

Verstehen, warum sich das Projektmanagement bzw. die Produktentwicklung ändern muss

Die modernen Auswirkungen von künstlicher Intelligenz, Diversität und Inklusion betrachten

Entdecken, wie agiles Projektmanagement zu agilem Produktmanagement wird

Die agile Produktentwicklung kennenlernen

Kapitel 1

Das Projektmanagement und die Produktentwicklung modernisieren

Das Wort »agil« ist ein Synonym für »anpassungsfähig«, »flexibel«, »reaktionsschnell« und »leichtgewichtig«. Wenn wir also davon sprechen, als Unternehmen agiler zu werden oder einen agilen Ansatz zu verfolgen, dann beziehen wir uns auf all diese Begriffe. Agilität beschreibt die Fähigkeit eines Unternehmens oder Teams, Probleme, Verbesserungspotenziale und Tendenzen zu erkennen und entsprechend darauf zu reagieren, Pläne an die Bedürfnisse von Kunden anzupassen und mit dem technologischen Wandel Schritt zu halten.

Im Projektmanagement konzentriert sich Agilität auf die frühzeitige Bereitstellung von Kundennutzen sowie die kontinuierliche Verbesserung des zu erstellenden Produkts und der dafür verwendeten Prozesse. Im Kern strebt eine agile Denkweise nach kundenorientierter Flexibilität beim Umfang, Team-Input und die Bereitstellung sorgfältig getesteter, wertvoller Produkte, die die Kundenbedürfnisse widerspiegeln.

In diesem Kapitel erfahren Sie, warum sich agile Ansätze seit Mitte der 1990er-Jahre im Bereich der Softwareentwicklung etabliert haben und warum diese Arbeits- und Denkweise die Aufmerksamkeit von Projektmanagern, Kunden und Führungskräften auf sich gezogen hat. Während Business-Agilität vor allem in der Softwareentwicklung verbreitet ist, finden sich agile Werte, Prinzipien und Techniken (die Sie in den folgenden Kapiteln kennenlernen) in einer Vielzahl von Branchen und Anwendungen wieder – nicht nur im

Softwarebereich. Zudem erläutern wir die Vorteile agiler Ansätze gegenüber herkömmlichen Projektmanagementmethoden.

Die überfällige Verjüngungskur des traditionellen Projektmanagements

Ein *Projekt* ist ein geplantes Vorhaben, dessen Umsetzung eine bestimmte Menge an Zeit, Aufwand und Planung erfordert. Projekte verfolgen in der Regel einen bestimmten Zweck und ein bestimmtes Ziel und müssen oft innerhalb eines festgelegten Zeitrahmens und/oder Budgets fertiggestellt werden.

Da Sie dieses Buch lesen, sind Sie wahrscheinlich entweder Projektmanager oder jemand, der Projekte initiiert, an Projekten arbeitet oder in irgendeiner Weise mit Projekten zu tun hat oder davon betroffen ist.

Agile Ansätze sind eine Antwort auf die Notwendigkeit, das Projektmanagement zu modernisieren. Um zu verstehen, wie und warum agile Ansätze Projekte bzw. die Produktentwicklung revolutionieren, ist es hilfreich, etwas über die Geschichte und den Zweck des Projektmanagements sowie über die Probleme zu wissen, mit denen Projekte und Produktentwicklungsvorhaben heute konfrontiert sind.

Die Ursprünge des modernen Projektmanagements

Projekte gibt es schon seit der Antike – angefangen bei der Chinesischen Mauer bis hin zur Erfindung des Internets. Als formale Disziplin existiert Projektmanagement, so wie wir es heute kennen, jedoch erst seit der Mitte des 20. Jahrhunderts. Während des Zweiten Weltkriegs erzielten Forscher weltweit große Fortschritte beim Bau und in der Programmierung von Computern, vor allem für das US-Militär. Um diese Projekte fertigzustellen, begann man mit der Entwicklung formaler Projektmanagementprozesse. Die ersten Prozesse basierten auf schrittweisen Fertigungsmodellen, die das US-Militär während des Zweiten Weltkriegs nutzte.

In der Computerbranche wurden diese schrittweisen Fertigungsprozesse übernommen, da frühe computerbezogene Projekte stark hardwareabhängig waren und Computer ganze Räume füllten. Im Gegensatz zu heute spielte Software in Computerprojekten eine untergeordnete Rolle. In den 1940er- und 1950er-Jahren verfügten Computer über Tausende von Vakuumröhren, aber über weniger als 30 Zeilen Programmcode. Die Herstellungsprozesse dieser Computer aus den 1940er-Jahren bildeten die Grundlage für das, was heute im Projektmanagement als Wasserfall-Methode bekannt ist.

Im Jahr 1970 veröffentlichte der Computerwissenschaftler Winston Royce für das Institute of Electrical and Electronics Engineers (IEEE) einen Artikel mit dem Titel »Managing the Development of Large Software Systems«, in dem er die Phasen oder Schritte des Wasserfall-Modells beschrieb. Der Begriff Wasserfall-Methode wurde erst später geprägt, jedoch entsprechen die Phasen, auch wenn sie manchmal anders bezeichnet werden, im

Prinzip der ursprünglichen Definition von Royce:

1. Anforderungen definieren
2. Entwurf/Design durchführen
3. System entwickeln
4. Systembestandteile integrieren
5. Gesamtsystem testen
6. System bereitstellen bzw. einführen

Bei Wasserfall-Projekten wird erst dann zur nächsten Phase übergegangen, wenn die vorherige abgeschlossen ist – daher auch die Bezeichnung Wasserfall-Modell.

Das reine Wasserfall-Projektmanagement, bei dem jeder Schritt vollständig abgeschlossen sein muss, bevor mit dem nächsten begonnen wird, ist genau genommen eine Fehlinterpretation der Vorschläge von Royce. Royce wies darauf hin, dass dieser Ansatz von Natur aus risikobehaftet ist, und empfahl daher, Produkte in Iterationen zu entwickeln und zu testen. Leider wurde diese Empfehlung von vielen Unternehmen, die das Wasserfall-Modell einführen, übersehen oder ignoriert.



Die Wasserfall-Methode war im Bereich der Softwareentwicklung der gängigste Projektmanagement-Ansatz, bis sie um das Jahr 2008 herum von verbesserten, auf agilen Techniken basierenden Ansätzen überholt wurde.

Das Problem mit dem Status quo

Die Computertechnologie hat sich seit dem letzten Jahrhundert natürlich immens verändert, ähnlich wie die Menschen, die sie entwickeln und nutzen. Anstelle riesiger Maschinen mit minimalen Programmen für wenige Forscher und das Militär wird heute sowohl Hardware als auch Software für die breite Masse produziert – und das in minimalem Format.

Viele Menschen tragen heute Computer am Handgelenk, die über mehr Rechenleistung, Speicher und Funktionen verfügen als die größten und teuersten Modelle zur Zeit der Einführung der Wasserfall-Methode. Weltweit nutzt heute fast jeder täglich – direkt oder indirekt – einen Computer. Software steuert unsere Autos, unsere Haushaltsgeräte, teilweise sogar unsere Häuser und Wohnungen, und sie liefert uns täglich Informationen und Unterhaltung. Selbst Kleinkinder nutzen heute smarte Geräte, wie Smartphones oder Tablets, und entsprechende Medien. Kein Wunder, ist es doch mittlerweile mithilfe von generativer künstlicher Intelligenz auch möglich, mobile Apps sowie professionelle Grafiken und Videos ganz einfach selbst zu erstellen. Die Nachfrage nach immer neuen, besseren Produkten ist konstant hoch und entwickelt sich rasant.

Aus irgendeinem Grund haben die Prozesse mit dem rasanten technologischen Wachstum jedoch nicht Schritt gehalten. Softwareentwickler verwenden noch immer Projektmanagementmethoden aus den 1950er-Jahren, und all diese Ansätze wurden aus Herstellungsprozessen abgeleitet, die für die hardwareintensiven Computer aus der Mitte des 20. Jahrhunderts entwickelt wurden.

Heutzutage leiden selbst erfolgreiche traditionelle Projekte vor allem unter einem Problem: *Umfangsaufblähung* (auch *Scope Bloat* genannt), also der Aufnahme unnötiger Produktfeatures.

Denken Sie an die Softwareprodukte, die Sie täglich nutzen. So bietet beispielsweise das Textverarbeitungsprogramm, mit dem wir diesen Text getippt haben, zahlreiche Funktionen und Tools. Doch obwohl wir alle ein solches Programm fast täglich nutzen, verwenden wir die meiste Zeit nur einige wenige der enthaltenen Features. Manche Tools verwenden wir nur selten, viele andere haben wir vielleicht noch nie genutzt – wie vermutlich die meisten Otto-Normal-Verbraucher. Solche Features, die nur von wenigen Anwendern genutzt, sind das Ergebnis von Umfangsaufblähung.

Umfangsaufblähung oder Scope Bloat findet sich in jeglicher Art von Software, angefangen von komplexen Unternehmensanwendungen bis hin zu Websites, die jeder nutzt. Abbildung 1.1 zeigt Daten aus einer Studie der Standish Group, die verdeutlichen, wie weit verbreitet das Phänomen der Umfangsaufblähung ist. Wie Sie sehen, werden ganze 80 Prozent der angeforderten Features nur selten oder so gut wie nie benutzt.

Sie verdeutlicht die Verschwendung von Zeit und Geld durch den Einsatz traditioneller Projektmanagement-Ansätze. Traditionelle Projektmanager und Stakeholder sehen Änderungen in einem laufenden Projekt als etwas Unerwünschtes und daher den Beginn eines Projekts als die beste Gelegenheit, potenzielle Features unterzubringen. Daher fragen sie gleich zu Beginn nicht nur nach Features, die sie wirklich benötigen oder sich wünschen, sondern auch nach solchen, die sie später »möglicherweise« noch brauchen oder sich wünschen »könnten«.

Das Ergebnis ist eine Funktionsflut, die wiederum zu den Statistiken in Abbildung 1.1 führt.

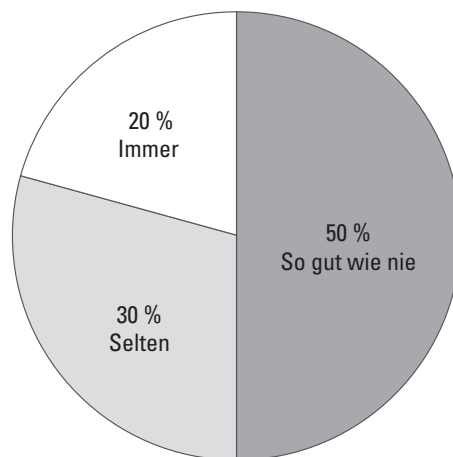


Abbildung 1.1: Tatsächliche Nutzung angeforderter Softwarefeatures

Die mit veralteten Management- und Entwicklungsansätzen einhergehenden Probleme sind nicht trivial. Sie verursachen jährlich Milliardenkosten. Kein Wunder also, dass jedes Jahr Milliarden von Dollar bei fehlgeschlagenen Wasserfall-Projekten verloren gehen (siehe

den Kasten »Erfolg und Misserfolg von Softwareprojekten«), insbesondere wenn Erfolg bedeutet, dass jedes Feature pünktlich, im Rahmen des Budgets und in perfekter Qualität geliefert werden muss.

In den letzten drei Jahrzehnten haben Projektbeteiligte die zunehmenden Probleme traditioneller Projektmanagement-Ansätze erkannt und daran gearbeitet, ein besseres Modell zu entwickeln: das agile Projektmanagement.

Moderne Auswirkungen künstlicher Intelligenz

Neue Technologien beeinflussen das Projektmanagement und die Produktentwicklung, wobei die *künstliche Intelligenz*, kurz KI, einen der bislang bedeutendsten Einflüsse verzeichnet. KI steht für die Theorie und Entwicklung von Computersystemen, die Aufgaben ausführen können, für die normalerweise menschliche Intelligenz erforderlich ist, wie beispielsweise visuelle Wahrnehmung, Spracherkennung, Entscheidungsfindung und Sprachübersetzung. KI verändert das Projektmanagement, indem sie den Verwaltungsaufwand für Teams reduziert und es ihnen ermöglicht, ihre Anstrengungen, ihre Zeit und ihr Denken neu auszurichten, um kreativer, lateraler und integrativer zu agieren.

KI bringt zwar viele Vorteile mit sich, birgt jedoch auch einige Gefahren, die gewisse Vorsichtsmaßnahmen erfordern. Denn KI ist nicht dazu geeignet, alles zu ersetzen, was Menschen leisten können. Zwar kann sie problemlos Dinge wie Terminplanung, Aufgabenverfolgung und Datenanalyse übernehmen, wenn es jedoch um riskante Entscheidungen von hoher Tragweite geht, ist nach wie vor die menschliche Intuition aufgrund von Erfahrung unerlässlich. Die Vor- und Nachteile des Einsatzes von KI sowie die entsprechenden Vorsichtsmaßnahmen behandeln wir in diesem Buch.

Moderne Auswirkungen von Vielfalt und Inklusion

Schnell voranschreitende Technologien wie KI sowie die zunehmende Mobilität und der bessere Zugang zu Informationen haben uns vielfältigere Erfahrungen und soziale Möglichkeiten eröffnet. Gleichzeitig haben sie zu mehr Diversität und der Notwendigkeit geführt, unterschiedliche Meinungen, Hintergründe, Erfahrungen, Kulturen, Stile und Perspektiven einzubeziehen.

Obwohl wir seit Jahrhunderten in einer globalen Wirtschaft leben, setzten sich Produktentwicklungsteams in Unternehmen traditionell aus Menschen zusammen, die aus bestimmten geografischen Regionen oder bestimmten beruflichen Netzwerken stammten. Heute erstreckt sich die Lösung von Produktproblemen über unterschiedliche Disziplinen und ganze Kontinente hinweg. Mit anderen Worten: Wir arbeiten nicht mehr tagtäglich nur mit Menschen zusammen, die so aussehen und denken wie wir selbst.

Je mehr wir diese Vielfalt nutzen, indem wir Menschen mit unterschiedlichen Perspektiven effektiv einbinden, desto leichter können wir uns auf die Bedürfnisse unserer Kunden einstellen. Auch für den Erfolg von Teams und Unternehmen ist es entscheidend, sich diese Unterschiede zunutze zu machen.

ERFOLG UND MISSERFOLG VON SOFTWAREPROJEKTEN

Die Stagnation bei den traditionellen Projektmanagement-Ansätzen macht sich auch in der Softwarebranche bemerkbar. Im Jahr 2020 veröffentlichte die Standish Group, ein Unternehmen, das unter anderem statistische Studien über den IT-Bereich erstellt, die Ergebnisse einer Studie zur Erfolgs- und Misserfolgsquote von 50.000 Softwareprojekten in den USA und weltweit:

- ✓ *66 Prozent aller Technologieprojekte scheitern teilweise oder vollständig.* Obwohl größere Projekte anfälliger für Misserfolge sind, scheitern selbst kleinere Projekte in 10 Prozent der Fälle.
- ✓ *31 Prozent der US-Projekte scheiterten vollständig.* Die Projekte wurden vor Abschluss abgebrochen und führten zu keinem Ergebnis. Sofern keine wichtigen Erkenntnisse für eine erfolgreiche Neuausrichtung des Unternehmens gewonnen wurden, lieferten diese Projekte keinen Mehrwert.
- ✓ *53 Prozent der US-Projekte standen vor Herausforderungen und wurden nur teilweise fertiggestellt.* Zwar wurden die Projekte abgeschlossen, 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 den tatsächlichen Projektergebnissen – im Hinblick auf Zeit, Kosten und nicht gelieferte Features – lag deutlich über 100 Prozent.
- ✓ *Lediglich 16 Prozent der untersuchten US-Projekte waren erfolgreich.* Die Projekte wurden abgeschlossen und lieferten das erwartete Produkt innerhalb des ursprünglich geplanten Zeitrahmens und Budgets.

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

Einführung in agiles Projektmanagement

Die Grundlagen agiler Techniken existieren schon seit langem, denn agile Werte, Prinzipien und Vorgehensweisen sind im Grunde nichts anderes als eine Kodifizierung des gesunden Menschenverstands. Abbildung 1.2 zeigt einen kurzen geschichtlichen Abriss des agilen Projektmanagements, der bis in die 1930er-Jahre zurückreicht, als Walter Shewhart den *Plan-Do-Study-Act*-Ansatz (PDSA) für die Verbesserung der Projektqualität entwickelte – und sogar bis ins 17. Jahrhundert zu Sir Francis Bacon.

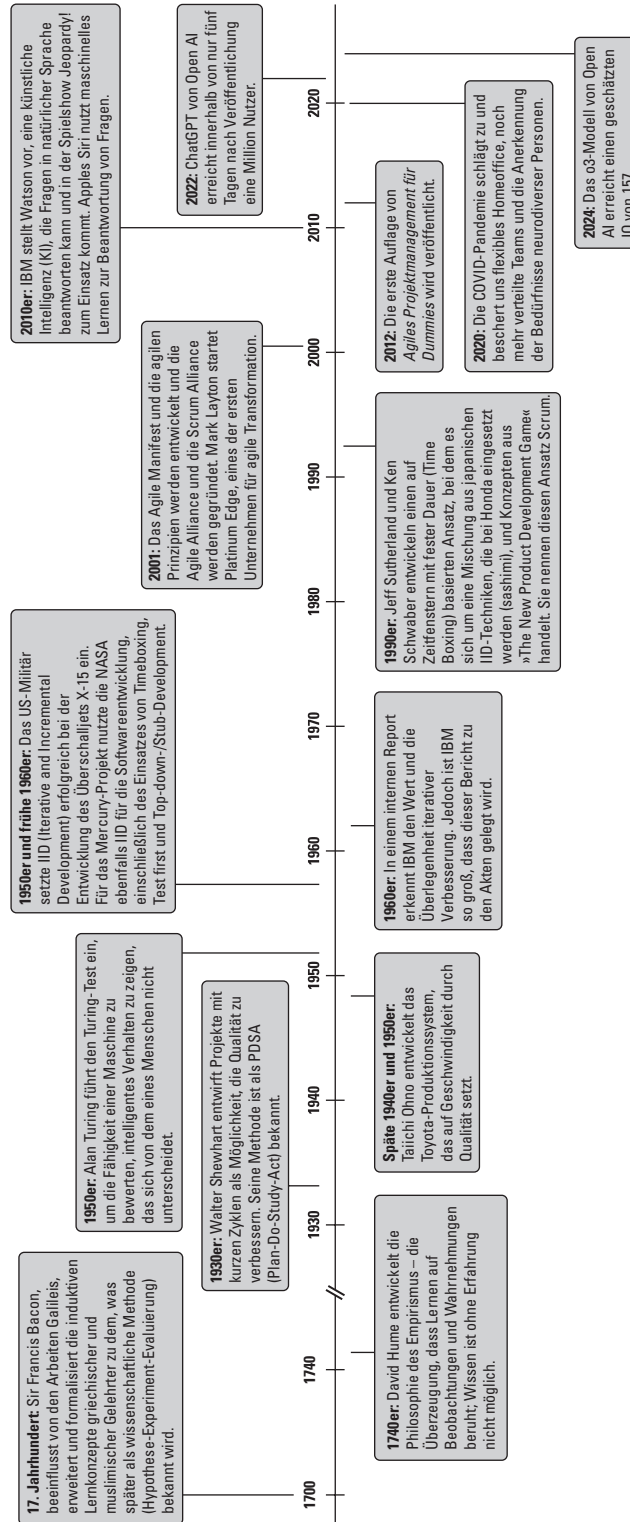


Abbildung 1.2: Zeitstrahl des agilen Projektmanagements

Im Jahr 1986 veröffentlichten Hirotaka Takeuchi und Ikujiro Nonaka im *Harvard Business Review* einen Artikel mit dem Titel »The New New Product Development Game«. Darin beschrieben die beiden Wissenschaftler eine schnelle und flexible Entwicklungsstrategie, um den rasanten Produkthanforderungen gerecht zu werden. In diesem Artikel wurde der Begriff Scrum erstmals mit der Produktentwicklung verknüpft. (*Scrum* bezeichnet ursprünglich eine Spielerformation im Rugby.) Scrum entwickelte sich schließlich zu einem der gebräuchlichsten Frameworks für agiles Projektmanagement bzw. für die Bereitstellung von Mehrwert für den Kunden.

Im Jahr 2001 traf sich eine Gruppe von Software- und Projektemperten, um die Gemeinsamkeiten ihrer erfolgreichen Experimente in der Projektmodernisierung zu besprechen. Diese Gruppe entwickelte das Manifest für Agile Softwareentwicklung (allgemein bekannt als Agiles Manifest), eine Werteerklärung für erfolgreiche Softwareentwicklung:

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 auch die Prinzipien hinter dem Agilen Manifest entwickelt, zwölf Verfahrensweisen, die Werte des Agilen Manifests zu unterstützen. Ausführlichere Informationen zum Agilen Manifest und zu den Agilen Prinzipien finden Sie in Kapitel 2.

Im Kontext der Produktentwicklung steht der Begriff Agilität für Projektmanagementansätze, die sich auf Menschen, Kommunikation, das Produkt und Flexibilität konzentrieren. Wenn Sie nach *der* agilen Methode suchen, werden Sie diese nicht finden. Jedoch haben einige agilen Methoden (z. B. Extreme Programming), Frameworks (z. B. Scrum), Techniken (z. B. User Storys) und Werkzeuge (z. B. physische Taskboards) eines gemeinsam: die Einhaltung des Agilen Manifests und der 12 Agilen Prinzipien.



Wenn Sie an Agilität denken, denken Sie an gesundes Arbeiten. Gesunde Unternehmen und Teams sind agil, widerstandsfähig, flexibel, anpassungsfähig und reaktionsschnell.

Wie agile Projekte funktionieren

Agile Ansätze basieren auf einer Methode der *empirischen Prozesssteuerung* – dem Prinzip, Entscheidungen auf Grundlage der im Projekt beobachteten Realitäten zu treffen. Im Kontext der Softwareentwicklung kann ein empirischer Ansatz sowohl bei der Entwicklung neuer Produkte als auch bei Verbesserungs- und Upgrade-Projekten effektiv sein. Durch regelmäßige und unmittelbare Überprüfung der bisherigen Arbeit können Sie bei Bedarf sofort Anpassungen vornehmen. Empirische Steuerung erfordert Folgendes:

- ✓ **Uneingeschränkte Transparenz:** Alle Projektbeteiligten wissen, was los ist und wie das Projekt voranschreitet.
- ✓ **Regelmäßige Überprüfung:** Die Personen, die am meisten in das Produkt und den Prozess involviert sind, evaluieren regelmäßig das Produkt und den Prozess.
- ✓ **Sofortige Anpassung:** Anpassungen werden schnell vorgenommen, um Probleme zu minimieren. Ergibt eine Überprüfung, dass etwas geändert werden sollte, wird es sofort geändert.

Um regelmäßige Überprüfungen und sofortige Anpassungen zu ermöglichen, arbeiten agile Projekte mit sogenannten *Iterationen* (kleineren Segmenten des Gesamtprojekts). Gemäß den Werten und Prinzipien des Agilen Manifests umfassen agile Projekte bzw. Produktentwicklungen dieselben Arbeitsschritte wie ein traditionelles Wasserfall-Projekt: Sie erstellen Anforderungen und Designs, entwickeln Produktfunktionen, dokumentieren, was getan wurde und warum, und integrieren kontinuierlich neue Funktionen. Sie testen das Produkt, beheben etwaige Probleme und stellen es bereit. Anstatt jedoch wie bei einem Wasserfall-Projekt jeweils einen dieser Schritte für alle Produktfeatures abzuschließen, führen Sie alle Arbeiten an einem kleinen Funktionsumfang durch, bis dieser abgeschlossen ist, und überprüfen die Auswirkungen sofort, bevor Sie zum nächsten Feature übergehen. Ein gängiger Ansatz zur Anwendung agiler Werte und Prinzipien ist das Arbeiten in relativ kurzen, iterativen Zyklen (sogenannten *Sprints*), um das Projekt abzuwickeln.

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

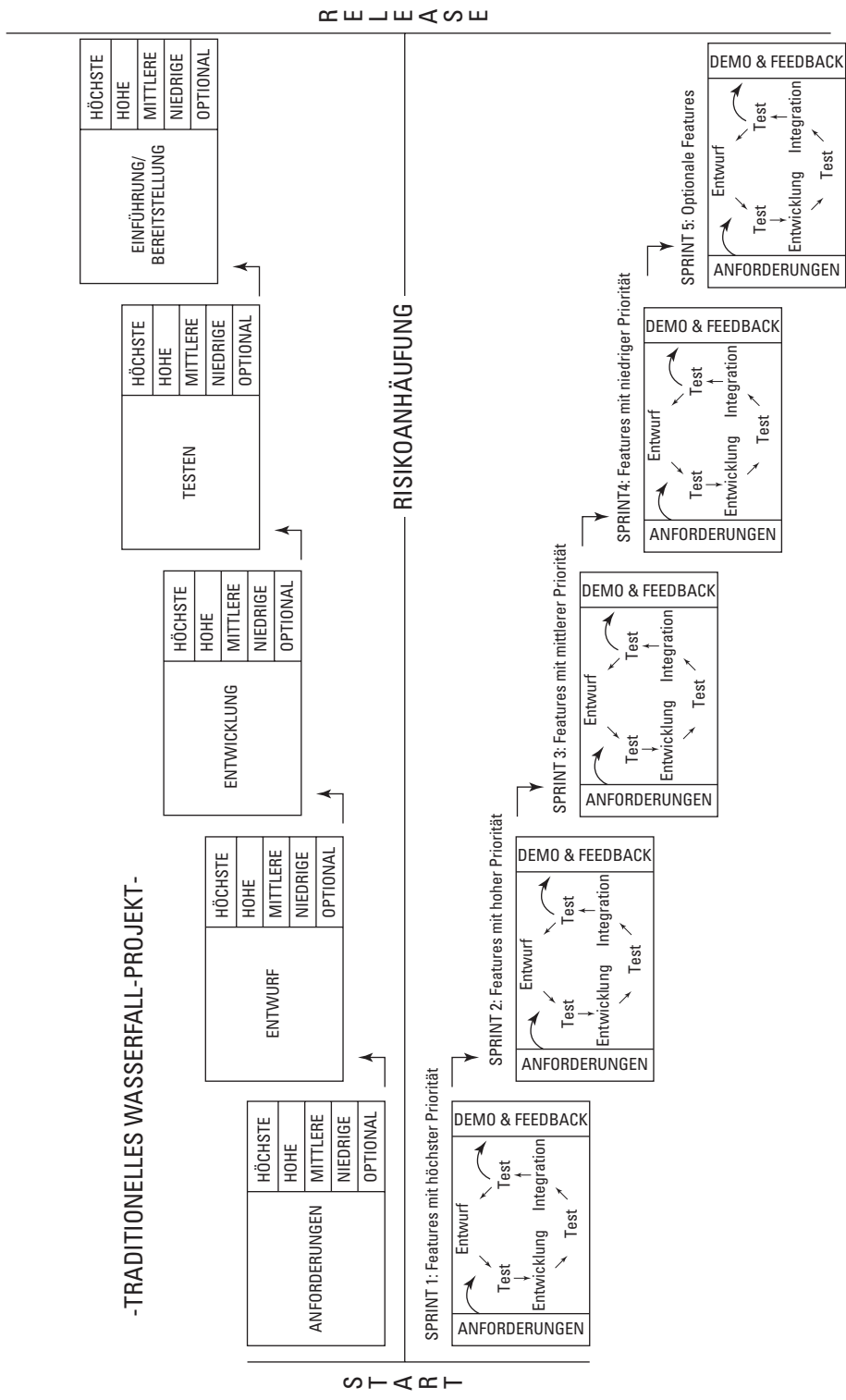


Abbildung 1.3: Wasserfall und agiles Projekt im Vergleich



Die Vermischung traditioneller Projektmanagementmethoden mit agilen Ansätzen entspricht in etwa folgendem Szenario: Stellen Sie sich vor, Sie fahren einen Porsche 911 Turbo, haben jedoch links vorne ein Wagenrad statt eines entsprechenden High-End-Reifens montiert. Und nun fragen Sie sich: »Was kann ich tun, damit mein Wagen ebenso schnell und effizient fährt wie die anderen Porsche?« Die Antwort lautet – wie sollte es auch anders sein –, dass das nicht möglich ist. Mit anderen Worten: Ihre Erfolgchancen erhöhen sich, wenn Sie sich uneingeschränkt auf agile Ansätze einlassen.

Agiles Projektmanagement wird zu agilem Produktmanagement

Traditionelle Projekte sind per Definition temporäre, ausschließlich auf die Entwicklung ausgerichtete Teamarbeiten, mit dem Ziel, bestimmte, in einem Business Case prognostizierte Vorteile zu erzielen. Zu Beginn eines Projekts, wenn noch am wenigsten bekannt ist, werden Budgets, Zeitpläne und Erwartungen festgelegt. Nach Abschluss des Projekts werden die Teams aufgelöst, und die Betreuung der Kunden und des Produkts bleibt unbekannt operativen Teams überlassen. Sind weitere Arbeiten erforderlich, werden neue Teammitglieder einem neuen Projekt zugeteilt, in dem sie sich erneut mit der Produktarchitektur vertraut machen müssen. Projekte enden in der Regel, sobald die Ergebnisse in die Produktion übernommen werden. Die Aufgabe, die Auswirkungen auf das Geschäft zu unterstützen und zu bewerten, obliegt dann anderen.

Heute werden Produkte als langfristige, wertschöpfende Vermögenswerte betrachtet. Dies erfordert nahezu permanente Teams, die Produkte iterativ ausarbeiten, entwerfen, erstellen, testen, integrieren, dokumentieren und sogar betreuen, bis die Geschäftsergebnisse erreicht sind. Ein leistungsstarkes Team überprüft und passt sich kontinuierlich an, bis die Probleme des Kunden gelöst sind, wobei das hart erarbeitete Wissen im Team verbleibt. Das Team und der Kunde arbeiten zusammen, um über die schriftlichen Spezifikationen hinaus Mehrwert zu schaffen.

Wir betrachten einen projektorientierten Ansatz zunehmend als Hindernis für eine frühzeitige und häufige Bereitstellung von Kundennutzen. Ein agiler Produktentwicklungsansatz gewährleistet die Bereitstellung des größtmöglichen Kundennutzens innerhalb des verfügbaren Zeitrahmens und Budgets. Unternehmen erzielen den größten Nutzen, wenn sie anhand der folgenden Formel bestimmen, wann es Zeit ist, Prioritäten zu verschieben:

$$AC + OC > V, \text{ d. h. tatsächliche Kosten} + \text{Opportunitätskosten} > \text{Wert}$$

Wenn die tatsächlichen Kosten (AC) für die Arbeit an den verbleibenden Anforderungen eines Produkts zuzüglich der Opportunitätskosten (OC) für die Nichtbearbeitung einer anderen Investitionsmöglichkeit den erwarteten Wert (V) der Umsetzung dieser verbleibenden Produkthanforderungen übersteigen, wechselt das Team zur Entwicklung der wertvolleren Investitionsmöglichkeit.



Bei einem agilen Ansatz zur Produktentwicklung gehen Ihnen nicht die Zeit oder das Geld aus, sondern vielmehr der investierbare Wert.

Unterschiede zwischen Projektmanagement und Produktentwicklung

Ein Projekt entwickelt per Definition ein Produkt, dennoch gibt es drei wesentliche Unterschiede zwischen Projektmanagement und Produktentwicklung:

- ✓ Produkte profitieren am meisten von stabilen, langlebigen und sogar permanenten Teams.
- ✓ Produkte können nicht nur kurzfristige, sondern auch langfristige Vermögenswerte sein. Aktive Produkte sind nie wirklich fertig, da sie Wartung und Verbesserungen erfordern.
- ✓ Produkte sind Teil eines Portfolios, das darauf ausgelegt ist, den Wert jenseits schriftlicher Spezifikationen zu maximieren.

Permanentes Team statt temporäres Team

Langlebige Produkte werden am besten von langlebigen, diversen und sogar permanenten Teams entwickelt und gewartet. Je länger ein Team gemeinsam iterativ an einer neuen Architektur arbeitet und seine Fähigkeiten und Leistungsfähigkeit erweitert, desto besser versteht es die Kunden und desto vorhersehbarer wird es. Projektorientierte Teams arbeiten für einen bestimmten Zeitraum zusammen und widmen sich danach neuen Aufgaben. Die am Ende eines Projekts gewonnenen Erkenntnisse lassen sich möglicherweise nicht einmal auf das nächste Projekt übertragen, da sich der Kontext in Bezug auf Personen, Technologien und Kunden höchstwahrscheinlich geändert hat. Stabile, diverse und permanente Teams ermöglichen Transparenz, Überprüfung und Anpassung (die zuvor erwähnte Methode der empirischen Prozesskontrolle).



Permanent bedeutet nicht, dass sich agile Produktteams nicht verändern oder Karrierewünsche unterdrückt werden. Personelle Veränderungen im Team sind jedoch eher die Ausnahme als die Regel. Teammitglieder – insbesondere diejenigen, die aufgrund ihrer wachsenden Fähigkeiten an Wert gewinnen – erhalten Möglichkeiten zur beruflichen Weiterentwicklung. Idealerweise verhalten sich permanente Teammitglieder eher wie eine Familie als wie eine temporäre Gruppe, die nur für ein bestimmtes Projekt zusammengestellt wurde. Ihre kulturellen, erfahrungsbezogenen und fachlichen Unterschiede stärken die Leistung und die Ergebnisse des Teams.

Produkte als langfristige Vermögenswerte statt als Projektleistungen

Produktentwicklung ist riskant. Unsicherheit lauert an jeder Ecke. Aber gerade diese Unsicherheit macht agile Produktentwicklung so ideal! Während bei traditionellen Projekten innerhalb eines festgelegten Zeitrahmens bestimmte Systemleistungen erbracht werden müssen, reduziert die agile Produktentwicklung die Unsicherheit iterativ, indem sie nutzbare, voll funktionsfähige Werte schafft und während der gesamten Entwicklung Feedback sammelt und umsetzt. So wird das Produkt zu einem kundenorientierten, problemlösenden Vermögenswert. Aktive Produkte sind niemals fertig, da sie gewartet und verbessert werden müssen.

Insbesondere bei den heutigen Investitionsstrategien sorgen Investitionen in Zeit, Geld und Personal dafür, dass Produkte zu abschreibungsfähigen Vermögenswerten werden, die nicht nur den Umsatz, sondern auch die Kosteneinsparungen verbessern. Betrachtet man die Produktentwicklung als Wertschöpfung und nicht als Kostenfaktor, verändert sich die Perspektive aller Beteiligten. Die kontinuierliche Bereitstellung von Kundennutzen durch agile Produktentwicklung erhöht die Wahrscheinlichkeit, zusätzliche Finanzmittel zu erhalten.



Investitionsausgaben, auch *CapEx* genannt, sind Mittel, die ein Unternehmen für den Erwerb, die Modernisierung und die Instandhaltung von Sachanlagen wie Grundstücken, Gebäuden, Industrieanlagen, Technologie und Ausrüstung verwendet. Unternehmen nutzen CapEx häufig, um neue Projekte oder Investitionen zu realisieren.

Wert statt Spezifikationen

Frühzeitiges Scheitern ist ein wesentlicher Bestandteil der Agilität. Agile Teams sind bereit, Risiken einzugehen, um Werte für ihre Kunden zu schaffen. Wie Wissenschaftler erstellen sie eine Hypothese, testen diese in der Praxis, bewerten die Ergebnisse und passen die Hypothese dann an, um sie erneut zu testen. Sie wiederholen diesen Prozess immer wieder und passen das Produkt mit jeder Iteration besser an die Bedürfnisse der Kunden an. Teams tauschen somit Unmengen an dokumentierten Spezifikationen gegen reales Kunden-Feedback ein. Bei der agilen Produktentwicklung werden die Prioritäten für die Funktionalität von den Personen festgelegt, die mit dem zu lösenden Problem am besten vertraut sind.

Warum agile Produktentwicklung besser funktioniert

In diesem Buch erfahren Sie, warum agile Projekte und Produktentwicklungen besser funktionieren als traditionelle Ansätze. Agile Ansätze können zu erfolgreicherer Produkten führen. Die im Kasten »Erfolg und Misserfolg von Softwareprojekten« erwähnte Studie der Standish Group ergab, dass 31 Prozent der traditionellen Projekte komplett scheiterten, während dieser Prozentsatz bei agilen Methoden deutlich geringer ausfällt.

Der Rückgang der Misserfolgsquote bei agilen Projekten bzw. bei der agilen Produktentwicklung ist darauf zurückzuführen, dass funktionsübergreifende, vielfältige Teams auf der Grundlage regelmäßiger Überprüfungen des Fortschritts und der Kundenzufriedenheit sofortige Anpassungen vornehmen.

Hier sind einige zentrale Bereiche, in denen agile Ansätze den traditionellen Methoden überlegen sind:

- ✓ **Projekterfolgsquote:** In Kapitel 17 erfahren Sie, wie und warum das Risiko eines totalen Projektversagens mit agilen Ansätzen nahezu auf null sinkt. Agile Ansätze, die nach Geschäftswert priorisieren, gewährleisten einen frühzeitigen Erfolg oder Misserfolg. Agile Ansätze, bei denen während der gesamten Produktentwicklung getestet wird, helfen sicherzustellen, dass Probleme frühzeitig erkannt werden und nicht erst dann, wenn bereits große Mengen an Zeit und Geld investiert wurden.
- ✓ **Umfangsprobleme:** In den Kapiteln 9, 10 und 14 erfahren Sie, wie agile Ansätze Änderungen im gesamten Projekt- bzw. Produktentwicklungsverlauf berücksichtigen und so dazu beitragen, Umfangsprobleme (wie z. B. eine schleichende Umfangserweiterung, auch *Scope Creep* genannt) zu minimieren. Nach den agilen Prinzipien können Sie zu Beginn jedes Sprints neue Anforderungen hinzufügen, ohne den Entwicklungsfluss zu unterbrechen. Indem Sie zuerst Features mit hoher Priorität vollständig entwickeln, verhindern Sie, dass Umfangsprobleme wichtige Funktionalität gefährden.
- ✓ **Überprüfung und Anpassung:** In den Kapiteln 12 und 16 erfahren Sie, wie regelmäßige Überprüfung und Anpassung in agilen Projekten bzw. in der agilen Produktentwicklung funktionieren. Durch häufiges Feedback aus vollständigen Entwicklungszyklen in Verbindung mit einsatz- und auslieferfähiger Funktionalität sind agile Teams in der Lage, ihre Prozesse und Produkte mit jedem Sprint zu verbessern.

Im Verlauf dieses Buchs erfahren Sie, wie Sie mithilfe von Business-Agilität die Kontrolle über Ihre agilen Projekt- und Produktergebnisse erlangen. Indem Sie frühzeitig testen, bedarfsgerechte Anpassungen der Prioritäten vornehmen, bessere Kommunikationstechniken einsetzen und regelmäßige Produktfunktionalität demonstrieren und veröffentlichen, können Sie Ihre Fähigkeit, frühzeitig und häufig Mehrwert zu liefern, über eine Vielzahl von Faktoren hinweg optimieren.