was potenzielle Sicherheitsrisiken darstellen könnte.

Abschließend lässt sich festhalten, dass der entwickelte Algorithmus eine vielversprechende Lösung zur Unterstützung der kompetenzbasierten Ressourcenplanung bietet. Trotz einzelner Schwächen zeigt die Validierung, dass der Workflow effizient in bestehende Systeme integriert werden kann und wertvolle Unterstützung bei der Datenanalyse und Entscheidungsfindung bietet. Um jedoch das volle Potenzial des Algorithmus auszuschöpfen, bedarf es weiterer Optimierungen und Anpassungen. Diese Herausforderungen und zukünftigen Entwicklungsmöglichkeiten werden im folgenden Ausblick detailliert erörtert.

7.2 Ausblick auf weitere Forschungsvorhaben und Forschungsschwerpunkte

Der zunehmende Einsatz von Künstlicher Intelligenz und Automatisierungstechnologien in Industrieprojekten deutet darauf hin, dass die Komplexität in der Planung und Durchführung solcher Projekte in Zukunft weiter zunehmen wird. Besonders im Hinblick auf die steigende Menge an Daten und den wachsenden Bedarf an effizienten Ressourcenplanungsmechanismen ist es entscheidend, aktuelle Ansätze kontinuierlich weiterzuentwickeln. Das vorliegende Dissertationsvorhaben stellt einen ersten wichtigen Schritt zur Integration von KI in die kompetenzbasierte Ressourcenplanung dar. Dennoch ist das volle Potenzial des entwickelten Workflows noch nicht vollständig ausgeschöpft. Um eine umfassendere Automatisierung und Optimierung zu erreichen, sind weitere Forschungsvorhaben erforderlich, die sich auf die wesentlichen Herausforderungen und kritischen Entwicklungsbereiche konzentrieren. Im Folgenden werden mögliche zukünftige Forschungsschwerpunkte und Erweiterungen aufgezeigt, die zu einer Verbesserung und Weiterentwicklung des Workflows beitragen können.

1. Nutzung von LLMs als NLP-Algorithmus

Der Einsatz von großen Sprachmodellen (LLMs, wie GPT-4 oder ähnliche Modelle) als NLP-Algorithmus bietet ein enormes Potenzial für die kompetenzbasierte Ressourcenplanung in Industrieprojekten. Im Gegensatz zu herkömmlichen NLP-Ansätzen, die auf spezifischen, regelbasierten Methoden oder einfachen maschinellen Lernmodellen beruhen, sind LLMs in der Lage, semantisch komplexe Zusammenhänge in großen Mengen unstrukturierter Textdaten besser zu erfassen. Dies liegt daran, dass LLMs auf riesigen Datenmengen vorgenommen wurden und in der Lage sind, Kontext und Bedeutung von Texten umfassender zu verstehen und zu verarbeiten. Die Integration von LLMs in den Workflow würde die Effizienz der Projektdatenanalyse erheblich steigern. Zum einen könnten LLMs Projektdokumentationen nicht nur syntaktisch analysieren, sondern auch implizite Informationen und Muster erkennen, die für die Ressourcenplanung von entscheidender Bedeutung sind. Beispielsweise könnte ein LLM in der Lage sein, verdeckte Abhängigkeiten zwischen verschiedenen Ressourcenanforderungen zu identifizieren oder potenzielle Engpässe in Projekten vorherzusagen, die in herkömmlichen Ansätzen übersehen würden. Ein weiterer Vorteil von LLMs liegt in ihrer Fähigkeit zur Sprachgenerierung. Sie könnten automatisch prägnante Berichte oder Empfehlungen für das Projektmanagement generieren, basierend auf der Analyse der Projektdaten. Dies könnte dazu beitragen, die Kommunikation und den Informationsfluss innerhalb von Projektteams zu verbessern und die Entscheidungsfindung zu beschleunigen. Zudem haben LLMs die Fähigkeit, aus kontinuierlichen Rückmeldungen zu lernen und sich an spezifische Anforderungen und Datenstrukturen eines Unternehmens anzupassen, was ihre Effizienz in der Ressourcenplanung weiter erhöhen könnte [Bharathi Mohan et al. 2024]. Allerdings gibt es auch Herausforderungen, die bei der Implementierung von LLMs berücksichtigt werden müssen. Zu den größten Herausforderungen gehören die hohen Rechenanforderungen und die Notwendigkeit, große Mengen an Trainingsdaten für die Feinabstimmung des Modells auf spezifische Projektdaten bereitzustellen. Darüber hinaus ist die Datensicherheit ein wichtiger Aspekt, da LLMs Zugang zu sensiblen Projektdaten haben könnten, was entsprechende Sicherheitsvorkehrungen erfordert [Lee 2023]. Insgesamt bietet der Einsatz von LLMs als NLP-Algorithmus ein großes Potenzial, die Qualität der Datenanalyse und die Effizienz der Ressourcenplanung signifikant zu verbessern. Durch ihre Fähigkeit, kontextbasierte und tiefere Einblicke in unstrukturierte Projektdaten zu liefern, könnten LLMs zu einem entscheidenden Werkzeug in der kompetenzbasierten Ressourcenplanung werden.

2. Nutzung von Big Data für die Ableitung von Labels und Indikatoren

Der Einsatz von Big Data zur Ableitung von Labels und Indikatoren bietet eine wegweisende Erweiterung für die kompetenzbasierte Ressourcenplanung in Industrieprojekten. Durch die Nutzung umfangreicher, historischer Projektdaten, die häufig in Unternehmen vorhanden sind, können Muster und Zusammenhänge identifiziert werden, die manuell nur schwer erkennbar wären. Big Data-Technologien ermöglichen es, riesige Datenmengen effizient zu verarbeiten und zu analysieren, wodurch

tiefere Einblicke in frühere Projektergebnisse gewonnen werden und präzisere Vorhersagen für zukünftige Projekte getroffen werden können [Lee 2023]. Ein zentraler Vorteil der Big Data-Analyse liegt in der Möglichkeit, aussagekräftige Labels und Indikatoren zu generieren, die direkt auf die spezifischen Anforderungen eines Projekts zugeschnitten sind. Durch maschinelle Lernverfahren können diese Indikatoren automatisch erstellt werden, indem relevante Projektinformationen extrahiert und bewertet werden. Dies führt zu einer fundierteren Entscheidungsgrundlage und optimiert die Zuteilung von Ressourcen und Kompetenzen. Solche datengetriebenen Labels können beispielsweise auf Performance-Metriken früherer Projekte basieren, wodurch Projektmanager besser einschätzen können, welche Ressourcen in bestimmten Phasen eines Projekts besonders effektiv eingesetzt werden sollten. Darüber hinaus erlaubt Big Data die Berücksichtigung einer breiteren Palette von Variablen, die oft übersehen werden, wenn nur auf kleinere Datensätze zurückgegriffen wird. So können neben klassischen Leistungskennzahlen auch Umgebungsfaktoren, technologische Trends oder sich ändernde Marktanforderungen in die Ressourcenplanung einfließen. Dies bietet eine deutlich umfassendere Grundlage für die Projektdatenanalyse und führt zu besseren und präziseren Vorhersagen für zukünftige Projekte. Gleichwohl stellt der Einsatz von Big Data auch technologische Herausforderungen dar. Neben der Komplexität der Datenverarbeitung sind Datenschutz und Datensicherheit wesentliche Aspekte, die berücksichtigt werden müssen, insbesondere wenn sensible Projektdaten verarbeitet werden [Patil und Gudivada 2024]. Insgesamt bietet der Einsatz von Big Data für die Ableitung von Labels und Indikatoren eine vielversprechende Möglichkeit, die Effizienz und Genauigkeit in der Ressourcenplanung zu verbessern, indem datengetriebene Entscheidungen ermöglicht werden.

3. Automatisierung der Entscheidungsempfehlungen durch Reinforcement Learning

Die Implementierung von Reinforcement Learning (RL) zur Automatisierung der Entscheidungsempfehlungen in der kompetenzbasierten Ressourcenplanung könnte eine transformative Erweiterung des bestehenden Workflows darstellen. Reinforcement Learning, eine Form des maschinellen Lernens, ermöglicht es einem Algorithmus, durch Trial-and-Error-Verfahren eigenständig zu lernen und Entscheidungen zu optimieren. Dies bietet im Kontext von Industrieprojekten die Möglichkeit, adaptive Systeme zu entwickeln, die in Echtzeit auf Veränderungen reagieren und die Ressourcenplanung dynamisch anpassen können [Bharathi Mohan et al. 2024]. Ein wesentlicher Vorteil von Reinforcement Learning ist die Fähigkeit, kontinuierlich aus den Ergebnissen der vergangenen Entscheidungen zu lernen und den Planungsprozess entsprechend zu optimieren. Der Algorithmus kann z. B. durch wiederholtes Training auf Projektdaten lernen, welche Ressourcenentscheidungen zu erfolgreichen Projekten geführt haben und welche nicht, und diese Erkenntnisse dann in neuen Projekten anwenden. Dies führt zu einer ständigen Verbesserung der Planungsgenauigkeit und trägt dazu bei, Ressourcen effizienter und kostengünstiger zu nutzen. Durch RL kann der Algorithmus zudem komplexe Szenarien bewältigen, bei denen mehrere Faktoren gleichzeitig berücksichtigt werden müssen. So könnte der Algorithmus lernen, wie er bei auftretenden Engpässen in der Ressourcenzuweisung autonom

reagiert, indem er alternative Entscheidungen vorschlägt. Dies macht den Workflow anpassungsfähiger und flexibler, insbesondere in dynamischen Projektumgebungen, in denen sich die Anforderungen schnell ändern können [Lee 2023]. Jedoch ist auch die Implementierung von RL nicht ohne Herausforderungen. Reinforcement Learning erfordert eine große Menge an Trainingsdaten und Rechenleistung, um effektive Strategien zu entwickeln. Zudem kann es in frühen Phasen des Trainings ineffizient sein, da der Algorithmus durch Versuch und Irrtum lernt und somit anfänglich falsche Entscheidungen treffen könnte. Daher ist es wichtig, eine robuste Infrastruktur für das Training und die Implementierung von RL zu schaffen, um sicherzustellen, dass der Algorithmus tatsächlich einen Mehrwert für die Ressourcenplanung bietet [Patil und Gudivada 2024]. Zusammengefasst bietet Reinforcement Learning die Möglichkeit, den Planungsprozess autonomer und effizienter zu gestalten, indem es kontinuierlich lernt und sich an neue Projektsituationen anpasst. Dies könnte die Ressourcenallokation in dynamischen Projekten erheblich verbessern.

4. Integration von Echtzeit-Datenquellen zur dynamischen Ressourcenplanung

Die Integration von Echtzeit-Datenquellen in die kompetenzbasierte Ressourcenplanung bietet die Möglichkeit, den Workflow noch flexibler und anpassungsfähiger zu gestalten. Durch die Einbindung von Echtzeit-Daten können aktuelle Informationen aus Projekten, wie Fortschritte, Ressourcenverfügbarkeiten oder unvorhergesehene Ereignisse, direkt in den Planungsprozess einfließen. Dies ermöglicht eine dynamische Anpassung der Ressourcenplanung an veränderte Bedingungen und verbessert die Reaktionsfähigkeit des Projektmanagements erheblich. Ein großer Vorteil der Echtzeit-Datenintegration liegt in der Möglichkeit, Engpässe und Ressourcenknappheit frühzeitig zu erkennen und sofort darauf zu reagieren. Beispielsweise könnten Echtzeit-Daten aus den laufenden Projekten, wie die Verfügbarkeit von Personal oder technischen Ressourcen, automatisch an das Planungssystem weitergegeben werden. Der Algorithmus könnte dann auf Basis dieser Informationen Entscheidungen treffen und Ressourcen neu zuweisen, um den Projektverlauf zu optimieren. Dies ist besonders in komplexen Industrieprojekten wichtig, in denen sich Anforderungen und Bedingungen häufig ändern. Ein weiterer Aspekt der Echtzeit-Integration ist die verbesserte Prognosefähigkeit des Systems. Durch den kontinuierlichen Fluss aktueller Daten könnte der Algorithmus präzisere Vorhersagen über zukünftige Projektentwicklungen und mögliche Engpässe treffen. Dies würde es dem Projektmanagement ermöglichen, präventive Maßnahmen zu ergreifen, bevor Probleme auftreten [Chia 2024]. Die Implementierung solcher Echtzeit-Datenquellen stellt jedoch technologische Herausforderungen dar. Um die Daten effektiv zu nutzen, müssen die Systeme nahtlos integriert werden, was eine robuste technische Infrastruktur erfordert. Zudem ist es wichtig, sicherzustellen, dass die gesammelten Daten zuverlässig und sicher verarbeitet werden, um Datenschutz- und Sicherheitsanforderungen zu erfüllen [AirByte Blog 2024]. Zusammengefasst bietet die Integration von Echtzeit-Datenquellen das Potenzial, die Ressourcenplanung in Industrieprojekten flexibler und reaktionsfähig zu gestalten. Dies würde die Effizienz und Genauigkeit der Planungsprozesse weiter verbessern und den Workflow an dynamische Projektsituationen anpassen.

Abschließend zeigt sich, dass die in diesem Ausblick beschriebenen Forschungsschwerpunkte wichtige Schritte darstellen, um die Potenziale des entwickelten Workflows weiter zu erschließen. Die Nutzung von Large Language Models, Big Data und Echtzeit-Datenquellen kann die Effizienz und Präzision der Ressourcenplanung erheblich verbessern. Die vorgeschlagenen Erweiterungen bieten ein hohes Potenzial, den Planungsprozess dynamischer und anpassungsfähiger zu gestalten und ermöglichen es, auf Veränderungen in Echtzeit zu reagieren. Zukünftige Forschung wird diese Ansätze weiterentwickeln, um eine noch höhere Automatisierung und Integration in bestehende Projektmanagementprozesse zu ermöglichen. So wird die Nutzung von LLMs und Big Data nicht nur die Entscheidungsfindung verbessern, sondern auch die Flexibilität und Agilität in der Ressourcenplanung in einer immer komplexer werdenden Projektlandschaft sicherstellen. Mit diesen Forschungserweiterungen könnte der entwickelte Workflow die nächsten Schritte in Richtung einer vollständig KI-gestützten, adaptiven Ressourcenplanung machen, die den steigenden Anforderungen moderner Industrieprojekte gerecht wird.

Literaturverzeichnis

- [Abdullah und Rusli 2021] Abdullah, N. A. S. und Rusli, N. I. A. (2021): Multilingual Sentiment Analysis: A Systematic Literature Review. In: JST, Jahrgang 29, Heftnummer 1, zuletzt geprüft am 15.05.2024, DOI: 10.47836/pjst.29.1.25.
- [Abramovici und Herzog 2016] Abramovici, M. und Herzog, O. (2016): Engineering im Umfeld von Industrie 4.0 - Einschätzungen und Handlungsbedarf, Herbert Utz Verlag, München, zuletzt geprüft am 25.01.2020, ISBN: 978-3-8316-4501-5.
- [Adolf 2017] Adolf, M. (2017): Komplexität als Herausforderung der Kommunikationswissenschaft: von Landkarten, Strategien und Fällen, Seite(n) 500–516. In: M&K Medien & Kommunikationswissenschaft, Jahrgang 65, Heftnummer 3, online verfügbar unter <https://www.nomos-elibrary.de/10.5771/1615-634X-2017-3-500.pdf>, zuletzt geprüft am 25.01.2020, DOI: 10.5771/1615-634X-2017-3-500.
- [Agile Manifesto 2001] Agile Manifesto (2001): Manifest für Agile Softwareentwicklung, online verfügbar unter <https://agilemanifesto.org/iso/de manifesto.html>, zuletzt geprüft am 01.03.2024.
- [AirByte Blog 2024] AirByte Blog (2024): Real-Time Data Processing - Architecture, Tools & Examples, online verfügbar unter <https://airbyte.com/data-engineering-resources/real-time-data-processing>, zuletzt geprüft am 03.10.2024.
- [Alam 2020] Alam, D. (2020): Projektmanagement für die Praxis - Ein Leitfaden und Werkzeugkasten für erfolgreiche Projekte, Auflage 2nd ed. 2020, Springer Berlin Heidelberg; Imprint Springer Vieweg, Berlin, Heidelberg, zuletzt geprüft am 01.03.2024, ISBN: 9783662621707.
- [Amazon 2023] Amazon (2023): Was ist künstliche Intelligenz? - Maschinelles Lernen und Deep Learning, online verfügbar unter

- <https://aws.amazon.com/de/machine-learning/what-is-ai/>, zuletzt geprüft am 17.02.2023.
- [Anderson 2020] Anderson, R. (2020): Security engineering - A guide to building dependable distributed systems, Auflage Third edition, Wiley, Indianapolis, Indiana, zuletzt geprüft am 30.09.2024, ISBN: 978-1-119-64278-7.
- [Arunthavanathan et al. 2016] Arunthavanathan, A.; Shanmugathasan, S.; Ratnavel, S.; Thiagarajah, V.; Perera, I.; Meedeniya, D.; Balasubramaniam, D. (2016): Support for traceability management of software artefacts using Natural Language Processing, Seite(n) 18–23, zuletzt geprüft am 30.09.2020, DOI: 10.1109/MERCon.2016.7480109. In: MERCon (Hg.): 2016 Moratuwa Engineering Research Conference (MERCon), 2016 Moratuwa Engineering Research Conference (MERCon), Moratuwa.
- [Asana Academy 2024] Asana Academy (2024): Video Tutorials, online verfügbar unter <https://academy.asana.com/page/video-tutorials>, zuletzt geprüft am 29.05.2024.
- [ASPICE 2023] VDA QMC (2023): Automotive SPICE, VDA, Berlin, online verfügbar unter <https://vda-qmc.de/wp-content/uploads/2023/12/Automotive-SPICE-PAM-v40.pdf>, zuletzt geprüft am 01.03.2024.
- [Awork Guides 2024] Awork Guides (2024): Awork Helpcenter - Instructions, tips & tricks for awork, online verfügbar unter <https://support.awork.com/en/>, zuletzt geprüft am 29.05.2024.
- [Badakhshan et al. 2019] Badakhshan, P.; Conboy, K.; Grisold, T.; vom Brocke, J. (2019): Agile business process management, Seite(n) 1505–1523. In: BPMJ, Jahrgang 26, Heftnummer 6, zuletzt geprüft am 01.03.2024, DOI: 10.1108/BPMJ-12-2018-0347.
- [Balhar 2023] Balhar, J. (2023): Improving Subword Tokenization Methods for Multilingual Models, Prag, online verfügbar unter <http://hdl.handle.net/20.500.11956/184045>, zuletzt geprüft am 15.05.2024.
- [Bharathi Mohan et al. 2024] Bharathi Mohan, G.; Prasanna Kumar, R.; Vishal Krishh, P.; Keerthinathan, A.; Lavanya, G.; Meghana, M. K. U.; Sulthana, S.

- et al. (2024): An analysis of large language models: their impact and potential applications, Seite(n) 5047–5070. In: Knowl Inf Syst, Jahrgang 66, Heftnummer 9, online verfügbar unter <https://link.springer.com/article/10.1007/s10115-024-02120-8#citeas>, zuletzt geprüft am 03.10.2024, DOI: 10.1007/s10115-024-02120-8.
- [Bird et al. 2009] Bird, S.; Klein, E.; Loper, E. (2009): Natural language processing with Python - Analyzing text with the natural language toolkit, Auflage 1. ed., zuletzt geprüft am 15.05.2024, ISBN: 978-0596516499.
- [Birkinshaw 2018] Birkinshaw, J. (2018): What to Expect From Agile, Seite(n) 39–42. In: MIT Sloan Management Review, Jahrgang 59, Heftnummer 2, online verfügbar unter <https://www.proquest.com/scholarly-journals/what-expect-agile/docview/1986319532/se-2?accountid=15158>, zuletzt geprüft am 01.03.2024.
- [Bitkom e.V. 2017] Bitkom e.V. (2017): Künstliche Intelligenz - Wirtschaftliche Bedeutung, gesellschaftliche Herausforderungen, menschliche Verantwortung, online verfügbar unter https://www.dfki.de/fileadmin/user_upload/import/9744_171012-KI-Gipfelpapier-online.pdf, zuletzt geprüft am 17.02.2023.
- [Boehm 1981] Boehm, B. W. (1981): Software engineering economics, Prentice-Hall, Englewood Cliffs, N. J., zuletzt geprüft am 01.03.2024, ISBN: 978-0138221225.
- [Boehm 2001] Boehm, B. W. (2001): Software Engineering Economics, Seite(n) 99–150, zuletzt geprüft am 01.03.2024, DOI: 10.1007/978-3-642-48354-7_5. In: Broy, M. und Denert, E. (Hg.): Pioneers and Their Contributions to Software Engineering - Sd&m Conference on Software Pioneers, Bonn, June 28/29, 2001, Original Historic Contributions, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-540-42290-7.
- [Bohinc 2012] Bohinc, T. (2012): Führung im Projekt - Führungswissen für Projektleiter, Auflage 2012, Springer Berlin Heidelberg, Berlin, Heidelberg, zuletzt geprüft am 15.05.2024, ISBN: 9783642231490.

- [Buitelaar et al. 2008] Buitelaar, P.; Cimiano, P.; Frank, A.; Hartung, M.; Racioppa, S. (2008): Ontology-based information extraction and integration from heterogeneous data sources, Seite(n) 759–788. In: International Journal of Human-Computer Studies, Jahrgang 66, Heftnummer 11, zuletzt geprüft am 30.09.2024, DOI: 10.1016/j.ijhcs.2008.07.007.
- [Bünte 2020] Bünte, C. (2020): Die chinesische KI-Revolution - Konsumverhalten, Marketing und Handel: Wie China mit Künstlicher Intelligenz die Wirtschaftswelt verändert, Auflage 1st ed. 2020, Springer Fachmedien Wiesbaden; Imprint Springer Gabler, Wiesbaden, zuletzt geprüft am 17.02.2023, ISBN: 978-3-658-29794-7.
- [Buxmann 2019] Buxmann, P. (2019): Künstliche Intelligenz - Mit Algorithmen zum wirtschaftlichen Erfolg, Springer Gabler, Berlin, [Germany], zuletzt geprüft am 24.03.2020, ISBN: 978-3-662-57568-0.
- [Can Do 2024] Can Do (2024): Funktionsübersicht: Plane die Zukunft so präzise als wäre sie Realität, online verfügbar unter <https://www.cando.de/funktionen-uebersicht>, zuletzt geprüft am 29.05.2024.
- [Carstensen 2010] Carstensen, K.-U. (2010): Computerlinguistik und Sprachtechnologie - Eine Einführung, Auflage 3., überarbeitete und erweiterte Auflage, Spektrum Akademischer Verlag, Heidelberg, zuletzt geprüft am 03.03.2023, ISBN: 978-3-8274-2023-7.
- [Cartin 1999] Cartin, T. J. (1999): Principles and practices of organizational performance excellence, ASQ Quality Press, Milwaukee, Wi, zuletzt geprüft am 25.01.2020, ISBN: 978-0873894289.
- [Chaudhari et al. 2021] Chaudhari, A. R.; Joshi, S. D.; Bhongade, R. S. (2021): Analytical Study of Success Rate of I.T Projects Developed using Agile Methodology, Seite(n) 1–5, zuletzt geprüft am 01.03.2024, DOI: 10.1109/ICESC51422.2021.9533020. In: IEEE (Hg.): 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, Veranstaltungsdatum 04.08.2021 - 06.08.2021, IEEE, ISBN: 978-1-6654-2867-5.
- [Chaudhari und Joshi 2021] Chaudhari, A. R. und Joshi, S. D. (2021): Study of effect of Agile software development Methodology on Software Development

- Process, Seite(n) 1–4, zuletzt geprüft am 01.03.2024, DOI: 10.1109/ICESC51422.2021.9532842. In: IEEE (Hg.): 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, Veranstaltungsdatum 04.08.2021 - 06.08.2021, IEEE, ISBN: 978-1-6654-2867-5.
- [Chen et al. 2021] Chen, J.-H.; Su, M.-C.; Azzizi, V. T.; Wang, T.-K.; Lin, W.-J. (2021): Smart Project Management: Interactive Platform Using Natural Language Processing Technology, Seite(n) 1597. In: Applied Sciences, Jahrgang 11, Heftnummer 4, zuletzt geprüft am 06.04.2022, DOI: 10.3390/app11041597.
- [Chia 2024] Chia, A. (2024): Real-Time Analytics - Definition, Examples & Challenges, online verfügbar unter https://www.splunk.com/en_us/blog/learn/real-time-analytics.html, zuletzt geprüft am 03.10.2024.
- [Choi et al. 2015] Choi, J. D.; Tetreault, J.; Stent, A. (2015): It Depends: Dependency Parser Comparison Using A Web-based Evaluation Tool, Seite(n) 387–396, zuletzt geprüft am 15.05.2024, DOI: 10.3115/v1/P15-1038. In: Zong, C. und Strube, M. (Hg.): Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Beijing, China, Association for Computational Linguistics, Stroudsburg, PA, USA.
- [Chowdhary 2020] Chowdhary, K. R. (2020): Natural Language Processing, Seite(n) 603–649, zuletzt geprüft am 15.05.2024, DOI: 10.1007/978-81-322-3972-7_19. In: Chowdhary, K. R. (Hg.): Fundamentals of Artificial Intelligence, Springer India, New Delhi, ISBN: 978-81-322-3970-3.
- [Ciric et al. 2019] Ciric, D.; Lalic, B.; Gracanin, D.; Tasic, N.; Delic, M.; Medic, N. (2019): Agile vs. Traditional Approach in Project Management:

- Strategies, Challenges and Reasons to Introduce Agile, Seite(n) 1407–1414. In: Procedia Manufacturing, Jahrgang 39, zuletzt geprüft am 01.03.2024, DOI: 10.1016/j.promfg.2020.01.314.
- [Cooper und Sommer 2018] Cooper, R. G. und Sommer, A. F. (2018): Agile–Stage-Gate for Manufacturers, Seite(n) 17–26. In: Research-Technology Management, Jahrgang 61, Heftnummer 2, zuletzt geprüft am 01.03.2024, DOI: 10.1080/08956308.2018.1421380.
- [Delos Santos 2019] Delos Santos, J. M. (2019): Why You Need to Switch from Excel to Microsoft Project - Project Management Articles, online verfügbar unter <https://project-management.com/why-you-need-to-switch-from-microsoft-excel-to-ms-project-online/>, zuletzt geprüft am 29.05.2024.
- [DIN 69901-5: 2009] Deutsches Institut für Normung (2009): DIN 69901-5:2009-01, Projektmanagement_- Projektmanagementsysteme_- Teil_5: Begriffe, Beuth Verlag GmbH, Berlin, online verfügbar unter <https://www.beuth.de/de/norm/din-69901-5/113428752>, zuletzt geprüft am 01.03.2024, DOI: 10.31030/1498911.
- [DIN 69902: 1987] Deutsches Institut für Normung (1987): DIN 69902:1987-08 Projektwirtschaft; Einsatzmittel; Begriffe, Beuth Verlag GmbH, Berlin, zuletzt geprüft am 01.03.2024.
- [DIN ISO 9000: 2015] Deutsches Institut für Normung (2015): DIN EN ISO 9000:2015-11, Qualitätsmanagementsysteme_- Grundlagen und Begriffe (ISO_9000:2015); Deutsche und Englische Fassung EN_ISO_9000:2015, Beuth Verlag GmbH, Berlin, online verfügbar unter <https://www.beuth.de/de/norm/din-en-iso-9000/235671064>, zuletzt geprüft am 09.10.2020, DOI: 10.31030/2325650.
- [Draganidis und Mentzas 2006] Draganidis, F. und Mentzas, G. (2006): Competency based management: a review of systems and approaches, Seite(n) 51–64. In: Information Management & Computer Security, Jahrgang 14, Heftnummer 1, zuletzt geprüft am 01.03.2024, DOI: 10.1108/09685220610648373.

- [EASA 2016] EASA (2016): Passenger information on Samsung Galaxy Note 7 | EASA, online verfügbar unter <https://www.easa.europa.eu/newsroom-and-events/news/passenger-information-samsung-galaxy-note-7>, zuletzt geprüft am 30.09.2020.
- [Erpenbeck et al. 2013] Erpenbeck, J.; Rosenstiel, L.; Grote, S. (2013): Kompetenzmodelle von Unternehmen - Mit praktischen Hinweisen für ein erfolgreiches Management von Kompetenzen, Auflage 1. Auflage 2013, Schäffer-Poeschel, Planegg, online verfügbar unter http://www.content-select.com/index.php?id=bib_view&ean=9783799267038, zuletzt geprüft am 01.03.2024, ISBN: 978-3-7992-6703-8.
- [Erpenbeck et al. 2017] Erpenbeck, J.; Rosenstiel, L.; Sauter, W.; Grote, S. (2017): Handbuch Kompetenzmessung - Erkennen, verstehen und bewerten von Kompetenzen in der betrieblichen, pädagogischen und psychologischen Praxis, Auflage 3. Auflage 2017, Schäffer-Poeschel, Planegg, online verfügbar unter http://www.content-select.com/index.php?id=bib_view&ean=9783791035123, zuletzt geprüft am 01.03.2024, ISBN: 978-3-7910-3511-6.
- [Farid 2012] Farid, W. M. (2012): The NORMAP Methodology: Lightweight Engineering of Non-functional Requirements for Agile Processes, Seite(n) 322–325, zuletzt geprüft am 30.09.2020, DOI: 10.1109/APSEC.2012.23. In: APEC (Hg.): 2012 19th Asia-Pacific Software Engineering Conference, 2012 19th Asia-Pacific Software Engineering Conference, Hong Kong, China, Band 1, ISBN: 1530-1362.
- [Farid und Mitropoulos 2012] Farid, W. M. und Mitropoulos, F. J. (2012): NORMATIC: A visual tool for modeling Non-Functional Requirements in agile processes, Seite(n) 1–8, zuletzt geprüft am 30.09.2020, DOI: 10.1109/SECon.2012.6196989. In: IEEE Southeastcon (Hg.): 2012 Proceedings of IEEE Southeastcon, 2012 Proceedings of IEEE Southeastcon, Orlando, USA, ISBN: 1558-058X.
- [Fast-Berglund et al. 2013] Fast-Berglund, Å.; Fässberg, T.; Hellman, F.; Davidsson, A.; Stahre, J. (2013): Relations between complexity, quality and cognitive automation in mixed-model assembly, Seite(n) 449–455. In: Journal of Manufacturing Systems, Jahrgang 32, Heftnummer

3, zuletzt geprüft am 25.01.2020, DOI: 10.1016/j.jmsy.2013.04.011.

- [Float 2024] Float (2024): Resource Management, Planning & Scheduling Software, online verfügbar unter <https://www.float.com/>, zuletzt geprüft am 29.05.2024.
- [Galler 1997] Galler, J. (1997): Vom Geschäftsprozeßmodell zum Workflow-Modell, Gabler Verlag, Wiesbaden, zuletzt geprüft am 24.03.2023, ISBN: 978-3-322-90848-3.
- [Gembrys und Herrmann 2009] Gembrys, S.-N. und Herrmann, J. (2009): Qualitätsmanagement, Band 137, Auflage 2., aktualisierte Aufl., Haufe, Planegg/München, zuletzt geprüft am 25.01.2020, ISBN: 978-3448091250.
- [Gemino et al. 2021] Gemino, A.; Horner Reich, B.; Serrador, P. M. (2021): Agile, Traditional, and Hybrid Approaches to Project Success: Is Hybrid a Poor Second Choice?, Seite(n) 161–175. In: Project Management Journal, Jahrgang 52, Heftnummer 2, zuletzt geprüft am 01.03.2024, DOI: 10.1177/8756972820973082.
- [Ghosh und Gunning 2019] Ghosh, S. und Gunning, D. (2019): Natural Language Processing Fundamentals : Build Intelligent Applications That Can Interpret the Human Language to Deliver Impactful Results, Packt Publishing, Limited, Birmingham, UNITED KINGDOM, online verfügbar unter <http://ebookcentral.proquest.com/lib/ubwuportal-ebooks/detail.action?docID=5744468>, zuletzt geprüft am 04.03.2020, ISBN: 9781789955989.
- [Grimm 1987] Grimm, W. (1987): Diagnosesystem für steuerungsperipherie Fehler an Fertigungseinrichtungen, Springer Berlin Heidelberg, Berlin, Heidelberg, zuletzt geprüft am 25.01.2020, ISBN: 978-3-642-83073-0.
- [Grimm 2003] Grimm, K. (2003): Software technology in an automotive company - major challenges, Seite(n) 498–503, zuletzt geprüft am 25.01.2020, DOI: 10.1109/ICSE.2003.1201228. In: ICSE (Hg.): 25th International Conference on Software Engineering, ICSE 2003, 3-10 May 2003, Portland, Oregon, USA - Proceedings, 25th International Conference on Software Engineering, 2003. Proceedings, Portland, OR, USA, Veranstaltungsdatum

- 5/10/2003 - 5/10/2003, IEEE, Los Alamitos, Calif., ISBN: 0-7695-1877-X.
- [Haarmeier 2021] Haarmeier, M. (2021): Künstliche Intelligenz für den Mittelstand - Erfolgreiche Einführung und Nutzung von KI-Anwendungen in Unternehmen, Auflage 1st ed. 2021, Springer Fachmedien Wiesbaden; Imprint Springer Gabler, Wiesbaden, zuletzt geprüft am 03.03.2023, ISBN: 978-3-658-36084-9.
- [Hartmann 2018] Hartmann, M. (2018): Machine Learning und IT-Security, Seite(n) 231–235. In: Datenschutz Datenschich, Jahrgang 42, Heftnummer 4, zuletzt geprüft am 17.02.2023, DOI: 10.1007/s11623-018-0913-5.
- [Hartwig 2010] Hartwig, A. (2010): Untersuchung und Bewertung ausgewählter Qualitätsmanagementmethoden, die Einordnung in Qualitätsmanagementsysteme und praktische Anwendung, Auflage 1. Auflage, Diplom.de, Hamburg, zuletzt geprüft am 25.01.2020, ISBN: 9783842803244.
- [Heilmann 1998] Heilmann, H. (1998): Organisatorische Flexibilität im intelligenten Unternehmen - Potentiale von Workflow-Management, Seite(n) 109–128, zuletzt geprüft am 24.03.2023. In: Bürgel, H. D. (Hg.): Wissensmanagement - Schritte zum intelligenten Unternehmen, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-642-71996-7.
- [Heinrich 1998] Heinrich, P. (1998): Wörterbuch der Mikropolitik - [Horst Boatzky als Festschrift zu seinem 60. Geburtstag], Leske + Budrich, Opladen, zuletzt geprüft am 01.03.2024, ISBN: 3810020133.
- [Heyse 2007] Heyse, V. (2007): Kompetenzmanagement, Waxmann Verlag, Münster, zuletzt geprüft am 01.03.2024, ISBN: 9783830968252.
- [Holert 2020] Holert, R. (2020): Microsoft Project 2019 - Projektmanagement mit Microsoft Project Project Server und Project Online. Das Handbuch für Projektleiter Projektmitarbeiter Ressourcenmanager und Führungskräfte, Holert, München, zuletzt geprüft am 29.05.2024, ISBN: 9783982125121.

- [Hruschka 2009] Hruschka, P. (2009): *Agility kompakt - Tipps für erfolgreiche Systementwicklung*, Auflage 2, Spektrum Akademischer Verlag, Heidelberg, zuletzt geprüft am 01.03.2024, ISBN: 978-3827420923.
- [Hussain et al. 2013] Hussain, I.; Kosseim, L.; Ormandjieva, O. (2013): Approximation of COSMIC functional size to support early effort estimation in Agile, Seite(n) 2–14. In: *Data & Knowledge Engineering*, Jahrgang 85, online verfügbar unter <http://www.sciencedirect.com/science/article/pii/S0169023X1200064X>, zuletzt geprüft am 30.09.2020, DOI: 10.1016/j.datak.2012.06.005.
- [Ionescu et al. 2017] Ionescu, V.-S.; Demian, H.; Czibula, I.-G. (2017): Natural Language Processing and Machine Learning Methods for Software Development Effort Estimation. In: *SIC*, Jahrgang 26, Heftnummer 2, online verfügbar unter <https://sic.ici.ro/wp-content/uploads/2017/06/SIC-2-2017-Art.10.pdf>, zuletzt geprüft am 30.09.2020, DOI: 10.24846/v26i2y201710.
- IONOS 2023: Hard Skills: Definition und Liste. In: *IONOS Startup Guide*, 2023, online verfügbar unter <https://www.ionos.de/startupguide/produktivitaet/hard-skills/>, zuletzt geprüft am 15.05.2024.
- [Jahnsen und Bosch 2005] Jahnsen, A. und Bosch, J. (2005): Software Architecture as a Set of Architectural Design Decisions, Seite(n) 109–120, zuletzt geprüft am 30.09.2024, DOI: 10.1109/IEEECONF11590.2005. In: Clements, P. und Nord, R. (Hg.): *5th Working IEEE/IFIP Conference on Software Architecture (WICSA'05) // Proceedings / 5th Working IEEE/IFIP Conference on Software Architecture, WICSA 2005 - 6 - 10 November 2005, Pittsburgh, Pennsylvania ; papers and working session results, 5th Working IEEE/IFIP Conference on Software Architecture (WICSA'05), Pittsburgh, PA, USA, Veranstaltungsdatum 06.11.2005 - 10.11.2005, IEEE; IEEE Computer Society, Los Alamitos, Calif., ISBN: 0-7695-2548-2.*
- [Jakoby 2015] Jakoby, W. (2015): *Projektmanagement für Ingenieure - Ein praxisnahes Lehrbuch für den systematischen Projekterfolg*, Auflage

- 3., aktualisierte u. erw. Aufl. 2015, Springer Fachmedien Wiesbaden, Wiesbaden, zuletzt geprüft am 15.05.2024, ISBN: 9783658026080.
- [Joosten et al. 1994] Joosten, S. M. M.; Duitshof, M.; Huffmeijer, R.; Mulder, E.; Aussems, G. J. A. (1994): An empirical study about the practice of workflow management, Auflage 2. ed., Universiteit Twente, Twente, zuletzt geprüft am 24.03.2023, ISBN: 90-365-0683-2.
- [Jurafsky und Martin 2009] Jurafsky, D. und Martin, J. H. (2009): Speech and language processing - An introduction to natural language processing computational linguistics and speech recognition, Auflage Pearson internat. ed., 2. ed., zuletzt geprüft am 15.05.2024, ISBN: 978-0135041963.
- [Känel 2020] Känel, S. von (2020): Projekte und Projektmanagement, Auflage 1. Auflage, Springer Fachmedien Wiesbaden; Imprint Springer Gabler, Wiesbaden, zuletzt geprüft am 15.05.2024, ISBN: 9783658300852.
- [Kao 2007] Kao, A. (2007): Natural Language Processing and Text Mining, Springer-Verlag London Limited, London, zuletzt geprüft am 03.03.2023, DOI: 10.1007/978-1-84628-754-1, ISBN: 978-1-84628-175-4.
- [Kapici 2005] Kapici, S. (2005): Ein stochastisches Risikomodell für komplexe Projekte - Dissertation, online verfügbar unter <https://open-data.uni-halle.de/bitstream/1981185920/10673/1/senka-pici.pdf>, zuletzt geprüft am 12.03.2020.
- [Kauffeld 2006] Kauffeld, S. (2006): Kompetenzen messen, bewerten, entwickeln - Ein prozessanalytischer Ansatz für Gruppen, Auflage 1. Auflage 2006, Schäffer-Poeschel, Planegg, online verfügbar unter http://www.content-select.com/index.php?id=bib_view&ean=9783799261685, zuletzt geprüft am 01.03.2024, ISBN: 978-3-7910-2508-7.
- [Khare et al. 2017] Khare, A.; Stewart, B.; Schatz, R. (2017): Phantom Ex Machina - Digital Disruption's Role in Business Model Transformation, Springer International Publishing, Cham, s.l., zuletzt geprüft am 25.01.2020, DOI: 10.1007/978-3-319-44468-0, ISBN: 9783319444673.

- [Kreutzer 2019] Kreutzer, R. T. (2019): Künstliche Intelligenz verstehen - Grundlagen – Use-Cases – unternehmenseigene KI-Journey, Springer Fachmedien Wiesbaden; Imprint Springer Gabler, Wiesbaden, zuletzt geprüft am 03.03.2023, ISBN: 9783658255619.
- [Kumar 2011] Kumar, E. (2011): Natural language processing, I.K. International Publishing House, New Delhi, zuletzt geprüft am 27.11.2021, ISBN: 9789380578774.
- [Kuster et al. 2019] Kuster, J.; Bachmann, C.; Huber, E. (2019): Handbuch Projektmanagement - Agil - klassisch - hybrid, Auflage 4., vollständig überarbeitete und erweiterte Auflage, Springer Gabler, Berlin, online verfügbar unter <https://link.springer.com/content/pdf/10.1007/978-3-662-57878-0.pdf#page=80&zoom=100,78,342>, zuletzt geprüft am 11.03.2020, ISBN: 9783662578780.
- [Lange 2015] Lange, S. (2015): Komplexität im Projektmanagement - Methoden und Fallbeispiele für erfolgreiche Projekte, Auflage 1. Aufl. 2015, zuletzt geprüft am 24.03.2020, ISBN: 978-3-658-09972-5.
- [Lang-von Wins 2009] Lang-von Wins, T. (2009): Biografiegestützte Kompetenzdiagnose, Seite(n) 265–284, zuletzt geprüft am 01.03.2024. In: G. Faix, W. (Hg.): Talent. Kompetenz. Management, Auflage 1, Steinbeis-Edition, Stuttgart, ISBN: 978-3-938062-88-3.
- [Lang-von Wins 2016] Lang-von Wins, T. (2016): Die Kompetenzhaltigkeit von Methoden moderner psychologischer Diagnostik-, Personalauswahl- und Arbeitsanalyseverfahren sowie aktueller Management-Diagnostik-Ansätze, Seite(n) 758–791, zuletzt geprüft am 01.03.2024. In: Sauter, W. und Staudt, A.-K. (Hg.): Kompetenzmessung in der Praxis, Springer Fachmedien Wiesbaden, Wiesbaden, ISBN: 978-3-658-11903-4.
- [Lassmann 2006] Lassmann, W. (2006): Wirtschaftsinformatik - Nachschlagewerk für Studium und Praxis, Gabler; Betriebswirtschaftlicher Verlag Dr. Th. Gabler | GWV Fachverlage GmbH Wiesbaden, Wiesbaden, zuletzt geprüft am 01.03.2024, ISBN: 978-3-409-12725-7.
- [Lee 2023] Lee, A. (2023): What Are Large Language Models Used For?, online verfügbar unter <https://blogs.nvidia.com/blog/what>

- are-large-language-models-used-for/, zuletzt geprüft am 03.10.2024.
- [Lehmann et al. 2015] Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D.; Mendes, P. N.; Hellmann, S. et al. (2015): DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia, Seite(n) 167–195. In: Semantic Web, Jahrgang 6, Heftnummer 2, zuletzt geprüft am 30.09.2024, DOI: 10.3233/SW-140134.
- [Leong et al. 2023] Leong, J.; May Yee, K.; Baitsegi, O.; Palanisamy, L.; Ramasamy, R. K. (2023): Hybrid Project Management between Traditional Software Development Lifecycle and Agile Based Product Development for Future Sustainability, Seite(n) 1121. In: Sustainability, Jahrgang 15, Heftnummer 2, zuletzt geprüft am 01.03.2024, DOI: 10.3390/su15021121.
- [Li et al. 2018] Li, S.-A.; Jeffs, L.; Barwick, M.; Stevens, B. (2018): Organizational contextual features that influence the implementation of evidence-based practices across healthcare settings: a systematic integrative review, Seite(n) 72. In: Systematic reviews, Jahrgang 7, Heftnummer 1, zuletzt geprüft am 24.05.2024, DOI: 10.1186/s13643-018-0734-5.
- [Li et al. 2022] Li, J.; Sun, A.; Han, J.; Li, C. (2022): A Survey on Deep Learning for Named Entity Recognition, Seite(n) 50–70. In: IEEE Trans. Knowl. Data Eng., Jahrgang 34, Heftnummer 1, zuletzt geprüft am 30.09.2024, DOI: 10.1109/TKDE.2020.2981314.
- [Lindemann 2009] Lindemann, U. (2009): Methodische Entwicklung technischer Produkte - Methoden flexibel und situationsgerecht anwenden, Springer Berlin Heidelberg, Berlin, Heidelberg, zuletzt geprüft am 01.03.2024, ISBN: 978-3-642-01422-2.
- [Linkedin 2019] Linkedin (2019): Global Talent Trends - The 4 trends transforming your workplace, online verfügbar unter <https://app.box.com/s/c5scskbsz9q6lb0hqb7euqeb4fr8m0bl/file/388525098383>, zuletzt geprüft am 15.05.2024.
- [Litke 2005] Litke, H.-D. (2005): Projektmanagement - Handbuch für die Praxis ; Konzepte - Instrumente - Umsetzung, zuletzt geprüft am 01.03.2024, ISBN: 3-446-22907-8.

- [Lucassen et al. 2016] Lucassen, G.; Dalpiaz, F.; van der Werf, J. M. E. M.; Brinkkemper, S. (2016): Improving agile requirements: the Quality User Story framework and tool, Seite(n) 383–403. In: Requirements Eng, Jahrgang 21, Heftnummer 3, zuletzt geprüft am 30.09.2020, DOI: 10.1007/s00766-016-0250-x.
- [Luhmann 2005] Luhmann, N. (2005): Komplexität, Seite(n) 255–276, zuletzt geprüft am 30.09.2020, DOI: 10.1007/978-3-663-11447-5_11. In: Luhmann, N. (Hg.): Aufsätze zur Theorie der Gesellschaft, Band / Niklas Luhmann ; 2, Auflage 5. Aufl., [Sondered.], VS Verl. für Sozialwiss, Wiesbaden, ISBN: 978-3-531-61281-2.
- [Madauss 2020] Madauss, B.-J. (2020): Projektmanagement - Theorie und Praxis aus einer Hand, Auflage 8th ed. 2020, Springer Berlin Heidelberg; Imprint Springer Vieweg, Berlin, Heidelberg, zuletzt geprüft am 15.05.2024, ISBN: 978-3-662-59383-7.
- [Magyar und Szenasi 2021] Magyar, D. und Szenasi, S. (2021): Parsing via Regular Expressions, Seite(n) 235–238, zuletzt geprüft am 15.05.2024, DOI: 10.1109/SAMI50585.2021.9378647. In: IEEE (Hg.): 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI), Herľany, Slovakia, Veranstaltungsdatum 21.01.2021 - 23.01.2021, IEEE, ISBN: 978-1-7281-8053-3.
- [Manning et al. 2014] Manning, C.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S.; McClosky, D. (2014): The Stanford CoreNLP Natural Language Processing Toolkit, Seite(n) 55–60, zuletzt geprüft am 15.05.2024, DOI: 10.3115/v1/P14-5010. In: Bontcheva, K. und Jingbo, Z. (Hg.): Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Baltimore, Maryland, Association for Computational Linguistics, Stroudsburg, PA, USA.
- [Martin 1991] Martin, J. (1991): Rapid Application Development, Macmillan Pub. Co, New York, zuletzt geprüft am 24.05.2024, ISBN: 9780023767753.

- [Masing 2021] Masing, W. (2021): Masing Handbuch Qualitätsmanagement, Auflage 7., überarbeitete Auflage, zuletzt geprüft am 01.03.2024, ISBN: 978-3-446-46230-4.
- [McCarthy et al. 2006] McCarthy, J.; Minsky, M. L.; Shannon, C. E. (2006): A proposal for the Dartmouth summer research project on artificial intelligence, Seite(n) 12–14. In: AI MAGAZINE, Jahrgang 27, Heftnummer 4, zuletzt geprüft am 17.02.2023.
- [Meisterplan Help Center 2023] Meisterplan Help Center (2023): Meisterplan Help Center - Video Tutorials, online verfügbar unter <https://help.meisterplan.com/hc/de/articles/4402182326034-Video-Tutorials>, zuletzt geprüft am 29.05.2024.
- [Mertens 1974] Mertens, D. (1974): Schlüsselqualifikationen. Thesen zur Schaltung für eine moderne Gesellschaft, Seite(n) 36–43. In: Mitteilungen aus der Arbeitsmarkt- und Berufsforschung, Jahrgang 7, Heftnummer 1, online verfügbar unter <https://EconPapers.repec.org/RePEc:iab:iabmit:v:07:i:1:p:036-043>, zuletzt geprüft am 01.03.2024.
- [Mircea 2019] Mircea, E. (2019): Project Management using Agile Frameworks, Seite(n) 34–43. In: EIJ, Jahrgang 19, Heftnummer 1/2019, zuletzt geprüft am 01.03.2024, DOI: 10.12948/ei2019.01.04.
- [Mistler et al. 2021] Mistler, M.; Schluter, N.; Lower, M. (2021): Agile Design of Organizations using the Information Flow Analysis and the Generic Systems Engineering, Seite(n) 492–497, zuletzt geprüft am 01.03.2024, DOI: 10.1109/SMC52423.2021.9658840. In: IEEE (Hg.): 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Melbourne, Australia, Veranstaltungsdatum 17.10.2021 - 20.10.2021, IEEE, ISBN: 978-1-6654-4207-7.
- [Mistler 2021] Mistler, M. (2021): Entwicklung eines Vorgehenskonzeptes zum modellbasierten agilen Anforderungsmanagement (Requirements Engineering und Requirements Management) für Organisationen – REMOt, Wuppertal, zuletzt geprüft am 27.11.2021, DOI: 10.25926/ZFTC-T276.

- [Monday Apps Framework 2024] Monday Apps Framework (2024): What is a monday app? - Learn all about monday apps., online verfügbar unter <https://developer.monday.com/apps/docs/intro>, zuletzt geprüft am 29.05.2024.
- [Müller 2007] Müller, M. (2007): Automotive SPICE in der Praxis - Interpretationshilfe für Anwender und Assessoren, Auflage 1. Aufl., dpunkt.Verl., Heidelberg, zuletzt geprüft am 01.03.2024, ISBN: 978-3898644693.
- [Müller-Lindenberg 2005] Müller-Lindenberg, M. (2005): Führung in zeitkritischen und komplexen Projekten - Theoriebildung am Beispiel der Softwareentwicklung, Auflage 1. Aufl., Dt. Univ.-Verl., Wiesbaden, zuletzt geprüft am 25.01.2020, ISBN: 978-3-8244-8294-8.
- [Muthee et al. 2022] Muthee, M. G.; Makau, M.; Amos, O. (2022): review of techniques for morphological analysis in natural language processing, Seite(n) 93–103. In: AJSTSS, Jahrgang 1, Heftnummer 2, zuletzt geprüft am 15.05.2024, DOI: 10.58506/ajstss.v1i2.11.
- [Navigli 2009] Navigli, R. (2009): Word sense disambiguation, Seite(n) 1–69. In: ACM Comput. Surv., Jahrgang 41, Heftnummer 2, zuletzt geprüft am 15.05.2024, DOI: 10.1145/1459352.1459355.
- [Nieke 2002] Nieke, W. (2002): Kompetenz, Seite(n) 13–27, zuletzt geprüft am 01.03.2024, DOI: 10.1007/978-3-663-08029-9_2. In: Otto, H.-U., Rauschenbach, T. und Vogel, P. (Hg.): Erziehungswissenschaft: Professionalität und Kompetenz, VS Verlag für Sozialwissenschaften, Wiesbaden, ISBN: 978-3-663-08030-5.
- [Nielsen 1993] Nielsen, J. (1993): Usability engineering, Auflage 1. print, AP Professional, Boston, Mass., zuletzt geprüft am 30.09.2024, ISBN: 978-0125184069.
- [Nonaka und Takeuchi 1995] Nonaka, I. und Takeuchi, H. (1995): The knowledge creating company - How Japanese companies create the dynamics of innovation, Harvard Business Review Press, Boston, zuletzt geprüft am 24.05.2024, ISBN: 0195092694.

- [North 2005] North, K. (2005): Kompetenzmanagement in der Praxis - Mitarbeiterkompetenzen systematisch identifizieren, nutzen und entwickeln, Gabler Verlag, Wiesbaden, zuletzt geprüft am 15.05.2024, ISBN: 9783322846341.
- [Özçelik und Ferman 2006] Özçelik, G. und Ferman, M. (2006): Competency Approach to Human Resources Management: Outcomes and Contributions in a Turkish Cultural Context, Seite(n) 72–91. In: Human Resource Development Review, Jahrgang 5, Heftnummer 1, zuletzt geprüft am 01.03.2024, DOI: 10.1177/1534484305284602.
- [Palmer et al. 2005] Palmer, M.; Gildea, D.; Kingsbury, P. (2005): The Proposition Bank: An Annotated Corpus of Semantic Roles, Seite(n) 71–106. In: Computational Linguistics, Jahrgang 31, Heftnummer 1, zuletzt geprüft am 15.05.2024, DOI: 10.1162/0891201053630264.
- [Patil und Gudivada 2024] Patil, R. und Gudivada, V. (2024): A Review of Current Trends, Techniques, and Challenges in Large Language Models (LLMs), Seite(n) 2074. In: Applied Sciences, Jahrgang 14, Heftnummer 5, zuletzt geprüft am 03.10.2024, DOI: 10.3390/app14052074.
- [Patzak und Rattay 2018] Patzak, G. und Rattay, G. (2018): Projektmanagement - Projekte Projektportfolios Programme und projektorientierte Unternehmen, Auflage 7., aktualisierte Auflage, zuletzt geprüft am 01.03.2024, ISBN: 978-3-7094-0888-9.
- [Peters-Kühlinger und John 2022] Peters-Kühlinger, G. und John, F. (2022): Soft Skills, Auflage 5. Auflage 2022, Haufe Lexware, Planegg, online verfügbar unter http://www.content-select.com/index.php?id=bib_view&ean=9783648160206, zuletzt geprüft am 15.05.2024, ISBN: 9783648160206.
- [Plewan und Poensgen 2011] Plewan, H.-J. und Poensgen, B. (2011): Produktive Softwareentwicklung - Bewertung und Verbesserung von Produktivität und Qualität in der Praxis, Auflage 1. Aufl., Dpunkt-Verl., Heidelberg, zuletzt geprüft am 01.03.2024, ISBN: 9783898646864.
- [Projektron GmbH 2024] Projektron GmbH (2024): Projektron BCS: Geschäftsprozess- & Projektmanagement, online verfügbar unter <https://www.projektron.de/projektmanagementsoftware/>, zuletzt geprüft am 29.05.2024.

- [Qiu et al. 2020] Qiu, X.; Sun, T.; Xu, Y.; Shao, Y.; Dai, N.; Huang, X. (2020): Pre-trained models for natural language processing: A survey, Seite(n) 1872–1897. In: Sci. China Technol. Sci., Jahrgang 63, Heftnummer 10, zuletzt geprüft am 15.05.2024, DOI: 10.1007/s11431-020-1647-3.
- [Rai und Borah 2021] Rai, A. und Borah, S. (2021): Study of Various Methods for Tokenization, Seite(n) 193–200, zuletzt geprüft am 15.05.2024, DOI: 10.1007/978-981-15-6198-6_18. In: Mandal, J. K., Mukhopadhyay, S. und Roy, A. (Hg.): Applications of Internet of Things, Springer Singapore, Singapore, ISBN: 978-981-15-6197-9.
- [Rane 2017] Rane, P. P. (2017): Automatic Generation of Test Cases for Agile using Natural Language Processing - Automatic Generation of Test Cases for Agile using Natural Language Processing, online verfügbar unter <https://vttechworks.lib.vt.edu/handle/10919/76680>, zuletzt geprüft am 30.09.2020.
- [Reiff und Schlegel 2022] Reiff, J. und Schlegel, D. (2022): Hybrid project management – a systematic literature review, Seite(n) 45–63. In: IJISPM, Jahrgang 10, Heftnummer 2, zuletzt geprüft am 01.03.2024, DOI: 10.12821/ijispmp100203.
- [Rich und Knight 1991] Rich, E. und Knight, K. (1991): Artificial intelligence, Auflage 2. ed., McGraw-Hill, New York, zuletzt geprüft am 17.02.2023, ISBN: 978-0070522633.
- [Richardson 2007] Richardson, L. (2007): RESTful web services, O'Reilly; Safari Books Online, Sebastopol, Calif., Boston, Mass., online verfügbar unter <https://learning.oreilly.com/library/view/-/9780596529260/?ar>, zuletzt geprüft am 30.09.2024, ISBN: 9780596529260.
- [Richter 2003] Richter, S. (2003): Feature based programming - Planung Programmierung Projekt-Management: über die Kunst systematisch zu planen und mit Agilität umzusetzen, zuletzt geprüft am 01.03.2024, ISBN: 3-8273-2077-1.
- [Rimscha 2017] Rimscha, M. von (2017): Algorithmen kompakt und verständlich, Springer Fachmedien Wiesbaden, Wiesbaden, zuletzt geprüft am 03.03.2023, ISBN: 978-3-658-18610-4.

- [Rosli 2022] Rosli, L. N. (2022): THE SUCCESS FACTOR OF USING AGILE TECHNOLOGY ON SOFTWARE PROJECT MANAGEMENT. In: Faculty of Information Science and Technology, online verfügbar unter https://www.researchgate.net/publication/360866225_THE_SUCCESS_FACTOR_OF_USING_A_GILE TECHNOLOGY_ON_SOFTWARE_PROJECT_MANAGEMENT, zuletzt geprüft am 01.03.2024.
- [Ruder 2019] Ruder, S. (2019): Neural transfer learning for natural language processing, Ireland, online verfügbar unter <https://researchrepository.universityofgalway.ie/entities/publication/cd6308ee-6911-460b-8ca7-27d37f44fb04>, zuletzt geprüft am 30.09.2024.
- [Russell und Norvig 2010] Russell, S. J. und Norvig, P. (2010): Artificial intelligence - A modern approach, Auflage 3. ed., Prentice Hall, Upper Saddle River, NJ, online verfügbar unter <https://scholar.alaqsa.edu.ps/9195/1/artificial%20intelligence%20a%20modern%20approach%20%283rd%20edition%29.pdf%20%28%20pdfdrive%20%29.pdf>, zuletzt geprüft am 17.02.2023, ISBN: 978-0-13-604259-4.
- [S.Phabmixay et al. 2018] S.Phabmixay, C.; Rodríguez-Escudero, A. I.; Rodríguez-Pinto, J. (2018): Organizational antecedents to designing a comprehensive complaint management system, Seite(n) 1–26. In: Journal of Management & Organization, Jahrgang 9, zuletzt geprüft am 25.01.2020, DOI: 10.1017/jmo.2018.73.
- [Santinbáñez Koref 2015] Santinbáñez Koref, I. (2015): Beispiele der Anwendung von Evolutionären Algorithmen in Forschung und Industrie, online verfügbar unter <http://docplayer.org/52751264-Beispiele-der-anwendung-von-evolutionären-algorithmen-in-forschung-und-industrie.html>, zuletzt geprüft am 03.03.2023.
- [SAP 2024] SAP (2024): SAP Enterprise Portfolio and Project Management - Funktionen, online verfügbar unter <https://www.sap.com/germany/products/scm/project-portfolio-management/features.html>, zuletzt geprüft am 29.05.2024.
- [Sauter und Staudt 2016a] Sauter, W. und Staudt, A.-K. (2016): Methoden der Kompetenzmessung, Seite(n) 7–21, zuletzt geprüft am 01.03.2024, DOI:

- 10.1007/978-3-658-11904-1_2. In: Sauter, W. und Staudt, A.-K. (Hg.): Kompetenzmessung in der Praxis, Springer Fachmedien Wiesbaden, Wiesbaden, ISBN: 978-3-658-11903-4.
- [Sauter und Staudt 2016b] Sauter, W. und Staudt, F.-P. (2016): Strategisches Kompetenzmanagement 2.0 - Potenziale nutzen – Performance steigern, Auflage 1. Aufl. 2016, Springer Fachmedien Wiesbaden; Imprint Springer Gabler, Wiesbaden, zuletzt geprüft am 01.03.2024, ISBN: 978-3-658-11293-6.
- [Saviom 2024] Saviom (2024): Resource Management and Planning Software, online verfügbar unter <https://www.saviom.com/resource-management-software/>, zuletzt geprüft am 29.05.2024.
- [Schatten et al. 2010] Schatten, A.; Demolsky, M.; Winkler, D.; Biffl, S.; Gostischa-Franta, E.; Östreicher, T. (2010): Best Practice Software-Engineering - Eine praxiserprobte Zusammenstellung von komponentenorientierten Konzepten, Methoden und Werkzeugen, Spektrum Akademischer Verlag, Heidelberg, zuletzt geprüft am 01.03.2024, ISBN: 978-3-8274-2486-0.
- [Schatz et al. 2014] Schatz, A.; Schöllhammer, O.; Jäger, J. (2014): Ansatz zum Umgang mit Komplexität in Unternehmen, Seite(n) 686–693. In: Controlling, Jahrgang 26, Heftnummer 12, online verfügbar unter https://elibrary.vahlen.de/10.15358/0935-0381_2014_12_686.pdf, zuletzt geprüft am 25.01.2020, DOI: 10.15358/0935-0381_2014_12_686.
- [Schlechtweg et al. 2020] Schlechtweg, D.; McGillivray, B.; Hengchen, S.; Dubossarsky, H.; Tahmasebi, N. (2020): SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection, Seite(n) 1–23, zuletzt geprüft am 15.05.2024, DOI: 10.18653/v1/2020.semeval-1.1. In: Herbelot, A., Zhu, X., Palmer, A., Schneider, N., May, J. und Shutova, E. (Hg.): Proceedings of the Fourteenth Workshop on Semantic Evaluation, Proceedings of the Fourteenth Workshop on Semantic Evaluation, Barcelona (online), International Committee for Computational Linguistics, Stroudsburg, PA, USA.
- [Schlueter et al. 2018] Schlueter, N.; Winzer, P.; Ansari, A.; Bielefeld, O.; Dransfeld, H.; Heinrichsmeyer, M. (2018): KAUSAL: A New Methodological Approach for Model Based Analysis of Complex Failure Chains

- by Example of an Electromobility Concept, Seite(n) 947–952, zuletzt geprüft am 27.11.2021, DOI: 10.1109/SMC.2018.00168. In: IEEE (Hg.): 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, Veranstaltungsdatum 07.10.2018 - 10.10.2018, IEEE, ISBN: 978-1-5386-6650-0.
- [Schlund 2011] Schlund, S. (2011): Anforderungsaktualisierung in der Produktentwicklung - Entwicklung einer Methodik zur Aktualisierung von Anforderungen durch die Einbindung anforderungsrelevanter Ereignisse, Band 2011,1, Shaker, Aachen, zuletzt geprüft am 01.03.2024, ISBN: 978-3-8440-0003-0.
- [Schlüter 2023] Schlüter, N. (2023): Generic Systems Engineering - Ein methodischer Ansatz zur Komplexitätsbewältigung, Auflage 3rd ed., Springer Berlin Heidelberg; Imprint Springer Vieweg, Berlin, Heidelberg, zuletzt geprüft am 22.11.2023, ISBN: 978-3-662-66788-0.
- [Schmidt et al. 2000] Schmidt, D. C.; Stal, M.; Rohnert, H. (2000): Patterns for concurrent and networked objects, Band Vol. 2, Wiley, Chichester, zuletzt geprüft am 30.09.2024, ISBN: 978-0471606956.
- [Schmitt et al. 2016] Schmitt et al. (2016): Datenbasierte Qualitätsregelung, Seite(n) 23–42, zuletzt geprüft am 25.01.2020. In: Refflinghaus, R., Kern, C. und Klute-Wenig, S. (Hg.): Qualitätsmanagement 4.0 -- Status Quo! Quo vadis?: Bericht zur GQW-Jahrestagung 2016 in Kassel, Kassel University Press GmbH, [Place of publication not identified], ISBN: 978-3-7376-0084-2.
- [Schott und Campana 2005] Schott, E. und Campana, C. (2005): Strategisches Projektmanagement, zuletzt geprüft am 01.03.2024, ISBN: 978-3-540-26836-9.
- [Sharma et al. 2019] Sharma, S.; Kumar, D.; Fayad, M. E. (2019): Simplified Agile Software Project Selection Model Using Natural Language Processing Based Upon Agile Team Skills, Seite(n) 264–269, zuletzt geprüft am 30.09.2020, DOI: 10.1109/ICCIKE47802.2019.9004392. In: ICCIKE (Hg.): 2019 International Conference on Computational Intelligence and

Knowledge Economy (ICCIKE), 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, UAE.

[Sinha und Das 2021]

Sinha, A. und Das, P. (2021): Agile Methodology Vs. Traditional Waterfall SDLC: A case study on Quality Assurance process in Software Industry, Seite(n) 1–4, zuletzt geprüft am 01.03.2024, DOI: 10.1109/IEMENTech53263.2021.9614779. In: IEMENTech (Hg.): 2021 5th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), 2021 5th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), Kolkata, India, Veranstaltungsdatum 24.09.2021 - 26.09.2021, IEEE, ISBN: 978-1-6654-1803-4.

[Smartsheet Inc. 2024]

Smartsheet Inc. (2024): Content Center, online verfügbar unter <https://de.smartsheet.com/content-center#numberOfResults=8>, zuletzt geprüft am 29.05.2024.

[SpaCy 2024]

SpaCy (2024): Facts and Figures - The hard numbers for spaCy and how it compares to other tools, zuletzt geprüft am 15.05.2024.

[Steinbuch 2000]

Steinbuch, P. A. (2000): Projektorganisation und Projektmanagement, Auflage 2., überarb. Aufl., Kiehl, Ludwigshafen (Rhein), zuletzt geprüft am 01.03.2024, ISBN: 3470485925.

[Sterrer 2014]

Sterrer, C. (2014): Das Geheimnis erfolgreicher Projekte - Kritische Erfolgsfaktoren im Projektmanagement - Was Führungskräfte wissen müssen // Kritische Erfolgsfaktoren im Projektmanagement – Was Führungskräfte wissen müssen, Springer Fachmedien Wiesbaden, Wiesbaden, zuletzt geprüft am 01.03.2024, ISBN: 978-3-658-04797-9.

[Sugimoto et al. 2004]

Sugimoto, T.; Ito, N.; Iwashita, S.; Sugeno, M. (2004): Towards Programming in Everyday Language: A Case for Email Management, Seite(n) 383–394, zuletzt geprüft am 15.05.2024, DOI: 10.1007/978-3-540-24630-5_46. In: Gelbukh, A. (Hg.): Computational linguistics and intelligent text processing - 5th international conference ; proceedings, Band 2945, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-540-21006-1.

- [Sundararajan et al. 2018] Sundararajan, M.; Srikrishnan, P.; Nayak, K. (2018): Requirements Complexity Definition and Classification using Natural Language Processing, Seite(n) 76–80, zuletzt geprüft am 30.09.2020, DOI: 10.1109/IC3I44769.2018.9007287. In: ICI3I (Hg.): 2018 3rd International Conference on Contemporary Computing and Informatics (IC3I), 2018 3rd International Conference on Contemporary Computing and Informatics (IC3I), Gurgaon, Indien.
- [Taulli 2022] Taulli, T. (2022): Natural Language Processing (NLP), Seite(n) 117–140, zuletzt geprüft am 15.05.2024, DOI: 10.1007/978-3-662-66283-0_6. In: Taulli, T. (Hg.): Grundlagen der Künstlichen Intelligenz, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-662-66282-3.
- [Teamdeck 2024] Teamdeck (2024): Teamdeck - Product Tour Videos, online verfügbar unter <https://teamdeck.io/product-tour-videos/>, zuletzt geprüft am 29.05.2024.
- [Teich 2020] Teich, I. (2020): Meilensteine der Entwicklung Künstlicher Intelligenz, Seite(n) 276–284. In: Informatik Spektrum, Jahrgang 43, Heftnummer 4, zuletzt geprüft am 17.02.2023, DOI: 10.1007/s00287-020-01280-5.
- [Thiebes und Plankert 2014] Thiebes, F. und Plankert, N. (2014): Umgang mit Komplexität in der Produktentwicklung – Komplexitätsbeherrschung durch Variantenmanagement, Seite(n) 165–185, online verfügbar unter https://doi.org/10.1007/978-3-658-01284-7_8, zuletzt geprüft am 15.02.2020, DOI: 10.1007/978-3-658-01284-7_8. In: Schoeneberg, K.-P. (Hg.): Komplexitätsmanagement in Unternehmen - Herausforderungen im Umgang mit Dynamik, Unsicherheit und Komplexität meistern, Springer Gabler, Wiesbaden, ISBN: 978-3-658-01284-7.
- [Titus 1994] Titus, A. A. (1994): Competence at Work, by Lyle M. Spencer, Jr., and Signe M. Spencer. (1993). New York: Wiley. 372 pp., \$55.00 cloth, Seite(n) 391–395. In: Human Resource Dev Quarterly, Jahrgang 5, Heftnummer 4, zuletzt geprüft am 01.03.2024, DOI: 10.1002/hrdq.3920050411.

- [Trello Guide 2024] Trello Guide (2024): Trello-Anleitungen: Einführung in Trello | Trello, online verfügbar unter <https://trello.com/guide>, zuletzt geprüft am 29.05.2024.
- [Tripathi und Agarwal 2014] Tripathi, K. und Agarwal, M. (2014): Competency Based Management In Organizational Context: A Literature Review, Seite(n) 349–356. In: Globual Journal of Finance and management, Jahrgang 6, zuletzt geprüft am 01.03.2024.
- [Vöcking et al. 2008] Vöcking, B.; Alt, H.; Dietzfelbinger, M.; Reischuk, R.; Scheideler, C.; Vollmer, H.; Wagner, D. (2008): Taschenbuch der Algorithmen, Springer; Springer-Verlag; Springer Berlin Heidelberg, Berlin, Heidelberg, zuletzt geprüft am 25.01.2020, DOI: 10.1007/978-3-540-76394-9, ISBN: 978-3-540-76394-9.
- [Vogler 1996] Vogler, P. (1996): Chancen und Risiken von Workflow-Management, Seite(n) 343–362, zuletzt geprüft am 24.03.2023, DOI: 10.1007/978-3-322-84940-3_18. In: Österle, H. und Vogler, P. (Hg.): Praxis des Workflow-Managements, Vieweg+Teubner Verlag, Wiesbaden, ISBN: 978-3-322-84941-0.
- Wagner 2016: Samsung Galaxy Note 7: Wie ein brennendes Handy ein ganzes Land verzweifeln lässt. In: DER SPIEGEL, 27.09.2016, online verfügbar unter <https://www.spiegel.de/spiegel/samsung-galaxy-note-7-ein-brennendes-handy-laesst-suedkorea-verzweifeln-a-1114098.html>, zuletzt geprüft am 30.09.2020.
- [Wagner und Alter 2016] Wagner, R. und Alter, W. (2016): Erfolgreiches Projektportfoliomangement - Wie Sie Projektportfolios systematisch gestalten und steuern, Auflage 1. Aufl., zuletzt geprüft am 01.03.2024, ISBN: 978-3-86329-676-6.
- [Wang et al. 2018] Wang, S.; Aggarwal, C.; Liu, H. (2018): Random-Forest-Inspired Neural Networks, Seite(n) 1–25. In: ACM Trans. Intell. Syst. Technol., Jahrgang 9, Heftnummer 6, zuletzt geprüft am 30.09.2024, DOI: 10.1145/3232230.
- [Weber 2020a] Weber, F. (2020): Künstliche Intelligenz, Seite(n) 37–72, zuletzt geprüft am 17.02.2023, DOI: 10.1007/978-3-658-29773-2_2. In: Weber, F. (Hg.): Künstliche Intelligenz für Business Analytics, Springer Fachmedien Wiesbaden, Wiesbaden, ISBN: 978-3-658-29772-5.

- [Weber 2020b] Weber, F. (2020): Künstliche Intelligenz für Business Analytics, Springer Fachmedien Wiesbaden, Wiesbaden, zuletzt geprüft am 17.02.2023, DOI: 10.1007/978-3-658-29773-2, ISBN: 978-3-658-29772-5.
- [Weicker 2015] Weicker, K. (2015): Evolutionäre Algorithmen, Springer Fachmedien Wiesbaden, Wiesbaden, zuletzt geprüft am 03.03.2023, ISBN: 978-3-658-09957-2.
- [Weinert 2014] Weinert, F. E. (2014): Vergleichende Leistungsmessung in Schulen - eine umstrittene Selbstverständlichkeit, Seite(n) 17–32, zuletzt geprüft am 15.05.2024. In: Weinert, F. E. (Hg.): Leistungsmessungen in Schulen, Auflage 3. Auflage, Beltz, Weinheim, ISBN: 9783407256904.
- [Wess 2019] Wess, S. (2019): Mit Künstlicher Intelligenz immer die richtigen Entscheidungen treffen, Seite(n) 143–160, zuletzt geprüft am 24.03.2020, DOI: 10.1007/978-3-662-57568-0_9. In: Buxmann, P. (Hg.): Künstliche Intelligenz - Mit Algorithmen zum wirtschaftlichen Erfolg, Springer Gabler, Berlin, [Germany], ISBN: 978-3-662-57568-0.
- [Wilson und Perumal 2010] Wilson, S. A. und Perumal, A. (2010): Waging war on complexity costs - Reshape your cost structure, free up cash flows, and boost productivity by attacking process, product, and organizational complexity // Reshape your cost structure, free up cash flows, and boost productivity by attacking process, product and organizational complexity, McGraw-Hill, New York, online verfügbar unter <http://www.loc.gov/catdir/enhancements/fy1011/2009046358-b.html>, zuletzt geprüft am 25.01.2020, ISBN: 9780071713504.
- [Wimalasuriya und Dou 2010] Wimalasuriya, D. C. und Dou, D. (2010): Ontology-based information extraction: An introduction and a survey of current approaches, Seite(n) 306–323. In: Journal of Information Science, Jahrgang 36, Heftnummer 3, zuletzt geprüft am 30.09.2024, DOI: 10.1177/0165551509360123.
- [Winzer 2016] Winzer, P. (2016): Generic systems engineering - Ein methodischer ansatz zur Komplexitätsbewältigung, Auflage 2. Auflage,

- Springer Vieweg, Berlin, [Germany], Heidelberg, [Germany], zuletzt geprüft am 25.01.2020, ISBN: 978-3-662-52893-8.
- [Winzer 2017] Winzer, P. (2017): Generic System Description and Problem Solving in Systems Engineering, Seite(n) 2052–2061. In: IEEE Systems Journal, Jahrgang 11, Heftnummer 4, zuletzt geprüft am 01.03.2024, DOI: 10.1109/JSYST.2015.2428811.
- [Wirke Inc. 2024] Wirke Inc. (2024): Webinare, online verfügbar unter <https://www.wrike.com/de/webinars/>, zuletzt geprüft am 29.05.2024.
- [Young et al. 2018] Young, T.; Hazarika, D.; Poria, S.; Cambria, E. (2018): Recent Trends in Deep Learning Based Natural Language Processing [Review Article], Seite(n) 55–75. In: IEEE Comput. Intell. Mag., Jahrgang 13, Heftnummer 3, zuletzt geprüft am 11.03.2020, DOI: 10.1109/MCI.2018.2840738.
- [Yun et al. 2018] Yun, J.; Jeon, J.; Park, K.; Zhao, X. (2018): Benefits and Costs of Closed Innovation Strategy: Analysis of Samsung’s Galaxy Note 7 Explosion and Withdrawal Scandal, Seite(n) 20. In: JOItmC, Jahrgang 4, Heftnummer 3, zuletzt geprüft am 30.09.2020, DOI: 10.3390/joitmc4030020.
- [Zeit 2017] Zeit (2017): Galaxy Note 7 - Samsung führt Brände auf fehlerhafte Akkus zurück, online verfügbar unter <https://www.zeit.de/wirtschaft/unternehmen/2017-01/galaxy-note-7-samsung-brand-fehler-batterie>, zuletzt geprüft am 30.09.2020.
- [Zichler und Helke 2017] Zichler, K. und Helke, S. (2017): Ontologiebasierte Abhängigkeitsanalyse im Projektlastenheft, Seite(n) 121–134, zuletzt geprüft am 30.09.2020. In: Dencker, P., Klenk, H., Keller, H. B. und Plödereder, E. (Hg.): Automotive - Safety & Security 2017 - Sicherheit und Zuverlässigkeit für automobile Informationstechnik : 30.-31. Mai 2017 Stuttgart, Germany, Band volume P-269, Gesellschaft für Informatik e.V. (GI), Bonn, ISBN: 9783885796633.

[Zimbardo und Boyd 2011]

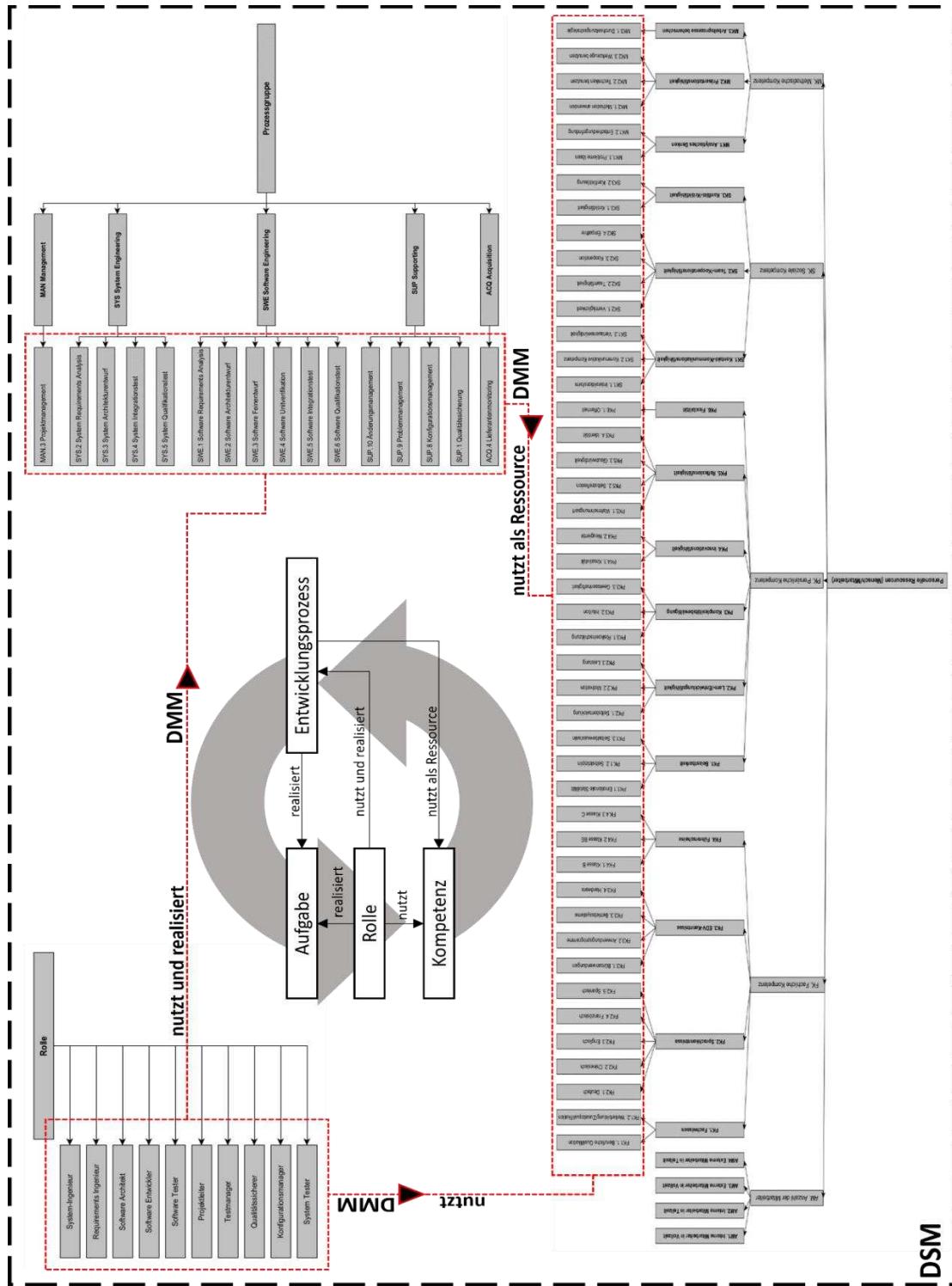
Zimbardo, P. G. und Boyd, J. (2011): Die neue Psychologie der Zeit - Und Wie Sie Ihr Leben Verandern Wird, Auflage Taschenbuchausg, Spektrum Akademischer Verlag GmbH, zuletzt geprüft am 15.05.2024, ISBN: 9783827428455.

[Zwirn 2013]

Zwirn, F. (2013): Analyse und Auswertung von gewichteten Anforderungen in technischen Spezifikationen, zuletzt geprüft am 30.09.2020, DOI: 10.18419/OPUS-3119.

Anhang

A.1 DMM und DSM der Ontologiegraphen



A.2 Knoten der Ontologiegraphen

Die folgende Tabelle listet die **Knoten** des Modells auf:

ID	Label	Color	Domain
MAN.3	Projektmanagement	#0070ff	Entwicklungsprozess
SYS.1	Requirements elicitation	#0070ff	Entwicklungsprozess
SYS.2	System requirements analysis	#0070ff	Entwicklungsprozess
SYS.3	System architectural design	#0070ff	Entwicklungsprozess
SYS.4	System integration and integration test	#0070ff	Entwicklungsprozess
SYS.5	System qualification test	#0070ff	Entwicklungsprozess
SWE.1	Software requirements analysis	#0070ff	Entwicklungsprozess
SWE.2	Software architectural design	#0070ff	Entwicklungsprozess
SWE.3	Software detailed design	#0070ff	Entwicklungsprozess
SWE.4	Software construction	#0070ff	Entwicklungsprozess
SWE.5	Software integration and integration test	#0070ff	Entwicklungsprozess
SWE.6	Software qualification test	#0070ff	Entwicklungsprozess
SUP.1	Quality assurance	#0070ff	Entwicklungsprozess
SUP.8	Configuration management	#0070ff	Entwicklungsprozess
SUP.9	Problem management	#0070ff	Entwicklungsprozess
SUP.10	Change management	#0070ff	Entwicklungsprozess
ACQ.4	Supplier monitoring	#0070ff	Entwicklungsprozess
SYSENG	System Engineer	#DAF7A6	Rollen
REQENG	Requirements Engineer	#DAF7A6	Rollen
SOFA	Software Architect	#DAF7A6	Rollen
SOFD	Software Developer	#DAF7A6	Rollen
SOFT	Software Tester	#DAF7A6	Rollen
PROMA	Project Manager	#DAF7A6	Rollen
TESMANAG	Test Manager	#DAF7A6	Rollen
QUAEN	Quality Engineer	#DAF7A6	Rollen
CONMA	Configuration Manager	#DAF7A6	Rollen
SYSTES	System Tester	#DAF7A6	Rollen
AM.1	Mitarbeiter Intern Voll	#FFC300	Kompetenzen
AM.2	Mitarbeiter Intern Teil	#FFC300	Kompetenzen
AM.3	Mitarbeiter Extern Voll	#FFC300	Kompetenzen
AM.4	Mitarbeiter Extern Teil	#FFC300	Kompetenzen
FK.1	Fachwissen	#FFC300	Kompetenzen
FK.2	Sprachkenntnisse	#FFC300	Kompetenzen
FK.3	EDV-Kenntnisse	#FFC300	Kompetenzen
FK.4	Führerscheine	#FFC300	Kompetenzen
PK.1	Blastbarkeit	#FFC300	Kompetenzen
PK.2	Lernfähigkeit	#FFC300	Kompetenzen
PK.3	Komplexitätbewältigung	#FFC300	Kompetenzen
PK.4	Innovationsfähigkeit	#FFC300	Kompetenzen
PK.5	Reflexionsfähigkeit	#FFC300	Kompetenzen

ID	Label	Color	Domain
PK.6	Flexibilität	#FFC300	Kompetenzen
SK.1	Kontaktfähigkeit	#FFC300	Kompetenzen
SK.2	Teamfähigkeit	#FFC300	Kompetenzen
SK.3	Kritikfähigkeit	#FFC300	Kompetenzen
MK.1	Analytisches Denken	#FFC300	Kompetenzen
MK.2	Präsentationsfähigkeit	#FFC300	Kompetenzen
MK.3	Arbeitsprozess beherrschen	#FFC300	Kompetenzen
FK1.1	Berufliche Qualifikation	#FF5733	Skill
FK1.2	Weiterbildung	#FF5733	Skill
FK2.1	Deutsch	#FF5733	Skill
FK2.2	Chinesisch	#FF5733	Skill
FK2.3	Englisch	#FF5733	Skill
FK2.4	Französisch	#FF5733	Skill
FK2.5	Spanisch	#FF5733	Skill
FK3.1	Büroanwendungen	#FF5733	Skill
FK3.2	Anwendungsprogramme	#FF5733	Skill
FK3.3	Betriebssysteme	#FF5733	Skill
FK3.4	Hardware	#FF5733	Skill
FK4.1	Klasse B	#FF5733	Skill
FK4.2	Klasse BE	#FF5733	Skill
FK4.3	Klasse C	#FF5733	Skill
PK1.1	Emotionale Stabilität	#FF5733	Skill
PK1.2	Selbstdisziplin	#FF5733	Skill
PK1.3	Selbstbewusstsein	#FF5733	Skill
PK2.1	Selbstentwicklung	#FF5733	Skill
PK2.2	Motivation	#FF5733	Skill
PK2.3	Leistung	#FF5733	Skill
PK3.1	Risikoeinschätzung	#FF5733	Skill
PK3.2	Intuition	#FF5733	Skill
PK3.3	Gewissenhaftfähigkeit	#FF5733	Skill
PK4.1	Kreativität	#FF5733	Skill
PK4.2	Neugierde	#FF5733	Skill
PK5.1	Wahrnehmungsart	#FF5733	Skill
PK5.2	Selbstreflexion	#FF5733	Skill
PK5.3	Glaubwürdigkeit	#FF5733	Skill
PK5.4	Identität	#FF5733	Skill
PK6.1	Offenheit	#FF5733	Skill
SK1.1	Interaktionsform	#FF5733	Skill
SK1.2	Kommunikativ	#FF5733	Skill
SK1.3	Vertrauenswürdigkeit	#FF5733	Skill
SK2.1	Verträglichkeit	#FF5733	Skill
SK2.2	Teamfähigkeit	#FF5733	Skill
SK2.3	Kooperation	#FF5733	Skill
SK2.4	Empathie	#FF5733	Skill

ID	Label	Color	Domain
SK3.1	Kiritikfähigkeit	#FF5733	Skill
SK3.2	Konfliktlösung	#FF5733	Skill
MK1.1	Problemlösung	#FF5733	Skill
MK1.2	Entscheidungfindung	#FF5733	Skill
MK2.1	Methoden anwenden	#FF5733	Skill
MK2.2	Techniken benutzen	#FF5733	Skill
MK2.3	Werkzeuge benutzen	#FF5733	Skill
MK3.1	Durchsetzungsstrategie	#FF5733	Skill

A.3 Kanten der Ontologiegraphen

Die folgende Tabelle listet die **Kanten** des Modells auf:

Source	Target	Type	Label	Color
SYSENG	SYS.1	Direct	nutzt und realisiert	#DAF7A6
SYSENG	SYS.3	Direct	nutzt und realisiert	#DAF7A6
REQENG	SYS.2	Direct	nutzt und realisiert	#DAF7A6
REQENG	SWE.1	Direct	nutzt und realisiert	#DAF7A6
SOFA	SWE.2	Direct	nutzt und realisiert	#DAF7A6
SOFD	SWE.3	Direct	nutzt und realisiert	#DAF7A6
SOFT	SWE.4	Direct	nutzt und realisiert	#DAF7A6
SOFT	SWE.5	Direct	nutzt und realisiert	#DAF7A6
PROMA	MAN.3	Direct	nutzt und realisiert	#DAF7A6
PROMA	SUP.9	Direct	nutzt und realisiert	#DAF7A6
PROMA	SUP.10	Direct	nutzt und realisiert	#DAF7A6
PROMA	ACQ.4	Direct	nutzt und realisiert	#DAF7A6
TESMANAG	SWE.6	Direct	nutzt und realisiert	#DAF7A6
TESMANAG	SWE.6	Direct	nutzt und realisiert	#DAF7A6
QUAEN	SUP.1	Direct	nutzt und realisiert	#DAF7A6
QUAEN	SUP.9	Direct	nutzt und realisiert	#DAF7A6
QUAEN	SUP.10	Direct	nutzt und realisiert	#DAF7A6
QUAEN	ACQ.4	Direct	nutzt und realisiert	#DAF7A6
CONMA	SUP.8	Direct	nutzt und realisiert	#DAF7A6
SYSTES	SYS.5	Direct	nutzt und realisiert	#DAF7A6
SYSENG	FK1.1	Direct	nutzt	#DAF7A6
SYSENG	FK1.2	Direct	nutzt	#DAF7A6
SYSENG	FK2.1	Direct	nutzt	#DAF7A6
SYSENG	FK2.3	Direct	nutzt	#DAF7A6
SYSENG	FK2.2	Direct	nutzt	#DAF7A6
SYSENG	FK2.4	Direct	nutzt	#DAF7A6
SYSENG	FK2.5	Direct	nutzt	#DAF7A6
SYSENG	FK3.1	Direct	nutzt	#DAF7A6
SYSENG	FK3.2	Direct	nutzt	#DAF7A6
SYSENG	FK3.3	Direct	nutzt	#DAF7A6

Source	Target	Type	Label	Color
SYSENG	FK3.4	Direct	nutzt	#DAF7A6
SYSENG	FK4.1	Direct	nutzt	#DAF7A6
SYSENG	FK4.2	Direct	nutzt	#DAF7A6
SYSENG	FK4.3	Direct	nutzt	#DAF7A6
SYSENG	PK3.1	Direct	nutzt	#DAF7A6
SYSENG	PK3.2	Direct	nutzt	#DAF7A6
SYSENG	PK3.3	Direct	nutzt	#DAF7A6
SYSENG	PK4.1	Direct	nutzt	#DAF7A6
SYSENG	PK4.2	Direct	nutzt	#DAF7A6
SYSENG	PK2.1	Direct	nutzt	#DAF7A6
SYSENG	MK2.1	Direct	nutzt	#DAF7A6
SYSENG	MK2.2	Direct	nutzt	#DAF7A6
SYSENG	MK2.3	Direct	nutzt	#DAF7A6
REQENG	FK1.1	Direct	nutzt	#DAF7A6
REQENG	FK1.2	Direct	nutzt	#DAF7A6
REQENG	FK2.1	Direct	nutzt	#DAF7A6
REQENG	FK2.3	Direct	nutzt	#DAF7A6
REQENG	FK3.1	Direct	nutzt	#DAF7A6
REQENG	PK3.1	Direct	nutzt	#DAF7A6
REQENG	PK3.2	Direct	nutzt	#DAF7A6
REQENG	PK4.1	Direct	nutzt	#DAF7A6
REQENG	PK4.2	Direct	nutzt	#DAF7A6
REQENG	PK5.1	Direct	nutzt	#DAF7A6
REQENG	PK5.2	Direct	nutzt	#DAF7A6
REQENG	PK6.1	Direct	nutzt	#DAF7A6
REQENG	SK1.1	Direct	nutzt	#DAF7A6
REQENG	SK1.2	Direct	nutzt	#DAF7A6
REQENG	SK2.4	Direct	nutzt	#DAF7A6
REQENG	MK1.1	Direct	nutzt	#DAF7A6
REQENG	MK2.1	Direct	nutzt	#DAF7A6
SOFA	FK1.1	Direct	nutzt	#DAF7A6
SOFA	FK1.2	Direct	nutzt	#DAF7A6
SOFA	FK2.1	Direct	nutzt	#DAF7A6
SOFA	FK2.3	Direct	nutzt	#DAF7A6
SOFA	FK3.2	Direct	nutzt	#DAF7A6
SOFA	FK3.3	Direct	nutzt	#DAF7A6
SOFA	FK3.4	Direct	nutzt	#DAF7A6
SOFA	PK3.1	Direct	nutzt	#DAF7A6
SOFA	PK3.2	Direct	nutzt	#DAF7A6
SOFA	PK3.3	Direct	nutzt	#DAF7A6
SOFA	PK4.1	Direct	nutzt	#DAF7A6
SOFA	PK4.2	Direct	nutzt	#DAF7A6
SOFA	PK5.1	Direct	nutzt	#DAF7A6
SOFA	PK5.2	Direct	nutzt	#DAF7A6

Source	Target	Type	Label	Color
SOFA	PK5.3	Direct	nutzt	#DAF7A6
SOFA	PK5.4	Direct	nutzt	#DAF7A6
SOFA	PK2.1	Direct	nutzt	#DAF7A6
SOFA	MK1.2	Direct	nutzt	#DAF7A6
SOFA	MK2.1	Direct	nutzt	#DAF7A6
SOFA	MK2.2	Direct	nutzt	#DAF7A6
SOFA	MK2.3	Direct	nutzt	#DAF7A6
SOFD	FK1.1	Direct	nutzt	#DAF7A6
SOFD	FK1.2	Direct	nutzt	#DAF7A6
SOFD	FK2.1	Direct	nutzt	#DAF7A6
SOFD	FK2.3	Direct	nutzt	#DAF7A6
SOFD	FK3.1	Direct	nutzt	#DAF7A6
SOFD	FK3.2	Direct	nutzt	#DAF7A6
SOFD	FK3.3	Direct	nutzt	#DAF7A6
SOFD	FK3.4	Direct	nutzt	#DAF7A6
SOFD	PK4.1	Direct	nutzt	#DAF7A6
SOFD	PK4.2	Direct	nutzt	#DAF7A6
SOFD	PK3.1	Direct	nutzt	#DAF7A6
SOFD	PK3.3	Direct	nutzt	#DAF7A6
SOFD	PK5.2	Direct	nutzt	#DAF7A6
SOFD	PK6.1	Direct	nutzt	#DAF7A6
SOFD	SK1.1	Direct	nutzt	#DAF7A6
SOFD	MK1.1	Direct	nutzt	#DAF7A6
SOFD	MK1.2	Direct	nutzt	#DAF7A6
SOFD	MK2.2	Direct	nutzt	#DAF7A6
SOFD	MK2.3	Direct	nutzt	#DAF7A6
SOFT	FK1.1	Direct	nutzt	#DAF7A6
SOFT	FK1.2	Direct	nutzt	#DAF7A6
SOFT	FK2.1	Direct	nutzt	#DAF7A6
SOFT	FK2.3	Direct	nutzt	#DAF7A6
SOFT	FK3.1	Direct	nutzt	#DAF7A6
SOFT	FK3.2	Direct	nutzt	#DAF7A6
SOFT	FK3.3	Direct	nutzt	#DAF7A6
SOFT	FK3.4	Direct	nutzt	#DAF7A6
SOFT	PK3.1	Direct	nutzt	#DAF7A6
SOFT	PK3.3	Direct	nutzt	#DAF7A6
SOFT	PK5.2	Direct	nutzt	#DAF7A6
SOFT	SK3.1	Direct	nutzt	#DAF7A6
SOFT	PK6.1	Direct	nutzt	#DAF7A6
SOFT	MK1.1	Direct	nutzt	#DAF7A6
SOFT	MK2.2	Direct	nutzt	#DAF7A6
SOFT	MK2.3	Direct	nutzt	#DAF7A6
PROMA	FK1.1	Direct	nutzt	#DAF7A6
PROMA	FK1.2	Direct	nutzt	#DAF7A6

Source	Target	Type	Label	Color
PROMA	FK2.1	Direct	nutzt	#DAF7A6
PROMA	FK2.3	Direct	nutzt	#DAF7A6
PROMA	FK2.2	Direct	nutzt	#DAF7A6
PROMA	FK2.4	Direct	nutzt	#DAF7A6
PROMA	FK2.5	Direct	nutzt	#DAF7A6
PROMA	FK3.1	Direct	nutzt	#DAF7A6
PROMA	PK2.2	Direct	nutzt	#DAF7A6
PROMA	PK2.3	Direct	nutzt	#DAF7A6
PROMA	PK3.1	Direct	nutzt	#DAF7A6
PROMA	PK3.2	Direct	nutzt	#DAF7A6
PROMA	PK4.1	Direct	nutzt	#DAF7A6
PROMA	PK2.1	Direct	nutzt	#DAF7A6
PROMA	PK5.2	Direct	nutzt	#DAF7A6
PROMA	PK5.3	Direct	nutzt	#DAF7A6
PROMA	PK6.1	Direct	nutzt	#DAF7A6
TESMANAG	FK1.1	Direct	nutzt	#DAF7A6
TESMANAG	FK1.2	Direct	nutzt	#DAF7A6
TESMANAG	FK2.1	Direct	nutzt	#DAF7A6
TESMANAG	FK2.3	Direct	nutzt	#DAF7A6
TESMANAG	FK3.1	Direct	nutzt	#DAF7A6
TESMANAG	FK3.2	Direct	nutzt	#DAF7A6
TESMANAG	FK3.3	Direct	nutzt	#DAF7A6
TESMANAG	FK3.4	Direct	nutzt	#DAF7A6
TESMANAG	PK3.1	Direct	nutzt	#DAF7A6
TESMANAG	PK3.2	Direct	nutzt	#DAF7A6
TESMANAG	PK4.1	Direct	nutzt	#DAF7A6
TESMANAG	PK5.2	Direct	nutzt	#DAF7A6
TESMANAG	PK5.3	Direct	nutzt	#DAF7A6
TESMANAG	SK1.2	Direct	nutzt	#DAF7A6
TESMANAG	SK2.2	Direct	nutzt	#DAF7A6
TESMANAG	SK3.1	Direct	nutzt	#DAF7A6
TESMANAG	MK1.2	Direct	nutzt	#DAF7A6
TESMANAG	MK2.1	Direct	nutzt	#DAF7A6
QUAEN	FK1.1	Direct	nutzt	#DAF7A6
QUAEN	FK1.2	Direct	nutzt	#DAF7A6
QUAEN	FK2.1	Direct	nutzt	#DAF7A6
QUAEN	FK2.3	Direct	nutzt	#DAF7A6
QUAEN	FK3.1	Direct	nutzt	#DAF7A6
QUAEN	PK3.1	Direct	nutzt	#DAF7A6
QUAEN	PK3.2	Direct	nutzt	#DAF7A6
QUAEN	PK3.3	Direct	nutzt	#DAF7A6
QUAEN	PK5.2	Direct	nutzt	#DAF7A6
QUAEN	PK5.3	Direct	nutzt	#DAF7A6
QUAEN	PK2.1	Direct	nutzt	#DAF7A6

Source	Target	Type	Label	Color
CONMA	FK1.1	Direct	nutzt	#DAF7A6
CONMA	FK1.2	Direct	nutzt	#DAF7A6
CONMA	FK2.1	Direct	nutzt	#DAF7A6
CONMA	FK2.3	Direct	nutzt	#DAF7A6
CONMA	FK3.1	Direct	nutzt	#DAF7A6
CONMA	FK3.2	Direct	nutzt	#DAF7A6
CONMA	FK3.3	Direct	nutzt	#DAF7A6
CONMA	FK3.4	Direct	nutzt	#DAF7A6
CONMA	PK3.1	Direct	nutzt	#DAF7A6
CONMA	PK3.3	Direct	nutzt	#DAF7A6
SYSTES	FK1.1	Direct	nutzt	#DAF7A6
SYSTES	FK1.2	Direct	nutzt	#DAF7A6
SYSTES	FK2.1	Direct	nutzt	#DAF7A6
SYSTES	FK2.3	Direct	nutzt	#DAF7A6
SYSTES	FK3.2	Direct	nutzt	#DAF7A6
SYSTES	FK3.3	Direct	nutzt	#DAF7A6
SYSTES	FK3.4	Direct	nutzt	#DAF7A6
SYSTES	PK3.1	Direct	nutzt	#DAF7A6
SYSTES	PK3.3	Direct	nutzt	#DAF7A6
SYSTES	PK5.2	Direct	nutzt	#DAF7A6
SYSTES	SK3.1	Direct	nutzt	#DAF7A6
SYSTES	PK6.1	Direct	nutzt	#DAF7A6
SYSTES	MK1.1	Direct	nutzt	#DAF7A6
SYSTES	MK2.2	Direct	nutzt	#DAF7A6
SYSTES	MK2.3	Direct	nutzt	#DAF7A6
MAN.3	FK1.1	Direct	nutzt als Ressource	#0070ff
MAN.3	FK1.2	Direct	nutzt als Ressource	#0070ff
MAN.3	FK2.1	Direct	nutzt als Ressource	#0070ff
MAN.3	FK2.3	Direct	nutzt als Ressource	#0070ff
MAN.3	FK2.2	Direct	nutzt als Ressource	#0070ff
MAN.3	FK2.4	Direct	nutzt als Ressource	#0070ff
MAN.3	FK2.5	Direct	nutzt als Ressource	#0070ff
MAN.3	FK3.1	Direct	nutzt als Ressource	#0070ff
MAN.3	PK2.2	Direct	nutzt als Ressource	#0070ff
MAN.3	PK2.3	Direct	nutzt als Ressource	#0070ff
MAN.3	PK3.1	Direct	nutzt als Ressource	#0070ff
MAN.3	PK3.2	Direct	nutzt als Ressource	#0070ff
MAN.3	PK5.2	Direct	nutzt als Ressource	#0070ff
MAN.3	PK5.3	Direct	nutzt als Ressource	#0070ff
MAN.3	PK6.1	Direct	nutzt als Ressource	#0070ff
MAN.3	SK1.1	Direct	nutzt als Ressource	#0070ff
MAN.3	SK1.2	Direct	nutzt als Ressource	#0070ff
MAN.3	SK1.3	Direct	nutzt als Ressource	#0070ff
MAN.3	SK2.2	Direct	nutzt als Ressource	#0070ff

Source	Target	Type	Label	Color
MAN.3	SK2.3	Direct	nutzt als Ressource	#0070ff
MAN.3	SK2.4	Direct	nutzt als Ressource	#0070ff
MAN.3	MK1.2	Direct	nutzt als Ressource	#0070ff
MAN.3	MK3.1	Direct	nutzt als Ressource	#0070ff
SYS.1	FK1.1	Direct	nutzt als Ressource	#0070ff
SYS.1	FK1.2	Direct	nutzt als Ressource	#0070ff
SYS.1	FK2.1	Direct	nutzt als Ressource	#0070ff
SYS.1	FK2.3	Direct	nutzt als Ressource	#0070ff
SYS.1	FK3.1	Direct	nutzt als Ressource	#0070ff
SYS.1	PK3.1	Direct	nutzt als Ressource	#0070ff
SYS.1	PK3.2	Direct	nutzt als Ressource	#0070ff
SYS.1	PK4.1	Direct	nutzt als Ressource	#0070ff
SYS.1	PK5.1	Direct	nutzt als Ressource	#0070ff
SYS.1	PK5.2	Direct	nutzt als Ressource	#0070ff
SYS.1	PK6.1	Direct	nutzt als Ressource	#0070ff
SYS.1	SK1.1	Direct	nutzt als Ressource	#0070ff
SYS.1	SK2.4	Direct	nutzt als Ressource	#0070ff
SYS.1	MK1.1	Direct	nutzt als Ressource	#0070ff
SYS.2	K1.1	Direct	nutzt als Ressource	#0070ff
SYS.2	FK1.2	Direct	nutzt als Ressource	#0070ff
SYS.2	FK2.1	Direct	nutzt als Ressource	#0070ff
SYS.2	FK2.3	Direct	nutzt als Ressource	#0070ff
SYS.2	FK3.1	Direct	nutzt als Ressource	#0070ff
SYS.2	PK3.1	Direct	nutzt als Ressource	#0070ff
SYS.2	PK3.3	Direct	nutzt als Ressource	#0070ff
SYS.2	PK4.1	Direct	nutzt als Ressource	#0070ff
SYS.2	PK4.2	Direct	nutzt als Ressource	#0070ff
SYS.2	PK5.2	Direct	nutzt als Ressource	#0070ff
SYS.2	MK1.1	Direct	nutzt als Ressource	#0070ff
SYS.2	MK2.1	Direct	nutzt als Ressource	#0070ff
SYS.3	FK1.1	Direct	nutzt als Ressource	#0070ff
SYS.3	FK1.2	Direct	nutzt als Ressource	#0070ff
SYS.3	FK2.1	Direct	nutzt als Ressource	#0070ff
SYS.3	FK2.3	Direct	nutzt als Ressource	#0070ff
SYS.3	FK3.2	Direct	nutzt als Ressource	#0070ff
SYS.3	FK3.3	Direct	nutzt als Ressource	#0070ff
SYS.3	FK3.4	Direct	nutzt als Ressource	#0070ff
SYS.3	PK3.1	Direct	nutzt als Ressource	#0070ff
SYS.3	PK3.2	Direct	nutzt als Ressource	#0070ff
SYS.3	PK4.1	Direct	nutzt als Ressource	#0070ff
SYS.3	PK4.2	Direct	nutzt als Ressource	#0070ff
SYS.3	PK5.2	Direct	nutzt als Ressource	#0070ff
SYS.3	MK2.1	Direct	nutzt als Ressource	#0070ff
SYS.3	MK2.2	Direct	nutzt als Ressource	#0070ff

Source	Target	Type	Label	Color
SYS.3	MK2.3	Direct	nutzt als Ressource	#0070ff
SYS.4	FK1.1	Direct	nutzt als Ressource	#0070ff
SYS.4	FK1.2	Direct	nutzt als Ressource	#0070ff
SYS.4	FK2.1	Direct	nutzt als Ressource	#0070ff
SYS.4	FK2.3	Direct	nutzt als Ressource	#0070ff
SYS.4	FK3.1	Direct	nutzt als Ressource	#0070ff
SYS.4	FK3.2	Direct	nutzt als Ressource	#0070ff
SYS.4	FK3.3	Direct	nutzt als Ressource	#0070ff
SYS.4	FK3.4	Direct	nutzt als Ressource	#0070ff
SYS.4	PK3.1	Direct	nutzt als Ressource	#0070ff
SYS.4	PK3.3	Direct	nutzt als Ressource	#0070ff
SYS.4	MK2.2	Direct	nutzt als Ressource	#0070ff
SYS.4	MK2.3	Direct	nutzt als Ressource	#0070ff
SYS.5	FK1.1	Direct	nutzt als Ressource	#0070ff
SYS.5	FK1.2	Direct	nutzt als Ressource	#0070ff
SYS.5	FK2.1	Direct	nutzt als Ressource	#0070ff
SYS.5	FK2.3	Direct	nutzt als Ressource	#0070ff
SYS.5	FK3.2	Direct	nutzt als Ressource	#0070ff
SYS.5	FK3.3	Direct	nutzt als Ressource	#0070ff
SYS.5	FK3.4	Direct	nutzt als Ressource	#0070ff
SYS.5	PK3.3	Direct	nutzt als Ressource	#0070ff
SYS.5	PK5.2	Direct	nutzt als Ressource	#0070ff
SYS.5	MK2.2	Direct	nutzt als Ressource	#0070ff
SYS.5	MK2.3	Direct	nutzt als Ressource	#0070ff
SWE.1	FK1.1	Direct	nutzt als Ressource	#0070ff
SWE.1	FK1.2	Direct	nutzt als Ressource	#0070ff
SWE.1	FK2.1	Direct	nutzt als Ressource	#0070ff
SWE.1	FK2.3	Direct	nutzt als Ressource	#0070ff
SWE.1	FK3.1	Direct	nutzt als Ressource	#0070ff
SWE.1	PK3.1	Direct	nutzt als Ressource	#0070ff
SWE.1	PK3.3	Direct	nutzt als Ressource	#0070ff
SWE.1	PK5.1	Direct	nutzt als Ressource	#0070ff
SWE.1	PK5.2	Direct	nutzt als Ressource	#0070ff
SWE.1	MK1.1	Direct	nutzt als Ressource	#0070ff
SWE.2	FK1.1	Direct	nutzt als Ressource	#0070ff
SWE.2	FK1.2	Direct	nutzt als Ressource	#0070ff
SWE.2	FK2.1	Direct	nutzt als Ressource	#0070ff
SWE.2	FK2.3	Direct	nutzt als Ressource	#0070ff
SWE.2	FK3.2	Direct	nutzt als Ressource	#0070ff
SWE.2	FK3.3	Direct	nutzt als Ressource	#0070ff
SWE.2	PK3.1	Direct	nutzt als Ressource	#0070ff
SWE.2	PK3.2	Direct	nutzt als Ressource	#0070ff
SWE.2	PK4.1	Direct	nutzt als Ressource	#0070ff
SWE.2	PK4.2	Direct	nutzt als Ressource	#0070ff

Source	Target	Type	Label	Color
SWE.2	PK5.2	Direct	nutzt als Ressource	#0070ff
SWE.2	MK2.1	Direct	nutzt als Ressource	#0070ff
SWE.2	MK2.2	Direct	nutzt als Ressource	#0070ff
SWE.2	MK2.3	Direct	nutzt als Ressource	#0070ff
SWE.3	FK1.1	Direct	nutzt als Ressource	#0070ff
SWE.3	FK1.2	Direct	nutzt als Ressource	#0070ff
SWE.3	FK2.1	Direct	nutzt als Ressource	#0070ff
SWE.3	FK2.3	Direct	nutzt als Ressource	#0070ff
SWE.3	FK3.2	Direct	nutzt als Ressource	#0070ff
SWE.3	FK3.3	Direct	nutzt als Ressource	#0070ff
SWE.3	PK3.3	Direct	nutzt als Ressource	#0070ff
SWE.3	PK5.2	Direct	nutzt als Ressource	#0070ff
SWE.3	MK2.2	Direct	nutzt als Ressource	#0070ff
SWE.4	FK1.1	Direct	nutzt als Ressource	#0070ff
SWE.4	FK1.2	Direct	nutzt als Ressource	#0070ff
SWE.4	FK2.1	Direct	nutzt als Ressource	#0070ff
SWE.4	FK2.3	Direct	nutzt als Ressource	#0070ff
SWE.4	FK3.2	Direct	nutzt als Ressource	#0070ff
SWE.4	FK3.3	Direct	nutzt als Ressource	#0070ff
SWE.4	PK3.3	Direct	nutzt als Ressource	#0070ff
SWE.4	PK5.2	Direct	nutzt als Ressource	#0070ff
SWE.4	MK2.2	Direct	nutzt als Ressource	#0070ff
SWE.5	FK1.1	Direct	nutzt als Ressource	#0070ff
SWE.5	FK1.2	Direct	nutzt als Ressource	#0070ff
SWE.5	FK2.1	Direct	nutzt als Ressource	#0070ff
SWE.5	FK2.3	Direct	nutzt als Ressource	#0070ff
SWE.5	FK3.2	Direct	nutzt als Ressource	#0070ff
SWE.5	FK3.3	Direct	nutzt als Ressource	#0070ff
SWE.5	PK3.3	Direct	nutzt als Ressource	#0070ff
SWE.5	PK5.2	Direct	nutzt als Ressource	#0070ff
SWE.5	MK2.2	Direct	nutzt als Ressource	#0070ff
SWE.6	FK1.1	Direct	nutzt als Ressource	#0070ff
SWE.6	FK1.2	Direct	nutzt als Ressource	#0070ff
SWE.6	FK2.1	Direct	nutzt als Ressource	#0070ff
SWE.6	FK2.3	Direct	nutzt als Ressource	#0070ff
SWE.6	FK3.2	Direct	nutzt als Ressource	#0070ff
SWE.6	FK3.3	Direct	nutzt als Ressource	#0070ff
SWE.6	PK3.3	Direct	nutzt als Ressource	#0070ff
SWE.6	PK5.2	Direct	nutzt als Ressource	#0070ff
SWE.6	MK2.2	Direct	nutzt als Ressource	#0070ff
SUP.1	FK1.1	Direct	nutzt als Ressource	#0070ff
SUP.1	FK1.2	Direct	nutzt als Ressource	#0070ff
SUP.1	FK2.1	Direct	nutzt als Ressource	#0070ff
SUP.1	FK2.3	Direct	nutzt als Ressource	#0070ff

Source	Target	Type	Label	Color
SUP.1	FK3.1	Direct	nutzt als Ressource	#0070ff
SUP.1	PK3.3	Direct	nutzt als Ressource	#0070ff
SUP.1	PK5.2	Direct	nutzt als Ressource	#0070ff
SUP.1	SK1.2	Direct	nutzt als Ressource	#0070ff
SUP.1	SK3.1	Direct	nutzt als Ressource	#0070ff
SUP.1	MK1.2	Direct	nutzt als Ressource	#0070ff
SUP.8	FK1.1	Direct	nutzt als Ressource	#0070ff
SUP.8	FK1.2	Direct	nutzt als Ressource	#0070ff
SUP.8	FK2.1	Direct	nutzt als Ressource	#0070ff
SUP.8	FK2.3	Direct	nutzt als Ressource	#0070ff
SUP.8	FK3.1	Direct	nutzt als Ressource	#0070ff
SUP.8	FK3.2	Direct	nutzt als Ressource	#0070ff
SUP.8	FK3.3	Direct	nutzt als Ressource	#0070ff
SUP.8	FK3.4	Direct	nutzt als Ressource	#0070ff
SUP.8	PK3.3	Direct	nutzt als Ressource	#0070ff
SUP.8	MK2.1	Direct	nutzt als Ressource	#0070ff
SUP.8	MK2.2	Direct	nutzt als Ressource	#0070ff
SUP.8	MK2.3	Direct	nutzt als Ressource	#0070ff
SUP.9	FK1.1	Direct	nutzt als Ressource	#0070ff
SUP.9	FK1.2	Direct	nutzt als Ressource	#0070ff
SUP.9	FK2.1	Direct	nutzt als Ressource	#0070ff
SUP.9	FK2.3	Direct	nutzt als Ressource	#0070ff
SUP.9	FK3.1	Direct	nutzt als Ressource	#0070ff
SUP.9	PK3.3	Direct	nutzt als Ressource	#0070ff
SUP.9	PK5.2	Direct	nutzt als Ressource	#0070ff
SUP.9	MK1.1	Direct	nutzt als Ressource	#0070ff
SUP.10	FK1.1	Direct	nutzt als Ressource	#0070ff
SUP.10	FK1.2	Direct	nutzt als Ressource	#0070ff
SUP.10	FK2.1	Direct	nutzt als Ressource	#0070ff
SUP.10	FK2.3	Direct	nutzt als Ressource	#0070ff
SUP.10	FK3.1	Direct	nutzt als Ressource	#0070ff
SUP.10	PK3.1	Direct	nutzt als Ressource	#0070ff
SUP.10	PK5.2	Direct	nutzt als Ressource	#0070ff
SUP.10	MK1.2	Direct	nutzt als Ressource	#0070ff
ACQ.4	FK1.1	Direct	nutzt als Ressource	#0070ff
ACQ.4	FK1.2	Direct	nutzt als Ressource	#0070ff
ACQ.4	FK2.1	Direct	nutzt als Ressource	#0070ff
ACQ.4	FK2.3	Direct	nutzt als Ressource	#0070ff
ACQ.4	FK3.1	Direct	nutzt als Ressource	#0070ff
ACQ.4	PK3.1	Direct	nutzt als Ressource	#0070ff
ACQ.4	PK3.3	Direct	nutzt als Ressource	#0070ff
ACQ.4	PK5.2	Direct	nutzt als Ressource	#0070ff
ACQ.4	PK6.1	Direct	nutzt als Ressource	#0070ff
ACQ.4	SK1.1	Direct	nutzt als Ressource	#0070ff

Source	Target	Type	Label	Color
ACQ.4	SK1.2	Direct	nutzt als Ressource	#0070ff
ACQ.4	SK1.3	Direct	nutzt als Ressource	#0070ff
ACQ.4	SK2.2	Direct	nutzt als Ressource	#0070ff
ACQ.4	MK1.2	Direct	nutzt als Ressource	#0070ff
ACQ.4	MK3.1	Direct	nutzt als Ressource	#0070ff
FK.1	FK1.1	Direct	unterteilt sich in	#FFC300
FK.1	FK1.2	Direct	unterteilt sich in	#FFC300
FK2	FK2.1	Direct	unterteilt sich in	#FFC300
FK2	FK2.2	Direct	unterteilt sich in	#FFC300
FK2	FK2.3	Direct	unterteilt sich in	#FFC300
FK2	FK2.4	Direct	unterteilt sich in	#FFC300
FK2	FK2.5	Direct	unterteilt sich in	#FFC300
FK3	FK3.1	Direct	unterteilt sich in	#FFC300
FK3	FK3.2	Direct	unterteilt sich in	#FFC300
FK3	FK3.3	Direct	unterteilt sich in	#FFC300
FK3	FK3.4	Direct	unterteilt sich in	#FFC300
FK4	FK4.1	Direct	unterteilt sich in	#FFC300
FK4	FK4.2	Direct	unterteilt sich in	#FFC300
FK4	FK4.3	Direct	unterteilt sich in	#FFC300
PK1	PK1.1	Direct	unterteilt sich in	#FFC300
PK1	PK1.2	Direct	unterteilt sich in	#FFC300
PK1	PK1.3	Direct	unterteilt sich in	#FFC300
PK2	PK2.1	Direct	unterteilt sich in	#FFC300
PK2	PK2.2	Direct	unterteilt sich in	#FFC300
PK2	PK2.3	Direct	unterteilt sich in	#FFC300
PK3	PK3.1	Direct	unterteilt sich in	#FFC300
PK3	PK3.2	Direct	unterteilt sich in	#FFC300
PK3	PK3.3	Direct	unterteilt sich in	#FFC300
PK4	PK4.1	Direct	unterteilt sich in	#FFC300
PK4	PK4.2	Direct	unterteilt sich in	#FFC300
PK5	PK5.1	Direct	unterteilt sich in	#FFC300
PK5	PK5.2	Direct	unterteilt sich in	#FFC300
PK5	PK5.3	Direct	unterteilt sich in	#FFC300
PK5	PK5.4	Direct	unterteilt sich in	#FFC300
PK6	PK6.1	Direct	unterteilt sich in	#FFC300
SK1	SK1.1	Direct	unterteilt sich in	#FFC300
SK1	SK1.2	Direct	unterteilt sich in	#FFC300
SK1	SK1.3	Direct	unterteilt sich in	#FFC300
SK2	SK2.1	Direct	unterteilt sich in	#FFC300
SK2	SK2.2	Direct	unterteilt sich in	#FFC300
SK2	SK2.3	Direct	unterteilt sich in	#FFC300
SK2	SK2.4	Direct	unterteilt sich in	#FFC300
SK3	SK3.1	Direct	unterteilt sich in	#FFC300
SK3	SK3.2	Direct	unterteilt sich in	#FFC300

Source	Target	Type	Label	Color
MK1	MK1.1	Direct	unterteilt sich in	#FFC300
MK1	MK1.2	Direct	unterteilt sich in	#FFC300
MK2	MK2.1	Direct	unterteilt sich in	#FFC300
MK2	MK2.2	Direct	unterteilt sich in	#FFC300
MK2	MK2.3	Direct	unterteilt sich in	#FFC300
MK3	MK3.1	Direct	unterteilt sich in	#FFC300



PRODUKT
SICHERHEIT
QUALITÄT



BERGISCHE
UNIVERSITÄT
WUPPERTAL