

Maschinen, Computer etc. zur Nutzung für andere freigegeben. Häufig beobachten Projektmanager gegen Ende des Projekts, dass Mitarbeiter des Projektteams bereits vermehrt an anderen Projekten arbeiten. Für den Ressourceneinsatz in planbasierten Projekten ist es jedoch wichtig, klare Abmachungen hinsichtlich der Verfügbarkeit des Projektteams zu treffen und einzuhalten. Projektmanager sollten für die Abschlussphase die noch anstehenden Aufgaben kommunizieren und sich der Verfügbarkeit der Mitarbeiter versichern.

Ein wichtiger Aspekt bei der Auflösung des Projektteams ist die *Würdigung der Beiträge der einzelnen Mitarbeiter*. Mitarbeiter in Projekten leiden in manchen Unternehmen darunter, dass ihre disziplinarischen Vorgesetzten nur wenig von ihrem Engagement und ihren Leistungen in den Projekten erfahren. Projektmanager sind aufgefordert, ihre Beobachtungen und Eindrücke nicht nur mit den Mitarbeitern zu teilen, sondern auch sicherzustellen, dass die Projektarbeit in angemessener Weise bei den Linienmanagern wahrgenommen wird.

Am Ende der Abschlussphase . . .

- ist der Projektgegenstand durch den Auftraggeber abgenommen,
- ist das Projekt analysiert,
- ist das Wissensmanagement abgeschlossen und
- das Team und die Infrastruktur aufgelöst.

Kontinuierliche Aufgaben des Projektmanagements

Wir haben in den vorangegangenen Kapiteln einige Grundlagen planbasierten Projektmanagements entlang zeitlich sortierter Projektphasen wiederholt. Neben den dort beschriebenen Aufgaben gibt es auch Aufgaben, die sich nicht einzelnen Phasen zuordnen lassen, sondern die über den gesamten Projektlebensweg bearbeitet werden. Hierzu gehören:

- Qualitätsmanagement
- Berichtswesen
- Risikomanagement
- Lieferantenmanagement
- Vertragsmanagement
- Nachforderungsmanagement
- Dokumentenmanagement
- Konfigurationsmanagement
- Änderungsmanagement
- Wissensmanagement

Um die Themen kurz ins Gedächtnis zu rufen, werden die einzelnen Aufgabenbereiche nachfolgend kurz umrissen. Für eine tiefer gehende Betrachtung wird auf entsprechende Grundlagenliteratur (Timinger 2017; Patzak und Rattay 2018) verwiesen.

Qualitätsmanagement

In Projekten tritt das Qualitätsmanagement auf zwei Ebenen auf:

- Qualitätsmanagement bezogen auf das Projektmanagement, dessen Prozesse, Rollen und Aufgaben sowie
- Qualitätsmanagement bezogen auf den Projektgegenstand, mit dem Ziel, alle Anforderungen an diesen zu erfassen und zu erfüllen.

Die Qualität des Projektmanagements hat dabei erheblichen Einfluss auf die Qualität des Projektgegenstands. Die Begriffe Qualität und Qualitätsmanagement definieren wir gemäß der DIN EN ISO 9000:

Qualität ist der Grad, bis zu dem ein Objekt bestimmte Anforderungen erfüllt.

Qualitätsmanagement umfasst die Festlegung der Qualitätspolitik, die Ableitung von Qualitätszielen, deren Planung und Steuerung sowie die stetige Verbesserung der Qualität.

Qualitätsmanagement ist ein kontinuierlicher Prozess der Verbesserung. Deming und Shewhart haben das im sogenannten **PDCA-Zyklus** beschrieben. PDCA steht für **Plan, Do, Check, Act**, zu Deutsch: planen, umsetzen, prüfen, handeln. Damit ist gemeint, Maßnahmen für Verbesserungen zu identifizieren, zu planen und umzusetzen. Der Erfolg der Maßnahmen wird anhand möglichst quantitativer Kriterien überprüft und bewertet. Daraus leiten sich weitere Handlungen ab, um das geplante Ziel der Verbesserung zu erreichen. Im Anschluss beginnt der Zyklus mit neuerlichen Planungen zur kontinuierlichen Verbesserung von vorne. Der PDCA-Zyklus ist Ausgangspunkt vieler Qualitätsmanagementsysteme in Unternehmen.

*Ein **Qualitätsmanagementsystem** ist Teil der Unternehmensführung. Es regelt alle Aktivitäten des Qualitätsmanagements mit dem Ziel der systematischen Erreichung der qualitätsbezogenen Ziele.*

Zu den wichtigen Elementen eines Qualitätsmanagementsystems gehören eine Qualitätsmanagementstrategie, Verfahrensanweisungen, die Abläufe zur Erledigung von qualitätsbezogenen Arbeiten festlegen, sowie konkrete Arbeitsanweisungen. Außerdem können (Dokumenten-)Vorlagen und Hilfsmittel Bestandteil des Qualitätsmanagementsystems sein.

Vier wichtige Teilaufgaben des Qualitätsmanagements mit Bezug zum Projektmanagement sind die

- **Qualitätsplanung:** Bei der Qualitätsplanung werden alle Aufgaben geplant, die die vom Auftraggeber geforderte Qualität an den Projektgegenstand sicherstellen sollen. Beispiele solcher Aufgaben sind Design Reviews und die Definition von Quality Gates.
- **Qualitätslenkung:** Während der Qualitätslenkung werden die in der Qualitätsplanung vorgesehenen Maßnahmen auf ihre Bearbeitung und Wirksamkeit zur Sicherstellung der Qualitätsziele überwacht und gegebenenfalls korrigiert.
- **Qualitätssicherung:** Die Qualitätssicherung umfasst alle geplanten und systematischen Aktivitäten, die zur Erreichung von Qualitätsanforderungen notwendig sind. Eine typische derartige Aktivität ist die Aufnahme neuer Erkenntnisse in das Qualitätsmanagementsystem. Damit wird verhindert, dass aktuelle oder mögliche Qualitätsprobleme künftig erneut auftreten.

- **Qualitätsverbesserung:** Die Qualitätsverbesserung zielt auf die Steigerung des Qualitätsniveaus. Beispiele sind die Festlegung einer geringeren Fehleranzahl oder eine schnellere Erreichung eines definierten Fehlerniveaus. Es reicht jedoch nicht, die Ziele zu erhöhen. Vielmehr müssen Maßnahmen initiiert werden, die die Zielerreichung auch ermöglichen.

Die Wahl und Ausgestaltung des Vorgehensmodells für Projektmanagement hat großen Einfluss auf die Qualität des Projektgegenstands, wie Sie im Abschnitt Qualitätssicherung in Kapitel 2 noch sehen werden.

Berichtswesen

Durch das Berichtswesen werden wichtige Stakeholder über den Status des Projekts informiert. Die Berichterstattung kann informell oder formell sein. Ein Beispiel für die **informelle Berichterstattung** ist der Aushang von Informationen und Projekterfolgen am Schwarzen Brett oder der Versand eines E-Mail-Newsletters.

Die **formelle Berichterstattung** erfolgt meistens über standardisierte Kommunikationskanäle, beispielsweise mithilfe eines Fortschrittsberichts oder als Statusbesprechung bei wichtigen Meilensteinen. Wichtigster Adressat ist der Auftraggeber des Projekts. Deshalb wird die formelle Berichterstattung auch häufig vom Auftraggeber oder dem Lenkungsausschuss vorgegeben. Die Vorgaben beziehen sich insbesondere auf die Häufigkeit und den Inhalt des Berichts.

Vorsicht

Einige Auftraggeber wünschen sich sehr häufige und umfassende Projektberichte und vergessen dabei, dass diese Zeit und Geld kosten. Zudem besteht die Gefahr, dass zu detaillierte Berichte mehr Fragen aufwerfen können, als sie Antworten liefern, da der Auftraggeber häufig nicht die notwendigen Hintergrundinformationen und auch nicht die notwendige Zeit hat, die Berichtsinhalte zu verarbeiten.

Ein effizientes Berichtswesen berücksichtigt die Bedürfnisse des Auftraggebers nach Information und wägt diese angemessen mit dem Aufwand ab, den die Berichterstattung nach sich zieht. Außerdem unterscheidet es zwischen den berechtigten Interessen nach Information über den Projektgegenstand und der operativen Projektdurchführung, die der Auftraggeber dem Projektteam überlassen sollte.

Bei der Festlegung des Berichtsumfangs sollte außerdem unterschieden werden zwischen Informationen, die zur Projektsteuerung wichtig sind, und Informationen, die in die Rubrik »wäre doch interessant zu wissen« fallen. Letztgenannte sollten nicht Eingang in den Projektbericht finden.

Risikomanagement

Im Rahmen des Risikomanagements werden Risiken identifiziert, analysiert, bewertet und gegebenenfalls Gegenmaßnahmen geplant. Projektgefährdende Risiken sind zu minimieren. Mit der gleichen Methodik können Chancen, die den Projekterfolg fördern, gesucht und maximiert werden.

Risiken und Chancen sind mögliche, ungeplante Ereignisse, die durch eine Eintrittswahrscheinlichkeit und eine Auswirkung bei Eintritt charakterisiert werden. Risiken wirken den Projektzielen entgegen, das heißt, dem Projekt entsteht ein Schaden. Chancen unterstützen die Ziele.

Die über Kreativitätstechniken, die Sichtung der bereits vorliegenden Projektunterlagen und über Checklisten identifizierten Projektrisiken (und -chancen) werden hinsichtlich ihrer Auswirkung analysiert. Abbildung 1.16 zeigt drei exemplarische Risiken einer zu organisierenden Gartenparty.

Liste der identifizierten Risiken und ihre Bewertung

#	Risiko	Auswirkung	EWK ¹⁾	SK ²⁾	RI ³⁾	Maßnahmen	neue EWK	neue SK	neuer RI
1	Schlechtes Wetter	Gäste werden nass	2 - mittel	3 - hoch	6 - inakzeptabel	Pavillon beschaffen	2 - mittel	1 - gering	2 - akzeptabel
2	Schlechtes Wetter	Gäste kommen mit PKW, zu wenig Parkplätze	2 - mittel	2 - mittel	4 - bei Bedarf minimieren	Bei Nachbarn zusätzlichen Parkraum anfragen	2 - mittel	1 - gering	2 - akzeptabel
3	Grillkohle brennt schlecht	Essen braucht mehr Zeit	1 - gering	2 - mittel	2 - akzeptieren	-			

¹⁾ Eintrittswahrscheinlichkeitsklasse: 1-gering, 2-mittel, 3-hoch

²⁾ Schadensklasse: 1-gering, 2-mittel, 3-hoch

³⁾ Risikoindex: EWK · SK

Visualisierung der Risiken nach Nummer (#) in einer Risikomatrix jeweils ohne (grau) und mit (schwarz) geplanten Maßnahmen

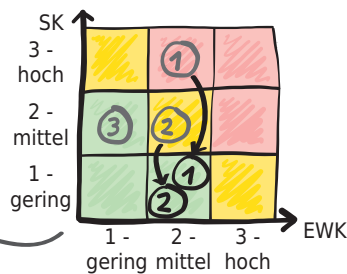


Abbildung 1.16: Analyse und Visualisierung identifizierter Risiken mit Maßnahmenplanung

Die Eintrittswahrscheinlichkeit des Risikos und die Schwere des Schadens werden bestimmt. Häufig wird die Wahrscheinlichkeit nicht in Prozent, sondern in definierten Klassen angegeben. Im Beispiel der Abbildung 1.16 gibt es beispielsweise die drei Klassen gering, mittel und hoch. Auch der Schaden kann entsprechend in Klassen angegeben werden. Der Risikoindex als Produkt aus Wahrscheinlichkeit und Schaden erlaubt den Vergleich der Tragweite mehrerer Risiken. Risiken mit großer Tragweite haben eine hohe Wahrscheinlichkeitsklasse und gleichzeitig eine hohe Schadensklasse. Im Risikomanagementplan sollte festgelegt werden, wie viele Klassen zur Bewertung herangezogen werden, wie diese definiert sind und welche Risiken zu minimieren sind beziehungsweise welche Risiken akzeptiert werden können.

Risiken, die in Zusammenhang mit dem Verhalten von Personen stehen, sollten im Rahmen des Stakeholdermanagements analysiert, bewertet und mit Maßnahmen versehen werden.

Lieferantenmanagement

Das Lieferantenmanagement dient der systematischen Auswahl, Planung und Steuerung der mit Lieferanten zusammenhängenden Aufgaben. Eine Unterscheidung in strategisches und operatives Lieferantenmanagement ist möglich. Beim strategischen Lieferantenmanagement geht es um übergeordnete Fragestellungen, wie

- der Festlegung, welche Produkte und Dienstleistungen an welcher Stelle der Wertschöpfungskette überhaupt beschafft werden sollen, und
- der Definition von generellen Anforderungen an Lieferanten, beispielsweise hinsichtlich deren Lieferfähigkeit, Qualitätsmanagementsystem etc.

Beim operativen Lieferantenmanagement geht es um die konkrete Bewertung, Auswahl und Steuerung eines Lieferanten für die Beschaffung eines bestimmten Produkts oder einer Dienstleistung. Einige Branchen und Normen wie die DIN 9001:2015 (DIN EN ISO 9001) haben spezielle Anforderungen an das Lieferantenmanagement. Zu den wichtigsten Aufgaben gehören

- die *Festlegung von Anforderungen* an zu beschaffende Produkte und Dienstleistung,
- die *Bewertung der Lieferanten* hinsichtlich ihrer Fähigkeit, diese Anforderungen zu erfüllen,
- die *Überwachung der Lieferanten* beziehungsweise der beschafften Produkte und Dienstleistungen,
- die *regelmäßige Neubewertung* der Lieferanten und die
- die *Weiterentwicklung des Leistungsniveaus* der Lieferanten.

Vertragsmanagement

Sobald unternehmensexterne Organisationen am Projekt beteiligt sind, beispielsweise Lieferanten und Entwicklungspartner, werden Vereinbarungen in Verträgen geschlossen. Aber auch unternehmensintern kann die Zusammenarbeit zwischen Abteilungen über Projektvereinbarungen, Service-Level-Agreements oder andere Verträge geregelt werden. Projektmanager sind meistens keine ausgebildeten Rechtsexperten. Einige rechtliche Grundlagen und Kenntnisse des Vertragsmanagements sollten sie aber mitbringen, um mit der Rechtsabteilung oder der Beschaffungsabteilung zusammenarbeiten zu können.

Verträge regeln die Beziehungen zwischen zwei oder mehreren Parteien. Sie halten fest, was von wem an wen zu leisten ist und welche Regeln dabei gelten. Verträge sind die Grundlage von Rechtsgeschäften.

Das **Vertragsmanagement** ist ein Aufgabengebiet des Projektmanagements. Es dient der Erstellung, dem Abschluss und der Abwicklung von Verträgen.

Die Abläufe und Methoden des Vertragsmanagements unterstützen Sie bei der Erreichung der Projektziele, indem

- die richtigen Verträge in einer dem Projekt dienlichen Form abgeschlossen werden,
- die Vertragsbestandteile während der Projektsteuerung auf Einhaltung überprüft werden,

- Vertragsabweichungen sorgfältig dokumentiert und von den Vertragspartnern nachgefordert werden.

Verbreitete Vertragsarten sind Kaufverträge, Werkverträge und Dienstverträge, die im Bürgerlichen Gesetzbuch (BGB) näher erläutert werden, siehe Abbildung 1.17.

Kaufvertrag	Werkvertrag	Dienstvertrag
<p>§ 433 BGB regelt:</p> <p>(1) Der Verkäufer einer Sache ist verpflichtet, dem Käufer die Sache zu übergeben und das Eigentum an der Sache zu verschaffen. Der Verkäufer hat dem Käufer die Sache frei von Sach- und Rechtsmängeln zu verschaffen.</p> <p>(2) Der Käufer ist verpflichtet, dem Verkäufer den vereinbarten Kaufpreis zu zahlen und die gekaufte Sache abzunehmen.</p>	<p>§ 631 BGB regelt:</p> <p>(1) Durch den Werkvertrag wird der Auftragnehmer zur Herstellung des versprochenen Werkes, der Auftraggeber zur Entrichtung der vereinbarten Vergütung verpflichtet.</p> <p>(2) Gegenstand des Werkvertrags kann sowohl die Herstellung oder Veränderung einer Sache als auch ein anderer durch Arbeit oder Dienstleistung herbeizuführender Erfolg sein.</p>	<p>§ 611 BGB regelt:</p> <p>(1) Durch den Dienstvertrag wird derjenige, welcher Dienste zusagt, zur Leistung der versprochenen Dienste, der andere Teil zur Gewährung der vereinbarten Vergütung verpflichtet.</p> <p>(2) Gegenstand des Dienstvertrags können Dienste jeder Art sein.</p>

Abbildung 1.17: Vertragsarten in der Übersicht

Nachforderungsmanagement

Das Nachforderungsmanagement ist eng mit dem Vertragsmanagement verbunden:

*Eine **Nachforderung** (englisch Claim) ist eine aus einem Vertrag resultierende Forderung, die an den jeweils anderen Vertragspartner gestellt werden kann. Voraussetzung ist, dass die Forderung bereits im Vertrag oder in einer Änderungsvereinbarung dokumentiert worden ist.*

*Das **Nachforderungsmanagement** dient der Überwachung von Abweichungen und Änderungen von Verträgen. Diese werden hinsichtlich ihrer Folgen für das Projekt beurteilt und gegebenenfalls in Form von Nachforderungen gegen den Vertragspartner durchgesetzt.*

Werden Aufgaben des Projekts über Verträge geregelt, bedeutet eine Abweichung vom Vertrag gleichzeitig eine Abweichung der festgelegten Aufgaben. Häufig ist dies mit Mehrkosten oder Zeitverzögerungen verbunden. Eine erfolgreiche Projektsteuerung setzt deshalb ein sorgfältiges Nachforderungsmanagement voraus. Hierfür müssen Projektmanager die Vertragsinhalte kennen und überwachen. Für die Überwachung bietet es sich an, wichtige Ereignisse, Termine und Lieferungen aus den Verträgen in die übrige Projektplanung zu übernehmen, kenntlich zu machen und bei Abweichungen zeitnah entsprechende Nachforderungen zu stellen.

Dokumentenmanagement

In Projekten werden meistens viele Dokumente erzeugt. Man kann sie unterteilen in Dokumente, die den Projektgegenstand beschreiben, wie beispielsweise Anforderungsdokumente, Design-Konstruktionszeichnungen, und Dokumente, die für das Projektmanagement benötigt werden, wie beispielsweise Projektauftrag, Pläne und Berichte.

Die Auflistung ist bei Weitem nicht vollständig. Hinzu kommt, dass Dokumente verschiedene Stadien (Entwurfsstadium, freigegebenes Dokument, überarbeitetes Dokument ...) durchlaufen, was ihre Anzahl weiter erhöht.

Vier wichtige Begriffe des Dokumentenmanagements sind:

Dokumente sind Informationen, die in Form von Texten, Abbildungen oder Video- und Tonaufzeichnungen in abgeschlossener Form zusammengefasst werden. Dokumente werden meistens in Papierform oder elektronisch als Datei oder Website abgelegt.

Unter **Dokumentation** versteht man den kompletten Lebenszyklus von Dokumenten, von deren Erstellung über die Aktualisierung bis hin zu ihrer Archivierung und Vernichtung.

Die **Projekttakte** bezeichnet die Menge aller projektbezogenen Dokumente.

Das **Dokumentenmanagement** hat zum Ziel, Dokumente mit vertretbarem Aufwand bereitzustellen. Dafür werden die Dokumente gekennzeichnet, registriert und archiviert.

Heute bedient man sich meistens einer datenbankgestützten Software, Dokumentenmanagementsystem genannt und manchmal mit DMS abgekürzt, um Dokumente abzulegen und abzurufen. Einfache, kleinere Projekte kommen ohne derartige Software aus. Stattdessen können Dokumente in Papierform im Büro des Projektmanagers oder elektronisch auf einem gemeinsamen Cloud-Speicherplatz oder einer Netzwerkfestplatte abgelegt werden. Achten Sie auf eine angemessene Datensicherung. Ohne ein geeignetes Dokumentenmanagementsystem können Dateien einfach gelöscht, versehentlich überschrieben oder manipuliert werden. Vielfach bieten auch Systeme für das Geschäftsprozessmanagement, sogenannte ERP-Systeme (englisch für Enterprise Resource Planning), entsprechende Funktionen für das Dokumentenmanagement.

Zu unterscheiden sind Systeme zur synchronen und asynchronen Dokumentenbearbeitung. Bei der synchronen Dokumentenbearbeitung können mehrere Personen an verschiedenen Standorten am gleichen Dokument arbeiten. Von einer Person durchgeführte Änderungen werden dann unmittelbar auch bei den anderen Bearbeitern des Dokuments angezeigt. Bei der asynchronen Dokumentenbearbeitung wird ein gerade bearbeitetes Dokument für andere Personen gesperrt. Damit wird verhindert, dass unterschiedliche Bearbeitungsstände an verschiedenen Standorten zu inkonsistenten Daten führen.

In jedem Fall sollte ein gutes Dokumentenmanagementsystem alte Revisionsstände eines Dokuments wiederherstellen können. Damit kann jeder beliebige zurückliegende Bearbeitungsstand des Dokuments wieder aufgerufen werden, beispielsweise im Falle von versehentlichen Änderungen.

Zu Projektbeginn sollten Sie als Projektmanager eine Projekttakte anlegen. Darin wird festgelegt, wie Dokumente im Projekt erzeugt und archiviert werden. Heute ist es üblich, Dokumente elektronisch bereitzustellen. Ein gemeinsames Laufwerk im Intranet ermöglicht allen Teammitgliedern, auf Dokumente zuzugreifen und eigene Dokumente abzulegen. Ohne Regeln wird die Dokumentenablage aber schnell unübersichtlich. Legen Sie deshalb den Aufbau und die Organisation der Projekttakte fest. Dazu gehören

- eine Übersicht über alle benötigten Dokumente,
- ein Schema zur Benennung von Dateien und Dokumenten,
- eine Übersicht über die Ordnerstruktur der Projekttakte,
- ein Verfahren, nach dem neue Dokumente erstellt und freigegeben werden.

Konfigurationsmanagement

Das Dokumentenmanagement hilft, Ordnung in die Vielzahl von Dokumenten im Projekt zu bringen. Eine ähnliche Zielsetzung, aber weiter gefasst und nicht auf Dokumente begrenzt, hat das Konfigurationsmanagement.

Eine **Konfiguration** beschreibt die funktionellen und physischen Merkmale eines Produkts, einer Leistung oder ganz allgemein eines Objekts.

Eine **Konfigurationseinheit** erfüllt eine der in der Konfiguration definierten Funktionen oder eines der Merkmale.

Das **Konfigurationsmanagement** sichert die Transparenz, Rückverfolgbarkeit und Übereinstimmung aller funktionellen und physischen Merkmale eines Objekts über dessen gesamte Lebensdauer.

Für das Konfigurationsmanagement existiert mit der DIN EN ISO 10007 eine eigene Norm. Die Ziele des Konfigurationsmanagements ergeben sich bereits aus der Definition des Begriffs. Um die angestrebte Transparenz, Rückverfolgbarkeit und Übereinstimmung von Anforderungen und Merkmalen zu erreichen, ist folgender Prozess üblich:

- Konfigurationsmanagementplanung – Definition der anzuwendenden Verfahren, Methoden und der Verantwortlichkeiten
- Konfigurationsidentifizierung – Festlegung einer Referenzkonfiguration und deren Beschreibung
- Konfigurationsüberwachung – Überwachung von Änderungen an der Konfiguration
- Konfigurationsbuchführung – Sicherstellung der Rückverfolgbarkeit durch Dokumentation der Referenzkonfiguration und aller im Rahmen der Überwachung freigegebenen Änderungen
- Konfigurationsauditierung – Überprüfung, ob ein Produkt mit seinen Anforderungen und der dokumentierten Konfiguration übereinstimmt

Änderungsmanagement

Selbst in gut geplanten Projekten mit sorgfältiger Zieldefinition und -analyse kommt es im Projektverlauf mit hoher Wahrscheinlichkeit zu Änderungen. Zu den Ursachen von Änderungen im Projekt gehören

- die Änderung von Rahmenbedingungen, beispielsweise neue Gesetze, Normen und Richtlinien,
- neue Technologien oder Abkündigungen von Komponenten, die Produktbestandteil werden sollten,
- geänderte Kundenwünsche, beispielsweise inspiriert durch neue Konkurrenzprodukte oder veränderte Märkte, oder
- Probleme bei der Umsetzung des geplanten Konzepts.

Änderungen können *notwendig* sein und den Projekterfolg überhaupt erst möglich machen. Sie können *sinnvoll* sein und das zu entwickelnde Produkt oder die zu entwickelnde Dienstleistung besser und wettbewerbsfähiger machen. Änderungen können Projekte aber auch in

ernste *Schwierigkeiten* bringen, wenn sie ungesteuert eingebracht und ohne Kenntnis der Auswirkungen umgesetzt werden. Änderungsmanagement sorgt für die transparente und gesteuerte Integration von Änderungen in das Projekt. Hierfür werden der Änderungsbedarf identifiziert, die Änderung beschrieben, die Auswirkungen der Änderungen ermittelt, die Änderungsanträge bewertet und die Änderungen gegebenenfalls genehmigt, umgesetzt und verifiziert.

Wir werden uns später genauer ansehen, welchen Einfluss die Wahrscheinlichkeit von Änderungen im Projektverlauf auf die Wahl des planbasierten, agilen oder hybriden Vorgehensmodells hat. So viel aber schon vorweg: Der Name *planbasiertes Vorgehensmodell* impliziert bereits, dass die Zukunft des Projekts planbar ist. Ist bereits zu Projektbeginn klar, dass es viele Änderungen geben wird, beispielsweise aufgrund unklarer Kundenanforderungen, können planbasierte Vorgehensmodelle trotz sorgfältigem Änderungsmanagement an ihre Grenzen stoßen.

Wissensmanagement

Das Wissensmanagement beschäftigt sich damit, Wissen und Erfahrungen einzelner Mitarbeiter oder ganzer Organisationseinheiten anderen Mitarbeitern und Organisationseinheiten zur Verfügung zu stellen. Bei Projekten besteht die Herausforderung darin, dass sie zeitlich begrenzt sind. Am Ende löst sich das Team auf. Neue Teams mit anderer Besetzung und neuen Zielen entstehen. Wissen und Erfahrungen müssen über diesen zeitlichen und organisatorischen Bruch gerettet werden. Dafür reicht es nicht aus, am Ende des Projekts noch schnell ein paar Informationen zu sammeln und weiterzugeben.

Wissensmanagement ist eine kontinuierliche Tätigkeit während des gesamten Projektlebenswegs. Erfahrungen und Erkenntnisse über eingetretene Risiken, wirksame oder wirkungslose Maßnahmen zur Risikobeherrschung oder Projektsteuerung sollten dann analysiert, bewertet und für die Zukunft gesichert werden, wenn die Erinnerung daran noch aktuell ist.

Grundlagen agilen Projektmanagements

Einführung

Agiles Projektmanagement wird manchmal als Gegenentwurf zum planbasierten, traditionellen Projektmanagement verstanden. Tatsächlich unterscheidet es sich in mehrfacher Weise deutlich. Wichtige Unterschiede sind der sogenannte agile Mindset und die geringere Bedeutung und Detailtiefe der Pläne. Obwohl auch im agilen Projektmanagement mit Plänen gearbeitet wird, stehen andere Dinge wie Flexibilität und kundenorientiertes Arbeiten im Vordergrund. Wir werden die Charakteristika agilen Projektmanagements nun gemeinsam herausarbeiten.

Eine vereinfachende und doch aussagekräftige Darstellung des Unterschieds liefert Wysocki (Wysocki 2014), die Sie bereits in Abbildung 1.1 kennengelernt hatten.

Während bei planbasierten Vorgehensmodellen das Projekt sequenziell definiert (Ziele festgelegt, Anforderungen ermittelt), geplant, implementiert und abgeschlossen wird, erfolgt die Bearbeitung bei agilen Projekten üblicherweise iterativ. Auch bei agilen Projekten wird das Projekt zunächst definiert. Die Planung beschränkt sich aber auf das notwendige Maß. Es wird nur die unmittelbare, tatsächlich vorhersehbare Zukunft geplant. Zeitlich darüber hinaus gehende Pläne werden nur *grob* umrissen. Zwischenergebnisse werden bewertet. Die Bewertung schließt eine Reflexion des eigenen Arbeitens und des Projektgegenstands mit ein. Häufig wird dann gemeinsam mit dem Kunden entschieden, ob eine weitere Iteration zur Verbesserung oder Ergänzung des Projektgegenstands notwendig ist. Diese wird dann wieder geplant, implementiert und neuerlich bewertet. Dieses iterative Vorgehen wird wiederholt, bis der Projektgegenstand im Sinne des Kunden fertig ist. Dann wird das Projekt abgeschlossen.

Neben diesen ablauf- oder prozessorientierten Unterschieden gibt es aber auch führungs- und organisationsorientierte Unterschiede, wie Sie nachfolgend noch sehen werden.

Agiles Arbeiten lässt sich bis mindestens in die 1980er-Jahre zurückverfolgen, auch wenn sich der heutige Begriff des agilen Projektmanagements erst später verfestigt hat. Wichtige Impulse lieferten 1986 Takeuchi und Nonaka in ihrer Veröffentlichung »The New New Product Development Game«. Dort nannten sie folgende Erfolgsfaktoren für besonders innovative und erfolgreiche Projekte (Takeuchi und Nonaka 1986):

- *Overlapping development phases:* Bei der phasenüberlappenden Bearbeitung wird die strikte Trennung nach Anforderungs-, Entwurfs-, Implementierungs- und Testphase aufgehoben. Eine integrierte phasenüberlappende Bearbeitung des Projektgegenstands fördert das Verständnis für diesen und verhindert, dass Probleme durch inkonsistente Anforderungen, ungeeignete Entwürfe und Fehler in der Implementierung erst spät erkannt werden.
- *Built-in instability:* Um die im Laufe der Projektbearbeitung gewonnenen Erkenntnisse und sich ändernde Anforderungen aufgreifen und zur Erzielung eines besseren Projektergebnisses nutzen zu können, dürfen Pläne nicht zum Selbstzweck werden. Die Erfüllung eines Plans ist kein Erfolg an sich. Projektteams sollen sich anspruchsvolle Ziele setzen und den zur Zielerreichung benötigten Freiraum zum Experimentieren bekommen. Nur durch experimentelles Vorgehen und das Bewusstsein, dass neue Lösungswege auch Fehlschläge beinhalten können, kann nach Meinung der beiden Autoren Innovation geschaffen werden.
- *Self-organizing project teams:* Projektteams agieren autonom. Dies setzt voraus, dass sie interdisziplinär besetzt sind, sodass sie alle anstehenden Aufgaben erledigen können. Autonomes Arbeiten erfordert ein hohes Maß an persönlicher Reife und sozialer Kompetenzen. Diese Aspekte sind bei der Zusammenstellung erfolgreicher Teams zu bedenken.
- *Subtle control:* Die Kontrolle und Steuerung erfolgen weitestgehend durch das Team selbst.
- *Multilearning:* Voneinander zu lernen, beispielsweise das Vermeiden von Fehlern und die Weitergabe von Erfolgsfaktoren, ist essenziell. Das Lernen geschieht mehrdimensional: Individuen reflektieren ihr Handeln. Das Team lernt voneinander und verbessert das eigene Arbeiten und den zu erarbeitenden Projektgegenstand.

- *Organizational transfer of learning*: Der Wissenstransfer vom Projektteam in die Stammorganisation ist fester Bestandteil des eigenen Lernens. Wissen endet nicht mit einem Projekt, sondern wird unternehmensweit weitergegeben und genutzt.

Interessanterweise tauchen bei Takeuchi und Nonaka die Begriffe agil oder Agilität an keiner Stelle im Text auf, wohl aber der Begriff Scrum, den die Autoren vom Rugbysport ableiteten. Im Rugby steht Scrum für Gedränge. Mit dem Begriff wollten die Autoren die Bedeutung der interaktiven Kooperation und Selbstorganisation eines Teams während der Produktentwicklung betonen.

Scrum als agiles Vorgehensmodell, wie wir es heute kennen, wurde maßgeblich von Ken Schwaber und Jeff Sutherland geprägt und 1995 auf der Konferenz Object-Oriented Programming, Systems, Languages & Applications (OOPSLA) vorgestellt (Schwaber 1997). Im Jahr 2001 folgte mit »Agile Software Development with Scrum« das erste Buch über Scrum, das von Schwaber und Beedle herausgebracht wurde (Schwaber und Beedle 2002).

Im gleichen Jahr wurde das Manifesto for Agile Software Development veröffentlicht (Beck 2001), an dem ebenfalls Schwaber, Sutherland und Beedle mitgewirkt haben.

Heute finden sich verschiedene agile Vorgehensmodelle im Einsatz. In einer Studie der Deutschen Gesellschaft für Projektmanagement wurde Scrum von den teilnehmenden Unternehmen als mit Abstand bedeutendstes Vorgehensmodell für die eigene Anwendung bewertet (Komus 2017). Auf den Plätzen folgen Kanban, Extreme Programming, Feature-Driven-Development, Lean Project Management und weitere Vorgehensmodelle.

In diesem Kapitel werden wir nachfolgend die Grundwerte agilen Projektmanagements wiederholen und einige für agile Projekte typische Methoden skizzieren.

Das Agile Manifest

Das Agile Manifest wurde 2001 von mehreren Softwareentwicklern aufgestellt und veröffentlicht. Begeben wir uns kurz auf eine Zeitreise in das Jahr 2001: Das Internet verbreitete sich stark, Software wurde komplexer und immer mehr Produkte nutzten sie, um Funktionalität für den Kunden anbieten zu können. Viele der damaligen Anwender können sich noch an instabile Computer-Betriebssysteme erinnern, die regelmäßige Neustarts erforderten. In dieser Zeit kamen Softwareentwickler zu dem Schluss, dass bis dahin etablierte Vorgehensmodelle an ihre Grenzen stoßen. Einige dieser Entwickler hatten Erfahrungen mit den Vorgehensmodellen Extreme Programming oder Scrum gemacht und deren Vorzüge für die Softwareentwicklung schätzen gelernt. Ohne sich auf ein agiles Vorgehensmodell mit konkreten Abläufen und Rollen festzulegen, wollten die Entwickler das Wesen agilen Arbeitens durch gemeinsame Werte und Prinzipien verdeutlichen. Die Verfasser des Agilen Manifests um Kent Beck, Mike Beedle, Ken Schwaber, Jeff Sutherland und andere schätzten folgende Werte (Beck 2001):

- **Individuen und Interaktionen** mehr als Prozesse und Werkzeuge
- **Funktionierende Software** mehr als umfassende Dokumentation
- **Zusammenarbeit mit dem Kunden** mehr als Vertragsverhandlung
- **Reagieren auf Veränderung** mehr als das Befolgen eines Plans

Beck und seine Mitstreiter betonten, dass die jeweils rechts genannten Aspekte nicht unwichtig, aber eben weniger bedeutsam als die links genannten und oben fett gedruckten Werte sind.

Weiter konkretisiert werden diese Werte durch zwölf ergänzende Prinzipien (Beck 2001):

- Unsere höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung wertvoller Software zufriedenzustellen.
- Heiße Anforderungsänderungen selbst spät in der Entwicklung willkommen. Agile Prozesse nutzen Veränderungen zum Wettbewerbsvorteil des Kunden.
- Liefere funktionierende Software regelmäßig innerhalb weniger Wochen oder Monate und bevorzuge dabei die kürzere Zeitspanne.
- Experten und Entwickler müssen während des Projekts täglich zusammenarbeiten.
- Errichte Projekte rund um motivierte Individuen. Gib ihnen das Umfeld und die Unterstützung, die sie benötigen, und vertraue darauf, dass sie die Aufgabe erledigen.
- Die effizienteste und effektivste Methode, Informationen an und innerhalb eines Entwicklungsteams zu übermitteln, ist im Gespräch von Angesicht zu Angesicht.
- Funktionierende Software ist das wichtigste Fortschrittsmaß.
- Agile Prozesse fördern nachhaltige Entwicklung. Die Auftraggeber, Entwickler und Benutzer sollten ein gleichmäßiges Tempo auf unbegrenzte Zeit halten können.
- Ständiges Augenmerk auf technische Exzellenz und gutes Design fördert Agilität.
- Einfachheit – die Kunst, die Menge nicht getaner Arbeit zu maximieren – ist essenziell.
- Die besten Architekturen, Anforderungen und Entwürfe entstehen durch selbstorganisierte Teams.
- In regelmäßigen Abständen reflektiert das Team, wie es effektiver werden kann, und passt sein Verhalten entsprechend an.

Das Agile Manifest wurde von Softwareentwicklern initiiert. Die Werte und Prinzipien werden heute aber auch in anderen Branchen respektiert und als Grundlage für das eigene Arbeiten angewandt. Dabei wird der Begriff »Software« durch »Projektgegenstand« oder »Liefergegenstand« ersetzt.

Die Werte und Prinzipien sind weitestgehend selbsterklärend. Trotz der Veröffentlichung des Agilen Manifests und dessen Verbreitung haben viele Organisationen große Schwierigkeiten bei der agilen Transformation. Agiles Arbeiten wird häufig gleichgesetzt mit dem Einsatz von Scrum oder Kanban. Diese beiden Vorgehensmodelle erleichtern zwar die Anwendung der Werte und Prinzipien, sie garantieren sie jedoch nicht.

Vorsicht

Agile Vorgehensmodelle wie Scrum und Kanban fördern die Anwendung und das Einhalten agiler Werte und Prinzipien. Werden Scrum und Kanban aber in hierarchischen, prozessorientierten und reglementierten Organisationen eingesetzt, kann die Einhaltung dieser Werte und Prinzipien schwierig oder gar unmöglich werden.

Erfahrene Projektmanager unterscheiden zwischen dem sogenannten **agilen Mindset**, das heißt dem Befolgen der Werte und Prinzipien und agilen Abläufen, das heißt dem Einsatz eines agilen Vorgehensmodells. Tatsächlich fördert letzteres agiles Arbeiten. Die wichtigere Voraussetzung für eine erfolgreiche agile Transformation ist jedoch der agile Mindset.

Agil arbeitende Organisationen halten sich unabhängig vom gewählten Vorgehensmodell an die agilen Werte und Prinzipien. Sie fördern Individuen und die persönliche Kommunikation, sie haben einen hohen Anspruch an die Erfüllung der Kundenwünsche, schaffen Rahmenbedingungen für eine gute Zusammenarbeit mit Kunden, Lieferanten und anderen Stakeholdern und reagieren angemessen flexibel auf Änderungswünsche.

Beispiel

Kennen Sie folgende Situation: Der Auftraggeber eines Projekts kommt kurz vor geplantem Projektende zum Projektteam und fordert eine den Zeitrahmen und das Budget überschreitende Änderung. Ohne Diskussion kommt die Aufforderung, diese umzusetzen und trotzdem Zeit und Kosten einzuhalten. Abschließend folgt der Kommentar: »Ihr schafft das schon, ihr arbeitet ja agil.«

Das vorgenannte Beispiel verdeutlicht ein typisches Missverständnis: Agiles Arbeiten fördert eine positive Grundhaltung gegenüber Änderungswünschen des Kunden. Es kann aber die Natur der Arbeit nicht umkehren. Änderungen verursachen üblicherweise Aufwände, denen in irgendeiner Form Rechnung getragen werden muss.

Agiles Projektmanagement wird häufig gleichgesetzt mit Scrum oder Kanban. Beides sind nach der Definition dieses Buchs Vorgehensmodelle und keine Einzelmethoden. Deshalb kommen wir zu Scrum und Kanban auch erst im folgenden Kapitel.

Sie wissen bereits, dass agiles Projektmanagement einer bestimmten Geisteshaltung und Offenheit gegenüber Veränderung entspringt. Agil denkende und handelnde Personen können deshalb auch mit eigentlich planbasierten Methoden wie dem Projektstrukturplan Agilität in ein Projekt tragen. Es gibt jedoch auch einige Methoden, die charakteristisch für agile Projekte sind und die nun auszugsweise kurz skizziert werden.

Initialisierung

Die Projektinitialisierung agiler Projekte kann ähnlich verlaufen wie bei planbasierten Projekten. In dieser frühen Phase geht es zunächst einmal darum, das Projektteam zu definieren und die Grobziele festzulegen. Außerdem sollte eine Vorentscheidung getroffen werden, welches Vorgehensmodell anzuwenden ist.

Zur Sammlung dieser Informationen kann auch bei agilen Projekten das bereits in Abbildung 1.3 vorgestellte Project Canvas eingesetzt werden.

Allerdings gibt es bei rein agil arbeitenden Projekten auch Unterschiede, die wir uns kurz erarbeiten: Wirklich agil arbeitende Projekte stellen das magische Dreieck des Projektmanagements auf den Kopf. Dies haben wir bereits weiter oben in diesem Kapitel anhand von Abbildung 1.4 diskutiert.

Bei planbasiert beziehungsweise traditionell arbeitenden Projekten wird üblicherweise der Leistungsumfang über Lasten- und Pflichtenhefte festgelegt. Daraus leiten sich die Aufgaben, deren Aufwände, die Termine und Kosten ab. In agil arbeitenden Projekten

herrscht teilweise ein anderes Verständnis vor: Die Kosten und Termine des Projekts werden vereinbart. Auch der Leistungsumfang wird definiert. Die Festlegung erfolgt allerdings als priorisierte, nicht notwendigerweise vollständige Liste. Es soll ja gerade die Möglichkeit der flexiblen Anpassung geschaffen werden. Das agil arbeitende Team wird nun alles daransetzen, die vom Auftraggeber hoch priorisierten Bestandteile des Projektgegenstands innerhalb des Kosten- und Terminrahmens abzuarbeiten. Weniger wichtige Dinge können jedoch gegebenenfalls weggelassen werden, um Kosten und Termine wie vereinbart einzuhalten.

Man nennt dieses Vorgehen auch Timeboxing.

Beim **Timeboxing** wird ein fester Zeitrahmen (eine Timebox) definiert. Innerhalb des vorgegebenen Zeitrahmens werden die Aufgaben in priorisierter Reihenfolge bearbeitet.

Selbstverständlich versuchen auch agile Teams, alle vorgesehenen Aufgaben zu bearbeiten. Im Vordergrund stehen aber bewusst die vom Kunden als besonders wichtig erachteten Aufgaben. Gleichzeitig wird durch die Timebox eine fristgerechte Abgabe sichergestellt.

Dieses Prinzip hat auch Auswirkungen auf den Umgang mit Änderungen und Verträgen zwischen Auftraggebern und Auftragnehmern agiler Projekte. Diese sind unter dem Leitsatz »(No) **Money for Nothing, Changes for Free**« bekannt.

Unter *Changes for Free* wird verstanden, dass der Auftraggeber kostenlos Änderungen fordern darf. Die Bedingungen hierfür sind, dass

- der Umfang der Änderung hinsichtlich des Umfangs geschätzt wird und dass
- eine Funktionalität oder Aufgabe mit ähnlich großem Umfang, die noch nicht umgesetzt wurde, im Tausch gegen die Änderung weggelassen wird.

Tipp

Vereinbarungen wie die *Changes for Free*-Klausel zeigen, dass Änderungen in agilen Projekten im Sinne eines besseren Projektergebnisses gewünscht sind. Aus den Regeln folgt aber auch: Kostenlose Änderungen gibt es nur im Tausch!

(No) *Money for Nothing* als weitere verbreitete Klausel in agilen Verträgen bedeutet, dass der Auftraggeber jederzeit das Recht hat, mit dem Projektgegenstand zufrieden zu sein, und das Projekt in diesem Fall frühzeitig kostenlos beenden kann. Kosten für ursprünglich vereinbarte, aber noch nicht implementierte Funktionalität werden nicht in Rechnung gestellt. Allerdings können Aufwände für begonnene Aufgaben oder Folgekosten beispielsweise für eine dadurch zu ändernde Produktdokumentation berechnet werden.

Das Prinzip (No) Money for Nothing stellt hohe Anforderungen an die organisatorische Reife. Ein Unternehmen muss ein frühzeitig beendetes Projekt mit Folgeaufträgen füllen und vorgezogen bearbeiten können.

Tipp

Agile Projekte müssen die beiden vorgestellten Vertragsklauseln nicht anwenden. Ihre Erläuterung hilft jedoch beim Verständnis dafür, wie weitreichend agiles Arbeiten sein kann.

Definition und Planung

Zieldefinition

Projekte zeichnen sich dadurch aus, dass sie einen gewissen Neuheitswert bezüglich ihrer Ziele oder des zur Zielerreichung notwendigen Lösungswegs haben. Der Neuheitswert geht mit einer Unsicherheit einher, die sich durch unklare Ziele oder unklare Lösungswege äußern kann.

Da die Ziele eines Projekts dessen Fundament darstellen und sich aus den übergeordneten Zielen Anforderungen und Maßnahmen ableiten lassen, sollte die Zielformulierung sorgfältig erfolgen.

Vorsicht

Die in der Praxis verbreitete Gleichsetzung von Agilität und Flexibilität führt manchmal dazu, dass eine sorgfältige Zieldefinition mit dem Satz »Wir arbeiten doch agil und legen die Ziele deshalb später fest« unterlassen wird. Auch wenn Agilität Änderungen berücksichtigt, verursachen Änderungen immer Aufwände, die von jemandem zu leisten und zu bezahlen sind.

Deshalb sollten Ziele, die sich mit etwas Mühe konkretisieren lassen, auch entsprechend konkret formuliert werden. Sind übergeordnete Ziele allerdings tatsächlich unklar, sollte mithilfe der Zielklärung eine Konkretisierung erwirkt werden.

Auch agil durchgeführte Projekte profitieren von gut formulierten Zielen, weshalb das SMART-Prinzip angewendet werden sollte. SMART steht hierbei für spezifisch, messbar, anspruchsvoll, realistisch und terminiert. Lassen sich Ziele zu Beginn eines Projekts noch nicht SMART formulieren, sollte eine spezielle Klärungsphase durchgeführt werden. Darin kann der Auftraggeber hinsichtlich seiner Vorstellungen befragt werden. Das Projektteam gibt die von ihm verstandenen Ziele dann zur Überprüfung möglichst anschaulich in eigenen Worten und Bildern wieder, um sie gemeinsam mit dem Auftraggeber zu reflektieren.

Tipp

Auftraggeber haben meist geringere Probleme damit, Kosten- und Terminziele vorzugeben, als konkrete Leistungs- und Qualitätsziele zu formulieren. Gerade hier entstehen aber häufig Missverständnisse. Je anschaulicher die Ziele beispielsweise durch Abbildungen, Animationen, Simulationen, 3D-Modellen etc. visualisiert werden, desto eher können Missverständnisse reduziert werden und desto eher ist der Auftraggeber bereit, sich selbst ausreichend mit der Ziel festlegung auseinanderzusetzen.

Anforderungen als User Story

In agilen Projekten werden Anforderungen gerne aus Nutzersicht formuliert. Die aus Scrum bekannten User Stories fördern diese Art der Formulierung und werden deshalb mittlerweile auch häufig in anderen agilen Vorgehensmodellen verwendet.

*Eine **User Story** ist eine Anforderung, die aus Sicht einer bestimmten Rolle, häufig der des Anwenders, geschrieben wird. Eine typische Formulierung lautet: Als Anwender möchte ich Dokumente mit nur einem Mausklick drucken können.*

Ein **Epic** ist eine große, noch vage User Story. Die Anforderung ist nur grob skizzierbar und keinesfalls hinsichtlich ihrer Größe (Komplexität) schätzbar.

Zur Formulierung guter User Stories kann die in Abbildung 1.18 illustrierte Schablone verwendet werden. Um die Anforderungen an den Projektgegenstand vollständig zu erfassen und als User Stories zu formulieren, müssen vorab alle relevanten Nutzerrollen identifiziert werden. Neben dem Endanwender können dies beispielsweise die eigenen Produktionsmitarbeiter, Servicetechniker und Vertriebsfachkräfte sein.



Abbildung 1.18: Schablone zur Formulierung von User Stories nach Cohn (Cohn 2009)

Typische User Stories unter Anwendung der Schablone können deshalb lauten:

- Als Nutzer der App will ich Waren mit der Bezahlungsfunktion meines Smartphones bezahlen können, um keine zusätzlichen Bankdaten eingeben zu müssen.
- Als Servicetechniker will ich das Maschinengehäuse mit dem im Standardwerkzeugkoffer verfügbaren Werkzeug öffnen können, sodass ich kein zusätzliches Spezialwerkzeug beschaffen und tragen muss.

Typischerweise werden User Stories auf Metaplankarten geschrieben und an Taskboards oder Kanbanboards aufgehängt. Beide Boards werden weiter unten in diesem Kapitel beschrieben.

Anforderungen als Tech Story

Sie kennen bereits das Timeboxing und wissen, dass hochpriorisierte Aufgaben zuerst bearbeitet werden. Meist werden kundenbezogene Funktionen und Funktionen, die nach außen besondere Strahlkraft entfalten, hoch priorisiert und entsprechend umgesetzt. Was aber ist mit Funktionen, die für das Funktionieren des Projektergebnisses wichtig sind, die der Kunde aber gar nicht wahrnimmt? Die gleiche Frage gilt für Funktionen, die dem Kunden nicht unmittelbar helfen, aber beispielsweise langfristig die Wartung eines Produkts vereinfachen? Noch schwerer haben es Aufgaben, die gar nicht erst mit einer unmittelbaren Funktion des Produkts in Verbindung stehen. Beispiele hierfür sind

- die Aktualisierung einer Softwarearchitektur,
- das Entfernen unnötiger Code-Bestandteile einer Software, damit diese übersichtlicher und wartungsfreundlicher wird, oder
- das Erstellen einer Dokumentenvorlage, damit Dokumente in künftigen Projekten einfacher erstellt werden.

Solche Aufgaben werden auch *Tech Stories* genannt. Meist kennt nur das Team selbst die Bedeutung dieser Aufgaben. Da die Priorisierung der Anforderungen an ein Produkt jedoch vom Kunden oder dessen unternehmensinternen Repräsentanten vorgenommen wird, haben solche Aufgaben kaum eine Chance, in einer Timebox berücksichtigt zu werden.

Eine Möglichkeit, solche langfristig wichtigen Aufgaben berücksichtigen zu können, ist das Festlegen eines bestimmten Zeitbudgets für Tech Stories. Damit kann sichergestellt werden, dass in jeder Timebox Zeit für interne Verbesserungen, das Aufräumen von Zwischenergebnissen und die regelmäßige interne Wartung des Produkts reserviert wird.

Definition of Done

Agile Projekte setzen auf sich selbst organisierende Teams. Um die Übergabe von Aufgaben von einem Teammitglied zum nächsten zu vereinfachen, ist es in agilen Projekten üblich, die Kriterien für die Fertigstellung einer Aufgabe festzulegen. Außerdem ist diese Festlegung wichtig, um zu definieren, welche Qualität eine Funktion oder ein Ergebnis aufweisen muss, um als fertig zu gelten. Diese Festlegung wird Definition of Done bezeichnet.

*Die **Definition of Done** beschreibt das Qualitätsverständnis eines Teams. Erst, wenn alle Qualitätsmerkmale erfüllt sind, ist die zugehörige Anforderung oder User Story abgeschlossen.*

Die Definition of Done kann auf Metaplankarten für jede Anforderung individuell festgelegt werden. Alternativ kann versucht werden, übergeordnete Kriterien festzuschreiben. So kann bei Softwareprojekten beispielsweise vermerkt werden, dass Programmcode grundsätzlich kommentiert werden muss, neue Funktionen im Firmen-Wiki zu dokumentieren sind und die Funktion vor Übergabe erfolgreich getestet werden muss. Allgemeine Definitionen sind aber nur für ähnlich geartete Aufgaben möglich.

Backlog

Während in planbasierten Projekten die zu erledigenden Aufgaben vollständig im Projektstrukturplan erfasst werden, erfolgt diese Sammlung in agilen Projekten häufig in sogenannten Backlogs:

*Ein **Backlog** bezeichnet eine Menge unerledigter Arbeit, die auf Bearbeitung wartet.*

Backlogs können als tabellarische Aufgabenlisten organisiert werden. Gebräuchlicher ist jedoch deren Visualisierung auf einer physischen oder digitalen Metaplanwand. Dann werden die anstehenden Aufgaben auf Karten geschrieben und den auf der Metaplanwand dargestellten Spalten »offen«, »in Arbeit« und »fertig« zugeordnet.

Eine konkrete Ausprägung eines Backlogs stellt das Kanbanboard dar, das neben den oben erläuterten Spalten weitere, flussspezifische Spalten aufweist, um die es später noch gehen wird. Sehr verbreitet sind das in Scrum übliche Product Backlog zur Sammlung der Kundenanforderungen, das Sprint Backlog zur Sammlung der Aufgaben für den nächsten Sprint und das Impediment Backlog zur Sammlung von Hindernissen, die das Team nicht selbst aus dem Weg räumen kann.

Kennzeichen eines Backlogs ist üblicherweise, dass das Team gleichberechtigt damit arbeiten kann, dass es sich dynamisch Veränderungen anpasst und dass es Wert auf eine kundenbezogene Aufgabenfokussierung legt.

Tipp

In der Praxis können die Übergänge der traditionellen Aufgabenplanung hin zum agilen Backlog fließend sein. Auch die in vielen traditionellen Projekten geführte *Liste offener Punkte* (auch kurz LOP-Liste genannt) stellt ein Backlog dar.

Größenschätzung

Die bekanntesten agilen Vorgehensmodelle wie Scrum und Kanban gehen nicht näher darauf ein, wie der Umfang einer Aufgabe zu ermitteln ist und in welcher Einheit er angegeben wird. Dennoch ist vor allem bei Scrum die Schätzung der Aufgabengröße weit verbreitet. Anstatt den Aufwand in Personenstunden oder Personentagen anzugeben, wird die Größe einer Aufgabe in der Einheit Story Points angegeben.

Die Größe ist eine relative Angabe über den Umfang einer Aufgabe. Diese berücksichtigt die Komplexität der Aufgabe und die Unsicherheit, die mit ihrer Bearbeitung einhergeht. Ausgeblendet werden soll hingegen der personenbezogene Aufwand, da zum Zeitpunkt der Schätzung bei sich selbst organisierenden Teams nicht klar ist, wer die Aufgabe später bearbeiten wird.

Zur Normierung der Einheit wird beispielsweise die kleinste zu bearbeitende Aufgabe ausgewählt und mit der Größe 1 Story Point versehen. Alle anderen (größeren) Aufgaben werden relativ zu dieser ausgewählten Aufgabe geschätzt.

Da ideale agile Teams über einen langen Zeitraum zusammenarbeiten, ergibt sich in der Schätzung eine gewisse Routine. Eingespielten Teams fällt diese Art der Schätzung nicht weiter schwer.

Für die Schätzung selbst können gängige Methoden der Aufwandsschätzung, wie die Delphi-Methode oder die Dreipunktschätzung ebenso eingesetzt werden wie spezielle agile Methoden, beispielsweise Planning Poker.

Task- und Kanbanboard

Das Agile Manifest betont die Wichtigkeit einer guten Zusammenarbeit und Kommunikation. Beides wird durch transparente und verständliche Pläne unterstützt. Hinzu kommt, dass Pläne einen anderen Stellenwert haben als in traditionellen Projekten: Nur die nähere, vorhersehbare Zukunft wird geplant und ein Plan muss so gestaltet sein, dass einfach Änderungen berücksichtigt werden können.

Eine in agilen Projekten verbreitete Art der Visualisierung von Plänen sind sogenannte Task- oder Kanbanboards.

Das **Taskboard** dient der Visualisierung des Sprint Backlogs bei Scrum. Die User Stories des aktuellen Sprints werden in Tasks aufgeteilt und meistens ihres Bearbeitungsstandes entsprechend in Spalten »offen«, »in Arbeit« und »abgeschlossen« sortiert. Verallgemeinert stellt das Taskboard die Aufgaben eines Projekts gemäß ihres Bearbeitungsstandes dar.

Das **Kanbanboard** ist das zentrale Visualisierungs- und Steuerungswerkzeug bei Kanban. Anders als beim Taskboard, bei dem der Status einzelner zu verrichtender Aufgaben dargestellt wird, wird beim Kanbanboard der Arbeitsfluss durch die einzelnen bearbeitenden Instanzen visualisiert. Für eine technische Anforderung kann dieser Fluss durch die Instanzen Anforderung, Design, Implementierung, Integration, Test, Dokumentation und Abgeschlossen visualisiert werden.

Ein Beispiel für beide Boards zeigt Abbildung 1.19. Beide Boards können auf einer Meta-planwand, einem Whiteboard oder mithilfe von Software auf einem Bildschirm realisiert werden. Gerade bei kleineren Teams, die an einem Ort zusammenarbeiten, werden diese Boards gerne als physisches Board realisiert. Das Team trifft sich dann direkt zur Stand-up-Besprechung an diesem Board, reflektiert den aktuellen Bearbeitungsstand und nutzt die Erkenntnisse daraus für die Projektsteuerung.

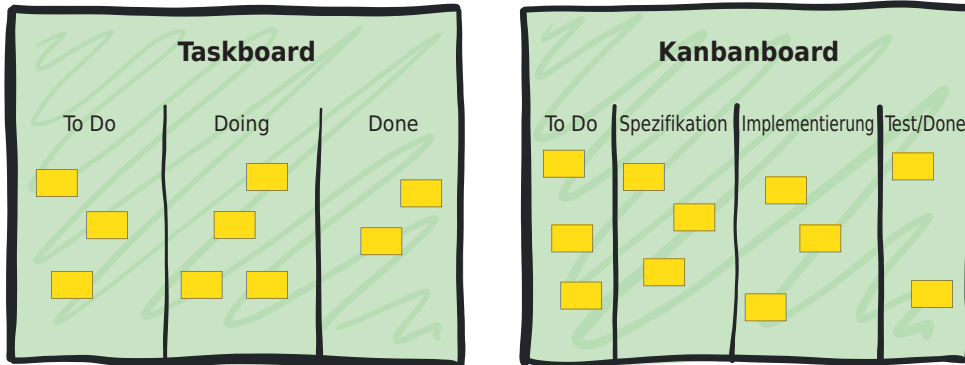


Abbildung 1.19: Beispiel eines Taskboards (links) und eines Kanbanboards (rechts)

Steuerung

Zielindikator

Ein Zielindikator ist eine Visualisierung der definierten Projektziele, beispielsweise auf einem Whiteboard oder Flipchart, bei dem das Team angibt, ob eine Zielerreichung nach aktuellem Stand des Projekts möglich erscheint, siehe Abbildung 1.20.

In diesem Beispiel werden wichtige Projektziele aufgelistet. Jedes Teammitglied markiert mit einem Strich bei »ja, ich halte die Zielerreichung für realistisch« oder bei »nein« seine Einschätzung über die Zielerreichungswahrscheinlichkeit. Alternativ können auch Punkte oder Prozentwerte vergeben werden.



Abbildung 1.20: Mögliche Darstellung eines Zielindikators

Ein für das gesamte Projektteam sichtbarer Zielindikator erfüllt mehrere Funktionen (Kniberg 2012):

- Er listet die wichtigsten Projektziele auf und ruft diese damit regelmäßig ins Gedächtnis.
- Durch das Feedback des Teams können Zweifel erkannt und analysiert werden sowie Maßnahmen zur Zielerreichung erarbeitet werden.

Die regelmäßige Konfrontation mit den Projektzielen hilft außerdem dabei, Arbeiten zu erkennen, die für die Zielerreichung irrelevant sind.

Stand-up-Besprechungen

In agilen Projekten ist die Interaktion untereinander wichtiger als Prozesse und Werkzeuge. Der beste Weg, Informationen zu streuen, ist das persönliche Gespräch (Beck 2001). Diese Werte und Prinzipien führen dazu, dass regelmäßige Stand-up-Besprechungen in agilen Projekten zum Standard gehören. Anstatt sich in größeren zeitlichen Abständen in langwierigen Besprechungen auszutauschen, kommt das Projektteam täglich zusammen und bespricht kurz und knapp

- was seit der letzten Stand-up-Besprechung erreicht wurde,
- was bis zur nächsten Stand-up-Besprechung erreicht werden soll und
- welche Hindernisse identifiziert wurden.

Der Name Stand-up-Besprechung ist wörtlich zu nehmen: Die Besprechungen finden im Stehen statt, was eine zügige Durchführung fördert. Mehr als 15 Minuten sollte eine solche Besprechung nicht dauern. Fragen, die nicht unmittelbar geklärt werden können, werden auf separate Besprechungen ausgelagert. Die Häufigkeit von Stand-up-Besprechungen wird in Abhängigkeit vom Projektgegenstand gewählt. Von Besprechung zu Besprechung muss es möglich sein, Fortschritt zu erzielen.

Burndown Chart

Das Burndown Chart ist ein Diagramm, in dem die noch offenen Arbeiten gegen eine Zeitachse aufgetragen werden. Die Zeitachse erstreckt sich vom Beginn einer Timebox bis zu deren Ende. Ein Beispiel zeigt Abbildung 1.21.

Die offene Arbeit kann grundsätzlich ganz traditionell in Personenstunden oder -tagen angegeben werden. In Scrum-Projekten ist die Angabe der offenen Arbeit in der Einheit Story Points verbreitet. Alternativ kann auch einfach die Anzahl noch nicht fertig bearbeiteter Karten auf dem Task- oder Kanbanboard gezählt werden. Letzteres bedingt allerdings, dass die Karten ungefähr gleich große Aufgaben repräsentieren, da das Ergebnis ansonsten verzerrt dargestellt wird.

Eine Timebox kann eine Projektphase, die Zeit zwischen zwei Meilensteinen oder eben ein Sprint bei einem Scrum-Projekt sein. Eine idealisierende Gerade verdeutlicht die lineare Abnahme der offenen Arbeit bis zum Ende der Timebox. Die tatsächliche Menge noch offener Arbeit wird ebenfalls zeitaufgelöst eingetragen. Verläuft diese Kurve über der idealisierenden Geraden, ist die vollständige Erledigung aller Aufgaben bis zum Ende der Timebox in

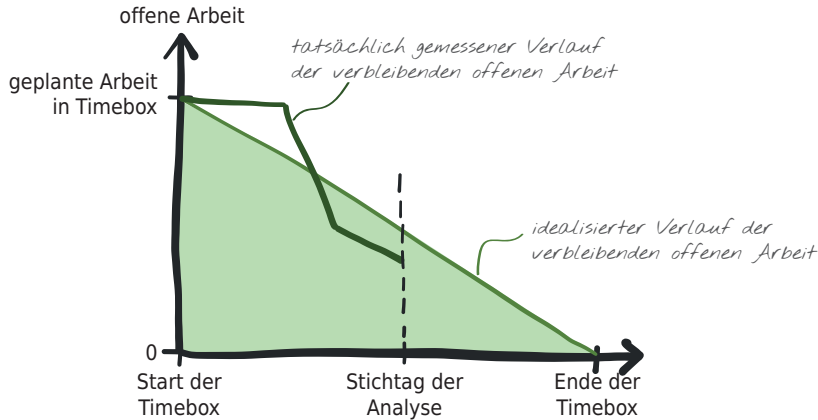


Abbildung 1.21: Beispiel eines Burndown Charts

Gefahr. Verläuft die tatsächliche Kurve unter der idealisierenden Geraden, schreitet das Projekt schneller voran als geplant.

Vorsicht

Häufig verläuft die tatsächlich gemessene Kurve offener Arbeit zu Beginn einer Timebox horizontal. Das Team muss sich erst in die neuen Aufgaben einarbeiten und überhaupt mit der Bearbeitung beginnen, bevor diese abgeschlossen werden. Nach kurzer Zeit sollte dann aber ein merklicher Rückgang der offenen Arbeit zu verzeichnen sein.

Kumulativer Fluss

Eine verbreitete Metrik zur Messung des Projektfortschritts und dessen Analyse ist der kumulative Fluss, siehe Abbildung 1.22.

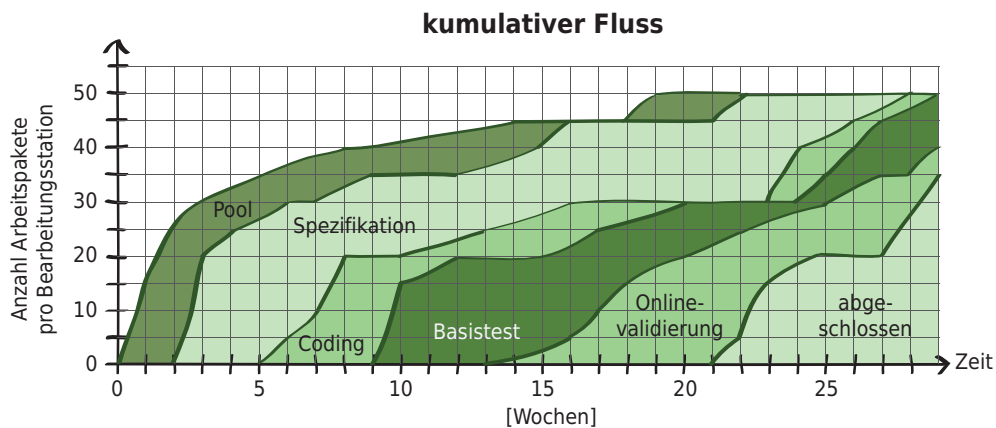


Abbildung 1.22: Beispielhafte Visualisierung des kumulativen Flusses

Zur Ermittlung des kumulativen Flusses werden für jede Bearbeitungsstation eines Teams die Anzahl dort offener Arbeiten gezählt. Die Zählung kann ähnlich wie beim Burndown Chart auf Basis gerade bearbeiteter Karten oder Story Points oder anhand der bearbeiteten Arbeitspakete erfolgen.

Beispiel

Betrachten wir zu Veranschaulichung des kumulativen Flusses die Situation in Woche 9 des in Abbildung 1.22 illustrierten Diagramms. In dieser Woche befinden sich 20 Arbeitspakete im »Coding«, 15 Arbeitspakete in der »Spezifikation« und noch weitere fünf Arbeitspakete im »Pool« komplett offener Aufgaben. Noch kein einziges Arbeitspaket wird gerade getestet, validiert oder ist gar abgeschlossen.

In Woche 20 hat sich die Situation wie folgt gewandelt: Es sind nach wie vor fünf Arbeitspakete im »Pool« offener Aufgaben und 15 Arbeitspakete in der »Spezifikation«. Allerdings befindet sich kein einziges Arbeitspaket im »Coding«, während nun zehn Arbeitspakete im »Basistest« und 20 Arbeitspakete in der »Online-Validierung« sind.

Der kumulative Fluss liefert keine Erklärung für die jeweiligen Situationen, kann aber Hinweise darauf geben, in welchen Bereichen nachgeforscht werden sollte. So ist in Woche 20 zu klären, weshalb kein einziges Arbeitspaket im »Coding« ist. Gründe können beispielsweise personelle Engpässe, fehlerhafte Spezifikationen oder Probleme mit der Entwicklungsumgebung sein.

Eine Bearbeitungsstation kann einer Spalte in einem Kanbanboard entsprechen oder aber ein Teilprojekt repräsentieren.

Abschluss

Einige Aspekte des Projektabschlusses unterscheiden sich nicht zwischen planbasiertem und agilem Projektmanagement. Der Projektgegenstand muss vom Projektteam auf den Auftraggeber übergehen. Wie formell das geschehen muss, hängt von vertraglichen Regelungen, dem Vertrauensverhältnis und den vereinbarten Formen der Zusammenarbeit ab. In jedem Fall gibt es rechtliche Aspekte, die auch in agilen Projekten gelten, wie den mit der Projektabnahme einhergehenden Gefahrenübergang.

Die Auflösung des Projektteams kann sich hingegen unterscheiden: Im agilen Ideal sollen Teams dauerhaft in gleicher (oder zumindest ähnlicher) Besetzung zusammenarbeiten. Eine Auflösung des Teams ist also nicht erforderlich. Stattdessen gilt es, den Übergang von der Bearbeitung eines Projekts zur Bearbeitung des nächsten Projekts so zu gestalten, dass das erste Projekt ordentlich abgeschlossen und das folgende Projekt adäquat gestartet wird. Tatsächlich lässt sich das agile Ideal der konstant zusammenarbeitenden Teams bei Weitem nicht in allen Konstellationen und Unternehmen realisieren. Dafür sind Projektgegenstände,

-umfänge und benötigte Kompetenzen häufig zu unterschiedlich. Dann müssen Teams wie beim planbasierten Projektmanagement geordnet aufgelöst und die Mitarbeiter neuen Aufgaben zugeordnet werden.

Auch Projektanalysen und die Weitergabe von Projekterfahrungen an andere Projekte und die Linienorganisation sind wichtig. Eine verbreitete und gleichzeitig einfache Form der Projektanalyse ist die Keep-Drop-Try-Methode, die nun kurz vorgestellt wird.

Keep-Drop-Try

Das Keep-Drop-Try-Schema ist eine einfache Methode zur Sammlung und Strukturierung von Lessons Learned. Die Teilnehmer sammeln auf Metaplankarten Aspekte, die ihnen während des zu analysierenden zurückliegenden Zeitraums aufgefallen sind. Diese werden dann den Kategorien **Keep**, **Drop** und **Try** zugeordnet.

Keep steht hierbei für Arbeitsweisen, die als gut bewertet wurden und in künftigen Projektphasen beibehalten werden sollten. Drop steht für Arbeitsweisen, die künftig nicht mehr angewendet werden sollen. In der Kategorie Try werden Ideen gesammelt, die künftig einmal ausprobiert werden sollen.

Aus den Beiträgen werden im nächsten Schritt konkrete Maßnahmen zur Veränderung hergeleitet und priorisiert. Für die als wichtig bewerteten Maßnahmen werden Verantwortliche benannt und deren Umsetzung wird besprochen.

Andere verbreitete Kategorien sind »more«, »less« und »same«. Diese sollen zum Ausdruck bringen, von welchen Gepflogenheiten, Methoden und Arbeitsweisen mehr, weniger oder gleich viel gewünscht wird.

Auf einen Blick

- Vorgehensmodelle des Projektmanagements bieten geeignete Methoden für unterschiedliche Konstellationen und Situationen.
- Projekte, deren Rahmenbedingungen stabil sind und die daher gut geplant werden können, bedienen sich sogenannter planbasierter Methoden. Diese werden manchmal auch traditionelle oder klassische Methoden des Projektmanagements genannt.
- Verbreitete planbasierte Methoden sind die Projektstrukturplanung, die Terminplanung sowie die Kosten- und Ressourcenplanung.
- Sind die Rahmenbedingungen nicht stabil oder sind Änderungen der Anforderungen an den Projektgegenstand absehbar, bieten agile Methoden des Projektmanagements eine gute Möglichkeit des Umgangs mit derartigen Situationen.
- Agile Methoden basieren auf agilen Werten und Prinzipien und erarbeiten den Projektgegenstand meist iterativ.

Übungsfragen und Beispiele

Aufgabe 1.1: Planbasiert oder agil?

Die Leiterin der Abteilung Organisation und Beschaffung gründet in ihrem Bereich ein Projektteam und beauftragt es damit, den Beschaffungsprozess zu optimieren. Die Kriterien zur Bewertung der Optimierung stehen ebenso wenig fest wie der beabsichtigte Umfang der Optimierung. Hierfür sollen vom Projektteam Vorschläge erarbeitet werden. Herausfordernd empfindet das Projektteam, dass mehrere Abteilungen eingebunden werden müssen, wie beispielsweise die Produktion, die Entwicklung und das Marketing.

Würden Sie sich auf Basis der hier vorliegenden Informationen für ein planbasiertes oder agiles Vorgehen des Projekts entscheiden?

Aufgabe 1.2: Phasenplanung und Projektziele

Der Auftraggeber eines Projekts gibt als Ziel vor, dass das Projekt bis Ende Juli abgeschlossen sein muss. Der Projektleiter erstellt auf Basis aller vorliegenden Informationen einen Phasenplan, der ein Projektende frühestens Mitte November realistisch erscheinen lässt. Was würden Sie dem Projektleiter nun raten?

Aufgabe 1.3: Projektplanung I

Wie lautet die chronologisch korrekte Abfolge der Pläne bei der Planung planbasierter Projekte? Muss diese Abfolge in jedem Fall eingehalten werden? Was ist, wenn sich während der Abarbeitung der Pläne neue Erkenntnisse ergeben, beispielsweise ein erhöhter Aufwand für die Bearbeitung eines Vorgangs?

Aufgabe 1.4: Projektplanung II

Weshalb gibt es bei einfachen Meilensteinplänen gute Gründe sowohl für das Setzen von vielen als auch für das Setzen möglichst weniger Meilensteine?

Aufgabe 1.5: Aufwandsschätzung

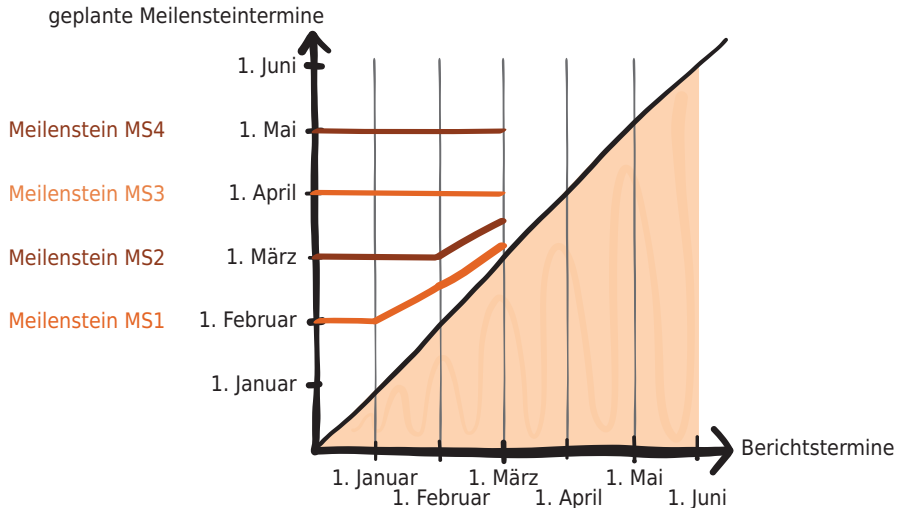
Sollten die Aufwände aller Vorgänge eines Projekts mit der gleichen Methode geschätzt werden? Welche Methode(n) bieten sich für folgenden Vorgang an: Im Rahmen der Entwicklung eines neuen Produkts sollen verschiedene Fügeverfahren (zum Beispiel Schweißen, Löten oder Kleben) analysiert und bewertet werden, um eine verlässliche und gleichzeitig in der späteren Serienfertigung einfache Verbindung zweier Bauteile herzustellen. Da es sich um neue Materialien handelt, existieren hierfür noch wenig Erfahrungen im Unternehmen.

Aufgabe 1.6: Earned-Value-Analyse

Ein Projekt bestehe aus mehreren Arbeitspaketen, die größtenteils noch nicht bearbeitet wurden. Nur die Arbeitspakete mit den Nummern 1.2.2 ($BAC_{1.2.2} = 10.000 \text{ €}$), 1.3.1 ($BAC_{1.3.1} = 20.000 \text{ €}$) und 1.3.5 ($BAC_{1.3.5} = 20.000 \text{ €}$) sind bereits abgeschlossen. Die geplanten Gesamtkosten des Projekts betragen 400.000 €. Von diesen sind laut Kostenplan noch 360.000 € offen. Tatsächlich verausgabt wurden bisher 60.000 €. Berechnen Sie den Kosten- und den Terminentwicklungsindex und interpretieren Sie sie.

Aufgabe 1.7: Meilensteintrendanalyse

Betrachten Sie folgenden Meilensteinverlauf bis zum Stichtag 1. März. Nennen Sie mindestens zwei Aspekte, die Ihnen hinsichtlich der Termintreue des weiteren Projektverlaufs Sorge bereiten.



Aufgabe 1.8: Projektabschluss

Können Sie als Auftraggeber die Abnahme des Projektgegenstands bei vorhandenen Fehlern generell verweigern? Wie würden Sie mit einer solchen Situation als Auftraggeber umgehen?

Aufgabe 1.9: Kontinuierliche Aufgaben

Im Rahmen der Planung einer Gartenparty müssen Sie folgende Situation abwägen: Die aktuelle Wettervorhersage sagt unbeständiges Wetter voraus. Sollte es zu einem unwahrscheinlichen, aber möglichen Regenschauer kommen, würden die Gäste schnell das Wohn- und Esszimmer aufsuchen. Es ist zu befürchten, dass im Nachgang eine Spezialreinigung in der Größenordnung von mehreren Hundert Euro notwendig wird. Lohnt sich vor diesem Hintergrund die Beschaffung eines Gartenpavillons?

Aufgabe 1.10: Agiler Mindset

Ein Projektteam nutzt das Vorgehensmodell Scrum, hat dies aber einigen unternehmensspezifischen Besonderheiten angepasst. So wird das Team von einem Projektmanager geführt, der wöchentlich an den Auftraggeber berichten muss. Das Team darf Aufgaben selbstorganisiert planen und steuern. Kommt es jedoch zu Planabweichungen, müssen Maßnahmen zur Planerfüllung erarbeitet und dem Steuerkreis zur Freigabe vorgelegt werden. Da der unternehmensinterne Auftraggeber insbesondere die Flexibilität agilen Arbeitens schätzt, kommt es regelmäßig während der Bearbeitungsphase zu Änderungen von User Stories.

Wie bewerten Sie den agilen Mindset der Projektbeteiligten und welche Probleme in der Anwendung des agilen Vorgehensmodells sehen Sie?

Aufgabe 1.11: Priorisierung

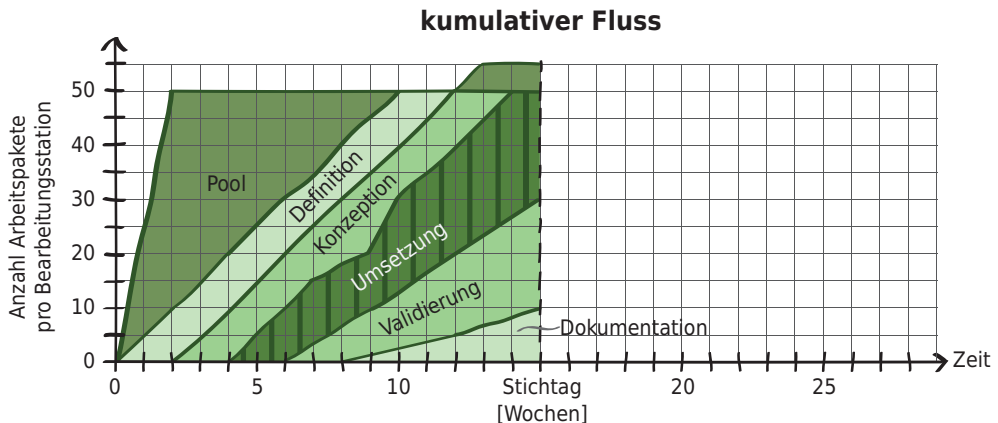
Ein agil arbeitendes Projektteam pflegt ein Anforderungs-Backlog. Dieses wird vom Auftraggeber priorisiert und vom Projektteam unter Würdigung dieser Priorisierung bearbeitet. Nur selten bleibt die Priorisierung unberücksichtigt, beispielsweise wenn sachlogische Zwänge eine andere Bearbeitungsreihenfolge erfordern. Dies ist der Fall, wenn mehrere Funktionen voneinander abhängen. Dann wird zunächst die Grundfunktionalität, die die Voraussetzung für andere Funktionen darstellt, implementiert.

Bei den regelmäßigen Projektreflexionen fällt auf, dass bestimmte vom Projektteam gewünschte Funktionen häufig unberücksichtigt bleiben. Beispiele für Funktionen, die vom Projektteam gewünscht werden, für den Auftraggeber aber keinen unmittelbaren Nutzen haben, sind Diagnosewerkzeuge zur Fehlerlokalisierung und eine Architektur für einfachere Aktualisierungen und Wartungsarbeiten.

Wie würden Sie mit dem vom Team geschilderten Problem umgehen?

Aufgabe 1.12: Kumulativer Fluss

Betrachten Sie nachfolgend illustrierten kumulativen Fluss eines Projekts am Stichtag nach 15 Wochen Projektlaufzeit.



Beschreiben und beurteilen Sie die Situation des Projekts.

Literaturverzeichnis

- Anderson, David J. (2010): Kanban. Successful evolutionary change for your technology business. Sequim, Washington: Blue Hole Press.
- Beck, Kent et al. (2001): Manifesto for Agile Software Development. Online verfügbar unter <http://agilemanifesto.org/>, zuletzt geprüft am 22.02.2018.
- Boehm, Barry (1979): Guidelines for verifying and validating software requirements and design specifications. In: EURO IFIP 79. North Holland, S. 711–719.
- Cohn, Mike (2009): User stories applied. For agile software development. 13. print. Boston Mass, u.a.: Addison-Wesley (The Addison-Wesley signature series).

- ISO/IEC 20926 (2009): Software and systems engineering—Software measurement—IFPUG functional size measurement method. 2. Aufl.: Beuth Verlag (35.080).
- Kniberg, Henrik (2012): Lean from the Trenches: O'Reilly Media.
- Komus, Ayelt (2017): Abschlussbericht: Status Quo Agile 2016/2017. 3. Studie über Erfolg und Anwendungsformen von agilen Methoden. BPM Labor Hochschule Koblenz. Online verfügbar unter <https://www.hs-koblenz.de/index.php?id=1932>, zuletzt geprüft am 20.04.2017.
- McConnell, Steve (2006): Software estimation. Demystifying the black art. Redmond, Washington: Microsoft Press (Safari Books Online). Online verfügbar unter <http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10762188>.
- Patzak, G.; Rattay, Günter (2018): Projektmanagement. Projekte, Projektportfolios, Programme und projektorientierte Unternehmen. 7., aktualisierte Auflage. Wien: Linde Verlag.
- DIN 69901-2, 2009: Projektmanagement – Projektmanagementsysteme – Teil 2: Prozesse, Prozessmodell.
- DIN EN ISO 9001, 2015: Qualitätsmanagementsysteme – Anforderungen.
- Royce, Winston (1970): Managing the Development of Large Software Systems. In: Proceedings IEEE WESCON, S. 328–338.
- Schwaber, Ken (1997): Scrum Development Process. In: Jeff Sutherland, Cory Casanave, Joaquin Miller, Philip Patel und Glenn Hollowell (Hg.): Business Object Design and Implementation. OOPSLA '95 Workshop Proceedings 16 October 1995, Austin, Texas. London: Springer London.
- Schwaber, Ken; Beedle, Mike (2002): Agile software development with Scrum. Upper Saddle River, NJ: Prentice Hall (Series in agile software development).
- Sutherland, Jeff; Casanave, Cory; Miller, Joaquin; Patel, Philip; Hollowell, Glenn (Hg.) (1997): Business Object Design and Implementation. OOPSLA '95 Workshop Proceedings 16 October 1995, Austin, Texas. London: Springer London. Online verfügbar unter <http://dx.doi.org/10.1007/978-1-4471-0947-1>.
- Takeuchi, Hirotaka; Nonaka, Ikujiro (1986): The New New Product Development Game. In: *Harvard Business Review* (Jan.–Feb.).
- Timinger, Holger (2017): Modernes Projektmanagement. Mit traditionellem, agilem und hybridem Vorgehen zum Erfolg. 1. Auflage: Wiley-VCH.
- Wertanalyse. Idee-Methode-System (2011). 6., völlig neu bearb. u. erw. Aufl. Berlin: Springer Berlin (VDI-Buch).
- Wysocki, Robert K. (2014): Effective project management. Traditional, agile, extreme. 7th ed. Indianapolis, Indiana: Wiley. Online verfügbar unter <http://lib.mylibrary.com?id=550382>.

