

Roman Pichler · Stefan Roock (Hrsg.)

# Agile Entwicklungspraktiken mit Scrum



dpunkt.verlag

Roman Pichler  
roman.pichler@romanpichler.com

Stefan Roock  
stefan@stefanroock.de

Lektorat: Christa Preisendanz · Vanessa Wittmer  
Copy-Editing: Ursula Zimpfer, Herrenberg  
Herstellung: Birgit Bäuerlein  
Umschlaggestaltung: Helmut Kraus, [www.exclam.de](http://www.exclam.de)  
Druck und Bindung: Media-Print Informationstechnologie, Paderborn

Bibliografische Information der Deutschen Nationalbibliothek  
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;  
detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

ISBN 978-3-89864-719-9

1. Auflage 2011  
Copyright © 2011 dpunkt.verlag GmbH  
Ringstraße 19 B  
69115 Heidelberg

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Die Verwendung der Texte und Abbildungen, auch auszugsweise, ist ohne die schriftliche Zustimmung des Verlags urheberrechtswidrig und daher strafbar. Dies gilt insbesondere für die Vervielfältigung, Übersetzung oder die Verwendung in elektronischen Systemen.

Es wird darauf hingewiesen, dass die im Buch verwendeten Soft- und Hardware-Bezeichnungen sowie Markennamen und Produktbezeichnungen der jeweiligen Firmen im Allgemeinen warenzeichen-, marken- oder patentrechtlichem Schutz unterliegen.

Alle Angaben und Programme in diesem Buch wurden mit größter Sorgfalt kontrolliert. Weder Autor noch Verlag können jedoch für Schäden haftbar gemacht werden, die in Zusammenhang mit der Verwendung dieses Buches stehen.

5 4 3 2 1 0

---

## Geleitwort

Ein weiteres Buch über Scrum? Ja! Aber nicht irgendeines, sondern eines, das ganz anders ist als alle anderen. Hier geht es um Technik und Techniken. Während es inzwischen unzählige Bücher auf dem Markt gibt, die sich mit agilen Prinzipien oder Scrum beschäftigen, gibt es nur sehr wenige, die die technischen Aspekte agiler Softwareentwicklung erläutern.

Vor mehreren Jahren haben wir Scrum bei der mobile.international GmbH eingeführt. Schnell haben wir gemerkt, dass die Vorteile von Scrum auch Nachteile mit sich bringen. Scrum ist ein sehr leichtgewichtiger und leicht zu beschreibender Prozess. Dafür ist Scrum umso schwerer zu lernen und nur Stück für Stück merkt man, was einem auf dem Weg noch fehlt und wie man seine Organisation und das ganze Umfeld verändern muss. Der Vorteil von Scrum, so dünn und leichtgewichtig zu sein, bringt aber mit sich, dass Scrum sich zu Entwicklungstechniken ausschweigt. Aber schnell wurde uns klar, dass wir auch an unserer technischen Exzellenz arbeiten müssen, um uns kontinuierlich zu verbessern. Agile Softwareentwicklung funktioniert nur gut, wenn man eine fast konstante Aufwandskurve erreicht – wenn also ein zu entwickelndes Feature zum Ende eines Projekts nicht wesentlich teurer ist als zu Beginn. Und dazu wird man irgendwann die Fragen beantworten müssen, wie man einen emergenten Architekturbegriff unterstützt, wie man einen inkrementellen Entwurf beherrscht und dies letztlich durch ständige – in Teilen automatisierte – Refactorings fördert. Wer richtig gut sein will, entwickelt von Anfang an testgetrieben.

Aber agile Softwareentwicklung setzt auch kurze Feedbackzyklen voraus und im Ideal dann auch ständige Releases. Und das wiederum ist alles nur zu erreichen mit Continuous Integration inklusive 10-Minuten-Build und letztlich auch automatisierten Akzeptanztests, um die Transaktionskosten für konstantes, schnelles Feedback gering halten zu können und um nicht zu größeren Iterationen oder Arbeitsschritten gezwungen zu werden.

Neben Prozess und Tools muss aber auch das Umfeld stimmen. Die Fülle der Themen macht klar, dass jeder von uns ständig neu dazulernen muss, denn die in diesem Buch beschriebenen Themen sind komplex, fordernd und unterliegen einer dynamischen Entwicklung. Am Ende heißt Software mit Scrum zu entwickeln

auch, Software auf technisch und handwerklich sehr hohem Niveau zu entwickeln. Und dafür muss das Umfeld stimmen und eine Lernkultur aufweisen, die ständiges Infragestellen des Status quo und Lernen unterstützt. Deswegen freue ich mich besonders über die Kapitel im Buch, in denen es um Themen geht, die uns auf diesem Weg selbst weitergebracht haben: »Dojos und Katas«, das wunderbare spielerische Formen des fokussierten Übens erklärt und nachvollziehbar macht, sowie »Pair Programming und Collective Ownership«, das alte, aber immer noch aktuelle und immer wieder neu interpretierte Themen frisch präsentiert.

Ein Kapitel über den vielleicht schwierigsten Aspekt agiler Softwareentwicklung, nämlich verteilte Teams, rundet das Buch zum Ende hin ab. Verteilte agile Entwicklung ist besonders schwierig, denn der Kern agiler Entwicklung ist Eliminierung von Overhead durch ständige, enge Kommunikation. Wie diese erreicht werden kann, obwohl Teams räumlich getrennt sind, wird in diesem Kapitel nachvollziehbar gemacht.

Man muss dieses Buch nicht von vorne bis hinten durchlesen – auch wenn dies sehr inspirierend ist und Anhaltspunkte gibt, welche Aufgaben noch zu lösen sind. Aber immer, wenn das ewig zwischen Prozessverbesserung und technischer Verbesserung hin- und herschwingende Pendel wieder bei der Technik landet, kann man sich dieses Buch schnappen und sicher sein, das passende Kapitel zu finden. Und jedes der Kapitel erläutert genau eines der erwähnten Themen grundlegend und praxisnah jeweils geschrieben von einem Autor, der zu den Praktikern und Größen der agilen Szene in Deutschland gehört. Jedes Kapitel beschreibt das Kondensat der praktischen Erfahrung eines Autors – erprobte agile Praktiken, die schon vielen Unternehmen geholfen haben, auf dem Weg zu technischer Exzellenz weiterzukommen.

Roman Pichler und Stefan Roock ist mit diesem Buch eine wunderbare Sammlung relevanter Aufsätze gelungen, die uns selbst und vielen anderen Agilisten noch lange Jahre helfen wird.

Markus Andrezak  
*Potsdam im Februar 2011*