

Der agile Ansatz und das Wasserfallmodell sind sich ähnlicher als gemeinhin angenommen

# Wasserfallprojektmanagement vs. agile Methoden

👤 [Markus Ernten](#) ⌚ 2 Jahren online 💬 [Kommentare](#)

**Agile Methoden werden in der Softwareentwicklung meist als völlig gegensätzlich zur klassischen Wasserfallmethode angesehen.**

**Tatsächlich gibt es aber bei beiden Ansätzen auch grundsätzliche Übereinstimmungen. Wo genau liegen die Gemeinsamkeiten und Unterschiede und wie wirken sie sich konkret auf das Projektmanagement und auf die Art der Zusammenarbeit aus?**



Zunächst ist festzustellen, dass beide Ansätze davon ausgehen, dass die drei Faktoren *Kosten*, *Zeitplan* und *Umfang* das Projekt bestimmen.

Unterschiede gibt es dagegen in der Vorgehensweise. Nach dem Wasserfallmodell ist zunächst der Projektumfang zu bestimmen, um anschließend eine Zeit- und Kostenplanung vornehmen zu können. Feedback wird mit „Nachbesserung“ gleichgesetzt – ein Zusatzaufwand, der sich durch eine bessere Planung vermeiden lässt. Der agile Ansatz geht dagegen davon aus, dass der Umfang eines Projekts stets variabel ist. Hierbei bildet Feedback einen wesentlichen, festen Bestandteil des Planungsprozesses, der während der Umsetzungsphase andauert.

Ein weiterer Unterschied: Beim agilen Ansatz entfällt die klassische Weisungshierarchie des Wasserfallmodells. Stattdessen werden Zusammenarbeit und fördernde Unterstützung – auch als „Servant Leadership“ bekannt – gepflegt. Allerdings sind bestimmte Voraussetzungen

nötig, damit die althergebrachte hierarchische Befehls- und Kontrollorganisation überflüssig wird: Die Teams müssen über alle Mittel verfügen, die sie für eine erfolgreiche Arbeit benötigen, der unternehmerische Wert ihrer Problemlösungen muss ihnen klar sein und man sollte ihnen den erforderlichen Freiraum und die Zeit gewähren, damit sie ihre Arbeit fertigstellen können.

Beim klassischen Projektmanagement ist es die Aufgabe des Projektmanagers, den Umfang eines Projekts, die Kosten und den Zeitplan miteinander in Einklang zu bringen. Zudem ist er für das Qualitätsmanagement, die Berichterstattung und die Lösung zwischenmenschlicher Probleme verantwortlich. Agiles Projektmanagement zeichnet sich hingegen dadurch aus, dass das gesamte Team gemeinsame Entscheidungen trifft und gemeinsam an der Umsetzung dieser Vorgaben arbeitet. Durch ein agiles Projektmanagement werden die Teammitglieder zu engagierten und motivierten Mitarbeitern, die in kürzerer Zeit eine qualitativ hochwertige Arbeit leisten.

Trotz der Unterschiede zwischen den Ansätzen des „Project Management Institute“ (PMI) und agilen Methoden sind viele der Vorgehensweisen, die im Projektmanagementstandard „Project Management Body of Knowledge“ (PMBOK) beschrieben werden, durchaus mit agilen Praktiken vereinbar.

Klassisch	Agil
<b>Initiierung:</b> Definition und Autorisierung des Projekts	<b>Visionierung:</b> Auseichende Definition des Projekts zur Erstellung einer Sandbox für die Projektarbeit
<b>Planung:</b> Beschreibung des Managements des Projekts	<b>Roadmap:</b> Umsetzung der Vision in konkrete Produktfunktionen und in einen voraussichtlichen Zeitrahmen („Timebox“) für deren Realisierung
<b>Umsetzung:</b> Förderung der Zusammenarbeit der Projektgruppen zur Erledigung der Aufgabe	<b>Release:</b> Ermöglichung der schrittweisen und iterativen Entwicklung von potenziell auslieferbarem Code durch das Team
<b>Überwachung und Steuerung:</b> Überprüfung des Projektfortschritts und Behebung von Problemen	<b>Anpassung:</b> Einfügen von geplanten Haltepunkten zur Überprüfung und Anpassung des Prozesses und des Produkts
<b>Abschluss:</b> Formaler Abschluss jeder einzelnen Phase oder des Projekts und Abnahme der Projektarbeit	<b>Abschluss:</b> Retrospektive Betrachtung der Ergebnisse und der Entscheidungsfindung in Form von Lessons Learned innerhalb des Teams

Tabelle 1: PMBOK-Praktiken und agile Praktiken im Vergleich

Bei einer konsequenten, strikten Vorgehensweise sind agile Methoden ebenso mit dem „Capability Maturity Model Integration“ (CMMI) vereinbar wie klassische Wasserfallmethoden. Der Unterschied liegt darin, wann und wie diese Praktiken ausgeführt werden und welche Terminologie verwendet wird.

Das PMBOK unterscheidet Initiierung, Planung, Ausführung, Steuerung und Abschluss als Prozessgruppen innerhalb des Projektmanagements. Die agilen Prozessphasen Visionierung, Roadmap, Release, Anpassung und Abschluss sind mit den PMBOK-Phasen vergleichbar, spiegeln jedoch besser wider, wie Softwareentwicklung tatsächlich abläuft. Diese agilen

Projektphasen passen zudem ideal zu einer iterativen Entwicklungsumgebung mit kurzen Iterationen bis hin zu längerfristigen Releases.

Das PMBOK unterscheidet neun zentrale Wissensbereiche: Integration, Umfang, Zeit, Kosten, Qualität, Personal, Kommunikation, Risiko und Beschaffung. Hier ein Vergleich von einigen der Wissensbereiche mit agilen Praktiken.

**Integrationsmanagement: Ein kollaborativer, iterativer Prozess**

Beim Integrationsmanagement ist die vom Projektmanager erstellte Projektcharta ein zentrales Arbeitsergebnis. In der agilen Softwareentwicklung, deren Schwerpunkt auf Just-in-Time-Design und partizipatorischer Entscheidungsfindung liegt, wird die Planung zu einem kollaborativen, iterativen Prozess. Das gesamte agile Team erstellt eigenverantwortlich eine Roadmap und ein Backlog, an denen es sich orientiert. Der agile Projektmanager unterstützt das Team bei einer kontinuierlichen, iterativen Planung und fördert die Kommunikation.

Gemäß dem Grundsatz „minimaler Aufwand, maximaler Nutzen“ beinhaltet die tägliche Routine von agilen Teams eine konsequente Änderungskontrolle. Das Management von Produktänderungen erfolgt mithilfe eines abgestuften Backlogs von Features, die dem als am wichtigsten angesehenen Kunden den größten geschäftlichen Nutzen bieten. Während der Release- und Iterationsplanung werden die am höchsten eingestufted Features zwecks Kodierung und Tests und anschließender Abnahme durch die Nutzer vom Backlog in Iterationen überführt. Feedbacksammlung und Prozessverbesserungen sind ein fester Bestandteil jedes Iterationszyklus, um nicht nur das Produkt, sondern auch die Zusammenarbeit der Stakeholder zu verbessern. Das Team (Kunde, Entwickler, Tester, Analytiker, technischer Redakteur und Projektmanager) wird zu einer Art Änderungskontrollgremium, das in einem regelmäßigen Rhythmus eine kollaborative, kontinuierliche Planung vornimmt.

Integrationsmanagementprozess (klassisch)	Agile Planung (agil)
Erstellung von Projektcharta oder Projektplan	Erstellung von Roadmap und Backlog
Ausführung des Projektplans	Ausführung von iterativen Aufgaben
Leitung, Management, Überwachung, Steuerung	Unterstützung, Leitung, Zusammenarbeit
Integrierte Arbeitskontrolle	Ständiges Feedback, abgestuftes Backlog

Tabelle 2: Integrationsmanagementprozess vs. agile Planung

**Umfang- und Zeitmanagement: Feste Timeboxes**

„Scope Creep“, das heißt schleichender Funktionszuwachs, ist von jeher der Fluch von klassischen Projektmanagern. Bei plangestützten Strategien werden Änderungen des Umfangs nach Möglichkeit vermieden, und zwar auch dann, wenn sich Kundenbedürfnisse und geschäftliche Anforderungen ändern. Agile Ansätze hingegen rechnen mit Umfangsänderungen und planen diese mit ein.

Wie erwähnt, erfolgt bei der agilen Strategie die Entwicklung innerhalb einer Timebox mit einem festen Zeitplan und festgelegten Kosten, um die Features mit dem größten Nutzen für den Kunden implementieren zu können.

Das Umfangs- und Zeitmanagement ist der Bereich der agilen Softwareentwicklung, der Projektmanagern gewöhnlich am meisten Unbehagen bereitet. Die am häufigsten gestellte Frage von Projektmanagern in diesem Zusammenhang lautet „Für wann kann ich meinem Vorgesetzten/Kunden den Abschluss dieses Projekts in Aussicht stellen?“ Wir arbeiten hier nach wie vor mit der Box „Zeit“. Anstatt jedoch mehr „Featurebausteine“ in eine einzelne, dünne Box zu packen, bis diese Timebox platzt, verwenden wir nun mehrere Timeboxes aus Stahl. Sobald die Box voll ist, füllen wir keine Bausteine mehr nach. Anschließend wird die Box für die gesamte Dauer dieser Iteration fest verschlossen, um die Arbeit bis zur Abnahme der Features fortzusetzen. Da wir eine Timebox nach der anderen abarbeiten, lässt sich nur schwer abschätzen, wie viel Arbeit über einen längeren Zeitraum hinweg zu erledigen ist.

Ein Projektmanager ist kein Wahrsager. Unabhängig vom jeweiligen PM-Ansatz kann ein Projektmanager lediglich fundierte Vermutungen anstellen, die auf seinem Urteilsvermögen und einer historischen Analyse beruhen. Der wichtigste Unterschied in der agilen Entwicklung besteht darin, dass die Festlegung des Umfangs im Rahmen der Iterationsplanung erfolgt, bei der Features definiert, geschätzt und zugeteilt werden. Wenn die Product Owner und die Kunden die implementierten Features prüfen, testen und abnehmen, kann der ursprüngliche Projektumfang anhand von genauen Fortschrittskennzahlen ermittelt und den Stakeholdern mitgeteilt werden.

Umfang und Zeitmanagement (klassisch)	Timeboxes (agil)
Sammlung der Anforderungen, Festlegung des Umfangs	Erstellung und Auswahl eines Product Backlogs
Erstellung eines Projektstrukturplans (PSP)	Festlegung von Features für Release/Sprint
Festlegung und Sequenzierung der Projektaktivitäten	Auswahl von Features und Aufgaben für den Sprint durch das Team
Kontrolle des Projektplans	Verfeinerung der User-Story-Schätzung durch Team Velocity

Tabelle 3: Umfangs- und Zeitmanagement vs. Timeboxes

### Qualitätsmanagement: Von vornherein integriert

Mitarbeiter der Qualitätssicherung und -kontrolle sind es gewohnt, das letzte Glied in der Kette zu sein. Sie rechnen mit verkürzten Zeitfenstern für Tests, nicht dem gelieferten Produkt entsprechenden Spezifikationen und einem geringen Interesse an ihrem Input bis zu den letzten Wochen des Projekts. In der agilen Entwicklung werden Mitarbeiter der Qualitätssicherung mit der Analyse und dem Design des Produkts von Beginn an in den Prozess eingebunden und den gesamten Lebenszyklus über in die Entscheidungsfindung einbezogen. Bei einer inkrementellen Codeentwicklung besteht Qualitätssicherung in Tests zu Beginn des Projekts, bevor das fertige Produkt präsentiert wird.

Bei agilen Praktiken trägt jeder Einzelne dazu bei, die Qualität des Produkts festzulegen, zu erhalten und zu verbessern. Entwickler leisten einen Beitrag zu den Testarbeiten, indem sie Modultests schreiben und das Rahmenkonzept zur Automatisierung von Regressions- und Abnahmetests unterstützen. Product Owner wirken ihrerseits bei der Ausarbeitung und der Durchführung von Abnahmetests mit. Zur Vermeidung von Fehlern beteiligen sich Mitarbeiter der Qualitätssicherung täglich an der Entscheidungsfindung des Entwicklungsteams. Ihr Input hilft den Entwicklern, einen besseren Code zu schreiben.

Zu Qualitätskontrollzwecken werden im Rahmen der Iteration Bugchecks durchgeführt. In der agilen Entwicklung übernehmen alle gemeinsam Verantwortung dafür, dass die zu kodierenden Features qualitativ hochwertig sind und den Erwartungen des Kunden entsprechen.

Qualitätsmanagement (klassisch)	Integrierte Qualität (agil)
Qualitätsplanung	Verantwortung des gesamten Teams
Qualitätssicherung	QS in Sprints und Retrospektiven integriert
Qualitätskontrolle	Modultests in jeder Iteration

Tabelle 4: Qualitätsmanagement vs. integrierte Qualität

**Personalmanagement: Die Teams organisieren sich selbst**

Das PMBOK definiert Personalplanung als einen Prozess, bei dem Rollen und Aufgaben verteilt werden. Teamentwicklung wird als Entwicklung der entsprechenden Fähigkeiten und Kompetenzen jedes einzelnen Teammitglieds verstanden. Das agile Pendant hierzu ist die Gründung von funktionsübergreifenden Teams, denen die Möglichkeit gegeben wird, sich eigenständig zu organisieren.

Die Gründung eines funktionsübergreifenden Teams im agilen Projektmanagement bedeutet den Aufbau eines Teams, dem nicht nur Programmierer, sondern alle für die Entwicklung eines Arbeitscodeinkrements benötigten Schlüsselbeteiligten angehören. Hierbei handelt es sich um Tester, Analytiker, Architekten, technische Redakteure, Fachexperten, Kunden/Product Owner und den Teamleiter (Projektmanager, Scrum Master usw.). Jeder Einzelne bringt bestimmte Kompetenzen mit ein, wobei jedes Teammitglied mit zunehmender Reife des Teams ein Verständnis für die Aufgaben und Bemühungen der anderen entwickelt.

Ein Team mit einer agilen Struktur ist besser in der Lage, Verantwortung für die Qualität und das Tempo seines Fortschritts zu übernehmen. Die Gesamtverantwortung des Teams für Planung, Ausführung und Überprüfung führt zu einer selbstgesteuerten Leistung, die durch einen kontinuierlichen Reflexions- und Verbesserungsprozess aufrechterhalten wird.

Klassische Projektmanager tun sich bisweilen schwer, dieses Konzept der kollaborativen Verantwortung mit selbstorganisierten Teams zu erfassen. Hier kommt der Wechsel von Befehls- und Kontrollmanagement zu Servant Leadership ins Spiel. Wer ein „Servant Leader“ werden möchte, muss lernen, Zusammenarbeit und Reflexion zu fördern, Beratung und



Unterstützung anzubieten und Hindernisse aus dem Weg zu räumen, damit das Team die bestmögliche Arbeit leisten kann. Natürlich kann sich die Dynamik eines Teams nicht über Nacht ändern. Teams, die auf den agilen Ansatz umstellen, profitieren von einer starken Führung und qualifizierten Teamleitern.

Personalmanagement (klassisch)	Selbstorganisierte Teams (agil)
Personalplanung	Dedizierte Teams aus 7 +/- 2 Mitgliedern
Teamentwicklung	Funktionsübergreifende, kollaborative Teams
Teammanagement	Servant Leadership, Teams organisieren sich selbst

Tabelle 5: Personalmanagement vs. selbstorganisiertes Team

Der Vergleich von Wasserfallmethode und Agile zeigt deutlich, wo die Vorteile von Agile liegen: Der iterative Entwicklungsprozess, der starke Fokus auf dem Kundenfeedback und das Engagement hochmotivierter, eigenverantwortlicher Mitarbeiter bringen schnellere und qualitativ hochwertigere Ergebnisse. Aufgrund dieser Erfolge werden agile Methoden sich weiterhin stark verbreiten.

Aufmacherbild: [waterfall in deep rain forest jungle](#) von Shutterstock /  
Urheberrecht: pimpic