



# Agiles Requirements Engineering – Klassische und agile Methoden im Anforderungsmanagement erfolgreich vereinbaren

Lars Möller, Pentasys

*Agile Methoden setzen sich nach und nach in Unternehmen durch. Dennoch findet man auch heute noch Prozesse, die sich nach der Wasserfall-Methode richten. Dies ist unter anderem dadurch zu verstehen, dass agile Methoden einerseits Ungewissheit in Bezug auf das Endprodukt mit sich bringen, andererseits kann und darf aufgrund regulatorischer sowie geschäftlicher Anforderungen nicht auf feste Meilensteine verzichtet werden.*

In diesem Dilemma entscheidet sich das Management häufig für die „konservative“ Variante: Nämlich weiterhin auf die klassischen Methoden des Anforderungsmanagements zu setzen. Dabei dominiert die Sicht auf Agilität als Zusatzrisiko, denn iterative Änderungen und noch nicht feststehende Ergebnisse lassen sich in Modellen schlecht darstellen.

Wird ein solches Vorgehen gewählt, bremst der klassische Ansatz jedoch die volle Anwendung der agilen Vorgehensweise aus – und damit auch ihre Vorteile. Aus dieser Herausforderung entsteht daher die Notwendigkeit, klassisches Requirements Engineering und agiles Projektmanagement in Einklang zu bringen. Im Folgenden erfahren Sie, wie dies gelingen kann.

## Wo klassisches Requirements Engineering und Agilität auseinandergehen

Das Anforderungsmanagement gehört zu den elementarsten Bereichen der Softwareentwicklung und ist dementsprechend stark standardisiert. Dies findet sich beispielsweise in den ISO-Normen ISO/IEC 12207 und ISO/IEC 15504 wieder, sowie in der Anwendung formalisierter Methoden zur Anforderungsspezifikation, beispielsweise Lastenhefte oder Templates zur Beschreibung der Anforderungen. Daneben haben sich eigene Dateiformate etabliert, wie etwa das ReqIF (Requirements Interchange Format) zum Austausch von Daten zwischen verschiedenen Softwarewerkzeugen.

Auch heute noch werden diese Methoden in einem Prozess implementiert, der an das Wasserfall-Modell erinnert. Dieser schreibt vor, dass Auftraggeber und Auftragnehmer die Anforderungen an das Projekt „Up-Front“, also bereits vor Projektbeginn, definieren müssen. Über den Projektverlauf hinweg werden sie dann Phase für Phase abgearbeitet. Dass dieses Vorgehen nach wie vor zum Einsatz kommt, ist insofern verständlich, als hiermit eine sehr granulare und präzise Festlegung des Projektablaufs und seiner Resultate im Vorab möglich ist. Insbesondere aus der geschäftlichen Perspektive kann die klassische Methode attraktiv erscheinen, weil sie die Allokation von Personal und Ressourcen zunächst vereinfacht und Unwägbarkeiten definitiv ausschließt.

Das agile Modell hingegen ist darauf ausgelegt, die Anforderungen kontinuierlich zu prüfen und gegebenenfalls anzupassen. Dies betrifft auch die Validierung der zugrunde liegenden Annahmen der definierten Anforderungen, zum Beispiel bestimmte Vorlieben der Endnutzer. Anforderungsmanagement findet daher laufend statt, von Beginn bis Ende des Projekts. Hieraus ergibt sich allerdings eine gewisse Offenheit in der Entwicklungsrichtung und bei den zu erwartenden Resultaten. Dies steht der Vorstellung von Requirements Engineering als a priori definiertem „Masterplan“ eindeutig entgegen!

## Auch agile Projekte brauchen Requirements Engineering

Wie lassen sich die beiden unterschiedlichen Vorgehensweisen also auf einen gemeinsamen Nenner bringen? Diese Frage sollte nicht unbeantwortet bleiben. Denn einerseits würde eine Rückkehr zum Projektmanagement nach der Wasserfall-Methode auch deren klare Nachteile gegenüber agilen Methoden zum Vorschein bringen: Eine verzögerte Time-to-Market, mehr Unsicherheit bei der Auslieferung des Endprodukts sowie ein mögliches Vorbei-Entwickeln an den

Nutzeranforderungen wären in einem solchen Szenario der „Preis“ für eine umfassendere und transparentere Up-Front-Planung. Die Nachteile überwiegen dabei allerdings die Vorteile. Grundsätzlich gilt daher, dass moderne IT-Entwicklung stets agile Methoden zugrunde legen sollte.

Andererseits wären agile Projekte ohne ein dediziertes Requirements Engineering unvollständig. Gerade in komplexen Projekten mit einem hohen Investitionsaufwand stellt das Requirements Engineering Planungssicherheit und eine Grundlage für das Risikomanagement her. In bestimmten Fällen ist es sogar regulatorische Pflicht. Zu guter Letzt braucht es das Requirements Engineering auch für ein gemeinsames Verständnis unter den beteiligten Stakeholdern. Insbesondere, wenn externe Dienstleister am Projekt beteiligt sind, spielt es eine wichtige Rolle bei der vertraglichen Vereinbarung mit dem Auftraggeber.

Damit wird die „Mission“ des agilen Requirements Engineering deutlich: Agiles Projektmanagement in Kombination mit Anforderungsmanagement ermöglichen und dabei die Vorteile beider zur Geltung bringen. Dies kann realistisch gesehen nur funktionieren, indem das agile Denken im Requirements Engineering den Vorrang erhält – wenn also das handwerkliche Rüstzeug des Anforderungsmanagements zwar weitgehend gleichbleibt, die dahinterliegende Philosophie sich aber wandelt.

## Das Mindset nach den Prinzipien des Agilen Manifests

Um den (scheinbaren) Widerspruch zwischen Requirements Engineering und agilen Methoden aufzulösen, ist es zunächst die Überlegung wert, auf die Grundprinzipien der Agilität zurückzukommen – wie sie im allseits bekannten Agilen Manifest niedergelegt sind. Hieraus lässt sich ein spezifisch agiles Verständnis für Requirements Engineering entwickeln, aus dem sich in einem zweiten Schritt dann konkrete Tools und Prozesse für die Praxis ableiten. Was Agilität für das Requirements Engineering bedeuten kann, soll anhand der folgenden vier Punkte aus dem Manifest für Agile Softwareentwicklung dargelegt werden:

- 1. Individuen und Interaktionen mehr als Prozesse und Werkzeuge:**  
Auch im Anforderungsmanagement muss das Verständnis für die Bedürfnisse und Wünsche der Anwender und Stakeholder an erster Stelle stehen. Dies macht die zahlreichen definierten Prozesse und Werkzeuge keineswegs obsolet – denn sie spielen eine wichtige Rolle als unterstützende Elemente, um das genannte Verständnis zu erreichen.
- 2. Funktionierende Software mehr als umfassende Dokumentation:**  
Das inhaltliche Augenmerk beim Anforderungsmanagement sollte stets auf der Funktionsfähigkeit der Software liegen. Wichtig dabei ist, diese aus der Anwendersicht zu verstehen. Denn nur, was beim User auch ankommt, funktioniert tatsächlich. Das Requirements Engineering erfüllt in diesem Zusammenhang die Aufgabe, verteiltes Wissen zusammenzuführen und im Projektablauf zu konsolidieren. Dies entbindet allerdings nie von der Pflicht, eine korrekte Dokumentation zu führen, die alle regulatorischen Anforderungen erfüllt. Über die formalen Anforderungen der Dokumentation sollte jedoch die Funktionsfähigkeit der Software nie vergessen werden.

### 3. Zusammenarbeit mit dem Kunden mehr als Vertragsverhandlung:

Agile Methoden leben vom Informationsaustausch sowie vom gegenseitigen Verständnis der Stakeholder. Dies gilt ganz besonders für das Requirements Engineering, das bereits vor dem Projektstart beginnt. In dieser Phase sind die Informationsbasis und das gegenseitige Verständnis noch am geringsten und müssen erst hergestellt werden. Sehr hilfreich dabei ist beispielsweise die Definierung von User Stories, entlang derer das Produkt über den Projektzeitraum hinweg erarbeitet wird.

### 4. Reagieren auf Veränderung mehr als das Befolgen eines Plans:

Agilität bringt es fast zwangsläufig mit sich, dass im Projektverlauf Änderungen auftreten. Erfreulicherweise stellt bereits das klassische Requirements Engineering Methoden und Techniken bereit, um auf solche zu reagieren. Diese können den agilen Prozess sehr gut unterstützen – sofern sie auch in einem agilen Mindset angewendet werden. Um Änderungen schnell und effizient durchführen zu können, ist die Verpflichtung zur Einfachheit in allen Belangen ein wichtiges Gebot.

Diese Betrachtungen mögen auf den ersten Blick sehr allgemein erscheinen – doch zunächst geht es genau darum. Die Anwendungen von Tools und Methoden sollten sich nämlich stets nach dem (agilen) Mindset und seiner Philosophie richten – und nicht umgekehrt. Dies gilt sowohl für das agile Projektmanagement als auch für das agile Anforderungsmanagement. Wer diesen Weg geht, wird schnell herausfinden, dass viele der Prozesse und Werkzeuge, die aus der klassischen Vorgehensweise bekannt sind, auch weiterhin sinnvoll sind. Sie müssen „lediglich“ auf agile Weise Anwendung finden!

## Fallstricke des agilen Requirements Engineering und wie sie sich lösen lassen

Von der agilen Transformation zu sprechen und sie umzusetzen, sind zweierlei Dinge. Denn in der Praxis lauern einige Herausforderungen, die im reinen Modell nicht abgebildet werden beziehungsweise überhaupt abgebildet werden können. Das klassische Requirements Engineering muss sich dementsprechend adaptieren. Denn schließlich war es in der Vergangenheit des Wasserfall-Modells noch möglich, modellhafte Anforderungen zu Beginn präzise vorzuschreiben, die durch das ganze Projekt hinweg strikt durchexerziert werden. Ob die getroffenen Annahmen tatsächlich zutreffen, wurde in einer sehr späten Phase oder gar erst am Ende des Projekts deutlich. Im agilen Modus hingegen gehen Modell und Empirie stets Hand in Hand.

Ein sehr wichtiger Praxisaspekt ist dabei unter anderem die Implementierung der definierten Anforderungen im laufenden Projekt. Werden nämlich Anforderungen vor Projektstart zwar entwickelt und festgeschrieben, dann jedoch überhaupt nicht beachtet, ist die Sinnhaftigkeit des gesamten Prozesses infrage zu stellen. Die Problemstellung erschwert sich meist dadurch, dass im Verlauf des Projekts dessen ursprüngliches Ziel aus den Augen gerät, da sich die Richtung im Laufe weniger Sprints stark verändern kann. Die Herausforderung besteht also darin, einerseits flexibel auf Änderungen zu reagieren, aber dennoch die definierte Zielsetzung des Projekts kontinuierlich weiterzuverfolgen.

Ein weiterer Aspekt, den es in der Praxis zu berücksichtigen gilt, ist der organisatorische Kontext, in dem das agile Projekt stattfindet. Überlastung der Stakeholder mit Informationen, übermäßige Kom-

plexität sowie kulturelle Hürden, etwa zwischen verschiedenen Abteilungen, sind dabei zu beachten und in das agile Anforderungsmanagement mit einzubeziehen. Ein häufiges Beispiel dafür ist die Zuordnung von Themen an eine Rolle, die in klassischen Organisationsstrukturen in der Regel an eine Hierarchie gekoppelt ist. Im agilen Denken spielt Hierarchie allerdings eine untergeordnete Rolle, wodurch ein Anpassungsbedarf entsteht.

Die genannten Hürden lassen sich nur durch einen Lern- und Reflexionsprozess überwinden, der alle Beteiligten miteinschließt. Das Gelingen dieses Prozesses ist glücklicherweise nicht dem Zufall überlassen, sondern lässt sich mithilfe verschiedener Werkzeuge unterstützen. Drei der wichtigsten werden hier kurz vorgestellt:

#### ■ Design Thinking:

Hierbei handelt es sich um eine Methode, die durch einen iterativen Prozess zu innovativen Produkten und Lösungen führen soll. Das Design Thinking ist vor allem für die Klärung von Fragestellungen und die Definierung von Zielen aus Sicht des Anwenders geeignet. Es sollte vor allem in einer sehr frühen Phase eines Projekts zur Anwendung kommen, in der die übergeordnete Zielsetzung noch einer Klärung bedarf.

#### ■ Lean Management:

Dieser Oberbegriff versammelt ein Set an wichtigen Prinzipien und Methoden zum Aufbau effizienter beziehungsweise „schlanker“ Organisationsformen. Ein wichtiger Aspekt ist auch hier die Kunden- und Anwenderorientierung. So sind eine Organisation oder auch deren Prozesse stets dahingehend zu befragen, ob sie aus Sicht des Kunden tatsächlich notwendig sind. Für das Thema Requirements Engineering ist dies vor allem von Bedeutung, wenn es um die richtige Zuweisung von Befugnissen und Ressourcen innerhalb einer bestehenden Organisation geht.

#### ■ Minimum Viable Product (MVP):

Dieses Vorgehen leitet sich aus den Prinzipien des Lean Management ab. Ein MVP ist ein Erzeugnis, das in einer Minimalkonfiguration zumindest eine Anforderung des Users abdeckt. Entwicklungsteams können mithilfe von MVPs Produkte und Lösungen bereits nach kurzer Entwicklungszeit einführen und testen, wodurch sich mögliche Fehlentwicklungen früh korrigieren lassen. Das MVP lässt sich im Rahmen des Anforderungsmanagements zum Beispiel als Meilenstein heranziehen.

Wichtig zu betonen ist, dass die genannten Hilfsmittel nicht nur einmalig zum Einsatz kommen müssen. Sie lassen sich auch zu jedem beliebigen Zeitpunkt des Projektablaufs wiederholen, um die initial aufgestellten Annahmen zu überprüfen und gegebenenfalls zu korrigieren. In der Praxis geht es am Ende vor allem darum, eng entlang der Nutzeranforderungen zu arbeiten. Diese findet man am besten durch kontinuierliches „Testen und Lernen“ heraus.

## Requirements Engineering lässt sich in Scrum einbetten

Im Vergleich zum Requirements Engineering sind die Werkzeuge und konkreten Abläufe in Scrum weniger formalisiert und standardisiert. Einen wichtigen Ausgangspunkt für Requirements Engineering in Scrum ist das Backlog, über das üblicherweise der Product Owner wacht. Hier sind auch die Anforderungen aus Nutzersicht

hinterlegt. Sie lassen sich anhand der Begriffe „Epic“ und „User Story“ definieren: Während die User Story einen einzelnen Schritt beschreibt, den der Anwender vollbringen muss, um das nächste (Teil-)Ziel zu erreichen, fassen Epics mehrere User Stories in einem Prozess zusammen, der zu einem übergeordneten Ziel führt.

Aus der Verknüpfung der Epics und User Stories entsteht schließlich die sogenannte „Story Map“, in der alle Elemente visuell verknüpft werden. Die Story Map eignet sich besonders gut, um einen schnellen Überblick über alle relevanten Anforderungen zu vermitteln und damit den Wissensaustausch unter den Stakeholdern zu vereinfachen. Der Product Owner kann alle Anforderungen bündeln, indem er Änderungen in der Story Map mit einem neuen Eintrag im Backlog dokumentiert. Damit lassen sich der aktuelle Stand der Anforderungen wie auch alle bisherigen Schritte jederzeit nachvollziehen.

Für die Beschreibung der einzelnen User Stories im Backlog können dann wiederum die bekannten Standards aus dem klassischen Requirements Engineering zum Einsatz kommen, beispielsweise vorformulierte Sätze oder auch Dateivorlagen. Was aus dem klassischen Handwerk sinnvoll ist oder auch nicht, ist je nach Projektkontext zu entscheiden. Wichtig ist, dass jede User Story den sogenannten „INVEST“-Kriterien entspricht: Independent (unabhängig), Negotiable (verhandelbar), Valueable (wertorientiert), Estimatable (abschätzbar), Small (klein), Testable (testbar).

Wie bereits erwähnt, kommt agiles Requirements Engineering über den gesamten Lebenszyklus eines Projekts hinweg zum Einsatz. Eine besonders wichtige Rolle spielt es jedoch in der Planungsphase

vor dem eigentlichen Projektbeginn. Hier entscheiden sich nämlich grundlegende Eckpfeiler wie Projektvision, Transparenz und Teamaufbau. Dabei ist ein besonderes Augenmerk auf das Thema Anforderungen und die entsprechenden User Stories zu legen, um eine adäquate Bemessung der benötigten Ressourcen zu erreichen. Dies erfüllt die Vorgaben des Lean Management und hilft nicht zuletzt dabei, auch die Business-orientierten Stakeholder zu überzeugen. Hervorzuheben ist, dass es für die Einbettung des Requirements Engineering in Scrum kein feststehendes „Rezept“ gibt. Vielmehr hängt die Vorgehensweise stark vom Umfeld ab, wobei das Stakeholder Management eine wichtige Rolle einnimmt.



Lars Möller

[lars.moeller@pentasys.de](mailto:lars.moeller@pentasys.de)

Lars Möller ist Business Architect bei der PENTASYS AG. Er ist der Ansicht, dass es für die Einbettung des Requirements Engineering in Scrum kein feststehendes „Rezept“ gibt. Vielmehr hängt die Vorgehensweise stark vom Umfeld ab, wobei das Stakeholder Management eine wichtige Rolle einnimmt.

Eventpartner: **AOUG** **SOUG** swiss oracle user group **iJUG** Verbund **ORACLE**

2019  
**DOAG**  
Konferenz + Ausstellung  
19. - 22. November in Nürnberg

EARLY BIRD  
BIS ZUM  
30. SEPT.



[2019.doag.org](http://2019.doag.org)