

# Scrum Guide

---

Der gültige Leitfaden für Scrum:  
Die Spielregeln



*Jeff Sutherland*



*Ken Schwaber*

*Juli 2011*

*Entwickelt und kontinuierlich verbessert von  
Ken Schwaber und Jeff Sutherland*

## Inhaltsverzeichnis

---

Zielsetzung des Scrum Guide.....	3
Scrum Überblick .....	3
Scrum Framework .....	3
Scrum Theorie .....	4
Scrum.....	5
Das Scrum Team .....	5
Der Product Owner.....	5
Das Entwicklungs-Team.....	6
Der Scrum Master.....	7
Scrum Ereignisse.....	8
Der Sprint.....	8
Sprint Sprint Planning.....	10
Daily Scrum .....	11
Sprint Review.....	12
Sprint Retrospective .....	13
Scrum Artefakte.....	13
Product Backlog.....	14
Sprint Backlog .....	15
Inkrement .....	16
Definition von“Done” .....	16
Fazit .....	17
Danksagungen .....	18
Menschen .....	18
Geschichte .....	18
Übersetzung .....	18

## Zielsetzung des Scrum Guide

Scrum ist ein Framework zur nachhaltigen Entwicklung komplexer Produkte. Dieser Leitfaden beinhaltet die Definition von Scrum. Diese Definition besteht aus den Scrum Rollen, Ereignissen, Artefakten, sowie den Regeln, die all dies zusammen halten. Ken Schwaber und Jeff Sutherland haben Scrum entwickelt; der Scrum Guide wird von ihnen verfasst und veröffentlicht. Beide stehen gemeinsam hinter dem Scrum Guide.

## Scrum Überblick

Scrum (n): Ein Rahmenwerk, mit dessen Hilfe Menschen komplexe adaptive Aufgabenstellungen angehen können, und durch das sie in die Lage versetzt werden, produktiv und kreativ Produkte mit dem höchstmöglichen Wert auszuliefern. Scrum ist:

- Leichtgewichtig
- Einfach zu verstehen
- Extrem schwer zu meistern

Scrum wird seit den frühen 1990er Jahren als Prozessframework bei der Umsetzung komplexer Produktentwicklungen verwendet. Scrum ist weder ein Prozess noch eine Technik zur Erstellung von Produkten; es ist vielmehr als Framework zu verstehen, innerhalb dessen verschiedene Prozesse und Techniken zum Einsatz gebracht werden können. Scrum macht die relative Wirksamkeit Ihres Produktmanagements und Entwicklungsvorgehens sichtbar, so dass Schritte zur Verbesserung eingeleitet werden können.

## Scrum Framework

Das Scrum Framework besteht aus Scrum Teams und den mit ihnen verbundenen Rollen, Ereignissen, Artefakten und Regeln. Jede Komponente des Frameworks dient einem spezifischen Zweck und ist wesentlich für die erfolgreiche Anwendung von Scrum.

Bestimmte Strategien zur Nutzung des Scrum Frameworks können variieren; sie sind an anderer Stelle beschrieben.

Durch die Regeln von Scrum werden die Beziehungen und Wechselwirkungen zwischen den Ereignissen, Rollen und Artefakten bestimmt. Die Regeln von Scrum sind in diesem Dokument beschrieben.

## Scrum Theorie

Scrum basiert auf der Theorie der empirischen Prozesssteuerung, oder des Empirismus. Empirismus basiert auf der Grundannahme, dass Wissen aus Erfahrung gewonnen wird **und dass** Entscheidungen auf Grundlage bekannter Fakten getroffen werden. In Scrum wird zur Verbesserung der Vorhersagbarkeit und Risikokontrolle ein iterativer und inkrementeller Ansatz verfolgt.

Jede Anwendung empirischer Prozesssteuerung stützt sich auf die drei Säulen: Transparenz, Überprüfung und Anpassung (inspection & adaptation).

### Transparenz

Die wesentlichen Aspekte des Prozesses müssen für diejenigen Personen erkennbar sein, die für das Prozessergebnis verantwortlich sind. Transparenz erfordert, dass diese Aspekte durch einen gemeinsamen Standard definiert sind, so dass Beobachter ein gemeinsames Verständnis über das Beobachtete teilen.

Beispiel:

- Eine gemeinsame Sprache, die sich auf den Prozess bezieht, muss von allen Beteiligten verwendet werden; und,
- Eine gemeinsame Definition von "Done"<sup>1</sup> muss sowohl von den arbeitenden als auch durch die das Arbeitsergebnis abnehmenden Personen geteilt werden.

### Überprüfung (Inspection)

Scrum Anwender müssen ständig die Scrum Artefakte und den Fortschritt auf dem Weg zur Erreichung des Ziels überprüfen, um unerwünschte Abweichungen zu entdecken. Die Häufigkeit der Überprüfungen sollte in dem Zusammenhang nicht so hoch sein, dass die eigentliche Arbeit dadurch behindert wird. Der größte Nutzen kann aus einer Überprüfung gezogen werden, wenn sie ständig durch sachkundige Personen am Ort des Geschehens vorgenommen wird.

### Anpassung (Adaptation)

Wenn bei einer Überprüfung festgestellt wird, dass einer oder mehrere Aspekte des Prozesses außerhalb akzeptabler Grenzen liegen und das Produktergebnis dadurch ebenfalls nicht zu akzeptieren sein wird, muss so schnell wie möglich eine Anpassung des Prozesses oder des Arbeitsgegenstandes vorgenommen werden, um weitere Abweichungen zu minimieren.

---

<sup>1</sup> Siehe Definition of "Done", S. 16.

Scrum schreibt vier offizielle Gelegenheiten für Überprüfungen und Anpassungen (inspection and adaptation) vor, die im Abschnitt *Scrum Ereignisse* dieses Dokumentes beschrieben werden.

- Das Sprint Planning Meeting
- Das Daily Scrum
- Das Sprint Review Meeting
- Die Sprint Retrospektive

## Scrum

Scrum ist ein Framework, das die Entwicklung komplexer Produkte unterstützt. Scrum besteht aus Scrum Teams und den zugehörigen Rollen, Ereignissen, Artefakten und Regeln. Jede Komponente des Frameworks dient einem bestimmten Zweck und ist wesentlich für die erfolgreiche Anwendung von Scrum.

Durch die Regeln von Scrum werden die Beziehungen und Wechselwirkungen zwischen den Ereignissen, Rollen und Artefakten bestimmt. Die Regeln von Scrum sind in diesem Dokument beschrieben.

## Das Scrum Team

Das Scrum Team besteht aus dem Product Owner, dem Entwicklungs-Team, und dem Scrum Master. Scrum Teams sind selbstorganisiert und interdisziplinär. Selbstorganisierte Teams entscheiden eigenständig darüber, wie die zu erledigende Arbeit am besten bewältigt werden kann. Es gibt niemanden, der einem selbstorganisierten Team von außen vorgibt, wie die Arbeit zu erledigen ist. Interdisziplinäre Teams verfügen über alle Kompetenzen, ihr Arbeitsergebnis zu erreichen, ohne dabei von anderen abhängig zu sein, die nicht Teil des Teams sind. Das hier beschriebene Scrum Team-Modell hat zur Aufgabe, die Flexibilität, Kreativität und Produktivität des Teams beständig zu verbessern.

Scrum Teams liefern Produkte in regelmäßigen Abständen (iterativ) und stufenweise erweiternd (inkrementell) aus. Dadurch werden die Gelegenheiten für Feedback maximiert. Die inkrementelle Lieferung "fertiger" Produkte stellt sicher, dass immer eine potentiell gebrauchsfertige und nutzbare Version des Produktes zur Verfügung steht.

## Der Product Owner

Der Product Owner ist für die Maximierung des Wertes des Produkts und der Arbeit, die das Entwicklungs-Team verrichtet, verantwortlich. Für unterschiedliche Organisationen, verschiedene Scrum Teams und Einzelpersonen kann es viele Wege zur Erreichung dieses Ziels geben.

Der Product Owner ist als einzige Person für die Verwaltung des Product Backlog verantwortlich.

„Product Backlog Verwaltung“ beinhaltet:

- Die klare Formulierung der Einträge im Product Backlog;
- Die Anordnung der Einträge („ordering“) in der gewünschten Fertigstellungsreihenfolge, durch welche die Ziele des Vorhabens optimal unterstützt werden;
- Die Sicherstellung der Wertschöpfung des Entwicklungs-Teams
- Die Sicherstellung, dass das Product Backlog für alle Beteiligten einsehbar, transparent sowie klar ist und dass es anzeigt, woran das Scrum Team als nächstes arbeiten wird; und
- Die Sicherstellung, dass das Entwicklungs-Team die Einträge des Product Backlog im erforderlichen Maße versteht.

Der Product Owner kann die aufgezählten Tätigkeiten selbst ausführen, oder durch das Entwicklungs-Team erledigen lassen. In jedem Fall bleibt der Product Owner verantwortlich.

Der Product Owner ist eine Person, kein Ausschuss oder Gremium. Der Product Owner kann ein Gremium repräsentieren, jedoch ist es nur dem Product Owner erlaubt, die Fertigstellungsreihenfolge der Einträge des Product Backlog zu verändern. Jede andere Person, die Einfluss auf diese Reihenfolge nehmen will, muss den Product Owner von der Änderung überzeugen.

Die Entscheidungen des Product Owners müssen durch die gesamte Organisation respektiert werden, anderenfalls kann der Product Owner seine Rolle nicht verantwortlich ausfüllen und auch keinen Erfolg haben. Die Entscheidungen des Product Owners manifestieren sich im Inhalt und in der Anordnung des Product Backlogs. Niemand darf das Entwicklungs-Team anweisen, mit anderen Anforderungen als den im Product Backlog festgelegten zu arbeiten und dem Entwicklungs-Team ist es nicht erlaubt, Arbeitsanweisungen von anderen Personen als dem Product Owner anzunehmen.

## Das Entwicklungs-Team

Das Entwicklungs-Team besteht aus Fachleuten, deren Aufgabe die Erstellung eines potentiell auslieferbaren und „fertigen“ („done“) Produktinkrements zum Ende eines jeden Sprints ist. Nur die Mitglieder des Entwicklungs-Teams erstellen dieses Inkrement.

Entwicklungs-Teams sind durch die umgebende Organisation zu selbstorganisierter Arbeit ermächtigt und so strukturiert, dass sie ihre Arbeit selbst verwalten können. Die sich daraus ergebende Synergie optimiert die Effizienz und Effektivität des Entwicklungs-Teams. Entwicklungs-Teams sind durch die folgenden Merkmale gekennzeichnet:

- Sie sind selbstorganisierend. Niemand (auch nicht der Scrum Master) macht dem Entwicklungs-Team Vorgaben, wie die Einträge des Product Backlogs in ein potentiell auslieferbares Produktinkrement transformiert werden;
- Entwicklungs-Teams sind interdisziplinär besetzt und verfügen als Team über alle nötigen Kompetenzen, die für die Erzeugung eines Produktinkrements erforderlich sind;
- Unabhängig von der verrichteten Arbeit werden die Mitglieder des Entwicklungs-Teams als Entwickler bezeichnet. Scrum sieht keine anderen Titel für die Mitglieder eines Entwicklungs-Teams vor; es gibt keine Ausnahme von dieser Regel;

- Einzelne Mitglieder eines Entwicklungs-Teams können spezialisierte Fertigkeiten und Kenntnisse haben, die zu einer Fokussierung auf spezifische Arbeitsbereiche führen. dennoch liegt die Verantwortlichkeit für das Arbeitsergebnis beim Entwicklungs-Team als Ganzes; und
- Entwicklungs-Teams enthalten keine Sub-Teams, die ausschließlich in definierten Arbeitsbereichen wie Test oder Analyse tätig sind.

### Größe des Entwicklungs-Teams

Ein Entwicklungs-Team sollte klein genug sein, um behände und flink zu bleiben, aber groß genug, um alle relevanten Arbeiten erledigen zu können. Ein Entwicklungs-Team mit weniger als drei Mitgliedern führt zu reduzierten Wechselwirkungen innerhalb des Teams und verringert die Produktivitätszuwächse. Kleine Entwicklungs-Teams laufen Gefahr, nicht über alle Kompetenzen und Fertigkeiten zu verfügen, die für die Verrichtung der Arbeit im Sprint benötigt werden, so dass die Auslieferung des Produktinkrements zum Sprint-Ende gefährdet sein kann. Entwicklungs-Teams mit mehr als neun Mitgliedern erfordern zu viel Koordinationsaufwand. Große Entwicklungs-Teams erzeugen eine zu hohe Komplexität, um durch einen empirischen Prozess gesteuert werden zu können. Product Owner und Scrum Master werden bei diesen Betrachtungen der Teamgröße nicht mitgezählt, es sei denn, sie arbeiten als Entwickler an den Elementen des Sprint Backlogs mit.

### Der Scrum Master

Der Scrum Master ist dafür verantwortlich, dass Scrum verstanden und korrekt angewandt wird. Dies erreichen Scrum Master durch die Sicherstellung, dass das Scrum Team sich konform zur Scrum Theorie verhält und die gültigen Praktiken und Regeln einhält. Der Scrum Master ist eine dienende Führungskraft („servant-leader“) für das Scrum Team.

Der Scrum Master unterstützt die Akteure außerhalb des Scrum Teams dabei, zu verstehen, welche ihrer Verhaltensweisen in Bezug auf das Scrum Team hilfreich sind und welche nicht. Der Scrum Master hilft außerdem, diese Verhaltensweisen so zu verändern, dass das Scrum Team einen maximalen Geschäftswert erzeugen kann.

### Der Dienst des Scrum Masters für den Product Owner

Der Scrum Master dient dem Product Owner unter anderem indem er:

- Techniken zur effektiven Verwaltung des Product Backlogs etabliert;
- Vision, Ziele und Einträge des Product Backlogs gegenüber dem Entwicklungs-Team klar kommuniziert;
- das Entwicklungs-Team darin unterrichtet, klare und schlüssige Product Backlog Einträge zu erstellen;
- Ein Verständnis dafür schafft, wie langfristige Produktplanung in einem empirischen Arbeitsumfeld funktioniert;
- ein Verständnis dafür schafft, was Agilität ist und wie man sie lebt;
- Scrum Ereignisse auf Anfrage oder nach Bedarf unterstützt („facilitate“).

## Der Dienst des Scrum Masters für das Entwicklungs-Team

Der Scrum Master dient dem Entwicklungs-Team. Dazu gehört:

- Coaching des Entwicklungs-Teams zu selbstorganisiertem und interdisziplinärem Handeln;
- Unterrichtung und Führung des Entwicklungs-Teams bei der Erstellung von Produkten mit hohem Wert;
- Entfernung von Hindernissen, die den Arbeitsfortschritt des Entwicklungs-Teams aufhalten;
- Unterstützung („facilitation“) von Scrum Ereignissen auf Anfrage oder nach Bedarf;
- Coaching des Entwicklungs-Teams in Unternehmensumfeldern, in denen Scrum noch nicht voll verstanden oder genutzt wird.

## Der Dienst des Scrum Masters für die Organisation

Der Scrum Master dient der Organisation. Dazu gehört:

- Führung und Coaching der Organisation bei der Umsetzung von Scrum;
- Planung von Scrum-Einführungen in der Organisation;
- Unterstützung der Mitarbeiter und Interessenvertreter dabei, Scrum und empirische Produktentwicklung zu verstehen und richtig anzuwenden;
- Anstoßen von Veränderungsprozessen, welche die Produktivität des Scrum Teams verbessern; und
- Zusammenarbeit mit anderen Scrum Mastern, um den effektiven Einsatz von Scrum innerhalb der Organisation voranzutreiben.

## Scrum Ereignisse

Vorgeschriebene Ereignisse werden in Scrum verwendet, um eine Regelmäßigkeit herzustellen und die Notwendigkeit für andere Besprechungen zu minimieren. Für alle Ereignisse sind in Scrum Zeitfenster („time-boxes“) vorgesehen, durch welche die maximale Dauer jedes Ereignisses begrenzt wird. Dadurch wird ein angemessener Zeitanteil für Planungszwecke vorgesehen, der keine Verschwendung zulässt.

Anders als der Sprint, der ein Container für alle anderen Scrum Ereignisse ist, bieten die übrigen Ereignisse in Scrum die Möglichkeit, Sachverhalte zu beobachten und anzupassen („inspect and adapt“). Die Ereignisse sind speziell dazu entworfen, kritische Transparenz zu erzeugen und eine Untersuchung zu ermöglichen. Werden Ereignisse ausgelassen, führt das zu reduzierter Transparenz und es wird eine Chance zur Überprüfung und Anpassung vergeben.

## Der Sprint

Der Sprint ist das Herz von Scrum. Der Sprint ist ein Zeitfenster von einem Monat oder kürzer, während dessen ein „done“ (fertiges), nutzbares und potentiell auslieferbares Produktinkrement hergestellt wird. Sprints haben während eines Entwicklungsvorhabens eine gleichmäßig feste Dauer. Jeder neue Sprint beginnt direkt nach der Beendigung des vorhergehenden Sprints.

Sprints enthalten und bestehen aus dem Sprint Planungstreffen, den Daily Scrums, der eigentlichen Entwicklungsarbeit, dem Sprint Review und der Sprint Retrospektive.

Während des Sprints:

- Werden keine Änderungen vorgenommen, die das Sprint-Ziel verändern;
- Bleiben die Zusammensetzung des Entwicklungs-Teams und die qualitativen Ziele des Sprints unverändert;
- Kann auf Grundlage von aus dem Sprint gewonnenen Erkenntnissen der inhaltliche Umfang zwischen dem Product Owner und dem Entwicklungs-Team genauer definiert und neu verhandelt werden.

Jeder Sprint kann als Projekt verstanden werden, für das ein zeitlicher Rahmen von maximal einem Monat zur Verfügung steht. Ebenso wie Projekte, werden Sprints dazu verwendet, um etwas zu erreichen. Jeder Sprint ist gekennzeichnet durch eine Definition dafür, was erstellt werden soll, einen Entwurf für die Erstellung, einen flexiblen Plan für die Umsetzung sowie die eigentliche Arbeit und das resultierende Arbeitsergebnis.

Sprints sind zeitlich auf einen Kalendermonat begrenzt. Wenn der Betrachtungshorizont für einen Sprint zu groß ist, kann sich die Definition des gewünschten Ergebnisses ändern sowie die Komplexität und das Risiko zunehmen. Sprints ermöglichen Vorhersagbarkeit durch garantierte und mindestens monatliche Überprüfung und Anpassung des Arbeitsfortschrittes gegen ein Ziel. Außerdem begrenzen Sprints das Kostenrisiko auf die Kosten, die innerhalb eines Monats entstehen.

### Abbruch eines Sprints

Ein Sprint kann vor seinem vorgesehenen Ende abgebrochen werden. Nur der Product Owner ist befugt, einen Sprint abubrechen, obwohl dies unter Berücksichtigung des Einflusses von Interessenvertretern, des Entwicklungs-Teams oder des Scrum Masters geschehen kann.

Ein Sprint würde dann abgebrochen, wenn das Sprint-Ziel überflüssig wird. Das kann z.B. eintreten, wenn das Unternehmen eine Richtungsänderung vornimmt oder wenn sich technologische oder Marktbedingungen verändert haben. Ganz allgemein sollte ein Sprint abgebrochen werden, wenn seine Fortführung unter den gegebenen Bedingungen nicht mehr sinnvoll ist. Unter Berücksichtigung der kurzen Dauer eines Sprints sollte das allerdings selten vorkommen.

Bei Abbruch eines Sprints werden alle bearbeiteten und als "Done" gekennzeichneten Product Backlog Elemente überprüft. Wenn Teile der fertig gestellten Arbeit potentiell auslieferbar sind, wird der Product Owner diese Arbeit im Regelfall akzeptieren. Die nicht fertig gestellten Elemente des Product Backlogs werden neu geschätzt und ins Product Backlog zurück genommen. Die Arbeit, die an diesen Einträgen verrichtet wurde, wird schnell an Wert verlieren und muss regelmäßig neu geschätzt werden.

Sprintabbrüche verbrauchen Ressourcen, weil die Beteiligten sich für den Beginn eines neuen Sprints in einem erneuten Sprint Planning sammeln und orientieren müssen.

Außerdem sind Sprintabbrüche oft ein traumatisches Erlebnis für das Scrum Team und sehr unüblich.

## **Sprint Planning**

Die Arbeit, die während des Sprints zu erledigen ist, wird im Sprint Planning geplant. Der Arbeitsplan wird in gemeinschaftlicher Arbeit durch das gesamte Scrum Team erstellt.

Das Sprint Planning ist für einen einmonatigen Sprint auf ein Zeitfenster von acht Stunden begrenzt (time-box). Für kürzere Sprints wird die Dauer des Sprint Plannings proportional angepasst. Zum Beispiel dauert das Sprint Planning für einen zweiwöchigen Sprint vier Stunden.

Das Sprint Planning ist in zwei Teile aufgliedert, von denen jeder Teil die Hälfte der Dauer einnimmt, die für die gesamte Besprechung zur Verfügung steht. Jeder der beiden Teile liefert eine Antwort auf eine spezifische Fragestellung:

- Was wird im Produktinkrement als Ergebnis des aktuellen Sprints ausgeliefert?
- Wie wird die Arbeit umgesetzt, um das gewünschte Ergebnis zu erreichen?

### **Teil Eins: Was wird in diesem Sprint getan?**

In diesem Teil erarbeitet das Entwicklungs-Team eine Prognose („forecast“) für die Funktionalität, die im Sprint entwickelt wird. Der Product Owner stellt dem Entwicklungs-Team die geordneten Einträge des Product Backlogs vor und das gesamte Scrum Team verschafft sich gemeinsam ein Verständnis über die im Sprint zu verrichtende Arbeit.

Eingabedaten für diese Besprechung sind das Product Backlog, das letzte Produktinkrement, die erwartete Kapazität des Entwicklungs-Teams für den kommenden Sprint und die bisher beobachtete Entwicklungsgeschwindigkeit des Entwicklungs-Teams. Ausschließlich das Entwicklungs-Team bestimmt, wie viele Einträge aus dem Product Backlog zur Erledigung im Sprint ausgewählt werden, da nur das Entwicklungs-Team bewerten kann, was es im kommenden Sprint zu leisten vermag.

Nachdem das Entwicklungs-Team eine Prognose darüber abgegeben hat, welche Elemente des Product Backlogs im Sprint geliefert werden, wird das Sprint-Ziel durch das Scrum Team erarbeitet. Das Sprint-Ziel ist eine Vorgabe, die innerhalb des Sprints durch die Umsetzung der Einträge des Product Backlogs verwirklicht wird. Das Sprint-Ziel begründet dem Entwicklungs-Team, warum das kommende Produktinkrement erstellt wird.

### **Teil Zwei: Wie wird die ausgewählte Arbeit fertig gestellt?**

Nachdem die Arbeit für den Sprint ausgewählt wurde, entscheidet das Entwicklungs-Team, wie diese Funktionalität während des Sprints in Form eines „done“ (fertigen) Inkrements umgesetzt wird. Die für den Sprint ausgewählten Einträge des Product Backlogs zusammen mit dem Plan, wie diese Einträge ausgeliefert werden, werden als Sprint Backlog bezeichnet.

Das Entwicklungs-Team beginnt für gewöhnlich mit dem Systementwurf und der Erstellung eines Arbeitsplans zur Transformation der Einträge aus dem Product Backlog in ein funktionierendes Produktinkrement. Die resultierende Arbeitsmenge oder der geschätzte

Aufwand können variieren, es wird während der Besprechung jedoch genug Arbeit geplant, damit dem Entwicklungs-Team eine Prognose darüber möglich ist, was es im Sprint leisten kann. Am Ende der Besprechung hat das Team die in den ersten Tagen des Sprints zu verrichtende Arbeit in kleine Einheiten zergliedert, die innerhalb eines Tages oder in kürzerer Zeit erledigt werden können. Sowohl im Sprint Planning als auch während des Sprints arbeitet das Entwicklungs-Team selbstorganisiert, um die Arbeit aus dem Sprint Backlog zu erfüllen.

Der Product Owner kann während dieses zweiten Teils der Besprechung anwesend sein, um für Rückfragen zu den ausgewählten Einträgen des Product Backlogs zur Verfügung zu stehen und um bei der Auflösung von Zielkonflikten zu helfen. Wenn das Entwicklungs-Team feststellt, dass es zu viel oder zu wenig Arbeit für den Sprint ausgewählt hat, kann es den Inhalt des Sprint Backlogs mit dem Product Owner erneut verhandeln. Das Entwicklungs-Team kann auch außen stehende Dritte zum Sprint Planning einladen, um Beratung in technischen oder fachlichen Fragen zu erhalten.

Am Ende des Sprint Plannings sollte das Entwicklungs-Team in der Lage sein, dem Product Owner und dem Scrum Master zu erläutern, wie es beabsichtigt, das Sprint-Ziel selbstorganisiert zu erreichen und wie das erwartete Produktinkrement erstellt werden soll.

### **Sprint-Ziel**

Das Sprint-Ziel gibt dem Entwicklungs-Team eine gewisse Flexibilität bei der Umsetzung der Funktionalität innerhalb des Sprints.

Während das Entwicklungs-Team arbeitet, behält es das Sprint-Ziel fest im Blick. Zur Erreichung des Sprint-Ziels implementiert das Team Funktionalitäten und Technologien. Stellt sich heraus, dass die Arbeit sich anders gestaltet als erwartet, arbeitet das Entwicklungs-Team gemeinsam mit dem Product Owner, um den Umfang des Sprint Backlogs während des Sprints neu zu verhandeln.

Das Sprint-Ziel kann ein Meilenstein innerhalb einer weiter gefassten Zielsetzung des strategischen Produktentwicklungspfades sein.

### **Daily Scrum**

Das Daily Scrum ist ein auf eine 15-minütige Timebox beschränktes Ereignis, in dem das Entwicklungs-Team seine Aktivitäten synchronisiert und einen Plan für die nächsten 24 Stunden erstellt. Das geschieht durch die Überprüfung der Arbeit seit dem letzten Daily Scrum und die Prognose der Arbeit, die bis zum nächsten Daily Scrum fertig gestellt werden könnte.

Das Daily Scrum wird jeden Tag zur gleichen Zeit am gleichen Ort abgehalten, um die Komplexität zu reduzieren. Im Meeting schildert jedes Mitglied des Entwicklungs-Teams:

- Was wurde seit dem letzten Meeting erreicht?
- Was wird vor dem nächsten Meeting erledigt?
- Welche Hindernisse („Impediments“) sind dabei im Weg?

Das Entwicklungs-Team nutzt das Daily Scrum, um seinen Fortschritt in Richtung des Sprintziels einzuschätzen, und sich ein Bild darüber zu verschaffen, wie der Trend in Richtung Fertigstellung der Arbeit im Sprint Backlog aussieht. Das Daily Scrum optimiert die Wahrscheinlichkeit, dass das Entwicklungs-Team sein Sprintziel erreicht. Das Entwicklungs-Team trifft sich häufig unmittelbar nach dem Daily Scrum, um den Rest der Arbeit im Sprint neu zu planen. Das Entwicklungs-Team sollte jeden Tag in der Lage sein, dem Product Owner und Scrum Master zu schildern, wie es bis zum Ende des Sprints als selbstorganisiertes Team zusammen arbeiten möchte, um das Ziel zu erreichen und das erwartete Inkrement fertig zu stellen.

Der Scrum Master stellt sicher, dass das Entwicklungs-Team dieses Meeting abhält, aber das Entwicklungs-Team ist selbst für die Durchführung des Daily Scrums verantwortlich. Der Scrum Master vermittelt dem Entwicklungs-Team, wie es das Daily Scrum innerhalb der 15-minütigen Timebox abhalten kann.

Der Scrum Master setzt auch die Einhaltung der Regel durch, dass nur Mitglieder des Entwicklungs-Teams am Daily Scrum teilnehmen. Das Daily Scrum ist kein Statusmeeting, sondern für die Mitarbeiter gedacht, welche die Product Backlog Einträge in ein Inkrement transformieren.

Daily Scrums verbessern die Kommunikation, machen andere Meetings überflüssig, identifizieren und identifizieren sowie beseitigen Hindernisse für die Entwicklung, fokussieren und fördern schnelle Entscheidungsprozesse und verbessern das Projektwissen im Entwicklungs-Team. Dadurch sind sie ein Schlüssel-Meeting für die Inspektion und Adaption.

## Sprint Review

Ein Sprint Review Meeting wird am Ende des Sprints abgehalten, um das Inkrement zu untersuchen, und das Product Backlog, wenn nötig, anzupassen. Im Sprint Review ermitteln das Scrum Team und die Stakeholder gemeinsam, was in dem Sprint fertig gestellt wurde. Auf der Basis dessen – und möglicher Änderungen am Product Backlog während des Sprints – erarbeiten die Teilnehmer die nächsten Dinge, die angegangen werden könnten. Es handelt sich hierbei um ein informelles Meeting; die Präsentation des Inkrements ist dazu gedacht, Feedback hervorzubringen und Zusammenarbeit zu fördern.

Das Sprint Review Meeting ist auf eine Timebox von vier Stunden für einmonatige Sprints begrenzt. Für kürzere Sprints wird entsprechend weniger Zeit eingeplant. So haben zum Beispiel zweiwöchige Sprints zweistündige Sprint Reviews.

Das Sprint Review beinhaltet die folgenden Elemente:

- Der Product Owner ermittelt, was „done“ (fertig) ist, und was nicht;
- Das Entwicklungs-Team diskutiert, was im Sprint gut lief, mit welchen Problemen es konfrontiert wurde, und wie es diese Probleme gelöst hat;
- Das Entwicklungs-Team demonstriert die Arbeit, die es fertig gestellt hat und beantwortet Fragen zum Inkrement;

- Der Product Owner erläutert den aktuellen Stand des Product Backlogs. Er oder sie rechnet wahrscheinliche Fertigstellungsdaten auf der Basis des momentanen Fortschritts hoch; und,
- Die gesamte Gruppe erarbeitet gemeinsam, was als nächstes angegangen werden sollte, damit das Sprint Review wertvollen Input für zukünftige Sprint Planning Meetings liefert.

Das Ergebnis des Sprint Reviews ist ein überarbeitetes Product Backlog, das die wahrscheinlichen Product Backlog Elemente für den nächsten Sprint definiert. Das Product Backlog kann auch insgesamt angepasst werden, um neue Gelegenheiten wahrnehmen zu können.

## Sprint Retrospektive

Die Sprint Retrospektive ist eine Gelegenheit für das Scrum Team, sich selbst zu untersuchen, und einen Plan für Verbesserungen aufzustellen, die im folgenden Sprint umgesetzt werden sollen.

Die Sprint Retrospektive findet nach dem Sprint Review und vor dem nächsten Sprint Planning Meeting statt. Sie ist auf eine Timebox von drei Stunden bei einmonatigen Sprints festgelegt. Proportional weniger Zeit wird für kürzere Sprints aufgewendet.

Das Ziel einer Sprint Retrospektive ist:

- Zu untersuchen, wie der letzte Sprint in Bezug auf die Mitarbeiter, Beziehungen, Prozesse und Werkzeuge verlaufen ist;
- Zu identifizieren und zu ordnen was gut lief und was möglicherweise verbessert werden könnte; und,
- Einen Umsetzungsplan für die Verbesserungen der Arbeitsweise des Scrum Teams anzufertigen.

Der Scrum Master ermutigt das Scrum Team, innerhalb des Scrum Prozess Frameworks seinen Entwicklungsprozess und seine Praktiken zu verbessern, um beide effektiver und befriedigender (Anm. d. Ü.: Hier ist durchaus auch „Spaß an der Arbeit“ gemeint) für den nächsten Sprint zu gestalten. In jeder Sprint Retrospektive plant das Team Wege, die Produktqualität zu erhöhen, indem es die „Definition of Done“ angemessen anpasst.

Zum Abschluss der Sprint Retrospektive sollte das Scrum Team Verbesserungen identifiziert haben, die es im nächsten Sprint umsetzen wird. Die Implementierung dieser Verbesserungen im nächsten Sprint ist die Anpassung als Folge der Selbstüberprüfung des Entwicklungs-Teams. Obwohl Verbesserungen zu jedem Zeitpunkt eingebracht werden können, bietet die Sprint Retrospektive ein dediziertes, auf die Inspektion und Adaption fokussiertes Ereignis.

## Scrum Artefakte

Die Artefakte von Scrum repräsentieren Arbeit oder Wert auf verschiedene nützliche Weisen, die Transparenz und Gelegenheit zur Überprüfung und Anpassung schaffen. Die in

Scrum definierten Artefakte sind spezifisch dafür entwickelt worden, die Transparenz über Schlüsselinformationen zu maximieren, die sicherstellen, dass Scrum Teams erfolgreich ein „done“ (fertiges) Inkrement liefern können.

## Product Backlog

Das Product Backlog ist eine geordnete Liste mit allem, was in dem Produkt benötigt werden könnte. Es ist die einzige Quelle von Anforderungen für jedwede Änderungen an dem Produkt. Der Product Owner ist für das Product Backlog verantwortlich, inklusive dessen Inhalte, Bereitstellung und Reihenfolge.

Ein Product Backlog ist niemals vollständig. Seine früheste Entwicklung skizziert nur die initial bekannten und am Besten verstandenen Anforderungen. Das Product Backlog entwickelt sich weiter, genau so wie das Produkt und die Umgebung, in der es eingesetzt wird, sich weiterentwickeln. Das Product Backlog ist dynamisch; es ändert sich permanent, um die Anforderungen an das Produkt in Bezug auf seine Angemessenheit, Wettbewerbsfähigkeit und Nützlichkeit zu identifizieren. So lange ein Produkt existiert, existiert auch ein Product Backlog.

Das Product Backlog führt alle Features, Funktionen, Anforderungen, Verbesserungen und Fehlerbehebungen auf, welche die Änderungen an dem Produkt in zukünftigen Auslieferungen ausmachen. Product Backlog Einträge verfügen über die Attribute Beschreibung, Rangfolge und Schätzung.

Das Product Backlog wird häufig nach Wert, Risiko, Priorität und Notwendigkeit angeordnet. Die Product Backlog Einträge an der Spitze bestimmen die unmittelbaren Entwicklungsaktivitäten. Je höher in der Rangfolge, desto mehr Betrachtung hat ein Product Backlog Eintrag erfahren – und desto größer ist der Konsens über seine Inhalte und seinen Wert.

Höherrangige Product Backlog Einträge sind klarer und detaillierter als niedriger angeordnete. Präzisere Schätzungen erfolgen auf der Basis der größeren Klarheit und Detailtiefe; je niedriger der Rang, desto weniger Detail. Product Backlog Einträge, mit denen sich das Team im kommenden Sprint beschäftigen wird, sind fein granuliert und derart herunter gebrochen, dass jeder Eintrag innerhalb der Timebox des Sprints fertig gestellt werden kann. Product Backlog Einträge, die von dem Entwicklungs-Team innerhalb eines Sprints fertig gestellt werden können, werden als „bereit“ („ready“) oder „einsetzbar“ für die Auswahl in einem Sprint Planning Meeting angesehen.

Im Laufe des Einsatzes des Produkts, seiner Wertschöpfung, und durch die Rückmeldungen des Marktes, wird das Product Backlog zu einer größeren und umfangreicheren Liste. Anforderungen hören nie auf sich zu ändern, was das Product Backlog zu einem lebenden Artefakt macht. Änderungen in den Geschäftsanforderungen, Marktbedingungen oder der Technologie können Änderungen im Product Backlog bedingen.

Häufig arbeiten mehrere Scrum Teams zusammen am gleichen Produkt. Ein einziges Product Backlog wird zur Beschreibung der anstehenden Arbeit an dem Produkt verwendet. Dazu wird ein Attribut zur Gruppierung der Product Backlog Einträge eingeführt.

Die Pflege („Grooming“) des Product Backlogs ist die Tätigkeit der Detaillierung, Schätzung und Umordnung der Einträge im Product Backlog. Es handelt sich dabei um einen fortlaufenden Prozess, in dem der Product Owner und das Entwicklungs-Team gemeinsam an den Details der Product Backlog Einträge arbeiten. Während der Pflege werden Einträge überprüft und überarbeitet. Darüber hinaus können sie jedoch grundsätzlich zu jedem Zeitpunkt durch den Product Owner oder mit seinem Einverständnis aktualisiert werden.

Die Pflege ist eine von Product Owner und Entwicklungs-Team während eines Sprints durchgeführte Teilzeitaktivität. Häufig besitzt das Entwicklungs-Team das Domänenwissen, um diese Überarbeitung selbst durchzuführen. Wie und wann diese Überarbeitung stattfindet, wird im Scrum Team entschieden. Die Pflege erfordert im Regelfall nicht mehr als 10% der Kapazität des Entwicklungs-Teams.

Das Entwicklungs-Team ist für alle Schätzungen verantwortlich. Der Product Owner kann Einfluss auf das Team nehmen, indem er für ein besseres Verständnis sorgt oder Kompromisse akzeptiert – aber die Mitarbeiter, welche die Arbeit erledigen, haben das letzte Wort bei der Schätzung.

### **Fortschritt in Richtung eines Ziels überwachen**

Zu jedem Zeitpunkt kann die gesamte zur Zielerreichung verbleibende Arbeit aufsummiert werden. Der Product Owner verfolgt diese gesamte restliche Arbeit zumindest zu jedem Sprint Review. Der Product Owner vergleicht diesen Betrag mit der restlichen Arbeit bei früheren Sprint Reviews, um den Fortschritt der Fertigstellung der geplanten Arbeit in Bezug auf den gewünschten Fertigstellungszeitpunkt des Ziels zu beurteilen. Diese Information wird allen Stakeholdern transparent gemacht.

Scrum betrachtet die für Product Backlog Einträge aufgewendete Zeit nicht. Die restliche Arbeit und das Datum sind die einzigen Variablen von Interesse.

Verschiedene Trend-Burndown-, Burnup- und andere projektive Praktiken wurden benutzt, um den Fortschritt vorherzusagen. Diese haben sich als nützlich erwiesen, sie ersetzen jedoch nicht die Wichtigkeit des Empirismus. In komplexen Umgebungen ist unbekannt, was geschehen wird. Nur was bereits geschehen ist lässt sich für die zukunftsorientierte Entscheidungsfindung nutzen.

### **Sprint Backlog**

Das Sprint Backlog ist die Menge der für den Sprint ausgewählten Product Backlog Einträge sowie der Plan für die Lieferung des Produktinkrements und die Erfüllung des Sprintziels. Das Sprint Backlog ist eine Prognose des Entwicklungs-Teams darüber, welche Funktionalität im nächsten Inkrement enthalten sein wird, und welche Arbeiten erforderlich sind, um diese Funktionalität zu liefern.

Das Sprint Backlog definiert die Arbeit, die das Entwicklungs-Team durchführen wird, um Product Backlog Einträge in ein „fertiges“ Inkrement zu überführen. Das Sprint Backlog macht all die Arbeit sichtbar, die das Entwicklungs-Team für notwendig erachtet, um das Sprintziel zu erreichen.

Das Sprint Backlog ist ein Plan mit ausreichender Detailtiefe, um Änderungen beim Fortschritt im Daily Scrum erkennen zu können. Das Entwicklungs-Team passt das Sprint Backlog während des Sprints an; das Sprint Backlog geht auch aus den Arbeiten im Sprint hervor. Diese Emergenz geschieht durch die Arbeit des Entwicklungs-Teams nach dem Plan, und seinem Erkenntnisgewinn über die erforderlichen Arbeiten zur Erreichung des Sprintziels.

Wenn weitere Arbeiten erforderlich sind, werden sie vom Entwicklungs-Team zum Sprint Backlog hinzugefügt. Wenn eine Arbeit durchgeführt oder abgeschlossen ist, wird die Schätzung der verbleibenden Arbeit aktualisiert. Wenn sich Bestandteile des Plans als unnötig erweisen, werden sie entfernt. Nur das Entwicklungs-Team kann sein Sprint Backlog während des Sprints ändern. Das Sprint Backlog ist ein hochgradig sichtbares Echtzeit-Bild der Arbeit, die das Entwicklungs-Team plant, in einem Sprint zu erreichen – und es gehört einzig und allein dem Entwicklungs-Team.

### **Fortschritt im Sprint überwachen**

Die gesamte verbleibende Arbeit an den Sprint Backlog Einträgen kann zu jedem Zeitpunkt im Sprint aufsummiert werden. Das Entwicklungs-Team verfolgt seine gesamte Restarbeit mindestens zu jedem Daily Scrum. Das Entwicklungs-Team überprüft diese Summen täglich, und bildet auf dieser Basis die Wahrscheinlichkeit der Erreichung des Sprintziels ab. Durch die Nachverfolgung der verbleibenden Arbeit während des Sprints kann das Entwicklungs-Team seinen Fortschritt steuern.

Scrum betrachtet nicht die Zeit, die an Sprint Backlog Einträgen gearbeitet worden ist. Die verbleibende Arbeit und das Datum sind die einzigen Variablen von Interesse.

### **Inkrement**

Das Inkrement ist die Summe aller im Sprint und aller vorherigen Sprints fertig gestellten Product Backlog Einträge. Am Ende eines Sprints muss das neue Inkrement „done“ sein, was bedeutet, dass es in einsatzfähigem Zustand sein und der „Definition of Done“ des Scrum Teams entsprechen muss. Es muss in einsatzfähigem Zustand sein, unabhängig davon ob der Product Owner entscheidet, es tatsächlich auszuliefern oder nicht.

### **Definition of “Done”**

Soll ein Product Backlog Eintrag oder ein Inkrement als „done“ („fertig“) bezeichnet werden, müssen alle das gleiche Verständnis darüber haben, was „done“ bedeutet. Obschon es hierbei signifikante Abweichungen zwischen verschiedenen Scrum Teams gibt, müssen die Mitglieder zur Gewährleistung der Transparenz ein gemeinsames Verständnis haben was es bedeutet, wenn eine Arbeit abgeschlossen ist. Dafür sorgt die „Definition of Done“ für das Scrum Team. Sie wird genutzt, um abzuschätzen, ob die Arbeit an dem Inkrement abgeschlossen ist.

Die gleiche Definition leitet das Entwicklungs-Team bei der Einschätzung, wie viele Product Backlog Einträge es in einem Sprint Planning Meeting auswählen kann. Der Sinn jedes Sprints

ist es, Inkremente potentiell auslieferbarer Funktionalität zu liefern, welche die aktuelle „Definition of Done“ des Scrum Teams erfüllen.

Entwicklungs-Teams liefern ein Inkrement an Produktfunktionalität pro Sprint. Dieses Inkrement ist einsetzbar, damit ein Product Owner sich dafür entscheiden kann, es sofort auszuliefern. Jedes Inkrement baut auf allen vorherigen Inkrementen auf; es ist gründlich getestet, um sicherzustellen, dass alle Inkremente zusammen funktionieren.

Mit zunehmendem Reifegrad eines Scrum Teams erwartet man, dass ihre „Definition of Done“ ausgeweitet wird, um stringendere Kriterien für höhere Qualität aufzunehmen.

## Fazit

Scrum ist kostenlos und wird in diesem Leitfaden angeboten. Die Rollen, Artefakte und Ereignisse von Scrum sind unantastbar. Obwohl es möglich ist, nur Teile von Scrum zu implementieren, ist das Ergebnis nicht Scrum. Scrum existiert nur in seiner Gesamtheit - und funktioniert gut als Container für andere Techniken, Methodologien und Praktiken.

## Danksagungen

### Menschen

Von den tausenden Menschen, die zu Scrum beigetragen haben, sollten wir diejenigen besonders hervorheben, die für Scrum in seinen ersten zehn Jahren von besonderer Bedeutung waren. Am Anfang standen Jeff Sutherland, in seiner Arbeit mit Jeff McKenna, und Ken Schwaber in seiner Arbeit mit Mike Smith und Chris Martin. Viele weitere haben in den Jahren seitdem ihren Beitrag geleistet; und ohne ihre Hilfe wäre Scrum nicht so ausgefeilt wie es das heute ist. David Starr trug mit grundlegenden Einsichten und redaktionellen Fähigkeiten zur Formulierung dieser Version des Scrum Guides bei.

### Geschichte

Ken Schwaber und Jeff Sutherland präsentierten Scrum gemeinsam zum ersten Mal auf der OOPSLA-Konferenz 1995. Diese Präsentation dokumentierte im Kern die Erkenntnisse, die Ken und Jeff während der letzten paar Jahre bei der Anwendung von Scrum gesammelt hatten.

Die Geschichte von Scrum wird bereits als lang angesehen. Um die ersten Stellen zu würdigen, wo es ausprobiert und verfeinert wurde, erwähnen wir hier Individual, Inc., Fidelity Investments und IDS (heute GE Medical).

### Übersetzung

Dieser Guide wurde von der englischen Originalversion, bereitgestellt von Ken Schwaber und Jeff Sutherland, übersetzt. Beigetragen zur Übersetzung haben:

Dominik Maximini, Andreas Schliep, Ulf Schneider, Wolfgang Wiedenroth

## Revisionen

Dieser Scrum Guide vom Juli 2011 unterscheidet sich von seinem Vorgänger, dem Scrum Guide vom Februar 2010. Wir haben insbesondere versucht, Techniken oder Best Practices aus dem Kern von Scrum zu entfernen. Solche werden je nach den Umständen variieren. Wir werden später mit einem „Best Practices“ Kompendium beginnen, um einige unserer Erfahrungen anzubieten.

Dieser Scrum Guide dokumentiert Scrum, wie es seit mehr als zwanzig Jahren von Jeff Sutherland und Ken Schwaber entwickelt und fortgeführt wurde. Andere Quellen bieten Ihnen Muster, Prozesse und Einsichten darüber, wie Praktiken, Kniffe („facilitations“) und Werkzeuge das Scrum Rahmenwerk ergänzen. Diese optimieren Produktivität, Wert, Kreativität und Stolz.

Release Notes über die folgenden Unterschiede zwischen dieser und der Version vom Februar 2010 werden an anderer Stelle veröffentlicht, inklusive Diskussionen über:

1. Release Planning
2. Release Burndown
3. Sprint Backlog
4. Product und Sprint Backlog Burndown
5. Commitment ist nun Vorhersage („forecast“)
6. Team (hin zu Entwicklungs-Team)
7. Pigs and Chickens ... die Sage von Scrum
8. Geordnet statt priorisiert