



**AKTUALISIERT FÜR SCRUM
GUIDE V. 2020**

AGILE LEADERSHIP IM SCRUM-KONTEXT

**SERVANT LEADERSHIP FÜR
AGILE LEADER UND SOLCHE,
DIE ES WERDEN WOLLEN.**



PAUL C. MÜLLER



Inhalt

Vorbemerkungen

Vorwort

Das Scrum-Framework verstehen und anwenden

Das agile Manifest

Zwölf Prinzipien des Agilen Manifests

Empirische Prozesssteuerung

Werte von Scrum

Rollen in Scrum

Scrum-Events

Scrum skalieren mit Nexus

Menschen und Teams entwickeln

Selbst organisierte Teams

Facilitation (Moderation)

Der Input

Die moderierte Veranstaltung selbst

Freie und informierte Entscheidungen

Das Ergebnis

Agile Führung

Agiles Produktmanagement

Vorhersagung und Release-Planung

Agile Produktentwicklung

Das Projektdreieck

Releaseplanung

Product Vision

- Vision Boarding
- Produktwert
- Unternehmensstrategie
- Stakeholder und Kunden
- Professionelle Produktherstellung und -auslieferung
- Software-Entwicklung
 - Emergent Software Development
 - Continuous Quality
 - Continuous Integration und Delivery
 - Optimierter Arbeitsfluss
 - Integriertes Risiko-Management
- Continuous Quality
 - Test First
- Eine agile Organisation aufbauen
 - Organisationsdesign und Kultur
 - Konkurrenzierende Werte
 - Zombie-Scrum überwinden
 - Die Aufgabe eines agilen Leaders in Bezug auf die Organisation
 - Portfolio-Planung
 - Evidence-Based Management
 - Der Einsatz von EBM in der Praxis
- Prüfungsvorbereitung “Professional Agile Leadership”
- Literaturempfehlungen

VORBEMERKUNGEN

Professional Agile Leadership™ (PAL E) ist wie auch die weiteren im Buch genannten Scrum-Zertifizierungen Eigentum der [Scrum.org](https://www.scrum.org). Das vorliegende Buch ist ohne Einflussnahme oder Auftrag der [Scrum.org](https://www.scrum.org) als Ausdruck der eigenen Auseinandersetzung des Autors mit den dargestellten Themen entstanden. Der einfacheren Lesbarkeit halber wurde im Text auf die Auszeichnung von Marken und Warenzeichen verzichtet. Diese sind aber jeweils mit gemeint und sollen so verstanden werden.

Das bereits vorgelegte Buch wurde im Dezember 2020 basierend auf der Scrum Guide der Version 2020 (November 2020 erschienen) durchgesehen und wo notwendig angepasst.

Anmerkungen: Auch wenn die deutsche Übersetzung der Scrum-Guide die Rollen inzwischen mit Product Owner:innen etc. bezeichnet behalte ich der besseren Lesbarkeit die männliche Form bei.

VORWORT

Ein Buch über ein agiles Framework zu schreiben ist schwierig, lebt Agilität doch gerade davon, dass nicht alles nach einem bestimmten Rezept abgearbeitet wird und nicht jede Fragestellung in einer Art Handbuch nachgeschlagen werden kann. Agilität lebt von Erfahrungen, Erkenntnissen, Experimenten und deren Evaluation und den daraus abgeleiteten Maßnahmen. Scrum nennt dies die drei Säulen von Scrum: Transparenz, Überprüfung und Anpassung. Man könnte es auch den agilen Regelkreis von Scrum nennen.

Als ich mich entschlossen hatte, dieses Buch zu schreiben, war dies von einer Kundenanfrage getragen. Ein Kunde stellte fest, dass es wohl kein deutschsprachiges Buch zur Vorbereitung auf die Professional Agile Leadership™ (PAL E) -Zertifizierungsprüfung der [Scrum.orggeben](#) würde und die Quellenlage auf der Webseite von [Scrum.org](#) so breit liege, dass eine Vorbereitung auf die Prüfung viel Aufwand bedeute.

Ich habe mich also entschlossen, für Menschen, die in derselben Lage wie mein Kunde sind und lieber selbst lernen als sich zwei Tage von einem Trainer durch den Stoff führen zu lassen, eine Darstellung der Prüfungsthemen zu schreiben. Dabei war mir aber wichtig, dass ich nicht nur einfach eine Vorbereitung auf einen bestehenden Fragenpool mache, sondern das vermittelte Wissen auch in einen Kontext stelle, der es dem Leser erlaubt, das Gelernte auch in der Praxis einzusetzen, mehr über die Werte und

Denkweisen hinter agiler Transformation und Entwicklung zu lernen und diese in seiner eigenen Praxis einsetzen zu können.

Dabei ist allerdings bitte zu beachten: Agil werden Sie und Ihr Team nicht durch die Lektüre eines Buches, sondern durch das Durchführen von Experimenten und den sich daraus ergebenden Erkenntnissen und Maßnahmen. Was ich dazu beitragen kann, sind ein paar Ansätze und Denkmuster, welche Sie für die Umsetzung gerne nutzen können. Beachten Sie dabei aber immer: Es geht nicht darum, dass Sie das Scrum des Autors umsetzen, sondern das Scrum Ihres/r Teams, basierend auf Ihren eigenen Erkenntnissen, getragen von Ihrem eigenen Entwicklungsprozess.

Ich wünsche Ihnen und Ihrem Team viel Erfolg bei der spannenden Reise durch eine agile Welt und hoffe, dass sie Ihnen genauso viel Freude und Erkenntnisse bringt wie mir.

Der Autor

DAS SCRUM-FRAMEWORK VERSTEHEN UND ANWENDEN

Bevor wir uns mit der Frage beschäftigen, was Scrum ist und was agile Führung im Kontext von Scrum bedeutet, scheint es wichtig zu sein, sich zunächst damit auseinanderzusetzen, was Agilität eigentlich bedeutet. Welche Ideen stehen dahinter und welchen Nutzen soll uns Agilität bringen? Um das zu erkunden, ist es sinnvoll, sich zunächst mit dem agilen Manifest, sozusagen der Verfassung der Agilität, auseinanderzusetzen:

Das agile Manifest

Im Februar 2001 trafen sich in den Wasatch-Bergen des amerikanischen Bundesstaates Utah in einer Ski-Lodge siebzehn Menschen, um gemeinsam zu reden, Ski zu fahren und zu entspannen. Sie alle waren mit der Art und Weise, wie Software-Entwicklung stattfand, nicht zufrieden und glaubten, dass Alternativen zu dokumentationsträchtigen, schwergewichtigen Software-Entwicklungsprozessen notwendig seien.

Diese Gruppe organisatorischer Anarchisten, die sich «The Agile Alliance» nannten, formulierten und unterschrieben gemeinsam das «Agile Manifest». Dabei ist zu beachten, dass die anwesenden Personen später ganz unterschiedliche Wege gegangen sind und unterschiedliche Methoden und Frameworks basierend auf der gemeinsamen Grundlage «Agiles Manifest» entwickelt haben. (Mit-) Entwickler von «Extreme Programming», «Scrum», «DSDM», «Adaptive

Software Development», «Crystal» und anderen legten hier einen gemeinsamen Grundstein für die weitere Entwicklung von Software-Entwicklung und in vielen Fällen auch für weit über die Software-Entwicklung hinausgehende Fragestellungen.

Weitere Informationen zur Entstehungsgeschichte finden Sie unter: <http://agilemanifesto.org/history.html>

Manifest für agile Softwareentwicklung

„Wir erschließen bessere Wege, Software zu entwickeln, indem wir es selbst tun und anderen dabei helfen.

Durch diese Tätigkeit haben wir diese Werte zu schätzen gelernt:

Individuen und Interaktionen mehr als *Prozesse und Werkzeuge*

Funktionierende Software mehr als *umfassende Dokumentation*

Zusammenarbeit mit dem Kunden mehr als *Vertragsverhandlung*

Reagieren auf Veränderung mehr als *das Befolgen eines Plans*

Das heißt, obwohl wir die Werte auf der rechten Seite wichtig finden, schätzen wir die Werte auf der linken Seite höher ein.“

In kurzen Worten beschreibt das Manifest damit die Grundlagen agilen Denkens.

Individuen und Interaktionen mehr als Prozesse und Werkzeuge

Das Befolgen von Prozessen und der richtige Einsatz von Werkzeugen ist zweifellos ein zentraler Erfolgsfaktor für die Durchführung erfolgreicher Entwicklungsprozesse. Trotzdem stellt ihnen das agile Manifest zu Recht «Individuen und Interaktionen» voran. Nur Individuen sind in der Lage, sich permanent weiterzuentwickeln und damit eine stetige Verbesserung des Entwicklungsprozesses und der entwickelten Produkte zu erzielen. Die Interaktion zwischen Individuen bietet zusätzliches Potential, dass aus der Arbeit eines Teams mehr wird als die Summe der Einzelleistungen. Andererseits kann ein Team nur dann optimal wirken, wenn jedes Teammitglied auch als Individuum mit Stärken, Schwächen und eigener Persönlichkeit wahrgenommen und als solches einbezogen wird.

Funktionierende Software mehr als umfassende Dokumentation

Das Festhalten von Anforderungen im Vorfeld sowie die Dokumentation der umgesetzten Resultate sind eine Grundlage für die Verständigung über Produktinhalte und die Grundlage für erfolgreiche Wartung und Weiterentwicklung. Gelingt es aber nicht, ein funktionierendes Software-Produkt zu erstellen, ist ihr Nutzen sehr begrenzt.

Zusammenarbeit mit dem Kunden mehr als Vertragsverhandlung

Verträge und Vereinbarungen zwischen Kunden und Herstellern sind von großer Bedeutung. Sie bilden eine

wichtige Basis dafür, sicherzustellen, dass alle Beteiligten einen Auftrag gleich verstehen. Trotzdem können Verträge nicht jedes mögliche Szenario detailliert abbilden und jedes mögliche Ereignis vorwegnehmen. Sie entstehen basierend auf dem Kenntnisstand zu einem bestimmten Zeitpunkt. Im Verlauf eines Projektes entwickeln sich aber sowohl Kunde als auch Lieferant weiter. Das verfügbare Wissen zum gemeinsamen Projekt nimmt zu und auch Rahmenbedingungen verändern sich. Um dieser Entwicklung im Rahmen des Projektes Rechnung zu tragen, ist eine gute und intensive Zusammenarbeit mit dem Kunden unabdingbare Voraussetzung.

Reagieren auf Veränderung mehr als das Befolgen eines Plans

Pläne sind wichtig und die Vorstellung, dass agile Vorgehensweisen, wie beispielsweise Scrum, ohne Pläne auskommen würden, ist absurd. Man könnte sogar sagen, dass bei agilen Techniken weit mehr geplant wird. Hauptunterschied ist, dass Planung in agilem Umfeld nicht primär zu (oder vor) Projektbeginn stattfindet, sondern laufend und dies immer mit einem «Just in Time»-Approach. Es macht keinen Sinn, auf Monate und Jahre hinaus detaillierte Pläne zu erstellen, wenn man das Feedback des Kunden und das Reagieren darauf als Grundlage für die Ermöglichung einer maximal wertschöpfenden Entwicklung begreift.

Zwölf Prinzipien des Agilen Manifests

Erstes Prinzip: «Unsere höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung wertvoller Software zufriedenzustellen.»

Die Zufriedenheit des Kunden ist unsere höchste Priorität. Durch wertbasierte Priorisierung in Kombination mit der Auslieferung umgesetzter Lösungsteile schon während des Entwicklungsprozesses schaffen wir nicht nur frühzeitig die Möglichkeit für ein tiefer greifendes Kundenfeedback, sondern tragen auch dabei, dass der Kunde, wenn möglich, schon vor Projektabschluss Nutzen realisieren kann.

Zweites Prinzip: «Heiße Anforderungsänderungen selbst spät in der Entwicklung willkommen. Agile Prozesse nutzen Veränderungen zum Wettbewerbsvorteil des Kunden.»

Während in komplett durchgeplanten Projekten jede Änderung eine Störung bedeutet, welche womöglich dazu führt, dass Wochen und Monate intensiver Planungs- und Koordinationsarbeit neu erstellt werden müssen, sind neue und angepasste Anforderungen in agiler Entwicklung nicht nur von vornherein mit eingeplant, sondern auch bewusst gewollt. Während der Entwicklung eines Produktes lernt sowohl der Auftraggeber als auch der Auftragnehmer laufend mehr über das zu entwickelnde Produkt. Die Resultate dieses Lernprozesses sollen nicht erst in einem möglichen Folgeprojekt oder mittels späterer Anpassung (nach Auslieferung) realisiert werden, sondern so früh wie möglich in die Entwicklung einfließen. Das kann heißen, dass bereits vorgesehene Funktionalität angepasst oder sogar weggelassen werden muss oder dass vollkommen neue Anforderungen plötzlich höchste Priorität genießen. Nur so sind wir in der Lage, unserem Kunden baldmöglichst einen größtmöglichen Nutzen zu ermöglichen.

Drittes Prinzip: «Liefere funktionierende Software regelmäßig innerhalb weniger Wochen oder Monate und bevorzuge dabei die kürzere Zeitspanne.»

Agile Entwicklung geschieht als gemeinschaftliche Leistung von Auftragnehmer und Auftraggeber. Während der Auftragnehmer seine Entwicklungsressourcen einbringt, bringt der Auftraggeber sein fachliches Know-how und seine Kenntnis der existierenden oder künftig gewünschten Prozesse ein. Diese gemeinsame Entwicklung kann dann besonders erfolgreich sein, wenn eine häufige und fundierte Kommunikation stattfindet. Diese wird durch häufige Auslieferung und das daraus entstehende Feedback gefördert. Sie gibt dem Entwickler die Möglichkeit, sicherzustellen, dass sich sein Produkt in die richtige Richtung entwickelt, und bietet dem Auftraggeber die Zuversicht, dass das erstellte Produkt seinen Anforderungen entspricht.

Viertes Prinzip: «Fachexperten und Entwickler müssen während des Projektes täglich zusammenarbeiten.»

Agile Software-Entwicklung ist vom laufenden Austausch von Fachexperten und Entwicklern geprägt. Dieser beschränkt sich nicht auf den Projektbeginn in der Anforderungsdefinition und zu Projektende in der Lösungsabnahme, wie das bei herkömmlichem Projektvorgehen oft der Fall ist. Die Zusammenarbeit findet laufend statt, sei es bei der Definition von Anforderungen und den entsprechenden Akzeptanzkriterien oder dann im Feedback bei Iterationsende. Auch dazwischen ist die Einbindung zur Klärung von Fragen und Anforderungen wichtig.

Fünftes Prinzip: «Errichte Projekte rund um motivierte Individuen. Gib ihnen das Umfeld und die