

Einsatz von agilen Methoden in der Produkt-/Systementwicklung

Masterarbeit von
Sabine Maierhofer

Eingereicht am
Institut für Unternehmensführung und Organisation
Technische Universität Graz
o.Univ.-Prof. Dipl.-Ing. Dr.sc.techn. Reinhard Haberfellner

Graz, 07.04.2010

Zusammenfassung

Um auf ändernde Anforderungen in der Entwicklung reagieren zu können, entwickelten sich vor allem im Bereich der traditionellen Softwareentwicklung verschiedenste flexible und anpassungsfähige Praktiken. Diese sogenannten Agilen Methoden sollen dabei helfen, die Projektdurchführung reaktionsfähiger zu gestalten und somit Veränderungen offen gegenüber zu stehen.

Da bisher noch keine empirischen Ergebnisse über die Wirkung des Einsatzes dieser agilen Methoden in der Entwicklung von technischen Systemen, bestehend aus Hard- und Software, vorhanden sind, stellt sich die Frage, ob agile Methoden auch hier eine Verbesserung des Projekterfolgs bei Anforderungsänderungen erbringen.

Im Zuge dieser Masterarbeit wird ein empirischer Nachweis für den Einfluss agiler Methoden auf den Projekterfolg beim Auftreten von Anforderungsänderungen erbracht. Es werden sowohl die Auswirkungen von Anforderungsänderungen während der Entwicklung, als auch konkrete agile Konstrukte auf ihren moderierenden Effekt untersucht.

Anfangs werden die theoretischen Grundlagen für Projekte und Anforderungsänderungen im Allgemeinen betrachtet und die nötigen Parameter für die empirische Erhebung diskutiert, bevor konkrete Managementmethoden vorgestellt werden. Es werden die bedeutendsten Abläufe für traditionelle und agile Projektmanagementmodelle dargestellt.

Für die empirische Erhebung werden die einzelnen Abschnitte des Fragebogens vorgestellt, sowie die Operationalisierung der einzelnen Variablen des Projekterfolgs, der Anforderungsänderungen und agilen Methoden ausführlich erklärt. Im Anschluss erfolgt die Auswertung der erhobenen Ergebnisse, wobei mithilfe der Faktorenanalyse sechs aussagekräftige, agile Konstrukte extrahiert werden. In Verbindung mit der errechneten Kennzahl der Anforderungsänderungen werden die sechs agilen Modelle zu Interaktionstermen für die Berechnung ihres Einflusses auf den Projekterfolg zusammengefasst.

Schlussendlich wird sowohl der negative Einfluss von Anforderungsänderungen auf den Projekterfolg mithilfe der Regressionsanalyse nachgewiesen, als auch ein moderierender Effekt der agilen Konstrukte der Endproduktanpassbarkeit, effizienten Ausführung und Bereitschaft der Mitarbeiter für Veränderungen durch multiple Regressionsanalyse aufgezeigt.

Abstract

To be able to respond to changing requirements during development, particularly in the field of traditional software engineering, a variety of flexible and adaptable practices have been developed. These so-called Agile Methods are supposed to help create a more responsive project process and thus face changes more openly.

Since there are no empirical results on the effects of the application of these agile methods in the development of technical systems consisting of hard- and software, the questions arises, whether those agile methods are able to increase the project success when requirement changes occur.

Over the course of this master's thesis empirical evidence for the influence of agile methods on the success of projects will be provided. Both, the impact of requirement changes during development, as well as the moderating effect of specific agile constructs are examined.

At the beginning, the general theoretical foundations of projects and requirement changes themselves are researched, before specific management models are presented. Therefore, the most important processes of traditional and agile management models are identified.

For the empirical survey, the individual sections of the questionnaire are presented and the operationalization for each variable of project success, requirement changes and agile methods are explained. Subsequently the obtained results are evaluated, using factor analysis to extract six sound constructs. In conjunction with the calculated key figure for requirement changes, the six agile models are consolidated into interaction terms to evaluate their influence on the project success.

Finally, both a negative impact of requirement changes on project success is demonstrated by the use of regression analysis, as well as a moderating effect of the agile constructs – the final product adaptability, efficient execution and willingness to accept changes – is shown with multiple regression analysis.

Inhaltsverzeichnis

ZUSAMMENFASSUNG	I
ABSTRACT	II
1. EINLEITUNG.....	1
1.1. PROBLEMSTELLUNG UND ZIELSETZUNG.....	2
1.2. AUFBAU DER ARBEIT	3
2. THEORETISCHE GRUNDLAGEN DER UNTERSUCHUNG	4
2.1. PROJEKTDEFINITION	4
2.1.1. Abgrenzung und Begriffserklärung	4
2.1.2. Klassifizierung von Projekten.....	5
2.1.2.1. Einsatzgebiet / Zielsetzung	5
2.1.2.2. Komplexität	6
2.1.2.3. Projektphase	7
2.1.3. Stakeholder	8
2.2. ANFORDERUNGSÄNDERUNGEN.....	10
2.2.1. Einflussfaktoren auf die Anzahl der zu erwartenden Änderungen.....	10
2.2.2. Klassifizierung von Änderungen	10
2.2.2.1. Bezahlte und unbezahlte Änderungen	10
2.2.2.2. Planänderungen und Zieländerungen.....	11
2.2.3. Auswirkungen von Änderungen	11
2.3. PROJEKTERFOLG.....	13
2.3.1. Abgrenzung zum Managementerfolg.....	13
2.3.2. Messung	14
2.3.2.1. „Balanced Scorecard“	15
2.3.2.2. „Success Sliders“	16
2.4. AGILE METHODEN	18
2.4.1. Ursprung agiler Modelle	18
2.4.2. Traditionelle Managementmethoden	19
2.4.2.1. Wasserfallmodell.....	19
2.4.2.2. V-Modell	21
2.4.2.3. Spiralmodell	22
2.4.2.4. Zusammenfassung	24
2.4.3. Philosophie agiler Methoden	25
2.4.3.1. Das Agile Manifest.....	25
2.4.3.2. Agile Prinzipien.....	26
2.4.4. Agile Managementmodelle	27
2.4.4.1. Feature Driven Development.....	27
2.4.4.2. Simultaneous Engineering	30
2.4.4.3. Lean Development	31
2.4.4.4. Dynamic Systems Development Method.....	36
2.4.4.5. Extreme Programming.....	38
2.4.4.6. Scrum	42
2.4.4.7. Crystal Clear.....	45
2.4.4.8. Zusammenfassung	46
2.4.5. Agile Systeme	47

3.	EMPIRISCHE ANALYSE	48
3.1.	METHODIK DER DATENERHEBUNG.....	48
3.1.1.	Aufbau des Fragebogens.....	48
3.1.2.	Auswahl der Stichprobe.....	49
3.1.3.	Erhebungsmethode.....	50
3.1.4.	Charakteristika der Stichprobe.....	51
3.2.	OPERATIONALISIERUNG UND MESSUNG DER VARIABLEN	53
3.2.1.	Operationalisierung des Projekterfolges	53
3.2.2.	Operationalisierung der Anforderungsänderungen	55
3.2.3.	Operationalisierung der agilen Methoden.....	58
3.2.3.1.	Mitarbeiter Agilität.....	59
3.2.3.2.	Fokus auf Kundennutzen.....	60
3.2.3.3.	Zyklische Entwicklung.....	61
3.2.3.4.	Flexibilität im Prozess.....	61
3.2.3.5.	Flexibilität im Endprodukt.....	62
3.2.3.6.	Managementmodell	63
3.3.	DATENAUSWERTUNG	64
3.3.1.	Explorative Faktorenanalyse.....	64
3.3.1.1.	Anfängliche Faktorenlösung.....	66
3.3.1.2.	Finale Faktorenanalyse.....	68
3.3.2.	Regressionsanalyse	73
3.3.2.1.	Einfache lineare Regressionsanalyse	73
3.3.2.2.	Analyseergebnis „Einfluss von Anforderungsänderungen auf den Projekterfolg“	74
3.3.2.3.	Multiple lineare Regressionsanalyse	75
3.3.2.4.	Analyseergebnisse „Einfluss agiler Methoden bei Anforderungsänderungen auf den Projekterfolg“	75
3.3.2.5.	Zusammenfassung	79
3.4.	ERKENNTNISSE.....	81
4.	SCHLUSSBETRACHTUNG.....	83
4.1.	ZUSAMMENFASSUNG.....	83
4.2.	AUSBLICK	85
5.	ABKÜRZUNGSVERZEICHNIS	86
6.	FORMELVERZEICHNIS.....	87
7.	ABBILDUNGSVERZEICHNIS.....	88
8.	TABELLENVERZEICHNIS.....	89
9.	LITERATURVERZEICHNIS.....	90
10.	ANHANG	95

1. Einleitung

In den vergangenen Jahrzehnten wurden unzählige Managementmodelle entwickelt, die durch den Einsatz bestimmter Methoden in der Projektabwicklung einen erfolgreichen Abschluss versprechen. Die vorgeschlagenen agilen Ansätze sollen dabei nicht nur eine schnellere und kostengünstigere Fertigung unterstützen, sondern vor allem den stetig steigenden negativen Einflüssen von Anforderungsänderungen entgegenwirken.

Unter dem Begriff agiler Methoden sind verschiedene Prinzipien und Praktiken für die Entwicklung zusammengefasst (Sliger & Broderick, 2008, S. 9), deren Ziel es ist den Projektprozess flexibler und somit anpassungsfähiger zu gestalten. Diese neue Herangehensweise an die Projektdurchführung soll bereits zu Beginn einen ganzheitlichen Blick auf die Projektziele fördern und die Entwicklung offen für Veränderungen zu lassen.

So ist beispielsweise eine neue Art der Kundenintegration, bei der eine andauernde Zusammenarbeit mit dem Auftraggeber die gesamte Entwicklung kennzeichnet, eine spezifische agile Praktik. Ebenso kann eine zyklisch geplante Projektdurchführung als agile Herangehensweise verstanden werden, da auch hier eines der Hauptziele eine flexiblere Abwicklung der nötigen Schritte für die Erfüllung aller Anforderungen darstellt.

Ziel ist den gesamten Entwicklungsprozess in der der Softwareentwicklung flexibler zu gestalten und somit die Vorgehensweise den Bedürfnissen des Marktes anzupassen. Für die erfolgreiche Erstellung von Software ist es aufgrund der Geschwindigkeit in der sich die vorhandene Technologie ändert nicht ausreichend eine bestimmte Programmiersprache zu beherrschen (Subramaniam & Hunt, 2006, S. 28f.). Es muss der Wille existieren neue Technologie zu verstehen und einsetzen zu lernen um konkurrenzfähig zu bleiben.

In diesem Sinne darf die Erfassung der Anforderungen nicht bei Beginn des Programmierens abgeschlossen werden; neue Anforderungen während der Entwicklungszeit oder Änderungen bereits bestehender Anforderungen können durchgehend auftreten und es muss daher ein Weg gefunden werden mit diesen Herausforderungen positiv umzugehen. Werden veränderte Kundenwünsche ignoriert, so kann das erstellte Produkt zwar den initialen Anforderungen entsprechen und trotzdem den Auftraggeber nicht zufriedenstellen und somit ein Fehlschlag sein. Anforderungsänderungen müssen in das Projekt einfließen, da das Hinauszögern dazu führt, dass sich explizit geforderte Funktionen zu einem späteren Zeitpunkt nicht mehr umsetzen lassen oder bei einer späten Umsetzung das Gesamtprojekt gefährdet ist (Subramaniam & Hunt, 2006, S. 3).

Kapitel 1.1 beschreibt die zugrunde liegende Fragestellung und resultierende Zielsetzung dieser Masterarbeit, bevor in Kapitel 1.2 ein kurzer Überblick über deren Aufbau gegeben wird.

1.1. Problemstellung und Zielsetzung

Aufgrund der Vielzahl integrierter Systeme, bestehend aus Hard- und Software, wird die Frage aufgeworfen, ob sich die agilen Ansätze aus der traditionellen Softwareentwicklung erfolgsversprechend auf die Entwicklung des Gesamtsystems übertragen lassen. „Extreme Programming“ stellt für traditionelle Softwareprojekte dabei das wohl bekannteste agile Modell dar, mit dem eine völlig neue Denkweise in der Projektdurchführung als auch Zusammenarbeit geprägt wurde.

Da bisher keine Ergebnisse für diesen Projekttyp vorhanden sind, ist es das Ziel der vorliegenden Arbeit sowohl die Auswirkung von Anforderungsänderungen auf den Projekterfolg, als auch die Einflüsse konkreter agiler Entwicklungsmethoden auf die Entwicklung zu untersuchen. Es soll erforscht werden, ob ein empirischer Nachweis für eine positive Beeinflussung des Projekterfolges bei Anforderungsänderungen durch die Anwendung agiler Methoden erbracht werden kann.

Abbildung 1.1 zeigt das zu untersuchende Erklärungsmodell für diese Arbeit, welches von der Annahme ausgeht, dass der negative Einfluss von Anforderungsänderungen auf den Projekterfolg durch den Einsatz agiler Methoden moderiert werden kann.

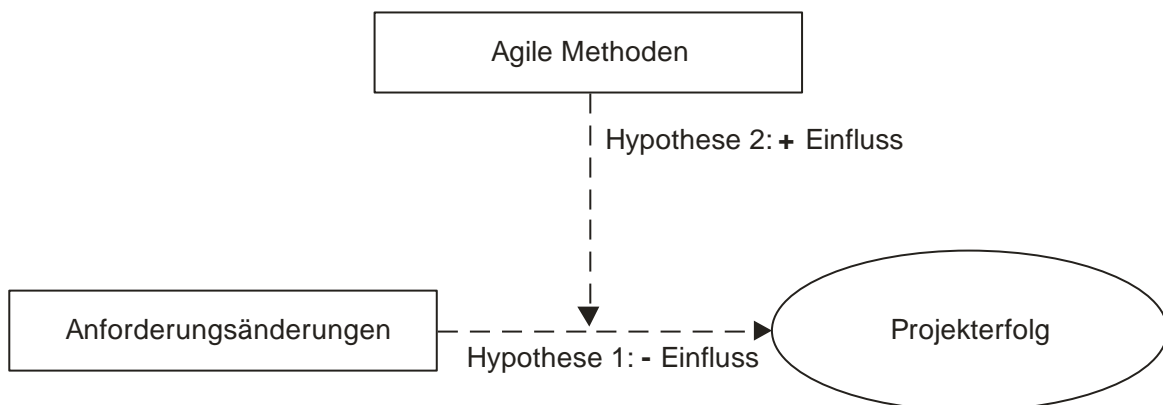


Abbildung 1.1: Zu betrachtendes Erklärungsmodell

Das Erklärungsmodell wird anhand einer quantitativen Studie verschiedener Unternehmen im Bereich der Entwicklung von kombinierten Hard- und Softwaresystemen überprüft, wobei die relevanten Erfolgsfaktoren, Anforderungsänderungen und spezifischen Entwicklungsmethoden erhoben werden.

Es wird zuerst überprüft, ob Anforderungsänderungen einen nachweislichen negativen Einfluss auf den Projekterfolg besitzen. In einem zweiten Schritt werden einzelne agile Praktiken auf ihren positiven Einfluss auf Anforderungsänderungen und somit den Projekterfolg betrachtet.

Die bisher nur vage formulierten Thesen sollen anhand theoretischer Grundlagen genauer erklärt und ihre Aussagekraft mithilfe der empirischen Auswertung überprüft werden.

1.2. Aufbau der Arbeit

In Abbildung 1.2 ist der schematische Aufbau der vorliegenden Arbeit dargestellt.

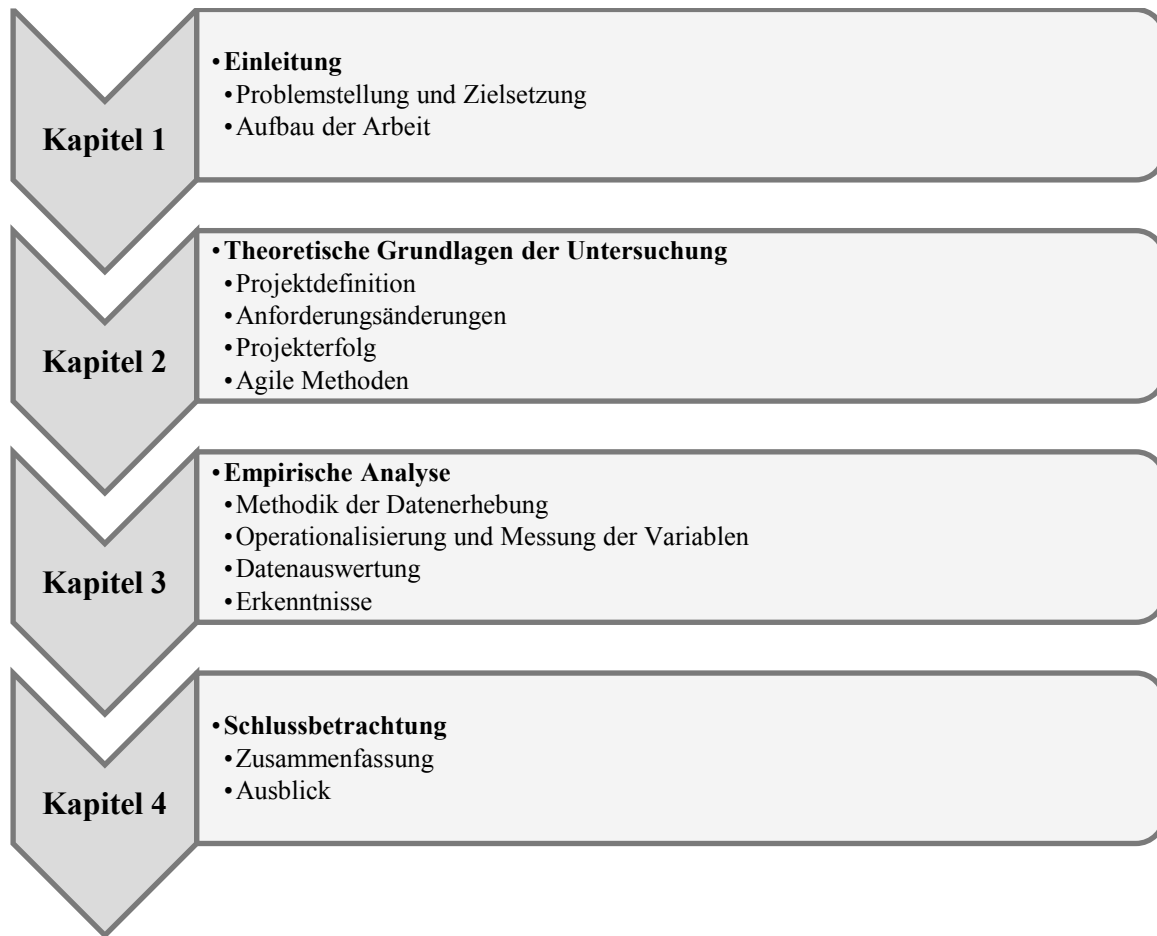


Abbildung 1.2: Aufbau der Arbeit

In diesem Kapitel wird ein kurzer Überblick über die zugrunde liegende Problemstellung und Zielsetzung gegeben. Anschließend wird in Kapitel 2 detailliert auf die theoretischen Grundlagen für die Untersuchung eingegangen und die verschiedenen Definitionsmöglichkeiten von Projekten, als auch Anforderungsänderungen betrachtet. Es werden die Methoden für die Bewertung des Projekterfolgs diskutiert, bevor die am häufigsten verwendeten traditionellen und agilen Managementmodelle vorgestellt werden.

Im Anschluss erfolgt in Kapitel 3 die Beschreibung und Auswertung des Ergebnisses für die empirische Erhebung. Es wird auf die Vorgehensweise für die Datenerhebung, sowie die Operationalisierung der einzelnen Messwerte eingegangen. Die verwendeten Analysemethoden werden detailliert beschrieben und die erhaltenen Ergebnisse auf ihre Aussagekraft untersucht.

Abschließend werden in Kapitel 4 die theoretischen Modelle und empirischen Auswertungsergebnisse kurz zusammengefasst, bevor ein Ausblick auf weitere mögliche Forschungsarbeiten auf diesem Gebiet gegeben wird.

2. Theoretische Grundlagen der Untersuchung

In diesem Kapitel werden die für die Zielsetzung nötigen Begriffe genauer definiert und die theoretischen Modelle des Projektmanagements ausführlich diskutiert.

Es wird zuerst auf die grundlegenden Aspekte von Projekten im Allgemeinen eingegangen, bevor die Vor- und Nachteile der verschiedenen Modelle für die Bestimmung des Projekterfolges betrachtet werden. Anschließend werden die am weitesten in der Wirtschaft verbreiteten Modelle der Projektabwicklung erklärt und auf die Unterschiede agiler zu traditioneller Entwicklung eingegangen.

2.1. Projektdefinition

Im folgenden Abschnitt werden die möglichen Kategorisierungen eines Projektes erläutert und die verschiedenen Definitionen von Erfolg genauer betrachtet. Des Weiteren werden ausgewählte praktische Methoden als auch theoretische Ansätze für eine Messung von Erfolg und Misserfolg einer Projektdurchführung vorgestellt und auf die jeweiligen Anwendungsgebiete eingegangen.

2.1.1. Abgrenzung und Begriffserklärung

In diesem Kapitel werden die verwendeten Begriffe für das Verständnis eines Projektes definiert, sowie wesentliche Aspekte des Umfeldes erörtert.

Bevor es möglich ist, den Erfolg oder Misserfolg eines Projektes zu messen, muss zuerst eine Begriffsabgrenzung durchgeführt werden, um festzulegen welche Tätigkeiten unter dem Begriff des Projektes zusammengefasst werden können. Abhängig von der angewandten Definition kann eine Vielzahl an Vorgängen und Abläufen mit zugehörigen Erwartungen beschrieben werden.

Eine simple Erklärung für den Begriff des Projekts legt fest, dass sich ein Projekt durch einen Anfangs- und Endzeitpunkt auszeichnet, wobei eine in sich abgeschlossene Einheit an Tätigkeiten durchgeführt wird und deren Fortschritt messbar ist. Es gibt dabei nicht nur einen Auftragsgeber und Auftragsnehmer, sondern eine Vielzahl an weiteren Interessensgruppen, sogenannte Stakeholder, welche in Verbindung mit dem Projekt stehen (Versteegen et al., 2005, S. 1ff.).

Weitere Definitionen können die akzeptablen Tätigkeiten für ein Projekt zusätzlich eingrenzen. Das Deutsche Institut für Normen schränkt ein Projekt als „Einmaligen Prozess, der aus einem Satz von abgestimmten und gelenkten Tätigkeiten mit Anfangs- und Endterminen besteht und durchgeführt wird, um unter Berücksichtigung von Zwängen bezüglich Zeit, Kosten und Ressourcen ein Ziel zu erreichen, das spezifische Anforderungen erfüllt“ (DIN Deutsches Institut für Normung e.V., 2008, S. 972), ein. Aufgabenstellungen bei denen es sich somit um die Wiederholung einer Tätigkeit – unabhängig vom Zeitraum der letzten bis zur aktuellen Durchführung – handelt, werden bei Anwendung dieser detaillierten Begriffsabgrenzung nicht als Projekt verstanden. J. Rodney Turner entwickelte aufgrund dieser Einschränkung in Anlehnung an seine zuvor festgehaltene Abgrenzung eine weiterfassende Definition, die ein Projekt als eine temporäre Organisation, welcher zur

Durchführung einer bestimmten Arbeit Ressourcen zum Erreichen von Zielen bereitstellt, festlegt (Turner, 2009, S. 2).

2.1.2. Klassifizierung von Projekten

Projekte können anhand verschiedenster Gesichtspunkte in Kategorien eingeteilt werden. Die Bildung der Kategorien kann dabei beispielsweise mithilfe der Unternehmensbranche, der Komplexität des durchgeführten Projektes oder der strategischen Relevanz durchgeführt werden.

Derartige Klassifikationen helfen nicht nur einen schnellen Überblick über die Bedeutsamkeit eines bestimmten Projektes zu gewinnen, sondern ermöglichen es Risiken schneller zu identifizieren und die am besten geeigneten Ressourcen bereitzustellen. Abhängig von den gewählten Kriterien zur Bewertung kann zusätzlich die Vereinbarung von Projekt- und Unternehmenszielen erleichtert werden. (Hill, 2008, S. 170)

Müller und Turner (2007) extrahierten im Zuge ihrer Forschungsstudie zur Untersuchung des Einflusses von Projektmanagern auf den Projekterfolg fünf projektbezogene Kategorien zur Bewertung von Projekten, welche in Tabelle 2.1 dargestellt sind.

	Kategorie				
	Einsatzgebiet	Komplexität	Strategische Bedeutung	Vertragstyp	Projektphase
Wertung	Entwicklung und Konstruktion	Hoch	Verpflichtend	Festpreis	Durchführbarkeitsprüfung
	Informations- und Kommunikationstechnologie	Mittel	Neupositionierung	Nachverrechnung	Entwurf
					Ausführung
	Organisatorische Entwicklung	Niedrig	Verlängerung	Kooperation	Abschluss
					Inbetriebnahme

Tabelle 2.1: Kategorisierung von Projekten nach Müller & Turner (2007, S. 301)

Im nachfolgenden Teil der Arbeit werden drei in der Literatur bedeutende Einteilungen genauer diskutiert.

2.1.2.1. Einsatzgebiet / Zielsetzung

Die Kategorie der Zielsetzung steht in enger Verbindung mit der von Turner und Müller vorgeschlagenen Einteilung nach Einsatzgebieten. Körner (2008) schlägt eine Unterteilung in drei mögliche Gebiete vor: Geschäfts-, Infrastruktur- und Kundenauftragsprojekte.

Unter Geschäftsprojekten werden strategische Projekte verstanden, die zur Verbesserung der Produkte, Prozesse und Strukturen der Wertschöpfungskette der Firma beitragen. Es werden dabei Veränderungen in der Unternehmensstruktur durchgeführt, die zu einem besseren Betriebsergebnis führen sollen.

Infrastrukturprojekte beschäftigen sich mit Maßnahmen zum Erhalt, Ausbau und Veränderungen von betriebsnotwendigen Gebäuden und Kapazitäten.

Bei Kundenauftragsprojekten ist es Ziel, eine spezifische Leistung für einen Kunden zu erbringen, ohne Veränderungen in der Organisationsstruktur vorzunehmen. (Körner, 2008, S. 13ff.)

2.1.2.2. Komplexität

Komplexität wird im Zusammenhang mit Systemen als die Anzahl benötigter Informationen für die Beschreibung des Systems verstanden (Whitty & Maylor, 2007). Da es sich bei dieser Definition von Komplexität um eine sehr allgemeine Formulierung handelt, haben sich in der Praxis die drei Einstufungen „Hoch, Mittel, Niedrig“ entwickelt (Wysocki, 2007, S. 14).

Die Einstufung eines Projektes ist dabei sehr subjektiv und abhängig von den durchschnittlich durchgeführten Projekten, so kann ein Projekt von zwei Personen unterschiedlich komplex eingestuft werden. Wysocki schlägt daher eine objektivere Unterteilung der Komplexität für Projekte, abhängig von der Ausführlichkeit der Projektziele und des Weges zum Erreichen dieser, vor. Es werden vier Quadranten gebildet, welche in Abbildung 2.1 veranschaulicht werden.

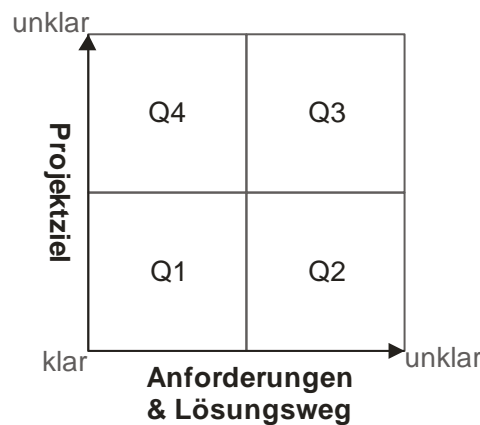


Abbildung 2.1: Projektlandschaft nach Wysocki (2007, S. 17)

Projekte bei denen sowohl das Ziel als auch der Lösungsweg eindeutig und klar definiert sind, fallen in den Quadranten 1. Es handelt sich hierbei um Projekte mit einer niedrigen Komplexität, da die Vorgehensweisen bereits bekannt sind und somit das Risiko eines Fehlschlages, durch ausführliche Spezifikationen, gering ist. Es besteht zwar die Gefahr, dass Probleme auftreten, jedoch sind die dazugehörigen Lösungswege bereits aus vergangenen Durchführungen bekannt und negative Einflüsse können somit minimiert werden. Das Risiko eines Projektes steht in positiver Korrelation mit der Komplexität, d.h. je komplexer ein Projekt, desto größer das Risiko.

Quadrant 2 besteht aus Projekten, bei denen das Ziel eindeutig, jedoch der Weg zum Erreichen unklar ist. Dieser Typus bildet den Großteil der in der Wirtschaft durchgeführten Projekte und erfordert eine andere Art der Herangehensweise als Projekte in Quadrant 1. Die geeigneten Lösungen sind nicht eindeutig und neue Wege und Mittel zum Erreichen des gesetzten Zieles müssen gesucht werden.

Projekte bei denen weder das Ziel noch der Lösungsweg eindeutig sind, fallen in den Quadrant 3. Hierbei handelt es sich zum Großteil um Forschungs- und Entwicklungsprojekte,

bei denen das Hauptaugenmerk auf dem Gewinn neuer Informationen liegt. Ziel ist die Generierung von Wissen, welches zukünftig einen positiven Einfluss nehmen soll.

Quadrant 4 repräsentiert Projekte, bei denen das Projektziel unbekannt, der Weg zum Erreichen jedoch klar definiert ist. In der Theorie handelt es sich dabei um einen Widerspruch, da ein Lösungsweg das Vorhandensein eines gestellten Problems erfordert. Ohne ein grundlegend definiertes Projektziel kann zwar ein Abschluss erreicht werden, jedoch ist es nicht möglich einen klar definierten Weg zum Erreichen eines unbekanntes Zieles zu haben. In der Praxis wird dennoch häufig versucht Projekte anhand dieses Ansatzes durchzuführen, indem unabhängig vom geforderten Ergebnis einer strikten und somit klaren Vorgehensweise gefolgt wird. (Wysocki, 2007, S. 18ff.)

Die Klassifizierung von Projekten anhand der Komplexität ist für die Realisierung besonders wichtig, da nicht nur das Potential eines Fehlschlages schnell abgeschätzt werden kann, sondern auch die Auswahl des am besten geeigneten Managementstiles vereinfacht wird. Ist weder das Projektziel noch der Lösungsweg klar definiert, muss sich sowohl die Herangehensweise an das Projekt selbst, als auch die Auswahl der Mitarbeiter ändern, um das Optimum zu erreichen.

Abbildung 2.2 zeigt die von Wysocki vorgeschlagenen Modelle unter Berücksichtigung des Projektquadranten. Eine genaue Ausführung der Abläufe und Unterschiede der jeweiligen Managementstile erfolgt in Kapitel 2.4 dieser Arbeit.

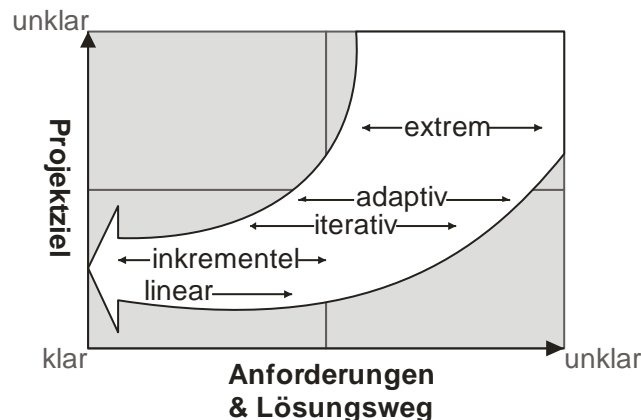


Abbildung 2.2: Managementmethoden-Modell von Wysocki (2007, S. 47)

2.1.2.3. Projektphase

Bei der Kategorisierung von Projekten nach Projektphase, werden diese anhand ihres Fortschrittes eingeteilt. Die einzelnen Projektphasen sind dabei stark vom gewählten Management Modell abhängig und variieren in ihrer Granularität.

Eine sehr simple Einteilung des Project Management Institute (2004, S. 23) definiert die drei aufeinander folgenden Phasen: Initial, Intermediate und Final.

Die erste Phase, Initial, beinhaltet die generelle Konzeption des Projektes, sowie die Bestimmung des Projektmanagementteams, welches die Entscheidung zur Durchführung des Projektes trifft. In der zweiten Stufe, der Intermediate Phase, sind alle zur Fertigstellung des Projektes notwendigen Schritte enthalten. Dies umfasst die genaue Planung, Durchführung

und Revisionen. Die letzte Phase, Final, ist durch die Akzeptanz und Übergabe des erstellten Projekterzeugnisses gekennzeichnet.

2.1.3. Stakeholder

Unter dem Begriff Stakeholder werden im Zusammenhang mit der Projektabwicklung die verschiedenen Personen und Interessensgruppen verstanden, welche ein direktes oder indirektes Interesse an dem Resultat eines Projektes haben (Gilb, 2005, S. 420). Es kann dabei ein gesetzliches oder moralisches Recht, ein Besitz, ein Interesse oder ein Beitrag in Form von Wissen oder Unterstützung bestehen. Ein allgemeines Interesse haben all jene Personen, welche direkt oder indirekt von einer Entscheidung, Handlung oder dem schlussendlichen Ergebnis betroffen sind. (Bourne, 2009, S. 31ff.)

Stakeholder sind ausschlaggebend für den Erfolg oder Fehlschlag eines Projektes, da sie die Anforderungen, Informationen und Prioritäten eines Projektes definieren und die Deckung ihrer Forderungen schlussendlich das Ziel der Projektabwicklung darstellt (Aguanno, 2005).

Vor allem die korrekte und frühzeitige Identifikation der kritischen Stakeholder ist für eine erfolgreiche Projektdurchführung wesentlich. Als kritische oder auch „Key Stakeholders“ (Cohen & Bradford, 2005, S. 236) für ein Projekt werden all jene Personen bezeichnet, die vom Ausgang des Projektes beeinflusst werden bzw. das Projektergebnis beeinflussen können (Gilb, 2005, S. 38).

Die zwei wichtigsten in Betracht zu ziehenden Stakeholder für ein Projekt werden von Young (2006, S. 72ff.) als der Projektsponsor, sowie der Auftraggeber identifiziert. Der Projektsponsor ist der unternehmensinterne Verantwortliche des Projektes. Zu seinen Aufgaben während der Durchführung zählen unter anderem die Autorität kritische Entscheidungen bei auftretenden Problemen zu treffen, als auch die Lenkung und den Fokus des Projektes durchgehend zu gewährleisten. Zusätzlich bildet er die Schnittstelle zwischen allen beteiligten Interessensgruppen, um eine optimale Abwicklung zu ermöglichen. Seine frühzeitige und anhaltende Integration in die Projektdurchführung ist essentiell, da er über die Kompetenz verfügt, benötigte Ressourcen für die Ausführung zuzuteilen. Der tatsächliche interne bzw. externe Auftraggeber des Projektes ist der zweite frühzeitig zu berücksichtigende Stakeholder.

Für Projekte, bei denen mehrere Auftraggeber involviert sind, ist es besonders wichtig diese korrekt zu identifizieren und die entsprechenden Kontaktpersonen zu definieren, um die Anforderungen, an das gewünschte Endergebnis möglichst präzise erfassen zu können. Da verschiedene Auftraggeber unterschiedliche Erwartungen haben können, ist es sinnvoll eine konkrete Person anstatt eines Komitees zu etablieren, welche die Vertretung aller Kunden übernimmt und die ausschlaggebenden Projektanforderungen konkretisiert.

Die Bedeutung weiterer Interessensgruppen ist abhängig vom durchgeführten Projekt und unterliegt einer starken Variation. James T. Brown (2007, S. 54f.) schlägt ein simples Modell zur Identifikation aller Stakeholder eines Projektes vor, indem er acht Richtlinien folgt, welche in Tabelle 2.2 aufgeschlüsselt werden.

Grundsatz	Fragestellung
Geld	Wer trägt die anfallenden Kosten? Wer erzielt Gewinn oder Verlust mit dem Projekt?
Ressourcen	Wer stellt Zeit, Räumlichkeiten oder Arbeitskraft zur Verfügung?
Zwischen-/Endprodukt	An wen werden anfallende Zwischenprodukte geliefert? Wer erhält das Endprodukt?
Unterschriften	Wer ist zeichnungsberechtigt für Änderungen? Wer übernimmt das Endprodukt?
Projektvergleich	Welche Interessensgruppen waren bei vergangenen Projekten involviert?
Organisationschart	Welche Teile der eigenen Organisation haben Interesse am Projekt?
Stakeholder fragen	Welche Interessenten werden von den bereits identifizierten Stakeholdern genannt?
Inoffizielle Einflussträger	Wer beeinflusst bereits identifizierte Stakeholder?

Tabelle 2.2: Stakeholder-Identifikation nach Brown (2007, S. 54f.)

Die Stakeholder spielen für ein positives Projektergebnis während der gesamten Durchführung eine wichtige Rolle. In der Literatur haben sich viele verschiedene Modelle für das Management der Stakeholder im Projektverlauf entwickelt, wobei sich alle Ansätze darauf einigen, dass eine durchgehende Rücksprache und Überwachung der Zufriedenstellung aller beteiligten Parteien bei Projekten notwendig ist.

Turner (2009, S. 77ff.) schlägt einen in Abbildung 2.3 dargestellten Prozess für den Umgang mit den Stakeholdern während des Projektes vor. Es werden dabei nach eingehender Identifizierung der Parteien, die jeweiligen Anforderungen festgestellt und eine Strategie zur Erfüllung dieser, in Übereinstimmung mit den Projektzielen entwickelt. Während der Durchführung wird die Akzeptanz der gewählten Lösungsansätze für den jeweiligen Stakeholder überwacht und gegebenenfalls Anpassungen vorgenommen, um das bestmögliche Ergebnis für alle beteiligten Parteien zu erzielen.

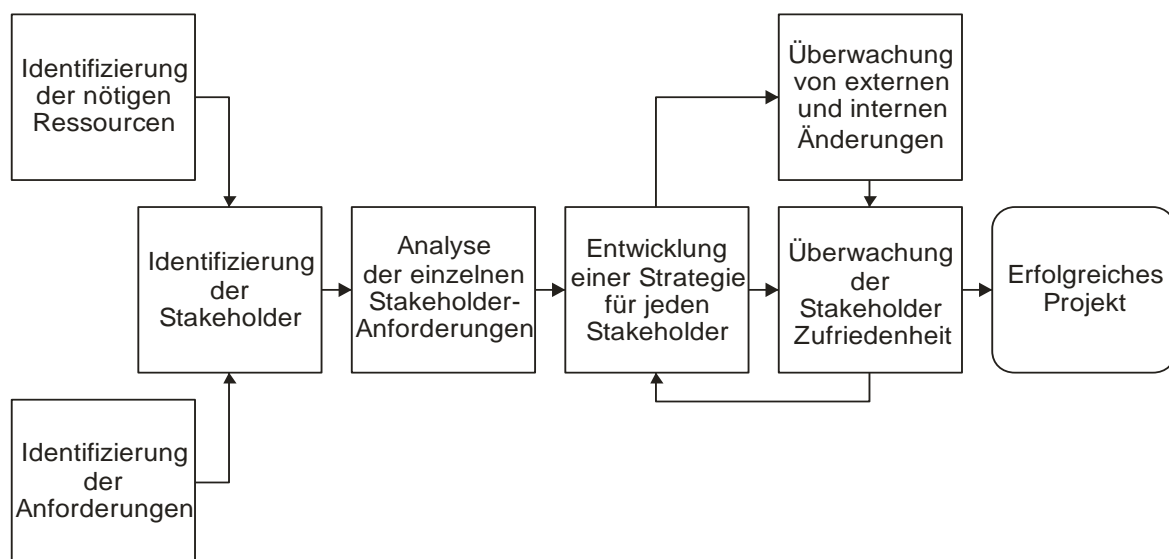


Abbildung 2.3: Stakeholder Managementprozess nach Turner (2009, S. 77)

2.2. Anforderungsänderungen

Im folgenden Abschnitt wird auf die möglichen Änderungen der Anforderungen während der Projektdurchführung eingegangen. Es werden sowohl die Möglichkeiten zur Klassifikation als auch die Auswirkungen von Änderungen auf den Projektverlauf betrachtet.

2.2.1. Einflussfaktoren auf die Anzahl der zu erwartenden Änderungen

Mithilfe einer Bewertung des Projektes anhand seines Komplexitätsgrades der Aufgabenstellung, welche in Kapitel 2.1.2.2 dieser Arbeit genauer erläutert ist, kann vorab zumindest eine grobe Abschätzung über die zu erwartenden Änderungen vorgenommen werden. Je unklarer das Projektziel und die benötigten Mittel sind, umso mehr Änderungen muss gerechnet werden und desto flexibler muss die Organisation auf Veränderungen reagieren können.

Die Größe eines Projektes stellt einen weiteren wichtigen Faktor zur Berücksichtigung der auftretenden Modifikationen dar. Jahren und Ashe (1990) stellten bei ihrer Untersuchung von Marineprojekten fest, dass sich die Anzahl der auftretenden Änderungen in Relation zur Größe des Projektes veränderte, d.h. je länger ein Projekt dauerte bzw. je mehr Personen involviert waren, desto häufiger kam es zu Änderungen. Um den negativen Auswirkungen entgegenzuwirken, verfügen diese Art von Projekten typischerweise über ein dediziertes Managementsystem, welches sich explizit mit der Verwaltung von Änderungen und damit auftretenden Problemen beschäftigt (Gunduz & Hanna, 2004).

2.2.2. Klassifizierung von Änderungen

Eine genaue Betrachtung von Anforderungsänderungen ist für die Evaluierung von Projekten unumgänglich, da nicht davon ausgegangen werden kann, dass die anfänglichen Anforderungen während der gesamten Projektlaufzeit ihre Gültigkeit behalten. Änderungen können dabei aufgrund geänderter Kundenwünsche, nötiger Anpassungen während der Entwicklung oder anderer formaler Forderungen nötig werden. (Lock, 2007, S. 403)

2.2.2.1. Bezahlte und unbezahlte Änderungen

Lock (2007, S. 403f.) schlug eine anfängliche Einteilung in Abhängigkeit vom Auslöser der Änderungen in „vom Auftraggeber“ und „vom Projektteam ohne Einbezug des Kunden“ verursachte Änderung vor. Da bei gewissen Modifikationen eine klare Abgrenzung jedoch nicht möglich ist, änderte Lock seine Kategorisierung mithilfe der kommerziellen Sichtweise auf die zu tragenden Kosten in „funded“ und „unfunded changes“ ab.

Als „funded changes“ bezeichnet Lock (2007, S. 404) dabei Änderungen, die aufgrund von geänderten Kundenwünschen zustande kommen. Diese Modifikationen stellen eine Änderung des Projektvertrages dar und anfallende Kosten müssen auf das vorhandene Projektbudget aufgerechnet werden. Abhängig vom Umfang der Änderungen kann es notwendig werden, eine Nachkalkulation aller weiteren Projektziele durchzuführen, um einen realistischen Überblick über die tatsächlichen Konsequenzen zu erhalten. So können gesetzte Deadlines, zugesprochene Ressourcen und weitere Faktoren für die Projektbewertung ebenfalls betroffen

sein. Vom Auftraggeber geforderte Modifikationen stellen Vertragsänderungen dar und sollten auch als diese innerhalb des Projektes vermerkt sein.

Treten während der Projektdurchführung nötige Änderungen auf, die in keinem direktem Zusammenhang mit dem Auftraggeber stehen, spricht Lock (2007, S. 404ff.) von „Unfunded changes“. Diese Art von Anpassungen wird vom Projektteam initialisiert und anfallende Kosten, als auch die Auswirkungen auf die weiteren Erfolgskriterien, sind dem Auftragnehmer zuzuschreiben.

2.2.2.2. Planänderungen und Zieländerungen

Dvir und Lechler (2004) unterscheiden bei ihrer Forschungsstudie zur Auswirkung von Anforderungsänderungen auf den Projekterfolg zwischen zwei verschiedenen Klassen an Änderungen, den sogenannten „goal changes“ – Änderungen des Projektzieles – und „plan changes“ – Änderungen des Projektablaufes. „Plan changes“ haben keine direkten Auswirkungen auf das Ergebnis des Projektes, sie verändern nicht die Zielsetzung, sondern nehmen Anpassungen an der Art und Weise des Erreichens der gesetzten Ziele vor. „Goal changes“ stellen eine direkte Modifikation des gesetzten Projektergebnisses dar und führen unwillkürlich zu Planänderungen.

Planänderungen werden typischerweise vom Projektumfeld verursacht und können beispielsweise aufgrund von einer geänderten Verfügbarkeit von Ressourcen, unerwarteten Umwelteinflüssen oder Ähnlichem auftreten. Es ist eine Anpassung des Projektplanes ohne oder mit möglichst geringer Änderung des Endergebnisses notwendig.

„Goal changes“ resultieren aus bewussten Entscheidungen der Stakeholder zur Änderung des Projektergebnisses. Diese Art der Änderung kann sowohl das direkte zu produzierende Endprodukt und dessen Spezifikationen als auch Projektkriterien, wie beispielsweise Deadlines oder das vorhandene Budget, betreffen.

2.2.3. Auswirkungen von Änderungen

Die Auswirkungen von Anforderungsänderungen auf die Projektabwicklung ist sowohl stark vom gewählten Managementmodell für die Durchführung als auch dem Zeitpunkt und Umfang der Modifikationen, sowie der Größe des Projektes abhängig.

Dynamische Managementmodelle planen Veränderungen über die Projektlaufzeit ein und bieten dadurch ein flexibles Umfeld um etwaige negative Konsequenzen zu minimieren. Eine ausführliche Diskussion der möglichen Methoden, welche diese Vorgehensweise unterstützen, erfolgt in Kapitel 2.4.

Statische und stark plangesteuerte Managementsysteme basieren auf der Annahme, dass die zu Projektbeginn definierten Anforderungen bis zum Projektabschluss unverändert bleiben, wodurch sich auftretende Änderungen besonders stark auf die Durchführung auswirken.

Dem Projektteam sind üblicherweise alle Arten von Änderungen unwillkommen, wobei späte Änderungen einen stärkeren negativen Effekt auf das Projektergebnis haben. Ein und dieselbe Modifikation kann dabei zu Beginn des Projektes lediglich eine neue Planung der Arbeitsschritte nach sich ziehen, vor Projektabschluss jedoch einen Fehlschlag der gesamten

Durchführung verursachen. Es ergibt sich daher, die in der Literatur wiederholt auftretende Regel, dass je später Änderungen bekannt werden, desto negativer sind die Auswirkungen. (Lock, 2007, S. 403)

Abbildung 2.4 bildet den exponentiellen Kostenanstieg der Anforderungsänderungen über den Projektverlauf nach Lock (2007, S. 404) ab.

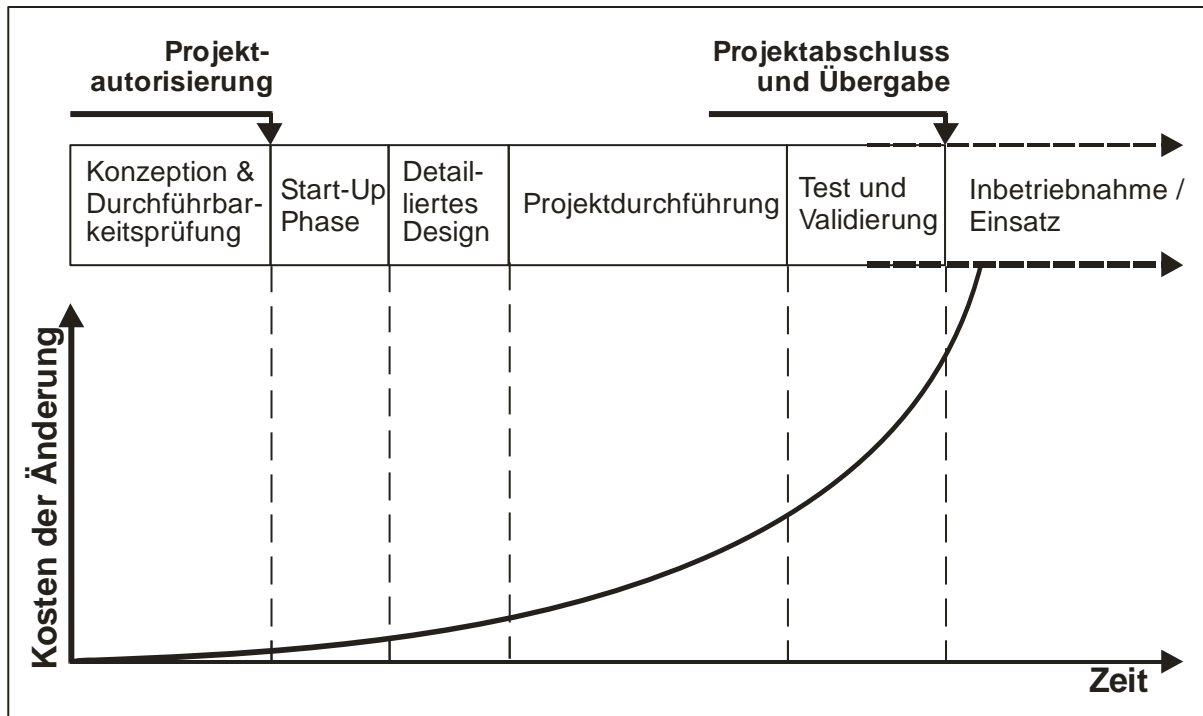


Abbildung 2.4: Änderungskosten Relation zur Projektphase in Anlehnung an Lock (2007, S. 404)

Aus diesen theoretischen Ansätzen der negativen Einflüsse von Anforderungsänderungen auf die Durchführung des Projektes, lässt sich folgende Hypothese für die Bewertung des Projekts ableiten:

Hypothese 1: Je größer die Anzahl der aufgetretenen Anforderungsänderungen, desto geringer ist der Projekterfolg.

2.3. Projekterfolg

Im diesem Kapitel werden die unterschiedlichen Erfolgsebenen genauer betrachtet, sowie ausgewählte Methoden zur Messung von Erfolg und Misserfolg eines Projektes diskutiert. Es werden dabei jene Ansätze genauer betrachtet, die sich für eine Messung des Resultats bei Kundenauftragsprojekten eignen.

2.3.1. Abgrenzung zum Managementenerfolg

Der Erfolg des Projektmanagements wird traditionellerweise mithilfe der drei Kriterien Zeit, Kosten und Qualität des von Doktor Martin Barnes entwickelten „Iron Triangle’s“ gemessen, welches in Abbildung 2.5 dargestellt ist.

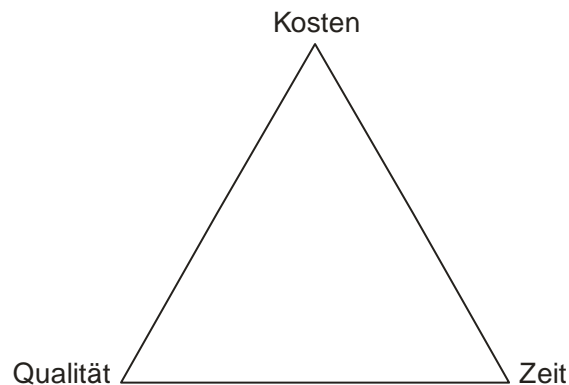


Abbildung 2.5: "Iron Triangle" nach Atkinson (1999)

Für ein erfolgreiches Projektmanagement muss ein Projekt innerhalb der definierten Zeit und dem angesetzten Budget fertiggestellt werden, sowie den festgelegten Qualitätsanforderungen entsprechen. In seiner Forschungsarbeit zum Thema „Measurement of project success“ stieß Anton de Wit (1988) jedoch auf unzählige Projekte, bei denen diese Kriterien nicht oder nur teilweise erfüllt waren, das Projekt selbst jedoch als erfolgreich erachtet wurde.

Durch genauere Betrachtung verschiedener Projekte leitete de Wit (1988) ab, dass ein insgesamt erfolgreiches Projekt nicht nur den technischen Anforderungen entsprechen, sondern ein hohes Niveau an Zufriedenheit mit dem gesamten Endresultat unter den involvierten Personen vorhanden sein muss, wobei diese sowohl Personen des Projektteams, der Unternehmung als auch Endnutzer und Auftraggeber beinhalten. Aufgrund dieser Feststellung ist eine Unterscheidung in Erfolg des Projektmanagements und Erfolg des Projektes selbst nötig, da ein gutes Projektmanagement zum Erfolg des Projektes beitragen, jedoch dessen Fehlschlag nicht verhindern kann. (de Wit, 1988)

In Unternehmungen wird diese Erkenntnis bereits in der Projektdurchführung berücksichtigt, wie Thomas und Fernández (2008) bei einer Untersuchung in 36 Firmen feststellten. So erkannten die betrachteten Firmen, dass es möglich ist, erfolgreiches Projektmanagement durchzuführen, ohne ein erfolgreiches Projekt zu erzielen.

2. Theoretische Grundlagen der Untersuchung - Projekterfolg

2.3.2. Messung

Zur Messung des Erfolges eines Projektes ist es nötig über Erfolgskriterien zu verfügen, anhand derer das Ergebnis evaluiert werden kann. Die Definition der Kriterien für Erfolg und Misserfolg ist in der Praxis jedoch äußerst komplex, da Projekte ihr Ziel während der Ausführung ändern können. So ist es möglich, dass ein anfänglich gestartetes Entwicklungsprojekt fehlschlägt, dabei jedoch neues Wissen für die Unternehmung generiert wird, wodurch es als erfolgreiches Forschungsprojekt gewertet werden kann. Diese Problematik führt dazu, dass Projekte oftmals ohne eine klare Definition, welches Ergebnis als Erfolg gewertet wird, gestartet werden. (de Wit, 1988)

Eine zusätzliche Hürde stellen die jeweiligen Anforderungen und Gesichtspunkte der verschiedenen Stakeholder dar. Auftraggeber, Zulieferer und Projektmitarbeiter haben andere Kriterien für den Erfolg. In Abhängigkeit von ihren Anforderungen kann dadurch ein und dasselbe Projekt sowohl Fehlschlag als auch Erfolg sein. Ein wirklich erfolgreiches Projekt muss alle Stakeholder zufriedenstellen. (Lock, 2007, S. 24f.)

Bei Berücksichtigung aller Stakeholder ist es daher nötig einen geeigneten Kompromiss der abzudeckenden Anforderungen zu finden. In der Realität muss infolgedessen mit einem gewissen Grad an Misserfolg, selbst für ein als „erfolgreich“ eingestuftem Projekt, gerechnet werden. Welcher Grad für ein Unternehmen dabei als akzeptabel gewertet wird, ist stark davon abhängig, welcher Strategie für die Unternehmensentwicklung gefolgt wird und welche Anforderungen priorisiert werden. (Graeme & Walter, 2008)

Avots argumentierte 1984 (de Wit, 1988), dass die Erfolgskriterien abhängig von der jeweiligen Projektphase sind. So kann zu Beginn des Projektes vor allem Wert auf eine termingerechte Fertigstellung gelegt, während der Durchführung vor allem die Kosten relevant und nach Projektabschluss Kosten und Zeit vernachlässigt werden, solange das Produkt die geforderten Spezifikationen einhält.

Für die Messung des Projekterfolges gibt es viele unterschiedliche Methoden, die nicht nur zwischen den verschiedenen Stakeholdern, sondern in Abhängigkeit der betrachteten Branche stark variieren. Da eine einheitliche Definition von Erfolg nicht vorhanden ist, kann auch keine objektive sondern immer nur subjektive Feststellung aufgrund des gewählten Modells der Messung stattfinden. De Wit (1988) ist sogar der Meinung, dass eine objektive Messung des Erfolges nicht möglich ist, sondern nur eine Illusion darstellt.

Im Folgenden werden zwei konkrete und in der Wirtschaft angewandte Methoden zur Messung des Projekterfolges für Kundenauftragsprojekte vorgestellt.

2. Theoretische Grundlagen der Untersuchung - Projekterfolg

2.3.2.1. „Balanced Scorecard“

Bei der von Robert Kaplan und David Norton entwickelten „Balanced Scorecard“ handelt es sich um ein strategisches Managementsystem zur gezielten Steuerung von Unternehmen anhand von vier Gesichtspunkten (Kaplan & Norton, 1992):

- Kundenperspektive
- Finanzwirtschaftliche Perspektive
- Innovations- und Lernperspektive
- Interne Unternehmensperspektive

Die vier verschiedenen Perspektiven werden durch Kennzahlen repräsentiert, welche anhand gesetzter Ziele definiert sind. Für das Ziel „Senkung der Lagerkosten“ auf der finanzwirtschaftlichen Perspektive, kann bspw. der Senkungsgrad der Kosten in Prozent als Kennzahl verwendet werden.

Abbildung 2.6 zeigt den kausalen Zusammenhang der vier Aspekte, der für die Steuerung eines Unternehmens anhand dieses Ansatzes nötig ist.

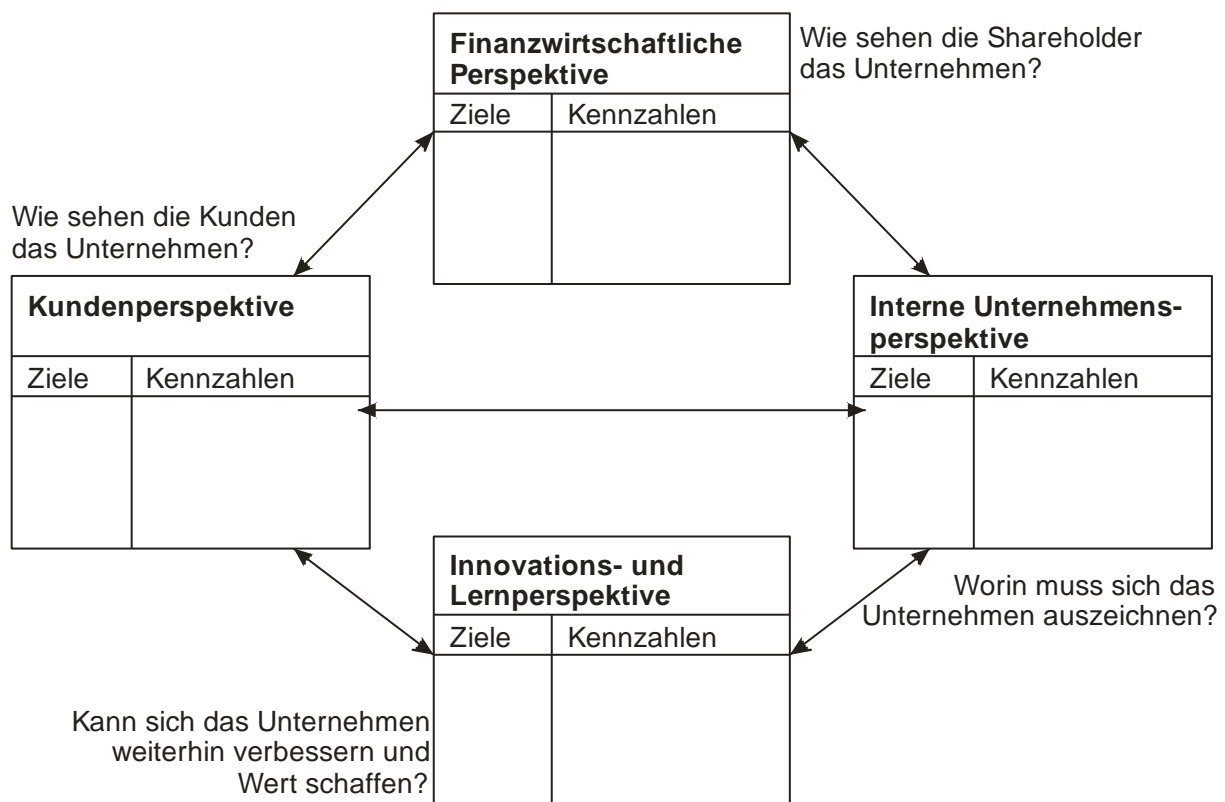


Abbildung 2.6: Kausale Zusammenhänge der „Balanced Scorecard“ nach Kaplan und Norton (1992)

Das Modell beruht darauf, dass sich die gewählten vier Ebenen gegenseitig beeinflussen lassen und Veränderungen einer Perspektive Auswirkungen in weiteren Bereichen der Unternehmung haben.

Bei den von Kaplan und Norton gewählten Perspektiven handelt es sich um die Standardperspektiven für die Steuerung eines Unternehmens. Abhängig auf welcher Ebene

2. Theoretische Grundlagen der Untersuchung - Projekterfolg

und für welche Zielsetzung innerhalb eines Unternehmens das Modell eingesetzt werden soll, können die Perspektiven frei gewählt werden. (Preißner, 2003, S. 22f.)

Zur Messung des Erfolges eines Projekts wird das „Balanced Scorecard“ Modell an die Ebene des Projektes angepasst und sowohl dem Projekt inhärente Anforderungen als auch alle anhängenden Ziele in die Bewertung aufgenommen. Während der Durchführung ist eine durchgängige Messung der Kennzahlen erforderlich. Um eine Steuerung zu ermöglichen, soll eine abschließende Bewertung durchgeführt und der Grad der Erreichung der gesetzten Ziele bei Projektabschluss bestimmt werden.

2.3.2.2. „Success Sliders“

Das „Success Slider“ Modell von Rob Thomsett berücksichtigt die schwankenden Anforderungen der verschiedenen Stakeholder für die Bewertung des Projekterfolges. Es wird dabei von der Annahme ausgegangen, dass die verschiedenen Projektziele und Anforderungen variable Prioritäten besitzen und aufgrund dessen auch das Erreichen dieser Ziele anhand ihrer Wichtigkeit für den Stakeholder gemessen werden sollten.

Thomsett (2002, S. 72ff.) identifiziert für sein Modell sieben Erfolgsfaktoren anhand deren der Erfolg gemessen werden soll:

- Zufriedenstellung der Stakeholder
- Erreichen der Projektziele und Anforderungen
- Einhaltung des Budgets / der Ressourcen
- Einhaltung der Deadlines
- Erschaffung von Zusatzwert
- Einhaltung der Qualitätsvorgaben
- Zufriedenstellung des Projektteams

Zur Bewertung der einzelnen Kriterien schlägt Thomsett eine fünfstellige Skala vor, welche von „OFF“ – für „nicht signifikant“ – bis zu „ON“ – für „absolut kritisch“ – unterteilt ist. Den einzelnen Erfolgsfaktoren werden, wie in Abbildung 2.7 anhand eines fiktiven Projektes dargestellt, durch ihre Wertung auf der Skala somit Prioritäten zugewiesen.

Die Bewertung der einzelnen Kriterien erfolgt dabei in Absprache mit den Stakeholdern, wobei durch die Festlegung des Erfolgsmaßes gleichzeitig kritische Merkmale des Projektes identifiziert werden. Kritische Merkmale werden durch die „Success Slider“ jene Faktoren, deren Wichtigkeit der Erfüllung gleich oder über 50 Prozent liegen. In extremen Fällen kann der Projektponsor sogar entscheiden, nur jene Kriterien zu berücksichtigen, die mit „ON“ bewertet wurden (Thomsett, 2002, S. 322ff.).

Thomsett weist darauf hin, dass bei den anfänglichen Bewertungen der Kriterien oftmals alle Slider durch die Stakeholder auf „ON“ gesetzt werden, da sie alle Punkte als erfolgskritisch beurteilen. Bei der praktischen Anwendung stieß Thomsett jedoch auf kein Projekt bei dem tatsächlich alle Faktoren hundertprozentig erfüllt werden mussten, damit das Projekt von den Stakeholdern als Erfolg gesehen wurde. (Thomsett, 2002, S. 72ff.)

2. Theoretische Grundlagen der Untersuchung - Projekterfolg

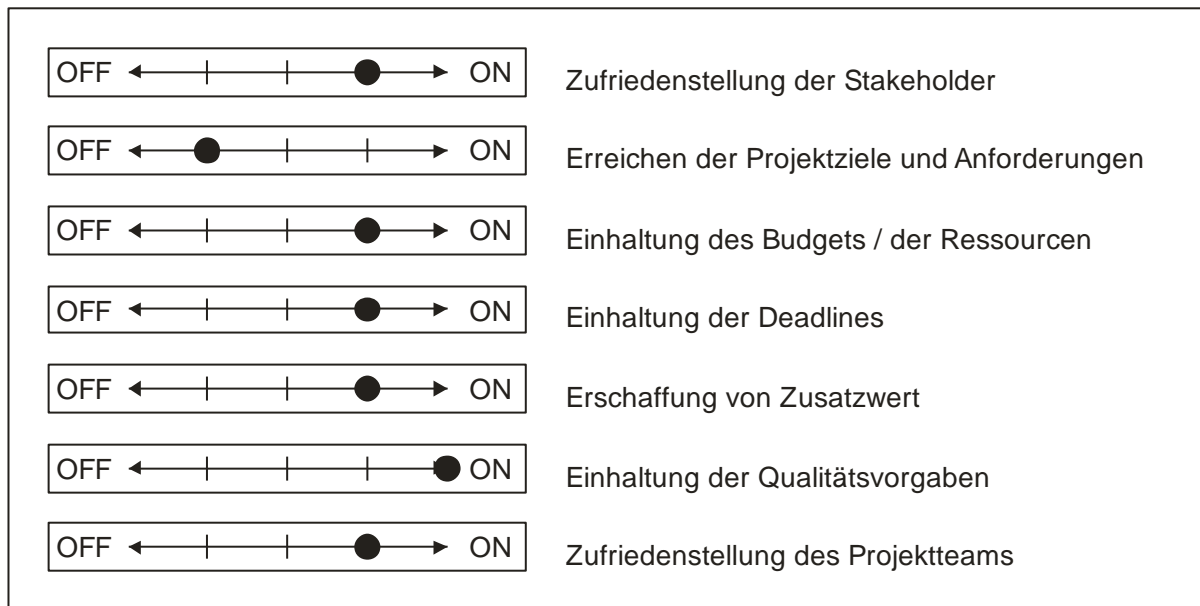


Abbildung 2.7: „Success Slider“ Bewertung nach Thomsett (2002, S. 74)

Während der Laufzeit des Projektes soll mithilfe der „Success Slider“ eine ständige Kontrolle der Erfolgskriterien vorgenommen werden, um dadurch eine zusätzliche Steuerungsmöglichkeit des Projektablaufes zu ermöglichen. Da die Wertigkeit der Anforderungen variabel ist, können sich die Kriterien zur Evaluierung des Projekterfolges während der Durchführung ändern und dies soll auch Ausdruck in dieser Methodik finden. (Thomsett, 2002, S. 322ff.)

Abbildung 2.8 zeigt eine beispielhafte Änderung der Wertigkeiten für die einzelnen Erfolgskriterien während der Projektdurchführung.

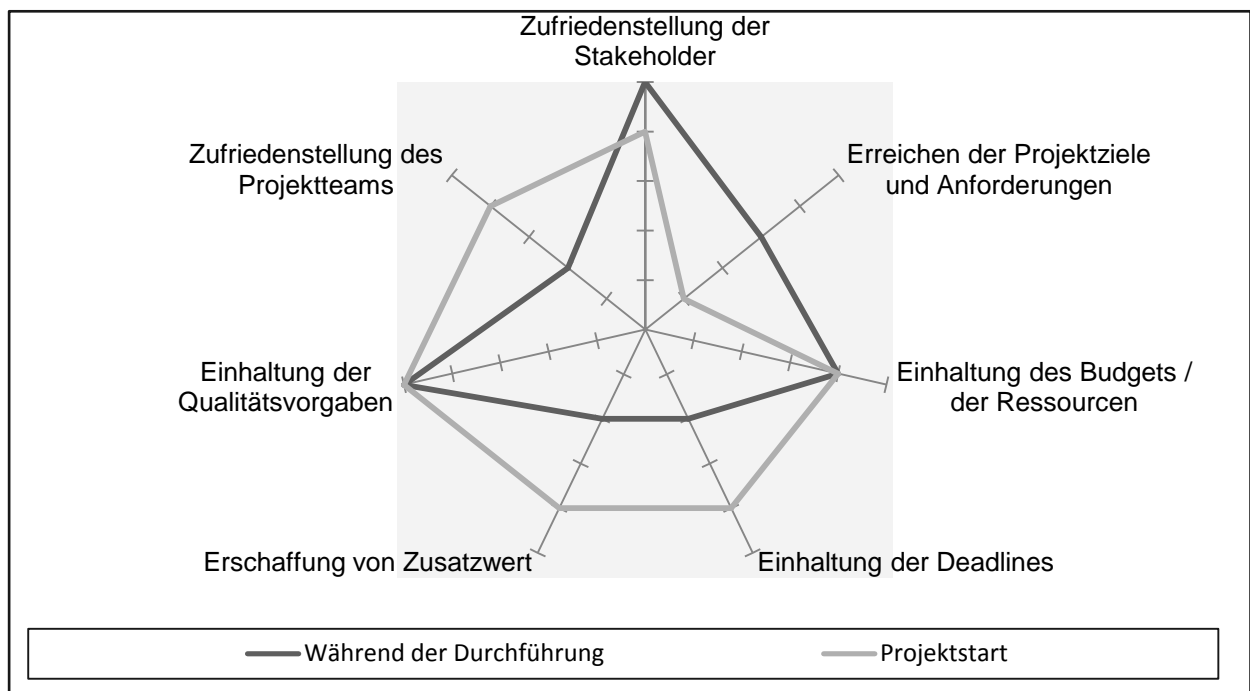


Abbildung 2.8: Wertigkeitsänderung der Erfolgskriterien in Anlehnung an Thomsett (2002, S. 322f.)

2.4. Agile Methoden

Der Begriff der Agilität lässt sich als die Fähigkeit oder Verhalten, welches erwartete und unerwartete Veränderungen zügig umsetzt und innerhalb der kürzesten möglichen Zeit unter Einsatz kostengünstiger, einfacher, sowie qualitativ hochwertiger Werkzeuge in einem dynamischen Umfeld mit laufend aktualisierten Wissen und Erfahrungen von seinem Umfeld zu lernen, beschreiben (Qumer & Henderson-Sellers, 2008).

Agile Methoden werden oft als ein spezifischer Managementprozess interpretiert; vielmehr handelt es sich dabei jedoch um eine veränderte Denkweise hinsichtlich des Entwicklungsprozesses (Smith & Sidky, 2009, S. 5f.). Es soll die Herangehensweise an Projekte geändert und bereits zu Beginn ein ganzheitlicher Blick gefördert werden, welcher für Veränderungen offen ist und flexibel auf neue Anforderungen reagieren kann. Basierend auf konkreten Werten und Prinzipien werden agile Methoden zum Einsatz in der Projektdurchführung ausgebildet.

Im Sinne der anfänglichen allgemeinen Definition von Agilität lassen sich somit agile Methoden, als eine Vorgehensweise mit folgenden immanenten Charakteristiken (Qumer & Henderson-Sellers, 2008):

- Menschen stehen im Vordergrund
- kommunikationsorientiert
- jederzeit flexibel und aufnahmebereit bezüglich erwarteter und unerwarteter Veränderungen
- schnell und sparsam, indem rasche Entwicklungszyklen gefördert werden um zügig Ergebnisse zu liefern, welche unter geringen Kosten die bestmögliche Qualität liefern
- lernfähig, durch die Suche nach Verbesserungsmöglichkeiten, während und nach der Entwicklung

In den folgenden Abschnitten wird genauer auf die Werte und Prinzipien agiler Methoden eingegangen, sowie spezifische, sequentielle und inkrementelle Managementmodelle vorgestellt und der Ursprung agiler Methoden untersucht.

2.4.1. Ursprung agiler Modelle

Der Einsatz agiler Vorgehensweisen zur Entwicklung stammt ursprünglich aus der Softwareentwicklung, wobei sich hier bereits mehrere konkrete Vorgehensweisen, basierend auf den agilen Prinzipien, etabliert haben.

Im Zuge der Prägung des traditionellen „Software Engineerings“ entwickelten sich ab den späten sechziger Jahren viele verschiedene Methoden (siehe Kapitel 2.4.2), die eine optimale Entwicklung garantieren sollten. Es wurden dabei über die Jahre immer detailliertere Ablaufpläne erzeugt, welche unabhängig vom gewünschten Endprodukt auf die Durchführung anwendbar sein sollten. Die so entstandenen Vorschriften legten dabei nicht nur die Abschnitte der Entwicklung sondern auch die darin enthaltenen Schritte genau fest. Aufgrund der Starrheit der Methoden empfanden viele Entwickler diese als starke Einschränkung für die Entstehung guter Software und siebzehn Softwareentwickler und Methodenforscher

erarbeiteten ein neues Konzept, welches die beteiligten Personen und ihre Interaktionen in den Vordergrund rücken sollen. (Hruschka et al., 2009, S. 1f.)

Im Zentrum dieser neuen Herangehensweisen stehen vor allem die beiden Werte Kommunikation und Einfachheit. Ziel ist dabei nicht die Quantität sondern Qualität des Informationsaustausches. Direkter und einfacher Austausch zwischen allen Projektbeteiligten soll gefördert werden und einen positiven Einfluss auf die Durchführung haben. Der Wert der Einfachheit lässt sich unter der Devise „so wenig wie möglich, so viel wie nötig“ (Hruschka et al., 2009, S. 5) zusammenfassen. In allen Bereichen der Projektdurchführung soll unnötiger Mehraufwand vermieden und dadurch der maximale Wert generiert werden. (Bleek & Wolf, 2008, S. 10f.)

2.4.2. Traditionelle Managementmethoden

Der folgende Abschnitt beschäftigt sich sowohl mit klassischen sequentiellen als auch iterativen Managementmodellen für Projekte. Sequentielle Methoden zeichnen sich dadurch aus, dass sie nicht nur in strikte Abschnitte unterteilt sind, sondern der Abschluss einer Projektphase grundsätzlich eine Rückkehr zu einer zuvor fertiggestellten Phase nicht mehr einplant. Es wird bei diesen Modellen davon ausgegangen, dass das Ergebnis einer Projektphase vollständig und korrekt ist und keine Veränderungen mehr nötig sind bzw. eintreffen werden. Iterative Modelle setzen eine zyklische Vorgehensweise voraus, bei der spezifische Schritte solange wiederholt werden bis das gesetzte Ziel für die Fertigstellung erreicht wurde.

2.4.2.1. Wasserfallmodell

Das Wasserfallmodell wurde in seinen Grundzügen 1970 von Winston Royce in seiner Seminararbeit zur Demonstration der Mängel in typischen Managementmodellen für Softwareprojekte entwickelt (Fairley, 2009, S. 55).

Der Name leitet sich von der wasserfallähnlichen Reihenfolge der einzelnen Phasen des Modells ab. Es muss immer erst eine Phase beendet werden bevor die nächste begonnen wird. Ist eine Phase einmal abgeschlossen, wird das resultierende Erzeugnis in die nächste Phase übergeben und darauf aufbauend weitergearbeitet. So müssen beispielsweise zuerst alle Anforderungen definiert worden sein, bevor mit dem Design begonnen wird. In der Designphase selbst wird dann jedoch explizit nur mit den zuvor erfassten Anforderungen gearbeitet.

Beim Abschluss einer jeden Phase wird eine Kontrolle der Ergebnisse durchgeführt und sowohl das Endprodukt als auch die nötigen Zwischenschritte zum Erreichen ausführlich dokumentiert. Da das Hauptprodukt der Phasen die Dokumentation darstellt, wird dieses Modell auch oft als „dokumentengetrieben“ bezeichnet (McConnell, 1996, S. 138).

Abbildung 2.9 stellt die am häufigsten verwendeten Phasen des Modells und ihre Reihenfolge dar.

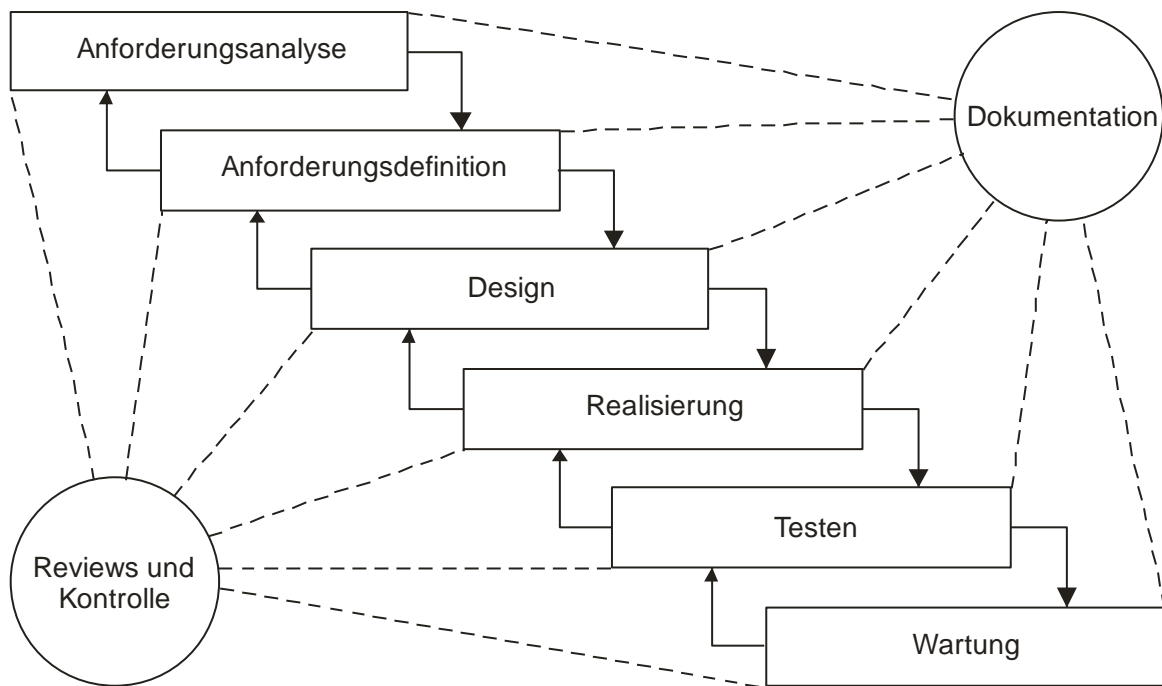


Abbildung 2.9: Wasserfallmodell basierend auf Rakitin (2001, S. 19) und Sommerville (2006, S. 66)

Der Austausch von Informationen findet nur zwischen zwei jeweils aufeinander folgenden Phasen statt. Dadurch kann es vorkommen, dass sich anfängliche Fehler über das gesamte Projekt hinweg durchziehen und erst sehr spät erkannt werden. Spät identifizierte Probleme können deshalb dazu führen, dass das gesamte Projekt neu gestartet werden muss. (Helms et al., 2005)

Projekte, die mithilfe dieses Modells durchgeführt werden, werden anhand von Terminen und der zu erzeugenden Dokumentation gelenkt, sodass zu erreichende Zwischenergebnisse aufgrund von gesetzten Fristen einen niedrigeren Rang einnehmen müssen. (Helms et al., 2005)

Das Modell eignet sich vor allem für den Einsatz in Projekten in denen das Endergebnis klar definiert ist, alle Anforderungen implizit bekannt sind und keine Technologieänderungen oder unbekannte Techniken zum Einsatz kommen sollen. Es minimiert den Planungsaufwand auf klar definierte Stufen und hilft hier Fehler in den frühen Stadien der Entwicklung zu finden. Sind die Anforderungen oder die zu verwendenden Technologien jedoch nicht eindeutig, ist diese Art der Vorgehensweise nicht zu empfehlen. (McConnell, 1996, S. 132ff.)

Tabelle 2.3 zeigt eine Zusammenfassung der Vor- und Nachteile des Modells nach Rakitin (2001, S. 20) und Fairley (2009, S. 57f.).

Vorteile	Nachteile
- Einfach zu verstehen und umzusetzen	- Nur sehr wenige Projekte können dem Modell tatsächlich folgen
- weit verbreitet und bekannt	- Reflektiert nicht den iterativen Charakter der Entwicklung
- Verpflichtet zu einer vollständigen Anforderungsaufnahme bevor mit dem Design begonnen wird	- Annahme, dass alle Anforderungen zu Projektbeginn feststehen ist unrealistisch
- Verpflichtet zu einem vollständigen Design bevor mit der Entwicklung begonnen werden darf	- Tatsächliches Produkt ist erst sehr spät im Projektablauf vorhanden, wodurch die Fehlerdiagnose erschwert wird
- Zwischentermine sind eindeutig bekannt	- Überprüfung des Zwischenstandes wird nur bei Abschluss einer Phase durchgeführt

Tabelle 2.3: Vor- und Nachteile des Wasserfallmodells

Aufgrund der frühen Entstehung des Modells haben sich basierend auf dessen Ansätzen nicht nur viele weitere Projektmanagementmethoden gebildet, sondern die hier vorgegebene Ablaufplanung hat sich auch auf die Strukturierung von Unternehmen umgeschlagen. Die räumliche Abgrenzung von Abteilungen anhand ihres Aufgabengebietes bei der Entwicklung innerhalb von Unternehmen kann auf das Wasserfallmodell zurückgeführt werden. (Helms et al., 2005)

2.4.2.2. V-Modell

Beim V-Modell handelt es sich um eine Erweiterung des in Kapitel 2.4.2.1 vorgestellten Wasserfallmodells. Der Name leitet sich aus der typischen Darstellung der Phasen in V-Form ab.

Die einzelnen Phasen werden gleich wie beim Wasserfallmodell sukzessiv durchlaufen, wobei für jede Planungsphase gleichzeitig die abschließenden Testkriterien für die Abnahme definiert werden. Nach Abschluss der Planungsphasen und Entwicklung wird innerhalb der Testphasen die Erfüllung der Anforderungen auf den verschiedenen Ebenen überprüft. Mithilfe von Unit Tests werden einzelne Funktionen des Codes auf Effizienz und Korrektheit getestet. Für die Integrationstests werden die einzelnen Teile des erzeugten Produkts zusammengefügt und in seiner Gesamtheit auf Fehler in der Zusammenarbeit untersucht. Die Systemtests überprüfen abschließend ob die anfangs definierten Anforderungen erfüllt sind und das Endprodukt alle gesetzten Ziele erreicht. (Charvat, 2003, S. 123)

Abbildung 2.10 zeigt den typischen Ablauf des V-Modells in Anlehnung an Ramachandran, Patel (Ramachandran & de Carvalho, 2009, S. 20) und Charvat (2003, S. 123).

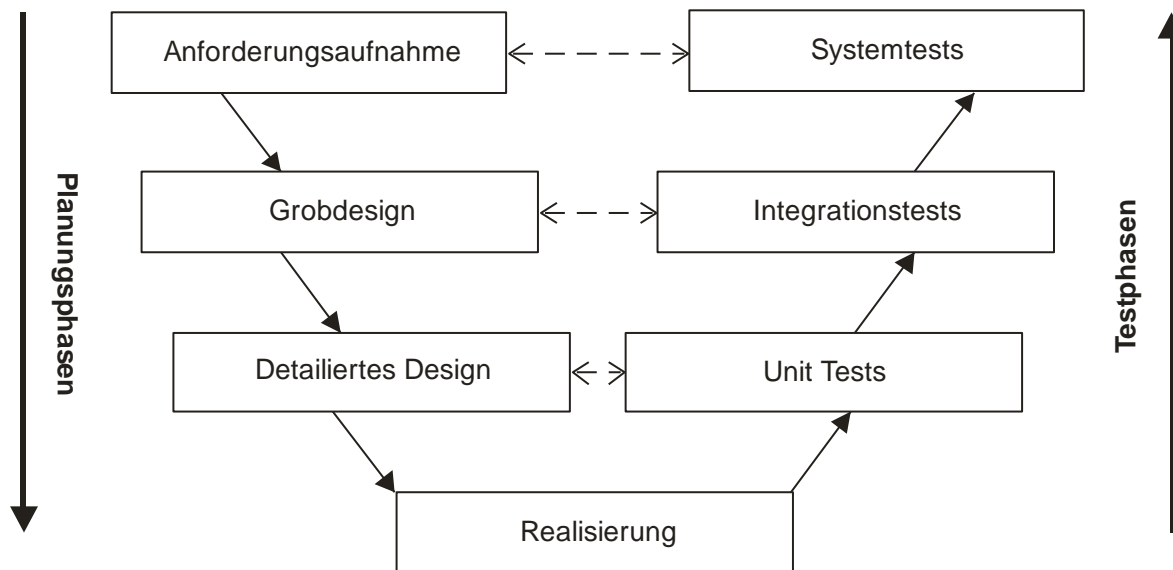


Abbildung 2.10: V-Modell basierend auf Ramachandran, Patel (Ramachandran & de Carvalho, 2009, S. 20) und Charvat (2003, S. 123)

Das V-Modell ist darauf ausgelegt, Fehler in der jeweiligen Konzeptphase aufzudecken und vor der Auslieferung abzufangen, sodass sich Probleme nicht erst bei Inbetriebnahme bemerkbar machen, wie dies im Wasserfallmodell der Fall sein kann.

2.4.2.3. Spiralmodell

Das Spiralmodell kombiniert die besten Eigenschaften des Wasserfallmodells mit der Erstellung von Zwischenprodukten. Ein großes Projekt wird dabei in Teilschritte zerlegt und über mehrere Iterationen die Funktionalität der Erzeugnisse bis zum endgültig geforderten Endprodukt erweitert. Jede Iteration besteht dabei aus vier Phasen der Anforderungsdefinition, Risikoanalyse, Entwicklung und Planung des nächsten Durchlaufs.

Eine Iteration beginnt jeweils mit der detaillierten Festlegung der Ziele und Anforderungen für den aktuellen Durchlauf. Es werden sowohl die entstehenden Risiken als auch mögliche Alternativen bewertet, bevor mit der Realisierung begonnen wird. Den Abschluss der Entwicklungsphase bilden Tests, die das erstellte Produkt auf Fehler überprüfen. Am Ende der Iteration findet die Planung des nächsten Durchlaufes statt, welche anhand von Reviews geprüft und zugestimmt werden muss. Erfüllt das bisher erstellte Produkt bereits alle gesetzten Anforderungen findet die Endabnahme statt. (McConnell, 1996, S. 141)

In Abbildung 2.11 wird ein beispielhafter Ablauf des Spiralmodells mit drei Iterationen dargestellt, wobei die Anzahl der Durchläufe in Abhängigkeit von der Komplexität des Projekts variiert.

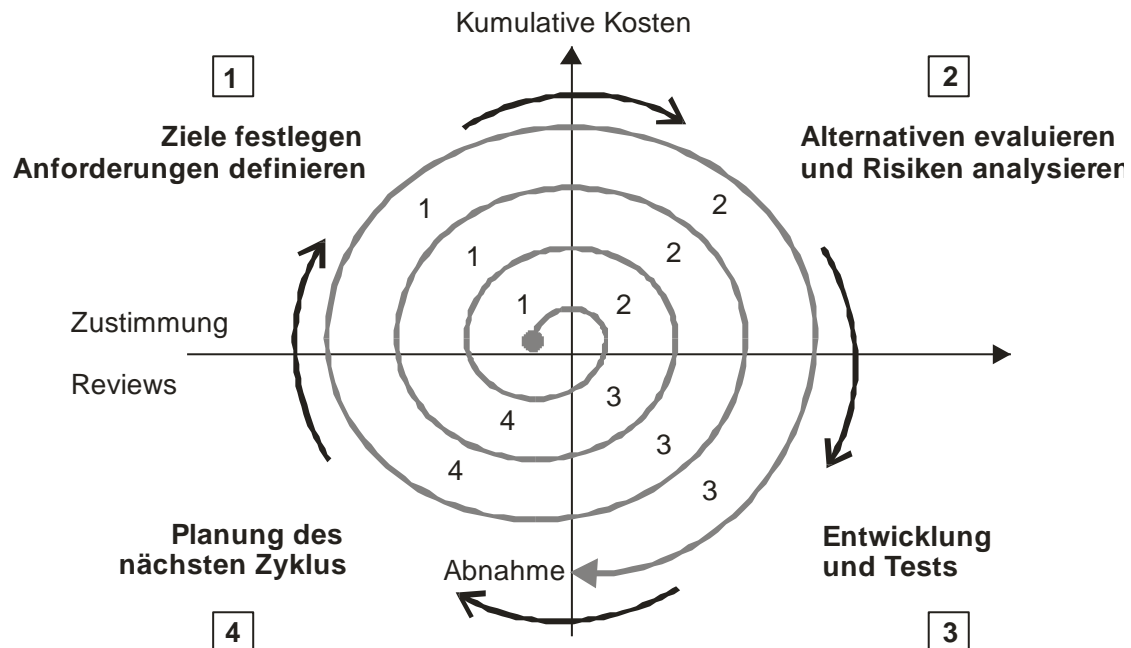


Abbildung 2.11: Spiralmodell basierend auf Helms et al. (2005)

Die Gesamtkosten eines Projekts unter Verwendung des V-Modells sind abhängig von der Anzahl der Iterationen, wobei frühe Durchläufe kostengünstiger sind als spätere. Zu Beginn nehmen die Kosten für die Entwicklung und Tests den Großteil der Gesamtsumme ein auch die Planung des nächsten Zyklus ist hier am kostengünstigsten. Je mehr Iterationen nötig sind, desto aufwändiger muss auch die Planung der jeweiligen Durchläufe werden und desto größer werden die dadurch anfallenden Aufwände. (McConnell, 1996, S. 141)

Das Modell eignet sich für komplexe Projekte, bei denen das Projektziel oder die zu verwendenden Mitteln zu Beginn nicht eindeutig definiert sind, da bei jedem Durchlauf nur ein Teil des zu erstellenden Produktes realisiert wird und somit nach Iterationsende eine genauere Planfestlegung stattfindet. Ein weiterer Vorteil des Modells besteht in der schnellen Sichtbarkeit von Ergebnissen sowohl für den Auftraggeber als auch die Projektleitung, da relativ rasch ein Zwischenprodukt für Feedback zur Verfügung steht.

Änderungen der Anforderungen während der Projektdurchführung werden zwar durch den repetitiven Charakter des Ablaufs abgefangen, führen jedoch weiterhin zu einer Kostensteigerung. Die Fähigkeit sich an Änderungen anzupassen ist davon abhängig wie flexibel das Modell ausgelegt wurde und kann stark variieren.

Die Nachteile des Modells liegen vor allem in seinem Anspruch an kompetente Überwachung und Planung aller Phasen einer Iteration. Für nichttechnisches Personal sind die einzelnen Abläufe des Modells oftmals nur schwer nachvollziehbar. (Rakitin, 2001, S. 26f.)

2.4.2.4. Zusammenfassung

Sequentielle Managementmethoden legen einen klaren Ablaufplan für die Projektdurchführung fest und können damit sowohl für klassische Fertigungsprojekte, als auch Entwicklungsprojekte, eingesetzt werden. Eine traditionelle Planung geht davon aus, dass die grundlegenden Anforderungen an das Projekt bereits zu Beginn definiert werden können und es während der Durchführung keine gravierenden Änderungen gibt. Der Einsatz strikt linearer Modelle ist daher nur dann zu empfehlen, wenn die Ziele und Wege zum Erreichen des Projektzieles eindeutig bei Projektbeginn feststehen.

Das Spiralmodell kann – als in der Industrie weit verbreitetes und anpassungsfähiges Modell – hier einen guten Kompromiss zwischen traditioneller Vorgehensweise und Flexibilität bilden. Wie tolerant das Modell dabei ausgeprägt ist, ist hier vom Projektmanagement abhängig und kann stark variieren.

In Tabelle 2.4 wird der Umgang mit spezifischen Anforderungen an das Projektmanagement, nach McConnell (1996, S. 156f.), verglichen.

Anforderung	Wasserfall- model	V-Modell	Spiralmodell
Ergebnis mit schlecht definierten Anforderungen	schlecht	eher schlecht bis gut	schlecht bis sehr gut
Ergebnis mit schlecht definierter Architektur	schlecht	eher schlecht bis gut	gut bis sehr gut
Kunde ist über Zwischenstand informiert	schlecht	eher schlecht	gut bis sehr gut
Auswirkung von Anforderungsänderungen während der Entwicklung	schlecht	eher schlecht	gut
Funktioniert mit wenig Managementenerfahrung	gut	gut	schlecht

Tabelle 2.4: Gegenüberstellung traditioneller Managementmodelle nach McConnell (1996, S. 156f.)

2.4.3. Philosophie agiler Methoden

Im folgenden Abschnitt wird ein Einblick in die Philosophie agiler Methoden gegeben, welche die Grundlage für die neue Generation an Managementmethoden bilden. Es werden die fundamentalen Werte vorgestellt und die Prinzipien beschrieben, aus denen diese abgeleitet wurden.

2.4.3.1. Das Agile Manifest

Die agilen Werte wurden 2001 im Manifest für agile Softwareentwicklung von einer Gruppe aus siebzehn Softwareentwicklern und Methodenforschern entwickelt, deren Ziel eine Verbesserung der zu stark plangesteuerten und einschränkenden Managementmethoden zu erreichen ist. Anhand ihrer Erfahrungen in der Entwicklung von Software veröffentlichten sie die in Abbildung 2.12 aufgelisteten folgenden vier grundlegenden Werte (Beck et al., 2001).

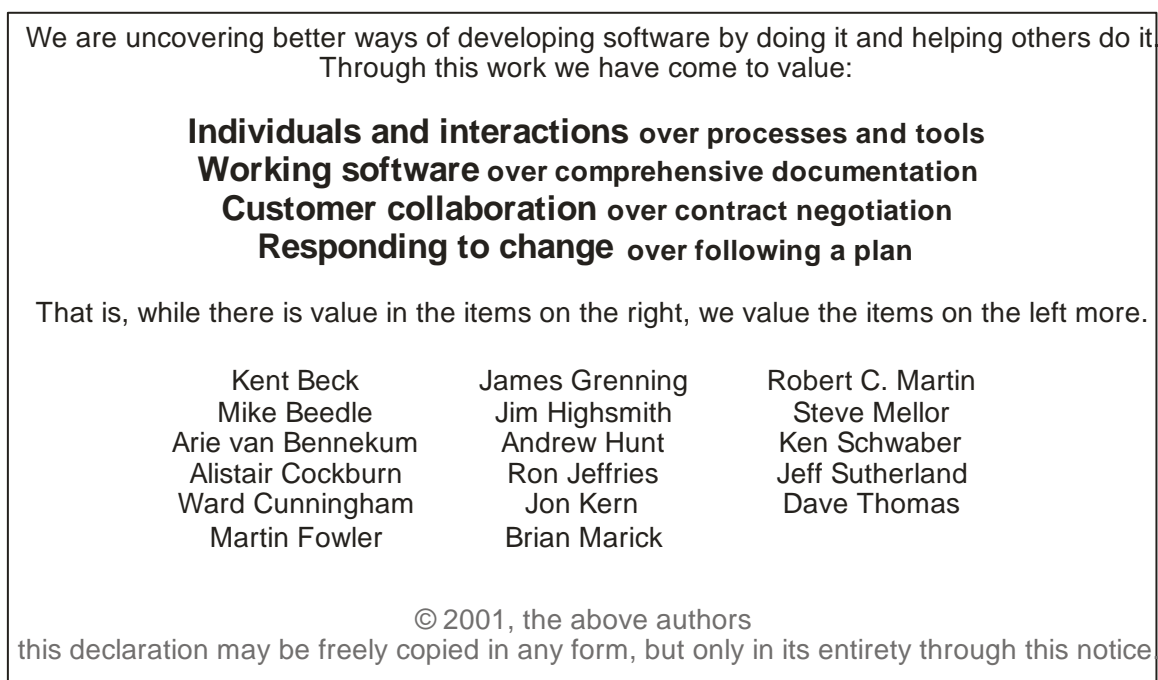


Abbildung 2.12: Das agile Manifest (Beck et al., 2001)

Das agile Manifest sagt aus, dass den Werten auf der linken Seite eine höhere Bedeutung als denen auf der rechten Seite zugewiesen werden soll. So soll beispielsweise den einzelnen Personen und ihren Interaktionen eine höhere Bedeutung als den Prozessen und Werkzeugen zukommen. Prozesse, Werkzeuge, Dokumentation, Vertragsverhandlungen und das Folgen von Plänen sollen nicht vollkommen vernachlässigt werden, jedoch anhand der neuen Philosophie nicht mehr die größte Bedeutung innerhalb eines Projektes oder einer Organisation erhalten.

Eine agile Philosophie fördert die Elemente der linken Seite und beinhaltet nur jene Teile der rechten Seite im Ausmaß, wie sie tatsächlich Wert schaffen und somit nötig sind. (Smith & Sidky, 2009, S. 6f.)

Individuen und Interaktionen vor Prozessen und Werkzeugen

Die Menschen, die ein Projekt ausmachen, sollen vor den Prozessen, auch den agilen Prozessen, und den verwendeten Werkzeugen stehen. Ihre Zusammenarbeit und ihre Fähigkeit zu kommunizieren schaffen die Voraussetzung für ein erfolgreiches Projekt. (Bleek & Wolf, 2008, S. 14f.)

Laufende Software vor ausführlicher Dokumentation

Bereits möglichst früh in der Laufzeit des Projektes soll laufende Software erzeugt werden, da daran der Projektfortschritt fortlaufend gemessen wird. Es ist wichtiger, sowohl für den Auftraggeber als auch das Management, dass ein möglichst großer Anteil der geforderten Software fertiggestellt wird, als dass alles vollständig und ausführlich in der Dokumentation enthalten, jedoch nicht umgesetzt ist. (Bleek & Wolf, 2008, S. 15)

Zusammenarbeit mit dem Kunden vor ausschweifenden Vertragsverhandlungen

Eine gute Zusammenarbeit und ehrliches Feedback über den Fortschritt und die Anforderungen vom Auftraggeber sind für ein erfolgreiches Projekt sehr viel wichtiger, als nur die in einem Vertrag festgehaltenen Punkte zu erfüllen. Vor allem bei Entwicklungsprojekten können im Vorhinein nicht mit hundertprozentiger Sicherheit alle nötigen Schritte und Ressourcen exakt abgeschätzt werden. Durch den noch immer weit verbreiteten Einsatz von Festpreisverträgen, bei denen ein fixer Preis für einen bestimmten Leistungsumfang definiert wird, kann es vorkommen, dass zwar alle vertraglichen Anforderungen erfüllt wurden, das Endprodukt jedoch nicht den tatsächlichen Erwartungen des Kunden entspricht (Bleek & Wolf, 2008, S. 14). Es ist daher wichtig einen regen Informationsaustausch mit dem Auftraggeber zu pflegen, um frühzeitig auf Fehler in den Anforderungen oder auf nicht berücksichtigten Erwartungen zu reagieren.

Auf Veränderungen reagieren vor strikter Ausführung nach Plan

Pläne sollen durch diesen Ansatz nicht komplett verworfen werden, sondern es soll einem Plan nur solange gefolgt werden als dieser auch tatsächlich mit den zu erreichenden Zielen korrespondiert. Treten während der Projektdurchführung Änderungen an den definierten Anforderungen auf, muss das Projektteam bereit sein, auf diese flexibel zu reagieren. Der agile Ansatz sagt sogar aus, dass mit Änderungen nicht nur gerechnet werden muss, sondern diese willkommen geheißen werden sollen und man davon ausgehen muss, häufig und regelmäßig neu zu planen (Smith & Sidky, 2009, S. 5).

2.4.3.2. Agile Prinzipien

Die nachfolgenden zwölf Prinzipien (Beck et al., 2001) agiler Vorgehensweise bilden die Grundlage, der in Kapitel 2.4.3.1 beschriebenen agilen Werte des agilen Manifests. Sie haben teilweise, zusätzlich zu dem agilen Grundgedanken, einen direkten Einzug in konkrete Managementmethoden genommen.

- Die Zufriedenstellung des Auftraggebers durch frühe und durchgängige Auslieferung nützlicher Software hat oberste Priorität.

- Änderungen, auch spät in der Entwicklung, werden willkommen geheißen. Um den Auftraggeber einen Wettbewerbsvorteil zu schaffen, sollen Anforderungsänderungen zu seinem Nutzen verarbeitet werden.
- Funktionsfähige Software soll wöchentlich oder monatlich produziert werden, wobei so kurz als mögliche Zeiträume gewählt werden sollen.
- Das Management muss mit dem Entwicklungsteam über die gesamte Projektlaufzeit auf täglicher Basis zusammenarbeiten.
- Im Zentrum der Projekte stehen motivierte Individuen. Es muss den Projektmitarbeitern die benötigte Arbeitsumgebung und Unterstützung bereitgestellt werden, ihre Arbeit zu leisten, als auch Vertrauen bestehen, dass sie ihre Aufgaben erfüllen.
- Die direkte, zwischenmenschliche Kommunikation bietet den effizientesten und effektivsten Weg Informationen auszutauschen.
- Funktionierende Software bildet das grundlegende Maß für den Projektfortschritt.
- Agile Methoden fördern eine nachhaltige Entwicklung. Die Projektsponsoren, Entwickler und Anwender sollen fähig sein, ein konstantes Arbeitstempo auf unbestimmte Zeit aufrechtzuerhalten.
- Ein durchgängiger Fokus auf technische Exzellenz und gutes Design verstärken die Agilität.
- Der Umfang an Arbeit, der nicht gemacht werden muss, soll durch das Einfachhalten maximiert werden.
- Projektteams, die sich selbst organisieren dürfen, kreieren die beste Architektur, Anforderungen und Designs.
- Es soll in regelmäßigen Abständen eine Reflexion des Projektteams stattfinden um Verbesserungsmöglichkeiten zu finden und die Vorgehensweise entsprechend zu optimieren.

Zwei weitere wichtige Grundsätze, die sich aus den Prinzipien ableiten lassen, jedoch nicht explizit in den vier Werten agiler Methoden festgehalten sind, stellen die Einfachhaltung des Projektes und durchgängige Reflexion dar. Unnötige Komplexität – ob in der Art des Informationsaustausches innerhalb des Projektteams oder in der Planung der Arbeit – soll soweit als möglich vermieden werden und fortlaufend und aktiv nach Verbesserungsmöglichkeiten gesucht werden.

2.4.4. Agile Managementmodelle

Im folgenden Abschnitt wird ein Überblick über die bekanntesten und in der Wirtschaft am häufigsten eingesetzten agilen Methoden gegeben. Es werden konkrete Managementmodelle aus der Softwareentwicklung im Detail vorgestellt und sowohl die Vor- und Nachteile der jeweiligen Praktiken genauer beschrieben.

2.4.4.1. Feature Driven Development

Bei Feature Driven Development (FDD) handelt es sich um eine iterative und inkrementelle Vorgangsweise in der Entwicklung. Hauptziel dieser Art der Entwicklung ist es innerhalb

kürzester Zeit greifbare Resultate für den Auftraggeber zu erzeugen und diese über die gesamte Dauer des Projektes beizubehalten.

Ein Projekt wird als Gesamtsumme der zu erstellenden Anforderungen gesehen und diese der Reihe nach einzeln entwickelt. Um eine solche Unterteilung vornehmen zu können, muss eingangs eine Planung der kompletten zu erstellenden Anforderungen vorgenommen werden (Koch, 2005, S. 7f.).

FDD basiert auf der Anwendung folgender acht Praktiken für die Projektdurchführung (Palmer & Felsing, 2002, S. 36ff.):

„Domain Object Modeling“ – Erstellung eines Gesamtmodelles

Den ersten Schritt für die Anwendung von FDD bildet die Erstellung eines umfassenden Gesamtmodelles für das zu entwickelnde System. Es müssen dafür noch keine detaillierten Designvorschläge festgelegt werden, sondern Ziel ist es die tatsächlichen Anforderungen aller Stakeholder durch eine Repräsentation des Endsystems zu erfassen und dadurch einen Plan für die Durchführung ableiten zu können. Das System wird in einzelne Objekte mit ähnlichen Aufgaben zerlegt, welche alle Anforderungen beinhalten müssen.

Das anfangs erstellte Modell soll nicht als starre Vorgabe für die Planung betrachtet werden, sondern wird nach jeder Iteration, wenn nötig, überarbeitet und anhand der aktuellen Version eine weitere Planung durchgeführt. Es können dadurch Anforderungen die zu Beginn noch unklar waren berichtigt werden und veränderte Kundenwünsche im System erfasst werden. (Koch, 2005, S. 249)

„Developing by Feature“ – Entwicklung anhand von Funktionen

Wie eingangs bereits erwähnt, findet die gesamte Entwicklung in kleinen Teilschritten statt, in denen jeweils eine spezifische Funktion des Gesamtsystems umgesetzt wird. Funktionen sollen dabei so gering als möglich gehalten werden und tatsächlich nur eine kleine Aufgabe erfüllen. In der Regel soll eine einzige Funktion maximal zwei Wochen für die Umsetzung in Anspruch nehmen dürfen, wobei der Großteil innerhalb einiger Stunden umsetzbar sein soll. Für komplexe Vorgänge hilft diese Regel das System zu vereinfachen, da eine Funktion, die das gesetzte Zeitlimit für die Entwicklung überschreiten würde, in ihre Einzelteile zerlegt wird und somit für ein simpleres Endprodukt sorgt. (Palmer & Felsing, 2002, S. 40f.)

Vor allem für Softwareprojekte ist die Unterscheidung zur Entwicklung anhand von Klassen wichtig. Unabhängig davon wie viele oder welche Stellen im Code betroffen sind, wird eine bestimmte Endfunktion der Software entwickelt; wenn nötig findet dabei eine Erweiterung bereits bestehender Klassen oder eine komplette Neuentwicklung statt.

„Individual Code Ownership“ – Verantwortung für individuellen Code

Innerhalb von FDD wird eine genaue Verantwortlichkeit für den erstellten Code zugewiesen. Wird bereits bestehender Code erweitert oder verändert, muss die aktuelle verantwortliche Person dafür sorgen, dass die anfänglichen Funktionen nicht negativ beeinträchtigt werden und die Gesamtintegrität erhalten bleibt. Um den Überblick über die Zuständigkeit bei komplexeren System behalten zu können, werden die Kompetenzen anhand des anfangs erstellen „Object models“ vergeben. Nachdem einem Entwickler ein bestimmtes Objekt

zugewiesen wurde, muss dieser bei der Entwicklung aller, sein Objekt betreffenden, Funktionen einen aktiven Beitrag leisten.

Da bei dieser Art der Aufteilung immer derselbe Entwickler für einen bestimmten Programmteil zuständig ist, wird Zeit für das oftmals langwierige Einlesen in fremden Code gespart. Die Nachteile liegen jedoch darin, dass es gegebenenfalls zu Wartezeiten kommen kann, falls die verantwortliche Person nicht verfügbar ist oder Änderungen anderer Objekte eine Anpassung fremder Codestellen erfordern. Beim Austritt eines Entwicklers aus der Firma kann zusätzlich hinzukommen, dass damit das Wissen über bestimmte Funktionen verloren geht. (Palmer & Felsing, 2002, S. 42ff.)

„Feature Teams“ – Funktionserstellung im Team

Die Erstellung von Funktionen erfolgt in Teams, welche aus dem Verantwortlichen für die Funktion und den Projektmitgliedern aller von dieser Funktion betroffenen Klassen besteht. Beim Funktionsverantwortlichen sollte es sich um einen erfahrenen Entwickler handeln, der alle nötigen Abhängigkeiten abschätzen kann und den Fortschritt überwacht. Wird während der Entwicklung festgestellt, dass weitere, bisher noch nicht aktive Objekte verändert werden müssen, so werden die zuständigen Personen zum aktiven Team hinzugefügt. Die Teamzusammensetzung ist somit dynamisch und kann sich auch nach dem Start einer Funktionsentwicklung noch ändern.

Für Projekte bei denen mehrere Funktionen parallel entwickelt werden, ist es sehr wahrscheinlich, dass ein Projektmitglied in verschiedenen Teams gleichzeitig aktiv ist. Ein Funktionsteam wird dann aufgelöst, wenn der zu erstellende Code verifiziert und in das Gesamtsystem integriert wurde. (Koch, 2005, S. 250f.)

„Inspections“ – Kontrollen

Das Ziel von Kontrollen besteht primär in der Auffindung von Fehlern, wobei als positive Nebeneffekte ebenfalls eine Wissensweitergabe stattfinden kann und zusätzlich die Konformität von Code durch eine ständige Kontrolle verbessert wird. Projektmitglieder sind sich durch laufende Überprüfungen immer darüber bewusst, welche Tests das von ihnen erstellte Produkt erfüllen muss, sodass sich Mängel nicht erst am Ende der Entwicklung bemerkbar machen.

„Regular Builds“ – Regelmäßige Zwischenprodukte

FDD legt keine expliziten Zeitpunkte für die Erstellung der Zwischenprodukte fest, sondern sagt nur, dass spätestens einmal pro Woche ein Ergebnis vorliegen muss. Dies hilft nicht nur den Entwicklern einen realistischen Überblick über die bereits erbrachte und noch zu leistende Arbeit zu erhalten, sondern auch den Auftraggeber anhand konkreter Ergebnisse über den Fortschritt zu informieren und Fehler in den definierten Anforderungen aufzudecken. (Koch, 2005, S. 251)

„Configuration Management“ – Redaktionsmanagement

Aufgrund des dynamischen Arbeitsablaufes, welcher die verteilte Entwicklung verschiedener, miteinander interagierender Funktionen fördert, ist ein globales „Content Managementsystem“, auch Redaktionsmanagement genannt, unumgänglich.

„Reporting of results“ – Berichterstattung über den Zwischenstand

Da viele Softwareprojekte unter dem „90% fertiggestellt Syndrom“ leiden, versucht FDD dem mit einem neuen Bewertungssystem für den Fortschritt entgegenzuwirken. Der Fortschritt einer einzelnen Funktion wird dabei durch sechs Milestones repräsentiert. Milestones stellen dabei Zwischenziele der Entwicklung dar und werden anhand eines konkreten Fertigstellungsdatums, sowie einer allgemeinen Zielsetzung definiert. Die ersten drei Milestones befassen sich hierbei mit dem Design, die darauffolgenden zwei mit dem tatsächlichen Programmieren sowie den dazugehörigen Tests und der letzte kann nach Integration des Codes in das Gesamtsystem abgeschlossen werden. Für jeden Milestone werden die fertiggestellten Prozente berechnet und der Durchschnitt ergibt den prozentuellen Fertigstellungsgrad einer Funktion. Der Durchschnitt aller Funktionen bildet den Zwischenstand für das gesamte zu erstellende System. (Palmer & Felsing, 2002, S. 53f.)

Zusammenfassung

Feature Driven Development stellt eine agile Herangehensweise für die Entwicklung dar. Es handelt sich dabei noch um kein vollständiges Managementmodell, sondern um eine Praktik, die in verschiedenen Modellen Einsatz finden kann.

Die Vorteile dieser Methode liegen im durchgängigen Überblick über den Fortschritt anhand konkreter Ergebnisse und der ständigen Verifikation. Die Gefahr besteht bei dieser Herangehensweise in dem Verlust von Wissen durch das Ausscheiden von Teammitgliedern, da kein allgemeingültiger Coding-Standard einzuhalten ist und dieser von den Projektteams selbst definiert werden kann bzw. sie darüber entscheiden, ob ein Standard einzuhalten ist.

2.4.4.2. Simultaneous Engineering

„Simultaneous Engineering“, in der deutschsprachigen Literatur oftmals mit SE bezeichnet bzw. im englischsprachigen Raum unter dem Namen „Concurrent Engineering“ und der Abkürzung CE bekannt, versucht eine Verbesserung der interdisziplinären Zusammenarbeit in Unternehmen durch das Parallelisieren von Prozessen herbeizuführen.

Die Anwendung von SE verfolgt im Wesentlichen drei Ziele (Ehrlenspiel et al., 2007, S. 44ff.):

Zeitgewinn in der Entwicklungs- und Produktionsphase

In linearen Managementsystemen nacheinander ausgeführte Arbeiten, werden bei SE soweit als möglich parallelisiert. Abbildung 2.1 verdeutlicht schemenhaft den erwarteten Zeitgewinn bei Einsatz von Simultaneous Engineering. Um korrekt entscheiden zu können, welche Arbeiten der verschiedenen Stufen vorgezogen werden sollen, muss ein durchgängiger Informationsaustausch zwischen allen beteiligten Personen stattfinden. Ein zentrales Team ist für die Kommunikation und Planung der verschiedenen Abläufe zuständig und überwacht den Fortschritt.

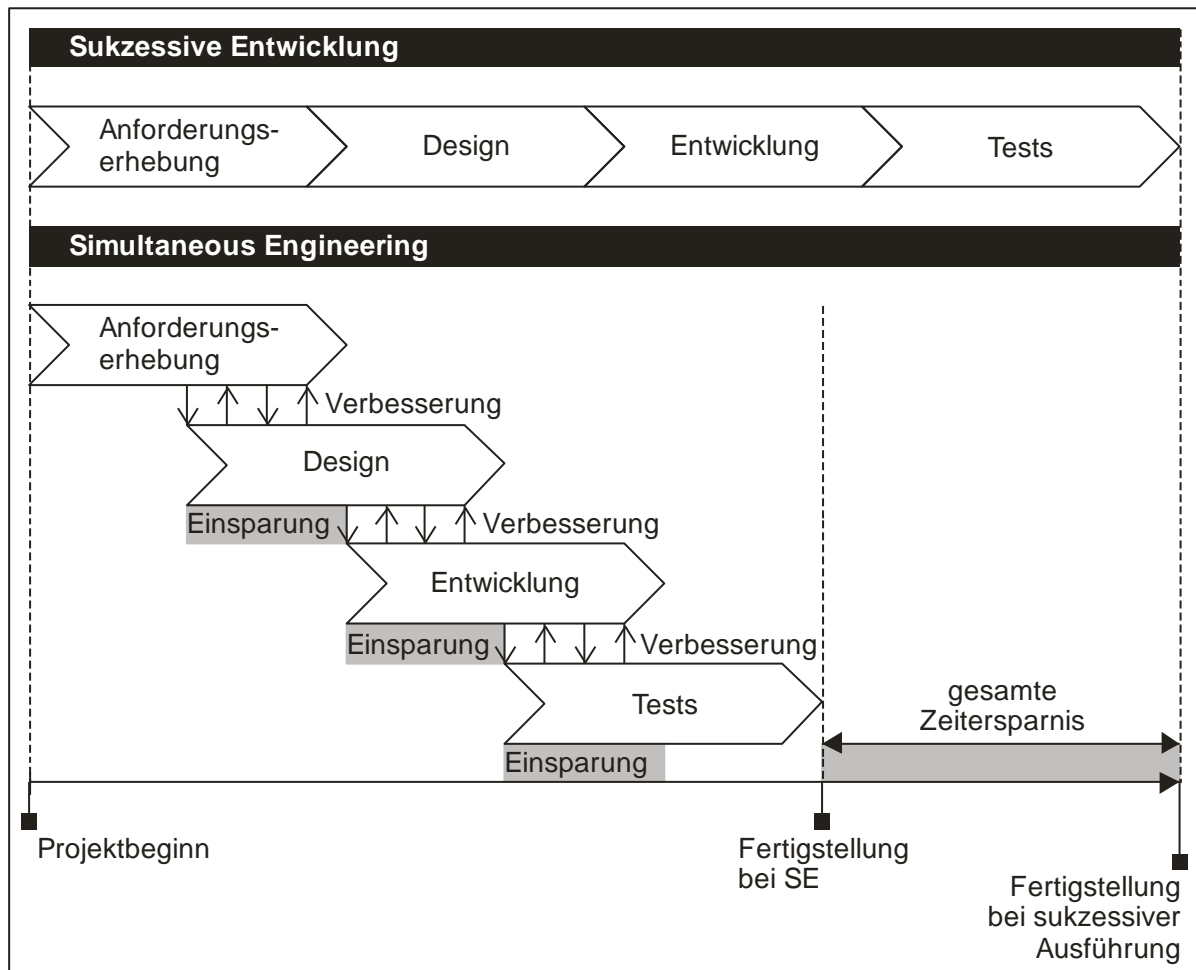


Abbildung 2.13: Zeiterparnis durch Simultaneous Engineering

Kostenminimierung

Durch die verkürzte Entwicklungszeit ist mit einer Minimierung der Gesamtkosten für die Entwicklung zu rechnen. Der genaue Betrag der Ersparnis lässt sich dabei jedoch nicht eindeutig aus der Zeiterparnis ableiten, da ein kurzfristig erhöhter Ressourcenaufwand für die schnellere Umsetzung wahrscheinlich ist.

Qualitätsverbesserung durch bessere Erfassung der Kundenanforderungen

Dem Auftraggeber kann durch die rasche praktische Umsetzung seiner Aufgabenstellung in relativ kurzer Zeit ein vorläufiges Ergebnis präsentiert werden. Anhand des Kundenfeedbacks können falsch erfasste Anforderungen korrigiert und so ein besseres Endprodukt erstellt werden. Voraussetzung ist eine kontinuierliche Zusammenarbeit mit dem Kunden und ehrliches Feedback zu den Zwischenergebnissen.

2.4.4.3. Lean Development

Seit einer Veröffentlichung des Erfolgskonzeptes von Toyota im Jahr 1990, entwickelte sich „Lean“ immer mehr zum Schlagwort in der Produktion. In einer fünfjährigen Studie stellte das Massachusetts Institute of Technology (MIT) fest, dass die Vorgehensweise des Autoherstellers um ein Vielfaches effektiver und effizienter war, als die traditionelle Herangehensweise der Konkurrenzunternehmen. Die von Toyota angewandten Konzepte

wurden unter dem Begriff „Lean“ zusammengefasst. (Lean Enterprise Institute, 2008, S. 105f.)

Lean stellt einen breiten Rahmen für die Entwicklung und Unternehmen dar, der durch agile Methoden verfeinert werden kann. Somit stellt Lean einen Oberbegriff und agile Methoden konkrete Anwendungen für die Umsetzung zur Verfügung. (Levine, 2009, S. xiii)

Lean, übersetzbar mit „schlank“ oder „schmal“, fokussiert die Minimierung von allem Unnötigen in den verschiedenen Bereichen. Insgesamt soll ein Prozess auf folgende Arten verbessert werden (Wilson L. , 2010, S. 9f.):

- weniger Materialien nötig
- geringere Investition nötig
- weniger Inventar nötig
- geringerer Platzverbrauch
- benötigt weniger Personal

Um die dafür nötigen Veränderungen zu erreichen, werden die in Abbildung 2.14 zusammengefassten Prinzipien (Wilson L. , 2010, S. 60ff.) verfolgt. Die Repräsentation als Haus dient dem einfacheren Verständnis der Umsetzung in einem Unternehmen – es muss zuerst eine gute Basis vorhanden sein, bevor die darüber liegenden Säulen erreicht werden können.

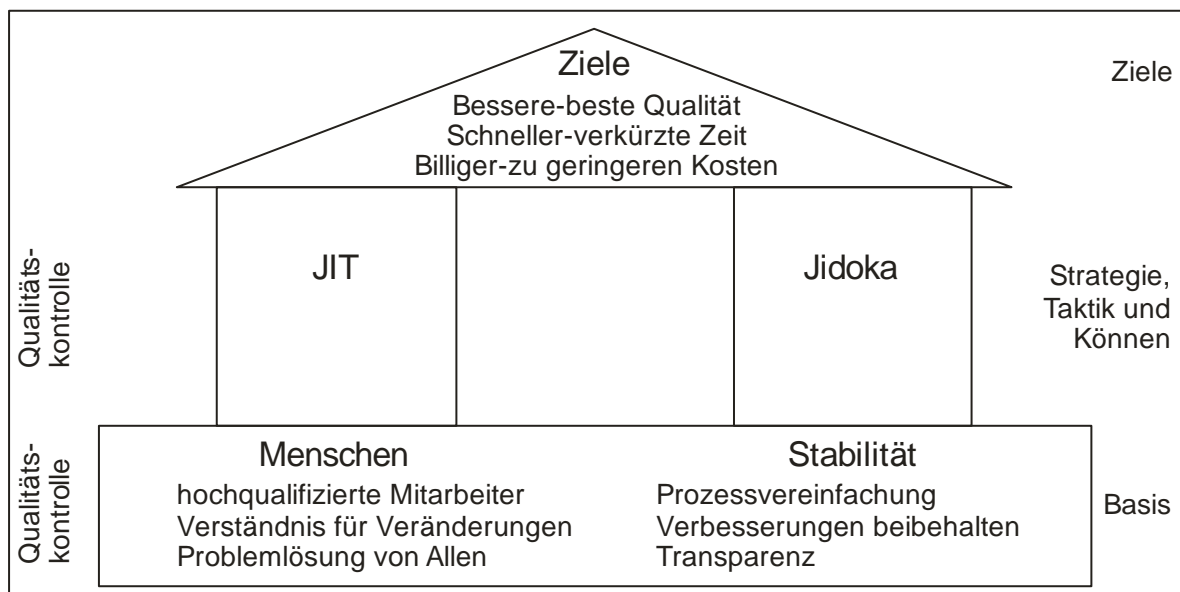


Abbildung 2.14: „House of Lean“ basierend auf Wilson (2010, S. 300)

Menschen

Für die Umsetzung von Lean müssen hochqualifizierte und in mehreren Bereichen ausgebildete Mitarbeiter vorhanden sein. Sie müssen nicht nur ein Verständnis für die benötigten Veränderungen mitbringen, sondern auch in kritischen Fällen selbst Entscheidungen für eine Verbesserung treffen können. Um das maximale Potential in jeder Abteilung zu erreichen, muss die Bereitschaft aller vorhanden sein, aktiv bei der Findung von Problemlösungen mitzuarbeiten.

Stabilität

Unter Stabilität wird die Verfügbarkeit von Prozessen und Leistungen zu den gewünschten Zeitpunkten verstanden. Die Vereinfachung von Prozessen, sowie eine transparentere Organisation der Abläufe, wobei hier wenn möglich eine tatsächliche Sichtbarkeit der einzelnen Abläufe für alle Beteiligten Personen erreicht werden soll, unterstützen dieses Ziel. Einmal erzielte Verbesserungen in allen Bereichen sollen zukünftig beibehalten und erweitert werden.

JIT

„JIT“ – Just-In-Time – beschreibt eine konkrete Methode für die Produktion, welche in verschiedenen Ausprägungen und durch unterschiedliche Mittel für die Zielerreichung eingesetzt werden kann. Bei einer Produktion, die nach diesem Prinzip arbeitet, wird ein zu erzeugendes Produkt in der exakt vereinbarten Qualität und zum genau vereinbarten Zeitpunkt geliefert. Vor allem in der Massenproduktion werden durch eine Produktion auf Nachfrage Verluste durch nicht verkäufliche Waren minimiert.

Jidoka

Bei Jidoka handelt es sich um eine von Toyota entwickelte, hundertprozentige Inspektion aller erzeugten Zwischen- und Endprodukte. Die durchgängige Kontrolle soll dabei verhindern, dass sich Fehler auf einer Ebene in der Entwicklung in das fertige Produkt durchziehen können. Werden beispielsweise Fehler im Design festgestellt, müssen diese zuerst behoben werden, bevor darauf aufbauende Tätigkeiten begonnen werden können.

Lean Software Development

„Lean Software Development“ leitet sich aus den Prinzipien des „Lean Manufacturing“ ab und überträgt die dort bereits bekannten Techniken auf die Entwicklung von Software. Ziel ist auch hier eine Minimierung unnötiger Prozesse, Tätigkeiten und Ressourcen zu erreichen und dadurch sowohl das Endprodukt zu verbessern als auch Kosten einzusparen.

Insgesamt haben sich sieben auf die Softwareentwicklung anwendbare Prinzipien entwickelt:

Eliminierung von Unnötigem

Die Eliminierung bezieht sich in der Softwareentwicklung auf unvollständig geleistete Arbeit, zusätzliche Prozesse, Zusatzfunktionen, gleichzeitiges Arbeiten an mehreren verschiedenen Aufgaben, Wartezeiten, unnötige Bewegungen – sowohl von Personen, als auch Dokumenten und Code – und Mängel. (Poppendieck & Poppendieck, 2003, S. 4ff.)

Späte Entscheidungsmöglichkeit

Entscheidungen, welche das Endprodukt unwiderruflich beeinflussen, sollen so spät als möglich in der Entwicklung getroffen werden. Vor allem durch unklare oder falsche anfänglich festgehaltene Anforderungen soll eine Änderung an der zu erstellenden Software solange als möglich, ohne negative Konsequenzen, ermöglicht werden. (Poppendieck & Poppendieck, 2003, S. 47ff.)

Eine konkrete Entwicklungsmethode, die dieses Konzept unterstützt stellt das in Kapitel 2.4.4.2 beschriebene Simultaneous Engineering dar. Die Entwicklung nach dem Prinzip des „Set-based-design“ ermöglicht durch einen anfänglichen Entwurf der möglichen Endvarianten

und die Konkretisierung des gewünschten Endproduktes durch Realisierung bestimmter Zwischenschritte ebenfalls einen möglichst langen Zeitraum für finale Entscheidungen.

Abbildung 2.15 stellt einen beispielhaften Ablauf des „Set-based designs“ nach Haberfellner und de Weck (2005) dar. Es werden zuerst die möglichen Endprodukte – Variants of Overall Concepts – und die dazugehörigen Realisationsschritte – Realisation-Steps – definiert. Bei der Umsetzung wird mit jenem Schritt begonnen, der sowohl die derzeit fixierten Anforderung als auch die größte Anzahl an Endvarianten ermöglicht. Durch Abschluss eines jeden Realisationsschrittes werden jene Endvarianten eliminiert, die sich folgerichtig nicht mehr verwirklichen lassen. Diese Herangehensweise hilft bei unklar definierten Anforderungen zumindest jene Tätigkeiten durchzuführen, welche definitiv für das gewünschte Produkt zu erledigen sind und minimiert so auch Verzögerungen in der Entwicklung.

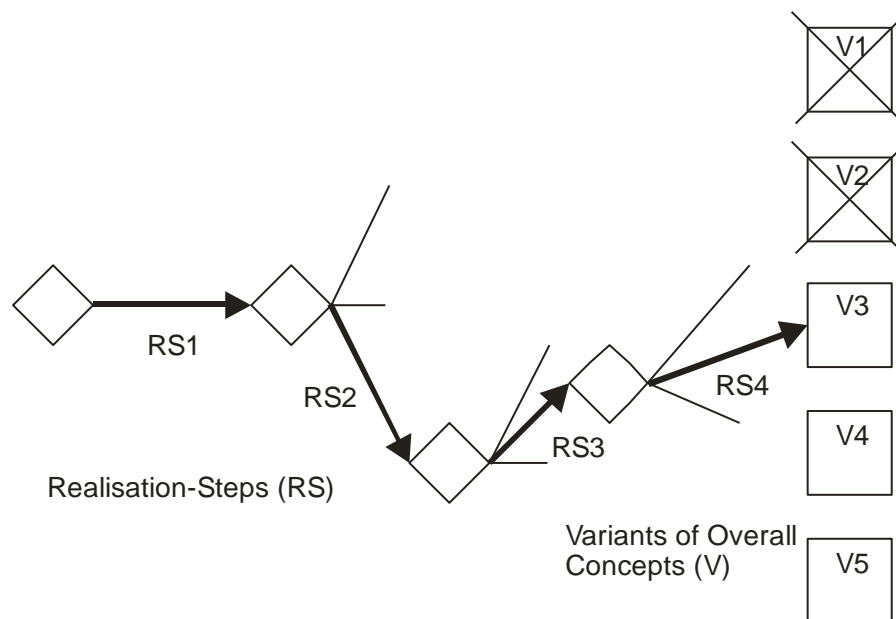


Abbildung 2.15: „Set-based design“ nach Shigley (Haberfellner & de Weck, 2005)

Kontinuierliches Lernen

Kontinuierliches Lernen bezieht sich sowohl auf die in der Entwicklung angewandte Vorgehensweise als auch die Anforderungen an das zu erstellende Endprodukt. Die Entwickler müssen dazu bereit sein in regelmäßigen Abständen ihren Fortschritt zu begutachten und Hindernisse, Probleme, aber auch positive Einflüsse festzustellen. Iterative Managementmodelle, wie das in Kapitel 2.4.2.3 beschriebene Spiralmodell, unterstützen dieses Prinzip durch kurze Entwicklungsphasen mit abschließenden Reviews der erbrachten Leistungen. (Poppendieck & Poppendieck, 2003, S. 15ff.)

Der „Set-based design“ Ansatz beruht ebenfalls auf der Idee, dass durchgehend neues Wissen gewonnen wird und dieses auch direkt für die Entwicklung verwendet werden soll. Es werden keine Lösungen fixiert, sondern nur Beschränkungen für den Weg zum Erreichen des Ziels, welche über die Projektlaufzeit mit konkreten Lösungen belegt werden.

Schnellstmögliche Lieferung

Dieses Prinzip leitet sich vor allem aus der steigenden Konkurrenz in allen Branchen ab. Auftraggeber erwarten eine konkrete Lösung für ihre Probleme in der kürzesten möglichen Zeit, daher soll auch das Ziel aller Entwicklern sein, dass gewünschte Ergebnis so schnell als möglich vorzulegen. Vor allem in der Softwareentwicklung ist die Erstellung von Zwischenprodukten zusätzlich für eine erleichterte Kommunikation zwischen technischen Personal und dem Auftraggeber essentiell, da Kunden oft wenig Ahnung von der Materie haben und erst durch konkrete Demonstration ihre Anforderungen genauer spezifizieren können. (Poppendieck & Poppendieck, 2003, S. 70)

Kurze Iterationen mit regelmäßigen Zwischenprodukten, realistische Zeitabschätzungen für die Umsetzung, sowie für alle Beteiligten erkenntliche Zeitpläne für die Entwicklung helfen dieses Ziel zu erreichen.

Bevollmächtigung des Teams

Um eine möglichst schlanke Entwicklung zu erreichen muss unnötige Bürokratie aus den Prozessen entfernt werden. Das Projektteam muss nicht nur die Verantwortung für den Erfolg der Entwicklung tragen, sondern auch die nötigen Entscheidungsfreiheiten besitzen, um ein gutes Produkt zu erzeugen. Motivierte und qualifizierte Mitarbeiter leisten die beste Arbeit, wenn man ihnen das nötige Vertrauen entgegenbringt, ihre Arbeit zu leisten ohne jeden Schritt zu überwachen. (Poppendieck & Poppendieck, 2003, S. 103ff.)

Integrierte Integrität

Hinter dem Prinzip der integrierten Integrität verbirgt sich das Ziel, ein vollständig zusammenhängendes und qualitativ hochwertiges System zu erstellen. Das Endprodukt soll nicht nur die derzeitigen Anforderungen erfüllen, sondern auch in Zukunft erweiterungsfähig sein, um den maximalen Nutzen für den Kunden zu schaffen.

Zur ganzheitlichen Integration von Integrität in das Endprodukt, wird einerseits die Wahrscheinlichkeit für Fehler durch die Wiederverwertung bereits vorhandener Technologien, Komponenten und Wissen, als auch verstärktes Testen aller Ebenen der Entwicklung eingesetzt. Nicht explizit geforderte Zusatzfunktionen, sowie unnötig verkomplizierte Konstrukte, werden aus dem System entfernt und das Endprodukt wird auf Verständlichkeit und Einsatzfähigkeit hin überprüft. Durch die Tests werden nicht nur die geforderten Funktionen des zu erstellenden Produktes, sondern auch dessen allgemeines Design kontrolliert. (Poppendieck & Poppendieck, 2003, S. 125)

Berücksichtigung des Ganzen

Für die Messung von Erfolg oder Misserfolg werden nicht nur die einzelnen Komponenten für sich, sondern das System als Ganzes betrachtet. In der Softwareentwicklung stellt die Anzahl der Fehler in der Software ein wichtiges Leistungsmaß dar, wobei nun die Möglichkeit bestehen würde, diese in Abhängigkeit vom jeweiligen Programmierer zu betrachten. Dieser Ansatz geht vom Gedanken aus, dass ein Fehler immer einer bestimmten Person zugeordnet werden kann. Es wird dabei die Idee, die Fehlerquelle im verwendeten System selbst zu suchen, vernachlässigt. Die Berücksichtigung des Ganzen fokussiert die zugrundeliegenden Probleme bei Fehlern und versucht diese durch die Zusammenarbeit aller Beteiligten zu eliminieren. Anstatt somit die Schuld für unerwartete Nebeneffekte neuen

Codes auf den jeweiligen Programmierer zu verlagern, sollen beispielsweise die definierten Anforderungen auf Unstimmigkeiten untersucht werden. (Poppendieck & Poppendieck, 2003, S. 153ff.)

2.4.4.4. Dynamic Systems Development Method

„Dynamic Systems Development Method“ (DSDM), auch unter dem Namen „Dynamic Systems Delivery Method“ (DSDM) bekannt, stellt ein Gerüst für iterative Entwicklung mit durchgängiger Kundenzusammenarbeit zur Verfügung. Unter Einhaltung des gesetzten Projektzeitraumes und des vorhandenen Budgets, ist es Ziel ein qualitativ akzeptables System zu erstellen, welches sich auch auf späte Anforderungsänderungen anpassen lässt.

Das Konzept basiert auf neun Prinzipien, von denen die ersten vier das Grundgerüst der Herangehensweise bilden und die restlichen fünf eine Struktur für die Durchführung vorgeben. Die neun Prinzipien dieser Vorgehensweise (The DSDM Consortium, 2003, S. 13ff.)

- Aktive Kundenbeteiligung ist unumgänglich
- DSDM Teams müssen dazu bevollmächtigt werden Entscheidungen zu treffen
- Der Fokus liegt auf einer regelmäßigen Auslieferung von Produkten
- Die Eignung des Produktes die Endanforderungen zu erfüllen, ist das essentielle Kriterium für die Akzeptanz von Zwischenprodukten
- Iterative und inkrementelle Entwicklung ist notwendig, um sich der gewünschten Lösung anzunähern
- Alle, während der Entwicklung umgesetzten, Änderungen sind reversibel
- Der gesamte Umfang und die allgemeinen Anforderungen werden vor Projektbeginn generell definiert
- Tests werden in den gesamten Projektzyklus integriert
- Kommunikation und Kooperation aller Stakeholder ist essentiell

Bei vielen Methoden ist die Einhaltung einer oder mehrere Prinzipien für das jeweilige Projekt oftmals nicht möglich und wird trotz dem Einsatz der ausgewählten Methode vernachlässigt. Im Gegensatz zu dieser gelockerten Haltung gegenüber dem Grundgerüst, besteht DSDM auf der Anwendung aller neun Prinzipien, um ein erfolgreiches Projekt liefern zu können. Das DSDM Consortium (2003, S. 13) geht in seiner Auslegung sogar soweit, dass von der Anwendung der Methode abgeraten wird, falls nicht allen Regeln gefolgt werden kann oder will bzw. zumindest Gegenmaßnahmen ergriffen werden sollen, um entstehende negative Konsequenzen abzuwenden.

Eine Grundvoraussetzung für den Einsatz von DSDM ist die Fähigkeit, das gewünschte Endprodukt in kleine funktionelle Teile für die Umsetzung zerlegen zu können. Das Prinzip der iterativen und inkrementellen Umsetzung und Auslieferung beruht darauf, dass die jeweils erzeugten Zwischenprodukte einen Nutzen für den Auftraggeber erbringen. Ist das zu erstellende Endprodukt nur in seiner vollständigen Form einsatzfähig, müssen zumindest die nötigen Arbeiten parallel durchführbar sein. (The DSDM Consortium, 2003, S. 25f.)

DSDM eignet sich vor allem für Projekte denen folgende Eigenschaften zugrunde liegen (The DSDM Consortium, 2003, S. 23ff.):

- Fertigstellung steht unter Zeitdruck
- Anforderungen sind nicht detailliert bzw. streng fixiert
- Aufgabenstellung des Projektes ist nicht überdurchschnittlich komplex
- Endanwender können eindeutig identifiziert werden

Da DSDM vor allem die Einhaltung eines gesetzten Projektzeitraumes und Budgets mit der darin höchstmöglichen Qualität fokussiert, eignet sich diese Vorgehensweise der Projektdurchführung bei der Entwicklung sicherheitskritischer Systeme nicht. Dies soll jedoch nicht heißen, dass eine Durchführung nach dem „quick and dirty“ Prinzip gefördert wird bei dem die Güte des Endresultats keine Rolle spielt solange eine zügige Umsetzung möglich ist, sondern ein akzeptables Qualitätsniveau definiert wird und es gilt dieses einzuhalten. Gewisse Fehler im System sind akzeptierbar, solange die Eckpfeiler der Anforderungen eingehalten werden. Es müssen dafür alle Anforderungen priorisiert werden und eine jeweils zu erreichende Qualität zugeordnet bekommen. (The DSDM Consortium, 2003, S. 65ff.)

Die Anwendung von DSDM setzt keine spezifischen Entwicklungstechniken voraus, solange die Einhaltung der neun Grundregeln nicht gefährdet ist. Das Vorgehensmodell lässt sich daher mit anderen spezifischen Entwicklungsmethoden – entweder teilweise oder vollständig – kombinieren und kann dadurch relativ simpel auf nicht softwarespezifische Projekte übertragen werden.

Ein ungewöhnlicher Ansatz dieser Methode liegt in der Bereitschaft die Qualität des Endproduktes für die Erreichung der Managementziele – Einhaltung von Zeit und Budget – anzupassen.

2.4.4.5. Extreme Programming

„Extreme Programming“, in der Literatur oftmals einfach mit XP abgekürzt, wurde als spezifische Methode noch vor der offiziellen Deklaration des „Agile Manifesto“ entwickelt. Kent Beck, ein Mitautor des agilen Manifests, erarbeitete das Grundkonzept des „Extreme Programming“ während seiner Arbeit an einem Gehaltsabrechnungssystem für „Chrysler Comprehensive Compensation System“ (C3). Ziel von C3 war es Verbesserungsmöglichkeiten in der Projektdurchführung zu finden und über mehrere Projektdurchläufe kristallisierten sich die später unter „Extreme Programming“ bekannt gewordenen Vorgehensweisen heraus. (Stephens & Rosenberg, 2003, S. 33)

Da XP das erste umfassende Modell für die Anwendung agiler Methoden in der Entwicklung darstellt, ist sie nicht nur die bekannteste, sondern auch die am meisten kritisierte Strategie. Es wird ein Überblick über die Grundzüge von XP gegeben und die am häufigsten kritisierten Punkte genauer betrachtet.

XP konzentriert sich auf den menschlichen Aspekt der Entwicklung, der häufig bei dem Einsatz von Managementmodellen vernachlässigt wird. Mithilfe der fünf Werte, auf denen XP aufgebaut ist, wird versucht zwischenmenschlichen Problemen entgegenzuwirken und dadurch eine bessere Entwicklung zu erreichen.

XP basiert auf den folgenden fünf Werten, welche die Entwicklung leiten sollen (Holcombe, 2008, S. 20ff.):

- Kommunikation

Zwischen allen Stakeholdern muss ein einfacher und durchgängiger Informationsaustausch möglich sein. XP unterstützt dies durch die Anwendung verschiedener Praktiken in der Entwicklung.

- Feedback

Feedback stellt eine zweite Dimension der Kommunikation dar und soll nicht nur zwischen dem Auftraggeber und dem Projektteam, sondern zwischen allen, am Projekt beteiligten, Personen gefördert werden.

- Einfachheit

Es soll immer die einfachste, mögliche Lösung gewählt werden, die die gesetzten Anforderungen erfüllt. Unnötige Prozesse, komplizierte Code-Konstrukte oder nicht geforderte und komplexe Funktionen für den Benutzer sollen aus der Entwicklung verbannt werden. Komplexität im System muss sich durch daraus entstehende Vorteile rechtfertigen lassen, so darf ein Prozess komplexer als unbedingt nötig sein, wenn das Gesamtsystem dadurch stabiler wird. Insgesamt lässt sich dieser Wert unter dem bekannten „You Ain’t Gonna Need It“ (YAGNI) Ansatz zusammenfassen – nur jene Teile die tatsächlich benötigt werden und einen Nutzen generieren, sollen umgesetzt werden.

- Mut

Mut wird hier synonym mit „Selbstvertrauen haben um riskante Tätigkeiten durchzuführen“ verwendet. Vor allem für die erstmalige Anwendung von XP ist es nötig, den Mut zu besitzen sich von traditionellen Managementsystemen und deren streng

geregelten Prozessen abzuwenden. Ein weiterer Aspekt, der bei agilen Methoden Courage erfordert, ist die Bereitschaft auf Anforderungsänderungen des Kunden während der Entwicklung positiv zu reagieren.

- Respekt

Jedem Projektbeteiligten muss der nötige Respekt entgegengebracht werden, seine Meinungen frei zu äußern und durch konstruktive Kritik eine Verbesserung des Gesamtsystems zu erreichen. Ein Großteil der Probleme eines Projektes hat ihren Ursprung in den zwischenmenschlichen Beziehungen der Beteiligten. Es muss daher ein geeignetes Arbeitsklima geschaffen werden, welches den Aufbau von Vertrauen und die Übernahme von Verantwortung unterstützt.

Die fünf Werte geben noch keine direkten Empfehlungen für den Einsatz bestimmter Vorgehensweisen in der Entwicklung ab, sondern legen ein Grundgerüst für die zwölf Praktiken von XP fest (Stephens & Rosenberg, 2003, S. 6ff.).

- Testgetriebene Entwicklung bzw. Permanentes Testen
- Planning-Game
- Teamzusammenarbeit bzw. Kundeneinbeziehung
- Iterationen
- Metaphern
- Einfaches Design
- Refactoring
- Kollektives Eigentum
- Pair-Programming
- Durchgehende Integration
- Keine Überstunden
- Coding-Standards

Um die Anforderungen an das zu erstellende System zu erfassen, werden diese mithilfe von „User-Stories“, in denen der Ablauf einer gewünschten Funktion in einfachen Worten und Metaphern beschrieben wird, festgehalten. Im sogenannten „Planning-Game“, wird somit das gesamte System beschrieben und der dazugehörige Aufwand für die Umsetzung abgeschätzt. Dieser Vorgang wird unter Einbezug aller Projektmitglieder, auch dem Kunden, bei jeder Iteration wiederholt und, wenn nötig, überarbeitet.

Die gesamte Entwicklung wird mithilfe von Tests gesteuert. Es werden immer zuerst die jeweiligen, zu bestehenden Tests entworfen und ausgeführt, bevor der dazugehörige Code geschrieben wird. Beim ersten Ausführen schlagen konsequenterweise alle Tests fehl, da noch kein Code entwickelt wurde, je mehr Code programmiert wurde, desto mehr Tests müssen bestanden werden. Werden alle Tests bestanden, gilt der Code als fertiggestellt. Die Tests ersetzen damit das Design und die genauen Spezifikationen, da die Funktionalität anhand der Testkriterien festgelegt wird.

Sobald die ersten „User-Stories“ fertig geschrieben wurden, wird bereits mit der ersten Iteration begonnen, um so dem Kunden in kurzen Abständen ein Zwischenprodukt präsentieren zu können. Es werden für den Abschluss einer Iteration alle fertiggestellten Teile zu einem lauffähigen System integriert. Der Zusammenbau des Codes zum Gesamtsystem, inklusive aller Tests, soll dabei nicht mehr als zehn Minuten dauern. Bei größeren Systemen darf diese Grenze nach oben ausgebessert werden. Anhand des Kundenfeedbacks wird der bereits erstellte Code überarbeitet und die nächste Iteration gestartet. Der Kunde sollte dabei ständig vor Ort bzw. durchgängig erreichbar sein. Falls dies nicht möglich ist, soll eine repräsentative Person den Kunden ersetzen. Wenn ein Ersatz für den tatsächlichen Kunden gefunden werden muss, ist es wichtig, dass trotzdem in regelmäßigen Abständen persönliche Rücksprache mit dem Kunden gehalten wird. Dies soll Diskrepanzen zwischen den vom Ersatzkunden verstandenen Anforderungen und den tatsächlichen Anforderungen verhindern.

Das gesamte Entwicklungsteam soll räumlich zusammen sein, d.h. alle Programmierer, Projektleiter, Tester, der Kunde und eventuell weiter beteiligte Personen sollen in einem Raum zusammen arbeiten. Die Programmierung selbst findet in Teams von zwei Personen, „Pair-Programming“, statt, wobei diese gemeinsam an einem Computer arbeiten sollen. In der Realität bedeutet dies, dass sich die zwei Personen über den zu erstellenden Code absprechen und einigen müssen, bevor eine Person direkt am Code schreibt und die andere über den Monitor beobachtet. Die Aufgabenverteilung der zwei Personen wechselt während des Zyklus durchgehend und bei Abschluss einer Iteration wechseln auch die Partner. Dadurch erhalten alle Programmierer ein Verständnis dafür, was gerade am Code geändert wird und welche Funktionalitäten bereits umgesetzt wurden. Zusätzlich wird durch das „Pair-Programming“ das Risiko des Verlustes von Informationen durch das Ausscheiden eines Programmierers aus dem Team verringert, da zumindest eine weitere Person direkt an derselben Arbeit beteiligt war.

Der ständige Partnerwechsel fördert des Weiteren die Kommunikation innerhalb des Teams und durch den direkten Kontakt mit den anderen Teammitgliedern wird das „Refactoring“ unterstützt. „Refactoring“ beschreibt die in XP durchgehende Tätigkeit des Verbesserns der Architektur, des Designs oder konkreter Stellen im Code. Durch persönlichen Kontakt mit dem zuständigen Programmierer lassen sich unklare Stellen im Code schneller klären und somit eine Gesamtverbesserung, entweder sofort oder in der nächsten Iteration, herbeiführen.

Der gesamte erstellte Code ist Eigentum des Teams. Kein Programmierer hat das Recht seinen Code geheim zu halten oder erst am Ende der Projektabwicklung abzuliefern. Jedes Projektmitglied soll sich für den gesamten Code verantwortlich fühlen und somit auch auf Fehler, die nicht in dem von ihm gerade bearbeiteten Programmteil sind, aufmerksam werden. Die einzelnen Teammitglieder sind jedoch nicht dazu berechtigt Änderungen an fremdem Code durchzuführen, d.h. sie dürfen immer nur jenen Teil bearbeiten, für den sie in der jeweiligen Iteration verantwortlich sind.

Wie beim „Pair-Programming“ bereits erwähnt, wechseln bei jeder Iteration die Partner und damit auch die Aufgaben in der Entwicklung durch. Damit Programmierer an nicht selbst erstelltem Code arbeiten können, ohne Zeit für das Einlesen in den fremden Code zu benötigen, muss ein Coding-Standard vorhanden sein. Dieser gibt vor wie der Code

strukturiert sein muss, wobei die Details für den Standard vom Team selbst bestimmt werden. Die einzige Regel für die Definition ist, dass der Standard vom gesamten Team verstanden und eingehalten werden muss.

Das gesamte Projekt soll, wenn möglich, ohne Überstunden durchgeführt werden bzw. sollen diese nur die Ausnahme bilden und weder besonders entlohnt noch erwartet oder gefördert werden. Überstunden zeugen bei XP von einer falschen Planung und Aufwandsabschätzung der „User-Stories“ und es muss daher bei der nächsten Iteration eine Überarbeitung der Aufwandsschätzungen durchgeführt werden, damit diese mit der Realität übereinstimmen.

Gegenstimmen

Als erstes konkretes Modell für die Anwendung agiler Methoden musste XP von vieler Seite heftige Kritik hinnehmen, im Anschluss werden die am häufigsten bemängelten Punkte kurz zusammengefasst.

XP besteht darauf, dass alle Werte und Praktiken verfolgt werden müssen und das Weglassen einer Praktik unwillkürlich den gesamten Ablauf negativ beeinflusst. In der Realität ist es jedoch oftmals sehr schwer sich an alle Vorgaben zu halten, besonders, wenn XP zum ersten Mal eingesetzt wird und beispielsweise die räumliche Zusammenlegung des Teams nötig wäre. (Stephens & Rosenberg, 2003, S. 57ff.)

Bereits zu Beginn wird davon ausgegangen, dass sich die „User-Stories“ mehrmals ändern und bei jeder Iteration überarbeitet werden. Es kann daher vorkommen, dass ein und derselbe Code immer wieder überarbeitet wird und kein wirklicher Fortschritt, messbar in neuer Funktionalität, stattfindet. Da sich die Anzahl der „User-Stories“ zusätzlich immer erweitern lässt, kann es geschehen, dass ein System nie fertiggestellt wird, wenn immer neue Funktionen vom Kunden hinzugefügt werden. (Stephens & Rosenberg, 2003, S. 249ff.)

Ein weiterer großer Kritikpunkt bei XP ist das Bedürfnis durchgehend einen Kunden vor Ort beim Entwicklungsteam zu haben. In der Realität ist es fast unmöglich dies mit dem wirklichen Auftraggeber umzusetzen, wodurch eine Ersatzperson gefunden werden muss, die das nötige Verständnis für das zu erstellende System besitzt. Gibt es für ein Projekt mehrere Auftraggeber, muss bei jeder Iteration ein Konsens mit allen Beteiligten gefunden werden, wodurch sich die Einbeziehung des Kunden weiter erschwert. (Stephens & Rosenberg, 2003, S. 117ff.)

Weitere von Stephens und Rosenberg (2003) hervorgehobene Kritikpunkte sind:

- „Pair-Programming“ wird vom Management als Verschwendung von Ressourcen gesehen, da Arbeit die zuvor von einer Person erledigt wurde, nun zwei Personen benötigt.
- Der Code und die Tests stellen das Design des Systems dar. Die gesamte Architektur der Software wird erst mit der Erstellung der Software generiert.
- YAGNI Ansatz kann in den späteren Entwicklungsphasen Zusatzkosten verursachen, da kein komplettes Systemdesign vorhanden ist und die Entwickler im Vorhinein nicht wissen, welche Funktionen wiederverwendet werden.

2.4.4.6. Scrum

Scrum stellt einen iterativen und inkrementellen Rahmen für die Projektdurchführung durch den Einsatz eines allgemeinen Ablaufschemas zur Verfügung, wobei keine spezifischen Mittel für die Ausführung vorgeschrieben werden. Es wird ein allgemeines Grundgerüst für die Projektmitarbeiter definiert in dem sie selbst alle Rechte besitzen, die für den Erfolg des Projektes notwendigen Entscheidungen zu treffen.

Innerhalb eines Scrum Projektes gibt es folgende drei Rollen mit jeweiligen Aufgaben (Gloger, 2008, S. 14f.):

- Product Owner

Anhand seiner Vision wird die Produktentwicklung gelenkt und geplant. Er arbeitet täglich mit dem Projektteam zusammen und ist dafür verantwortlich, dass die gesetzten Anforderungen erfüllt werden, sowie das Budget und der Zeitplan eingehalten werden. Der Product Owner verteilt Prioritäten für alle gesetzten Anforderungen und entscheidet für jede Iteration, welche Funktionen umgesetzt werden sollen. Im Scrum Prozess übernimmt er, als firmeninterne Person, die Vertretung für den tatsächlichen Kunden.

- Scrum Master

Der Scrum Master ist als Projektverantwortlicher ohne Weisungsbefugnis zu verstehen. Er kümmert sich darum, dass das Team effektiv arbeiten kann und der Scrum Prozess eingehalten wird. Zusätzlich vermittelt er zwischen dem Management, dem Product Owner, sowie dem Auftraggeber des Projektes und dem Projektteam (Koch, 2005, S. 257f.).

- Team

Das Projektteam darf eigenständig Entscheidungen zum Erreichen der gesetzten Ziele treffen und trägt im Gegenzug dazu auch alle Verantwortung für die erfolgreiche Fertigstellung des Projekts. Es entscheidet selbst, welches Arbeitspensum innerhalb einer Iteration erledigt werden kann und welche Vorgehensweisen am besten für die Zielerreichung geeignet sind. Das Team organisiert sich selbst und ist nur dazu verpflichtet, den Scrum Prozess bei der Durchführung einzuhalten.

Die Durchführung eines Projektes mithilfe von Scrum folgt nur einem sehr groben Ablaufplan ohne spezifische Definitionen für einzusetzende Werkzeuge oder Methoden und ist daher auf fast alle Sparten der Wirtschaft einfach anwendbar. Es ist möglich ein mit Scrum gemanagtes Projekt mit anderen spezifischen Methoden zu kombinieren, wodurch sich die Ausprägung der agilen Methoden von Projekt zu Projekt sehr unterscheiden kann.

Ein Projekt beginnt mit der Definition der Anforderungen an das zu erstellende Endprodukt im sogenannten „Produkt Backlog“. Vor dem Beginn jeder Iteration wird allen darin enthaltenen Aufgaben eine Priorität zugeteilt und der Aufwand für die Umsetzung der einzelnen Anforderungen abgeschätzt. In Summe stellt der „Produkt Backlog“ die noch zu leistende Arbeit für die Fertigstellung des Projektes dar. Während der Projektdurchführung können alle Stakeholder Veränderungen an den noch umzusetzenden Features vornehmen und

neue Funktionalitäten hinzufügen oder entfernen, wobei der „Product Owner“ schlussendlich die Verantwortung für alle enthaltenen Anforderungen trägt. (Koch, 2005, S. 258)

Für jede Iteration wählt das Projektteam jene Anforderungen aus dem „Produkt Backlog“ für die Umsetzung aus, die mit der höchsten Priorität bewertet worden sind und somit den meisten Wert für den Kunden generieren. Die aktuell ausgewählten Funktionen werden dabei als „Sprint Backlog“ bezeichnet und sollen innerhalb von dreißig Tagen, dem Zeitraum für einen sogenannten „Sprint“, umsetzbar sein. Während dieser dreißigtägigen Sprint Phase finden täglich fünfzehn-minütige Treffen des gesamten Projektteams statt, bei denen über den Zwischenstand und etwaige Probleme berichtet wird. Dieser „24-Stunden Sprint“ oder auch „tägliches Scrum“ genannte Prozess, hilft dem Team darüber zu entscheiden, welche Tätigkeiten bis zum nächsten Treffen erledigt werden müssen, sowie einen Überblick über die Aktivitäten des gesamten Teams zu erhalten. (Schwaber, 2004, S. 7f.)

Der Sprint wird abgebrochen, wenn während der Durchführung Probleme auftreten, die das Erreichen der gesetzten Ziele unmöglich machen oder die aktuell umzusetzenden Anforderungen ihre Gültigkeit verlieren (Koch, 2005, S. 260).

Den Abschluss einer Sprint Phase stellt ein vierstündiges Projektteammeeting dar, in welchem die Ergebnisse der Iteration präsentiert werden. Zusätzlich findet im Anschluss ein weiteres dreistündiges Meeting des Teams mit dem „Scrum Master“ statt, in welchem das Team über den Ablauf der Iteration reflektiert und Verbesserungsmöglichkeiten für die nächste Iteration gefunden werden sollen. Diese gesamte Reflexionsphase des Entwicklungsteams ist auch als Retrospektive bekannt und hilft zusätzlich dabei, noch nicht erfüllte Anforderungen aufzufinden.

Die genauen Zeitvorgaben, sowohl für einen Sprint als auch für alle Meetings innerhalb des Scrum Modells, sollen dem Team helfen keine Zeit durch unnötig lange Meetings oder Entwicklungsphasen zu verschwenden (Schwaber, 2004, S. 109). Zusätzlich soll das Team lernen, wie viel Arbeit es tatsächlich innerhalb eines Zyklus leisten kann und daraus für die nächste Iteration lernen (Bleek & Wolf, 2008, S. 151).

In Abbildung 2.16 wird der Ablauf eines Scrum Projektes, in Anlehnung an Schwaber (2004, S. 5) und Gloger (2008, S. 13), noch einmal schematisch dargestellt.

Die erste Planungsphase (Sprint Planung 1) bezieht sich dabei auf die Überarbeitung des „Produkt Backlog“ Kataloges, wobei die gesamten offenen Aufgaben (Produkt Backlog) noch einmal überarbeitet werden. Das zweite Planungsstadium (Sprint Planung 2) beschäftigt sich bereits mit der Analyse und Auswahl der spezifischen Features für den nächsten Sprint (Spring Backlog). Nach Fertigstellung der Sprintphase, soll eine neue Funktionalität erstellt worden sein. Für den Sprint ausgewählte, jedoch nicht fertiggestellte Anforderungen werden in der Retrospektive aufgedeckt und überarbeitet, bevor sie bei Beginn des nächsten Sprints zurück in den Gesamtkatalog (Produkt Backlog) gegeben werden.

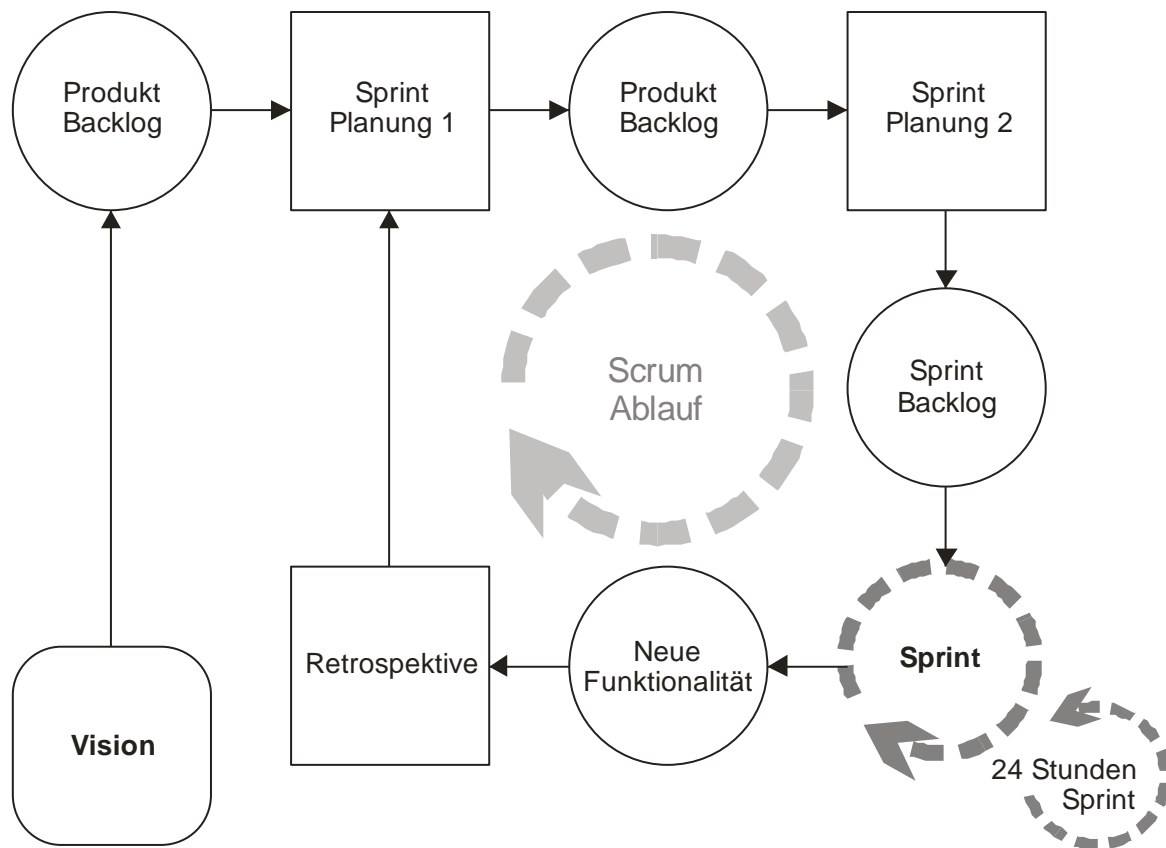


Abbildung 2.16: Schematischer Ablauf von Scrum basierend auf Schwaber (2004, S. 5) und Gloger (2008, S. 13)

Im Vergleich zu anderen agilen Managementmodellen kann Scrum auch mit relativ wenig Kundeninteraktion durchgeführt werden. Der Auftraggeber muss nur zu Projektbeginn seine Anforderungen spezifizieren und übergibt dem „Product Owner“ seine Funktion. Abhängig davon wie genau die initialen Anforderungen definiert wurden, gibt der Kunde zu den jeweiligen Zwischenprodukten der dreißigtägigen Sprints ein Feedback für die weitere Vorgehensweise.

Da jedes Projektmitglied jederzeit Anforderungen stellen oder verändern kann, muss sowohl das Team als auch der Product Owner regelmäßig den Verlauf des „Produkt Backlog’s“ überprüfen um dadurch auf Probleme im Projekt aufmerksam zu werden. Nach den ersten Iterationen dürfen sich die offenen Anforderungen noch stärker verändern, da hier die Stakeholder vielleicht erst einen wirklichen Überblick über das Projekt und ihre eigenen Erwartungen erhalten, danach muss jedoch eine stetige Abnahme der noch zu realisierenden Funktionen stattfinden. Tritt dies nicht ein, müssen Entscheidungen über das Ausmaß des Projektes und die wirklichen Anforderungen getroffen werden. (Koch, 2005, S. 258)

2.4.4.7. Crystal Clear

Die „Crystal“ Managementmodellfamilie wurde von Alistair Cockburn (2004) für die Anwendung agiler Konzepte in Abhängigkeit der Kritikalität des Projektes entworfen. Je nachdem wie viele Personen bei einem Projekt beteiligt sind und wie negativ die Auswirkungen etwaiger Fehler sind, wird eine stärker strukturierte Crystal Methode empfohlen. Die Namensgebung der Crystal Familie ist aufsteigend anhand der Farbskala, beginnend bei „Crystal Clear“, für unkritische Projekte, und „Crystal Red“, für sehr anspruchsvolle Projekte (Hruschka et al., 2009, S. 53ff.).

Abbildung 2.17 zeigt vier verschiedenen Crystal Modelle, abhängig von der Komplexität des Projektes nach Hruschka et al. (2009, S. 55).

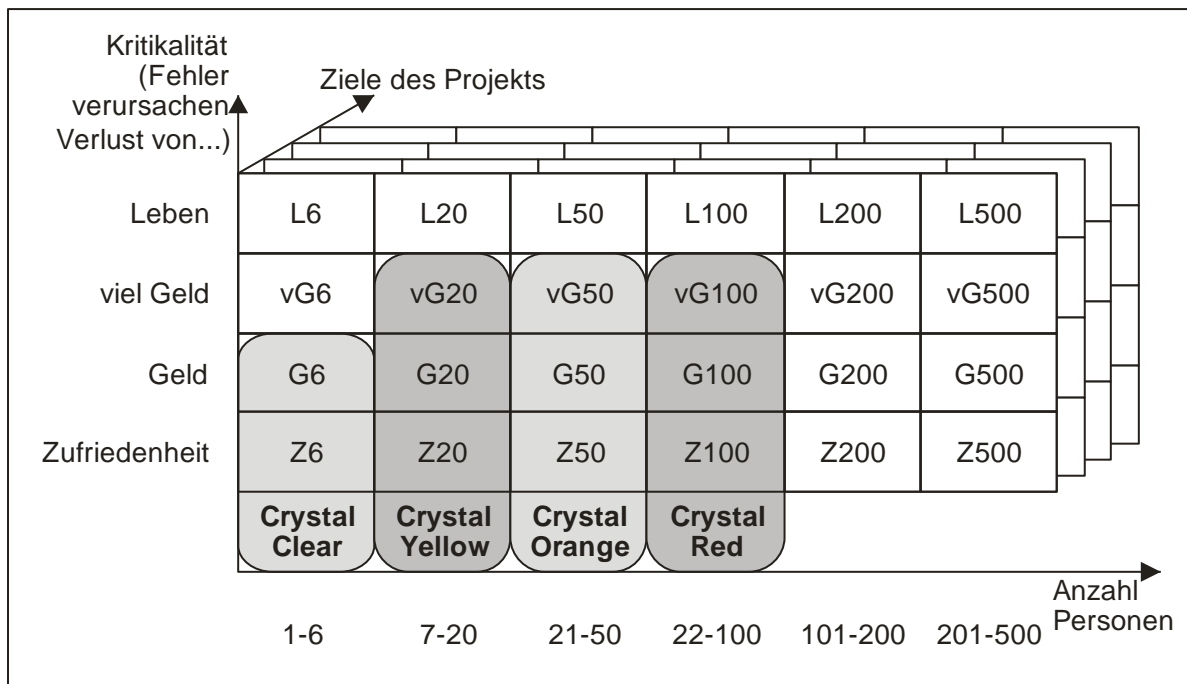


Abbildung 2.17: Die „Crystal“ Methodenfamilie (Hruschka et al., 2009, S. 55)

Allen Crystal Methoden liegen folgende Prinzipien zugrunde (Cockburn, 2006, S. 339):

- Die Menschen und ihre Kommunikation stehen im Mittelpunkt

Die für das Projekt notwendigen Mittel, Vorgehensweisen und Werkzeuge sind als unterstützende Elemente für das Team und seine Werte und Fähigkeiten anzusehen und sind weniger wichtig als die Individuen und ihre Wissensaustausch.

- Die gewählte Methode passt sich an das Projektumfeld an

Das Projektteam entscheidet wie flexibel und mit welchen Mitteln der Prozess gestaltet wird.

Grundsätzlich müssen bei allen Crystal Methoden nur zwei Regeln für die Entwicklung eingehalten werden (Cockburn, 2006, S. 339):

- Inkrementelle Entwicklung

Das Projekt muss in kleinen Schritten, Iterationen von maximal vier Monaten, durchgeführt werden. Es soll dabei immer der kürzeste mögliche Iterationszyklus gewählt werden, der einen Fortschritt in der Entwicklung liefert.

- Meetings vor und nach einer Iteration

Vor einer Iteration soll eine Planung des anstehenden Zyklus mit einer Planung der zu erledigenden Anforderungen durchgeführt werden. Wie detailliert die Planung einer Iteration durchgeführt wird, ist stark abhängig von der zusätzlich gewählten Managementmethode, da Crystal keine spezifischen Herangehensweisen vorschreibt.

Nach Abschluss einer Iteration muss ein Reflexionsmeeting stattfinden, bei dem sowohl auf Probleme innerhalb des Teams, als auch des zu entwickelnden Produktes eingegangen werden soll.

Von den verschiedenen Crystal Methoden wurden bisher erst drei detailliert, mit den vorgeschlagenen Zwischenschritten Aufgaben und Zwischenergebnissen, von Cockburn erläutert. Crystal Clear, Crystal Orange und Crystal Orange Web sind daher auch die einzigen Methoden, von denen ein Einsatz in der Wirtschaft bekannt ist. Crystal Orange Web wurde für die gleichzeitige Arbeit an mehreren Projekten, bei denen viele Informationen nicht nur generiert, sondern auch ständig verfügbar sein müssen entwickelt (Cockburn, 2006, S. 345f.).

2.4.4.8. Zusammenfassung

Agile Methoden können eine unterschiedlich starke Ausprägung in ihrer Definition besitzen. Abhängig von der gewählten Methode kann der Einsatz konkreter Techniken vorgeschrieben werden oder nur ein Grundgerüst für die Projektdurchführung zur Verfügung gestellt werden.

Eines der wichtigsten Ziele agiler Methoden ist die Projektabwicklung für späte Änderungen der Anforderung offen zu gestalten. Sowohl dem Kunden als auch den Projektmitarbeitern soll die Möglichkeit gegeben werden, während der Durchführung Verbesserungen und Änderungen umsetzen zu können. Aus diesem Ansatz agiler Methoden lässt sich folgende allgemeine Hypothese ableiten:

Hypothese 2: Je agiler die Projektdurchführung gestaltet wird, desto weniger stark wirken sich Anforderungsänderungen negativ auf den Projekterfolg aus.

Die Anwendung einer bzw. die Kombination mehrerer agiler Methoden kann den Erfolg eines Projektes dabei jedoch nicht garantieren. Wann und ob einem agilen Vorgehensmodell gefolgt werden soll, ist stark davon abhängig, wie sehr sich das ausgewählte Projekt für den Einsatz eignet. Ist beispielsweise für den Einsatz einer Methode eine gute und kontinuierliche Zusammenarbeit mit dem Auftraggeber unerlässlich, dieser jedoch nicht bereit die nötige Zeit für diese Art der Projektabwicklung zu investieren, so sollte der Einsatz des Modells überdacht werden.

Auch das firmeninterne Umfeld spielt eine wichtige Rolle bei der Auswahl, so werden dem Projektteam bei Scrum sehr viele Freiheiten gegeben, die es in traditionelleren Managementmodellen nicht besitzt (Koch, 2005, S. 258) wodurch die Firmenleitung dem Einsatz skeptisch gegenüber stehen kann.

2.4.5. Agile Systeme

Bei agilen Systemen handelt es sich um Produkte, welche über eine integrierte Flexibilität verfügen, durch die sie sich auch nach ihrer erstmaligen Inbetriebnahme noch an ändernde Anforderungen anpassen können. Im Gegensatz zu agilen Methoden wird die Reaktionsfähigkeit auf Änderungen somit nicht im Entwicklungsprozess, sondern erst im erstellten Endprodukt fokussiert.

Ein System muss laut Haberfellner und de Weck (2005) folgende drei Eigenschaften aufweisen, um agil zu sein:

- flexible Elemente im System, die einen Austausch und Anpassungen schnell und einfach ermöglichen
- Sensoren für die Überwachung der externen Gegebenheiten, die auf die Notwendigkeit von Änderungen am System aufmerksam machen
- Entscheidungsmechanismus der die Vor- und Nachteile, Kosten und Nutzen, von Änderungen abschätzt und nötige Veränderungen auslöst

Vor allem für kostspielige und langlebige Systeme ist die Fähigkeit sich auf Umwelteinflüsse anpassen zu können wichtig. Je nachdem wie groß die anfallenden Kosten für einen Austausch des Gesamtsystems sind und wie wahrscheinlich mit zukünftigen Änderungen der Anforderungen gerechnet werden muss, sollte über die Entwicklung agiler Systeme entschieden werden.

3. Empirische Analyse

Nachdem in Kapitel 2 die theoretischen Grundlagen der Untersuchung ausführlich diskutiert wurden, erfolgt im Anschluss die Erläuterung der empirischen Auswertung für die Überprüfung der erstellten Erklärungsmodelle.

Es wurde eine Querschnittstudie, mittels einmaliger Befragung von Projektmanagern bezüglich ihres zuletzt fertiggestellten Entwicklungsprojektes bestehend aus Hard- und Software durchgeführt. Die empirische Erhebung ermittelt allgemeine Daten des Projektes, sowie spezifische Praktiken der Produktentwicklung, die Erfolgskriterien und deren Erfüllung, sowie Informationen über allfällige Änderungen der Anforderungen.

In Kapitel 3.1 wird dazu die Vorgehensweise für die Erhebung der empirischen Daten beschrieben, bevor in Abschnitt 3.2 die Operationalisierung der einzelnen Werte erklärt wird. Kapitel 3.3 dient schlussendlich der Datenauswertung und Überprüfung der in Kapitel 2.2.3 und 2.4.4.8 aufgestellten Hypothesen.

3.1. Methodik der Datenerhebung

In Abschnitt 3.1 wird ein Überblick über die Auswahl der Stichprobe, die Vorgehensweise zur Datenerhebung, als auch den Aufbau des Fragebogens gegeben.

3.1.1. Aufbau des Fragebogens

Der für die empirische Analyse erstellte Fragebogen gliedert sich in vier Bereiche, welche anhand der Zielsetzung der Forschungsarbeit konzipiert wurden.

Bewertung des Projekterfolges

Im ersten Teil des Fragebogens erfolgt die Bewertung des Erfolges für das ausgewählte Projekt. Dabei wird sowohl eine Gewichtung der einzelnen Faktoren für die Messung, als auch der Grad der Erfüllung des jeweiligen Faktors abgefragt, um eine möglichst realistische Kennzahl zu erhalten. Zusätzlich wird eine allgemeine Einschätzung über den Gesamterfolg des Projektes vorgenommen.

Erfassung der Anforderungsänderungen

Der zweite Teil der Umfrage dient der Erfassung der während der Entwicklungsprojektzeit aufgetretenen Anforderungsänderungen. Für die Untersuchung wurden diesbezüglich nur Änderungen erfasst, welche der Auftraggeber selbst veranlasst hat.

Ausprägung agiler Methoden

In diesem Abschnitt des Fragebogens werden einzelne Konzepte agiler Methoden auf den Grad ihrer Anwendung innerhalb des ausgewählten Projektes ermittelt. Zusätzlich werden bekannte Projektmanagementmethoden abgefragt.

Allgemeine Projektdaten

Die allgemeinen Projektdaten werden im letzten Teil des Fragebogens abgedeckt. Hier werden beispielsweise die Laufzeit, das vorhandene Budget, als auch die Art des Auftraggebers ermittelt.

Um eine möglichst hohe Rücklaufquote mit verwertbaren Antworten zu erreichen, wurde darauf geachtet die Anzahl „offener Fragen“ auf ein Minimum zu beschränken und diese nur bei allgemeinen Fragen zur Firma bzw. dem Projekt einzusetzen. Es wurden Antwortmöglichkeiten auf Skalenbasis bevorzugt um ein nachträgliches Hinzufügen von Messfehlern durch die falsche Interpretation von Antworten zu verhindern. (Groves et al., 2004)

Insgesamt umfasst die Umfrage 56 Fragen, von denen 46 als Pflichtfragen definiert wurden, welche für den erfolgreichen Abschluss des Fragebogens nötig sind. Alle Pflichtfragen wurden explizit gekennzeichnet und zu Beginn jeden Abschnittes wird zusätzlich darauf hingewiesen, dass diese für die Fertigstellung ausgefüllt werden müssen.

Um eine höhere Lesbarkeit des Fragebogens zu gewähren, wurde darauf geachtet, dass bei der digitalen Anzeige die Skalenbenennung immer zusammen mit der Frage angezeigt wird. Es wurden dazu die Fragen zur Anwendung von agilen Methoden in drei Bereiche unterteilt, welche den kognitiven Aufwand für den Befragten minimieren sollen.

Zusätzlich wurde eine Skip Logik im Bereich der Fragen zu den Anforderungsänderungen eingebaut, welche es dem Befragten ermöglicht, alle Fragen des Abschnittes zu überspringen, falls es zu keinen Änderungen der Anforderungen gekommen sein sollte. Dadurch wird verhindert, dass bei dem sehr seltenen Fall eines Projektes ohne Veränderungen nicht zutreffende Fragen beantwortet werden müssen.

Der Fragebogen wurde sowohl auf die Formulierung der Fragestellung, als auch auf die Bereitschaft von Firmen, spezifische Daten preiszugeben, mithilfe einer Fokusgruppe, getestet. Es wurden dazu sieben repräsentative Firmen für die Grundgesamtheit der zu untersuchenden Projekte ausgewählt, mit denen der gesamte Fragebogen bei einem persönlichen Interviewtermin besprochen wurde. Fokus dieses Tests war die Eindeutigkeit der Fragestellung, als auch die Antwortbereitschaft der Firmen bezüglich interner Daten. Zusätzlich wurde der Fragebogen auf Gestaltung und Zeitaufwand beurteilt.

Aus den Interviews mit den Fokusgruppenteilnehmern ging hervor, dass die Beantwortung zu firmeninternen Daten, wie beispielsweise Projektbudget oder Jahresumsatz relativ gering ist. Da diese Daten oft erst nachgeschlagen werden müssen und sich so die Bearbeitungszeit deutlich verlängert, könnten die Interviewpartner dadurch zum Abbruch des Fragebogens veranlasst werden. Nicht relevante Informationen für die Studie wurden aus diesem Grund zur Beantwortung als nicht verpflichtende Fragen deklariert. Ferner wurde die Skalenbezeichnung überarbeitet, um eine einfachere Einstufung vornehmen zu können. Hinsichtlich der Länge und Dauer des Fragebogens mussten keine Veränderungen vorgenommen werden, da die Beantwortung maximal fünfzehn Minuten in Anspruch nahm und dies für alle Testgruppenteilnehmer als akzeptabel eingestuft wurde.

3.1.2. Auswahl der Stichprobe

Für die empirische Erhebung der benötigten Daten wurde eine klare Definition der Stichprobe vorgenommen. Da sich diese Arbeit speziell auf den Einsatz agiler Methoden bei der gemeinsamen Entwicklung von Hard- und Softwarekomponenten bezieht, wurde die Auswahl der betrachteten Firmen im ersten Schritt anhand ihrer Branchenzugehörigkeit nach

österreichischer Klassifikation der wirtschaftlichen Tätigkeiten (ÖNACE) eingeschränkt. Zusätzlich wurde darauf geachtet, dass die in die engere Auswahl gezogenen Firmen in die Bereiche der Herstellung, Produktion, Entwicklung und Konstruktion von Geräten bzw. Komponenten fallen.

Es wurde anhand dieser Einschränkungen eine Liste mit 637 österreichischen Firmen erstellt, welche in die Grundgesamtheit aufgenommen wurden. Für die empirische Analyse wurden die Firmen aus dieser ausgewählt und genauer auf ihre Qualifikation untersucht.

Für das jeweils betrachtete Projekt der ausgewählten Firma musste ein abteilungsexterner Auftraggeber vorhanden sein. Es durfte sich somit nicht um ein reines Forschungsprojekt für die eigene Firma handeln. Zusätzlich musste sowohl Hard- als auch Software für das Endprodukt erstellt worden sein. Anhand dieser Einschränkungen wurde das zuletzt durchgeführte Projekt, welches diese Voraussetzungen erfüllt, vom Projektleiter selbst ausgewählt.

3.1.3. Erhebungsmethode

Für die Erhebung der empirischen Daten wurden die Projektmanager und die Kapitel 2.1.3 identifizierten Projektsponsoren herangezogen. Die dadurch ausgewählten Personen besaßen nicht nur umfassende Kenntnisse über den internen Projektablauf, sondern konnten aufgrund ihrer Zusammenarbeit mit den verschiedenen Stakeholdern auch Auskunft über die zu erfüllenden Erfolgskriterien geben.

Für die in die Grundgesamtheit aufgenommenen Firmen wurden die jeweils zuständigen Personen telefonisch kontaktiert und in einem einleitenden Gespräch sowohl die Zielsetzung und wissenschaftliche Bedeutung der Forschungsstudie genauer erklärt, als auch die Tauglichkeit der tatsächlich durchgeführten Projekte individuell überprüft. Aufgrund dieser persönlichen Absprache der Einschränkungen für akzeptable Projekte, minimierte sich die Anzahl der in Frage kommenden Unternehmen für die Studie auf 397 potentielle Firmen.

Allen beteiligten Kontaktpersonen wurde die Möglichkeit geboten, zwischen einer Teilnahme mittels persönlichem Telefoninterview und dem webgestützten Fragebogen zu wählen, wobei der Termin für die Teilnahme per Telefon von den Projektverantwortlichen frei entschieden werden konnte. Bei Bevorzugung des Online-Fragebogens erhielten die Teilnehmer den Zugangslink, sowie eine abermalige kurze Zusammenfassung der Zielsetzung der Studie per Email.

Um einen Anreiz für die Teilnahme zu bieten, erhielten alle Teilnehmer durch Eingabe ihre Kontaktdaten bzw. zumindest einer gültigen Email-Adresse die Möglichkeit die ausgewerteten Umfrageergebnisse zugesandt zu bekommen. Die Unternehmen können somit durch ihre Teilnahme direkt von der Studie profitieren und sich selbst anfallende Forschungskosten ersparen.

Da im Zuge der Umfrage persönliche Daten der Teilnehmer als auch firmeninterne Informationen gesammelt wurden, erfolgte die Erfassung auf einem internen Server der Technischen Universität Graz, womit die Möglichkeit eines Zugriffs von unberechtigten

Dritten verhindert wurde. Die Teilnehmer wurden darüber aufgeklärt, dass ihre Daten nur anonymisiert ausgewertet und persönliche Informationen keinesfalls weitergegeben werden.

Die Erhebung der Daten erfolgte über einen Zeitraum von vier Monaten; von Anfang November 2009 bis Ende Februar 2010. Trotz der sehr spezifischen Anforderungen an die Art des Projektes, sowie der Einschränkung der Teilnehmer auf Projektverantwortliche, konnte eine Rücklaufquote von 18 Prozent erreicht werden. Von den 397 in Betracht kommenden Firmen wurden 107 Fragebögen ausgefüllt, von welchen 72 vollständig beantwortet und für die Auswertung herangezogen wurden.

3.1.4. Charakteristika der Stichprobe

In diesem Abschnitt wird die Stichprobe anhand allgemeiner Firmen- und Projektmerkmale näher beschrieben. Es soll ein Überblick über die erfassten Firmen gegeben werden um somit einen Rückschluss auf die Grundgesamtheit zu ermöglichen.

Anhand ausgewählter Charakteristika der in Kapitel 2.1.2 beschriebenen Klassifizierungsmöglichkeiten von Projekten werden die betrachteten Projekte weiter definiert.

Die Analyse der Stichprobe anhand der Gesamtanzahl der beschäftigten Mitarbeiter in Abbildung 3.1 zeigt, dass Firmen unterschiedlicher Größe in der Erhebung vertreten sind. Abbildung 3.2 stellt die Anzahl der Mitarbeiter für das ausgewählte Projekt dar, wobei auch hier verschieden große Teams gut vertreten sind. Somit kann angenommen werden, dass die Stichprobe eine repräsentative Aussage ermöglicht.

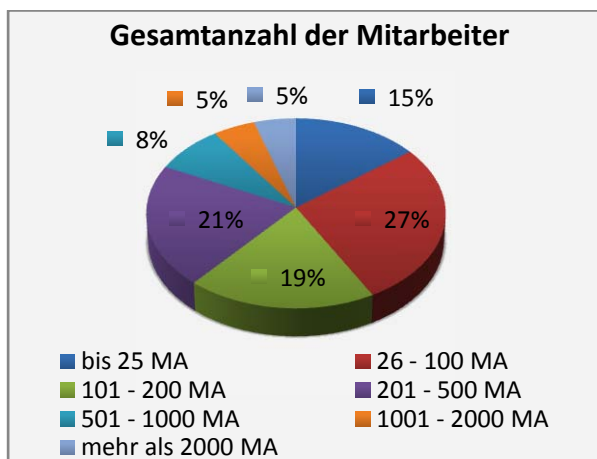


Abbildung 3.1: Gesamtanzahl der Mitarbeiter

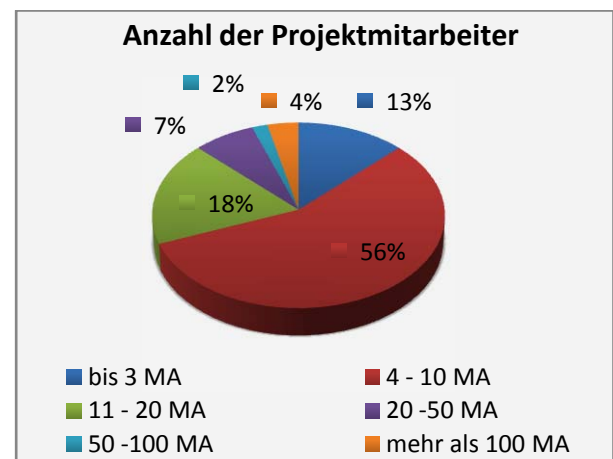


Abbildung 3.2: Anzahl der Projektmitarbeiter

Für eine allgemeine Klassifikation lässt sich ebenfalls die Dauer des Projektes heranziehen. Wie in Abbildung 3.3 ersichtlich ist, konnten 50 Prozent aller untersuchten Projekte innerhalb eines Jahres abgeschlossen werden, wobei nur ein einziges Projekt länger als drei Jahre dauerte. Die Mehrheit aller Projekte hatte eine Laufzeit von zwei Jahren.

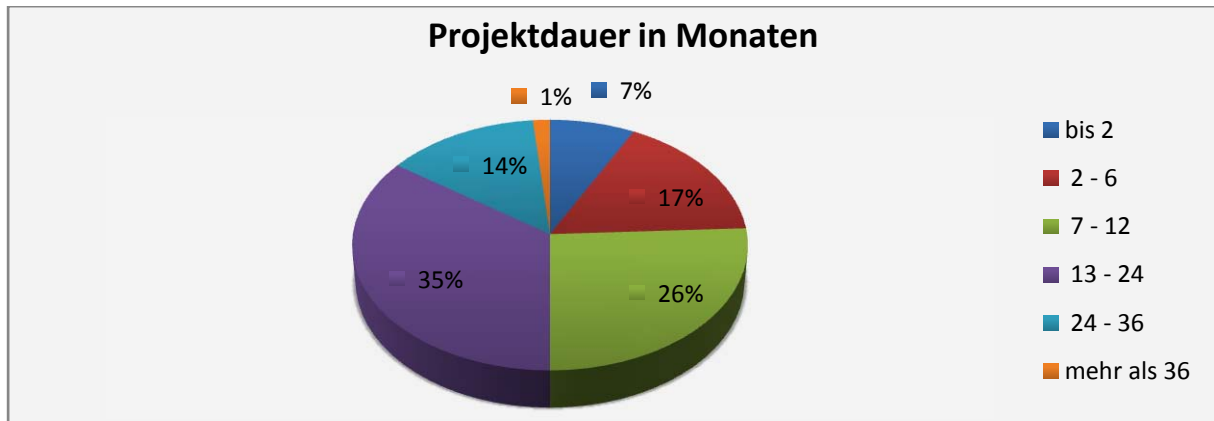


Abbildung 3.3: Projektdauer in Monaten

Abbildung 3.4 stellt die allgemeine Komplexität des betrachteten Projektes im Vergleich zu den üblicherweise durchgeführten Projekten der Firma dar. Die Bewertung der Komplexität erfolgte anhand einer siebenstufigen Likert-Skala, wobei insgesamt nur vier Unternehmen ihr ausgewähltes Projekt überdurchschnittlich komplex beurteilten. Der Großteil der Projekte wurde als „eher simpel“ eingestuft und entsprach dem Standard des Projektteams.

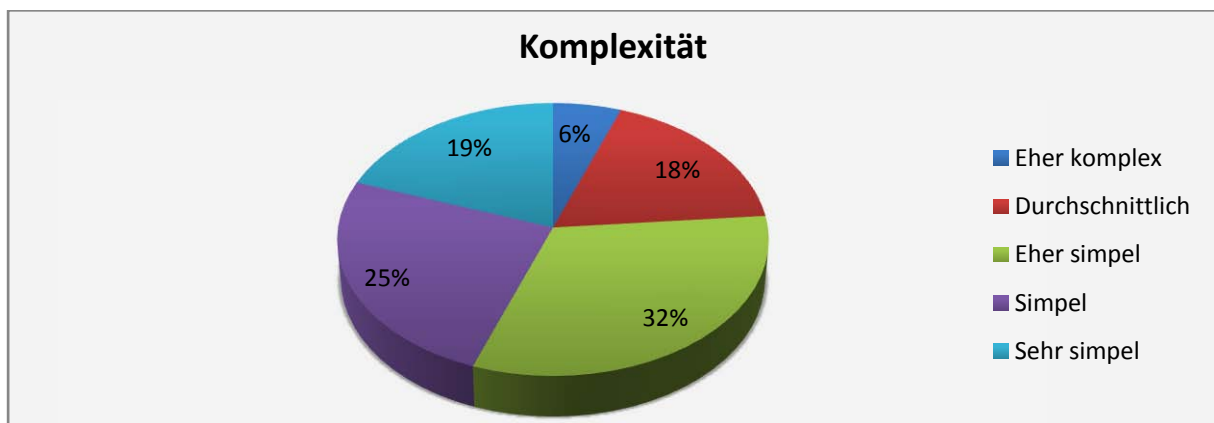


Abbildung 3.4: Komplexität des betrachteten Projektes

Wesentlich an der Komplexitätsbewertung ist, dass keine eindeutige Aussage für sehr komplexe Projekte anhand der Erhebung getroffen werden kann, da diese nicht in der Stichprobe enthalten sind.

3.2. Operationalisierung und Messung der Variablen

Kapitel 3.2 dient der Operationalisierung der Dimensionen der Agilität, welche aus den in Kapitel 2.4 beschriebenen theoretischen Modellen abgeleitet wurden, sowie den aufgetretenen Anforderungsänderungen und dem Projekterfolg. Die entwickelten Skalen basieren auf den in der Literatur vorkommenden Einteilungen und wurden zur Erstellung quantitativer Werte angepasst und erweitert.

In den folgenden Abschnitten werden die verwendeten Variablen und zugrundeliegenden Indikatoren für die drei betrachteten Aspekte dieser Arbeit näher erklärt, sowie die Vorgehensweise zur Operationalisierung dargestellt.

3.2.1. Operationalisierung des Projekterfolges

Um den Projekterfolg messbar zu machen, wird mit Hilfe einer gekürzten Version der in Kapitel 2.3.2.2 beschriebenen „Success Slider“ Methode von Rob Thomsett (2002) eine Klassifizierung der Projektziele, sowie deren Erfüllung vorgenommen. Es handelt sich dabei um eine subjektive Quantifizierung des Projekterfolges, wobei eine objektive Berechnung nur mit firmeninternen Daten und genauer Betrachtung jedes einzelnen Projektes möglich wäre.

Das Konstrukt des Projekterfolges baut sich in dieser vereinfachten Version aus den drei Faktoren des „Iron Triangle“ (Atkinson, 1999, S. 338), der Zufriedenstellung des Auftraggebers und der Wertschaffung für das Unternehmen auf. Das Originalkriterium „meet quality requirements“ wurde unter „Erfüllung der Projektspezifikationen“ vereinfacht zusammengefasst, um Diskrepanzen bei der Definition für Qualität auf Seiten der Befragten vorzubeugen. Die Zufriedenheit des Auftraggebers wurde aus dem Original Kriterium der „have satisfied stakeholders“ abgeleitet, um einerseits die Beantwortung für den Fragebogen zu erleichtern und andererseits das Hauptaugenmerk für die Berechnung des Projekterfolges auf die Zufriedenstellung des Kunden zu legen.

Variablen	Kategorie	Indikator	Skalenart	Dimension
Projekterfolg	Gewicht der Erfüllung	Zufriedenstellung des Auftraggebers	Likert	Fünfstufig
		Erfüllung der Projektspezifikationen	Likert	Fünfstufig
		Einhaltung des Projektbudgets	Likert	Fünfstufig
		Termingerechte Fertigstellung	Likert	Fünfstufig
		Wertschaffung für das Unternehmen	Likert	Fünfstufig
	Grad der Erfüllung	Zufriedenstellung des Auftraggebers	Likert	Siebenstufig
		Erfüllung der Projektspezifikationen	Likert	Siebenstufig
		Einhaltung des Projektbudgets	Likert	Siebenstufig
		Termingerechte Fertigstellung	Likert	Siebenstufig
		Wertschaffung für das Unternehmen	Likert	Siebenstufig

Tabelle 3.1: Faktoren für Projekterfolg

Die Auswertung belegt, dass weitere Kriterien, als nur jene des „Iron Triangles“, für die Erfolgsbewertung eine wichtige Rolle spielen. Abbildung 3.5 veranschaulicht die Gewichtung der gewählten Faktoren durch die Projektverantwortlichen. Die „Zufriedenstellung des Auftraggebers“ wurde von allen Befragten als wichtigstes Kriterium für ein erfolgreiches

Projekt gewählt. Der Durchschnittswert aller 72 Bewertungen stellt die „Einhaltung des Projektbudgets“ an die letzte Stelle.

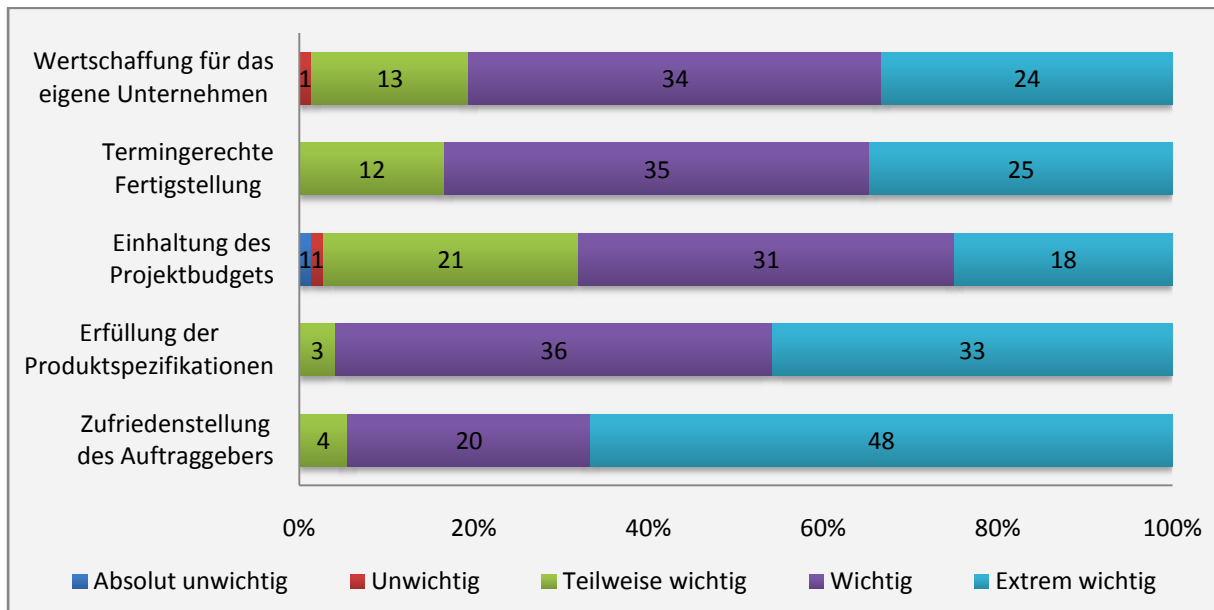


Abbildung 3.5: Gewicht der Erfüllung für den Projekterfolg

Für die Berechnung eines quantitativen Wertes des Projekterfolges wird zuerst für jedes Kriterium der maximal erreichbare Wert berechnet. Die fünf so entstandenen Werte ergeben in Summe den absoluten Erfolg des Projektes. Das Gewicht der Erfüllung kann dabei einen Prozentwert zwischen 0% und 100% einnehmen, welcher in 25% Schritte unterteilt ist. Aufgrund der verwendeten siebenstufigen Likert-Skala, welche Werte zwischen 0 und 6 zulässt, wird für die „Konstante möglicher Punkte“ der Wert 6 als maximal erreichbare Punkteanzahl eingesetzt. Dieser Vorgang beruht auf den von Rob Thomsett vorgestellten Überlegungen zur Überprüfung des Erreichens der gesetzten Ziele, wobei hier alle Kriterien in die Bewertung einfließen und nicht nur jene, die eine hundertprozentige Wertigkeit besitzen (Thomsett, 2002).

$$\text{Absoluter Erfolg} = \text{Gewicht der Erfüllung} \times \text{Konstante möglicher Punkte}$$

Formel 3.1: Absoluter Erfolg pro Kriterium des Projekterfolges

Der relative Erfolg jedes einzelnen Kriteriums wird mithilfe der dem Erfüllungsgrad zugrunde liegende Punkteanzahl multipliziert und mit der Wichtigkeit der Erfüllung (Gewichtung) errechnet. Die Werte für den Grad der Erfüllung können dabei zwischen 0 und 6 liegen.

$$\text{Relativer Erfolg} = \text{Gewicht der Erfüllung} \times \text{Grad der Erfüllung}$$

Formel 3.2: Relativer Erfolg pro Kriterium des Projekterfolges

Die Summe der relativen Erfolge der Kriterien, geteilt durch den absoluten Erfolg, ergibt den prozentuellen Projekterfolg.

$$\text{Projekterfolg} = \frac{\sum \text{Relativer Erfolg}}{\sum \text{Absoluter Erfolg}} \times 100$$

Formel 3.3: Projekterfolg in Prozent

Die Auswertung der 72 vollständig ausgefüllten Fragebögen ergab eine durchschnittliche Erfolgsquote von 79 Prozent anhand dieser modifizierten „Success Slider“ Methodik. Abbildung 3.6 zeigt die Verteilung des Projekterfolgs aller ausgewerteten Firmen. Bei insgesamt 68 Firmen lag das berechnete Ergebnis über 50 Prozent. 40 Projekte konnten 80 Prozent der gesetzten Anforderungen erfüllen.

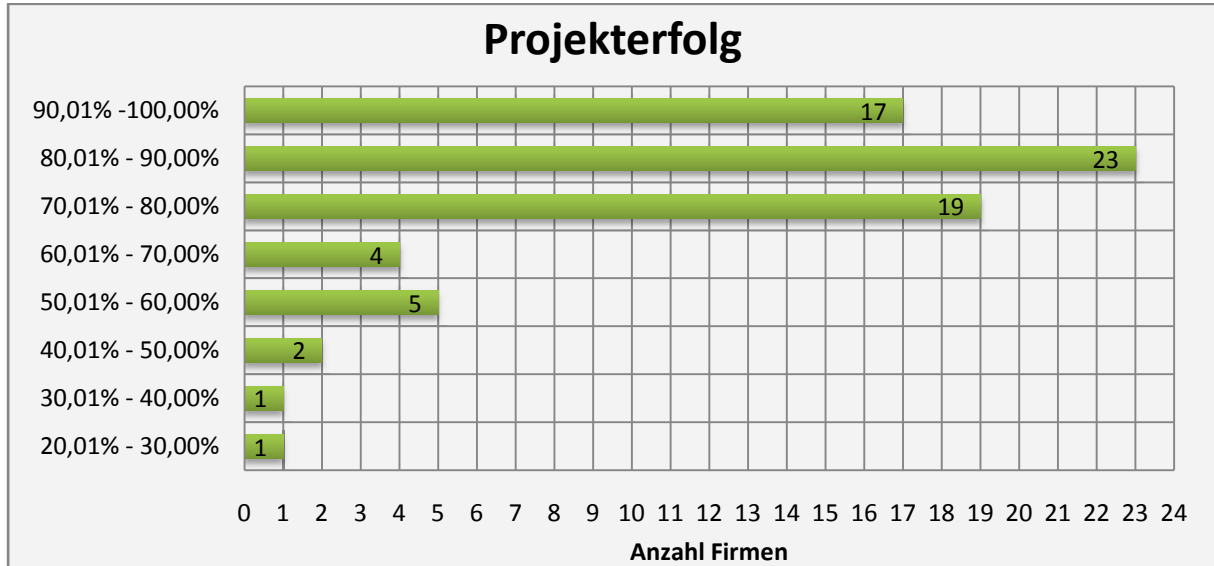


Abbildung 3.6: Berechnete Projekterfolgsverteilung

Zusätzlich wird anhand einer einfachen Skala von eins bis zehn eine subjektive Bewertung des Projekterfolgs durch den Umfrageteilnehmer vorgenommen. Dies dient zur Gegenüberstellung mit dem durch die „Success Slider“ Methode erhaltenen Ergebnisse. Die Datenauswertung ergab, dass der durchschnittlich geschätzte Projekterfolg bei 76 Prozent liegt und somit eine größtenteils sehr realistische Einschätzung durch die Umfrageteilnehmer vorgenommen wurde.

3.2.2. Operationalisierung der Anforderungsänderungen

Für die empirische Erhebung werden nur Anforderungsänderungen betrachtet, die durch den Kunden direkt oder indirekt verursacht wurden. Indirekte Anforderungsänderungen beziehen sich in diesem Zusammenhang auf Änderungen, die aufgrund von neuen oder geänderten Projektzielen entstanden sind bzw. die der Kunde in Folge externer, auf ihn einwirkender, Einflüsse fordern musste. Eine explizite Unterscheidung in direkte und indirekte Anforderungsänderungen durch den Kunden wird im Fragebogen nicht vorgenommen, da dies einer Beurteilung durch den Kunden selbst bedürfen würde.

Die Summe der Änderungen wird prozentuell über die Gesamtmenge der Anforderungen abgefragt. Anhand einer Kombination der von Dennis Lock (2007) und dem Project Management Institut (2004) vorgeschlagenen Modelle, wird zusätzlich die relative Anzahl und die Schwere der Änderungen zu vier, allen Projekten zugrundeliegenden, Zeitpunkten bestimmt.

In Abbildung 3.7 werden die zur Auswertung herangezogenen Phasen der Projektdurchführung veranschaulicht.

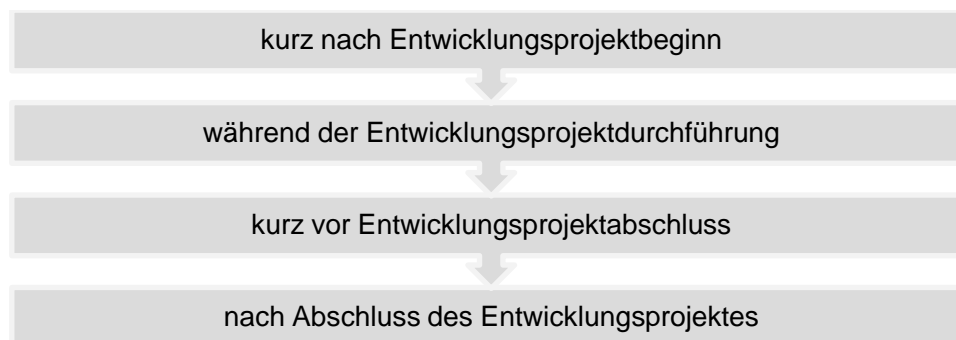


Abbildung 3.7: Relevante Entwicklungsprojektphasen für Anforderungsänderungen

Tabelle 3.2 gibt einen Überblick über die verwendeten Indikatoren zur Operationalisierung der Anforderungsänderungen. Es wird illustriert, welche Indikatoren innerhalb der Kategorien der Anforderungsänderungen abgefragt werden, sowie die dazu verwendeten Skalenarten und ihre Dimension.

Variable	Kategorie	Indikator	Skalenart	Dimension
Anforderungsänderungen	Prozent der Anforderungsänderungen	Summe der Prozent an Anforderungsänderungen	-	Eingabefeld
	Anzahl der Anforderungsänderungen	kurz nach Entwicklungsprojektbeginn	Likert	Siebenstufig
		während der Entwicklungsprojektdurchführung	Likert	Siebenstufig
		kurz vor Entwicklungsprojektabschluss	Likert	Siebenstufig
		nach Abschluss der Entwicklungsprojektes	Likert	Siebenstufig
	Umfang der Anforderungsänderung	kurz nach Entwicklungsprojektbeginn	Likert	Siebenstufig
		während der Entwicklungsprojektdurchführung	Likert	Siebenstufig
		kurz vor Entwicklungsprojektabschluss	Likert	Siebenstufig
		nach Abschluss der Entwicklungsprojektes	Likert	Siebenstufig

Tabelle 3.2: Faktoren für die Berechnung einer Kennzahl der Anforderungsänderungen

Um eine Kennzahl für die gesamten Anforderungsänderungen zu erhalten, wird im ersten Schritt die Anzahl der Anforderungsänderungen anhand ihrer Wertigkeit aufsummiert, wofür die siebenstufige Likert-Skala dazu mit Werten von null bis sechs hinterlegt ist. Die angegebenen Prozent der Anforderungsänderungen werden durch den erhaltenen Wert der Anzahl an Änderungen dividiert, um einen Faktor für die Berechnung der Prozent je Projektabschnitt zu erhalten.

$$Faktor = \frac{\text{Prozent Anforderungsänderungen}}{\Sigma \text{ Anzahl Änderungen}}$$

Formel 3.4: Faktor für die prozentuelle Aufteilung der Änderungen

Durch Multiplikation der gewählten Anzahl pro Projektabschnitt mit dem berechneten Faktor werden die Gesamtprozent der Änderungen auf den jeweiligen Abschnitt aufgeteilt. In Formel 3.5 wird der dafür angewandte Rechenweg veranschaulicht.

$$PA = Faktor \times \text{Anzahl Projektabschnitt}$$

Formel 3.5: Prozentuelle Abschätzung der Anforderungsänderungen

Der Umfang der Änderungen im jeweiligen Projektabschnitt wird durch eine siebenstufige Likert-Skala erfasst und fließt anhand der Bewertung in 1/6 Schritten in die Berechnung ein. In Tabelle 3.3 sind die zur Berechnung verwendeten Multiplikatoren für die zur Auswahl stehenden Antworten abgebildet.

Sehr groß	Groß	Eher groß	Unentschieden	Eher klein	Klein	Sehr klein
1	0,8333	0,6667	0,5	0,3333	0,1667	0,0000

Tabelle 3.3: Multiplikatoren des Umfangs der Anforderungsänderungen

Für jede Phase des Projektes ergibt sich eine eigene Kennzahl der Anforderungsänderungen, durch Multiplikation der berechneten Prozentzahl mit dem Multiplikator des Umfangs für den jeweiligen Zeitpunkt.

$$Kennzahl \text{ der Projektphase} = \text{Umfang} \times \text{Änderungen in Prozent}$$

Formel 3.6: Kennzahl der Anforderungsänderungen pro Projektphase

In Summe ergibt sich aus den errechneten Kennzahlen der Projektphasen der Wert für die angegebenen Anforderungsänderungen.

$$Kennzahl \text{ des Projektes} = \Sigma \text{ Kennzahlen der Projektphasen}$$

Formel 3.7: Kennzahl der Anforderungsänderungen des Projektes

Die Kennzahl der Anforderungsänderungen nimmt durch die angewandten Rechenschritte einen Wert zwischen 0,00 und 100,00 ein. Die Verteilung der berechneten Kennzahlen für die Anforderungsänderungen wird in Abbildung 3.8 illustriert. Von den 72 betrachteten Projekten gab es bei insgesamt 5 keine Änderungen der Anforderungen, die Einfluss auf den Erfolg des Projektes nahmen. Für den Großteil der Projekte, 49 in Summe, liegt die Kennzahl der Änderungen zwischen 0,01 und 20 und weist auf eine sehr moderate Beeinflussung des Projektablaufs hin.

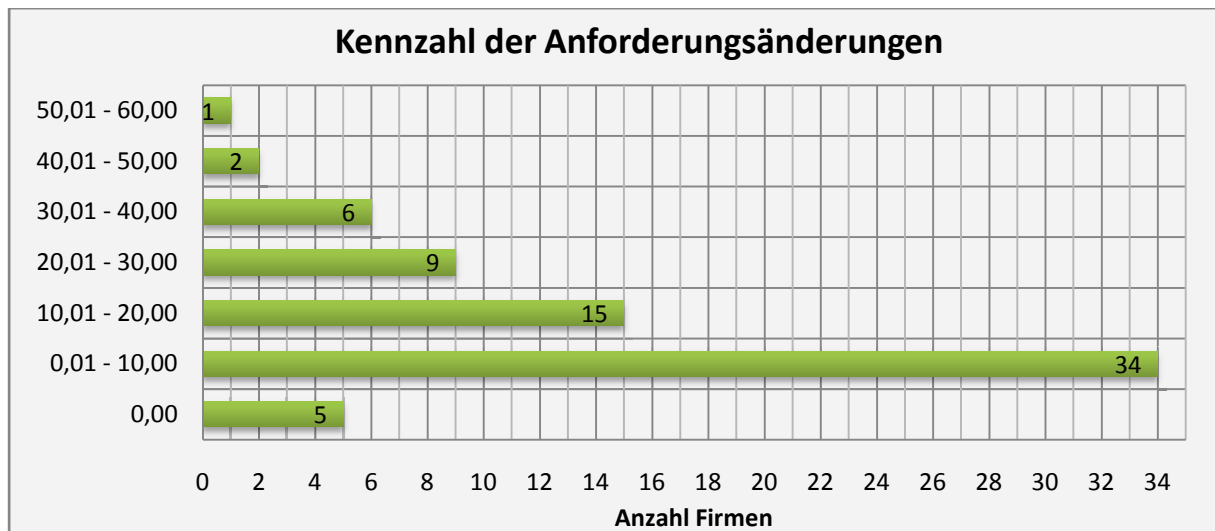


Abbildung 3.8: Kennzahlenverteilung der Anforderungsänderungen

3.2.3. Operationalisierung der agilen Methoden

Da Agilität als Variable selbst nicht direkt messbar ist, wurde, anhand der in Kapitel 2.4 dargestellten theoretischen Ansätze, ein Modell zur Erfassung ausgewählter Komponenten mittels Indikatoren erstellt. Anhand ausgewählter Prinzipien agiler Methoden wird eine Bewertung der Verwendung agiler Methoden vorgenommen. Es handelt sich dabei nicht um eine vollständige Beurteilung von Agilität selbst, sondern um eine Messung, der für diese Arbeit in Betracht gezogenen agilen Ansätze und ihrer Anwendung.

Zur Auswahl der zu verwendenden Indikatoren zur Operationalisierung wurden die Prinzipien des „Agile Manifesto“ (Beck et al., 2001) mit den weiteren, in der Literaturrecherche vorgeschlagenen Methoden, aus Kapitel 2.4.4 dieser Arbeit, abgestimmt.

Die Ausprägung der agilen Methoden wird über insgesamt vierundzwanzig Multiple-Item Indikatoren im Fragebogen gemessen, welche sich in fünf theoretische Zweige agiler Methoden zusammenfassen lassen. Da agile Methoden und der Einsatz spezieller Praktiken in der Entwicklung ein sehr komplexes Gebiet darstellt, müssen die verschiedenen, zu einem theoretischen Konstrukt zusammengefassten Indikatoren in der Wirtschaft nicht unbedingt gemeinsam auftreten. Die thematische Zusammenfassung dient der Extraktion von fünf wesentlichen Aspekten der Anwendungsbereiche und hilft die Grundlagen innerhalb dieser abzudecken.

Konstrukt	Indikator	Skalenart	Dimension
Mitarbeiter Agilität	Motivation der Projektmitarbeiter	Likert	Siebenstufig
	Bereitschaft für Veränderungen	Likert	Siebenstufig
	Erfahrenheit der Mitarbeiter	Likert	Siebenstufig
	Kenntnisse der Mitarbeiter	Likert	Siebenstufig
	Komplexität des Wissensaustauschs	Likert	Siebenstufig
	Räumliche Verteilung des Teams	Likert	Siebenstufig
	Arbeitsklima innerhalb des Teams	Likert	Siebenstufig
Fokus auf Kundennutzen	Integration des Kunden/Endnutzers	Likert	Siebenstufig
	Feedback des Kunden/Endnutzers	Likert	Siebenstufig
	Priorisierung der Anforderungen	Likert	Siebenstufig
Zyklische Entwicklung	Erstellung von Zwischenprodukten	Likert	Siebenstufig
	Zugänglichkeit der Zwischenprodukte für Kunden	Likert	Siebenstufig
	Zyklische/Iterative Entwicklung	Likert	Siebenstufig
	Kundenfeedback zu Zwischenprodukte zur Produktverbesserung	Likert	Siebenstufig
Flexibilität im Prozess	Zeitpunkt der Projektplanung	Likert	Siebenstufig
	Aktive Verbesserung des Projektprozesses	Likert	Siebenstufig
	Möglichkeit der Mitarbeiter Prozesse selbst zu verändern	Likert	Siebenstufig
Flexibilität im Endprodukt	Funktionalität des Endprodukts	Likert	Siebenstufig
	Dokumentation	Likert	Siebenstufig
	Simplizität des Endprodukts	Likert	Siebenstufig
	Wiederverwendung vorhandenen Wissens	Likert	Siebenstufig
	Reihenfolge der Hard-/Softwareentwicklung	Likert	Siebenstufig
	Modularität	Likert	Siebenstufig
	Architektur auf Veränderungen ausgerichtet	Likert	Siebenstufig

Tabelle 3.4: Indikatoren zur Operationalisierung agiler Methoden

3.2.3.1. Mitarbeiter Agilität

„Mitarbeiter Agilität“ stellt ein komplexes Gebiet für die Messung dar und wird infolgedessen durch sieben Indikatoren operationalisiert.

Der erste Indikator bezieht sich auf die Motivation der Projektmitarbeiter und leitet sich direkt aus dem Grundsatz „Build projects around motivated individuals“ des agilen Manifestes (Beck et al., 2001) ab. Für den zweiten Indikator wird die Aufgeschlossenheit der Projektmitarbeiter hinsichtlich Veränderungen erfasst. Der Projektmanager wird dazu aufgefordert zu beurteilen, in welchem Ausmaß die negativ formulierte Aussage „Die Projektmitglieder lehnten Veränderungen ab“, zutrifft. Offenheit und Aufnahmebereitschaft für Änderungen ist ein wichtiges Grundprinzip agiler Methoden (Beck et al., 2001) und wird direkt anhand der Einstellung der Projektmitarbeiter und somit des Teams betrachtet (Smith & Sidky, 2009, S. 13).

Der dritte und vierte Indikator beschäftigen sich sowohl mit der Erfahrung der Mitarbeiter, als auch mit den benötigten Kenntnissen für das spezifische Projekt. Es wird damit nicht nur die allgemeine Eignung der Mitarbeiter für das ausgewählte Projekt betrachtet, sondern zusätzlich

darauf eingegangen, dass für die Anwendung agiler Vorgehensweisen besonders erfahrene Mitarbeiter bevorzugt werden sollen (Shore & Warden, 2008, S. 363f.).

Mithilfe des fünften Indikators wird die Komplexität des Wissensaustausches zwischen den Projektmitarbeitern gemessen. Das agile Manifest fordert eine tägliche Zusammenarbeit der Entscheidungsträger mit den restlichen Projektmitgliedern (Beck et al., 2001). Der hier abgefragte Parameter relativiert dies für komplexe Projekte, indem nur überprüft wird ob der Informationsaustausch schnell und einfach möglich war und keine bestimmten Intervalle nötig sind.

Die räumliche Verteilung der Projektmitarbeiter wird über den sechsten Indikator ermittelt. Agile Methoden bevorzugen eine direkte Kommunikation, welche durch eine örtliche Zusammenfassung des Teams ermöglicht wird (Koch, 2005, S. 73-81).

Als letzter Parameter wird das Arbeitsklima innerhalb des Projektteams betrachtet, womit ein Gesamteindruck über die Motivation sowie die Arbeitsweise gesammelt wird.

Alle Informationen für die Agilität des Projektteams werden mithilfe einer siebenstufigen Likert-Skala gesammelt. Bei der Formulierung der Aussagen wurde darauf geachtet, die von Floyd J. Fowler, Jr. (Fowler, 1995) erarbeiteten Regeln zur Fragenkonstruktion einzuhalten um eine möglichst hohe Aussagekraft der Antworten zu erhalten.

3.2.3.2. Fokus auf Kundennutzen

Der Kunde steht im Zentrum der agilen Vorgehensweise. Ziel aller angewandten Verhaltensmuster soll es sein, maximalen Nutzen für den Kunden zu generieren. Die Prinzipien des agilen Manifests (Beck et al., 2001) konzentrieren sich auf das Erreichen des bestmöglichen Ergebnisses für den Kunden.

Als erster Indikator wird die Integration des Kunden bzw. Endnutzers in die Produktentwicklung herangezogen. Bei agil durchgeführten Projekten soll es Ziel sein, den Kontakt mit dem Auftraggeber über die gesamte Projektlaufzeit aufrecht zu erhalten (Gunasekaran, 2001). In direktem Zusammenhang mit dem zweiten Indikator steht das durchgehende Feedback des Kunden oder Endnutzers zum Projektfortschritt und dient somit nicht nur zur Messung der Interaktion, sondern auch zur Validierung der Kundenintegration. Durch ständiges Feedback soll gewährleistet werden, dass das erzeugte Produkt den tatsächlichen Vorstellungen des Kunden entspricht (Gunasekaran, 2001, S. 118ff.)

Der dritte Indikator untersucht, ob die geforderten Spezifikationen vom Auftraggeber priorisiert wurden. Die Priorisierung konzentriert sich in ihrer Gesamtheit auf den Nutzen des Kunden. In traditionellen Projekten wird die Reihenfolge der benötigten Anforderungen und Tätigkeiten meist durch das Projektteam zu ihren Gunsten festgelegt. Dies kann im letzten Stadium des Projektes dazu führen, dass Features, die für den Kunden essentiell gewesen wären, aus zeitlichen Gründen nicht mehr umgesetzt werden (Cohn, 2006, S. 16f.). Durch eine Priorisierung der Anforderungen durch den Kunden wird gewährleistet, dass die für ihn wichtigen Produktspezifikationen erfüllt sind und er bei der Verwendung von Prototypen, das für ihn jeweils interessanteste Ergebnis erhält.

3.2.3.3. Zyklische Entwicklung

Mithilfe des ersten Indikators wird festgestellt ob Zwischenprodukte vor der Fertigstellung des Endproduktes vorhanden waren. Der zweite Indikator gibt Information darüber, ob die gegebenenfalls vorhandenen Zwischenprodukte dem Kunden oder Endnutzer für eine Bewertung zur Verfügung gestellt wurden. Diese beiden Indikatoren leiten sich aus dem Ziel der frühen und durchgehenden Erstellung von Projektergebnissen an den Kunden ab (Beck et al., 2001).

Als dritter Indikator wird direkt abgefragt, ob die Entwicklung zyklisch bzw. iterativ durchgeführt wurde. Der Einsatz dieser Methodik fördert nicht nur die Entwicklung von Zwischenprodukten für Kundenfeedback, sondern hilft fehlende oder falsche Anforderungen frühzeitig festzustellen (Larman, 2003). Der vierte und letzte Indikator bezieht sich auf die Verwendung des Kundenfeedbacks zu vorhandenen Prototypen und dient in der Praxis den maximalen Wert für den Kunden zu erzeugen (Larman, 2003).

3.2.3.4. Flexibilität im Prozess

Die Flexibilität im Prozess wird als Multiple-Item Konstrukt über drei Indikatoren gemessen. Diese beziehen sich auf die Möglichkeiten, die nötigen Prozesse während der Durchführung zu ändern und somit eine Verbesserung des Projektes herbeizuführen.

Mithilfe der Aussage, ob der gesamte Projektablauf bereits vor Beginn des Entwicklungsprojektes detailliert geplant wurde, lässt sich bestimmen, ob detailliert festgelegten Ablauf gefolgt wurde oder anfangs nur eine grobe Planung stattgefunden hat. Da eine ausführliche Planung zu Projektbeginn bei Anforderungsänderungen immer überarbeitet werden muss, minimiert die Verteilung der Planung über den gesamten Projektzeitraum den Aufwand, indem nur dann Arbeitsabläufe festgelegt werden, wenn diese tatsächlich bevorstehen. Im Gegensatz zur traditionellen Projektdurchführung, soll beim Einsatz agiler Abläufe die durchgehende Projektplanung, anhand der Fortschritte und Anforderungen des jeweiligen Stadiums, die Risiken eines Misserfolges minimieren und die tatsächlich existierenden Zwischenstände messen (Cohn, 2006).

Zur Messung des zweiten Indikators für die Operationalisierung der Flexibilität im Prozess, muss der befragte Projektmanager einschätzen, wie sehr aktiv nach Verbesserungsmöglichkeiten des Entwicklungsprozesses während der Projektdurchführung gesucht wurde. Während der Entwicklung soll nicht nur eine aktive Messung der wichtigen Projektparameter (Gunasekaran, 2001, S. 119ff.), sondern eine Reflexion und Verbesserung durchgeführt werden (Beck et al., 2001).

Der dritte Indikator konzentriert sich auf die Handlungsfreiheit der Projektmitarbeiter, eigenständig Entscheidungen und Änderungen herbeizuführen. Ziel einer agilen Entwicklung sollte es sein, den Mitarbeitern soviel Freiraum zur Verbesserung der nötigen Prozesse wie möglich zu geben (Beck et al., 2001). Diese Entscheidungsfreiheit soll sowohl dem zu erzeugenden Produkt und den Prozessen als auch der Motivation der Mitarbeiter zugute kommen (Koch, 2005, S. 62).

3.2.3.5. Flexibilität im Endprodukt

Das Konstrukt der „Flexibilität im Endprodukt“ basiert auf dem erzeugten Endprodukt des Entwicklungsprozesses, sowie spezifischen Entwicklungsmethoden, welche es ermöglichen sollen, genau jenes Produkt zu erzeugen, das vom Kunden gefordert wurde. In der Literatur sind viele der hier beitragenden Konzepte oftmals auch unter dem Stichwort „lean“ (Leach, 2005) wiederzufinden. Es soll das bestmögliche Ergebnis mit dem minimal nötigen Aufwand erzeugt werden.

Als erster Indikator wird untersucht, ob alles so ausführlich wie möglich dokumentiert wurde. Agile Methoden legen das Hauptaugenmerk auf ein zufriedenstellendes Produkt und wollen die Dokumentation auf das Nötigste beschränken (Koch, 2005, S. 99-104). Durch die – für agile Methoden – negative Formulierung dieser Aussage soll festgestellt werden, ob umfangreiche Dokumentation für den Projektmanager ein Zeichen für die Schaffung von Wert ist.

Der zweite Parameter prüft, ob das Endprodukt Funktionen enthält, die nicht explizit vom Kunden gefordert wurden (siehe Abbildung 3.9). Im Idealfall sollten nur die in den Produkthanforderungen festgehaltenen Spezifikationen erfüllt und keine Zusatzaufgaben verwirklicht worden sein.

Die Wiederverwendung bereits vorhandener Technologien, entwickelter Module und Komponenten wird anhand des dritten Indikators gemessen. Ziel einer Produktentwicklung soll es nicht sein, bereits vorhandenes Wissen neu zu entdecken, sondern dieses sinnvoll für den Projekterfolg einzusetzen.

Die Fragestellung des vierten Indikators steht in Zusammenhang mit dem zweiten Indikator und soll untersuchen, ob das Produkt so einfach als möglich aufgebaut wurde. Abbildung 3.9 zeigt die Ergebnisse der Erhebung, wobei ein Großteil der Produkte nicht geforderte Zusatzfunktionen enthielt und so einfach als möglich aufgebaut war. Die Zusatzfunktionen können in diesem Sinne der Simplizität des Endproduktes gedient haben, indem sie die Erstellung explizit geforderter Anforderungen erleichterten. Eine unnötige Komplizierung des Endproduktes kann dabei ebenfalls ohne die Erstellung von Zusatzfunktionen zustande gekommen sein.

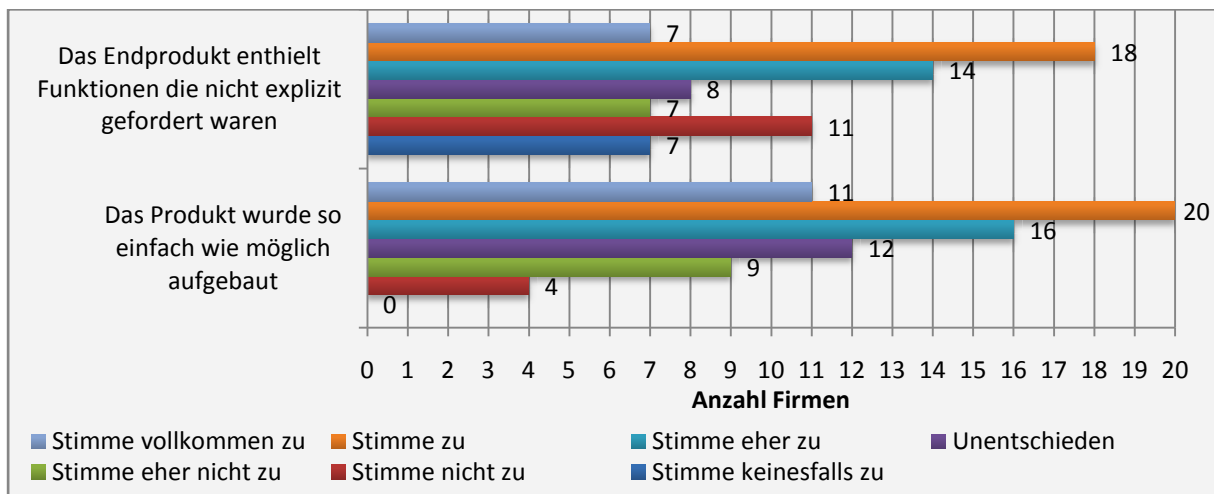


Abbildung 3.9: Simplizität der Ausführung des Endprodukts

Der fünfte Indikator untersucht in welcher Reihenfolge die Entwicklung der Hard- und Software stattgefunden hat.

Ob das Produkt modular aufgebaut wurde, wird mit dem sechsten Parameter untersucht. Ein modularer Aufbau soll nicht nur dabei helfen, bei Anforderungsänderungen durch Austausch oder Hinzufügen neuer Module die gewünschte Funktionalität zu erreichen (Gunasekaran, 2001, S. 732), sondern gleichzeitig eine einfache Verteilung der Zuständigkeiten über die Aufteilung der Modulentwicklung innerhalb eines Projektteams zu ermöglichen (Shore & Warden, 2008, S. 194).

Als siebter und letzter Indikator für die Operationalisierung wird die Architektur des Endproduktes in ihrer Gesamtheit für zukünftige Änderungen betrachtet. Es wird danach gefragt, wie sehr das Produkt darauf ausgelegt wurde, sich notwendigen Änderungen anzupassen.

3.2.3.6. Managementmodell

Die Teilnehmer der Studie wurden nach der Anwendung eines bestimmten Projektmanagementmodells befragt, wobei eine Mehrfachauswahl möglich war. Die zur Auswahl gegebenen Modelle, sowie deren Ausprägung unter den Befragten ist Abbildung 3.10 dargestellt.

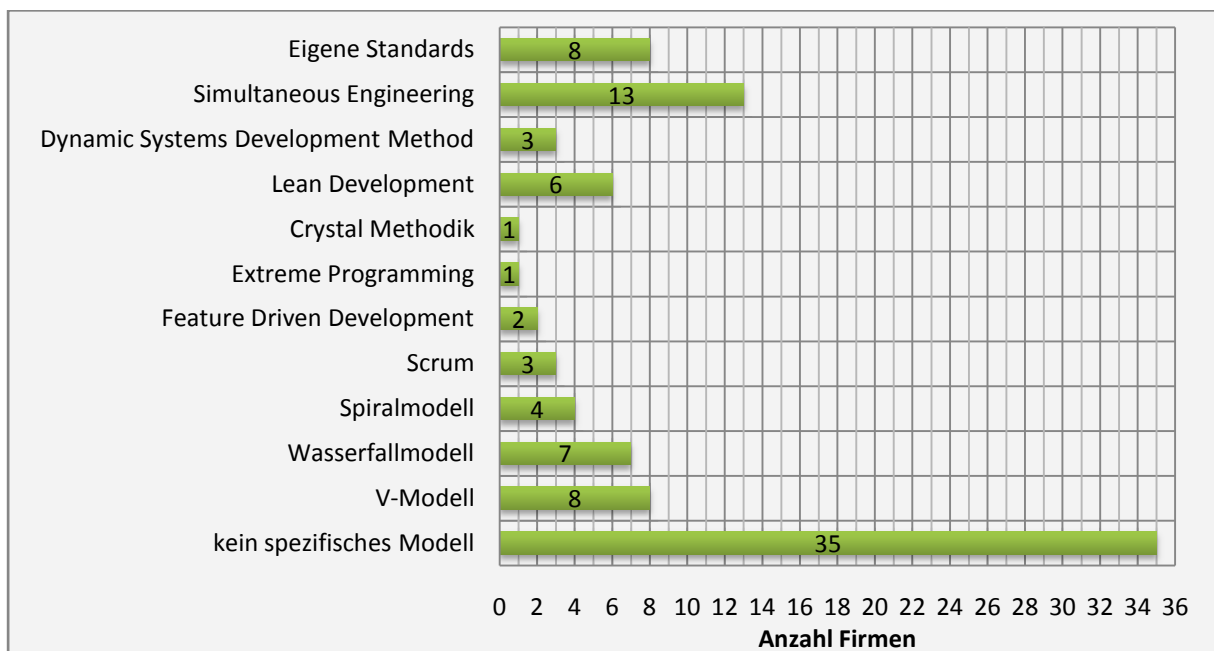


Abbildung 3.10: Erhobene Verteilung von Projektmanagementmodellen

3.3. Datenauswertung

Im folgenden Abschnitt wird die Auswertung der erhobenen Daten erläutert. Kapitel 3.3.1 beschäftigt sich mit der Extraktion einzelner Einflussfaktoren aus den in Kapitel 3.2.3 definierten agilen Methoden. Es werden dabei verschiedene Vorgehensweisen zu einem Konstrukt für die Auswertung zusammengefasst, welche durch die Datenerhebung nachweislich gemeinsam in der Projektdurchführung auftreten. Die einzelnen Konstrukte dienen der Verfeinerung der eingangs formulierten Hypothese, dass sich der Einsatz agiler Methoden bei Änderungen der Anforderungen positiv auf den Projekterfolg auswirkt.

Abschnitt 3.3.2 beschäftigt sich mit der Überprüfung der erarbeiteten Erklärungsmodelle unter Zuhilfenahme der Regressionsanalyse.

Die für die Auswertung verwendete Bezeichnung und zugehörige Fragestellung der einzelnen Variablen ist im Anhang dieser Arbeit zu finden. Negativ formulierte Aussagen sind explizit gekennzeichnet und die Ergebnisse wurden für die Analyse invertiert.

3.3.1. Explorative Faktorenanalyse

Die explorative Faktorenanalyse bildet die Grundlage für die Auswertung des komplexen Konstrukts agiler Methoden. Es wird dabei untersucht, ob sich die Kovarianz der erfassten empirischen Daten durch eine geringere Anzahl unbeobachteter gemeinsamer Faktoren oder latenter Variablen erklären lässt (Landau & Everitt, 2004, S. 277).

Die Faktorenanalyse lässt sich in zwei grundlegende Abschnitte unterteilen, wobei in der ersten Phase eine allgemeine Beschreibung der Faktoren und ihrer Korrelationen durchgeführt wird. Im zweiten Schritt findet eine Vereinfachung der erhaltenen Ergebnisse durch Faktorenrotation statt, wodurch die Interpretierbarkeit der Ergebnisse erleichtert werden soll.

Für den Einsatz der Faktorenanalyse setzen Hair et al. (2009, S. 101) eine fünffache höhere Samplegröße als die Anzahl der zu analysierenden Variablen voraus, wobei das absolute Minimum bei 50 verwertbaren Antworten liegt. Zur Auswertung der vierundzwanzig für die agilen Methoden operationalisierten Werte, wären somit 120 vollständig ausgefüllte Fragebögen nötig um eine eindeutige Analyse zu ermöglichen. Die Faktorenanalyse wird trotz der zu kleinen Auswertungsmenge durchgeführt und bei der Interpretation der Ergebnisse auf diese Diskrepanz Rücksicht genommen.

Um die Anwendbarkeit der Faktorenanalyse zu überprüfen, wird das in der Literatur (Backhaus et al., 1996, S. 206) als empfehlenswertes „measure of sampling adequacy“ (MSA) Testkriterium, auch als Kaiser-Meyer-Olkin (KMO) Kriterium bekannt, auf die gesammelte Datenmenge angewandt. Die resultierenden Werte liegen in einem Bereich von 0 bis 1, wobei Ergebnisse von unter 0,5 als ungeeignet für die Analyse anzusehen sind. Für die vierundzwanzig Variablen der agilen Methoden kann der Gesamtwert von 0,653 berechnet werden, womit sich die Daten als „mittelmäßig“ geeignet für die Analyse klassifizieren lassen. Diese Analyse kann auch für den Ausschluss weniger gut geeigneter Variablen in der Analysemenge herangezogen werden, wobei sich ein sukzessiver Ausschluss anhand weiterer Prüfkriterien empfiehlt. Im Zuge der Auswertung werden einzelne Variablen anhand ihrer Korrelationsfaktoren aus der Analyse entfernt.

Bei Einsatz der Faktorenanalyse stellen die extrahierten Faktoren Linearkombination der zugrundeliegenden empirischen Daten dar. Abbildung 3.11 illustriert ein sehr vereinfachtes Modell der angewandten Faktorenextraktion. X_1 wird durch die gewichtete Summe des Faktors F und dem einzigartigen Faktor U_1 gebildet, in diesem Sinne setzt sich die Variable X_2 aus F und dem einmaligen Faktor U_2 zusammen (Kim & Mueller, 1978, S. 12f.).

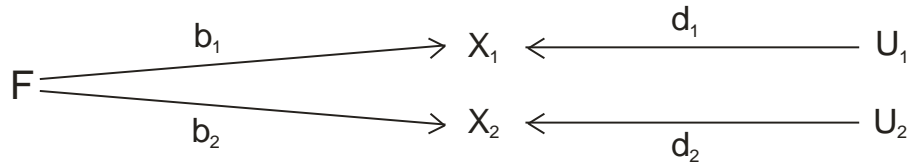


Abbildung 3.11: Schematische Darstellung einer Faktorenabbildung mit zwei zugrundeliegenden Variablen (Kim & Mueller, 1978, S. 13)

Für eine offene Anzahl an Variablen und Faktoren lässt sich die Faktorenanalyse folgendermaßen verdeutlichen:

$$\begin{aligned}
 x_1 &= \lambda_{11}f_1 + \lambda_{12}f_2 \dots + \lambda_{1k}f_k + u_1 \\
 x_2 &= \lambda_{21}f_1 + \lambda_{22}f_2 \dots + \lambda_{2k}f_k + u_2 \\
 x_q &= \lambda_{q1}f_1 + \lambda_{q2}f_2 \dots + \lambda_{qk}f_k + u_q
 \end{aligned}$$

Formel 3.8: Gleichungsmodell der Faktorenanalyse (Landau & Everitt, 2004, S. 278)

Die abgebildeten Gleichungen lassen sich in folgender Formel zusammenfassen:

$$x = \Lambda f + u$$

Formel 3.9: Faktorenextraktion (Landau & Everitt, 2004, S. 278)

X stellt dabei die Matrix der beobachteten Variablen dar, welche sich aus den gewichteten Faktoren (λf) sowie den einmaligen Einflussfaktoren (u) zusammensetzen.

Da die Faktoren nicht direkt beobachtet werden können, wird ihre Ladung anhand der Kovarianzmatrix bestimmt. Es wird hier bei der Berechnung von einer orthogonalen Relation ausgegangen, welche die Eigenwerte der Matrix der erhobenen Variablen für die Auswertung heranzieht und die Ladungsquadrate für die korrelierenden Messvariablen bestimmt.

Die Faktorenextraktion wird mithilfe der Hauptkomponentenanalyse durchgeführt, bei der davon ausgegangen wird, dass sich die Varianz einer Ausgangsvariablen vollständig durch die Faktoren erklären lässt. Es wird dabei eine standardisierte Gesamtvarianz jeder Variablen von 1 angenommen, welche sich aus einer durch die Faktoren erklärbaren Varianz – der Kommunalität –, und einem Restwert zusammensetzt. Abbildung 3.12 zeigt die erklärte Varianz einer Variablen mit 0,8 bei zwei extrahierbaren Faktoren und einem Restwert von 0,2.

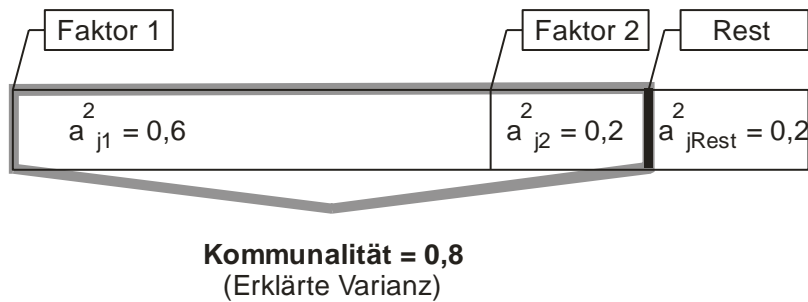


Abbildung 3.12: Kommunalität mit Restwert bei 2 zugrundeliegenden Faktoren nach Backhaus (1996, S. 221)

Da sich nicht alle Werte eindeutig durch die extrahierten Faktoren erklären lassen, wird bewusst ein Informationsverlust in Kauf genommen. Die Anwendung dieses Verfahrens ist nötig um ohne Vorkenntnisse über die Korrelationen eine Aussage über die extrahierten Faktoren treffen zu können.

Die Berechnung der Kommunalitäten erfolgt im Zuge der ersten Faktorenanalyse, die dazu dienen soll die Gesamtmenge der erhobenen Variablen auf jene zu minimieren, welche über die größte Aussagekraft verfügen. Mithilfe des Kaiser-Kriteriums werden die zugrundeliegenden Faktoren extrahiert, wodurch die Anzahl der zu extrahierenden Faktoren gleich der Zahl der Faktoren mit Eigenwerten größer eins festgelegt ist. Der Eigenwert eines jeden Faktors wird als Summe der quadrierten Faktorladungen über aller Variablen bestimmt (Backhaus et al., 1996, S. 226). Die Vorgehensweise des Kaiser-Kriteriums beruht darauf, dass die Ladung einer einzelnen Variablen bereits eins beträgt und somit Faktoren mit einer Ladung unter eins eine geringere Aussagekraft als die anfänglichen Werte besitzen.

3.3.1.1. Anfängliche Faktorenlösung

Für die vierundzwanzig Eingangsvariablen können durch die erste Faktorenanalyse insgesamt sieben Faktoren mit einem Eigenwert größer eins extrahiert werden (siehe Tabelle 3.5 bzw. Anhang für die vollständigen Werte).

Komponente	Anfängliche Eigenwerte			Rotierte Summe der quadrierten Ladungen		
	Gesamt	% der Varianz	Kumulierte %	Gesamt	% der Varianz	Kumulierte %
1	4,281	17,839	17,839	3,011	12,546	12,546
2	4,035	16,813	34,652	2,708	11,283	23,829
3	1,928	8,034	42,685	2,506	10,441	34,271
4	1,674	6,973	49,659	2,378	9,908	44,179
5	1,484	6,182	55,841	2,178	9,074	53,253
6	1,374	5,726	61,567	1,667	6,945	60,197
7	1,167	4,864	66,431	1,496	6,234	66,431
8	,983	4,098	70,529			

Tabelle 3.5: Eigenwerte der extrahierten Faktoren

Bevor eine Interpretation der bestimmten Faktoren durchgeführt werden kann, werden jene Variablen mit einer zu geringen Faktorladung bzw. Gesamtvarianz aus der Analyse entfernt um ein besseres Endergebnis zu erhalten. In Anlehnung an Backhaus (1996, S. 229) werden

3. Empirische Analyse - Datenauswertung

Variablen mit einem maximalen Ladungswert oder Kommunalität unter 0,5 aus der Analysemenge eliminiert. Das Ergebnis der anfänglichen Faktorenanalyse mit den Ladungswerten jeder Variable auf die zuvor extrahierten Faktoren, sowie die jeweiligen Kommunalitäten sind in Tabelle 3.6 abgebildet, wobei nur Ladungswerte über 0,4 ausgegeben wurden.

Die abgebildeten Faktorenladungen wurden durch Varimax-Rotation leichter interpretierbar gemacht. Es handelt sich dabei um eine rechtwinkelige Drehung der Faktorladungsrichtungen, bei der die Aussagekraft der gefundenen Faktoren unverändert bleibt. (Backhaus et al., 1996, S. 229f.)

	Komponente							Kommunalitäten
	1	2	3	4	5	6	7	
AM3_ZwProd	,903							,847
AM3_ZwProdKunde	,896							,875
AM3_VerwFeedb	,853							,791
AM1_Klima		,793						,762
AM1_Motivation		,746						,677
AM1_Information		,701						,710
AM1_Raeumlich		,502					-,475	,588
AM1_Erfahrung								,612
AM4_DetPlanung			-,753					,656
AM5_AusfDoku			-,637					,507
AM5_Refactoring			,631					,456
AM4_SucheVerbes			,630					,648
AM4_FixProzess			-,456					,481
AM2_Feedback				,809				,815
AM2_Integration				,695				,682
AM2_Prioritaet				,606				,435
AM3_Iterativ	,472			,496				,683
AM5_Modular					,814			,739
AM5_ArchVeraend					,775			,753
AM5_HWvorSW					,500			,590
AM5_EinfachProd								,506
AM5_Zusatzfunkt						,865		,755
AM1_Aenderungen							,846	,764
AM1_Kenntnisse							,569	,611

Tabelle 3.6: Variablen mit Faktorenladung und Kommunalitäten der erste Faktorenanalyse; Ladungswerte unter 0,4 wurden von der Anzeige ausgeschlossen und die Variablen wurden absteigend ihrer Faktorladung angeordnet

Die Variable AM1_Erfahrung verfügt über keine eindeutige Faktorladung über dem Schwellwert von 0,4 und wird deshalb vor Beginn der nächsten Faktorenanalyse entfernt. AM5_Refactoring besitzt eine erklärte Varianz von 0,456 wodurch sich der erfasste Wert nicht eindeutig auf den Einfluss eines oder mehrerer Faktoren zurückführen lässt und somit für die Interpretation keine deutlichen Ergebnisse liefert.

Für die Variable AM1_Raeumlich liegt der Ladungswert auf Faktor 2 mit 0,502 über 0,5 im Vergleich zu -0,475 auf Faktor 6. Die Variable lässt sich somit Faktor 2 zuweisen und wird für die nächste Analyse beibehalten.

Eine weitere problematische Variable für die Analyse stellt AM4_FixProzess dar. Sowohl die Ladung auf Faktor 3 liegt mit einem Wert von -0,456 unterhalb der signifikanten Grenze, als auch die erklärbare Varianz von 0,481 lässt keine eindeutigen Aussagen über den zu erwartenden Wert zu. AM3_Prioritaet wird ebenfalls aus der Datenmenge entfernt, da auch hier die berechenbare Kommunalität mit 0,435 nicht den Anforderungen entspricht.

Die Variable AM3_Iterativ verfügt über zwei Ladungswerte auf Faktor 1 und Faktor 4 unter dem Grenzwert von 0,5, wodurch sie sich keinem Faktor eindeutig zuordnen lässt und wird deshalb ebenfalls eliminiert.

Als letzte Variable wird AM5_EinfachProd, welche sich auf die Simplizität des Endproduktes bezieht, aus der Datenmenge entfernt.

In Summe werden durch diese Voranalyse folgende sechs Variablen aus der Menge entfernt: AM1_Erfahrung, AM5_Refactoring, AM4_FixProzess, AM3_Prioritaet, AM3_Iterativ, AM5_EinfachProd.

3.3.1.2. Finale Faktorenanalyse

Für die finale Extraktion der Faktoren, wird nach dem Entfernen der sechs unzureichend aussagekräftigen Werten, erneut eine Faktorenanalyse mit den verbleibenden achtzehn Variablen durchgeführt. Um ein absolut eindeutiges Ergebnis für die Auswertung der achtzehn Variablen zu erhalten, wäre abermals die fünffache Samplegröße der Eingangswerte, von somit 90 ausgefüllten Fragebögen, notwendig. Obwohl diese Forderung nicht vollständig erfüllt werden konnte, führt die finale Faktorenanalyse zu einem zufriedenstellenden Ergebnis, welches im Anschluss erläutert wird.

Es wird erneut unter Einsatz der Hauptkomponentenanalyse und Zuhilfenahme des Kaiser-Kriteriums eine Auswertung durchgeführt. Um die erhaltenen Faktorladungswerte leichter interpretierbar zu gestalten, werden die extrahierten Werte durch Verwendung der Varimax Rotation dargestellt.

Insgesamt können sieben Faktoren (siehe Tabelle 3.7) extrahiert werden, bei denen die Eigenwerte über eins, als auch die Kommunalitäten aller Variablen über dem Schwellwert von 0,5 liegen. Mit Ausnahme von AM1_Raeumlich laden alle Variablen mit einem Wert von über 0,5 eindeutig auf einen einzigen Faktor. Da der Faktorladungswert von AM1_Raeumlich auf Faktor 2 nur 0,405 beträgt, wird die Variable Faktor 6 zugewiesen.

	Komponente							Kommunalitäten
	1	2	3	4	5	6	7	
AM3_ZwProd	,914							,872
AM3_ZwProdKunde	,905							,886
AM3_VerwFeedb	,835							,787
AM1_Klima		,841						,784
AM1_Motivation		,805						,722
AM1_Information		,713						,719
AM5_Modular			,820					,785
AM5_ArchVeraend			,766					,761
AM5_HWvorSW			,608					,625
AM2_Feedback				,873				,854
AM2_Integration				,817				,805
AM4_SucheVerbes					-,757			,765
AM4_DetPlanung					,745			,647
AM5_AusfDoku					,621			,762
AM5_Zusatzfunkt						-,848		,755
AM1_Raeumlich		,405				,538		,580
AM1_Aenderungen							,902	,855
AM1_Kenntnisse							,523	,665

Tabelle 3.7: Ergebnisse der zweiten Faktorenanalyse; Ladungswerte unter 0,4 wurden von der Anzeige ausgeschlossen und die Variablen wurden absteigend ihrer Faktorladung angeordnet

Um die Zuverlässigkeit der Zusammenhänge unter den einzelnen Variablen der extrahierten Faktoren für die Messung zu untersuchen, wird „Cronbach's Alpha“ herangezogen. Durch Anwendung dieses Maßes auf alle Variablen eines Faktors kann eine Aussage darüber getroffen werden, ob diese tatsächlich alle auf denselben zugrundeliegenden Faktor schließen lassen. Für die Berechnung (siehe Formel 3.10) wird die quadrierte Anzahl aller einem Faktor zugehörigen Variablen (N^2) mit der durchschnittlichen Kovarianz multipliziert (\overline{Cov}) und durch die Summe der Varianzen einer Variablen ($\sum s_{item}^2$) sowie die Summe der Kovarianzen dieser Variablen ($\sum Cov_{item}Cov_{item}$) mit allen restlichen berechnet. (Field, 2005, S. 667f.)

$$\alpha = \frac{N^2 \overline{Cov}}{\sum s_{item}^2 + \sum Cov_{item}}$$

Formel 3.10: Cronbach's Alpha (Field, 2005, S. 667)

Der erhaltene Alpha Wert kann als Korrelationskoeffizient interpretiert werden, der eine Aussage über den linearen Zusammenhang der in die Gleichung eingebrachten Variablen trifft. Bei der Bewertung einer Skala anhand des α Wertes ist die Anzahl der zu testenden Variablen zu berücksichtigen. Da α abhängig von der Anzahl der zu betrachtenden Variablen ansteigt, d.h. für wenige gut korrelierende Variablen kann ein schlechterer Alpha Wert, als für viele schlecht korrelierende Variablen berechnet werden. Alpha sollte daher nicht ohne zusätzliche Kontrolle der einzelnen Korrelationen eingesetzt werden. (Field, 2005, S. 668f.)

3. Empirische Analyse - Datenauswertung

Für die gesammelten Items der einzelnen Faktoren können folgende Alpha Werte berechnet werden:

Faktor	1	2	3	4	5	6	7
Anzahl Items	3	3	3	2	3	2	2
Cronbach's Alpha	0,907	0,785	0,631	0,842	-0,108	-0,637	0,575

Tabelle 3.8: Cronbach's Alpha für 7 Faktoren

Allgemein werden Alpha Werte zwischen 0,7 und 0,9 als optimal für weitere Auswertungen angesehen, wobei abhängig von der Anwendung auch Werte von unter 0,6 noch akzeptabel für eine Weiterverwendung sind (Lehman et al., 2005, S. 145). Die negativen α Werte für Faktor 5 und Faktor 6 lassen sich auf die negative Korrelation der Variablen zurückführen, welche auf eine Multidimensionalität in der Messung des zugrundeliegenden Faktors hinweisen (Howard et al., 2008). Bevor eine abermalige Faktorenanalyse durchgeführt wird, werden die negativ korrelierten Variablen AM4_SucheVerbes und AM5_Zusatzfunkt aus der Grundmenge entfernt. Es bleiben dadurch 16 Variablen für die Faktorenanalyse erhalten, für welche 80 vollständig ausgefüllte Fragebögen nötig wären um ein absolut aussagekräftiges Ergebnis zu erhalten. Da durch die Senkung der Forderung nur mehr eine geringe Diskrepanz zu den tatsächlich vorhandenen 72 auswertbaren Umfrageergebnisse besteht, liefert die Analyse eine genügend gute Aussagekraft um daraus konkrete Modelle ableiten zu können.

Die Ergebnisse der finalen Faktorenanalyse mit der erklärbaren Varianz aller Variablen, sowie den Alpha Werten sind in Tabelle 3.9 abgebildet.

	Komponente						Kommunalität n
	1	2	3	4	5	6	
AM3_ZwProd	,910						,868
AM3_ZwProdKunde	,903						,879
AM3_VerwFeedb	,836						,784
AM1_Information		,769					,727
AM1_Raumlich		,707					,712
AM1_Klima		,694					,772
AM1_Motivation		,598					,694
AM5_Modular			,853				,787
AM5_ArchVeraend			,781				,739
AM5_HWvorSW			,580				,607
AM2_Feedback				,844			,824
AM2_Integration				,806			,794
AM5_AusfDoku					,847		,741
AM4_DetPlanung					,747		,605
AM1_Aenderungen						,848	,794
AM1_Kenntnisse						,691	,664
Anzahl Items	3	4	3	2	2	2	
Cronbach's Alpha	0,907	0,615	0,631	0,842	0,641	0,575	

Tabelle 3.9: Ergebnisse der finalen Faktorenanalyse inklusive Cronbach's Alpha; Ladungswerte unter 0,4 wurden von der Anzeige ausgeschlossen und die Variablen wurden absteigend ihrer Faktorladung angeordnet

Alle sechzehn analysierten Variablen verfügen über eine erklärbare Varianz, sowie positive Faktorladung von über 0,5 und können einem spezifischen Faktor eindeutig zugeordnet werden. Die Ergebnisse der Reliabilitätsanalyse liefern Alpha Werte zwischen 0,907 und 0,575. Der für Faktor 6 berechnete Alpha Wert von 0,575 wird aufgrund der geringen Anzahl der zugeordneten Variablen, sowie der eindeutigen Korrelationen der Variablen, als zufriedenstellend für die Weiterverwendung eingestuft.

Die sechs zugrundeliegenden Faktoren werden im Anschluss genauer klassifiziert und ermöglichen eine Spezifikation des in Kapitel 2.4.4.8 allgemein definierten Erklärungsmodells des Einflusses agiler Methoden auf den Projekterfolg bei dem Auftreten von Anforderungsänderungen.

Iterative Entwicklung (F1 IterativeEntwicklung)

Dieser Faktor wird durch die Variablen AM3_ZwProd (Erstellung von Zwischenprodukten), AM3_ZwProdKunde (Bereitstellung der Zwischenprodukte für den Kunden) und AM3_VerwFeedb (Kundenfeedback zu Zwischenprodukten für Verbesserung des Endprodukts) repräsentiert. Das extrahierte Konstrukt ist durch seine Werte stark an das in Kapitel 3.2.3.3 erstellte theoretische Modell angelehnt. Mit Ausnahme der expliziten Fragestellung nach iterativer Entwicklung konnten alle Variablen auf einen gemeinsamen Einflussfaktor zurückgeführt werden.

Hypothese 2a: Je stärker der Faktor der iterativen Entwicklung ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.

Gutes Arbeitsklima des Projektteams (F2 Arbeitsklima)

AM1_Information (Komplexität des Wissensaustauschs), AM1_Raeumlich (Verteilung des Projektteams), AM1_Klima (allgemeines Arbeitsklima im Projektteam) und AM1_Motivation (Motivation der Teammitglieder), welche sich als Arbeitsklima zusammenfassen lassen, basieren Großteils auf dem in Kapitel 3.2.3.1 erstellten Konstrukt der Mitarbeiteragilität. Konkret konnten die agilen Ansätze für die Interaktionen der Mitarbeiter extrahiert werden.

Hypothese 2b: Je stärker der Faktor des guten Arbeitsklimas ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.

Anpassbarkeit des Endprodukts (F3 Endproduktanpassbarkeit)

Der dritte Faktor bringt die Anpassbarkeit des erstellten Produktes zum Ausdruck und wird durch die Variablen AM5_Modular (modularer Aufbau des Endprodukts), AM5_ArchVeraend (Architektur ist offen für Änderungen) und AM5_HWvorSW (Reihenfolge der Hard- und Softwareentwicklung) dargestellt. Die drei korrelierenden Variablen leiten sich aus dem theoretischen Modell in Kapitel 3.2.3.5 ab.

Hypothese 2c: Je stärker der Faktor der Anpassbarkeit des Endproduktes ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.

Integration des Kunden (F4 Kundenintegration)

Der Faktor der Kundenintegration in die Entwicklung wird durch AM2_Feedback (durchgehendes Kundenfeedback) und AM2_Integration repräsentiert, wobei sich dieser aus dem in Kapitel 3.2.3.2 erstellten Modell ableiten lässt. Die Variable AM2_Prioritaet (Priorisierung der Anforderungen durch Kunden) des theoretischen Konstrukts wurde aufgrund der zu geringen Kommunalität aus der Analysemenge entfernt.

Hypothese 2d: Je stärker der Faktor der Integration des Kunden ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.

Effiziente Durchführung (F5 EffizienteAusfuehrung)

Die Effizienz der Projektdurchführung setzt sich aus den Werten AM5_AusfDoku (ausführliche Dokumentation) und AM4_DetPlanung (detaillierte Planung vor Projektbeginn) zusammen. Das erhaltene Konstrukt lässt sich durch eine stärker zielgerichtete Entwicklungsmethode erklären, bei der die zufriedenstellende Fertigstellung des Endproduktes im Vordergrund steht und unnötige Tätigkeiten eliminiert werden (Shore & Warden, 2008, S. 378).

Hypothese 2e: Je stärker der Faktor der effizienten Durchführung ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.

Änderungsbereitschaft des Projektteams (F6 Aenderungsbereitschaft)

AM1_Aenderungen (Aufnahmebereitschaft der Mitarbeiter für Veränderungen) und AM1_Kenntnisse (Fachkenntnisse der Projektmitarbeiter) bilden den sechsten Faktor der Änderungsbereitschaft des Projektteams. Dieses Konstrukt lässt sich dadurch erklären, dass Mitarbeiter offener für Veränderungen sind, wenn sie bereits genug Kenntnisse für eine erfolgreiche Entwicklung besitzen (Subramaniam & Hunt, 2006, S. 31f.).

Hypothese 2f: Je stärker der Faktor der Änderungsbereitschaft des Projektteams ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.

Die Überprüfung der Zusammenhänge für die aufgestellten Erklärungsmodelle erfolgt anschließend in Kapitel 3.3.2 unter Einsatz der Regressionsanalyse.

3.3.2. Regressionsanalyse

Die Regressionsanalyse stellt ein flexibles statistisches Analyseverfahren für die Messung der Zusammenhänge einer messbaren Zielgröße und einer oder mehreren messbaren Einflussgrößen dar. Es wird dabei zwischen „einfacher Regressionsanalyse“, bei welcher die Auswirkungen einer Einflussgröße auf die Zielvariable untersucht werden sollen, und „multipler Regressionsanalyse“ mit mehreren Einflussgrößen auf einen Endwert, unterschieden. (Beucher, 2007, S. 295f.)

Ziel kann es sein, die Ursache eines Messwertes und damit den Einfluss von Variablen auf eine Zielgröße festzustellen (Ursachenanalyse), die Auswirkung von Änderungen der Einflussgrößen auf den Endwert (Wirkungsanalyse) oder die Veränderung der abhängigen Variable über einen Zeitraum zu prognostizieren (Zeitreihenanalyse). (Backhaus et al., 1996, S. 5)

In den folgenden Abschnitten wird die multiple Regressionsanalyse für die Schätzung der zuvor erarbeiteten Erklärungsmodelle unter Anwendung agiler Methoden verwendet, sowie eine einfache lineare Analyse für den Einfluss von Anforderungsänderungen durchgeführt. Eingangs wird die Vorgehensweise für die Auswertung kurz zusammengefasst, bevor in den Kapitel 3.3.2.2 und 3.3.2.4 die erhaltenen Ergebnisse erläutert werden.

3.3.2.1. Einfache lineare Regressionsanalyse

Mithilfe der einfachen linearen Regressionsanalyse wird der Zusammenhang zwischen einer unabhängigen Variablen, dem Regressor und einer abhängigen Variable, dem Regressanden untersucht. Da die allgemeine Formel (siehe Formel 3.11) einer geraden Linie sehr ähnlich ist, wird sie auch als lineare Regressionsanalyse bezeichnet. Es wird der Schätzwert für die abhängige Variable (\hat{Y}), ausgehend von einem Startpunkt (b_0) und des Anstiegs (b_1) der unabhängigen Variablen (X_1), berechnet. Epsilon (ε) repräsentiert die durch die unabhängige Variable X_1 unerklärte mittlere Abweichung des Beobachtungswertes vom Schätzwert und soll minimiert werden, um ein möglichst akkurates Ergebnis zu erhalten. (Backhaus et al., 1996, S. 10ff.)

$$\hat{Y} = b_0 + b_1 X_1 + \varepsilon$$

Formel 3.11: Einfache lineare Regressionsfunktion (Backhaus et al., 1996, S. 14)

Da eine Regressionsanalyse auf eine kleine Stichprobe der Grundgesamtheit durchgeführt wird, muss die Güte der Aussagekraft überprüft werden. Der „wahre“ Zusammenhang wird aufgrund der aus der Stichprobe erhaltenen Resultate überprüft (Backhaus et al., 1996, S. 7).

Es wird dazu eine sogenannte Nullhypothese formuliert, die besagt, dass eine Veränderung der gewählten unabhängigen Variablen keinen Einfluss auf die abhängige Variable hat. Diese These soll nun verworfen werden und damit die Alternativhypothese, dass die unabhängigen Variablen einen Einfluss auf die abhängige Variable haben, belegt werden.

$$H_0: b_1 = 0$$

Formel 3.12: Nullhypotes für einfache lineare Regression (Chattefuee & Hadi, 2006, S. 132)

Die Nullhypothese wird anhand des F-Tests der Alternativhypothesen überprüft, welcher den benötigten erklärbaren Anteil der Varianz für die ausgewerteten Beobachtungen mit Berücksichtigung der Anzahl der betrachteten unabhängigen Variablen berechnet. Der errechnete F-Wert muss einen Wert über einer bestimmten Grenze einnehmen, damit die Alternativhypothese aussagekräftig ist und die Nullhypothese abgelehnt werden kann. (Field, 2005, S. 232f.)

3.3.2.2. Analyseergebnis „Einfluss von Anforderungsänderungen auf den Projekterfolg“

In diesem Abschnitt wird das erste Erklärungsmodell des Einflusses von Anforderungsänderungen auf den Projekterfolg mithilfe einer einfachen linearen Regression untersucht.

Für die Analyse der extrahierten agilen Methode kann der zuvor allgemein definierte Regressionssatz folgendermaßen spezifiziert werden:

$$\text{Erwarteter Projekterfolg} = b_0 + b_1 \text{Anforderungsänderungen}$$

Formel 3.13: Regressionssatz des Erklärungsmodells für den Einfluss von Anforderungsänderungen

In Tabelle 3.15 sind die Auswertungsergebnisse der linearen Regression für die abhängige Variable des Projekterfolgs und die unabhängige Variable „Anforderungsänderungen“ abgebildet.

	Modell 1		
	Beta	T-Wert	Sig.
Anforderungsänderungen	-0,300	-2,636	0,010
R ²	0,090		
F	6,947		

Tabelle 3.10: Regressionsanalyseergebnis Auswirkung von Anforderungsergebnissen auf Projekterfolg

Um über eine allgemeine Aussagekraft zu verfügen, muss der F-Wert des Modells für 72 ausgewertete Fragebögen mit einem Regressor über dem Grenzwert (Field, 2005, S. 757) von 3,96 (für eine Signifikanz von 0,05) bzw. 6,96 (für eine Signifikanz von 0,01) liegen. Da der erreichte F-Wert mit 6,947 nahe genug an dem Minimum liegt wird das Ergebnis als zufriedenstellend für die Erklärung betrachtet.

Der T-Wert des Modells gibt zusätzlich Aufschluss über die statistische Signifikanz, sowie die Richtung des Einflusses der unabhängigen Variablen (Firer & Swartz, 2007) und liegt mit einem absoluten Wert von 2,636 über dem unteren Grenzwert von 1,66 (Field, 2005, S. 755).

Mit einem Wert von 0,010 ist das Erklärungsmodell hoch signifikant, sodass ein Zusammenhang zwischen der Anzahl der Anforderungsänderungen und dem Projekterfolg eindeutig nachgewiesen werden kann. Der Zusammenhang lässt sich somit von der Stichprobe auf die Grundgesamtheit übertragen.

Der Beta Koeffizienten verdeutlicht die Stärke und Richtung der Veränderung der abhängigen Variablen des Projekterfolgs bei Anstieg der Anforderungsänderungen um eine Einheit. Aus der Auswertung des Erklärungsmodelles folgt somit eine Abnahme des Projekterfolgs um 30 Prozent für jede Erhöhung der Anforderungsänderungen.

Das Ergebnis bestätigt die zuvor definierte Hypothese und belegt den allgemeinen negativen Einfluss von Anforderungsänderungen auf den Projekterfolg. Etwaige positive oder negative Einflüsse agiler Methoden wurden bei dieser Auswertung nicht explizit berücksichtigt und werden im Anschluss in Kapitel 3.3.2.4 separat betrachtet.

3.3.2.3. Multiple lineare Regressionsanalyse

Multiple Regressionsanalyse untersucht die mathematischen Zusammenhänge mehrerer unabhängiger Variablen, den Regressoren, mit einer abhängigen Variablen, dem Regressanden. Sie basiert in ihrer Anwendung auf der einfachen linearen Regressionsanalyse, wobei die Komplexität der Berechnung aufgrund der Anzahl unabhängiger Variablen gesteigert wird. Der geschätzte Regressionssatz lautete wie folgt (Backhaus et al., 1996, S. 17):

$$\hat{Y} = b_0 + b_1X_1 + b_2X_2 + \dots + b_jX_j + \dots + b_jX_j$$

Formel 3.14: Allgemeiner Regressionssatz

\hat{Y} repräsentiert den geschätzten Endwert der abhängigen Variablen Y, wobei X die Werte der unabhängigen Variablen für die Analyse darstellen. Mithilfe der Parameter b_0 , welche als konstantes Glied den Startpunkt der Geraden bildet, und der Regressionskoeffizienten b_1 bis b_j , anhand derer die Geradensteigung einberechnet wird, soll der Wert der tatsächlichen Variablen Y geschätzt werden.

Ziel der Regressionsanalyse ist es die Parameter b so zu bestimmen, dass die Summe der nicht erklärten Streuung minimiert wird. Diese Zielfunktion der Berechnung ist in folgender Formel dargestellt (Backhaus et al., 1996, S. 17):

$$\sum_{k=1}^K e_k^2 = \sum_{k=1}^K [y_k - (b_0 + b_1x_{1k} + b_1x_{1k} + \dots + b_jx_{jk} + \dots b_jx_{jk})]^2 \rightarrow \min!$$

Formel 3.15: Zielfunktion der Regressionsanalyse

Der quadrierte, zu minimierende Fehlerwert e wird über die Anzahl der Beobachtungen (K) für die Ausprägung der abhängigen Variable (y) minus deren über den allgemeinen Regressionssatz geschätzten Wert zum Quadrat (vgl. Formel 3.14) berechnet, wobei dieser so klein als möglich gehalten werden soll.

Die Aussagekraft der multiplen Regressionsfunktion wird abermals mithilfe der Nullhypothese überprüft, wobei die Auswirkung aller unabhängigen Variablen auf den Regressanden einbezogen wird. Um die Nullhypothese zu widerlegen, muss bewiesen werden, dass zumindest ein Regressor die abhängige Variable beeinflusst.

3.3.2.4. Analyseergebnisse „Einfluss agiler Methoden bei Anforderungsänderungen auf den Projekterfolg“

Dieser Abschnitt beschäftigt sich mit den Analyseergebnissen der Regressionsmodelle agiler Methoden. Die in Kapitel 3.3.1.2 definierten Hypothesen 2a-2f werden unter Zuhilfenahme der multiplen linearen Regressionsanalyse auf ihre Signifikanz und Aussagekraft überprüft und die Ergebnisse im Detail erklärt.

Für die Analyse der extrahierten agilen Methode kann der zuvor allgemein definierte Regressionssatz folgendermaßen erstellt werden:

$$\text{Erwarteter Projekterfolg} = b_0 + b_1\text{AAE} + b_2\text{AF} + b_3\text{AF} \times \text{AAE}$$

Formel 3.16: Regressionssatz der Erklärungsmodelle agiler Methoden

Der zu erwartende Projekterfolg wird durch die Summe der Anforderungsänderungen (AAE), dem jeweiligen agilen Faktor (AF) und dem Einflusskonstrukt aus agilen Methoden auf die Anforderungsänderungen (AF×AAE) bestimmt.

Die Auswirkung der jeweiligen agilen Methode auf die Anzahl der Anforderungsänderungen wird durch Multiplikation der beiden Regressoren in direkte Beziehung gesetzt und in die Regressionsanalyse integriert. Für die Auswertung des jeweiligen Erklärungsmodells sind somit die Werte des Einflusskonstrukts ausschlaggebend.

In Tabelle 3.11 sind die Ergebnisse der ersten Auswertung für den Einfluss des Faktors „Iterative Entwicklung“ bei Anforderungsänderungen auf den Projekterfolg dargestellt. Die Werte des ersten Modells beziehen sich dabei auf eine Analyse unter Ausschluss des Einflusstems „Anforderungsänderungen × F1 Iterative Entwicklung“ und haben noch keine Bedeutung für die Bestätigung oder Widerlegung des Erklärungsmodells.

Die Resultate von Modell 2 werden für die Beurteilung der Regression genauer betrachtet. Der F-Wert ist mit 2,582 unter dem Grenzwert von 2,74 (Field, 2005, S. 757) für drei unabhängige Variablen, wodurch eine Allgemeingültigkeit nicht explizit bestätigt werden kann.

Der berechnete T-Wert liegt mit einem Wert von 0,037 ebenfalls unter der Signifikanzgrenze, wodurch sich die Hypothese des Einflusses von Faktor 1 „Iterativer Entwicklung“ nicht bestätigen lässt.

	Modell 1			Modell 2		
	Beta	T-Wert	Sig.	Beta	T-Wert	Sig.
Anforderungsänderungen	-0,272	-2,301	0,024	-0,272	-2,278	0,026
F1 Iterative Entwicklung	-0,113	-0,959	0,341	-0,113	-0,942	0,349
Anforderungsänderungen × F1 Iterative Entwicklung				0,004	0,037	0,971
R ²	0,102			0,102		
R ² Änderung				0,000		
F	3,929			2,582		

Tabelle 3.11: Regressionsanalyseergebnis Faktor 1 – Iterative Entwicklung

In Tabelle 3.12 sind die Ergebnisse der Regressionsanalyse für den Einfluss des zweiten Faktors bei Anforderungsänderungen auf den Projekterfolg abgebildet. Modell 2 verfügt mit einem F-Wert von 6,867 über eine gute Allgemeingültigkeit, wodurch die Ergebnisse der Stichprobe auf die Grundgesamtheit übertragbar sind. Aufgrund des zu geringen T-Wertes von 0,054 kann kein positiver Einfluss für das Erklärungsmodell nachgewiesen werden.

	Modell 1			Modell 2		
	Beta	T-Wert	Sig.	Beta	T-Wert	Sig.
Anforderungsänderungen	-0,239	-2,238	0,028	-0,239	-2,110	0,039
F2 Arbeitsklima	0,382	3,575	0,001	0,383	3,539	0,001
Anforderungsänderungen × F2 Arbeitsklima				0,006	0,054	0,957
R ²	0,232			0,233		
R ² Änderung				0,000		
F	10,449			6,867		

Tabelle 3.12: Regressionsanalyseergebnis Faktor 2 – Arbeitsklima

Die Ergebnisse der Regressionsanalyse für den Einfluss des dritten Faktors „Endproduktanpassbarkeit“ bei Anforderungsänderungen auf den Projekterfolg sind in Tabelle 3.13 ersichtlich. Mit einem F-Wert von 6,700 kann auf eine Allgemeingültigkeit der Auswertung geschlossen werden. Der Einfluss des Interaktionsterms ist signifikant und weist durch den Beta-Koeffizienten auf eine Steigerung des Projekterfolgs um 18,7 Prozent hin.

	Modell 1			Modell 2		
	Beta	T-Wert	Sig.	Beta	T-Wert	Sig.
Anforderungsänderungen	-0,240	-2,188	0,032	-0,205	-1,855	0,068
F3 Endproduktanpassbarkeit	0,331	3,020	0,004	0,293	2,650	0,010
Anforderungsänderungen × F3 Endproduktanpassbarkeit				0,187	1,671	0,099
R ²	0,196			0,228		
R ² Änderung				0,032		
F	8,436			6,700		

Tabelle 3.13: Regressionsanalyseergebnis Faktor 3 – Endproduktanpassbarkeit

Tabelle 3.15 bildet die Ergebnisse der Analyse für den Einfluss des vierten Faktors „Kundenintegration“ bei Anforderungsänderungen auf den Projekterfolg ab. Das Modell kann anhand des F-Wertes von 2,868 als allgemeingültig eingestuft werden, wobei kein signifikanter Einfluss für den Interaktionsterm „Anforderungsänderungen × F4 Kundenintegration“ nachgewiesen werden kann.

Aufgrund des Ergebnisses für den Einfluss des Faktors der Kundenintegration, wird die Hypothese 2d folgendermaßen verfeinert:

Hypothese 2di: Je stärker der Faktor der Integration des Kunden ausgeprägt ist, desto größer ist die Kundenzufriedenheit.

Mithilfe einer linearen Regressionsanalyse wird der Einfluss des Faktors „Kundenintegration“ auf das Erfolgskriterium „Zufriedenstellung des Kunden“, bestehend aus der Wichtigkeit und dem Erfüllungsgrad der Kundenzufriedenstellung allgemein für Projekte überprüft.

3. Empirische Analyse - Datenauswertung

	Modell 1		
	Beta	T-Wert	Sig.
Kundenzufriedenstellung	0,249	2,155	0,035
R ²	0,062		
F	4,645		

Tabelle 3.14: Regressionsanalyse – Einfluss der Kundenintegration auf Kundenzufriedenheit

Tabelle 3.14 zeigt die Ergebnisse der Analyse, welche mit einem F-Wert von 4,645 über dem Grenzwert von 3,96 (Field, 2005, S. 757), sowie dem T-Wert von 2,155 ebenfalls über der unteren Grenze von 1,66 (Field, 2005, S. 755) liegt und somit als allgemeingültig eingestuft werden kann. Aus der Auswertung geht hervor, dass die Kundenintegration das Erfolgskriterium der „Zufriedenheit des Kunden“, unabhängig von Anforderungsänderungen, moderat signifikant positiv beeinflusst.

	Modell 1			Modell 2		
	Beta	T-Wert	Sig.	Beta	T-Wert	Sig.
Anforderungsänderungen	-0,320	-2,786	0,007	-0,317	-2,750	0,008
F4 Kundenintegration	0,140	1,218	0,227	0,134	1,156	0,252
Anforderungsänderungen × F4 Kundenintegration				0,054	0,471	0,639
R ²	0,109			0,112		
R ² Änderung				0,073		
F	4,239			2,868		

Tabelle 3.15: Regressionsanalyseergebnis Faktor 4 – Kundenintegration

Die Ergebnisse der Auswertung für den Einfluss des fünften Faktors „Effiziente Durchführung“ bei Anforderungsänderungen auf den Projekterfolg ist in Tabelle 3.16 dargestellt. Mit einem F-Wert von 5,334 ist das Modell allgemeingültig und das Konstrukt „Anforderungsänderungen × F5 Effiziente Durchführung“ anhand des T-Wertes von 1,945 moderat signifikant. Der Beta-Koeffizient lässt auf eine positive Veränderung des Projekterfolgs von 21,4 Prozent schließen.

	Modell 1			Modell 2		
	Beta	T-Wert	Sig.	Beta	T-Wert	Sig.
Anforderungsänderungen	-0,294	-2,638	0,010	-0,314	-2,863	0,006
F5 Effiziente Durchführung	-0,235	-2,111	0,038	-0,256	-2,334	0,023
Anforderungsänderungen × F5 Effiziente Durchführung				0,214	1,945	0,056
R ²	0,145			0,191		
R ² Änderung				0,045		
F	5,874			5,334		

Tabelle 3.16: Regressionsanalyseergebnis Faktor 5 – Effiziente Durchführung

In Tabelle 3.17 sind die Ergebnisse der Analyse des sechsten Faktors „Änderungsbereitschaft“ bei Anforderungsänderungen auf den Projekterfolg. Die Allgemeingültigkeit der Auswertung wird mit einem F-Wert von 10,313 bestätigt, wobei das Konstrukt „Anforderungsänderungen × F6 Änderungsbereitschaft“ selbst mit einem T-Wert

von 1,944 moderat signifikant ist. Anhand des Beta-Koeffizienten kann ein Anstieg des Projekterfolgs um 21,5 Prozent nachgewiesen werden.

	Modell 1			Modell 2		
	Beta	T-Wert	Sig.	Beta	T-Wert	Sig.
Anforderungsänderungen	-0,215	-2,059	0,043	-0,165	-1,561	0,123
F6 Änderungsbereitschaft	0,438	4,186	0,000	0,372	3,442	0,001
Anforderungsänderungen × F6 Änderungsbereitschaft				0,215	1,944	0,056
R ²	0,275			0,313		
R ² Änderung				0,038		
F	13,054			10,313		

Tabelle 3.17: Regressionsanalyseergebnis Faktor 6 – Änderungsbereitschaft

3.3.2.5. Zusammenfassung

Unter Einsatz einer einfachen linearen Regressionsanalyse konnte ein negativer Einfluss von Anforderungsänderungen auf den Projekterfolg nachgewiesen werden. Das erstellte Erklärungsmodell wurde zusätzlich auf seine Aussagekraft überprüft, wodurch von den Ergebnissen der Stichprobe eine Allgemeingültigkeit bewiesen werden konnte.

Die empirische Überprüfung der in Kapitel 3.3.1.2 extrahierten Faktoren konnte drei Erklärungsmodelle als allgemeingültig bestätigen. Für den Interaktionsterm der „Anforderungsänderungen × Endproduktanpassbarkeit“ konnte eine signifikante positive Veränderung des Projekterfolgs nachgewiesen werden. Ein Anstieg des Projekterfolgs konnte ebenfalls für die Anwendung der agilen Methoden des fünften (Effiziente Durchführung) und sechsten Faktors (Änderungsbereitschaft) bewiesen werden.

Für den Einfluss des Faktors „Iterative Entwicklung“ konnte keine allgemeingültige Aussage getroffen werden, da weder das Modell noch das Einflusskonstrukt über eine ausreichend starke Signifikanz verfügten.

Eine Steigerung des Projekterfolgs konnte für den Einsatz der agilen Methoden für Faktor zwei (Arbeitsklima) und vier (Kundenintegration) bei dem Vorhandensein von Anforderungsänderungen nicht bestätigt werden.

Tabelle 2.1 zeigt die Zusammenfassung der Analyseergebnisse für die zu untersuchenden Hypothesen.

Hypothese		Analyse- ergebnis	Allgemein- gültigkeit	Signifikanz
1	Je größer die Anzahl der aufgetretenen Anforderungsänderungen , desto geringer ist der Projekterfolg	Bestätigt	Bestätigt	Hoch signifikant
2a	Je stärker der Faktor der iterativen Entwicklung ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.	Nicht bestätigt	Nicht bestätigt	Nicht signifikant
2b	Je stärker der Faktor des guten Arbeitsklimas ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.	Nicht bestätigt	Bestätigt	Nicht signifikant
2c	Je stärker der Faktor der Anpassbarkeit des Endproduktes ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.	Bestätigt	Bestätigt	Signifikant
2d	Je stärker der Faktor der Integration des Kunden ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.	Nicht bestätigt	Bestätigt	Nicht signifikant
2d i	Je stärker der Faktor der Integration des Kunden ausgeprägt ist, desto größer ist die Kundenzufriedenheit.	Bestätigt	Bestätigt	Moderat signifikant
2e	Je stärker der Faktor der effizienten Durchführung ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.	Bestätigt	Bestätigt	Moderat signifikant
2f	Je stärker der Faktor der Änderungsbereitschaft des Projektteams ausgeprägt ist, desto größer ist der Projekterfolg bei Änderungen der Anforderungen.	Bestätigt	Bestätigt	Moderat signifikant

Tabelle 3.18: Analyseergebnisse für die erstellten Hypothesen

3.4. Erkenntnisse

Im folgenden Abschnitt werden die Implikationen der empirischen Untersuchung für die theoretischen Grundlagen genauer betrachtet und Empfehlungen für den Einsatz agiler Methoden abgeleitet.

Generell konnten die zuvor in Kapitel 2.2 diskutierten Änderungen der Anforderungen während der Projektzeit als negativer Einfluss auf den Projekterfolg bestätigt werden. Da nur der Endwert des Projekterfolges empirisch erhoben wurde, kann nicht eindeutig festgestellt werden, welche Erfolgskriterien in der jeweiligen Projektphase durch die Anforderungsänderungen beeinflusst wurden.

Die Offenheit der Mitarbeiter für neue Methoden konnte eindeutig als positiver Einfluss auf den Projekterfolg ermittelt werden. Diese positive Haltung kann sich sowohl auf Änderungswünsche des Kunden, als auch veränderte Arbeitsabläufe beziehen, wobei diese positive Einstellung anhand der Faktorenanalyse in Relation mit den Kenntnissen der Mitarbeiter gesetzt werden kann. Demzufolge müssen die Projektmitarbeiter einen grundlegenden Lösungsansatz für das gestellte Problem des Projektzieles erstellen können, um für Veränderungen aufgeschlossen zu sein.

In der Literatur findet sich diese Erkenntnis ebenfalls wieder, indem der Einsatz neuer Methoden von den Teammitgliedern aufgrund eines ständigen Weiterlernens forciert wird (Larman, 2003, S. 30). Abhängig von der Erfolgs- und Misserfolgsrate der Anzahl durchgeführter Projekte, könnten sich somit Fortbildungsmaßnahmen für Mitarbeiter positiv auf den gesamten Firmenerfolg auswirken. Den Unternehmen kann daher empfohlen werden eine Kosten- sowie Ursachenanalyse durchzuführen.

Obwohl für die Anwendung agiler Methoden in der Theorie ein gutes Arbeitsklima innerhalb des Projektteams oftmals vorausgesetzt wird, konnte dadurch eine Verbesserung des Projekterfolgs nicht festgestellt werden. Für die Forschung wäre eine explizite Überprüfung der Auswirkungen des Arbeitsklima auf spezifische agile Methoden selbst interessant, wodurch gegebenenfalls ein impliziter Einfluss auf den Gesamterfolg überprüft werden kann. Grundlegend kann ausgesagt werden, dass die Analyse der Stichprobe keinen Unterschied zwischen Projekten mit schlechtem, zu jenen mit einem ausgezeichneten Arbeitsklima ergeben hat.

Ebenso konnte keine positiven Auswirkungen auf den Projekterfolg durch den Einsatz der ausgewählten Methoden der Kundenintegration gefunden werden. Eine separate Untersuchung des Einflusses des Faktors der Kundenintegration hat jedoch ergeben, dass dadurch das Erfolgskriterium der Zufriedenheit des Kunden gesteigert wird. Die Integration des Kunden kann dadurch im Allgemeinen empfohlen werden, da sich die Zufriedenheit bei Projekten positiv beeinflussen lässt.

Der Projekterfolg lässt sich anhand einer modularen und erweiterbaren Architektur des Endproduktes steigern. Die Eigenschaften dieses Faktors können dabei das Produkt bereits in der Entwicklung offener für Änderungen gestalten und dadurch die negativen Auswirkungen von Anforderungsänderungen abschwächen. Ein modularer Aufbau stellt zusätzliche Vorteile

für die Verteilung von Aufgaben und Verantwortungen innerhalb des Entwicklungsteams dar und sollte deshalb gefördert werden.

Durch das Minimieren der Dokumentation, sowie der detaillierten Vorabplanung konnte ein deutlicher Anstieg des Projekterfolgs bei Anforderungsänderungen nachgewiesen werden. Dies lässt sich darauf zurückführen, dass traditionelle Modelle der Entwicklung oftmals einen rigorosen Dokumentationsprozess beinhalten, welcher weder für den Auftraggeber, noch für das Projektteam einen tatsächlichen Nutzen bringt. Da Anforderungsänderungen die Überarbeitung bereits definierter Arbeitsschritte nach sich ziehen und somit jedes Mal den tatsächlich nötigen Planungsaufwand erhöhen, ist eine durchgehende Planung besser für das Endresultat.

Ein Wegfall dieser eigentlich unnötigen Arbeit minimiert somit den Aufwand für das Team und wirkt sich möglicherweise positiv auf das Erreichen der zeitlichen, als auch finanziellen Projektziele aus.

Der Einsatz expliziter iterativer Methoden konnte nicht signifikant als positiver Einfluss auf den Erfolg des Projektes nachgewiesen werden, wobei aufgrund der geringen Datenmenge keine Allgemeingültigkeit zugesprochen werden kann. Die Erstellung und Entwicklung anhand des Feedbacks von Zwischenprodukten ist jedoch für die Stichprobe nicht ausreichend, um die negativen Auswirkungen von Anforderungsänderungen auszugleichen.

Da in der vorliegenden empirischen Erhebung für jedes Projekt nur ein Datensatz erhoben wurde, kann natürlich nicht ausgeschlossen werden, dass sich ein ggf. vorhandener Bias der Befragten auf die Ergebnisse ausgewirkt hat. Es wäre daher für eine höhere Allgemeingültigkeit der Modelle empfehlenswert, die Daten weitere Projektmitglieder zu erheben um somit einen Durchschnittswert für die Analyse zu erhalten (Fowler, 1995, S. 147f.).

4. Schlussbetrachtung

Im folgenden Kapitel erfolgt eine Zusammenfassung der Vorgehensweise und Resultate der vorliegenden Arbeit, sowie kurzer Ausblick für weitere wissenschaftliche Arbeiten.

4.1. Zusammenfassung

Stetig ändernde Anforderungen in der Entwicklung stellen eine Gefahr für die erfolgreiche Durchführung von Projekten dar. Im Bereich der Softwareentwicklung bildeten sich aufgrund dessen verschiedene Vorgehensweisen aus, welche eine flexiblere Gestaltung des Entwicklungsprozesses ermöglichen und den Einflüssen von Anforderungsänderungen entgegenwirken sollen. Mit Veröffentlichung des agilen Manifests (Beck et al., 2001) wurden diese Praktiken und Prinzipien unter dem Begriff der agilen Methoden bekannt.

Aufgrund der immer stärker werdenden Vernetzung von Hard- und Software in Produkten wurde die Frage aufgeworfen, ob sich die agilen Ansätze aus der traditionellen Softwareentwicklung erfolgreich auf die Systementwicklung übertragen lassen. Es wurde ein allgemeines Erklärungsmodell konstruiert, welches von der Annahme ausging, dass Anforderungsänderungen einen negativen Einfluss auf den Projekterfolg haben und agile Methoden den Projekterfolg bei Anforderungsänderungen verbessern können. Ziel dieser Arbeit war es, eine empirische Analyse für die Einflüsse von Anforderungsänderungen im Zusammenhang mit agilen Methoden auf den Erfolg durchzuführen.

Eingangs wurden dazu die wichtigsten Aspekte von Projekten, sowie auftretenden Anforderungsänderungen durch Literaturrecherche näher betrachtet. Für die Beurteilung des Projekterfolgs wurden verschiedene Konzepte untersucht und die wesentlichen Erfolgskriterien für die Bewertung herausgearbeitet.

Für die Auswahl konkreter Praktiken wurden die bekanntesten agilen Managementsysteme und Ansätze auf ihre grundsätzliche Vorgehensweise untersucht und in Kombination mit dem agilen Manifest (Beck et al., 2001) eine Auswahl von vierundzwanzig Methoden getroffen.

Durch persönliche Telefoninterviews, sowie einem webgestützten Fragebogen wurden die Daten österreichischer Firmen erhoben, welche Produkte bestehend aus Hard- und Software erstellen. Insgesamt konnten 72 vollständig ausgefüllte Fragebögen für die Analyse erhoben werden.

Um möglichst aussagekräftige agile Konzepte bestätigen zu können, wurde eine Faktorenanalyse der vierundzwanzig einzelnen agilen Konzepte durchgeführt, welche sechs zugrunde liegende Faktoren identifizieren konnte. Anhand der extrahierten Konstrukte wurde die eingangs formulierte Hypothese des positiven Einflusses agiler Methoden auf den Projekterfolg näher spezifiziert. Es wurden sechs konkrete agile Einflussfaktoren für die weitere Untersuchung erstellt.

Mithilfe einer einfachen linearen Regressionsanalyse wurden die Auswirkungen von Anforderungsänderungen auf den Projekterfolg untersucht. Es konnte bestätigt werden, dass eine Steigerung der Änderungen einen negativen Einfluss auf den Gesamterfolg eines Projektes hat.

Die sechs definierten agilen Konstrukte wurden in Kombination mit den Anforderungsänderungen auf ihre Beeinflussung des Projekterfolgs analysiert. Insgesamt wiesen drei der sechs Konstrukte eine Steigerung des Projekterfolgs beim Auftreten von Anforderungsänderungen auf. Die Analyse ergab, dass eine effiziente Ausführung, bei der unnötiger Mehraufwand minimiert wird, sowie die Erstellung eines leicht anpassbaren Endproduktes, sich positiv auf den Gesamterfolg auswirken. Ebenso konnte die Bereitschaft der Mitarbeiter sich an Änderungen anzupassen als positiver Einflussfaktor nachgewiesen werden.

Eine ausdrücklich iterative Entwicklung zeigte anhand der durchgeführten Analyse keine positive Auswirkung auf den Projekterfolg. Die durchgehende Integration des Auftraggebers, sowie ein gutes Arbeitsklima innerhalb des Projektteams wiesen ebenfalls keine Verbesserung des Projekterfolgs bei Anforderungsänderungen auf. Eine genauere Betrachtung der Kundenintegration und Kundenzufriedenheit ergab jedoch, dass sich allgemein die Zufriedenheit durch eine durchgehende Integration des Auftraggebers steigern lässt.

Im Zuge dieser Forschungsarbeit konnte festgestellt werden, dass nicht alle agilen Methoden einen positiven Einfluss auf den Erfolg bei Anforderungsänderungen ausüben. Für weitere Studien wäre es interessant eine Ausdehnung der Erhebung auf weitere europäische Staaten vorzunehmen und zu überprüfen, ob sich die in Österreich erhaltenen Ergebnisse bestätigen lassen.

4.2. Ausblick

Für weitere Forschungsarbeiten wäre eine erweiterte Datensammlung empfehlenswert um die Allgemeingültigkeit der Ergebnisse für alle erhaltenen Konstrukte bestätigen zu können. Um die nötige Anzahl der Datensätze zu erhalten, wäre eine Ausweitung der Studie auf den deutschsprachigen Raum in einem ersten Schritt empfehlenswert. Für einen internationalen Vergleich wäre gegebenenfalls eine Erweiterung auf Europa interessant.

Im Zuge dieser Arbeit wurden bereits die Daten zur Verwendung konkreter Managementmodelle erhoben, jedoch keine explizite Auswertung bezüglich der konkreten Methoden, welche für die jeweiligen Modelle vorgeschlagen sind, durchgeführt, da dies den Umfang dieser Arbeit überstiegen hätte. Zukünftig wäre es denkbar die Projekte konkret auf den Einsatz der jeweiligen Vorgehensweise für das ausgewählte Managementmodell zu überprüfen um somit eine Aussage treffen zu können, inwieweit den Praktiken der einzelnen Methoden in der Praxis tatsächlich gefolgt wird.

5. Abkürzungsverzeichnis

AAE	Anforderungsänderungen
AF	Agiler Faktor
C3	Chrysler Comprehensive Compensation System
CE	Concurrent Engineering
Cov	Covariance
DSDM	Dynamic Systems Development Method / Dynamic Systems Delivery Method
FDD	Feature Driven Development
JIT	Just-In-Time
KMO	Kaiser-Meyer-Olkin Kriterium
MA	Mitarbeiter
MIT	Massachusetts Institute of Technology
min	minimal
MSA	measure of sampling adequacy
ÖNACE	Österreichische Nomenclature européenne des activités économiques
PA	Prozentuelle Abschätzung
RS	Realisation-Steps
SE	Simultaneous Engineering
Sig	Signifikanz
V	Variants of Overall Concepts
XP	Extreme Programming
YAGNI	You Ain't Gonna Need It

6. Formelverzeichnis

FORMEL 3.1: ABSOLUTER ERFOLG PRO KRITERIUM DES PROJEKTERFOLGES	54
FORMEL 3.2: RELATIVER ERFOLG PRO KRITERIUM DES PROJEKTERFOLGES	54
FORMEL 3.3: PROJEKTERFOLG IN PROZENT	54
FORMEL 3.4: FAKTOR FÜR DIE PROZENTUELLE AUFTEILUNG DER ÄNDERUNGEN	57
FORMEL 3.5: PROZENTUELLE ABSCHÄTZUNG DER ANFORDERUNGSÄNDERUNGEN	57
FORMEL 3.6: KENNZAHL DER ANFORDERUNGSÄNDERUNGEN PRO PROJEKTPHASE	57
FORMEL 3.7: KENNZAHL DER ANFORDERUNGSÄNDERUNGEN DES PROJEKTES	57
FORMEL 3.8: GLEICHUNGSMODELL DER FAKTORENANALYSE (LANDAU & EVERITT, 2004, S. 278)	65
FORMEL 3.9: FAKTORENEXTRAKTION (LANDAU & EVERITT, 2004, S. 278)	65
FORMEL 3.10: CRONBACH'S ALPHA (FIELD, 2005, S. 667)	69
FORMEL 3.11: EINFACHE LINEARE REGRESSIONSFUNKTION (BACKHAUS ET AL., 1996, S. 14)	73
FORMEL 3.12: NULLHYPOTHESE FÜR EINFACHE LINEARE REGRESSION (CHATTEFUEE & HADI, 2006, S. 132)	73
FORMEL 3.13: REGRESSIONSSATZ DES ERKLÄRUNGSMODELLS FÜR DEN EINFLUSS VON ANFORDERUNGSÄNDERUNGEN	74
FORMEL 3.14: ALLGEMEINER REGRESSIONSSATZ.....	75
FORMEL 3.15: ZIELFUNKTION DER REGRESSIONSANALYSE	75
FORMEL 3.16: REGRESSIONSSATZ DER ERKLÄRUNGSMODELLE AGILER METHODEN	76

7. Abbildungsverzeichnis

ABBILDUNG 1.1: ZU BETRACHTENDES ERKLÄRUNGSMODELL	2
ABBILDUNG 1.2: AUFBAU DER ARBEIT	3
ABBILDUNG 2.1: PROJEKTLANDSCHAFT NACH WYSOCKI (2007, S. 17)	6
ABBILDUNG 2.2: MANAGEMENTMETHODEN-MODELL VON WYSOCKI (2007, S. 47).....	7
ABBILDUNG 2.3: STAKEHOLDER MANAGEMENTPROZESS NACH TURNER (2009, S. 77)	9
ABBILDUNG 2.4: ÄNDERUNGSKOSTEN RELATION ZUR PROJEKTPHASE IN ANLEHNUNG AN LOCK (2007, S. 404)..	12
ABBILDUNG 2.5: "IRON TRIANGLE" NACH ATKINSON (1999)	13
ABBILDUNG 2.6: KAUSALE ZUSAMMENHÄNGE DER „BALANCED SCORECARD" NACH KAPLAN UND NORTON (1992)	15
ABBILDUNG 2.7: „SUCCESS SLIDER" BEWERTUNG NACH THOMSETT (2002, S. 74)	17
ABBILDUNG 2.8: WERTIGKEITSÄNDERUNG DER ERFOLGSKRITERIEN IN ANLEHNUNG AN THOMSETT (2002, S. 322F.).....	17
ABBILDUNG 2.9: WASSERFALLMODEL BASIEREND AUF RAKITIN (2001, S. 19) UND SOMMERVILLE (2006, S. 66) 20	
ABBILDUNG 2.10: V-MODELL BASIEREND AUF RAMACHANDRAN, PATEL (RAMACHANDRAN & DE CARVALHO, 2009, S. 20) UND CHARVAT (2003, S. 123)	22
ABBILDUNG 2.11: SPIRALMODELL BASIEREND AUF HELMS ET AL. (2005).....	23
ABBILDUNG 2.12: DAS AGILE MANIFEST (BECK ET AL., 2001)	25
ABBILDUNG 2.13: ZEITERSPARNIS DURCH SIMULTANEOUS ENGINEERING	31
ABBILDUNG 2.14: „HOUSE OF LEAN“ BASIEREND AUF WILSON (2010, S. 300)	32
ABBILDUNG 2.15: „SET-BASED DESIGN“ NACH SHIGLEY (HABERFELLNER & DE WECK, 2005).....	34
ABBILDUNG 2.16: SCHEMATISCHER ABLAUF VON SCRUM BASIEREND AUF SCHWABER (2004, S. 5) UND GLOGER (2008, S. 13)	44
ABBILDUNG 2.17: DIE „CRYSTAL“ METHODENFAMILIE (HRUSCHKA ET AL., 2009, S. 55)	45
ABBILDUNG 3.1: GESAMTANZAHL DER MITARBEITER	51
ABBILDUNG 3.2: ANZAHL DER PROJEKTMITARBEITER.....	51
ABBILDUNG 3.3: PROJEKTDAUER IN MONATEN	52
ABBILDUNG 3.4: KOMPLEXITÄT DES BETRACHTETEN PROJEKTES	52
ABBILDUNG 3.5: GEWICHT DER ERFÜLLUNG FÜR DEN PROJEKTERFOLG.....	54
ABBILDUNG 3.6: BERECHNETE PROJEKTERFOLGSVERTEILUNG	55
ABBILDUNG 3.7: RELEVANTE ENTWICKLUNGSPROJEKTPHASEN FÜR ANFORDERUNGSÄNDERUNGEN.....	56
ABBILDUNG 3.8: KENNZAHLENVERTEILUNG DER ANFORDERUNGSÄNDERUNGEN	58
ABBILDUNG 3.9: SIMPLIZITÄT DER AUSFÜHRUNG DES ENDPRODUKTS	62
ABBILDUNG 3.10: ERHOBENE VERTEILUNG VON PROJEKTMANAGEMENTMODELLEN	63
ABBILDUNG 3.11: SCHEMATISCHE DARSTELLUNG EINER FAKTORENABBILDUNG MIT ZWEI ZUGRUNDELIEGENDEN VARIABLEN (KIM & MUELLER, 1978, S. 13).....	65
ABBILDUNG 3.12: KOMMUNALITÄT MIT RESTWERT BEI 2 ZUGRUNDELIEGENDEN FAKTOREN NACH BACKHAUS (1996, S. 221)	66

8. Tabellenverzeichnis

TABELLE 2.1: KATEGORISIERUNG VON PROJEKTEN NACH MÜLLER & TURNER (2007, S. 301).....	5
TABELLE 2.2: STAKEHOLDER-IDENTIFIKATION NACH BROWN (2007, S. 54F.).....	9
TABELLE 2.3: VOR- UND NACHTEILE DES WASSERFALLMODELLS.....	21
TABELLE 2.4: GEGENÜBERSTELLUNG TRADITIONELLER MANAGEMENTMODELLE NACH MCCONNELL (1996, S. 156F.).....	24
TABELLE 3.1: FAKTOREN FÜR PROJEKTERFOLG.....	53
TABELLE 3.2: FAKTOREN FÜR DIE BERECHNUNG EINER KENNZAHL DER ANFORDERUNGSÄNDERUNGEN.....	56
TABELLE 3.3: MULTIPLIKATOREN DES UMFANGS DER ANFORDERUNGSÄNDERUNGEN	57
TABELLE 3.4: INDIKATOREN ZUR OPERATIONALISIERUNG AGILER METHODEN.....	59
TABELLE 3.5: EIGENWERTE DER EXTRAHIERTEN FAKTOREN	66
TABELLE 3.6: VARIABLEN MIT FAKTORENLOADUNG UND KOMMUNALITÄTEN DER ERSTE FAKTORENANALYSE; LADUNGSWERTE UNTER 0,4 WURDEN VON DER ANZEIGE AUSGESCHLOSSEN UND DIE VARIABLEN WURDEN ABSTEIGEND IHRER FAKTORLOADUNG ANGEREiht	67
TABELLE 3.7: ERGEBNISSE DER ZWEITEN FAKTORENANALYSE; LADUNGSWERTE UNTER 0,4 WURDEN VON DER ANZEIGE AUSGESCHLOSSEN UND DIE VARIABLEN WURDEN ABSTEIGEND IHRER FAKTORLOADUNG ANGEREiht	69
TABELLE 3.8: CRONBACH'S ALPHA FÜR 7 FAKTOREN.....	70
TABELLE 3.9: ERGEBNISSE DER FINALEN FAKTORENANALYSE INKLUSIVE CRONBACH'S ALPHA; LADUNGSWERTE UNTER 0,4 WURDEN VON DER ANZEIGE AUSGESCHLOSSEN UND DIE VARIABLEN WURDEN ABSTEIGEND IHRER FAKTORLOADUNG ANGEREiht	70
TABELLE 3.10: REGRESSIONSANALYSEERGEBNIS AUSWIRKUNG VON ANFORDERUNGSERGEBNISSEN AUF PROJEKTERFOLG	74
TABELLE 3.11: REGRESSIONSANALYSEERGEBNIS FAKTOR 1 – ITERATIVE ENTWICKLUNG	76
TABELLE 3.12: REGRESSIONSANALYSEERGEBNIS FAKTOR 2 – ARBEITSKLIMA	77
TABELLE 3.13: REGRESSIONSANALYSEERGEBNIS FAKTOR 3 – ENDPRODUKTANPASSBARKEIT.....	77
TABELLE 3.14: REGRESSIONSANALYSE – EINFLUSS DER KUNDENINTEGRATION AUF KUNDENZUFRIEDENHEIT	78
TABELLE 3.15: REGRESSIONSANALYSEERGEBNIS FAKTOR 4 – KUNDENINTEGRATION	78
TABELLE 3.16: REGRESSIONSANALYSEERGEBNIS FAKTOR 5 – EFFIZIENTE DURCHFÜHRUNG.....	78
TABELLE 3.17: REGRESSIONSANALYSEERGEBNIS FAKTOR 6 – ÄNDERUNGSBEREITSCHAFT	79
TABELLE 3.18: ANALYSEERGEBNISSE FÜR DIE ERSTELLTEN HYPOTHESEN.....	80

9. Literaturverzeichnis

- Aguanno, K. J. (2005). *Managing Agile Projects*. Lakefield: Multi-Media Publications Inc.
- Atkinson, R. (1999). Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria. *International Journal of Project Management*, 17 (6), 337-342.
- Backhaus et al. (1996). *Multivariate Analysemethoden: Eine anwendungsorientierte Einführung*. Berlin: Springer.
- Beck et al. (2001). *Manifesto for Agile Software Development*. Abgerufen am 20. September 2009 von Manifesto for Agile Software Development: <http://agilemanifesto.org/>
- Beucher, O. (2007). *Wahrscheinlichkeitsrechnung und Statistik mit MATLAB* (Zweite, bearbeitete Auflage Ausg.). Heidelberg: Springer.
- Bleek, W.-G., & Wolf, H. (2008). *Agile Softwareentwicklung, Werte, Konzepte und Methoden*. Zutphen, Niederlande: dpunkt.verlag.
- Bourne, L. (2009). *Stakeholder relationship management: a maturity model for organisational*. Cornwall: Gower Publishing Limited.
- Brown, J. T. (2007). *The Handbook of Program Management: How to Facilitate Project Success with Optimal Program Management*. New York: McGraw-Hill.
- Charvat, J. (2003). *Project Management Methodologies: Selecting, Implementing, and Supporting Methodologies and Processes for Projects*. Hoboken, New Jersey: John Wiley & Sons, Inc.
- Chattefuee, S., & Hadi, A. S. (2006). *Regression Analysis by Example*. Hoboken, New Jersey: John Wiley & Sons.
- Cockburn, A. (2006). *Agile Software Development: The Cooperative Game* (Second Edition Ausg.). Boston, MA: Addison Wesley.
- Cockburn, A. (2004). *Crystal Clear A Human-Powered Methodology for Small Teams*. Boston, MA: Addison Wesley.
- Cohen, A. R., & Bradford, D. L. (2005). *Influence without Authority*. Hoboken: John Wiley & Sons, Inc.
- Cohn, M. (2006). *Agile Estimating and Planning*. Stoughton, Massachusetts: Prentice Hall.
- de Wit, A. (1988). Measurement of Project Success. *Journal of Project Management, Band 6, Heft 3*, 164-170.
- DIN Deutsches Institut für Normung e.V. (2008). *Einführung in die DIN-Normen*. Wiesbaden: B.G. Teubner.

- Dvir, D., & Lechler, T. (2004). Plans are nothing, changing plans is everything: the impact of changes on project success. *Research Policy* (33), 1-15.
- Ehrlenspiel et al. (2007). *Kostengünstig Entwickeln und Konstruieren: Kostenmanagement bei der integrierten Produktentwicklung* (Sechste Auflage Ausg.). Heidelberg: Springer.
- Fairley, R. E. (2009). *Managing and Leading Software Projects*. Hoboken, New Jersey: John Wiley & Sons, Inc.
- Field, A. (2005). *Discovering Statistics using SPSS*. London: Sage.
- Firer, S., & Swartz, N. (2007). The market for audit services in the 'New' South Africa. *Southern African Business Review* , 11 (11), 45-64.
- Fowler, F. J. (1995). *Improving survey questions*. USA: Sage Publications.
- Gilb, T. (2005). *Competitive Engineering - A Handbook for Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage*. Great Britain: Elsevier Butterworth-Heinemann.
- Gloger, B. (2008). *Scrum - Produkte zuverlässig und schnell entwickeln*. München: Carl Hanser Verlag.
- Graeme, T., & Walter, F. (2008). Success in IT projects: A matter of definition? *International Journal of Project Management* (26), 733-742.
- Groves et al. (2004). *Survey Methodology*. New Jersey: John Wiley & Sons, Inc.
- Gunasekaran, A. (2001). *Agile Manufacturing: The 21st Century Competitive Strategy*. Kidlington, Oxford: Elsevier.
- Gunduz, M., & Hanna, A. S. (2004). Benchmarking change order impacts on productivity for electrical and mechanical projects. *Building and Environment* , 1068-1075.
- Haberfellner, R., & de Weck, O. (2005). *Agile SYSTEMS ENGINEERING versus AGILE SYSTEMS engineering*. Rochester, NY: INCOSE.
- Hair et al. (2009). *Multivariate Data Analysis* (7th Edition Ausg.). Upper Saddle River, NJ: Prentice Hall.
- Helms et al. (2005). A field study of the Wheel—a usability engineering process model. *The Journal of Systems and Software* (79), 841-858.
- Hill, G. M. (2008). *The Complete Project Management Office Handbook* (Second Edition Ausg.). Boca Raton, Florida, United States of America: Auerbach Publications.
- Holcombe, M. (2008). *Running an Agile Software Development Project*. Hoboken, NJ: John Wiley & Sons, Inc.

- Howard et al. (2008). Reliability of an observational tool to assess the organization of work. *International Journal of Industrial Ergonomics* , 260-266.
- Hruschka et al. (2009). *Agility kompakt, Tipps für erfolgreiche Systementwicklung* (Bd. 2. Auflage). Heidelberg: Spektrum Akademischer Verlag.
- Jahren, C. T., & Ashe, A. M. (1990). Predictors of Cost-Overrun Rates. *Journal of Construction Engineering and Management* , 547-552.
- Kaplan, R. S., & Norton, D. P. (1992). The Balanced Scorecard - Measures that Drive Performance. *Harvard Business Review* , 71-79.
- Kim, J.-O., & Mueller, C. W. (1978). Introduction to Factor Analysis: What it Is and How to Do It. *Quantative Applications in the Social Sciences* , Sage University Paper Series on *Quantative Applications in the Social Sciences (07-013)* . Newbury Park, CA: Sage.
- Koch, A. S. (2005). *Agile Software Development - Evaluating the methods for your Organization*. Norwood, Massachusetts: Artech House Inc.
- Körner, M. (2008). *Geschäftsprojekte zum Erfolg führen*. Leipzig: Springer.
- Landau, S., & Everitt, B. S. (2004). *A handbook of statistical analyses using SPSS*. Boca Raton, Florida: Chapman & Hall/CRC Press LLC.
- Larman, C. (2003). *Agile and Iterative Development: A Manager's Guide*. Boston, Massachusetts: Pearson Education.
- Leach, L. P. (2005). *Lean Project Management: Eight Principles for Success*. Advanced Projects.
- Lean Enterprise Institute. (2008). *Lean Lexicon: a graphical glossary for Lean Thinkers* (4th edition Ausg.). Cambridge, MA: The Lean Enterprise Institute, Inc.
- Lehman et al. (2005). *JMP for Basic Univariate and Multivariate Statistics: A Step-by-Step Guide*. Cary, North Carolina: SAS Press.
- Levine, M. K. (2009). *A Tale of Two Systems: Lean and Agile Software Development for Business Leaders*. New York, NY: CRC Press.
- Lock, D. (2007). *Project Management*. Hampshire, England, United Kingdom: Gower Publishing Limited.
- McConnell, S. (1996). *Rapid Development*. Redmond, Washington: Microsoft Press.
- Müller, R., & Turner, R. (2007). The Influence of Project Managers on Project Success Criteria and Project Success by Type of Project. *European Management Journal* , 25 (4), 298-309.

- Palmer, S. J., & Felsing, J. M. (2002). *A Practical Guide to Feature-Driven Development*. Upper Saddle River, NJ: Prentice Hall.
- Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Upper Saddle River, NJ: Addison Wesley.
- Preißner, A. (2003). *Balanced Scorecard anwenden*. Hanser Wirtschaft.
- Project Management Institute. (2004). *A Guide to the Project Management Body of Knowledge (PMBOK Guide) Third Edition*.
- Qumer, A., & Henderson-Sellers, B. (2008). An evaluation of the degree of agility in six agile methods and its applicability for method engineering. *Information and Software Technology* (50), 280-295.
- Rakitin, S. R. (2001). *Software Verification and Validation for Practitioners and Managers* (Second Edition Ausg.). Norwood, MA: Artech House Inc.
- Ramachandran, M., & de Carvalho, R. A. (2009). *Handbook of Research on Software Engineering and Productivity Technologies: Implications of Globalization*. Hershey, PA: Engineering Science Reference.
- Schwaber, K. (2004). *Agile Project Management with Scrum*. Redmond, Washington: Microsoft Press.
- Shore, J., & Warden, S. (2008). *The Art of Agile Development*. USA: O'Reilly Media.
- Sliger, M., & Broderick, S. (2008). *The Software Project Manager's Bridge to Agility*. Crawfordsville, Indiana: Pearson Education, Inc.
- Smith, G., & Sidky, A. (2009). *Becoming Agile in an imperfect world*. Greenwich, Connecticut: Manning Publications.
- Sommerville, I. (2006). *Software Engineering* (Eight Edition Ausg.). Harlow, Essex: Addison Wesley.
- Stephens, M., & Rosenberg, D. (2003). *Extreme Programming Refactored: The Case Against XP*. Berkeley, CA: Apress.
- Subramaniam, V., & Hunt, A. (2006). *Practices of an Agile Developer - Working in the Real World*. Raleigh: Pragmatic Bookshelf.
- The DSDM Consortium. (2003). *DSDM Business Focused Development*. London: Addison Wesley.
- Thomsett, R. (2002). *Radical Project Management*. Upper Saddle River, New Jersey, United States of America: Prentice Hall PTR.
- Turner, J. R. (2009). *The Handbook of Project-Based Management*. London: McGraw-Hill.

Versteegen et al. (2005). *Prozessübergreifendes Projektmanagement*. Berlin: Springer.

Whitty, S. J., & Maylor, H. (2007). And then came Complex Project Management (revised). *International Journal of Project Management* (27), 304-310.

Wilson, L. (2010). *How to Implement Lean Manufacturing*. New York, NY: McGraw-Hill.

Wysocki, R. K. (2007). *Effective Project Management, Traditional, Adaptive, Extreme* (4th Edition Ausg.). Indianapolis, Indiana, United States of America: Wiley Publishing, Inc.

Young, T. L. (2006). *Successful Project Management* (Second Edition Ausg.). Bodmin, Cornwall, Great Britain: Kogan Page Limited.

10. Anhang

Fragebogen

Eigenwerte der extrahierten Faktoren

Häufigkeitstabellen der Variablen agiler Methoden

Fragebogen

Sehr geehrte Damen und Herren!

Vielen Dank im Voraus für Ihre Teilnahme an der Umfrage zu den in der Industrie eingesetzten Managementmethoden!

Alle Fragen im Fragebogen beziehen sich auf **ein spezifisches Projekt**, welches folgenden Anforderungen entsprechen sollte:

- das **Endprodukt** bestand aus **Hard- und Software**
- das Projekt wurde **nicht für die eigene Abteilung entwickelt**
- das Projekt wurde erst **vor Kurzem abgeschlossen**

Diese Umfrage enthält 21 Fragen.

Abschnitt A

Alle Pflichtfragen in dieser Umfrage sind mit einem roten Asterisk * gekennzeichnet und müssen beantwortet werden, bevor Sie zur nächsten Seite weitergehen können.

Obwohl alle anderen Fragen absolut optional sind, würde auch die Beantwortung dieser sehr geschätzt werden.

1	Wie erfolgreich würde Sie Ihr Projekt auf einer Skala von 1-10 bewerten?														
	(1 = absoluter Fehlschlag, 10 = absoluter Erfolg)														
	Bitte bewerten Sie anhand der Skala.	1	2	3	4	5	6	7	8	9	10				
		<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>				

2	Wie wichtig war die Erfüllung folgender Punkte für Ihr Projekt?						
	Bitte bewerten Sie anhand der Skala. *						
		<u>Absolut unwichtig</u>	Unwichtig	Teilweise wichtig	Wichtig	<u>Extrem wichtig</u>	
	Zufriedenstellung des Auftraggebers	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	Erfüllung der Produktspezifikationen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	Einhaltung des Projektbudgets	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	Termingerechte Fertigstellung	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
	Wertschaffung für das eigene Unternehmen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

10. Anhang

3 In welchem Ausmaß wurden folgende Punkte in Ihrem Projekt abgedeckt? Bitte bewerten Sie anhand der Skala. *							
	Stimme <u>vollkommen</u> zu	Stimme zu	Stimme eher zu	Unentschieden	Stimme eher nicht zu	Stimme nicht zu	Stimme <u>keinesfalls</u> zu
Auftraggeber war mit dem Ergebnis zufrieden	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Produktspezifikationen wurden eingehalten	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Projektbudget wurde eingehalten	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Projekt wurde termingerecht fertiggestellt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Projekt generierte Wert für die Firma	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Abschnitt B

Die folgenden Fragen beziehen sich nur auf Änderungen, welche vom Auftraggeber des Projekts ausgelöst wurden.

Äußerte der Auftraggeber bspw. einen konkreten Wunsch, welcher diverse Änderungen nach sich zog, so sollen diese hier berücksichtigt werden.

Gab es während der Durchführung neue Erkenntnisse durch das Projektteam, die Änderungen mit sich brachten, so sollen diese hier nicht miteinbezogen werden.

4 Wie viel Prozent der Anforderungen wurden insgesamt <u>vom Auftraggeber</u> über den ganzen Projektzeitraum geändert? *

5 Bitte geben Sie an <u>wie oft</u> es zu Änderungen der Produktspezifikationen durch den Auftraggeber zum jeweiligen Zeitpunkt kam. *							
	Durchgehend	Sehr häufig	Häufig	Gelegentlich	Selten	Sehr selten	Nie
kurz nach Beginn der Entwicklung	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
während der Entwicklung	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
kurz vor Entwicklungsende	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
nach Abschluss des Entwicklungsprojektes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

6 <u>Wie umfangreich</u> waren die <u>Änderungen</u> der Produktspezifikationen zum jeweiligen Zeitpunkt, verglichen mit den ursprünglichen Anforderungen? *							
	<u>Sehr groß</u>	Groß	Eher groß	Unentschieden	Eher klein	Klein	<u>Sehr klein</u>
kurz nach Beginn der Entwicklung	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
während der Entwicklung	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
kurz vor Entwicklungsende	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
nach Abschluss des Entwicklungsprojektes	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Handelte es sich bei den Änderungen um große oder kleine Änderungen?

Bspw. wurde *nur* die Farbe eines Teiles geändert würde dies wahrscheinlich eine sehr kleine Änderung sein, wurde die Form eines Teiles geändert würde dies eher eine große Änderung entsprechen.

10. Anhang

7 Aus welchen Gründen kam es zu Anforderungsänderungen des Kunden? Bitte kreuzen Sie alle zutreffenden Punkte an.	
Bitte wählen Sie alle zutreffenden Antworten aus:	
<input type="radio"/>	falsche/fehlende Spezifikationen zu Beginn
<input type="radio"/>	Kosten senken
<input type="radio"/>	Zeitraum einhalten
<input type="radio"/>	zusätzliche Funktionalität
<input type="radio"/>	neue Technologie
<input type="radio"/>	Anpassung an Markt
<input type="radio"/>	Gesetzesänderung
<input type="radio"/>	Sonstiges:

Abschnitt C

8 Bitte bewerten Sie inwiefern folgende Aussagen für Ihr Projekt zutreffen. *							
	Stimme <u>vollkommen</u> zu	Stimme zu	Stimme eher zu	Unent- schieden	Stimme eher nicht zu	Stimme nicht zu	Stimme <u>keinesfalls</u> zu
Die Projektmitarbeiter waren sehr motiviert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Projektmitarbeiter lehnten Veränderungen ab	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Projektmitarbeiter waren überdurchschnittlich erfahren	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Projektmitarbeiter waren mit der Aufgabenstellung des Projektes überfordert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Der Wissensaustausch zwischen den Projektmitarbeitern war schnell und unkompliziert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Das Projektteam war räumlich voneinander getrennt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Das Arbeitsklima innerhalb des Projektteams war sehr gut	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

10. Anhang

9 Bitte bewerten Sie inwiefern folgende Aussagen für Ihr Projekt zutreffen. *							
	Stimme <u>vollkommen</u> zu	Stimme zu	Stimme eher zu	Unent- schieden	Stimme eher nicht zu	Stimme nicht zu	Stimme <u>keinesfalls</u> zu
Der Kunde/Endnutzer war durchgehend in die Produktentwicklung integriert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Es gab durchgehend Feedback vom Kunden/Endnutzer	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Produkthanforderungen wurden vom Kunden priorisiert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Es gab Zwischenprodukte bevor das Endprodukt fertig war ¹	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Zwischenprodukte wurden dem Kunden zur Verfügung gestellt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Entwicklung fand zyklisch bzw. iterativ statt ²	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Das Feedback des Kunden zu den Zwischenprodukten wurde zur Verbesserung des Endproduktes verwendet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Der gesamte Projektablauf wurde detailliert vor Beginn des Entwicklungsprojektes geplant	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Während der Entwicklung wurde aktiv nach Möglichkeiten zur Verbesserung des Entwicklungsprozesses gesucht	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Es wurde Alles so ausführlich wie möglich dokumentiert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

¹ Falls **keine Zwischenprodukte** vorhanden waren, bitte beantworten Sie alle weiteren Fragen diesbezüglich mit "Stimme keinesfalls zu"

² Bei **iterativer bzw. zyklischer Entwicklung** werden die einzelnen Phasen wiederholt durchlaufen, bis das Endprodukt fertiggestellt ist. In jeder Phase findet eine Situationsanalyse, Weiterentwicklung, Test des erzeugten Ergebnisses und Suche nach Verbesserungsmöglichkeiten statt. Die Phasen werden solange wiederholt, bis das spezifizierte Produkt fertiggestellt ist.

10 Bitte bewerten Sie inwiefern folgende Aussagen für Ihr Projekt zutreffen. *							
	Stimme <u>vollkommen</u> zu	Stimme zu	Stimme eher zu	Unent- schieden	Stimme eher nicht zu	Stimme nicht zu	Stimme <u>keinesfalls</u> zu
Die Mitarbeiter mussten einem fix vorgegebenen Entwicklungsprozess folgen	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Bereits entwickelte Module/Technologien/Komponenten wurden wiederverwendet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Das Produkt wurde so einfach als möglich aufgebaut	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Hardware wurde vor der Software entwickelt	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Das Endprodukt enthielt Funktionen die nicht explizit gefordert waren	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Architektur des Produktes lies Veränderungen leicht zu	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Das Produkt wurde vollständig modular aufgebaut	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

10. Anhang

11 Wurde bei der Projektdurchführung bestimmten Methoden gefolgt?	
Bitte wählen Sie alle zutreffenden Antworten aus:	
<input type="radio"/>	Nein
<input type="radio"/>	V-Modell
<input type="radio"/>	Wasserfallmodell
<input type="radio"/>	Spiralmodell
<input type="radio"/>	Scrum
<input type="radio"/>	Feature Driven Development (FDD)
<input type="radio"/>	eXtreme Programming
<input type="radio"/>	Crystal (Clear/Yellow/Orange/Red/Magenta/Blue)
<input type="radio"/>	Lean Software Development (LD)
<input type="radio"/>	Dynamic Systems Development Method (DSDM)
<input type="radio"/>	Simultaneous Engineering
<input type="radio"/>	Sonstiges:

Abschnitt D

Alle Pflichtfragen in dieser Umfrage sind mit einem roten Asterisk * gekennzeichnet und müssen beantwortet werden, bevor Sie zur nächsten Seite weitergehen können.

Obwohl alle anderen Fragen absolut optional sind, würde auch die Beantwortung dieser sehr geschätzt werden.

12 Womit beschäftigte sich Ihr Projekt?
Bitte schreiben Sie Ihre Antwort hier

13 Wie komplex würden Sie das Projekt im Vergleich mit Ihren üblicherweise durchgeführten Projekten beurteilen? *							
	Sehr komplex	Komplex	Eher komplex	Durchschnittlich	Eher simpel	Simpel	Sehr simpel
Bitte bewerten Sie anhand der Skala.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

14 Bitte bewerten Sie inwieweit folgende Aussagen zutreffen. *							
	Stimme vollkommen zu	Stimme zu	Stimme eher zu	Unentschieden	Stimme eher nicht zu	Stimme nicht zu	Stimme keinesfalls zu
Das zu entwickelnde Endprodukt war von Anfang an klar definiert	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Die Vorgehensweise zur Entwicklung des Endproduktes war eindeutig	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

15 Wer war Auftraggeber des Projektes?	
<input type="radio"/>	andere Abteilung in eigener Firma
<input type="radio"/>	andere Firma
<input type="radio"/>	öffentliche Hand
<input type="radio"/>	Sonstiges

10. Anhang

16 Wie viele Monate dauerte das Projekt?

Bitte schreiben Sie Ihre Antwort hier

17 Wie viele Personen waren bei der Durchführung des Projektes beteiligt?

Bitte schreiben Sie Ihre Antwort hier

18 Wie hoch war das angesetzte Projektbudget?

Bitte schreiben Sie Ihre Antwort hier

19 Wie viele Personen sind in Ihrem Unternehmen beschäftigt?

Bitte schreiben Sie Ihre Antwort hier

20 Wie hoch war der Umsatz Ihrer Firma im letzten abgeschlossenen Geschäftsjahr?

Bitte schreiben Sie Ihre Antwort hier

Abschluss

Vielen Dank für Ihre Mitarbeit!

Bitte geben Sie nachfolgend Ihre Kontaktdaten an um die Ergebnisse der Umfrage zu erhalten.

Ihre persönlichen Daten werden nicht an Dritte weitergegeben. Die Ergebnisse der Umfrage werden nur anonymisiert weiterverwendet!

21 Kontaktdaten *

Name

Name der Firma

Email Adresse

Vielen Dank für die Beantwortung des Fragebogens.

Eigenwerte der extrahierten Faktoren

Komponente	Anfängliche Eigenwerte			Rotierte Summe der quadrierten Ladungen		
	Gesamt	% der Varianz	Kumulierte %	Gesamt	% der Varianz	Kumulierte %
1	4,281	17,839	17,839	3,011	12,546	12,546
2	4,035	16,813	34,652	2,708	11,283	23,829
3	1,928	8,034	42,685	2,506	10,441	34,271
4	1,674	6,973	49,659	2,378	9,908	44,179
5	1,484	6,182	55,841	2,178	9,074	53,253
6	1,374	5,726	61,567	1,667	6,945	60,197
7	1,167	4,864	66,431	1,496	6,234	66,431
8	,983	4,098	70,529			
9	,915	3,811	74,340			
10	,852	3,552	77,892			
11	,766	3,193	81,085			
12	,685	2,854	83,939			
13	,600	2,502	86,441			
14	,520	2,167	88,607			
15	,492	2,050	90,658			
16	,402	1,675	92,333			
17	,388	1,617	93,950			
18	,319	1,328	95,278			
19	,303	1,260	96,539			
20	,233	,972	97,511			
21	,209	,872	98,382			
22	,158	,659	99,041			
23	,138	,576	99,616			
24	,092	,384	100,000			

Häufigkeitstabellen agiler Methoden

Die Projektmitarbeiter waren sehr motiviert

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme eher nicht zu	1	1,4	1,4	1,4
	Unentschieden	3	4,2	4,2	5,6
	Stimme eher zu	14	19,4	19,4	25,0
	Stimme zu	31	43,1	43,1	68,1
	Stimme vollkommen zu	23	31,9	31,9	100,0
	Gesamt	72	100,0	100,0	

Die Projektmitarbeiter lehnten Veränderungen ab

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	8	11,1	11,1	11,1
	Stimme nicht zu	25	34,7	34,7	45,8
	Stimme eher nicht zu	10	13,9	13,9	59,7
	Unentschieden	16	22,2	22,2	81,9
	Stimme eher zu	9	12,5	12,5	94,4
	Stimme zu	4	5,6	5,6	100,0
	Gesamt	72	100,0	100,0	

Die Projektmitarbeiter waren überdurchschnittlich erfahren

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme nicht zu	6	8,3	8,3	8,3
	Stimme eher nicht zu	7	9,7	9,7	18,1
	Unentschieden	18	25,0	25,0	43,1
	Stimme eher zu	15	20,8	20,8	63,9
	Stimme zu	20	27,8	27,8	91,7
	Stimme vollkommen zu	6	8,3	8,3	100,0
	Gesamt	72	100,0	100,0	

Die Projektmitarbeiter waren mit der Aufgabenstellung des Projektes überfordert

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	15	20,8	20,8	20,8
	Stimme nicht zu	31	43,1	43,1	63,9
	Stimme eher nicht zu	16	22,2	22,2	86,1
	Unentschieden	3	4,2	4,2	90,3
	Stimme eher zu	5	6,9	6,9	97,2
	Stimme zu	2	2,8	2,8	100,0
	Gesamt	72	100,0	100,0	

Der Wissensaustausch zwischen den Projektmitarbeitern war schnell und unkompliziert

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme nicht zu	5	6,9	6,9	6,9
	Stimme eher nicht zu	1	1,4	1,4	8,3
	Unentschieden	8	11,1	11,1	19,4
	Stimme eher zu	16	22,2	22,2	41,7
	Stimme zu	30	41,7	41,7	83,3
	Stimme vollkommen zu	12	16,7	16,7	100,0
	Gesamt	72	100,0	100,0	

Das Projektteam war räumlich voneinander getrennt

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	9	12,5	12,5	12,5
	Stimme nicht zu	15	20,8	20,8	33,3
	Stimme eher nicht zu	6	8,3	8,3	41,7
	Unentschieden	6	8,3	8,3	50,0
	Stimme eher zu	7	9,7	9,7	59,7
	Stimme zu	11	15,3	15,3	75,0
	Stimme vollkommen zu	18	25,0	25,0	100,0
	Gesamt	72	100,0	100,0	

Das Arbeitsklima innerhalb des Projektteams war sehr gut

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme nicht zu	1	1,4	1,4	1,4
	Stimme eher nicht zu	2	2,8	2,8	4,2
	Unentschieden	2	2,8	2,8	6,9
	Stimme eher zu	8	11,1	11,1	18,1
	Stimme zu	36	50,0	50,0	68,1
	Stimme vollkommen zu	23	31,9	31,9	100,0
	Gesamt	72	100,0	100,0	

Der Kunde/Endnutzer war durchgehend in die Produktentwicklung integriert

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	2	2,8	2,8	2,8
	Stimme nicht zu	5	6,9	6,9	9,7
	Stimme eher nicht zu	11	15,3	15,3	25,0
	Unentschieden	3	4,2	4,2	29,2
	Stimme eher zu	12	16,7	16,7	45,8
	Stimme zu	25	34,7	34,7	80,6
	Stimme vollkommen zu	14	19,4	19,4	100,0
	Gesamt	72	100,0	100,0	

Es gab durchgehend Feedback vom Kunden/Endnutzer

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	3	4,2	4,2	4,2
	Stimme nicht zu	6	8,3	8,3	12,5
	Stimme eher nicht zu	10	13,9	13,9	26,4
	Unentschieden	9	12,5	12,5	38,9
	Stimme eher zu	14	19,4	19,4	58,3
	Stimme zu	20	27,8	27,8	86,1
	Stimme vollkommen zu	10	13,9	13,9	100,0
	Gesamt	72	100,0	100,0	

Die Produktanforderungen wurden vom Kunden priorisiert

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	1	1,4	1,4	1,4
	Stimme nicht zu	8	11,1	11,1	12,5
	Stimme eher nicht zu	8	11,1	11,1	23,6
	Unentschieden	12	16,7	16,7	40,3
	Stimme eher zu	12	16,7	16,7	56,9
	Stimme zu	23	31,9	31,9	88,9
	Stimme vollkommen zu	8	11,1	11,1	100,0
	Gesamt	72	100,0	100,0	

Es gab Zwischenprodukte bevor das Endprodukt fertig war

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	18	25,0	25,0	25,0
	Stimme nicht zu	3	4,2	4,2	29,2
	Stimme eher nicht zu	3	4,2	4,2	33,3
	Unentschieden	5	6,9	6,9	40,3
	Stimme eher zu	7	9,7	9,7	50,0
	Stimme zu	22	30,6	30,6	80,6
	Stimme vollkommen zu	14	19,4	19,4	100,0
	Gesamt	72	100,0	100,0	

Die Zwischenprodukte wurden dem Kunden zur Verfügung gestellt

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	21	29,2	29,2	29,2
	Stimme nicht zu	5	6,9	6,9	36,1
	Stimme eher nicht zu	5	6,9	6,9	43,1
	Unentschieden	4	5,6	5,6	48,6
	Stimme eher zu	3	4,2	4,2	52,8
	Stimme zu	23	31,9	31,9	84,7
	Stimme vollkommen zu	11	15,3	15,3	100,0
	Gesamt	72	100,0	100,0	

Die Entwicklung fand zyklisch bzw. iterativ statt

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	17	23,6	23,6	23,6
	Stimme nicht zu	8	11,1	11,1	34,7
	Stimme eher nicht zu	7	9,7	9,7	44,4
	Unentschieden	3	4,2	4,2	48,6
	Stimme eher zu	13	18,1	18,1	66,7
	Stimme zu	15	20,8	20,8	87,5
	Stimme vollkommen zu	9	12,5	12,5	100,0
	Gesamt	72	100,0	100,0	

Das Feedback des Kunden zu den Zwischenprodukten wurde zur Verbesserung des Endproduktes verwendet

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	19	26,4	26,4	26,4
	Stimme nicht zu	1	1,4	1,4	27,8
	Stimme eher nicht zu	2	2,8	2,8	30,6
	Unentschieden	5	6,9	6,9	37,5
	Stimme eher zu	7	9,7	9,7	47,2
	Stimme zu	22	30,6	30,6	77,8
	Stimme vollkommen zu	16	22,2	22,2	100,0
	Gesamt	72	100,0	100,0	

Der gesamte Projektablauf wurde detailliert vor Beginn des Entwicklungsprojektes geplant

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	4	5,6	5,6	5,6
	Stimme nicht zu	2	2,8	2,8	8,3
	Stimme eher nicht zu	8	11,1	11,1	19,4
	Unentschieden	5	6,9	6,9	26,4
	Stimme eher zu	17	23,6	23,6	50,0
	Stimme zu	20	27,8	27,8	77,8
	Stimme vollkommen zu	16	22,2	22,2	100,0
	Gesamt	72	100,0	100,0	

Während der Entwicklung wurde aktiv nach Möglichkeiten zur Verbesserung des Entwicklungsprozesses gesucht

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	4	5,6	5,6	5,6
	Stimme nicht zu	9	12,5	12,5	18,1
	Stimme eher nicht zu	8	11,1	11,1	29,2
	Unentschieden	8	11,1	11,1	40,3
	Stimme eher zu	19	26,4	26,4	66,7
	Stimme zu	16	22,2	22,2	88,9
	Stimme vollkommen zu	8	11,1	11,1	100,0
	Gesamt	72	100,0	100,0	

Die Mitarbeiter mussten einem fix vorgegebenen Entwicklungsprozess folgen

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	3	4,2	4,2	4,2
	Stimme nicht zu	3	4,2	4,2	8,3
	Stimme eher nicht zu	9	12,5	12,5	20,8
	Unentschieden	8	11,1	11,1	31,9
	Stimme eher zu	21	29,2	29,2	61,1
	Stimme zu	18	25,0	25,0	86,1
	Stimme vollkommen zu	10	13,9	13,9	100,0
	Gesamt	72	100,0	100,0	

Es wurde Alles so ausführlich wie möglich dokumentiert

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	2	2,8	2,8	2,8
	Stimme nicht zu	4	5,6	5,6	8,3
	Stimme eher nicht zu	10	13,9	13,9	22,2
	Unentschieden	12	16,7	16,7	38,9
	Stimme eher zu	19	26,4	26,4	65,3
	Stimme zu	17	23,6	23,6	88,9
	Stimme vollkommen zu	8	11,1	11,1	100,0
	Gesamt	72	100,0	100,0	

Bereits entwickelte Module/Technologien/Komponenten wurden wiederverwendet

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme nicht zu	3	4,2	4,2	4,2
	Stimme eher nicht zu	5	6,9	6,9	11,1
	Unentschieden	8	11,1	11,1	22,2
	Stimme eher zu	12	16,7	16,7	38,9
	Stimme zu	26	36,1	36,1	75,0
	Stimme vollkommen zu	18	25,0	25,0	100,0
	Gesamt	72	100,0	100,0	

Das Produkt wurde so einfach als möglich aufgebaut

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme nicht zu	4	5,6	5,6	5,6
	Stimme eher nicht zu	9	12,5	12,5	18,1
	Unentschieden	12	16,7	16,7	34,7
	Stimme eher zu	16	22,2	22,2	56,9
	Stimme zu	20	27,8	27,8	84,7
	Stimme vollkommen zu	11	15,3	15,3	100,0
	Gesamt	72	100,0	100,0	

Die Hardware wurde vor der Software entwickelt

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	4	5,6	5,6	5,6
	Stimme nicht zu	4	5,6	5,6	11,1
	Stimme eher nicht zu	5	6,9	6,9	18,1
	Unentschieden	16	22,2	22,2	40,3
	Stimme eher zu	7	9,7	9,7	50,0
	Stimme zu	21	29,2	29,2	79,2
	Stimme vollkommen zu	15	20,8	20,8	100,0
	Gesamt	72	100,0	100,0	

Das Endprodukt enthielt Funktionen die nicht explizit gefordert waren

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	7	9,7	9,7	9,7
	Stimme nicht zu	11	15,3	15,3	25,0
	Stimme eher nicht zu	7	9,7	9,7	34,7
	Unentschieden	8	11,1	11,1	45,8
	Stimme eher zu	14	19,4	19,4	65,3
	Stimme zu	18	25,0	25,0	90,3
	Stimme vollkommen zu	7	9,7	9,7	100,0
	Gesamt	72	100,0	100,0	

Die Architektur des Produktes lies Veränderungen leicht zu

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme keinesfalls zu	1	1,4	1,4	1,4
	Stimme nicht zu	10	13,9	13,9	15,3
	Stimme eher nicht zu	9	12,5	12,5	27,8
	Unentschieden	8	11,1	11,1	38,9
	Stimme eher zu	19	26,4	26,4	65,3
	Stimme zu	18	25,0	25,0	90,3
	Stimme vollkommen zu	7	9,7	9,7	100,0
	Gesamt	72	100,0	100,0	

Das Produkt wurde vollständig modular aufgebaut

		Häufigkeit	Prozent	Gültige Prozente	Kumulierte Prozente
Gültig	Stimme nicht zu	12	16,7	16,7	16,7
	Stimme eher nicht zu	4	5,6	5,6	22,2
	Unentschieden	13	18,1	18,1	40,3
	Stimme eher zu	14	19,4	19,4	59,7
	Stimme zu	22	30,6	30,6	90,3
	Stimme vollkommen zu	7	9,7	9,7	100,0
	Gesamt	72	100,0	100,0	