

Scrum

Inhalt

So funktioniert Scrum: Rollen, Regeln, Methoden	2
Wichtige Begriffe, Elemente und Tools für Scrum	11
Scrum-Projekt: Der Ablauf Schritt für Schritt mit Beispielen	17
Product Backlog erstellen & planen mit Vorlage	35
Retrospektive: Leitfaden mit Beispielen, Methoden & Praxistipps	46
Probleme und Fehler bei Scrum, die Sie kennen sollten	59
Empfehlungen aus dem Management-Handbuch	64
Autoren	66

So funktioniert Scrum: Rollen, Regeln, Methoden

Scrum ermöglicht im agilen Projektmanagement die Festlegung einfacher, klarer Regeln. Doch welche Rollen gibt es bei Scrum? Welche Methoden werden eingesetzt? Wie läuft der Scrum-Prozess ab? Was bedeutet das Wörtchen „agil“ im Kontext Scrum? Und was haben Kanban-Board und Burndown-Chart damit zu tun?

Zuletzt geändert am 27.04.2023

Von Dr. Jürgen Fleig, b-wise GmbH



Worum geht es bei Scrum?

Viele Projekte scheitern. Es wird geplant und dokumentiert, aber die Ziele erreicht man am Ende nicht. Scrum und andere Methoden des agilen Projektmanagements versprechen Abhilfe. Der Kerngedanke: Wenn Methoden, Werkzeuge und Vorgehensweisen aus dem klassischen Projektmanagement nicht zu den erwünschten Erfolgen führen, dann lassen wir es einfach.

Stattdessen vertrauen Unternehmen darauf, dass ein erfahrenes Projektteam es schon irgendwie schaukeln wird, wenn es die Freiheiten dazu hat. Die Voraussetzung dafür: Der Projektfortschritt muss für alle transparent sein und die Projektmitarbeiter stimmen sich täglich ab.

Was ist Scrum?

Scrum ist ein Modell oder eine Methode des agilen Projektmanagements. Der Begriff stammt aus dem Rugby. Dort meint Scrum einen dichten Haufen von Spielern, die sich um das Spielgerät scharen (deutsch: Gedränge).

Übertragen auf das agile Projektmanagement bedeutet das für Scrum: Alle Mitarbeiterinnen und Mitarbeiter eines Projekts treffen sich jeden Tag, um über anstehende Aufgaben und den bisherigen Verlauf im Projekt zu sprechen.

Dazu kommen weitere Regeln und Methoden, die bei der Zielerreichung helfen. Diese Herangehensweise soll die manchmal schwerfällige und unflexible Projektplanung ersetzen und das Projektmanagement agiler machen.

Stichwort

Agiles Projektmanagement

Agiles Projektmanagement umfasst unterschiedliche Methoden, die vor allem auf Flexibilität und Anpassung setzen. Statt ausführlicher und umfangreicher Planung zu Beginn eines Projekts werden das adaptive Planen sowie die schnelle Abstimmung im Team unterstützt. Agiles Projektmanagement hat insbesondere bei der Softwareentwicklung an Bedeutung gewonnen.

Warum ist man mit Scrum agil?

In einem Scrum-Projekt gelten wenige und einfache Regeln. Sie sind nur dafür da, das Projektteam zu unterstützen, um das gemeinsame Projektziel zu erreichen. Die wichtigste Regel ist, dass ein Team beim agilen Projektmanagement sich selbst organisieren kann und darf. Das Projektteam bekommt einen Auftrag und legt mit dem Auftraggeber die Projektziele fest.

Danach kann es agil die eigene Projektarbeit organisieren. Es gibt keinen fertigen Projektplan, den das Projektteam abarbeiten muss.

Methoden und Regeln beim Projekt-Sprint

Das Projekt läuft in mehreren Phasen ab. Eine Phase wird als Projekt-Sprint bezeichnet. In jedem Sprint gibt es die täglichen Meetings, das Daily Scrum.

Es gibt verschiedene Methoden und Tools, die bei der Prüfung und Durchführung der einzelnen Aufgaben und Aktivitäten des Sprints helfen. Mit ihnen wird auch geprüft, was nach einem Sprint erreicht wurde. Besonders bekannt: das Product Backlog, das Sprint Backlog, das Kanban-Board, das Sprint Review und das Burndown-Chart.

Immer wieder wird die Zusammenarbeit im Team reflektiert. Das nennt man Sprint Retrospective.

Welche Rollen gibt es bei Scrum-Projekten?

Es gibt bei Scrum drei Rollen:

- der Produkteigner (Product Owner)
- der Scrum-Master
- das Mitglied im Projektteam

Produkteigner

Der Produkteigner vertritt die Anwender des Produkts oder die Stakeholder des Projekts. Das sind alle, die betroffen sind und ein Interesse am Erfolg des Projekts haben. Wenn eine neue Software oder IT-Lösung im Unternehmen eingeführt wird, sind das die Nutzer, denn sie wollen reibungslos mit ihren Programmen arbeiten. Bei Produkten sind es die Produktmanager, die als Stimme der Kunden auftreten. Der Produkteigner sollte wissen, was die Kunden haben wollen. Aber auch das Marketing, der Vertrieb und der Kundendienst können Anforderungen an das Projektteam stellen.

Scrum-Master (Project-Master)

Der Scrum-Master trägt die Verantwortung für den Scrum-Prozess. Der Scrum-Master ist Moderator und Unterstützer für das Projektteam. Er beseitigt Hindernisse und fördert die gute Zusammenarbeit im Team. Er beschafft die notwendigen Ressourcen und ist Ansprechpartner für Außenstehende. Und er hilft dem Team bei methodischen Problemen und stellt sicher, dass die Regeln des agilen Projektmanagements eingehalten werden.

Projektteam

Ein Scrum-Team sollte zwischen fünf und zehn Mitarbeiterinnen und Mitarbeitern umfassen. Die Teammitglieder organisieren alle Aufgaben selbst. Es gibt im Team keine Hierarchie. Jeder hat dieselben Rechte und Pflichten, aber unterschiedliche Kompetenzen. Alle Fachbereiche, die zur Lösung beitragen, sollten vertreten sein. Wichtig ist, dass alle Teammitglieder aus eigenem Antrieb dabei sind. Sie sollten sich ihre Projekte selbst aussuchen können. Das setzt Vertrauen des Managements und Verantwortungsbewusstsein der Mitarbeitenden voraus.

Neben den Rollen ist es vor allem der Prozess, der kennzeichnend ist für Scrum.

Die 5 Schritte des Scrum-Prozesses

Am Anfang steht eine Produkt-Vision; eine Idee des Produkts, die der Auftraggeber des Projekts vorantreiben will und die auch den Anwendern einen Nutzen bringt. Eine solche grobe Vorstellung vom Produkt oder der Lösung, die im Scrum-Projekt erarbeitet werden soll, ist der Auftrag. Die Produkt-Vision wird in Story Cards oder User Stories überführt, die aus Sicht des Anwenders einzelne Elemente, Merkmale und Funktionen des Produkts beschreiben.

Mit dieser Grundlage startet ein agiles Projekt nach Scrum. Dann umfasst der Scrum-Prozess folgende Schritte:

1. Product Backlog anlegen und pflegen

Aus den Anforderungen des Produkteigners und den Story Cards wird ein sogenanntes Product Backlog zusammengestellt. Das ist eine Sammlung sämtlicher Funktionen und Merkmale, die das Produkt haben soll. Am Anfang ist diese Zusammenstellung noch grob, doch im Projektverlauf wird sie immer genauer.

Im Product Backlog werden Prioritäten vergeben. Eine hohe Priorität erhalten die Elemente und Funktionen die am wichtigsten sind und eine hohe Zufriedenheit der Anwender sicherstellen. Andere Anforderungen sind nicht so wichtig, können aussortiert werden, werden mit anderen zusammengelegt, sind technisch nicht realisierbar oder werden verschoben. Sie werden dann bei der Überarbeitung oder bei der Erweiterung des Produkts behandelt.

2. Im Sprint Planning das Sprint Backlog erstellen

Agiles Projektmanagement mit Scrum ist ein iterativer Prozess, der sich aus mehreren Sprints zusammensetzt, bis das gewünschte Produkt fertiggestellt ist. Der Sprint ist der Kern eines Scrum-Projekts. Er ist eine fest vorgegebene Zeitdauer (Time Box) von maximal einem Monat.

Im Sprint Planning werden die Aufgaben für den als Nächstes anstehenden Sprint geplant und das Sprint-Ziel formuliert. Sie ergeben sich aus den Einträgen auf dem Product Backlog und den Prioritäten. Mit dem Sprint Planning wird geklärt:

- Was wird im nächsten Sprint entwickelt, erstellt oder durchgeführt?
- Wie werden die entsprechenden Aufgaben und Arbeiten erledigt?

Das Scrum-Team erstellt daraus ein Sprint Backlog. Das ist eine Auswahl der Product-Backlog-Einträge für den nächsten Sprint ergänzt um den Umsetzungsplan. Außerdem ist klar, woran sichtbar ist, ob die Aufgabe erledigt und das Teil-Produkt (Inkrement) des Sprints erstellt ist. Die sogenannte „Definition of Done“ wird formuliert.

Eine einzelne Aufgabe, die dann zu erledigen ist, wird Ticket genannt. Alle Tickets sind im sogenannten Sprint Backlog aufgeführt. Das ist gewissermaßen der Maßnahmenplan und der Arbeitsvorrat für das Entwickler-Team für den nächsten Sprint. Jedes Teammitglied übernimmt eigenverantwortlich einzelne Tickets (Verpflichtungserklärung). Im Sprint arbeiten die Teammitglieder an ihren Aufgaben, den Tickets, bis diese fertig sind und das „Done“ erreicht ist.

3. Im Daily Scrum den Arbeitsfortschritt besprechen

Während des täglichen, 15-minütigen Treffens, dem sogenannten Daily Scrum, berichtet jeder der Reihe nach: Was er seit dem letzten Daily Scrum gemacht hat, was er bis zum nächsten Daily Scrum tun wird und was ihn bei seiner Arbeit behindert. Dabei haben alle das Sprint-Ziel im Blick und prüfen, ob es noch erreicht werden kann. Der Scrum-Master muss die Hindernisse aufgreifen und helfen, dass sie beseitigt werden. In einem Sprint-Burndown-Chart wird sichtbar gemacht, wie das Projekt voranschreitet.

4. Im Sprint Review die Sprint-Ergebnisse prüfen und abnehmen

Jeder Sprint wird durch ein Sprint-Review-Meeting abgeschlossen. Das Team stellt die Ergebnisse und die Teil-Produkte dem Produkteigner vor. Er prüft, ob sie den Kriterien entsprechen, die mit der Definition of Done festgelegt wurden. Ist das der Fall, nimmt er die Sprint-Ergebnisse ab. Der Eintrag im Sprint Backlog ist damit abgehakt. Gleichzeitig werden die Einträge im Product Backlog aktualisiert oder angepasst.

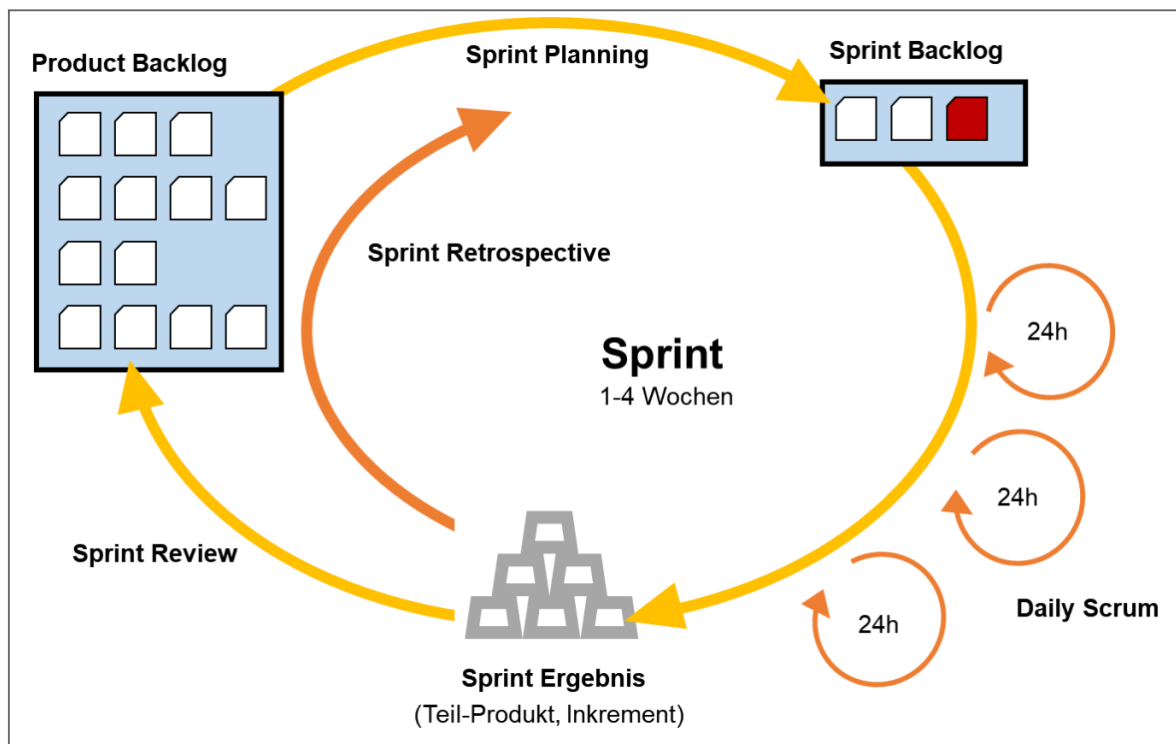
Mit den Ergebnissen aus dem Sprint Review und mit dem überarbeiteten Product Backlog kann der nächste Sprint starten. Der Prozess beginnt wieder mit Schritt 2. Es folgen so viele Sprints, bis das Produkt entwickelt und das Projekt abgeschlossen ist.

5. Mit einer Sprint Retrospective die Zusammenarbeit besprechen

In gesonderten Treffen zwischen einem Sprint Review und dem nächsten Sprint Planning können die Teammitglieder besprechen, wie der Sprint in Bezug auf die Zusammenarbeit der beteiligten Personen, Abläufe, Kommunikation und Werkzeuge

verlief. Sie halten fest, was für den nächsten Sprint verbessert werden sollte. Die Erkenntnisse sollten für zukünftige Scrum-Projekte genutzt werden; so wird ein Lernprozess unterstützt.

Alle Schritte im Scrum-Prozess sind in der folgenden Abbildung zusammengefasst.



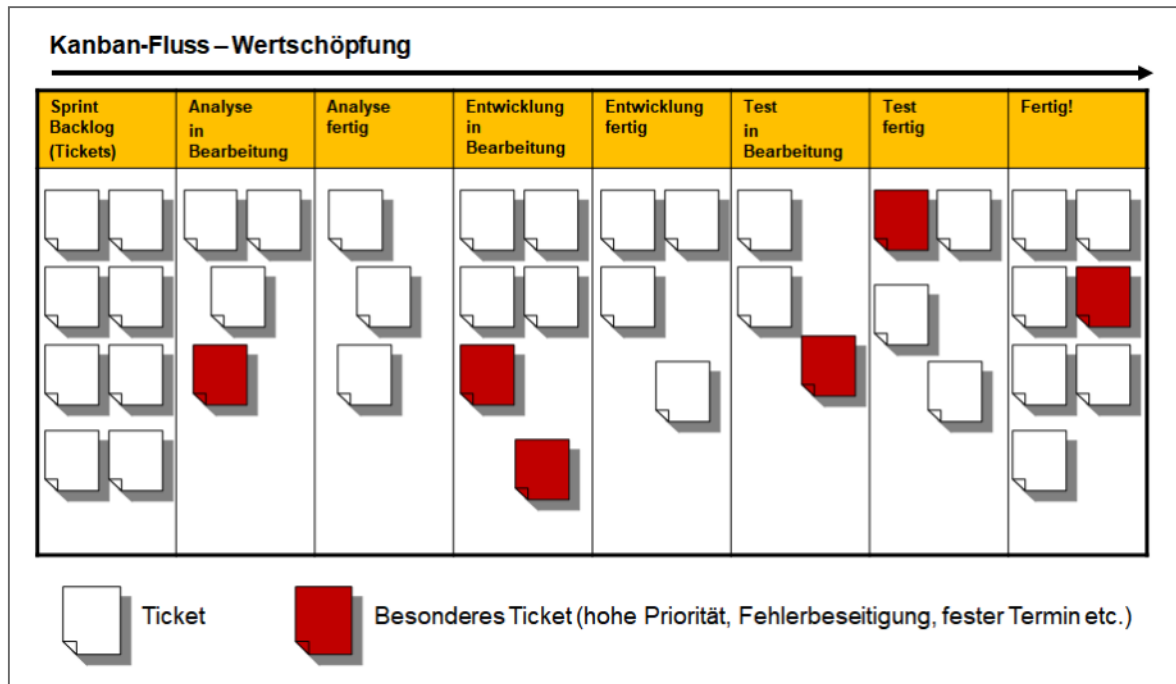
Scrum-Prozess und Sprint im Überblick

Das Kanban-Board zeigt den Projektfortschritt

Damit alle Mitarbeiterinnen und Mitarbeiter im Projekt eine Übersicht über den Projektstand und den Verlauf haben, werden bei Scrum alle Tickets eines Sprint Backlogs auf einer Tafel festgehalten (Task-Board). Dort wird sichtbar, was noch ansteht, was gerade in Bearbeitung ist und was schon fertig ist. Das Sprint Backlog, der Aufgabenplan, hängt also für alle sichtbar an der Wand. Mit der Task-Board kann der Projektablauf transparent gesteuert werden. Sie wird dann Kanban-Board genannt.

Nach dem Kanban-Prinzip werden nur so viele Tickets für die Bearbeitung freigegeben, wie vom Projektteam bearbeitet werden können. Geht es an einer Stelle nicht voran, stauen sich dort die Tickets. Es wird schnell sichtbar, wo der Engpass liegt und was

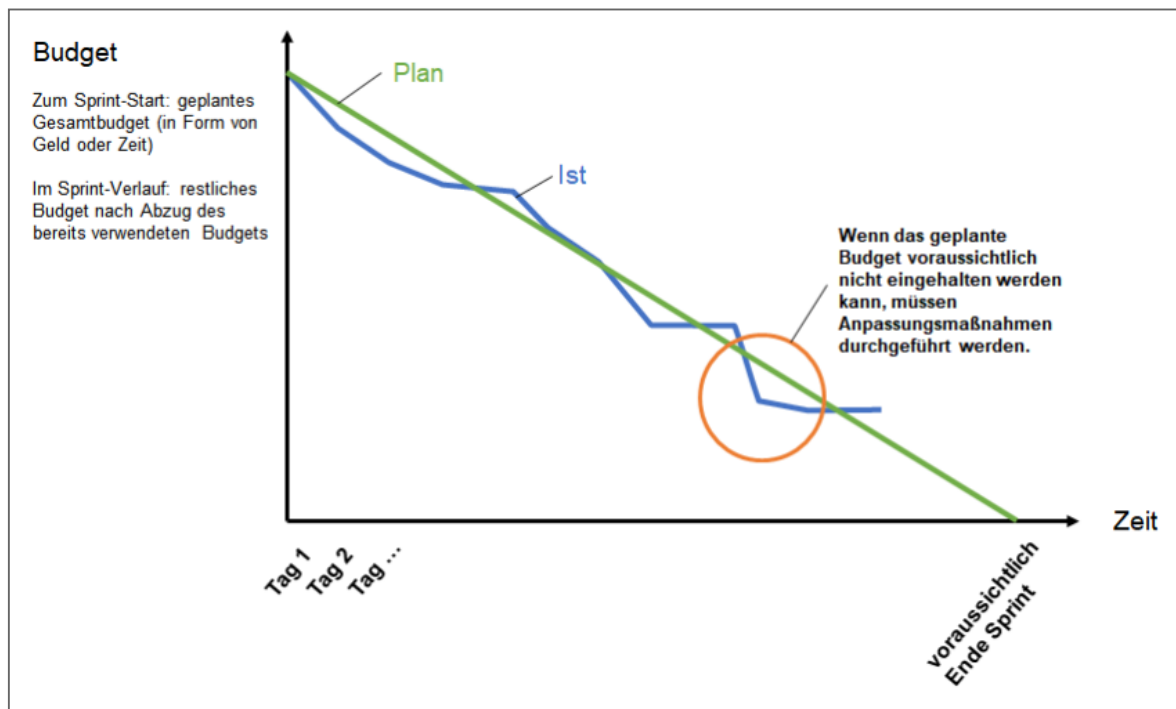
getan werden kann. Eilige Tickets können vorgezogen werden. So steuert das Team die Aufgabenverteilung und den Arbeitsfluss völlig selbstständig mit dem Kanban-Board als Wandtafel.



Kanban-Board als detailliertes Sprint-Task-Board

Das Burndown-Chart hilft bei der Zeit- und Kostenkontrolle

Für das Projektcontrolling und die Budgetüberwachung in einem Sprint wird bei Scrum ein Burndown-Chart eingesetzt. Es ist eine einfache Abbildung, die sichtbar macht, ob das Projekt im ursprünglich vereinbarten Zeit- und Kostenrahmen bleibt oder ob Budget oder Zeitplan angepasst werden müssen. Durch die täglichen Treffen sind der Budgetverbrauch und der Projektfortschritt immer aktuell und Korrekturmaßnahmen können sehr schnell eingeleitet werden.



Burndown-Chart für das Controlling im Scrum-Sprint

Praxis

Sie wollen ein Projekt nach den Methoden und Regeln des agilen Projektmanagements und mit Scrum durchführen? Dann sollten Sie zunächst prüfen, inwiefern die Rahmenbedingungen und Voraussetzungen gegeben sind, die es bei Scrum braucht.

Projektmitarbeiter und ihre Rollen

Klären Sie die Kompetenzen, die Qualifikationen, das Know-how und den Erfahrungsschatz der Mitarbeiterinnen und Mitarbeiter, die für das Projekt zur Verfügung stehen:

- Welche Kompetenzen werden benötigt?
- Wer hat welche Stärken und welche Schwächen?

Ermitteln Sie insbesondere, welche Fähigkeiten oder Kompetenzen im Projektteam fehlen, die im Projekt aber benötigt werden; das sind die Lücken im Diagramm der folgenden Vorlage.

**Stärken-und-Schwächen-Profil der Teammitglieder bei Scrum**

Art.-Nr. 99.143.01

Datei business-wissen_9914301.xlsx

Microsoft Excel 2016

Weisen Sie die geeigneten Mitarbeiterinnen und Mitarbeiter den einzelnen Scrum-Rollen zu. Klären Sie, welche User oder Customer der Produkteigner (Product Owner) vertritt. Nutzen Sie für die Zuweisung der einzelnen Rollen und Aufgaben nach Scrum die folgende Vorlage.

**Rollen und Personen im Scrum-Projekt**

Art.-Nr. 99.143.02

Datei business-wissen_9914302.xlsx

Microsoft Excel 2016

Wichtige Begriffe, Elemente und Tools für Scrum

Projekte laufen mit Scrum anders als beim klassischen Projektmanagement ab. Scrum basiert auf wenigen Regeln und Prinzipien. Wichtig ist, dass die Projektbeteiligten ihre Rollen und Aufgaben wahrnehmen und dass der Scrum-Prozess transparent und mit viel Kommunikation abläuft. In einem Scrum-Projekt werden spezielle Werkzeuge eingesetzt, die im Folgenden erläutert werden.

Zuletzt geändert am 03.02.2023

Von Dr. Jürgen Fleig, b-wise GmbH



Wichtige Begriffe, Elemente und Tools für Scrum

Im Rahmen von Scrum werden oft Begriffe verwendet, die spezielle **Artefakte (Elemente) oder Werkzeuge** bezeichnen. Sie sind dazu da, um für das Scrum-Team maximale Transparenz und minimalen Verwaltungsaufwand zu gewährleisten. Sie helfen also, alle Informationen übersichtlich, einfach und verständlich zu vermitteln. Und sie helfen bei der Zusammenarbeit und Abstimmung im Scrum-Team.

User Storys und Story Cards

Die Nutzer (User) oder Kunden (Customer) schreiben in ihren Worten die Anforderungen, Wünsche oder Ziele an das Produkt oder die zu entwickelnde Lösung als ihre User Story auf Story Cards.

Product Backlog

Liste oder Sammlung aller Aufgaben, Funktionen und Merkmale des Produkts oder der Lösung, die aus den Story Cards abgeleitet werden. Sie können wie ein Use Case beschrieben werden.

Backlog-Item

Die einzelne Aufgabe, Funktion oder das einzelne Merkmal im Product Backlog.

Story Points

Wenn die zunächst groben Anforderungen der Nutzer oder Kunden durch den Product Owner geklärt sind, muss das Projektteam abschätzen, wie komplex eine Aufgabe, Funktion oder Merkmal ist. Daraus leitet sich der Aufwand für die Realisierung ab. Diese Schätzung wird in sogenannten Story Points ausgedrückt. Wie viele Stunden Entwicklungsaufwand mit einem Story Point verbunden sind, ergibt sich aus der Erfahrung oder aus dem Projektverlauf.

Things-That-Matter-Matrix

Tabelle, die Beziehungen zwischen Anforderungen der User oder Customer einerseits und den Backlog-Items andererseits darstellt.

Definition of Done

Im Vorfeld muss sich das Scrum-Team darüber verständigen, wann ein Projekt- oder Entwicklungsergebnis als „fertig“ oder „erledigt“ bezeichnet werden kann. Dazu werden Indikatoren, Funktionen oder Tests benannt, die nach der Entwicklung des Ergebnisses durchgeführt werden und bestätigen, dass es funktioniert.

Sprint

Einzelner Arbeitszyklus (Teilschritt oder Phase) im Projekt, in dem jeder Projektmitarbeiter die von ihm ausgewählte Teilaufgabe (Sprint-Task) bearbeitet und eine fertige Lösung für diesen Task liefert.

Time Box

Für einen Sprint wird eine feste Zeitdauer vorgegeben, die nicht überschritten werden darf. Dieser Zeitrahmen wird als Time Box bezeichnet. Er beträgt meistens zwischen einer und vier Wochen.

Sprint-Task (Ticket)

Teilaufgabe, die von einem Projektmitarbeiter bearbeitet wird und deren Ergebnis ein Baustein (Teillösung, Funktion, Feature) für das Produkt oder das Gesamtprojekt ist.

Sprint-Backlog

Liste aller Sprint-Tasks; Maßnahmenplan für den nächsten Sprint.

Sprint-Taskboard

Eine Art Plantafel oder Tabelle, auf der alle Teilaufgaben eines Sprints aufgeführt sind. Sie ist für alle Projektmitarbeiter jederzeit sichtbar. Die Teilaufgaben werden dort nach Bearbeitungsstand sortiert.

Daily Scrum

Tägliches Zusammenkommen aller Mitarbeiterinnen und Mitarbeiter im Scrum-Team, um den Arbeitsfortschritt zu besprechen.

Impediment Backlog

Liste aller Probleme und Hindernisse, die während eines Sprints aufgetaucht sind und die beseitigt werden müssen.

Burn-Down-Chart

Grafik oder Diagramm, das den Verbrauch der Ressourcen darstellt, indem der geplante Aufwand dem tatsächlichen Aufwand gegenübergestellt wird. Es gibt ein Burn-Down-Chart für die Budgetüberwachung in einem Scrum-Zyklus sowie ein Burn-Down-Chart für das Gesamtprojekt; für das Gesamtprojekt wird gelegentlich auch ein Burn-Up-Chart verwendet.

Sprint-Review-Meeting

Nach einem Sprint stellen die Projektmitarbeiter ihre Lösungen des Sprints dem Produkteigner und den Kunden und Nutzern vor. Diese müssen die Lösung abnehmen und akzeptieren, wenn das gewünschte Ergebnis erzielt ist. Ansonsten wird nachgearbeitet.

Product Increment

Ein Teilprodukt oder Teilergebnis, das im Rahmen eines Sprints entwickelt oder erzielt wird, ist ein Product Increment. Es ist insoweit vollständig, als es vom Anwender und Product Owner geprüft und abgenommen werden kann.

Velocity

Ist die Geschwindigkeit, mit der das Projekt- und Entwicklerteam in den Sprints vorankommt. Sie zeigt, welche Arbeitsmenge geschafft und wie viele Story Points in den einzelnen Sprints erfüllt oder erledigt werden. Dies ist im Velocity-Chart visualisiert.

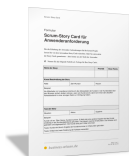
Sprint-Retrospective

In dieser gesonderten Besprechung wird die Zusammenarbeit während eines Sprints beleuchtet. Probleme werden angesprochen und Lösungen für die Prozessverbesserung werden gesucht.

Praxis

Vorlagen als Werkzeuge für Ihr Scrum-Projekt

Die folgenden Word- und Excel-Vorlagen können Sie als Werkzeuge in Scrum nutzen. Damit halten Sie die relevanten Informationen fest und machen diese für das Projektteam transparent. Wie die Vorlagen und Werkzeuge im Scrum-Prozess eingesetzt werden, wird im folgenden Abschnitt erläutert.



Story Card für Anwenderanforderung im Scrum-Projekt

Art.-Nr. 99.143.18

Datei business-wissen_9914318.docx

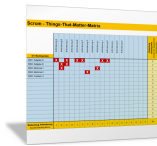
Die Datei wurde erstellt für Word 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab Word 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

**Product Backlog mit Backlog-Items und User Storys**

Art.-Nr. 99.143.03

Datei business-wissen_9914303.xlsx

Microsoft Excel 2016

**Things-That-Matter-Matrix für Projektaufgaben**

Art.-Nr. 99.143.04

Datei business-wissen_9914304.xlsx

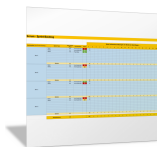
Microsoft Excel 2016

**Things-That-Matter-Matrix mit Aufwandsschätzung**

Art.-Nr. 99.143.05

Datei business-wissen_9914305.xlsx

Microsoft Excel 2016

**Sprint Backlog mit Statusüberwachung und Aufwandsschätzung**

Art.-Nr. 99.143.06

Datei business-wissen_9914306.xlsx

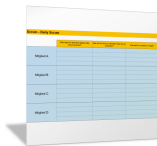
Microsoft Excel 2016

**Sprint-Taskboard mit Statusüberwachung für Maßnahmen und Teilaufgaben**

Art.-Nr. 99.143.07

Datei business-wissen_9914307.xlsx

Microsoft Excel 2016

**Persönlicher Statusbericht im täglichen Scrum**

Art.-Nr. 99.143.08

Datei business-wissen_9914308.xlsx

Microsoft Excel 2016

**Impediment Backlog mit Hindernissen und Problemen**

Art.-Nr. 99.143.09

Datei business-wissen_9914309.xlsx

Die Datei wurde erstellt für Excel 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab Excel 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

**Burn-Down-Chart (Liniendiagramm)**

Art.-Nr. 99.143.10

Datei business-wissen_9914310.xlsx

Microsoft Excel 2016

**Burn-Down-Chart (Säulendiagramm)**

Art.-Nr. 99.143.11

Datei business-wissen_9914311.xlsx

Microsoft Excel 2016

**Burn-Down-Chart mit variablem Zielkorridor**

Art.-Nr. 99.143.12

Datei business-wissen_9914312.xlsx

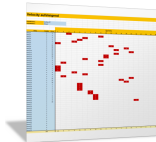
Microsoft Excel 2016

**Burn-Down-Chart mit Planung von Zusatzaufwand**

Art.-Nr. 99.143.13

Datei business-wissen_9914313.xlsx

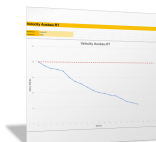
Microsoft Excel 2016

**Velocity-Chart: Arbeitsfortschritt im Scrum-Projekt aufsteigend**

Art.-Nr. 99.143.20

Datei business-wissen_9914320.xlsx

Microsoft Excel 2016

**Velocity-Chart: Arbeitsfortschritt im Scrum-Projekt absteigend**

Art.-Nr. 99.143.21

Datei business-wissen_9914321.xlsx

Microsoft Excel 2016

Im folgenden Abschnitt wird der **Ablauf eines Scrum-Projektschritt für Schritt** erläutert. Ausgangspunkt ist der Anlass oder Auftrag für ein Projekt. Daraus wird ein Product Backlog entwickelt, das dann in mehreren Sprints abgearbeitet wird, bis das Projekt fertig ist.

Scrum-Projekt: Der Ablauf Schritt für Schritt | mit Beispielen

Scrum läuft nach einfachen Regeln ab: Wenn die Anforderungen und Aufgaben für das Projekt geklärt sind, verteilen die Projektmitarbeiter alle Aufgaben in eigener Verantwortung. Sie treffen sich täglich, um den Projektfortschritt zu besprechen. Ist ein Sprint (Arbeitszyklus) abgeschlossen, wird das Ergebnis überprüft. Die Projektmitarbeiter reflektieren dabei, was gut und was schlecht läuft im Projekt.

Zuletzt geändert am 03.02.2023

Von Dr. Jürgen Fleig, b-wise GmbH



Warum wird ein Scrum-Projekt begonnen?

Am Anfang steht eine **Produkt-Vision**. Das ist eine grobe Vorstellung vom Produkt oder Projektergebnis, die die Geschäftsleitung oder der Product Owner als Auftraggeber hat. Für eine solche Produkt-Vision und Vorstellung gibt es meist einen konkreten Anlass oder Auslöser.

Beispiele dafür sind:

- Anwender oder Kunden sind mit einem bestehenden Produkt nicht mehr zufrieden, sie haben andere Anforderungen und Wünsche, weil Wettbewerber das auch bieten.
- Es gibt neue, bessere technische Möglichkeiten für ein bestehendes oder ganz neues Produkt, die man nutzen will.
- Technik ist fehleranfällig, Prozesse sind schwerfällig und teuer; das soll verbessert werden.
- Es gibt eine gesetzliche Anforderung oder andere Rahmenbedingungen, die es notwendig machen, dass etwas umgesetzt oder geändert wird.

Letztlich wollen Geschäftsleitung und Product Owner ein Produkt oder ein Projektergebnis, das einen größeren Nutzen bringt oder Schaden für das Unternehmen abwendet. Mit solchen Auslösern und Anlässen startet ein Scrum-Projekt.

Das läuft dann aber anders ab als ein klassisches Projekt; es gelten die Regeln des agilen Projektmanagements. Der Scrum-Prozess ist darauf ausgerichtet, die Anforderungen der Auftraggeber, der Nutzer, der Kunden und des Product Owners, möglichst korrekt zu erfassen, dann ohne großen Planungsaufwand zu bearbeiten und am Ende funktionierende Ergebnisse zu liefern. Im Einzelnen werden dazu folgende Schritte durchgeführt.

1. Anforderungen für das Scrum-Projekt ermitteln und in User Storys und Story Cards festhalten

Aus der ersten, aber möglichst schon anschaulichen und verständlichen Produkt-Vision und der groben Produkt-Vorstellung lassen sich einzelne Funktionen und Merkmale des Produkts oder der Lösung ableiten, die im Projekt zu entwickeln sind. Die Kunden und Nutzer beschreiben ihre Anforderungen und Erwartungen dazu als **User Story** aus ihrer Anwenderperspektive und in ihren Worten – nicht im Jargon der Techniker.

Daraus werden die Funktionen und Merkmale abgeleitet, die das Produkt haben soll. Sie werden auf einer **Story Card** formuliert. Die Story Card ist das Medium, auf dem die User Story möglichst exakt erfasst und dargestellt wird. Die User Storys und Story Cards können im Gespräch oder in einem Workshop mit dem Product Owner und gegebenenfalls mit ausgewählten Nutzern oder Kunden ermittelt und formuliert werden.

Mit diesen Anforderungen wird bereits deutlich, welche Ressourcen das Projekt braucht. Das betrifft zum einen die Art der Qualifikationen und des Know-hows im Projektteam und zum anderen eine grobe Schätzung des Aufwands für die Entwicklung des Produkts oder der Lösung.



Tipp: So formulieren Sie gute User Storys und Story Cards

Wie Sie die Kundenanforderungen beim agilen Projektmanagement und Scrum ermitteln und auf einer Story Card beschreiben, erfahren Sie in diesem Beitrag zur [Formulierung einer User Story](#). Außerdem lesen Sie zum [Thema Lastenheft](#), wie Sie Anforderungen mithilfe der SOPHIST-Satzschablone exakt und nachvollziehbar formulieren können.

2. Product Backlog und Backlog-Items erarbeiten

Aus den Anforderungen des Product Owners und der Nutzer oder Kunden und aus den User Storys wird ein **Product Backlog** zusammengestellt. Das ist eine Sammlung sämtlicher Funktionen und Merkmale, die das Produkt oder die Lösung für das Projekt haben soll.

Im Product Backlog können außer Merkmalen und Funktionen auch Aufgaben oder Arbeitspakete für das Scrum-Projekt zusammengestellt sein. Eine einzelne Aufgabe, eine Funktion oder ein Merkmal wird als Backlog-Item bezeichnet.



Tipp: Wie Sie das Product Backlog als Use Case beschreiben

Für die Beschreibung von Product Backlogs können Sie die Methode der Use Cases nutzen. Im Beitrag zu [User Story und Use Case](#) erkennen Sie an einem Beispiel, wie ein Use Case formuliert ist.

Am Anfang ist die Beschreibung der einzelnen **Backlog-Items** noch grob. Denn im Projektteam ist noch nicht genau geklärt, welche Teilaufgaben mit einer Lösung verknüpft sind oder welche Eigenschaften die Funktion im Detail haben soll. Das ergibt sich, wenn das jeweilige Backlog-Item entwickelt oder ausgearbeitet wird.

Dazu müssen gegebenenfalls weitere Gespräche mit dem Nutzer, Kunden oder Project Owner geführt werden. Im Projektverlauf werden die einzelnen Aufgaben, Funktionen und Merkmale deshalb immer detaillierter formuliert und entsprechend beschrieben.

3. Definition of Done festlegen

Dann ist es notwendig, bei der Beschreibung des Backlog-Items genau zu klären und festzulegen, was zeigt, dass das Item fertig ist. Dies wird mit der Beschreibung einer „Definition of Done“ erfüllt. Dazu werden Indikatoren, Funktionen oder Tests benannt, die nach der Entwicklung des Ergebnisses durchgeführt werden und bestätigen, dass die Lösung funktioniert oder ein Produktmerkmal vorhanden ist.

4. Entwicklungsaufwand mit Story Points darstellen

Wenn die Anforderungen der Nutzer, Kunden oder des Product Owners geklärt sind, muss das Projektteam abschätzen, wie komplex eine Aufgabe, Funktion oder Merkmal, also das einzelne Backlog-Item, ist. Die Komplexität ist meistens ein Indikator für den (zeitlichen) Aufwand für die Realisierung. Diese **Schätzung** wird bei Scrum in **Story Points** ausgedrückt.

Ein Backlog-Item mit 10 Story Points ist doppelt so komplex und aufwendig wie ein Backlog-Item mit 5 Story Points. Wie viele Stunden Entwicklungsaufwand mit einem Story Point verbunden sind, ergibt sich aus der Erfahrung oder aus dem Projektverlauf. Für eine Zeit- und Kostenkalkulation kann ein Story Point mit einem Zeit- oder Geldfaktor verknüpft werden.

5. Prioritäten für Backlog-Items vergeben

Im nächsten Schritt werden Prioritäten vergeben. Eine **hohe Priorität** erhalten die Aufgaben, Merkmale und Funktionen, die zur Zufriedenheit der Anwender besonders viel beitragen, indem sie deren besonders wichtigen Anforderungen erfüllen. Maßgeblich ist auch, dass das Produkt und das Projektergebnis insgesamt am Ende des Projekts „funktionieren“ und eine Lösung sind, die für den Anwender brauchbar ist und einen Vorteil bringt.

Eine **geringere Priorität** erhalten solche Aufgaben, Merkmale und Funktionen, die mit Anforderungen verbunden sind, die für den Anwender nicht so wichtig sind. Sie werden mit anderen zusammengelegt, werden verschoben, zurückgestellt oder gar nicht entwickelt, weil sie nicht in den Zeitrahmen und ins Budget passen.

Möglich ist, dass diese weniger wichtigen Aufgaben, Merkmale und Funktionen bei einer Überarbeitung oder bei der Erweiterung des Produkts oder der Lösung entwickelt werden. Im Bereich der Software-Entwicklung spricht man von einem Release-Wechsel.

6. Ablauf des Projekts besprechen

In einer ersten Besprechung des Scrum-Masters mit seinem Projektteam zum Projektablauf werden die Rahmenbedingungen abgesteckt und ein gemeinsames Verständnis vom Produkt und den wichtigen Aufgaben sowie vom zeitlichen Ablauf entwickelt. Diese Fragen werden dazu besprochen:

- Wie soll der allgemeine Aufbau des Produkts oder der Lösung aussehen?
- Was sind die Meilensteine?
- Welche Funktionen und Aufgaben müssen bis zu welchem Zeitpunkt in jedem Fall erledigt sein?
- Was sind die Teilziele für eine einzelne Funktion oder Aufgabe?
- In welche Teilschritte, Sprints, lässt sich das Projekt untergliedern?
- Auf welche Details oder Besonderheiten ist zu achten?
- Welche Konventionen und Schnittstellen sind einzuhalten?
- Wo und wann kann sich das Team regelmäßig für den Daily Scrum (täglich!) treffen?

Ein wesentliches Merkmal von Scrum-Projekten ist, dass die Ablaufplanung nur grob ist und kein großer Aufwand in die Detailplanung investiert wird; denn im Laufe des Projekts wird sich ohnehin vieles wieder ändern. Wichtiger ist, dass ein gemeinsames Verständnis im Projektteam dazu entsteht, was die K.-o.-Kriterien sind, welche Ziele besonders wichtig sind und welche Meilensteine beachtet werden müssen.

7. Sprint Planning: Einzelne Aufgaben bestimmen

Wenn die Ziele und Rahmenbedingungen geklärt sind, erfolgt das Sprint Planning. Dazu bespricht das Projektteam, wie die Backlog-Items aus dem Product Backlog in Einzel- oder Teilaufgaben zerlegt werden. Diese werden dann innerhalb sogenannter Sprints geplant und bearbeitet.

Ein **Sprint** ist ein Arbeitszyklus (Teilschritt oder Phase) im Projekt, in dem jeder Projektmitarbeiter die von ihm ausgewählte Teilaufgabe (Sprint-Task) bearbeitet und eine fertige Teillösung dafür liefert. Die Zeit für einen Sprint hängt vom Umfang und von der Komplexität der jeweils bearbeiteten Aufgaben ab. Sie wird als sogenannte **Time Box** festgelegt und kann ein bis vier Wochen umfassen.

Abbildung 1 zeigt im Überblick, wie dieser Kernprozess im Scrum-Projekt, der Sprint, funktioniert.

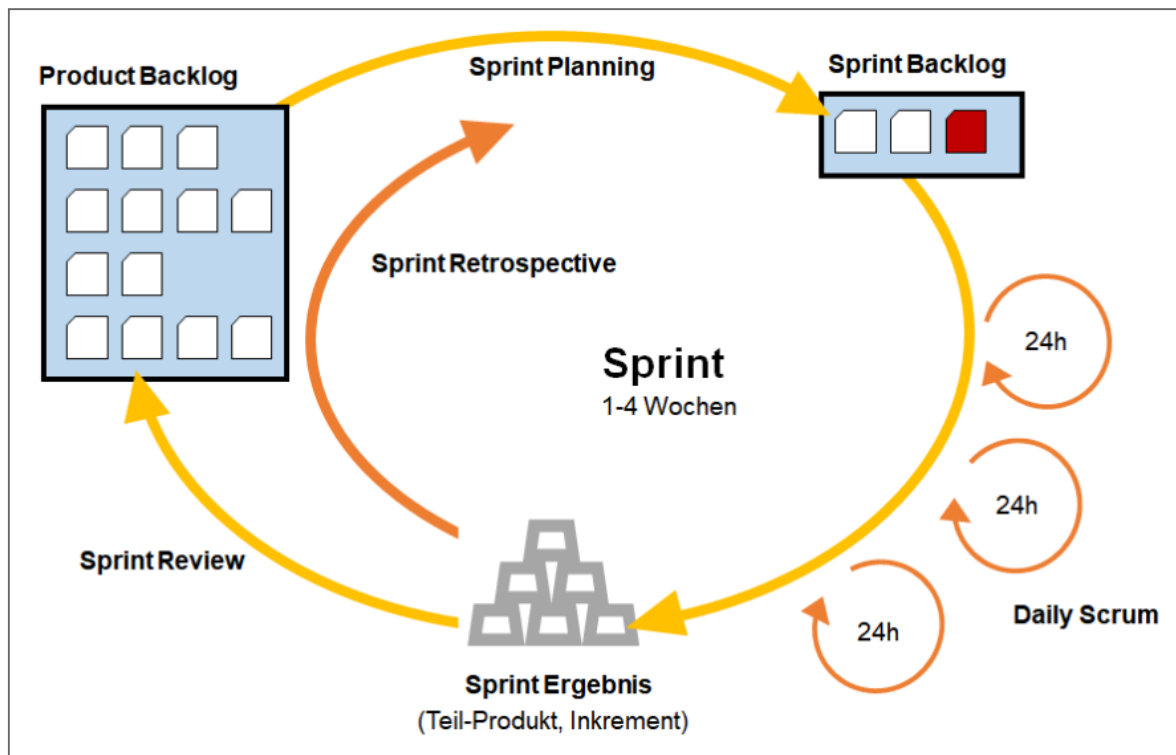


Abbildung 1: Scrum-Prozess und Sprint im Überblick

Das Ergebnis des Sprint Planning sind die Sprint-Tasks oder **Tickets**. Diese Teilaufgaben werden im Sprint Backlog (Maßnahmenplan) aufgeführt und am Sprint-Taskboard visualisiert. Das **Sprint Backlog** und das **Taskboard** sind gewissermaßen der Maßnahmenplan und der Arbeitsvorrat für das Entwicklerteam für den nächsten Arbeitszyklus (Sprint). Jedes Teammitglied übernimmt eigenverantwortlich eine Teilaufgabe beziehungsweise ein Ticket und verpflichtet sich damit, dafür eine Lösung zu entwickeln oder zu finden.

8. Aufgabendurchlauf mit dem Sprint Taskboard oder Kanban-Board steuern

Die Planung und der Durchlauf der einzelnen Aufgaben (Sprint-Tasks) wird mit einem Sprint-Taskboard für alle Projektmitglieder sichtbar. Das Taskboard kann in Form eines

Kanban-Boards stärker untergliedert und verbessert werden. Das Kanban-Board macht anschaulich, in welcher **Phase** sich eine Sprint-Task oder ein Ticket gerade befindet. Es bildet die Wertschöpfungskette des Projekts ab.

Wichtig ist: Es werden nur so viele Tickets für die Bearbeitung freigegeben, wie vom Projektteam bearbeitet werden können. Geht es an einer Stelle nicht voran, stauen sich dort die Tickets. Es wird schnell sichtbar, wo der **Engpass** liegt und was getan werden kann. Eilige Tickets können beispielsweise vorgezogen werden. So steuert das Team die Aufgabenverteilung und den Arbeitsfluss völlig selbstständig.

Abbildung 2 zeigt den Aufbau eines Kanban-Boards im Scrum-Projekt. An einer Wandtafel sind alle einzelnen Aufgaben bis zum nächsten Meilenstein auf Karten formuliert. Engpässe oder eilige Tickets können farblich markiert werden. So sind der Arbeitsablauf, der Stand der Arbeiten und die Engpässe für alle immer sichtbar.

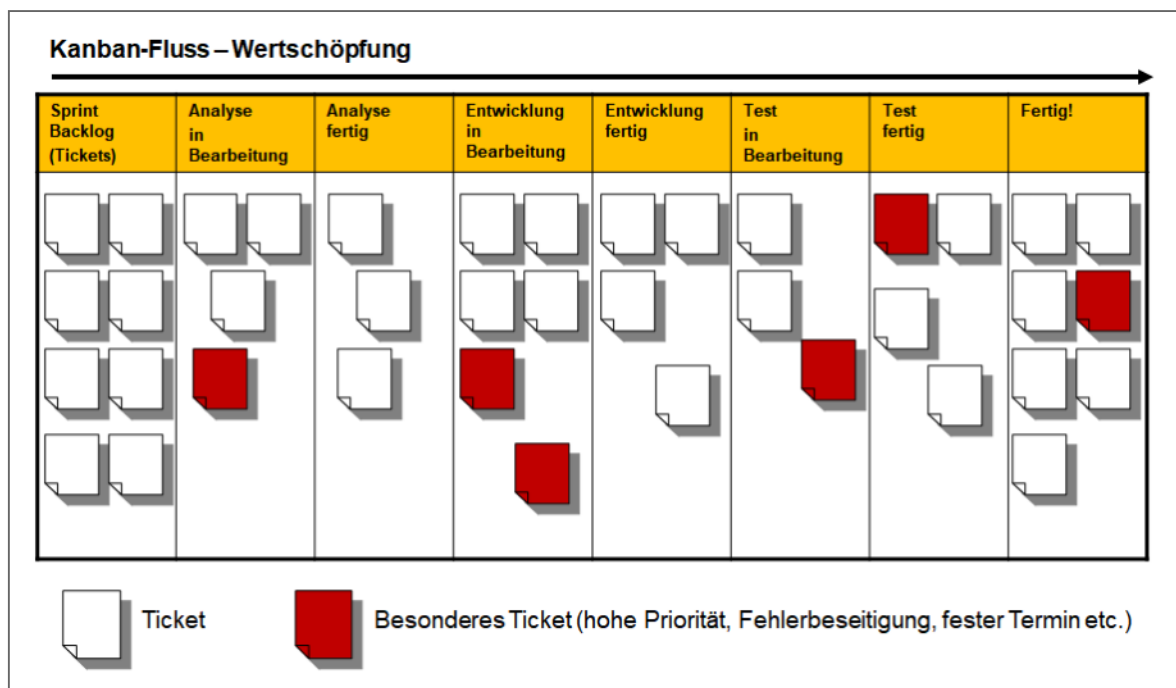


Abbildung 2: Kanban-Board als detailliertes Sprint-Taskboard

9. Aufgaben erledigen

Im Sprint arbeiten die Teammitglieder an ihren Aufgaben (Sprint-Tasks). Die Projektmitarbeiter und Entwickler orientieren sich dabei an den Story Cards, mit deren Hilfe sie prüfen und testen können, ob sie eine gute Lösung entwickelt haben. Sie haben dafür den mit der Time Box definierten Zeitrahmen.

Die Teammitglieder haben deshalb den Zeitplan, den Aufwand und die Meilensteine im Blick, um im vereinbarten Zeitrahmen und Kostenrahmen zu bleiben. Die Story Points zeigen, wie viel Budget zur Verfügung steht.

10. Daily Scrum: Lösungen regelmäßig besprechen

Während eines täglichen, **15-minütigen Treffens**, dem eigentlichen Daily Scrum (Gedränge), berichtet jeder der Reihe nach:

- Was wurde seit dem letzten Scrum gemacht?
- Was ist bis zum nächsten Scrum zu tun?
- Was behindert bei der Arbeit oder was verhindert die weitere Bearbeitung?

Die Probleme werden im Impediment Backlog festgehalten. Der Scrum-Master muss diese Hindernisse aufgreifen und helfen, dass sie beseitigt werden. Die Projektmitglieder können mit dem Arbeitsfortschritt erkennen, ob der geplante Zeitaufwand, wie er in den Story Points dargestellt ist, dem tatsächlichen Aufwand entspricht. Ungeplanter Aufwand und seine Ursachen werden beim Daily Scrum besprochen.

11. Plan- und Ist-Aufwand im Burn-Down-Chart darstellen

In einem Sprint-Burn-Down-Chart wird sichtbar gemacht, wie das Projekt voranschreitet. Dazu wird festgehalten, welche Ressourcen oder Arbeitszeiten bereits verbraucht wurden. Dieser Ist-Aufwand wird mit dem Plan-Aufwand verglichen und in einem Diagramm, dem Burn-Down-Chart bei Scrum, dargestellt.

Das Burn-Down-Chart ist eine einfache Abbildung, die sichtbar macht, ob das Projekt im ursprünglich vereinbarten **Zeit- und Kostenrahmen** bleibt. Oder ob Budget oder Zeitplan angepasst werden müssen. Durch die täglichen Treffen können diese sehr aktuell geführt und Korrekturmaßnahmen sehr schnell eingeleitet werden. Abbildung 3 zeigt, wie ein Burn-Down-Chart für einen Sprint aussehen kann.

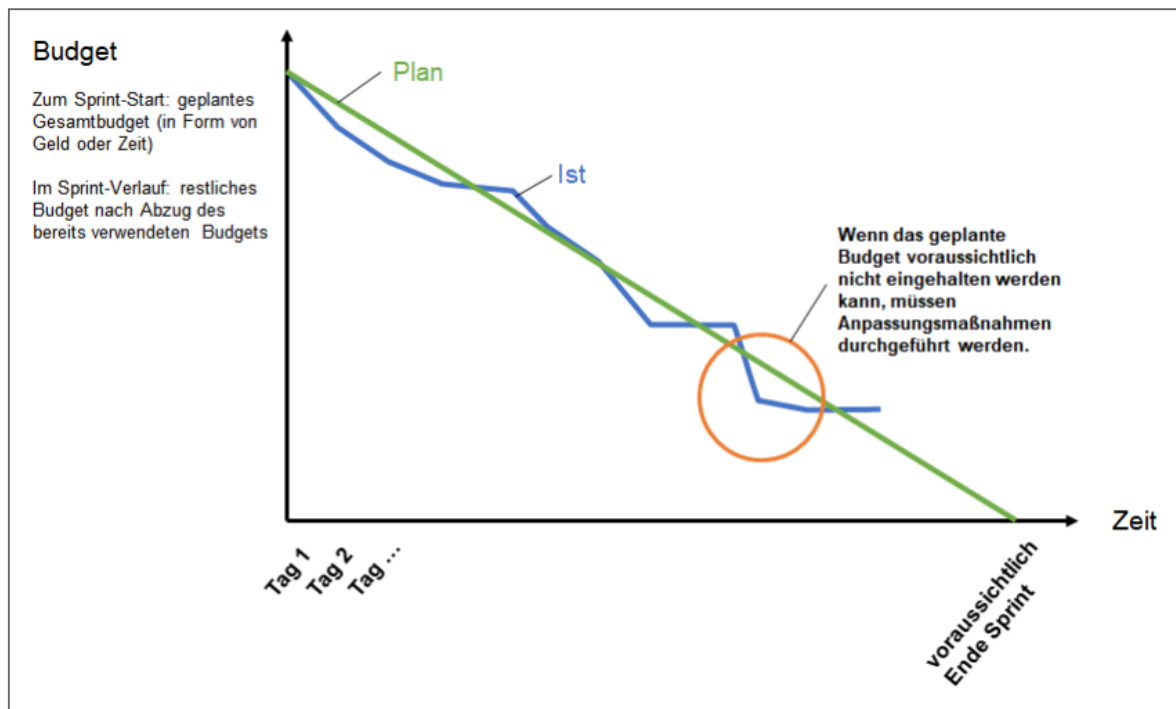


Abbildung 3: Burn-Down-Chart für das Controlling im Scrum-Sprint

12. Lösungen im Sprint-Review-Meeting als fertig abnehmen - nächste Schritte

Jeder Sprint wird durch ein Sprint-Review-Meeting abgeschlossen. Das Team stellt die Ergebnisse, das sogenannte Product Increment, dem Product Owner oder den Kunden und Anwendern vor. Sie müssen die **Lösungen akzeptieren und abnehmen**.

Dazu wird geprüft, ob das Teilergebnis aus diesem Sprint (Product Increment) der mit dem Product Backlog festgelegten Definition of Done entspricht. Falls der Product Owner nicht zufrieden ist, muss das Produkt verbessert werden. Es wird festgehalten, was nachbearbeitet werden muss.

Ist das Teilprodukt, das in diesem Sprint erstellt wurde, in Ordnung und vom Product Owner und den Anwendern oder Kunden abgenommen, werden die nächsten Aufgaben aus dem Product Backlog bearbeitet. Dabei wird wieder die grobe Beschreibung im Product Backlog geprüft und detailliert und für den nächsten Sprint eingeplant. Im Rahmen dieses **Product-Backlog-Refinements** kann sich auch herausstellen, dass einzelne Backlog-Items nicht mehr oder erst später gebraucht werden.

13. Scrum Velocity ermitteln

Mit jedem Sprint wird klar, wie schnell das Team im Projekt insgesamt vorankommt. Um diese Geschwindigkeit zu ermitteln, wird ein sogenanntes **Velocity-Chart** erstellt und nach jedem Sprint aktualisiert. Das hat zwei Zwecke:

- Das Team lernt, ob seine Einschätzungen zum Aufwand und zum Zeitablauf realistisch waren und ob Änderungen im Arbeitsablauf geholfen haben; damit wird das Sprint Planning (Schritt 7) genauer.
- Der Product Owner kann aus dem Velocity-Chart auch erkennen, bis wann er mit dem Projektabschluss und dem fertigen Produkt rechnen kann.

Im Velocity-Chart werden die Aufgaben aus dem Product Backlog, die komplett fertiggestellt sind, dem Sprint zugeordnet, in dem sie erledigt wurden. Die dazugehörigen Story Points werden addiert und kumuliert. So ergibt sich ein ähnliches Bild wie beim Burn-Down-Chart. Abbildung 4 zeigt ein Beispiel für ein Velocity-Chart.

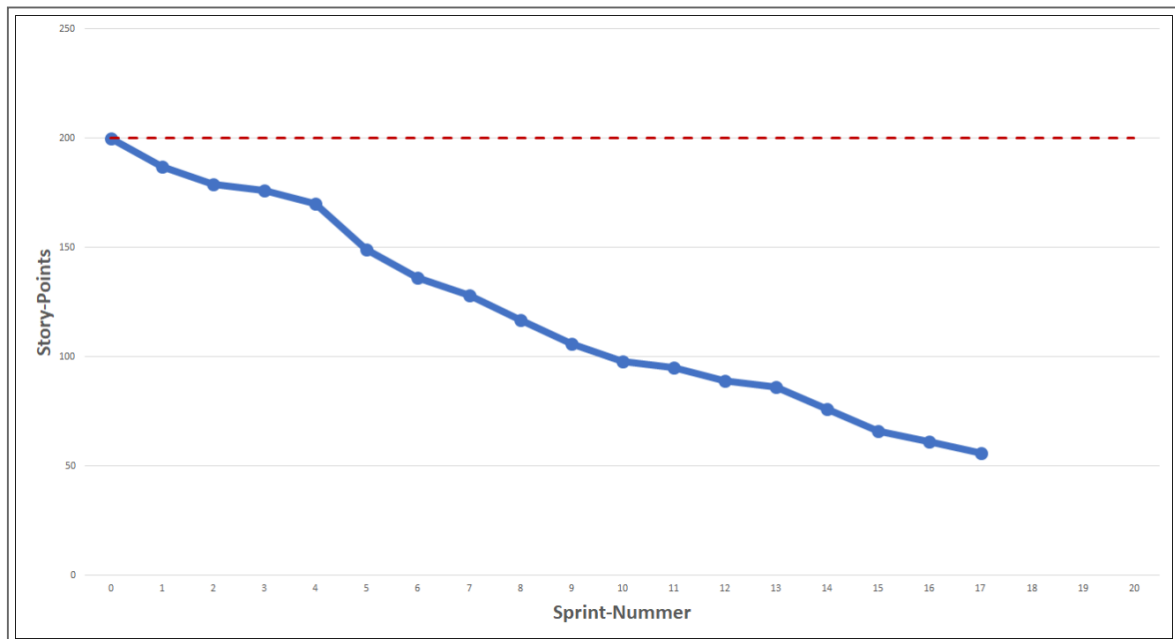


Abbildung 4: Velocity-Chart als Darstellung der Sprint-Geschwindigkeit und des Arbeitsfortschritts bei Scrum

14. Sprint-Retrospective durchführen

In gesonderten Treffen, der Sprint-Retrospective, können die Projektmitarbeiter ihre Zusammenarbeit im Team überprüfen. Sie besprechen und festhalten, was intern verbessert werden sollte, um die Projektziele gemäß Anforderung, Zeitplan und Kostenplan zu erreichen.

Anlässe für ein Sprint-Retrospective ergeben sich insbesondere dann, wenn Konflikte zwischen den Teammitgliedern oder auch zwischen Team und Product Owner entstanden sind oder wenn das Projekt nicht vorankommt. Im Rahmen dieser Meetings wird dann nach einer gemeinsamen und tragfähigen Lösung gesucht, sodass das Team weiterarbeiten kann und die Projektziele erreicht werden.

15. Projekt abschließen und Project-Review durchführen

Sind alle Sprints durchlaufen, wird das fertige Produkt an den Product Owner geliefert. Oder allgemein: Das Projekt kann abgeschlossen werden.

Der **Product Owner prüft** das Produkt und das Projektergebnis insgesamt vor dem Hintergrund der Anforderungen der Nutzer und der Kunden, wie sie mit den Story Cards formuliert wurden. Dabei werden Änderungen beachtet, die sich während der Projektlaufzeit ergeben haben. Außerdem wird berücksichtigt, welche Prioritäten zu Beginn des Projekts vereinbart und vergeben wurden.

Mit der Prüfung wird geklärt, welche Anforderungen, Merkmale und Funktionen des Produkts in diesem Projekt (doch nicht) bearbeitet wurden und welche Gründe das hatte. Sie werden gegebenenfalls erst in einem anschließenden Projekt bearbeitet.

Mit dem Projektabschluss kann ein **Project-Review** erfolgen. Hier werden alle Aspekte angesprochen, die die Zusammenarbeit erschwert oder den Prozess behindert haben. Daraus ergeben sich meistens Möglichkeiten, das nächste Scrum-Projekt besser zu machen.

Praxis

Anforderungen ermitteln, Story Cards schreiben

Stellen Sie die Anforderungen des Produkteigners und der Anwender und Kunden zusammen.

- Führen Sie Gespräche mit den Kunden und Anwendern, in denen diese ihre Anforderungen an das Produkt, das von Ihnen entwickelt werden soll, oder an das Projektergebnis, das Sie herbeiführen sollen, benennen und erläutern.
- Lassen Sie die Anwender oder Kunden für Ihr Produkt oder Projekt Story Cards schreiben. Das sind kurze Beschreibungen dessen, was das Produkt oder Projekt – aus Sicht der Anwender – leisten und können soll.
- Diskutieren und reflektieren Sie die Anforderungen auf den Story Cards. Fragen Sie nach.

Die Story Cards sind immer in den Worten der Anwender und in ihren Bildern formuliert. Nutzen Sie das folgende Formular, um Ihre Anwender-Anforderungen zu ermitteln und in der Story Card zu beschreiben.



Story Card für Anwenderanforderung im Scrum-Projekt

Art.-Nr. 99.143.18

Datei business-wissen_9914318.docx

Die Datei wurde erstellt für Word 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab Word 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

Product Backlog mit Backlog-Items erstellen

Leiten Sie aus den User Storys auf den Story Cards und den einzelnen Anforderungen die Aufgaben für das Projekt oder die Funktionen und Merkmale des Produkts und der Lösung ab, die Sie entwickeln sollen.

- Beschreiben Sie kurz die Backlog-Items; zum Beispiel auf der Grundlage von Use Cases.
- Stellen Sie den Zusammenhang dar zwischen dem Backlog-Item (Aufgabe, Funktion, Merkmal) und der Anforderung der Kunden oder Anwender (User Story).
- Legen Sie fest, woran Sie die Definition of Done festmachen.
- Welche Merkmale, Indikatoren oder Funktionen drücken aus, dass dieses Backlog-Item „fertig“ ist und die Anforderungen erfüllt werden?
- Was muss dazu getestet werden?

Wie Sie aus einer User Story einen Use Case oder ein Backlog-Item erstellen, zeigt das [Beispiel im Beitrag zu User Story und Use Case](#).

Komplexität und Aufwand schätzen und Story Points vergeben

Vergeben Sie für jedes Backlog-Item Story Points. Schätzen Sie dazu grob den Aufwand, der mit der Realisierung einer Funktion oder eines Merkmals verbunden ist.

Prioritäten klären

Klären Sie, welche Priorität eine Anforderung der Anwender oder Kunden sowie die damit verknüpften Aufgaben, Funktionen und Merkmale (Backlog-Items) haben soll. Diese Prioritäten haben Sie gegebenenfalls schon bei der Ermittlung der Anforderungen mit den Anwendern festgelegt und in der Story Card dokumentiert.

Sie können diese Einschätzung nun im Projektteam überprüfen. Fassen Sie diese Informationen in der folgenden Vorlage zusammen.

**Product Backlog mit Backlog-Items und User Storys**

Art.-Nr. 99.143.03

Datei business-wissen_9914303.xlsx

Microsoft Excel 2016

Things-That-Matter-Matrix für die Aufwandsschätzung und Prioritätensetzung

Eine Things-That-Matter-Matrix kann bei der Einschätzung des Aufwands und der Prioritäten hilfreich sein. Hier zeigen Sie den Zusammenhang zwischen Aufgabe, Funktion, Merkmal (Backlog-Item) einerseits und Anforderung der Kunden oder Anwender andererseits.

Dabei wird deutlich, welche Anforderung besonders aufwendig wird und mit welchem Backlog-Item Sie unterschiedliche Anforderungen gleichzeitig erfüllen können (Mehrfachverwendung).

**Things-That-Matter-Matrix für Projektaufgaben**

Art.-Nr. 99.143.04

Datei business-wissen_9914304.xlsx

Microsoft Excel 2016

In der folgenden Matrix können Sie zusätzlich den groben Aufwand zur Realisierung der Funktion angeben. Zur Schätzung des Aufwands wird einfach nach Größenklassen unterschieden – wie bei Kleidung: S, M, L für einen geringen, mittleren oder hohen Aufwand. So ergeben sich die Story Points und Prioritäten.

**Things-That-Matter-Matrix mit Aufwandsschätzung**

Art.-Nr. 99.143.05

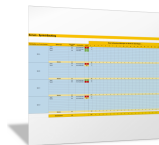
Datei business-wissen_9914305.xlsx

Microsoft Excel 2016

Backlog-Items in Teilaufgaben, Sprint-Tasks zerlegen und Maßnahmenplan erarbeiten

Zerlegen Sie die Backlog-Items in Teilaufgaben. Gehen Sie dabei nach Prioritäten vor; die wichtigsten zuerst. Die Teilaufgaben sind die Sprint-Tasks. Stellen Sie alle Sprint-Tasks im Sprint-Backlog zusammen. Das ist der Maßnahmenplan für Ihr Projektteam.

Halten Sie diesen Maßnahmenplan immer auf dem Laufenden. Planen Sie die gerade anstehenden Aufgaben und überprüfen Sie täglich, welche Aufgaben noch anstehen, welche gerade bearbeitet werden und welche abgeschlossen sind. Überprüfen Sie den Aufwand, der noch notwendig ist und vergleichen Sie ihn mit dem ursprünglich geplanten Aufwand.



Sprint Backlog mit Statusüberwachung und Aufwandsschätzung

Art.-Nr. 99.143.06

Datei business-wissen_9914306.xlsx

Microsoft Excel 2016

Projekt über tägliche Besprechungen – Daily Scrum – steuern

Visualisieren Sie den Arbeitsablauf und den Fortschritt im Projekt mithilfe eines Sprint-Taskboards. Nutzen Sie dazu die folgende Vorlage. Sie können die Projektphasen auch feiner gliedern und unterscheiden. Dies ist dann ein Kanban-Board für Ihr Scrum-Projekt.



Sprint-Taskboard mit Statusüberwachung für Maßnahmen und Teilaufgaben

Art.-Nr. 99.143.07

Datei business-wissen_9914307.xlsx

Microsoft Excel 2016



Kanban-Board für Tickets im Sprint Backlog

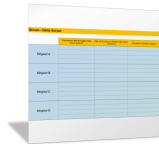
Art.-Nr. 99.143.16

Datei business-wissen_9914316.pptx

Die Datei wurde erstellt für PowerPoint 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab PowerPoint 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

Klären Sie im täglichen Scrum den Stand des Projekts. Alle Teammitglieder berichten:

- was sie **erreicht** haben,
- was nun **ansteht** und
- was **Probleme** bereitet hat.

**Persönlicher Statusbericht im täglichen Scrum**

Art.-Nr. 99.143.08

Datei business-wissen_9914308.xlsx

Microsoft Excel 2016

Burn-Down-Chart pflegen

- Halten Sie bei den täglichen Sprint-Meetings (Scrum) den Projektaufwand im Burn-Down-Chart fest.
- Identifizieren Sie Abweichungen.
- Besprechen Sie, welche Gegenmaßnahmen eingeleitet werden können.

Die folgenden einfachen Diagramme machen für alle sichtbar, ob das Projekt im Zeit- und Kostenplan liegt. Im Diagramm wird „rückwärts gezählt“; nicht das verbrauchte Budget steht im Vordergrund, sondern das noch verfügbare. Das fördert die Motivation.

**Burn-Down-Chart (Liniendiagramm)**

Art.-Nr. 99.143.10

Datei business-wissen_9914310.xlsx

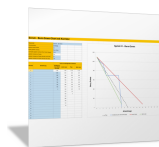
Microsoft Excel 2016

**Burn-Down-Chart (Säulendiagramm)**

Art.-Nr. 99.143.11

Datei business-wissen_9914311.xlsx

Microsoft Excel 2016

**Burn-Down-Chart mit variablem Zielkorridor**

Art.-Nr. 99.143.12

Datei business-wissen_9914312.xlsx

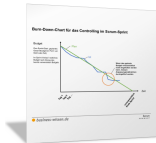
Microsoft Excel 2016

**Burn-Down-Chart mit Planung von Zusatzaufwand**

Art.-Nr. 99.143.13

Datei business-wissen_9914313.xlsx

Microsoft Excel 2016

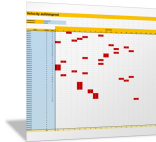
**Burn-Down-Chart für das Controlling im Scrum-Sprint**

Art.-Nr. 99.143.17

Datei business-wissen_9914317.pptx

Die Datei wurde erstellt für PowerPoint 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab PowerPoint 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

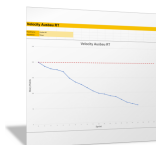
Wenn ein Sprint abgeschlossen und das Ergebnis vom Produkteigner (Product Owner) abgenommen ist, dann führen Sie Ihr Velocity-Chart fort. Sie dokumentieren den Arbeitsfortschritt und die Projektgeschwindigkeit für die Teammitglieder und den Produkteigner. Nutzen Sie dazu eine der beiden Vorlagen.

**Velocity-Chart: Arbeitsfortschritt im Scrum-Projekt aufsteigend**

Art.-Nr. 99.143.20

Datei business-wissen_9914320.xlsx

Microsoft Excel 2016

**Velocity-Chart: Arbeitsfortschritt im Scrum-Projekt absteigend**

Art.-Nr. 99.143.21

Datei business-wissen_9914321.xlsx

Microsoft Excel 2016

Sprint-Retrospective durchführen

- Welche Hindernisse sind im Arbeitsablauf aufgetaucht?
- Wie können diese beseitigt werden?

Halten Sie das Problem fest und sorgen Sie mit den Teammitgliedern dafür, dass es beseitigt wird. Wie kann das geschehen? Halten Sie Ihre Verbesserungsmaßnahmen in der folgenden Vorlage fest.

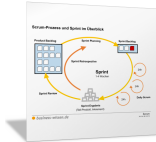
**Impediment Backlog mit Hindernissen und Problemen**

Art.-Nr. 99.143.09

Datei business-wissen_9914309.xlsx

Die Datei wurde erstellt für Excel 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab Excel 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

In der folgenden Vorlage ist der Ablauf eines Sprints mit den einzelnen Schritten und Elementen noch einmal in der Übersicht dargestellt.



Scrum-Prozess und Sprint im Überblick

Art.-Nr. 99.143.19

Datei business-wissen_9914319.pptx

Die Datei wurde erstellt für PowerPoint 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab PowerPoint 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

Im folgenden Abschnitt werden spezielle Probleme und Hindernisse erläutert, die bei Scrum-Projekten auftauchen oder die sich im Vorfeld ergeben können. Denn: die besonderen Regeln und Prinzipien von Scrum und agilem Projektmanagement setzen viel Transparenz, Verantwortungsbewusstsein und Vertrauen voraus, die nicht immer in der notwendigen Form gegeben sind. Außerdem muss die Projektaufgabe passen, damit Scrum funktioniert.

Product Backlog erstellen & planen | mit Vorlage

Dieser Beitrag richtet sich an neue Product Owner, die erstmals ein Product Backlog erstellen, sowie an Personen, die ihre Arbeit mit Product Backlogs optimieren möchten. Er bietet eine Schritt-für-Schritt-Anleitung zur Planung und Erstellung, mit Beispieldaten zur Orientierung. Außerdem: Tipps zur Priorisierung, Aktualisierung und Bereinigung des Backlogs.

Zuletzt geändert am 22.07.2024

Von Redaktion business-wissen.de, b-wise GmbH



Was ist ein Product Backlog?

Ein Product Backlog ist eine priorisierte Liste von Aufgaben und Anforderungen, die in einem Projekt umgesetzt werden sollen. Es ist ein zentrales Element des agilen Projektmanagements, insbesondere in Scrum. Damit können Projektteams ihre Arbeit organisieren und priorisieren.

Der **Product Owner** ist in der Regel verantwortlich für die Pflege und Priorisierung des Product Backlogs. Er muss:

- die Anforderungen und Wünsche der Stakeholder verstehen,
- das Backlog regelmäßig pflegen sowie priorisieren und
- sicherstellen, dass das Backlog immer aktuell und vollständig ist.

Was wird mit dem Product Backlog bezweckt?

Ein Product Backlog stellt sicher, dass das Team über eine priorisierte Liste von Aufgaben verfügt, die abgearbeitet werden muss. Dies ermöglicht:

- **Fokussierung auf wichtige Aufgaben**

Indem die wichtigsten Aufgaben zuerst bearbeitet werden, steigt der Wert des Produkts, da wesentliche Features und Verbesserungen frühzeitig umgesetzt und von Kunden genutzt werden können, besonders im Software-Bereich.

- **Transparenz**

Alle Teammitglieder und Stakeholder können den Fortschritt und die Prioritäten des Projekts nachvollziehen.

- **Flexibilität**

Neue Anforderungen können schnell hinzugefügt und priorisiert werden.

Beispiel für ein Product Backlog

So könnte das (fiktive) Product Backlog für ein Projekt zur Entwicklung einer mobilen App zur Essenslieferung aussehen:

Benutzerregistrierung und -anmeldung

Ziel: Benutzer sollen sich registrieren und anmelden können.

- Benutzer kann sich registrieren (E-Mail, notwendige Kontaktdaten)
- Benutzer kann sich anmelden (E-Mail, Passwort)
- Benutzer kann Passwort zurücksetzen

Benutzerprofil

Ziel: Benutzer sollen ihr Profil selbst verwalten können.

- Benutzer kann Profilinformationen und Kontaktdaten bearbeiten (Name, Adresse, Telefonnummer)
- Benutzer kann ein Profilbild hochladen
- Benutzer kann bevorzugte Zahlungsmethoden speichern

Restaurantsuche und -anzeige

Ziel: Benutzer sollen nach Restaurants suchen und Informationen dazu anzeigen können.

- Benutzer kann nach Restaurants suchen (Name, Kategorie, Standort)
- Benutzer sieht detaillierte Restaurantseite (Beschreibung, Öffnungszeiten, Bewertungen)

Menü und Bestellung

Ziel: Benutzer sollen die Speisekarte eines Restaurants anzeigen und Bestellungen aufgeben können.

- Benutzer kann die Speise- und Getränkekarte eines Restaurants anzeigen
- Benutzer kann angebotene Speisen und Getränke zum Warenkorb hinzufügen
- Benutzer kann Bestellungen aufgeben
- Benutzer kann Sonderwünsche angeben (zum Beispiel: keine Zwiebeln, glutenfrei)

Bezahlung

Ziel: Benutzer sollen Bestellungen sicher und flexibel bezahlen können.

- verschiedene Zahlungsmethoden (Kreditkarte, PayPal, Apple Pay) werden unterstützt
- Zahlungsabwicklung ist sicher
- Bestellbestätigung wird angezeigt

Bestellverfolgung

Ziel: Benutzer sollen den Status ihrer Bestellung in Echtzeit verfolgen können.

- Benutzer kann den Status seiner Bestellung verfolgen (zum Beispiel in Vorbereitung, unterwegs, geliefert)
- Echtzeit-Tracking des Lieferanten ist möglich

Benachrichtigungen

Ziel: Benutzer sollen über wichtige Ereignisse und Angebote informiert werden.

- Push-Benachrichtigungen für wichtige Ereignisse (Bestellbestätigung, Statusänderungen, Sonderangebote) werden empfangen
- E-Mail-Benachrichtigungen für Bestellungen und Werbeaktionen werden empfangen

Sicherheit und Datenschutz

Ziel: Die App soll sicher und konform mit Datenschutzbestimmungen sein.

- Implementierung von SSL-Verschlüsselung
- Einhaltung der Datenschutzbestimmungen (DSGVO)
- regelmäßige Sicherheitsüberprüfungen

Beispiel einer User Story

Grundlage für das Product Backlog sind die sogenannten User Storys. Damit sind kurze, einfache Beschreibungen der Funktionalitäten oder Anforderungen aus der Perspektive des Anwenders gemeint. Eine User Story kann zum Beispiel so aussehen:

Titel:

Benutzerregistrierung

Beschreibung:

Als neuer Benutzer möchte ich mich registrieren können, damit ich auf die personalisierten Inhalte zugreifen kann.

Daraus leiten sich die Anforderungen für das Product Backlog ab:

1. Ein Registrierungsformular mit Feldern für Name, Rechnungsadresse, Lieferadresse, E-Mail und Passwort ist vorhanden.
2. Eine Bestätigungs-E-Mail wird nach erfolgreicher Registrierung versendet.
3. Fehlerhafte Eingaben werden dem Benutzer angezeigt.

Für die Realisierung dieser Anforderungen wird der Aufwand abgeschätzt:

5 Story Points



User Story schreiben

Im Beitrag [So erstellen Sie gute User Storys](#) erläutert der Experte, wie Sie eine User Story aufbauen und formulieren.

Wie erstellt man ein Product Backlog?

So gehen Sie bei der Erstellung des Product Backlogs vor:

Schritt 1: Vision und Ziele verstehen

Erstellen Sie eine klare und inspirierende Produktvision, die das langfristige Ziel des Produkts beschreibt.

Definieren Sie danach konkrete, messbare Ziele, die erreicht werden sollen, um die Vision zu verwirklichen. Halten Sie sich bei der Formulierung an die SMART-Regeln.



SMARTe Ziele definieren

Im Beitrag [SMARTe Ziele und wie sie formuliert werden](#) erfahren Sie, wie man Ziele spezifisch, messbar, erreichbar, relevant und terminiert definiert.

Schritt 2: Stakeholder identifizieren und konsultieren

Bestimmen Sie alle relevanten Stakeholder, einschließlich Kunden, Benutzer, Geschäftsführung und Entwicklungsteam.

Führen Sie Interviews, Umfragen und Workshops durch, um Anforderungen und Wünsche der Stakeholder zu sammeln.

Schritt 3: Erste Backlog-Items erstellen

Beginnen Sie mit großen Anforderungen, sogenannten Epics, und unterteilen Sie diese in Features.

Erstellen Sie aus den Features detaillierte User Stories. Jede User Story sollte folgendes Format haben: „Als [Benutzertyp] möchte ich [Ziel/Wunsch], damit [Nutzen].“

Leiten Sie aus den User Storys die Anforderungen (Features, Funktionen, Produktmerkmale) ab, die ins Product Backlog kommen. Das sind die einzelnen Backlog-Items.

Schritt 4: Backlog-Items priorisieren

Bewerten Sie jedes Backlog-Item basierend auf dem geschäftlichen Wert und dem geschätzten Aufwand.

Verwenden Sie Techniken wie MoSCoW (Must, Should, Could, Won't), die Eisenhower-Matrix oder das Kano-Modell zur Priorisierung.

Ordnen Sie die Backlog-Items nach Priorität. Die wichtigsten und wertvollsten Items kommen ganz nach oben auf Ihrem Product Backlog.



Wie das Kano-Modell funktioniert, erläutert der Beitrag [Mit dem Kano-Modell die Kundenzufriedenheit messen](#).

Schritt 5: Detaillierung und Aufwandsschätzung

Definieren Sie klare Akzeptanzkriterien für jedes Backlog-Item, die beschreiben, wann das Item als fertig gilt; die Definition of Done.

Lassen Sie das Entwicklungsteam jeden Eintrag auf dem Product Backlog schätzen, zum Beispiel mit Story Points oder notwendigen Arbeitsstunden.

Regelmäßige Pflege und Aktualisierung

Achten Sie darauf, dass das Product Backlog immer auf dem allerneuesten Stand ist, indem Sie:

- das Product Backlog regelmäßig überprüfen, mindestens vor jedem Sprint Planning Meeting;
- Feedback von Stakeholdern und dem Entwicklungsteam kontinuierlich in das Backlog integrieren und
- regelmäßige Backlog Refinement Meetings (auch: Grooming) abhalten, um die Backlog-Items weiter zu verfeinern und neu zu priorisieren.

Welche Tools eignen sich für ein Product Backlog?

In der Regel benötigen Sie Tools, um das Product Backlog zugänglich und nutzbar zu machen und Software, mit der Sie den Status visualisieren.

Um das Product Backlog zu pflegen und für das Team zugänglich zu machen, nutzen Sie Backlog-Management-Tools wie:

- Jira
- Trello
- Asana

Zur Visualisierung nutzen Sie Kanban-Boards oder andere Visualisierungstechniken, um den Status der Backlog-Items und deren Fortschritt sichtbar zu machen.



Der Beitrag [Mit dem Kanban-Board den Workflow verbessern](#) erläutert, wie das Kanban-Board funktioniert.

Wie werden Einträge im Product Backlog priorisiert?

Die Priorisierung der Einträge im Product Backlog erfolgt anhand mehrerer Kriterien:

- Geschäftswert: Wie viel Wert bringt die Umsetzung der Aufgabe für das Unternehmen?
- Dringlichkeit: Gibt es Fristen oder externe Abhängigkeiten, die die Dringlichkeit erhöhen?
- Risiko: Wie risikobehaftet ist die Aufgabe und kann sie andere Aufgaben blockieren?
- Komplexität: Wie komplex ist die Umsetzung und wie hoch ist der geschätzte Aufwand?

Wie oft sollte das Product Backlog aktualisiert werden?

Das Product Backlog wird regelmäßig aktualisiert, um sicherzustellen, dass es immer den aktuellen Prioritäten und Anforderungen entspricht.

Überprüfen Sie es mindestens einmal pro Sprint, zum Beispiel im Zuge der Sprint-Planung oder im Backlog Refinement Meeting. Das bedeutet, dass Sie etwa einmal pro Woche das Product Backlog prüfen und bei Bedarf aktualisieren. Letztlich hängt die Häufigkeit der Aktualisierung vom Umfang und von der Komplexität der jeweils bearbeiteten Aufgaben ab.

Als Product Owner sollten Sie außerdem kontinuierlich Feedback von Stakeholdern und dem Entwicklungsteam einholen und das Backlog entsprechend anpassen.

Wie bereinigen Sie ein überfrachtetes Product Backlog?

Ein überfrachtetes Product Backlog kann durch verschiedene Maßnahmen bereinigt werden. Zunächst kann die **Archivierung** älterer oder weniger wichtiger Einträge helfen, das Backlog zu entschlacken. Diese Einträge können entweder archiviert oder vollständig gelöscht werden, um Platz für relevantere Items zu schaffen.

Ein weiterer Schritt ist die **Kategorisierung**, bei der Einträge in thematische Kategorien oder größere Epics zusammengefasst werden. Dies erleichtert die Übersicht und priorisiert die Arbeit. (Epics sind große, übergeordnete User Stories im Product Backlog, die umfangreiche Funktionen oder Anforderungen beschreiben und in kleinere, detailliertere User Stories unterteilt werden können.)

Zudem sind **regelmäßige Reviews** durch Sie als Product Owner unerlässlich. Diese regelmäßigen Überprüfungen und Bereinigungen stellen sicher, dass das Backlog aktuell bleibt und nur die wichtigsten und relevanten Einträge enthält.

Wie misst man den Erfolg eines Product Backlogs?

Der Erfolg eines Product Backlogs wird gemessen anhand:

- der Erfüllung der Geschäftsziele,
- der Kundenzufriedenheit,
- der Produktivität des Teams und
- der Flexibilität.

Was sind die häufigsten Fehler bei der Verwaltung eines Product Backlogs?

Zu den häufigsten Problemen gehören:

- **Überfrachtung:** Ein zu großes Backlog kann unübersichtlich werden und die Priorisierung erschweren.
- **unklare Prioritäten:** Wenn die Prioritäten nicht klar sind, kann das Team nicht effizient arbeiten.
- **mangelnde Pflege:** Ein veraltetes Backlog spiegelt nicht die aktuellen Anforderungen und Prioritäten wider.
- **Kommunikationsprobleme:** Missverständnisse zwischen dem Product Owner, dem Entwicklungsteam und den Stakeholdern können zu Unstimmigkeiten führen.

Praxis

Product Backlog planen und aufbauen

Diese Vorlage begleitet Sie Schritt für Schritt bei der Erstellung eines Product Backlogs – von der Definition Ihrer Projektziele über das Sammeln und Priorisieren von User Stories bis zur Schätzung der Aufwände und regelmäßigen Pflege des Backlogs.



Product Backlog erstellen

Art.-Nr. 99.143.22

Datei business-wissen_9914322.docx

Microsoft Word 2016

Verwenden Sie den Leitfaden, indem Sie die enthaltenen Tabellen und Checklisten ausfüllen, erweitern und an Ihre eigenen Projektanforderungen anpassen.

Product Backlog erstellen

Mit dieser Vorlage erstellen Sie Ihr Product Backlog zügig mit Excel.



Product Backlog mit Backlog-Items und User Storys

Art.-Nr. 99.143.03
Datei business-wissen_9914303.xlsx
Microsoft Excel 2016

Sie tragen in die Excel-Vorlage ein:

- ID und Titel des Backlog-Items oder der Aufgabe
- zugehörige User Story (oder: User Card)
- In welchem Sprint wird umgesetzt?
- Priorität
- Story Points
- Status
- bei Bedarf: Notizen

Backlog-Items priorisieren

Mit der folgenden Vorlage werten Sie die Ergebnisse einer Kundenbefragung nach dem Kano-Modell aus und visualisieren diese in einem Diagramm. Auf Basis dessen priorisieren Sie Inhalte Ihres Product Backlogs, indem Sie Kundenanforderungen in Basis-, Leistungs- und Begeisterungsmerkmale einteilen.

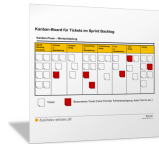


Kano-Diagramm

Art.-Nr. 99.129.35
Datei business-wissen_9912935.xlsx
Microsoft Excel 2016

Backlog nutzen und zugänglich machen

Mit diesem Kanban-Board für Scrum-Projekte behält Ihr Scrum-Team jederzeit den Überblick über alle Arbeiten im Sprint und über den Stand des Sprint Backlogs und der Tickets.



Kanban-Board für Tickets im Sprint Backlog

Art.-Nr. 99.143.16

Datei business-wissen_9914316.pptx

Die Datei wurde erstellt für PowerPoint 2016 und neuer. Die Datei kann auch mit älteren Versionen genutzt werden (ab PowerPoint 2010). Doch sind dabei geringfügige Darstellungsfehler nicht ausgeschlossen.

Das Kanban-Board hilft, Aufgaben aus dem Product Backlog durch den Workflow zu bewegen. Es visualisiert, wie diese Aufgaben bearbeitet werden, und stellt sicher, dass das Team effizient arbeitet und Engpässe im Prozess identifiziert und beseitigt werden.

Retrospektive: Leitfaden mit Beispielen, Methoden & Praxistipps

Welche Ziele verfolgt man mit einer Retrospektive? Und wie erreicht man diese Ziele? Sie erfahren, wie eine Retrospektive abläuft, der Moderator seiner Rolle gerecht wird und die Kommunikation während der Retrospektive verbessert werden kann. Außerdem werden verschiedene Methoden für Anfänger und Fortgeschrittene praxisnah erklärt.

Zuletzt geändert am 20.01.2023

Von Redaktion business-wissen.de, b-wise GmbH



Definition: Was versteht man unter einer Retrospektive?

Eine Retrospektive ist ein Team-Meeting, bei dem aus vergangenen Fehlern und Erfolgen gelernt werden soll. Das Team blickt gemeinsam zurück und analysiert die Zusammenarbeit: Was hat gut und was weniger gut funktioniert? Das Team passt sein Arbeitsverhalten sowie die Prozesse flexibel (Stichwort: agil), kurzfristig und gemeinsam an.

Während der Retrospektive werden Maßnahmen zur Verbesserung der Zusammenarbeit und der Prozesse in der Teamarbeit beschlossen und protokolliert. Bei der nächsten Retrospektive prüft das Team: Haben wir die Maßnahmen erfolgreich umgesetzt? Und: Was können wir noch verbessern?

Wie oft werden Retrospektiven durchgeführt?

Retrospektiven werden meist – aber nicht ausschließlich – im agilen Umfeld und im agilen Projektmanagement (Scrum) genutzt. Wie häufig und wann die Retrospektive stattfindet, hängt daher von der Dauer des Projekts oder der Länge eines Sprints im Scrum-Projekt ab.

Manche Teams führen eine Retrospektive am Ende eines jeden Sprints durch, andere am Ende eines Teilprojekts. Eine weitere Möglichkeit ist, Retrospektiven in einem festen Rhythmus durchzuführen; dazu werden fixe Termine vereinbart.

Die meisten Teams führen alle zwei bis vier Wochen eine Retrospektive durch, manche reflektieren sich im einwöchigen Turnus. Handelt es sich um ein großes Team, kann die Retrospektive auch seltener durchgeführt werden – alle drei Monate etwa.

Wie lange dauert eine Retrospektive?

Um den Regeln des Sprint-Plannings zu entsprechen, wird die Retrospektive innerhalb eines festen Zeitraums durchgeführt. Der eingeplante Zeitrahmen darf nicht überschritten werden. Wichtig hierbei:

- Jede und jeder kommt zu Wort.
- Damit das funktioniert, sollten sich die Teilnehmenden kurzfassen.
- Besteht Redebedarf über die Retrospektive hinaus, werden separate Termine vereinbart.
- An diesen nehmen nur die direkt betroffenen Personen teil, um das Zeitkontingent Nicht-Beteiligter zu schonen.
- Je kürzer die Zeitabstände zwischen zwei Retrospektiven und je kleiner das Team, desto kürzer dauert die Retrospektive.

Meistens dauert die Retrospektive zwischen einer und bis zu drei Stunden.

Warum führt man eine Retrospektive durch?

Die Retrospektive bietet einen Rahmen, damit sich das Team Zeit für die Reflexion zur Zusammenarbeit nehmen kann. Erhalten Mitarbeitende keine Möglichkeit zum Zurückblicken auf vergangene Sprints, ihre Arbeit und die Zusammenarbeit mit anderen, können sie nicht daraus lernen. Fehler werden im schlechtesten Fall wiederholt; Verbesserungen bleiben aus.

Ziele der Retrospektive

- Der Zusammenhalt im Team und das Wir-Gefühl werden gestärkt.
- Alle reflektieren gemeinsam, Wissen wird ausgetauscht und eine gemeinsame Erfahrungsbasis geschaffen.
- Teammitglieder lernen sich besser kennen, indem die individuelle Sicht auf die gemeinsame Arbeit kommuniziert wird.

Wer nimmt an einer Retrospektive teil?

Wenn es sich um eine Sprint-Retrospektive handelt, nehmen mindestens alle Mitglieder des Scrum-Teams, der Scrum-Master und der Product-Owner teil.

Wird die Retrospektive in großen zeitlichen Abständen durchgeführt, lädt man alle Mitarbeitenden ein, die laufend oder sporadisch im Projekt und an den Sprints beteiligt waren. Dritte (freie Mitarbeiter, Personen aus anderen Teams ...) können zum Erkenntnisgewinn beitragen – manchmal sogar besser als feste Teammitglieder, weil Sie „den Blick von außen“ haben.

Was unterscheidet Retrospektive und Review?

Bei einer **Review** geht es darum, die Wünsche, Vorstellungen und Ansprüche der Kunden zu verstehen. Teams besprechen im Zuge der Review die Ergebnisse und stellen sie einander vor. Neue Ideen oder Lösungsansätze werden während der Review besprochen.

Bei der **Retrospektive** soll hingegen die Arbeitsweise und die Zusammenarbeit verbessert werden. Es geht um das Vorgehen als solches, weniger um das Endergebnis oder das fertige Produkt.

Generell sollten Review und Retrospektive klar voneinander getrennt werden. Die Termine werden nicht zusammengelegt, etwa um Zeit zu sparen.

Wie läuft eine Retrospektive ab?

Die Retrospektive wird in fünf Phasen unterteilt:

1. Rahmenbedingungen schaffen

Der Moderator oder der Scrum-Master (meist gehören beide Rollen zur selben Person) sorgen für eine lockere, freundliche, sichere Atmosphäre. Sie sind auch dafür verantwortlich, dass alle relevanten Personen von Anfang bis Ende teilnehmen.

Es wird ein gemeinsames Bewusstsein geschaffen im Sinne von: Jede und jeder hat das Beste gegeben, unabhängig davon, ob heute Schwächen oder Fehler identifiziert werden.

2. Informationen zusammentragen

Das Team blickt gemeinsam zurück. Was ist gut gelaufen? Was ist schlecht gelaufen? Diese Punkte werden gesammelt und (schriftlich) festgehalten.

3. Neue Erkenntnisse gewinnen

Die Ursachen für positive und negative Dinge werden identifiziert. Das geschieht nicht, indem Schuldige gesucht werden. Stattdessen bleiben alle sachlich und nennen Fehler bei der Zusammenarbeit, der Organisation oder fachliche Fehler ohne persönlichen Bezug.

In diesem Schritt entsteht die Basis, um Maßnahmen für Verbesserungen zu entwickeln. Das Ziel: Nicht die Symptome werden analysiert, sondern die Ursachen gefunden.

4. Entscheidungen treffen

Maßnahmen werden festgelegt. Sie sind:

- realistisch und umsetzbar
- konkret und überprüfbar
- sinnvoll und (von allen) akzeptiert

5. Retrospektive abschließen

Alle Ergebnisse werden kurz dokumentiert, sodass jede und jeder sie nach der Retrospektive ansehen kann. Die nächsten Schritte werden gemeinsam geplant.

Was macht eine gute Retrospektive aus?

Das Ziel einer Retrospektive ist es, aus Vergangenem zu lernen. Werden Probleme und Erfolge gleichermaßen erkannt, im Team analysiert und Maßnahmen daraus abgeleitet, war die Retrospektive erfolgreich.

Ein weiterer wichtiger Faktor: die positive Grundstimmung im Team. Nur wenn sich alle Mitarbeitenden wohlfühlen und sicher fühlen, werden die Potenziale voll ausgeschöpft – denn so werden auch Aspekte betrachtet, die nur Einzelnen aufgefallen sind. Diese Aspekte können wichtig sein und sollten behandelt werden.

Ist die Stimmung in der Gruppe gelöst, dürfen auch unangenehme Themen angesprochen werden. Dazu gehören etwa eine intransparente Kommunikation während des Sprints oder fachliche Fehler.

Die Führungskraft vermittelt in diesem Kontext zentrale Grundsätze der Retrospektive:

- Was in der Retro besprochen wird, darf im Nachhinein niemandem zur Last gelegt werden.
- Niemand formuliert Kritik persönlich.
- Keiner reagiert nachtragend.
- Diese Grundsätze gelten für jede Retrospektive – ausnahmslos.

Tipps für Moderatoren der Retrospektive

Bereiten Sie die Teilnehmenden vor.

Ein kurzes Einzelgespräch mit allen oder zumindest einigen Teilnehmenden trägt zum Erfolg der Retrospektive bei. Introvertierte Menschen oder neue Teammitglieder ermuntern Sie dazu, ihre Punkte vorzutragen. Sie übernehmen die Rolle des Coaches.

Bei einem Einzelgespräch erfahren Sie außerdem, ob es auch persönliche Probleme gibt, die derjenige nicht in der Retrospektive ansprechen möchte.

Sorgen Sie für Effizienz.

Achten Sie darauf, dass während der Retrospektive wenige, realistische und nur Erfolg versprechende Maßnahmen ausgearbeitet werden. Wenn jede noch so abstruse Idee besprochen wird, dauert das zu lange und die Konzentration leidet.

Setzen Sie den Fokus auf das Wesentliche.

Wenn jemand ausschweifend berichtet oder vom Thema abweicht, lenken Sie ihn zum Kern der Retrospektive zurück. Führen Sie sich vor Augen: Das Ziel dieses Meetings ist die Entwicklung konkreter Maßnahmen, nicht das endlose Argumentieren oder Diskutieren.

Sorgen Sie für eine freundliche, offene Kommunikation.

Alle Teilnehmenden kommunizieren sachlich, respektvoll und gleichberechtigt. Wird diese Regel nicht eingehalten, greifen Sie sofort ein. Ihre Teammitglieder verlassen sich darauf, dass sie in einem sicheren, fairen Umfeld sprechen und sich auf geltende Regeln verlassen dürfen.

Regen Sie zur Diskussion an und lenken Sie diese.

Alle sollen sich an der Retrospektive beteiligen. Um dieses Ziel zu erreichen, verteilen Sie die Redezeit fair. Wenn jemand viel oder laut spricht, erhält er nicht automatisch mehr Redezeit. Sprechen Sie stattdessen ruhige Teilnehmende gezielt an. Fragen Sie nach deren Meinung, ohne sie dabei unter Druck zu setzen oder als „Schweiger“ bloßzustellen.

Beispiele: Maßnahmen aus der Retrospektive ableiten

Zu ergreifende Maßnahmen werden nach der SMART-Regel formuliert. Sie sind spezifisch, messbar, attraktiv, realistisch und terminiert.

Einige Beispiele, **wie Maßnahmen nicht formuliert werden** sollten:

1. Wir kommunizieren in E-Mails, Chats und persönlichen Gespräch höflich miteinander.
2. Wir wollen bei Besprechungen weniger Zeit verschwenden.
3. Aufgaben werden genauer formuliert.

Die obigen Beispiele entsprechen nicht den SMART-Regeln, denn sie sind zu allgemein formuliert.

Nun die **positiven Beispiele zur Formulierung von Maßnahmen**:

1. Nach unserem wöchentlichen Teamtreffen nehmen wir uns zehn Minuten Zeit, um die Kommunikationsweise zu reflektieren und gegebenenfalls die Kommunikationsregeln anzupassen oder zu besprechen.
2. Im Daily-Scrum-Meeting erhält ab sofort jede Teilnehmerin und jeder Teilnehmer exakt zwei Minuten Redezeit.
3. Wenn eine Aufgabe übergeben wird, halten wir schriftlich fest, was die Aufgabe genau umfasst und an wen man sich bei Fragen wenden kann.

Tipp

Formulieren Sie herausfordernde Maßnahmen

Probieren Sie eine vielversprechende Maßnahme auch dann aus, wenn einige Teilnehmende diese für zu herausfordernd halten. Sehen Sie die Maßnahme als ein Experiment, an welchem das Team wächst und lernt.

Solche Experimente sind erlaubt, wenn sie nur für einen begrenzten Zeitraum gelten – für eine Woche oder einen Monat etwa. Danach prüfen Sie gemeinsam: Hat die Maßnahme zum gewünschten Ergebnis geführt? Falls die Antwort „nein“ lautet: warum nicht?

Vier beliebte Methoden für die Retrospektive im Überblick

Start-Stop-Continue-Retrospektive

Sie besprechen mit Ihrem Team:

- Start Doing This: Diese Dinge werden zukünftig geändert oder diese neuen Vorgehensweisen eingeführt.
- Stop Doing This: Diese Dinge liefen nicht gut und werden daher ab sofort nicht mehr (auf diese Weise) getan.
- Continue: Diese Dinge behalten wir genauso bei, weil sie sich bewährt haben.

Die vereinfachte Form dieser Methode heißt „Start-Stop-Retrospektive“. Sie eignet sich für besonders Eilige oder wenn Sie die Retrospektive zur Einführung so simpel wie möglich gestalten möchten. Es gibt dann nur noch die Aspekte „Was lief gut?“ und „Was lief schlecht?“.

4L-Retrospektive

Läuft die Retrospektive nach dem 4-L-Modell ab, definiert Ihr Team vier verschiedene Bereiche in dieser Reihenfolge:

- Liked: Was lief gut?
- Learned: Was lernt das Team daraus?
- Lacked: Was lief schlecht, hat gefehlt oder war nicht ausreichend vorhanden (zum Beispiel Zeit, Motivation, Austausch ...)?
- Longed for: Was hätte das Team sich gewünscht?

Optional fügen Sie – dauerhaft oder zeitweise – einen fünften Punkt hinzu. Unter dem Aspekt „Loathed“ werden Dinge gesammelt, die Ihr Team genervt haben, weil sie beispielsweise unnötig viel Zeit gekostet oder Stress verursacht haben.

FLAP-Retrospektive

Der Wortbestandteil „FLAP“ setzt sich zusammen aus:

- Future Considerations: Worauf muss das Team in der Zukunft achten?
- Lessons Learned: Was haben wir durch den Rückblick gelernt?
- Accomplishments: Welche Ziele haben wir erreicht? Und: Was haben wir verbessert?
- Problem Areas: Welche Probleme gab es?

Die Erfolge des Teams stehen im Vordergrund. Wenn Sie Ihr Team motivieren und den Zusammenhalt stärken möchten, wenden Sie diese Methode zur Retrospektive an. Natürlich kann sie auch im Wechsel mit einer anderen der vier Methoden genutzt werden.

Timeline-Retrospektive

Bei dieser Methode steht der zeitliche Ablauf im Vordergrund: Wann sind Probleme aufgetreten oder Erfolge eingetreten? Gibt es einen zeitlichen Zusammenhang zwischen dem Auftreten des einen Problems und einem darauffolgenden anderen Problem?

Taiichi Ohnos „5 Whys“ für die Retrospektive nutzen

Von Taiichi Ohno, der Erfinder des Toyota-Produktionssystems, stammt die Aussage: „Stelle fünf Warum-Fragen über ein Thema.“ Er geht davon aus, dass das mehrfache Warum-Fragen und die Beschäftigung mit den Antworten dabei hilft, die Ursache für ein Problem zu identifizieren.

Diese Regel lässt sich auf die Retrospektive übertragen:

Stellen Sie sich vor, Ihr Team hat nicht das Ziel erreicht, seltener, aber klarer und damit effizienter miteinander zu kommunizieren. Sie hatten eigentlich vereinbart, dass lange E-Mails mit komplizierten Inhalten nicht mehr untereinander verschickt werden, sondern der Video-Call das bevorzugte Kommunikationsmittel ist. (Es handelt sich um ein hybrides Team; einige Mitarbeitende arbeiten dauerhaft im Homeoffice.)

Nun stellen Sie die folgenden Fragen:

- Warum hat die Kommunikation nicht reibungslos funktioniert?
Antwort: Es gab Missverständnisse.
- Warum gab es Missverständnisse?
Antwort: Komplizierte Sachverhalte wurden in langen E-Mails beschrieben.
- Warum haben wir komplizierte Sachverhalte als E-Mail formuliert?
Antwort: Wir konnten keine Videokonferenzen durchführen.

- Warum haben wir nicht die Videokonferenz genutzt?
Antwort: Einige Teammitglieder waren per Video-Call nicht erreichbar.
- Warum waren manche Personen nicht per Video-Call erreichbar?
Antwort: ...

Wenn Ihr Team die Antworten auf diese Fragen reflektiert und bespricht, werden entsprechend Maßnahmen abgeleitet.

Als **Ursache** für das Problem haben Sie etwa ermittelt, dass die Terminkalender der Mitarbeitenden oft so voll sind, dass sich spontan keine gemeinsamen freien Zeitfenster für spontane Videokonferenzen finden.

Eine mögliche **Maßnahme**: Sie planen für jeden Arbeitstag ein Zeitfenster ein, welches ausschließlich für die Kommunikation innerhalb des Teams gedacht ist. Dieser Termin ist fix und wird im Kalender geblockt. Damit lösen Sie das Problem der vielen Missverständnisse und der mangelhaften Kommunikation.

Wie Sie die Retrospektive abschließen

Dem Moderator, dem Scrum-Master oder der Führungskraft kommt die Aufgabe zu, einen runden Abschluss für die Retrospektive zu gestalten.

Erreicht werden soll:

- Die Mitarbeitenden verlassen das Meeting mit einem guten Gefühl,
- das Zusammengehörigkeitsgefühl wird gestärkt und
- die beschlossenen Maßnahmen sind akzeptiert.

Zudem sollte niemand nachtragend reagieren, falls er direkt von einem möglichen Kritikpunkt oder Problem betroffen war.

Retrospektive abschließen mit der „You and me“-Methode

Zuerst wird ein Dank an eine Kollegin oder einen Kollegen („you“) formuliert. Danach wird ein persönliches Vorhaben ausgesprochen („me“).

Beispiel:

Ich bedanke mich bei Anna, dass sie meine fachlichen Fragen immer zeitnah beantwortet.

Ich nehme mir vor, meine E-Mails an Kollegen noch präziser, verständlicher und kürzer zu formulieren.

Retrospektive abschließen mit der Appreciation Shower

Das Team wird in Dreiergruppen aufgeteilt. Zwei Personen sitzen sich gegenüber, der Dritte mit dem Rücken zu den beiden anderen. Die Partner haben zwei Minuten Zeit, sich über das dritte Teammitglied zu unterhalten. Dabei darf ausschließlich Positives gesagt werden.

Nach Ablauf der Zeit wechseln die Rollen, bis jeder die positiven Aussagen über die eigene Person einmal gehört hat.

Variante der Retrospektive für Fortgeschrittene: Futurespective

Sie ahnen es aufgrund der Bezeichnung: Bei der Futurespective geht es darum, in die Zukunft zu schauen. Die Durchführung bietet sich daher an, bevor ein neues Projekt beginnt. Ihr Team soll bereits im Voraus Ziele definieren und Risiken identifizieren.

Die folgenden beiden Methoden eignen sich, wenn Sie schon seit Längerem agil arbeiten und Retrospektiven durchführen.

Pre-Mortem-Methode

Hier handelt es sich um eine kreative Methode: Sie stellen sich vor, das Projekt sei bereits abgeschlossen. Es ist kolossal gescheitert. Jetzt stellen Sie sich zusammen mit Ihrem Team die Fragen:

- Warum ist das Projekt gescheitert?
- Hätte man das Scheitern verhindern können?
- Was hätte man anders machen müssen, um das Projekt erfolgreich abzuschließen?

„Remember The Future“-Methode

Hier gehen Sie genau gegenteilig an das Projekt heran. Sie nehmen an, dass das Projekt erfolgreich beendet wurde, und fragen:

- Warum wurde das Projekt erfolgreich beendet?
- Welche Erfolge haben wir erzielt?
- Schießt das Ergebnis eventuell sogar über die Ziele hinaus?
- Welche Fehler haben wir vermieden, damit das Projekt ein Erfolg wird?

Praxis

Planen Sie für Ihre Teamarbeit und im Projektverlauf regelmäßig Retrospektiven ein. Orientieren Sie sich bei der Vorbereitung der Retrospektive an dem oben beschriebenen Ablauf.

Nutzen Sie Methoden, um gemeinsam Ergebnisse zu erzielen und Maßnahmen zur Verbesserung der Zusammenarbeit umzusetzen.

Achten Sie darauf, dass die Grundsätze und Regeln der Retrospektiv von allen eingehalten werden.

Um Probleme und Ursachen der Zusammenarbeit mit der 5-Why-Methode zu besprechen, nutzen Sie die folgende Vorlage.



5-Why-Methode für das Qualitätsmanagement

Art.-Nr. 99.129.03

Datei business-wissen_9912903.xlsx

Microsoft Excel 2016

Um zu Beginn oder am Ende der Retrospektive die Stimmung im Team abzufragen, können Sie den Happiness-Index ermitteln. Nutzen Sie dazu die folgende Vorlage.

**Happiness-Index für die Stimmung im Projektteam**

Art.-Nr. 99.143.14

Datei business-wissen_9914314.xlsx

Microsoft Excel 2016

Probleme und Fehler bei Scrum, die Sie kennen sollten

Scrum will das Projektmanagement und den Projektablauf vereinfachen. Damit sollen bessere Projektergebnisse entstehen. Aber es kommt auch zu Konflikten. Nicht alle Projektmitarbeiter kommen mit der Transparenz und Offenheit zurecht, Projektleiter verlieren an Status und das Management muss dem Scrum-Team absolut vertrauen.

Zuletzt geändert am 03.02.2023

Von Dr. Jürgen Fleig, b-wise GmbH



Mögliche Konflikte im Scrum-Team

In einem Scrum-Projekt kann sich kein Projektmitarbeiter verstecken. Die Leistung jedes einzelnen Teammitglieds wird sehr schnell sichtbar und bewertbar. Denn jeder muss bei Daily Scrum erläutern, was er getan hat und wie er vorankommt.

Wenn es Verzögerungen gibt, weil ein Teammitglied seine Aufgaben nicht rechtzeitig erledigt, dann kommen auch die anderen nicht voran. Die vorgebrachten und die tatsächlichen Gründe für Verzögerungen im Arbeitsablauf werden in täglichen Besprechungen offensichtlich, weil sie offen und direkt besprochen werden.

Diese Transparenz und das direkte Feedback zu den eigenen Arbeitsergebnissen kann zu Problemen und Konflikten führen:

- Manche kommen mit der direkten Kritik nicht zurecht; sie fühlen sich angegriffen.
- Dadurch können latente oder offene Konflikte entstehen.
- Das erschwert die Kommunikation; die Mitglieder im Team sprechen nicht genug miteinander.
- Die Motivation sinkt; im Projekt geht es nicht mehr richtig voran.
- Teilaufgaben werden hin und her geschoben.

Die Folgen können sein, dass das Team nicht mehr gemeinsam an der Zielerreichung arbeitet oder dass einzelne Mitarbeiter das Team verlassen.

Andere Konflikt- und Reibungspotenziale ergeben sich dann, wenn einzelne Mitarbeiterinnen oder Mitarbeiter das Projektteam dominieren. Sie setzen ihre Interessen und Meinungen durch, ohne dass diese im Sinne des Projekterfolgs richtig sein müssen. Hier gibt es zu wenig Korrekturmöglichkeiten, da das Team ja selbstbestimmt arbeiten soll und auch der Scrum-Master nicht eingreifen darf.

Wie Sie ein Scrum-Team zusammenführen

Um die Stimmung im Team zu verbessern, ist es wichtig, dass die Teammitglieder möglichst rasch (wieder) zusammenfinden. Sie müssen eine gemeinsame Arbeitsweise entwickeln, die für alle Teammitglieder akzeptabel ist. Alle sollen dazu beitragen, dass Konflikte konstruktiv bearbeitet und gelöst und die Projektziele erreicht werden. Es muss Teamentwicklung geben.

Oft durchläuft ein Team, dessen Mitglieder sich im Rahmen eines Projekts zusammenfinden, mehrere Phasen, die sogenannten Teamentwicklungsphasen. Das sind:

1. **Forming:** Jeder muss seine Rolle finden. Die Teammitglieder wollen sich kennenlernen und halten sich noch zurück.
2. **Storming:** Es entstehen Konflikte und Machtkämpfe, die immer häufiger offen ausgetragen werden.
3. **Norming:** Es werden gemeinsam Normen und Regeln für die Zusammenarbeit im Projektteam festgelegt.
4. **Performing:** Das Team hat sich und seinen Rhythmus gefunden. Es wird jetzt produktiv und selbstbestimmt gearbeitet. Meinungsverschiedenheiten und Konflikte werden konstruktiv und sachlich gelöst.



Tipp: Wie Sie Teamentwicklung betreiben

Teamentwicklung ist ein dauerhafter Prozess, den Sie gezielt unterstützen können. Nutzen Sie dafür spezielle **Methoden und Maßnahmen zur Teamentwicklung**. Zum Thema Teamarbeit finden Sie außerdem Tipps, **wie Sie im Team besser miteinander umgehen**.

Projektleiter verlieren ihren Status

Eine Projektleitung im klassischen Sinn gibt es in einem Scrum-Projekt nicht. Stattdessen gibt es den sogenannten Scrum Master oder Project Master. Hat die Projektleitung beim klassischen Projektmanagement eine zentrale Rolle und Führungsfunktion und laufen bei ihr alle Fäden zusammen, so hat der Scrum Master beim agilen Projektmanagement nur die Rolle eines Trainers, Moderators oder Coachs.

Deshalb fühlen sich Projektleiterinnen und Projektleiter, die das klassische Projektmanagement kennen, in ihrer Bedeutung zurückgesetzt. Projektmanagement nach der „Methode Scrum“ ist kein Feld, auf dem man sich für den nächsten Karriereschritt empfehlen kann.

Wofür Scrum sich eignet – und wofür weniger

Scrum ist in Projekten entstanden, in denen es um die Entwicklung von Software ging. Dementsprechend sind agile Methoden des Projektmanagements auf solche Projekte ausgerichtet, in denen Produkte oder Dienstleistungen entwickelt werden oder in denen durch die IT-Abteilung im Unternehmen eine Technologie eingeführt wird.

Projekte zur Organisationsentwicklung, zum Change-Management oder für die Personalentwicklung haben oft kein genau beschreibbares Ergebnis, das ein Produkteigner abnehmen könnte. Hier lassen sich allenfalls Teilaufgaben mit den agilen Methoden bearbeiten.

Welche Rahmenbedingungen Scrum erschweren

Manchmal sind es die organisatorischen Rahmenbedingungen im Unternehmen, die agiles Projektmanagement erschweren. Das kann beispielsweise sein:

- Das Projektteam ist räumlich getrennt und kann sich nicht täglich treffen.
- Spezialisten stehen nur für eine begrenzte Zeit oder nur in eingeschränktem Umfang zur Verfügung.
- Die Kundenanforderungen bleiben unklar; der Produkteigner kann die Anforderungen nicht genau artikulieren, weil der direkte Kontakt zu Anwendern und Kunden nicht möglich ist.

Fehlendes Vertrauen durch Management und Geschäftsleitung

Besonders kritisch ist, wenn das Management im Unternehmen nicht hinter diesem radikalen Paradigmenwechsel des Projektmanagements steht. Es muss darauf vertrauen, dass die Projektteams selbstverantwortlich arbeiten und dass am Ende das rauskommt, was sie erwarten. Ein solcher Vertrauensvorschuss für die Mitarbeitenden und ein so hohes Maß an Autonomie, das passt nicht zu jeder Unternehmenskultur und auch nicht zu jeder Führungspersönlichkeit.

Agiles Projektmanagement setzt demnach eine ganz andere Sichtweise auf die Projektorganisation eines Unternehmens voraus. Ohne eine entsprechende Organisationskultur, die auf Vertrauen, Eigeninitiative und Verantwortungsbewusstsein setzt, und ohne die Mitarbeiterinnen und Mitarbeiter, die neben der Fachkompetenz auch soziale und methodische Qualifikationen benötigen, wird sich ein Unternehmen mit agilen Konzepten schwertun.

Warum sich Scrum trotz potenzieller Schwierigkeiten lohnt

Wer diese Barrieren überwindet, eine förderliche Organisationskultur schafft und die Mitarbeitenden dafür gewinnen kann, der erntet mit dem agilen Projektmanagement besonderen Erfolg. Wie beim schlanken Management wird bei der agilen Vorgehensweise zumeist die Qualität der Ergebnisse verbessert, und weniger Ressourcen werden verschwendet.

Alle fokussieren sich auf das Wesentliche. Die Produktivität steigt, aber auch die Zufriedenheit der Kunden und der Stakeholder. Die ständige und regelmäßige Kommunikation fördert die Transparenz. Engpässe werden schnell erkannt und können beseitigt werden. So kommt das Projekt insgesamt schneller voran.

Praxis

Zusammenarbeit im Scrum-Team überprüfen

Klären Sie regelmäßig, wie die Stimmung im Scrum-Team ist. Fragen Sie bei der täglichen Teambesprechung, wie die Projektmitarbeiterinnen und -mitarbeiter die persönliche Zusammenarbeit und das Vorankommen im Projekt einschätzen.



Happiness-Index für die Stimmung im Projektteam

Art.-Nr. 99.143.14

Datei business-wissen_9914314.xlsx

Microsoft Excel 2016

Teamentwicklung voranbringen

Überprüfen Sie den Zusammenhalt in Ihrem Scrum-Team. In welcher Phase der Teamentwicklung befindet sich das Projektteam? Stellen Sie fest, wie gut das Team zusammenarbeitet, wie sich die einzelnen Mitglieder einbringen und halten Sie dies in der folgenden Vorlage fest. Damit können Sie abschätzen, ob Maßnahmen für die Teamentwicklung notwendig sind.



Phasen der Teamfindung und Teamentwicklung im Scrum-Projekt

Art.-Nr. 99.143.15

Datei business-wissen_9914315.xlsx

Microsoft Excel 2016

Empfehlungen aus dem Management-Handbuch

Projektmanagement

Was Sie in jedem Projekt beachten und im Projektmanagement planen und steuern sollten. Mit Tipps, Vorlagen und Checklisten zur Projektplanung und zur Projektüberwachung.

<https://www.business-wissen.de/id/kapitel/89/>

Projektmanagement mit Excel

Excel-Tools für das Projektmanagement, mit denen Sie alle Projektaufgaben erledigen. Die Vorlagen sind für die Projektvorbereitung, Projektplanung, Projektsteuerung und den Projektabschluss. Sie können alle Excel-Vorlagen für Ihr Projekt leicht anpassen.

<https://www.business-wissen.de/id/kapitel/120/>

Projektkommunikation planen

Wie Sie die Akzeptanz und Motivation für Ihr Projekt im Projektteam und im Unternehmen erhöhen. Informieren Sie und fördern Sie den Dialog mit Stakeholdern Ihres Projekts. Mit einer Kommunikationsmatrix für die Maßnahmenplanung und einem Fragebogen für die Messung der Wirksamkeit der Projektkommunikation.

<https://www.business-wissen.de/id/kapitel/88/>

Nutzen Sie als
Premium-Mitglied
alle
Handbuch-Kapitel
mit mehr als
3.000 Checklisten und Excel-Vorlagen

Jetzt anmelden

www.business-wissen.de/anmelden/

Autoren

Dr. Jürgen Fleig



Dr. Jürgen Fleig ist Betreiber von www.business-wissen.de sowie Autor und Redakteur für Managementthemen. Er trainiert und berät seit 1990 in den Bereichen Konzeptentwicklung, Prozessgestaltung und Projektmanagement.

E-Mail redaktion@business-wissen.de
Telefon +49 721 18397-0
Anschrift b-wise GmbH
Bismarckstr. 21
76133 Karlsruhe
Deutschland
Web <https://www.business-wissen.de>

Redaktion business-wissen.de

E-Mail redaktion@business-wissen.de
Telefon +49 721 18397-0
Anschrift b-wise GmbH
Bismarckstr. 21
76133 Karlsruhe
Deutschland
Web <https://www.business-wissen.de>

Impressum

b-wise GmbH
Bismarckstraße 21
76133 Karlsruhe
DEUTSCHLAND

service@business-wissen.de
Telefon +49 721 18397-0

Copyright 2025, b-wise GmbH, All Rights Reserved