

Verstehen, warum sich das Projektmanagement ändern muss

Lernen Sie das agile Projektmanagement kennen

# Kapitel 1

## Projektmanagement modernisieren

**A**giles Projektmanagement ist ein Modell für das Projektmanagement, bei dem der Fokus auf der frühen Lieferung des Geschäftswerts, der kontinuierlichen Verbesserung des Produkts und der Prozesse des Projekts, Flexibilität beim Umfang, Input durch das Team und der Auslieferung sorgsam getesteter Produkte liegt, die den Kundenbedürfnisse entsprechen.

In diesem Kapitel werden Sie sehen, warum seit der Mitte der 1990er-Jahre agile Prozesse als Ansatz für die Softwareentwicklung entstanden sind und warum agile Methoden die Aufmerksamkeit von Projektmanagern, von Kunden, die in die Entwicklung neuer Software investieren, sowie von Führungskräften, deren Unternehmen Softwareentwicklungsabteilungen finanzieren, geweckt haben. Außerdem erläutert dieses Kapitel die Vorzüge, die agile Methoden gegenüber seit langem bestehenden Ansätzen für das Projektmanagement haben.

## Projektmanagement braucht eine gründliche Veränderung

Ein *Projekt* ist ein geplantes Arbeitsprogramm, dessen Fertigstellung eine bestimmte Menge an Zeit, Aufwand und Planung erfordert. Projekte dienen einem Zweck, haben Ziele und müssen oft innerhalb einer festgelegten Zeit und innerhalb eines vorgegebenen Budgets fertiggestellt werden.

Da Sie dieses Buch lesen, ist es wahrscheinlich, dass Sie entweder ein Projektmanager sind oder jemand, der Projekte initiiert, an Projekten arbeitet oder auf andere Weise von Projekten betroffen ist.

Agile Ansätze sind eine Antwort auf die Notwendigkeit, Projektmanagement zu modernisieren. Um verstehen zu können, warum agile Ansätze Projekte revolutionieren, ist es hilfreich, ein wenig über die Geschichte und den Zweck von Projektmanagement zu wissen und darüber, vor welchen Problemen Projekte heutzutage stehen.

## Die Ursprünge modernen Projektmanagements

Projekte gibt es schon seit ewigen Zeiten. Angefangen bei der Chinesischen Mauer bis zu den Maya-Pyramiden in Tikal, von der Erfindung des Buchdrucks bis zur Erfindung des Internets, haben Menschen große und kleine Leistungen in Projekten erbracht.

Als formale Disziplin besteht Projektmanagement, so wie wir es kennen, erst seit der Mitte des 20. Jahrhunderts. In der Zeit des Zweiten Weltkriegs haben Forscher auf der gesamten Welt große Fortschritte im Bau und in der Programmierung von Computern gemacht, vor allem für das Militär der USA. Um diese Projekte fertigzustellen, begannen sie mit der Entwicklung formaler Projektmanagementprozesse. Die ersten Prozesse basierten auf schrittweisen Fertigungsmodellen, die das Militär der USA während des Zweiten Weltkriegs nutzte.

Mitarbeiter aus dem Fachgebiet Datenverarbeitung haben diese schrittweisen Herstellungsprozesse übernommen, da frühe computerbasierte Projekte in erheblichem Maße auf Hardware basierten; die damaligen Computer nahmen ganze Räume in Beschlag. Im Gegensatz dazu war Software nur ein kleiner Bestandteil von Computerprojekten. In den 1940er- und 1950er-Jahren bestanden Computer aus Tausenden von Vakuumröhren, aber weniger als 30 Zeilen Programmcodes. Die Herstellungsprozesse aus den 1940er-Jahren, die bei diesen ersten Computern genutzt wurden, bildeten die Grundlage der Projektmanagementmethode, die unter dem Namen Wasserfall bekannt ist.

Im Jahre 1970 schrieb der Computerwissenschaftler Winston Royce unter dem Titel »Management the Development of Large Software Systems« einen Artikel für die Zeitschrift des IEEE (Institute of Electrical and Electronics Engineers), der die Phasen oder Schritte der Wasserfallmethode beschrieb. Der Begriff Wasserfall wurde erst später geprägt, jedoch sind die Phase, auch wenn sie manchmal anders bezeichnet werden, im Prinzip mit denen identisch, die ursprünglich durch Royce definiert wurden:

1. Anforderungen
2. Entwurf
3. Entwicklung
4. Integration
5. Testen
6. Einführung, Bereitstellung

Bei Wasserfallprojekten machen Sie erst dann mit der nächsten Phase weiter, wenn die vorherige fertiggestellt ist – daher der Name Wasserfall.

Das reine Wasserfallprojektmanagement – die komplette Fertigstellung jedes Schritts, bevor mit dem nächsten Schritt weitergemacht werden kann –, ist genau genommen eine Fehlinterpretation der Empfehlungen von Royce. Royce wies darauf hin, dass dieser Ansatz inhärent risikoreich ist, und empfahl daher, bei der Produkterstellung in Iterationen zu entwickeln und zu testen – Empfehlungen die von vielen Organisationen, die die Wasserfallmethode einsetzen, übersehen wurden.

## Erfolg und Misserfolg von Softwareprojekten

Unglücklicherweise wird auch die Softwareindustrie von der Stagnation bei den traditionellen Projektmanagementansätzen eingeholt. Im Jahre 2015 veröffentlichte die Standish Group, eine Firma, die unter anderem statistische Studien über den IT-Bereich erstellt, die Ergebnisse einer Studie über die Erfolgs- und Misserfolgsquote von 10.000 Softwareprojekten in den USA. Die Ergebnisse dieser Studie zeigen:

- ✓ **29 Prozent der traditionellen Projekte scheiterten komplett.** Die Projekte wurden abgebrochen, bevor sie fertig waren, und führten zu keinem Produktrelease. Diese Projekte lieferten überhaupt keinen Wert.
- ✓ **60 Prozent der traditionellen Projekte standen vor Herausforderungen und wurden teilweise erfolgreich abgeschlossen.** Die Projekte wurden fertiggestellt, jedoch gab es bei Kosten, Dauer, Qualität oder einer Kombination dieser Elemente eine Kluft zwischen den erwarteten und den tatsächlichen Ergebnissen. Die durchschnittliche Differenz zwischen den erwarteten und tatsächlichen Projektergebnissen – im Hinblick auf Zeit, Kosten und nicht gelieferte Features – betrug weit mehr als 100 Prozent.
- ✓ **11 Prozent der Projekte waren erfolgreich.** Die Projekte wurden fertiggestellt und lieferten innerhalb der ursprünglich geplanten Zeit und im erwarteten Budget das erwartete Produkt.

Von den mehreren Hundert Milliarden Dollar, die allein in den USA für Projekte zur Anwendungsentwicklung ausgegeben wurden, wurden Milliarden Dollar in Projekten vergeudet, die niemals auch nur ein einzelnes Stück an Funktionalität bereitstellten.

Die Wasserfallmethode war im Bereich der Softwareentwicklung der gängigste Projektmanagementansatz, bis sie um das Jahr 2008 herum von verbesserten Ansätzen überholt wurde, die auf agilen Techniken basierten.

## Das Problem mit dem Status quo

Die Computertechnologie hat sich seit dem letzten Jahrhundert erheblich geändert. Viele Menschen tragen an ihrem Handgelenk einen Computer, der mehr Rechenleistung und Speicher hat sowie mehr Möglichkeiten bietet als die größten und teuersten Maschinen, die zu der Zeit existierten, als man begann, Wasserfallmethoden einzusetzen.

Gleichzeitig haben sich auch die Menschen geändert, die Computer einsetzen. Anstatt Maschinenungetüme mit minimalen Programmen für wenige Forscher und das Militär zu

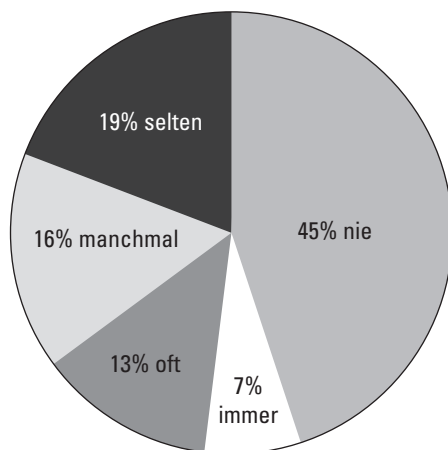
erstellen, wird Hard- und Software heute für die breite Masse produziert. In vielen Ländern nutzt fast jeder – direkt oder indirekt – einen Computer, und zwar jeden Tag. Software steuert unsere Autos, unsere Haushaltsgeräte, unsere Wohnungen; sie sorgt für unsere tägliche Information und unsere tägliche Unterhaltung. Sogar kleine Kinder nutzen Computer – ein zwei Jahre altes Kind ist häufig geschickter mit einem iPhone als seine Großeltern.

Irgendwie haben während dieser Wachstumsphase der Technologie die Prozesse nicht Schritt gehalten. Softwareentwickler nutzen weiterhin Projektmanagementmethoden aus den 1950er-Jahren, und all diese Ansätze wurden von Herstellungsprozessen abgeleitet, die für vor allem aus Hardware bestehenden Computern aus der Mitte des 20. Jahrhunderts gedacht waren.

Heutzutage leiden traditionelle Projekte, die erfolgreich sind, vor allem an einem Problem: *Umfangsaufblähung*, die Aufnahme unnötiger Produktfeatures in ein Projekt.

Denken Sie an die Softwareprodukte, die Sie jeden Tag einsetzen. So hat beispielsweise die Textverarbeitung, in der wir gerade tippen, viele Features und Tools. Obwohl wir jeden Tag mit diesem Programm schreiben, verwenden wir die meiste Zeit nur einige dieser Features. Andere Elemente setzen wir weniger häufig ein. Eine ganze Menge der Tools haben wir noch nie benutzt – und wenn wir es genau bedenken, kennen wir niemand anderen, der sie benutzt hat. Die Features, die nur wenige Anwender nutzen, sind das Ergebnis der Umfangsaufblähung.

Umfangsaufblähung findet sich in jeglicher Software, angefangen von komplexen Enterprise-Anwendungen bis hin zu Websites, die jeder verwenden. Abbildung 1.1 zeigt Daten aus einer Studie der Standish Group, die belegen, wie weit verbreitet Umfangsaufblähung ist. In der Abbildung können Sie erkennen, dass 64 Prozent der gewünschten Features nur selten oder nie benutzt werden.



**Abbildung 1.1:** Tatsächliche Nutzung angeforderter Softwarefeatures

Die Zahlen in Abbildung 1.1 verdeutlichen eine erhebliche Verschwendung an Zeit und Geld. Diese Verschwendung ist ein direktes Ergebnis traditioneller Projektmanagementansätze, die nicht in der Lage sind, Änderungen zu berücksichtigen. Projektmanager und Stakeholder wissen, dass Änderungen in einem laufenden Projekt nicht begrüßt werden. Daher ist die beste Gelegenheit, ein potenziell gewünschtes Feature zu bekommen, der Beginn eines Projekts. Daher fragen sie nach

- ✓ allem, was sie brauchen
- ✓ allem, von dem sie denken, dass sie es brauchen
- ✓ allem, was sie sich wünschen
- ✓ allem, von dem sie denken, dass sie es sich wünschen

Das Ergebnis ist die Aufblähung der Features, die zu den Statistiken in Abbildung 1.1 führt.

Die Probleme, die mit der Verwendung überalterter Management- und Entwicklungsansätze verbunden sind, sind nicht trivial. Diese Probleme verschwenden jedes Jahr Milliarden Dollar. Die Milliarden Dollar, die im Jahre 2015 als Folge gescheiterter Projekte verloren wurden (siehe den Kasten »Erfolg und Misserfolg von Softwareprojekten«), entsprechen weltweit Millionen von Arbeitsplätzen.

Im Verlauf der vergangenen zwei Jahrzehnte haben die Menschen, die an Projekten arbeiten, die zunehmenden Probleme mit traditionellem Projektmanagement erkannt und daran gearbeitet, ein besseres Modell zu erstellen: agiles Projektmanagement.

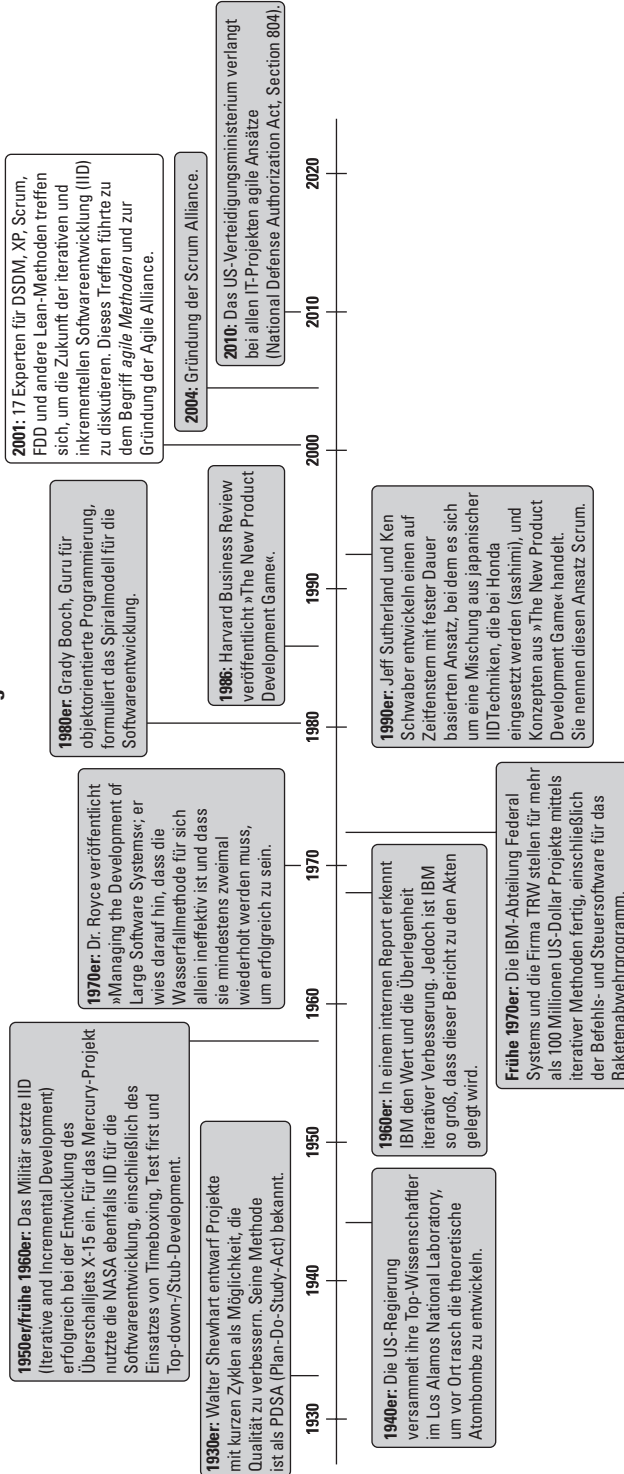
## Einführung in agiles Projektmanagement

Das Saatgut für agile Techniken gibt es schon seit längerer Zeit. Eigentlich sind agile Werte, Prinzipien und Vorgehensweisen einfach nur eine Kodifizierung des gesunden Menschenverstands. Abbildung 1.2 zeigt einen kurzen geschichtlichen Abriss des agilen Projektmanagements, der bis in die 1930er-Jahre zurückreicht, in denen Walter Shewhart den Plan-Do-Study-Act-Ansatz (PDSA) für die Verbesserung der Projektqualität entwickelte.

Im Jahre 1986 veröffentlichten Hirotaka Takeuchi und Ikujiro Nonaka im *Harvard Business Review* einen Artikel mit dem Titel »The New Product Development Game«. Der Artikel von Takeuchi und Nonaka beschreibt eine schnelle, flexible Entwicklungsstrategie, um auf temporeiche Produktansprüche reagieren zu können. Dieser Artikel war der erste, der den Begriff Scrum im Zusammenhang mit Produktentwicklung nutzt. (Scrum bezeichnet ursprünglich eine Spielerformation im Rugby.) Scrum entwickelte sich letztendlich zu einem der gebräuchlichsten Regelwerke für agiles Projektmanagement.

Im Jahre 2001 kam eine Gruppe aus Software- und Projektextperten zusammen, um darüber zu sprechen, was die Gemeinsamkeiten ihrer erfolgreichen Projekte sind. Diese Gruppe entwickelte das Agile Manifest, ein Statement zu Werten für erfolgreiche Softwareentwicklung.

**Geschichtlicher Abriss agiler Techniken**



**Abbildung 1.2:** Zeitstrahl des agilen Projektmanagements

## Manifest für agile Softwareentwicklung

Wir erschließen bessere Wege, Software zu entwickeln, indem wir es selbst tun und anderen dabei helfen.

Durch diese Tätigkeit haben wir diese Werte zu schätzen gelernt:

**Individuen und ihre Interaktionen** stehen über Prozessen und Werkzeugen

**Funktionierende Software** steht über einer umfassenden Dokumentation

**Zusammenarbeit mit dem Kunden** steht über der Vertragsverhandlung

**Reagieren auf Veränderung** steht über dem Befolgen eines Plans

Das heißt, obwohl wir die Werte auf der rechten Seite wichtig finden, schätzen wir die Werte auf der linken Seite höher ein.

© 2001: Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas

*This declaration may be freely copied in any form, but only in its entirety through this notice.*

Diese Experten haben ebenfalls die Prinzipien hinter dem Agilen Manifest entwickelt, 12 Verhaltensweisen, die dazu beitragen, die Werte im Agilen Manifest zu unterstützen. In Kapitel 2 führen wir die Agilen Prinzipien auf und befassen uns ausführlicher mit dem Agilen Manifest.

Agilität, im Hinblick auf Produktentwicklung, ist eine Bezeichnung für Projektmanagementansätze, die sich auf Menschen, Kommunikation, das Produkt und Flexibilität fokussieren. Wenn Sie nach der agilen Methode suchen, dann werden Sie diese nicht finden. Jedoch haben alle agilen Methoden (beispielsweise Crystal), Frameworks (beispielsweise Scrum), Techniken (beispielsweise User-Story-Anforderungen) und Werkzeuge (beispielsweise relative Schätzung) eines gemeinsam: die Einhaltung des Agilen Manifests und der 12 Agilen Prinzipien.

## Wie agile Projekte funktionieren

Agile Ansätze basieren auf einem *empirischen Steuerungsverfahren* – einem Prozess, bei dem Entscheidungen auf den im Projekt beobachtbaren Realitäten gefällt werden. Im Kontext der Methoden zur Softwareentwicklung kann ein empirischer Ansatz sowohl bei neuer Produktentwicklung als auch bei der Verbesserung oder beim Upgraden von Projekten eingesetzt werden. Durch häufige und direkte Überprüfungen der bisherigen Arbeit können Sie, sollte es erforderlich sein, sofort Anpassungen vornehmen. Empirische Steuerung erfordert:

- ✓ **Uneingeschränkte Transparenz:** Jeder, der an einem agilen Projekt beteiligt ist, weiß, was los ist und wie das Projekt fortschreitet.
- ✓ **Regelmäßige Überprüfung:** Die Menschen, die am meisten in das Produkt und den Prozess investieren, evaluieren regelmäßig das Produkt und den Prozess.
- ✓ **Sofortige Anpassung:** Anpassungen werden schnell vorgenommen, um Probleme zu minimieren. Falls eine Überprüfung ergibt, dass sich etwas ändern sollte, wird es unmittelbar geändert.

Um regelmäßige Überprüfung und sofortige Anpassung zu ermöglichen, arbeiten agile Projekte mit Iterationen (kleinere Segmente des Gesamtprojekts). Ein agiles Projekt beinhaltet die gleiche Art von Arbeiten wie in einem traditionellen Wasserfallprojekt: Sie erstellen Anforderungen und Entwürfe, entwickeln das Produkt, dokumentieren es und integrieren das Produkt, falls erforderlich, mit anderen Produkten. Sie testen es, beheben jegliche Probleme und stellen es für die Verwendung bereit. Jedoch statt all diese Schritte für alle Produktfeatures gleichzeitig durchzuführen, wie in einem Wasserfallprojekt, teilen Sie das Projekt in Iterationen auf, die sogenannten *Sprints*.

Abbildung 1.3 zeigt den Unterschied zwischen einem linearen Wasserfallprojekt und einem agilen Projekt.



Das Mischen von traditionellen Projektmanagementmethoden mit agilen Ansätzen entspricht in etwa dieser Frage: »Ich habe einen Porsche 911 Turbo. Jedoch habe ich links vorn ein Wagenrad. Was kann ich tun, damit mein Wagen so schnell ist wie die anderen Porsche?« Die Antwort ist, wie sollte es auch anders sein, dass das nicht geht. Wenn Sie sich einem agilen Ansatz uneingeschränkt verpflichten, ist die Wahrscheinlichkeit für ein erfolgreiches Projekt größer.

## Warum agile Projekte besser funktionieren

Im Verlauf dieses Buchs sehen Sie, warum agile Projekte besser funktionieren als traditionelle Projekte. Agile Projektmanagementansätze führen zu mehr erfolgreichen Projekten. Die Studie der Standish Group, die im Kasten »Erfolg und Misserfolg von Softwareprojekten« erwähnt wird, ergab, dass während 29 Prozent der traditionellen Projekte komplett scheitern, dieser Prozentsatz bei agilen Projekten auf 9 Prozent sinkt. Die Abnahme bei den Misserfolgen agiler Projekte ist ein Ergebnis davon, dass agile Projekte unmittelbar Anpassungen vornehmen, die auf regelmäßiger Überprüfung des Fortschritts und auf der Kundenzufriedenheit basieren.

Hier sehen Sie einige zentrale Bereiche, in denen agile Ansätze den traditionellen Projektmanagementmethoden überlegen sind.

- ✓ **Projekterfolgsquote:** In Kapitel 15 werden Sie sehen, warum das Risiko eines katastrophalen Misserfolgs des Projekts bei agilen Projekten bis fast auf null sinkt. Agile Ansätze der Priorisierung nach Geschäftswert und Risiko stellen frühen Erfolg oder schnelles Scheitern sicher. Agile Ansätze, um während des gesamten Projektverlaufs zu testen, tragen dazu bei, dass Probleme früh erkannt werden und nicht erst dann, wenn bereits eine große Menge an Zeit und Geld investiert wurde.
- ✓ **Umfangsprobleme:** In den Kapiteln 7, 8 und 12 werden Sie sehen, wie agile Ansätze im gesamten Projektverlauf Änderungen ermöglichen und so Umfangsprobleme minimieren. Bei agilen Projekten können Sie neue Anforderungen zu Anfang jedes Sprints ergänzen, ohne dass dies den Entwicklungsfluss beeinträchtigt. Indem Sie zuerst Features mit hoher Priorität vollständig entwickeln, vermeiden Sie es, dass Umfangsprobleme wichtige Funktionalität bedrohen.



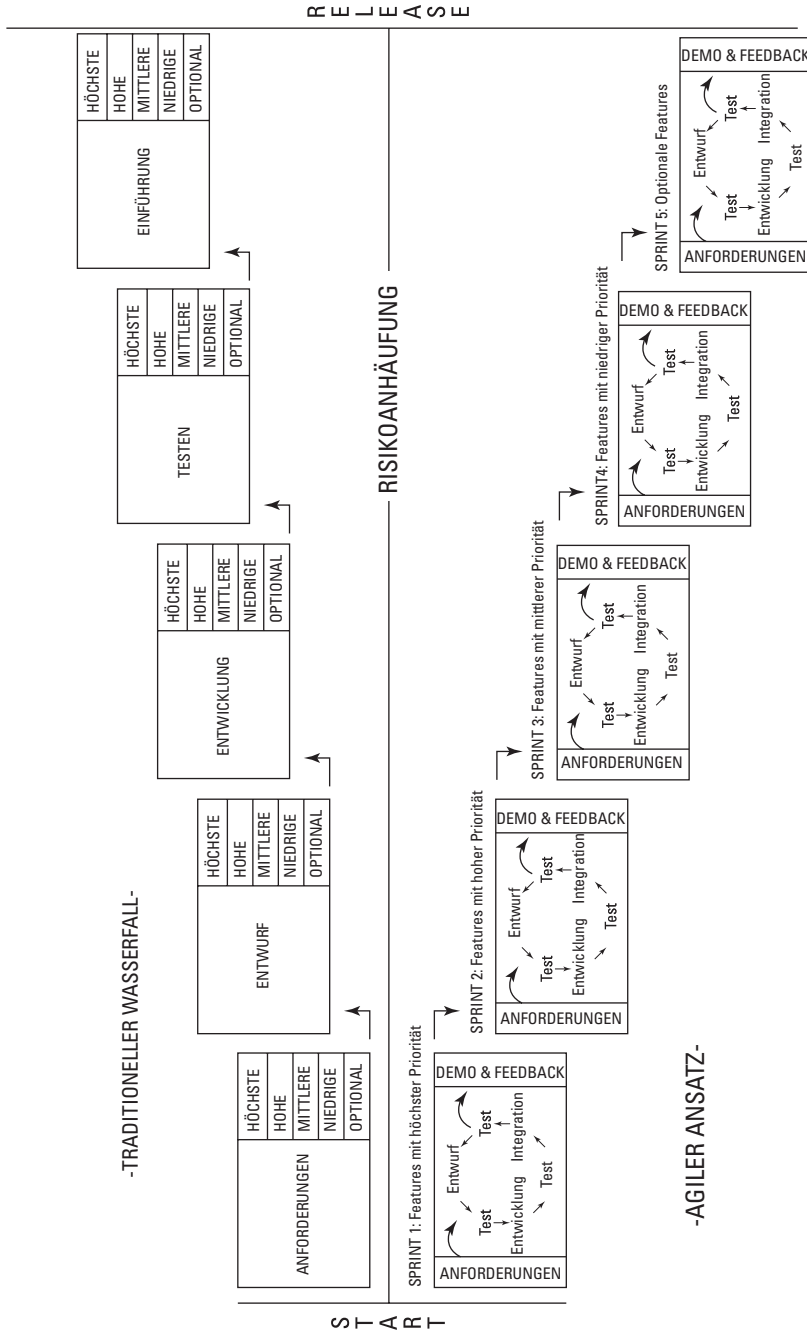


Abbildung 1.3: Wasserfall und agiles Projekt im Vergleich

- ✓ **Überprüfung und Anpassung:** In den Kapiteln 10 und 15 finden Sie Details dazu, wie regelmäßige Überprüfung und Anpassung in agilen Projekten funktionieren. Agile Projektteams können – gestärkt durch regelmäßiges Feedback aus kompletten Entwicklungszyklen mit einsatzfähiger und potenziell marktfähiger Funktionalität – Ihre Prozesse und Ihre Produkte mit jedem Sprint verbessern.

Im Verlauf vieler Kapitel dieses Buchs werden Sie entdecken, wie Sie Kontrolle über das Ergebnis eines agilen Projekts erhalten. Früh und oft testen, Prioritäten anpassen wenn nötig, bessere Kommunikationstechniken verwenden und regelmäßig Produktfunktionalität demonstrieren und veröffentlichen erlauben es Ihnen, Ihre Kontrolle über eine Vielzahl von Faktoren in agilen Projekten zu optimieren.