

1 Einleitung

1.1 Ausgangssituation und Problemstellung

Die digitale Transformation durchdringt alle Branchen und definiert neu, wie Unternehmen durch den Einsatz digitaler Technologien Werte schaffen (s. HESS ET AL. 2016). Digitale Technologien und die damit verbundenen Industrie 4.0-Konzepte sind für produzierende Unternehmen essenziell, um wettbewerbsfähig zu bleiben (s. SCHUH ET AL. 2020). Produzierende Unternehmen stehen bei dem Einsatz von digitalen Technologien allerdings vor der Herausforderung, die steigende Nachfrage nach digitalen Lösungen zu erfüllen (s. HOOGSTEEEN U. BORGMAN 2022), da ein internationaler Mangel an IT-Fachkräften, insbesondere im Bereich der Softwareentwicklung, herrscht (s. NITHIHANATCHINNAPAT U. JOSHI 2019). Die steigende Nachfrage und der gleichzeitige Mangel an qualifizierten IT-Ressourcen, führt zu Engpässen und einer zunehmenden Diskrepanz zwischen IT-Nachfrage und -Angebot (s. LEBENS ET AL. 2022). Gleichzeitig führt die fortschreitende Digitalisierung zu spezifischeren Problemen, wodurch das Einbeziehen der Nutzer von digitalen Technologien zunehmend wichtiger wird (s. URBACH U. AHLEMANN 2019). Wenn IT-Abteilungen der IT-Nachfrage der Nutzer nicht gerecht werden können, greifen Fachabteilungen zunehmend auf sogenannte Schatten-IT zurück (s. KNOLL U. STRAHRINGER 2017). Obwohl diese Schattenanwendungen anfangs die Produktivität steigern, werden die Anwendungen nicht von der IT-Abteilung unterstützt und können zu IT-Sicherheitsproblemen und einem Mangel an Transparenz führen (s. HEUER ET AL. 2022).

Als Lösungsansatz kann die IT-Abteilung ihre Organisation agiler gestalten und die IT-Architektur im Unternehmen modernisieren (s. KNOLL U. STRAHRINGER 2017). Low-Code-Plattformen, mit einer grafischen Oberfläche und vorgefertigten Code-Bausteinen, ermöglichen es auch Nicht-IT-Spezialist*innen, digitale Lösungen zu entwickeln (siehe Kapitel 2.1.2) und somit zur digitalen Transformation ihrer Organisationen beizutragen (s. IHO ET AL. 2021). Dies ermöglicht einen schnelleren und effizienteren Entwicklungsprozess, da die Anforderungen nicht vollständig zwischen den Fachbereichen und der IT abgestimmt und übersetzt werden müssen (s. CARROLL ET AL. 2022). IT-Abteilungen können so entlastet werden. Abbildung 1-1 zeigt die drei Hauptgründe für die Verwendung von Low-Code-Plattformen.

Hauptgründe für die Verwendung von Low-Code Plattformen

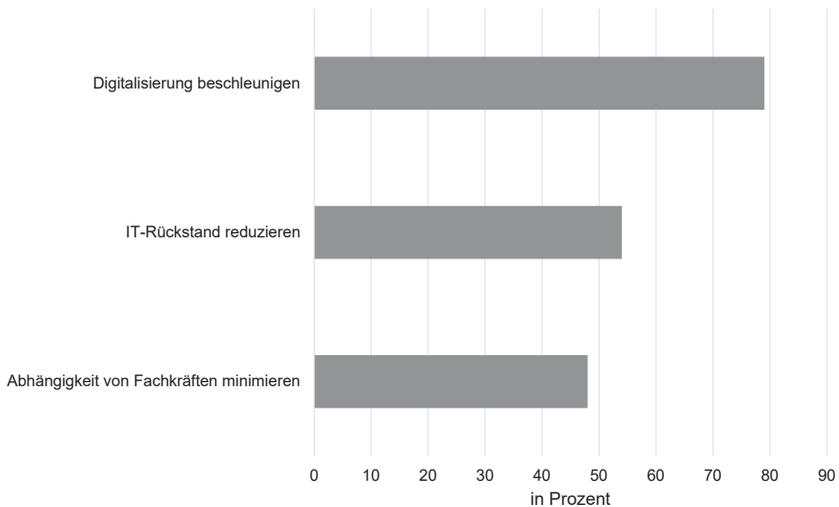


Abbildung 1-1: Hauptgründe für die Verwendung von Low-Code-Plattformen (i. A. a. JUHAS ET AL. 2022)

Durch Low-Code-Plattformen kann die Abhängigkeit von IT-Fachkräften minimiert und der IT-Rückstand reduziert werden, was zu einer beschleunigten Digitalisierung des Unternehmens führen kann (s. JUHAS ET AL. 2022). Trotz dieser Vorteile und des erheblichen Potenzials von Low-Code-Plattformen liegt die Softwareentwicklung in vielen Unternehmen weiterhin vorrangig in den Händen der IT-Abteilungen (s. TALESRA U. G. S. 2021). Oft besteht bei den Fachbereichen Zurückhaltung, digitale Lösungen eigenständig zu entwickeln, selbst wenn ein signifikanter Rückstau an weniger priorisierten IT-Projekten existiert (s. TECHCONSULT GMBH 2021). Datensicherheit und Datenschutz werden dabei als einer der Hauptgründe genannt, derzeit keine Low-Code-Plattformen einzusetzen oder deren Einsatz auch in Zukunft auszuschließen (s. TECHCONSULT GMBH 2021). Darüber hinaus stellt die richtige Auswahl der Low-Code-Plattformen eine Herausforderung dar. Unterschiedliche Low-Code-Plattformen haben verschiedene Beschränkungen in Bezug auf die Anpassungsfähigkeit und eignen sich unterschiedlich gut für die Entwicklung komplexer Anwendungen (s. JUHAS ET AL. 2022). Da der Funktionsumfang der Low-Code-Plattformen stark variiert, eignet sich je nach Art der Low-Code-Anwendung eine unterschiedliche Low-Code-Plattform (s. SCHAFFRY 2023).

In der Praxis zeigt sich, dass Geschäft und IT ihre jeweilige Rolle in Bezug auf den Einsatz von verschiedenen Low-Code-Anwendungen noch finden müssen und dass sie daher ihre Zusammenarbeitsmentalität ändern müssen (s. PRINZ ET AL. 2022). Dabei ist es von entscheidender Bedeutung, dass sich die Low-Code-Entwickler*innen aus den Fachbereichen der von der IT festgelegten Richtlinien und Standards bewusst

sind (s. BINZER U. WINKLER 2022). Das Fehlen von Governance- und Überwachungsmechanismen beim Einsatz von Low-Code führt zu einem unkontrollierten Einsatz von Datenschnittstellen und zu einem Mangel an Kontrolle über die Anwendungsentwicklung und -bereitstellung (s. HEUER ET AL. 2022). Diese Situation birgt Risiken in Bezug auf Datensicherheit, Compliance und das Potenzial für die Schaffung von Schatten-IT-Umgebungen (s. HEUER ET AL. 2022).

Um die Nutzung von Low-Code-Plattformen effektiv zu steuern und zu überwachen, bedarf es einer starken IT-Governance Struktur, um sicherzustellen, dass entwickelte Anwendungen den Unternehmensstandards entsprechen und Sicherheitsrisiken minimiert werden (s. SANCHIS ET AL. 2020). Eine Herausforderung bei der Nutzung von Low-Code-Plattformen ist es, die Konsistenz und Qualität von Anwendungen zu gewährleisten. Insbesondere dann, wenn Low-Code-Entwickler*innen ohne tiefe technische Kenntnisse involviert sind (s. KHORRAM ET AL. 2020). Es müssen Regeln und Maßnahmen bereitgestellt werden, die eine qualitativ hochwertige und sichere Anwendungsentwicklung fördern, während gleichzeitig die Flexibilität und Agilität, die Low-Code-Plattformen bieten, erhalten bleiben (s. KHORRAM ET AL. 2020). Dies erfordert eine Anpassung der bestehenden IT-Governance mit gut definierten Entscheidungsprozessen, Rollen und Verantwortlichkeiten sowie relationalen Mechanismen für die Implementierung und den Betrieb von Low-Code (s. PRINZ ET AL. 2022). Die Wissenschaft weist jedoch auf aktuelle Forschungslücken hinsichtlich der IT-Governance von Low-Code-Plattformen hin (s. PRINZ ET AL. 2022).

Um die bestehenden Forschungslücken zu schließen und praktischen Nutzen zu generieren, ist es notwendig, dass die Autorin dieser Dissertation den Einfluss unterschiedlicher Low-Code-Anwendungsfälle auf Unternehmen sowohl technisch als auch organisatorisch beleuchtet. Dazu wird untersucht, welche Low-Code-Anwendungsfälle in der Praxis in Unternehmen vorkommen. Anschließend werden die Risiken beim Einsatz von Low-Code beschrieben und in Form von Regeln zur Risikovermeidung festgehalten. Im nächsten Schritt wird die Wirkbeziehung zwischen den verschiedenen Low-Code-Anwendungsfällen und den Low-Code-Regeln erklärt. Die gewonnenen Erkenntnisse werden anschließend in ein strukturiertes Regelwerk überführt, das neben den zuvor identifizierten Regeln auch Gestaltungsmaßnahmen enthält, die beschreiben, wie die Regeln anwendungsspezifisch eingehalten werden können. Durch dieses Vorgehen wird untersucht, wie sich die Implementierung und der Einsatz von Low-Code-Anwendungen auf die IT-Architektur sowie auf die Organisationsstruktur von Unternehmen auswirken. Das Regelwerk soll dazu dienen, die bestehende IT-Governance von produzierenden Unternehmen in Bezug auf Low-Code zu erweitern.

1.2 Zielsetzung und Forschungsfrage

Im Kontext der zuvor beschriebenen Ausgangslage und der identifizierten Problemstellung verfolgt diese Dissertation das Ziel, ein Regelwerk für Low-Code-Anwendung in produzierenden Unternehmen zu entwickeln. Dieses Regelwerk soll präzise Regeln und unterstützende Maßnahmen für spezifische Anwendungsfälle bereitstellen und als

Erweiterung der bestehenden IT-Governance verwendet werden. Es wird ein systematischer Ansatz vorgeschlagen, der für verschiedene Low-Code-Anwendungsfalltypen spezifische Low-Code-Regeln liefert. Auf dieser Grundlage können anwendungsfallspezifische Maßnahmen abgeleitet werden. Daraus ergibt sich die zentrale Forschungsfrage:

Welche Regeln sollte ein produzierendes Unternehmen für Low-Code-Anwendung mindestens festlegen?

Diese Hauptforschungsfrage wird durch vier detaillierte und aktionsorientierte Unterfragen ergänzt und vertieft. Die Zielsetzung, Hauptforschungsfrage sowie die vier Unterfragen sind in Abbildung 1-2 dargestellt.

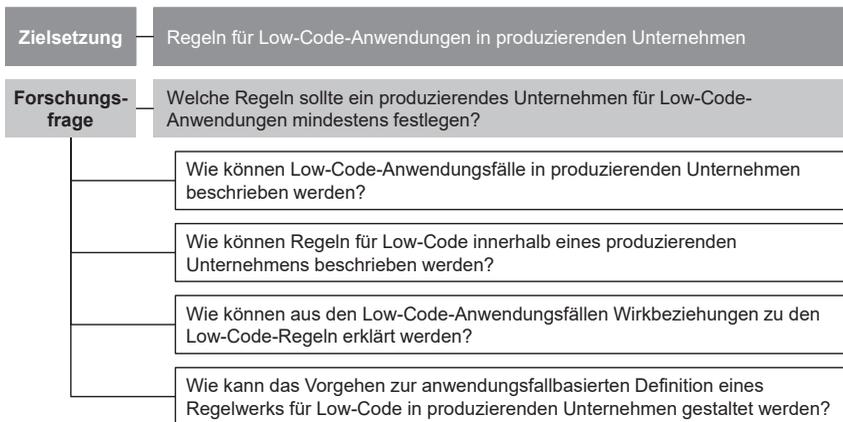


Abbildung 1-2: Zielsetzung und Forschungsfragen der vorliegenden Arbeit (eigene Darstellung)

Das erste Teilziel zur Erreichung des Forschungsziels besteht in der systematischen Analyse, Identifikation und Strukturierung von Low-Code-Anwendungsfällen. Die technologische Entwicklung im Bereich Low-Code gewinnt zunehmend an Bedeutung für die digitale Transformation produzierender Unternehmen. Diese Entwicklung zeichnet sich durch ein breites Spektrum an Low-Code-Anwendung aus. Es existieren keine Typisierungsansätze für die verschiedenen Arten von Low-Code-Anwendung. Aufgrund der schnellen Entwicklungsfortschritte fehlt ein klares Verständnis dafür, wie verschiedene Anwendungsmerkmale die Organisationsstruktur und die IT-Architektur eines Unternehmens beeinflussen. Deshalb ist die Entwicklung eines konsistenten Verständnisses für die verschiedenen Arten von Low-Code-Anwendungsfällen erforderlich.

Das zweite Teilziel konzentriert sich darauf, relevante Regeln für die Implementierung und den Betrieb von Low-Code-Anwendung zu präzisieren. Durch die Integration von bestehenden IT-Governance Rahmenwerken können Strukturierungsdimensionen für das Low-Code-Regelwerk identifiziert werden. Basierend auf den in der Literatur

identifizierten Risiken und Herausforderungen von Low-Code können relevante Low-Code-Regeln für produzierende Unternehmen abgeleitet werden.

Das dritte Teilziel befasst sich mit der Untersuchung der gegenseitigen Einflüsse zwischen den verschiedenen Typen von Low-Code-Anwendungsfällen und den Low-Code-Regeln. Entsprechend einem praxisnahen Forschungsansatz sollen die erzielten Erkenntnisse genutzt werden, um geeignete Maßnahmen zur Adressierung der identifizierten Risiken und Herausforderungen zu entwickeln.

Das vierte Teilziel strebt danach aus den zuvor identifizierten Maßnahmen konkrete Gestaltungsempfehlungen zu formulieren. Dazu werden sowohl typenunabhängige als auch anwendungsfalltypspezifische Gestaltungsempfehlungen auf Basis der Maßnahmen präsentiert.

Zusammengefasst zielt die in dieser Arbeit entwickelte systematische Methode darauf ab, IT- und Fachbereiche zu befähigen, die Risiken der Komplexität von Low-Code-Anwendungsfällen und deren Auswirkungen auf die Organisationsstruktur und IT-Architektur mittels spezifischer Gestaltungsempfehlungen effektiv zu bewerten und gezielt zu managen. Das vorgeschlagene Regelwerk unterstützt dabei, anwendungsfallspezifische Maßnahmen zu ergreifen, um eine nachhaltige Implementierung und den Betrieb von Low-Code-Anwendung in produzierenden Unternehmen sicherzustellen.

1.3 Wissenschaftlicher Bezugsrahmen und Forschungskonzeption

Im Anschluss an die Darlegung der Ziele wird nun der wissenschaftliche Bezugsrahmen dieser Arbeit erörtert, wofür ein Verständnis der Wissenschaftstheorie, auch als „Wissenschaft der Wissenschaften“ (s. ULRICH U. HILL 1976, S. 305) bezeichnet, erforderlich ist. Abbildung 1-3 zeigt die Aufteilung der Wissenschaften nach ihren Zielen und bietet Beispiele für jede Kategorie.

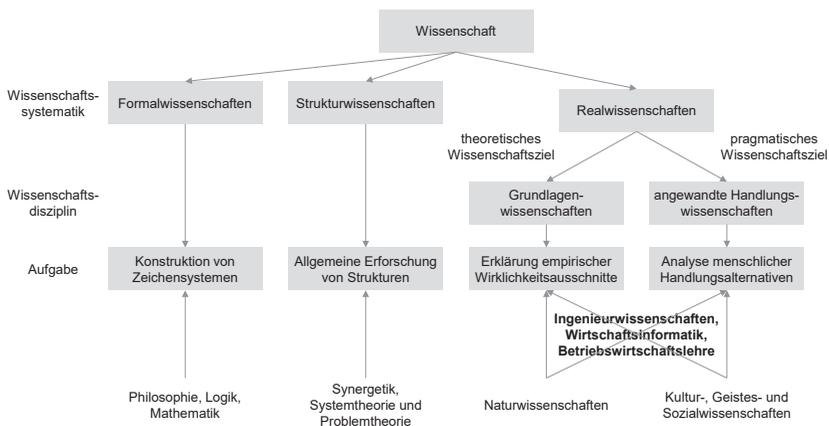


Abbildung 1-3: Wissenschaftssystematik (i. A. a. ULRICH U. HILL 1976, S. 305; ZELEWSKI 2008, S. 3; HOFFMANN 2018, S. 6)

Die Formalwissenschaften, wie zum Beispiel die Mathematik, beschäftigen sich mit der Konstruktion von wissenschaftlichen Sprachsystemen, von Sprachen und Zeichen, wie zum Beispiel die Mathematik, und verfolgen dabei eine logische Vorgehensweise (s. ULRICH U. HILL 1976, S. 305). Strukturwissenschaften, wie beispielsweise die Systemtheorie oder die Synergetik, befassen sich mit Strukturen von realen und formalen Objekten (s. ZELEWSKI 2008, S. 3). Die Realwissenschaften untersuchen dagegen real existierende Objekte und Vorkommnisse, wie es zum Beispiel die Naturwissenschaften, aber auch Geistes- und Sozialwissenschaften sowie die Wirtschaftswissenschaften tun (s. JUNG 2012, S. 22).

Die Realwissenschaften können weiter in Grundlagenwissenschaften und angewandte Handlungswissenschaften unterteilt werden. Bei den Grundlagenwissenschaften steht das Ziel der Erklärung von real existierenden Gesetzmäßigkeiten im Vordergrund (s. ZELEWSKI 2008, S. 5). Dagegen verfolgen die angewandten Handlungswissenschaften ein eher pragmatisches Wissenschaftsziel – die Anwendbarkeit der Modelle steht vor der Allgemeingültigkeit von Theorien (ULRICH 1984, S. 174s. ; LEHNER ET AL. 2008, S. 20; ÖSTERLE ET AL. 2010, S. 665). Sie beschäftigen sich daher mit der Gestaltung künftiger Realitäten, während sich die Grundlagenwissenschaften mit der Erklärung existierender Realitäten befassen (s. ULRICH 1984, S. 179).

Die vorliegende Arbeit befasst sich mit Fragestellungen der Ingenieurwissenschaften, der Betriebswirtschaftslehre und der Wirtschaftsinformatik. Ingenieurwissenschaften sind wegen der Untersuchung technisch orientierter Objekte genauso den angewandten Handlungswissenschaften zuzuordnen wie die Betriebswirtschaftslehre, die Modelle bildet, um anwendungsorientierte Handlungsempfehlungen zu geben (s. ULRICH U. HILL 1976, S. 305; WÖHE ET AL. 2016, S. 12). Die Wirtschaftsinformatik hat zwar einen strukturwissenschaftlichen Charakter, kann aber wegen ihres Anwendungsbezugs auf ein sachliches Objektsystem auch als Ingenieurwissenschaft eingeordnet werden (s. FINK ET AL. 2005, S. 256). Die in der vorliegenden Arbeit behandelten Themen der Low-Code-Anwendungsfälle und Regeln sind aus der Praxis hergeleitet und es wird die künftige Gestaltung eines Low-Code-Regelwerks erforscht. Damit und mit der Einordnung der grundsätzlichen Fragestellungen kann die vorliegende Arbeit den angewandten Handlungswissenschaften zugeordnet werden.

In den Realwissenschaften müssen grundsätzlich die Probleme der Subjektivität und der Kommunikation angegangen werden. Subjektivität bezieht sich auf die persönlichen Wahrnehmungen der Forschenden, die durch individuelle Erfahrungen und gewohnte Interpretationsmuster geprägt sind. Das Problem der Kommunikation hingegen betrifft die Herausforderung, komplexe Sachverhalte präzise und verständlich zu vermitteln, was insbesondere in abstrakten, jedoch praxisorientierten Forschungsarbeiten von großer Bedeutung ist (s. ULRICH U. HILL 1976, S. 306). Angesichts dieser Problematiken lässt sich, basierend auf der Einordnung dieser Arbeit, ein geeigneter Forschungsprozess ableiten, der auf die Überwindung dieser Herausforderungen ausgerichtet ist (siehe Abbildung 1-4).

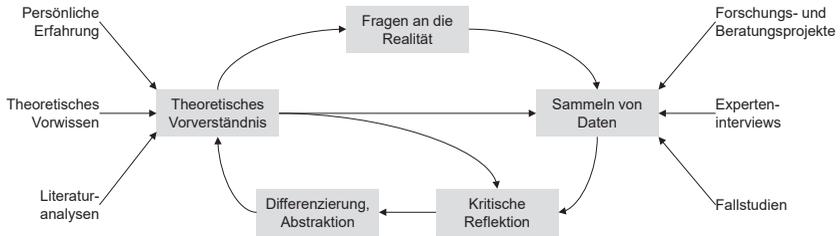


Abbildung 1-4: Forschungskonzeption (eigene Darstellung i. A. a. KUBICEK 1977, S. 14-15)

Der Forschungsprozess basiert auf dem Prinzip des zunehmenden Verständnisses der Realität nach KUBICEK. Dieser Ansatz, der ein exploratives Vorgehen favorisiert, ermöglicht es, auf der Grundlage eines vorhandenen Vorverständnisses im Forschungsfeld sowie der Erfahrungen von Expert*innen, gezielt Fragen zu stellen und auf Basis eigener Hypothesen neue Erkenntnisse zu generieren (s. EISEND U. KUß 2017, S. 12-13; KUBICEK 1977, S. 14-15).

Um dem Problem der Subjektivität zu begegnen, ist das Vorverständnis der Autorin entscheidend (s. MAYRING 2002, S. 25). Dieses ist durch deren berufliche Erfahrung unter anderem durch die Mitarbeit und Leitung von industriellen Beratungsprojekten in produzierenden Unternehmen sowie durch die Bearbeitung von praxisnahen Forschungsprojekten am Forschungsinstitut für Rationalisierung e. V. an der RWTH Aachen (FIR) geprägt. Dem in dieser Arbeit adressierten Kommunikationsproblem wird durch die Bereitstellung umfassender Definitionen im zweiten Kapitel begegnet. Präzise definierte Begrifflichkeiten sind entscheidend, um die Verlässlichkeit theoretischer Aussagen zu gewährleisten (s. EISEND U. KUß 2017, S. 14). Hierbei wird, soweit möglich, auf bereits etablierte Definitionen im Bereich des Informationsmanagements zurückgegriffen und diese werden bei Bedarf um praxisrelevante Termini erweitert.

Die sich aus dem dargelegten Forschungsprozess ergebenden Aufgaben lassen sich nach ULRICH U. HILL in terminologisch-deskriptive, empirisch-induktive und analytisch-deduktive Aufgaben aufteilen. Zu den terminologisch-deskriptiven Aufgaben zählen die Definition von Begriffen und deren Anwendung sowie die Abgrenzung des Forschungsbereichs durch Festlegung relevanter Dimensionen und die Bildung von Typologien. Im Bereich der empirisch-induktiven Aufgaben erfolgt die Hypothesengenerierung basierend auf der Beobachtung von Zusammenhängen sowie die empirische Überprüfung dieser Hypothesen. Analytisch-deduktive Aufgaben umfassen schließlich logische Prozesse, wie die Modellentwicklung und die Ableitung von situationsspezifischen Handlungsalternativen (s. ULRICH U. HILL 1976, S. 348).

Zur Forschungskonzeption dieser Arbeit werden neben den bereits genannten Methoden auch Verfahren der Sozialforschung hinzugefügt. Hierzu gehören strukturierte Literaturrecherchen sowie Expert*inneninterviews, um das Forschungsvorgehen kontinuierlich zu überprüfen und weiterzuentwickeln (s. DÜNNEBACKE 2015, S. 13).

1.4 Aufbau der Arbeit

Die Dissertation dokumentiert den Forschungsprozess in einer strukturierten Form und ist in zehn Kapitel unterteilt. Sie orientiert sich an den Phasen der angewandten Forschung gemäß ULRICH und wird in Abbildung 1-5 visualisiert (s. ULRICH 1984, S. 20).

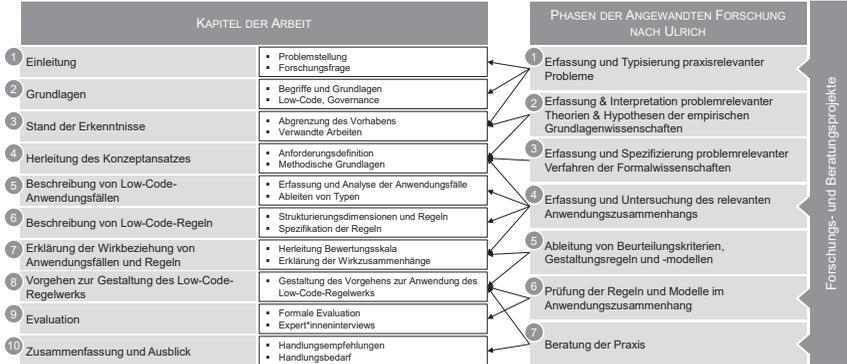


Abbildung 1-5: Phasen der angewandten Forschung nach ULRICH U. HILL (eigene Darstellung i. A. a. HOFFMANN 2018, S. 10)

Die Einleitung legt den Grundstein dieser Arbeit und wird gefolgt von Kapitel 2, das sich der Erläuterung der für das Verständnis der Leserschaft wichtigen Grundlagen widmet. Hierzu zählen die Definition zentraler Begriffe sowie die Abgrenzung des Untersuchungsbereichs der Arbeit. Kapitel 3 deckt auf Basis strukturierter Literaturrecherchen den Forschungsbedarf auf, der sich aus der Sichtung aktueller deutschsprachiger und internationaler Literatur ergibt. Es werden zudem relevante Vorarbeiten anderer Autor*innen vorgestellt, die in dieser Arbeit aufgegriffen und weitergeführt werden. Kapitel 4 etabliert die methodischen Grundpfeiler für die im Verlauf der Arbeit entwickelten Modelle, legt die Anforderungen fest und beschreibt das methodische Vorgehen.

Im Mittelpunkt der Arbeit stehen die zwei Beschreibungsmodelle. Das erste Beschreibungsmodell in Kapitel 5 konkretisiert zunächst den Begriff der Low-Code-Anwendungsfälle anhand konstituierender Merkmale. Die detaillierenden Merkmale von Low-Code-Anwendungsfällen werden im Anschluss mit ihren Ausprägungen vorgestellt. Die Typisierung der Anwendungsfälle erfolgt anhand der logischen und verifizierbaren Verknüpfung der zuvor vorgestellten Merkmalsausprägungen. Das zweite Beschreibungsmodell in Kapitel 6 widmet sich der Darstellung von Low-Code-Regeln, strukturiert nach dem IT-Governance COBIT-Kernmodell. Durch eine strukturierte Inhaltsanalyse werden einzelne Low-Code-Regeln abgeleitet und detailliert beschrieben.

Das Erklärungsmodell in Kapitel 7 verwendet die Low-Code-Anwendungsfalltypen und Regeln, um anwendungsfallspezifische Maßnahmen für die Einhaltung der Low-Code-Regeln zu identifizieren. Kapitel 8 beschäftigt sich mit der praktischen Anwendung der Modelle zur Gestaltung eines Low-Code-Regelwerks für produzierende Unternehmen.

Eine begrenzt empirisch-induktive Verifikation der Ergebnisse durch die exemplarische Anwendung der Modelle in Fallstudien wird in Kapitel 9 vorgestellt.

Den Abschluss der Arbeit bildet Kapitel 10 mit einer Zusammenfassung der zentralen Ergebnisse und einem Ausblick auf zukünftige Forschungsthemen, die aus den Erkenntnissen dieser Arbeit abgeleitet werden können.