

# Damit es ein Erfolg wird

## Risikomanagement in agilen Projekten

von Dieter Ehart, Florian von Reitzenstein und Thorsten von Thaden



Ein professionelles Risikomanagement im Projekt erlaubt, frühzeitig auf Abweichungen zu reagieren, und ermöglicht somit ein effizientes Projektmanagement. Gerade der Einsatz agiler Methoden, wie beispielsweise Scrum, bietet verschiedene Vorteile, die im vorliegenden Artikel dargestellt werden. Aus diesen ergibt sich, dass der Fokus des Projektleiters im Risikomanagement auf den externen Umfeldfaktoren liegen kann.

### Initiales Risikomanagement

Als Vorgehensmodell macht Scrum keine Aussagen dazu, wie ein Projekt aufgesetzt werden soll. Es konzentriert sich stattdessen auf die Steuerungsphase des Projektmanagements nach DIN 69901. Entsprechend existieren in der Scrum-Community unterschiedliche Ansätze für das Risikomanagement in einem Projekt. Häufig wird hierbei allerdings übersehen, dass es bereits in der Planungsphase des Projektes sinnvoll ist, eine initiale Risikoanalyse durchzuführen. Denn ein Risikoworkshop

mit dem gesamten Team noch vor Projektstart ermöglicht einen gemeinsamen Blick auf die Risiken und vermittelt dadurch ein besseres, gemeinsames Projektverständnis. Zum Einsatz kommen dabei die bekannten Prozesse und Methoden:

- > **Identifizieren:** Aufdecken und Qualifizieren aller Risiken in einem Risikoworkshop, Input aus der Umfeldanalyse, Vergleich mit anderen Projekten, Kreativitätstechniken.
- > **Analysieren:** Untersuchen der Risiken hinsichtlich deren Ursachen und Auslösern und Quantifizieren der Eintrittswahrscheinlichkeit und der Schadenshöhe. Berechnen des Risikowertes und Erstellung des Risikoportfolios.
- > **Planen:** Festlegen von korrektiven oder präventiven Maßnahmen für jedes Risiko.
- > **Steuern:** Risiken regelmäßig auf Aktualität und die ergriffenen Maßnahmen auf Wirksamkeit prüfen. Reagieren bei Eintreten von Risiken.

Die Umfeldanalyse gliedert das Umfeld des Projektes in interne und externe Faktoren oder in direkte und indirekte Faktoren in Abhängigkeit ihrer jeweiligen Nähe zu dem beziehungsweise ihrer Wirkung auf das Projekt. Aus den jeweiligen Faktoren können sich Risiken für das Projekt ergeben. Typische Beispiele für interne Umfeldfaktoren sind:

- > das Projektteam,
- > direkte Stakeholder (Auftraggeber, Lenkungsreis, Anwender, Betriebsrat),
- > Projektträume oder
- > Projektorganisation.

Beispiele für externe Umfeldfaktoren sind:

- > Gesetze und Normen,
- > technische Restriktionen,
- > gesellschaftliche und politische Faktoren,
- > die wirtschaftliche Entwicklung oder der Wettbewerb.

Den sich aus dem – für jedes Projekt unterschiedlichen – Projektumfeld ergebenden Risiken muss auch in agilen Projekten begegnet werden. Daher ist es wichtig, sich bereits zum Projektstart über die Risikoexposition des Projektes Klarheit zu verschaffen.

### Laufendes Risikomanagement

Beim Einsatz agiler Vorgehensmodelle erlebt man häufig eine Konzentration des Teams auf die Steuerungsphase des Projektes. Dies führt dazu, dass es in der Praxis häufig Scrum-Projekte gibt, die kein dediziertes Risikomanagement betreiben. Sei es aus dem falschen Verständnis heraus, dass dies mit Scrum nicht notwendig sei, oder einfach, weil Scrum nicht als Vorgehensmodell, sondern als Projektmanagementmethode verstanden wird.

Oft sind es nicht die dem Management berichteten und transparenten Risiken, die den Projekterfolg gefährden, sondern

zunächst verdeckte und sich gegenseitig verstärkende Risiken. Scrum, genauso wie andere agile Vorgehensmodelle auch, setzt auf grundlegenden Prinzipien und Methoden auf:

- > Agiles Manifest
- > Scrum-Ereignisse
- > Transparenz
- > Impediment Backlog
- > User Stories
- > Schätzungen im Product Backlog
- > Verantwortung
- > Inspect and adapt
- > Potentially shippable increment

Diese unterstützen dabei, genau diese verdeckten Risiken aufzudecken und geeignete Gegenmaßnahmen zu ergreifen.

### Agiles Manifest

Das agile Manifest fordert dazu auf, Veränderungen zu begrüßen und mit ihnen umzugehen. Außerdem, den Kunden aktiv einzubinden und wertvolle Software zu schaffen, die möglichst früh geliefert werden kann. Erfolg wird durch die Zufriedenheit des Kunden definiert. Das Team soll den Fokus auf Einfachheit legen und die Kunst „the amount of work not done“ maximieren. Die tägliche Projektarbeit zeigt: Je weiter in die Zukunft geplant wird, desto größer ist die Wahrscheinlichkeit, dass ein Ereignis eintritt, mit dem in der Planungsphase nicht gerechnet wurde. In Scrum wird deshalb in der Planungsphase der nächste Sprint fokussiert. So kann das Projekt flexibler auf geänderte Bedingungen seines Umfelds reagieren. Risiken, die während eines Sprints erkannt werden, können schon bei der Planung des Folgesprints berücksichtigt werden. Das gemeinsame agile Wertesystem fordert alle Projektbeteiligten dazu auf, Herausforderungen und Hindernisse aktiv anzugehen, Risiken direkt aus dem Weg zu räumen und den Erfolg des Projektes nachhaltig zu sichern.

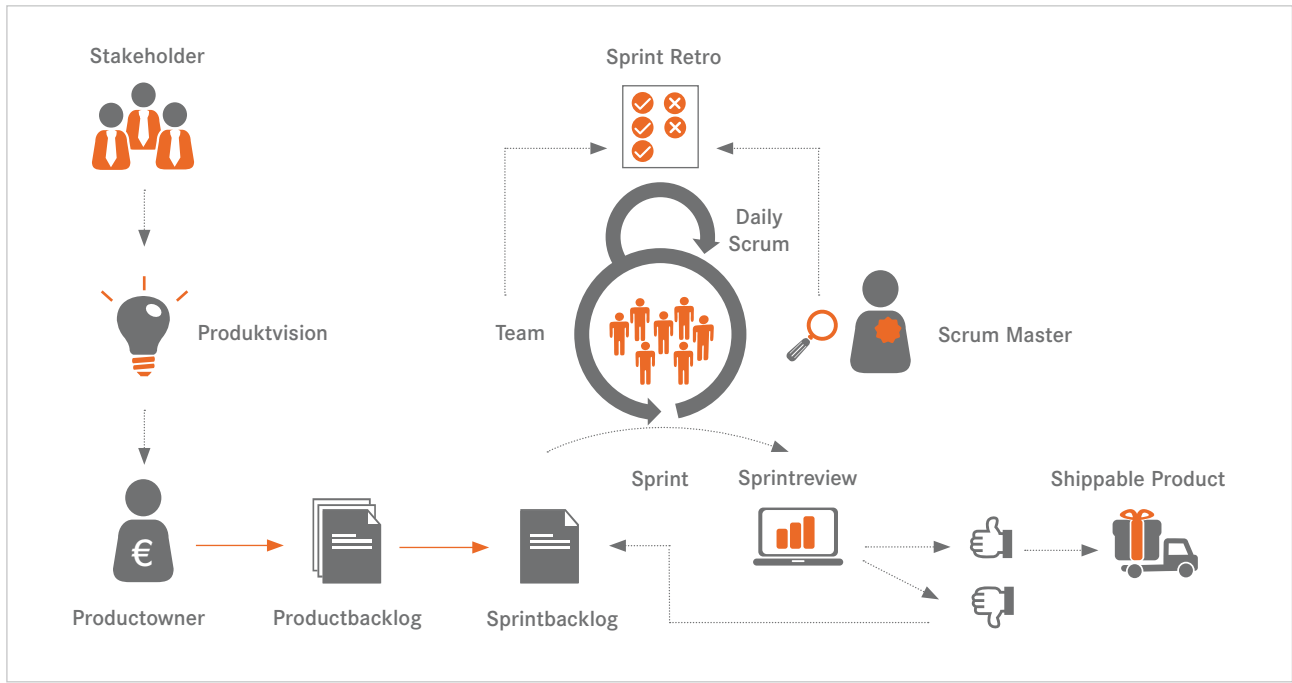


Abbildung 1: Scrum-Ereignisse

## Scrum-Ereignisse

Scrum bietet dem Projektleiter, dem Product Owner und dem Team vier wesentliche Eingriffspunkte für das Risikomanagement: das Sprint Planning, das Daily Scrum, den Sprint Review und die Sprint-Retrospektive.

Im **Sprint Planning** werden die beiden Fragen „Was wird im nächsten Sprint getan?“ und „Wie erreichen wir das Ziel?“ beantwortet. Im Rahmen der Planerstellung für den nächsten Sprint werden hier Ressourcenverfügbarkeiten berücksichtigt und ein gemeinsames Verständnis des zu erreichenden Ziels erarbeitet. Risiken werden durch das Team direkt im Forecast für den nächsten Sprint berücksichtigt. Durch diese Selbstorganisation im Team erhöht sich das Verantwortungsbewusstsein und damit auch die gelieferte Qualität.

Das **Daily Scrum** bietet dem Team die Möglichkeit, den Fortschritt auf einer täglichen Basis abzuschätzen und zu beobach-

ten, um frühzeitig steuernd einzugreifen. Das Team synchronisiert hier seine Tätigkeiten für die nächsten 24 Stunden und gibt eine Prognose ab, welche Arbeiten bis zum nächsten Daily Scrum erledigt sein werden. Risiken können so während der Entwicklung erkannt und entweder direkt im Team oder in Zusammenarbeit mit dem Product Owner und dem Scrum Master gelöst werden.

Der **Sprint Review** gibt den Projektstakeholdern die Möglichkeit, die geleistete Arbeit zu sichten und hinsichtlich ihres Nutzens zu bewerten. Dies erlaubt es, frühzeitig festzustellen, ob das Projekt auf dem richtigen Kurs ist. Dabei gilt: Je größer die Begeisterung für das Produkt ist, desto größer wird auch die Motivation des Entwicklungsteams. Umgekehrt zeigt eine negative Kundenreaktion, dass das Projekt auf dem falschen Weg ist, und versetzt den Projektleiter und den Product Owner in die Lage, korrigierend einzugreifen. Im Extremfall kann das Projekt frühzeitig abgebrochen werden, beispielsweise wenn erkannt wird, dass der Business Value nicht mehr erbracht werden kann.

Insgesamt bringt der Review durch den Austausch von Entwicklern und Stakeholdern ein gesteigertes gegenseitiges Verständnis, indem alle Diskussionen auf dem erlebbaren Produkt aufsetzen. Dadurch können neue Erkenntnisse darüber gewonnen werden, wie das Produkt weiter verbessert werden kann.

In der **Sprint-Retrospektive** rekapituliert das Team den letzten Sprint und identifiziert Verbesserungspotenziale. Dazu werden die Umsetzung und Wirksamkeit der Maßnahmen aus dem letzten Meeting analysiert und neue Maßnahmen definiert. Ziel ist es, den Entwicklungsprozess und die Teamzusammenarbeit zu verbessern und effektiver zu gestalten. Im Rahmen dieser Meetings können aufgetretene Risiken des letzten Sprints analysiert und Lösungen für den zukünftigen Umgang mit diesen erarbeitet werden.

Insgesamt zeichnen sich zwei Ansatzpunkte für das Risikomanagement durch die Scrum-Ereignisse ab: Einerseits wird das Team stärker in die Verantwortung genommen und aktiv in das Risikomanagement eingebunden. Andererseits bieten sich durch die definierten Abläufe vielfältige Möglichkeiten sowohl zur Identifikation von Risiken als auch zur Definition von Maßnahmen und damit zur Risikosteuerung.

## Transparenz

Durch die kurzen Sprintzyklen und die Sprint Reviews wird der aktuelle Projektstand regelmäßig dem Team und den Stakeholdern transparent gemacht. Dies verhindert Missverständnisse im Reporting und führt dazu, dass Risiken sowohl durch das Team als auch durch die Stakeholder bereits frühzeitig identifiziert werden und somit frühzeitig Maßnahmen zur Steuerung ergriffen werden können. Dies führt zu dem, dass Risiken früher erkannt werden, und ermöglicht zum anderen, ein Chancenmanagement zu etablieren.

Die Entscheidung, welche Stakeholder aktiv am Review teilnehmen, wird im Zuge der Stakeholderanalyse getroffen.

## Impediment Backlog

Das Impediment Backlog zeigt alle vom Team identifizierten Hindernisse mit ihrem aktuellen Status. Es liefert einen Überblick, welche Risiken im Projektverlauf in der Praxis tatsächlich auftreten und wie man ihnen begegnen kann. Diese Erkenntnisse können in andere Projekte einfließen und ermöglichen es, das Risikomanagement stetig weiter zu verbessern.

## User Stories

User Stories dienen zwei großen Zielen: Einerseits wird die Komplexität in beherrschbare Teile aufgeteilt, andererseits zeigen sie immer auch den Nutzen für einen Personenkreis auf: Damit <Nutzen> möchte ich als <Rolle> folgende <Funktion>. Dadurch wird verhindert, dass Funktionen implementiert werden, deren Nutzen nicht nachgewiesen ist. Die Entwickler verstehen so, zu welchem Zweck die Funktion verwendet werden soll, und werden auf diese Weise dazu befähigt, aktiv Verbesserungen einzubringen. Das Format der User Stories fördert die Diskussion zwischen allen Beteiligten auf einer nutzenbasierten Ebene. Das Ziel der User Stories kann leichter nachvollzogen werden, um so gemeinsam zu einer bestmöglichen Lösung zu kommen. Durch die Verwendung von User Stories zur Anforderungserfassung wird aktiv das Risiko von Missverständnissen minimiert, sodass der Kunde das bekommt, was er benötigt, auch wenn er es nicht detailliert spezifiziert hat.

## Schätzungen und das Product Backlog

Alle Einträge im Product Backlog müssen geschätzt sein. Diese Forderung führt dazu, dass zu große oder komplexe Einträge identifiziert und weiter detailliert werden können. Auch wird während des Schätzens erkannt, ob alle Beteiligten das gleiche Verständnis der Aufgabe haben. User Stories, die zu unspezifisch oder zu groß sind, werden so frühzeitig erkannt und müssen vom Product Owner in Zusammenarbeit mit dem Team verkleinert werden. Hierdurch werden Risiken, die sich aus mangelnder Detaillierung oder fehlendem Verständnis ergeben, angegangen und entschärft.

Darüber hinaus ist das Product Backlog nach Wertbeitrag sortiert, sodass immer als Nächstes diejenigen Anforderungen umgesetzt werden, die dem Kunden den größten (Zusatz-)Nutzen bringen. Dies verringert das Risiko, dass gegen Projektende wesentliche Funktionen aufgrund fehlender Zeit oder Ressourcen nicht mehr umgesetzt werden können.

### Verantwortung

In Scrum übernimmt der Product Owner die Verantwortung für den wirtschaftlichen Erfolg des Produktes, der Scrum Master verantwortet die optimale Anwendung der Prozesse, und das Team ist für die technische Qualität verantwortlich. Durch diese Verantwortungsteilung und die damit verbundene Identifikation mit dem Produkt wird erreicht, dass ein neues Risikobewusstsein entsteht. Das führt dazu, dass sich alle Beteiligten aktiv an der Risikoidentifikation, der Risikoanalyse und der Planung und Kontrolle von Maßnahmen beteiligen. Die Übertragung von Verantwortung auf das Team steigert seine Motivation. Dies wiederum fördert die Teambildung und mindert somit die damit verbundenen Risiken.

### Inspect and adapt

Durch laufendes Überprüfen und Anpassen der Zusammenarbeit, der Scrum-Artefakte und des Fortschritts auf dem Weg zum gemeinsamen Ziel werden Abweichungen frühzeitig aufgedeckt. Ken Schwaber und Jeff Sutherland – die Entwickler von Scrum – empfehlen, die Überprüfung regelmäßig vor Ort durchzuführen. In der Praxis sollte das gesamte Scrum-Team den Fortschritt überprüfen, etwa im Daily Scrum. Werden Abweichungen erkannt und liegen diese außerhalb der tolerierbaren Grenzen müssen, geeignete Maßnahmen ergriffen werden, um den Arbeitsgegenstand oder den Prozess wieder zu korrigieren. Sonst besteht die Gefahr, dass das Produkt am Ende des Sprints ebenfalls außerhalb der tolerierbaren Grenzen liegt und damit das gemeinsam vereinbarte Sprintziel verfehlt wird.

### Potentially shippable increment

In Projekten mit klassischen Vorgehensmodellen liegt üblicherweise eine zu Anfang geringere Steigung der Wertschöpfungskurve vor, die im weiteren Verlauf des Projektes zunimmt. Durch das Credo „Maximize the amount of work not done“ wird bei agilen Projekten das Risiko, Arbeit zu bezahlen, die nicht benötigt wird, deutlich reduziert. Auch deshalb wird typischerweise von Beginn an eine stärker steigende Wertschöpfungskurve erreicht (vgl. Abbildung 2).

Dieser Faktor ist insbesondere bei Neuentwicklungen und Forschungsprojekten nicht zu unterschätzen: In klassischen Vorgehensmodellen wird zunächst eine detaillierte Dokumentation der zu erstellenden Software erstellt, ohne dass deren Beitrag zur Wertschöpfung gemessen werden kann: Solange kein Produkt erstellt ist, solange lässt sich auch der Wert der Dokumentation inhaltlich nicht bewerten. In Scrum hingegen wird von Anfang an Software produziert, die auf dem höchsten Kundennutzen basiert. Damit ergeben sich sowohl eine niedrigere Time-to-Market als auch die Möglichkeit, das Produkt schneller an den Erfordernissen des Marktes auszurichten.

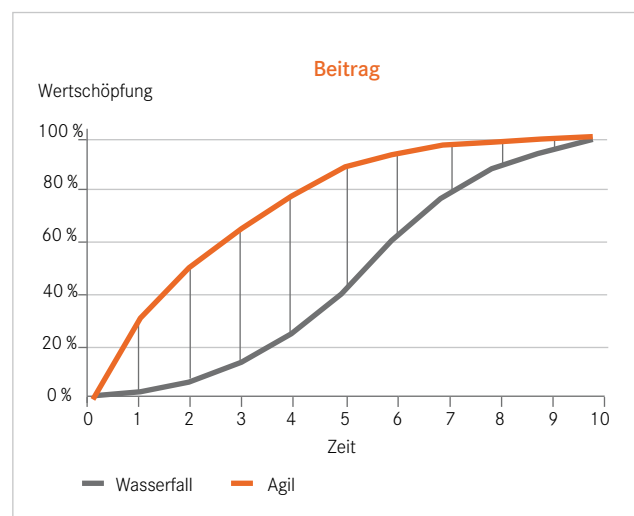


Abbildung 2: Vergleich typischer Wertbeiträge in klassischen und agilen Vorgehen



Abbildung 3: Verteilung der dargestellten Prinzipien und Methoden auf die Umfeldfaktoren

## Fazit

Auch wenn Scrum keineswegs ein professionelles Risikomanagement ersetzt, bietet es doch Mittel und Wege, offener mit Risiken umzugehen. Außerdem können Chancen im Projektverlauf leichter erkannt und genutzt werden. Für alle Projektbeteiligten bedeutet Scrum ein Umdenken hin zu einem Risikomanagement, in das alle Beteiligten aktiv eingebunden sind: Jedes Teammitglied ist selbstständig dafür verantwortlich, Risiken zu adressieren und aktiv an deren Beseitigung mitzuwirken. Der Rahmen, den Scrum durch die Sprints bietet, ist ideal dafür geeignet, ein laufendes Risikomanagement zu betreiben und für Transparenz auf allen Seiten zu sorgen. Die dargestellten Ansätze, die sich aus dem agilen Vorgehen ergeben, beziehen sich im Wesentlichen auf die internen Umfeldfaktoren des Projektes, wie etwa Team, direkte Stakeholder, Projekträume und Projektorganisation. Für die externen Umfeldfaktoren bietet Scrum keine expliziten Methoden oder Vorgehensweisen an (vgl. Abbildung 3).

Der Fokus des Projektleiters muss folglich im Risikomanagement und in der Projektsteuerung auf den aus diesen externen Umfeldfaktoren resultierenden Risiken liegen. Das Management der internen Umfeldfaktoren und der damit einhergehenden Risiken wird durch Scrum bestmöglich unterstützt. Häufig wird der Projektleiter im agilen Umfeld mit diesen Risiken gar nicht erst in Berührung kommen.

Der Artikel zeigt, dass durch den Einsatz agiler Vorgehensweisen frühzeitig Fehlentwicklungen erkannt werden können und möglichst schnell ein Produkt erstellt wird, das ausgeliefert und verwendet werden kann. Somit wird eine frühere Wertschöpfung erreicht und das mit der Projektumsetzung einhergehende finanzielle Risiko vermindert.

### Autor



#### Dieter Ebhart

Leiter CoC Projektmanagement

> +49 (0) 7252 / 9350 - 286

> dieter.ebhart@msg-gillardon.de