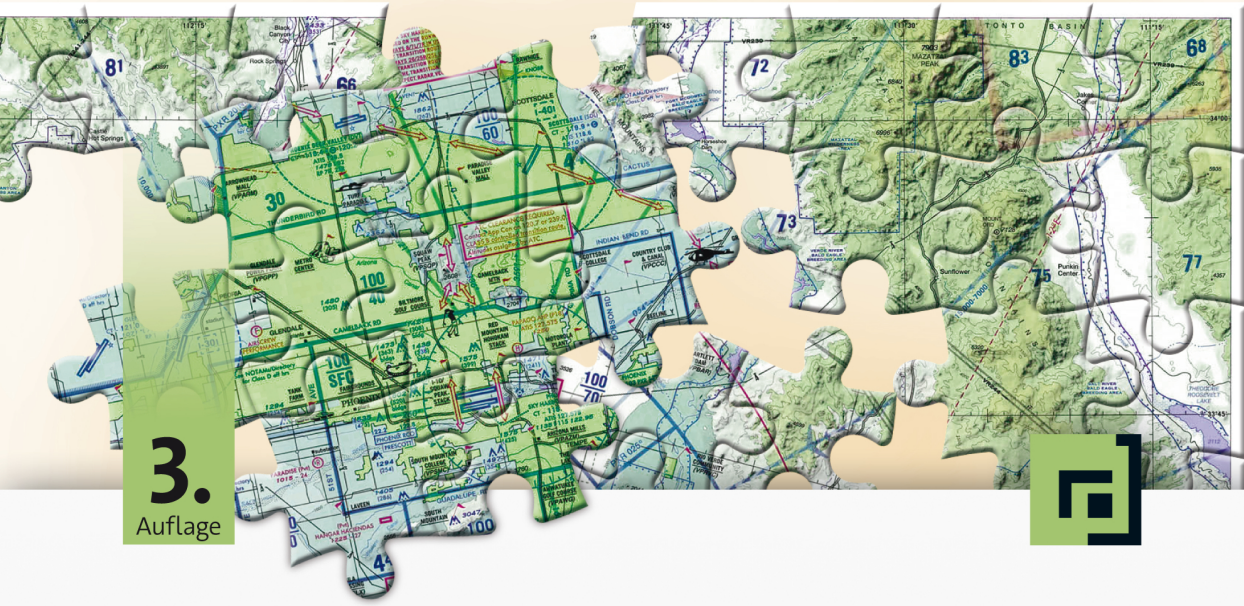


3. Auflage

Johannes Bergsmann

Requirements Engineering

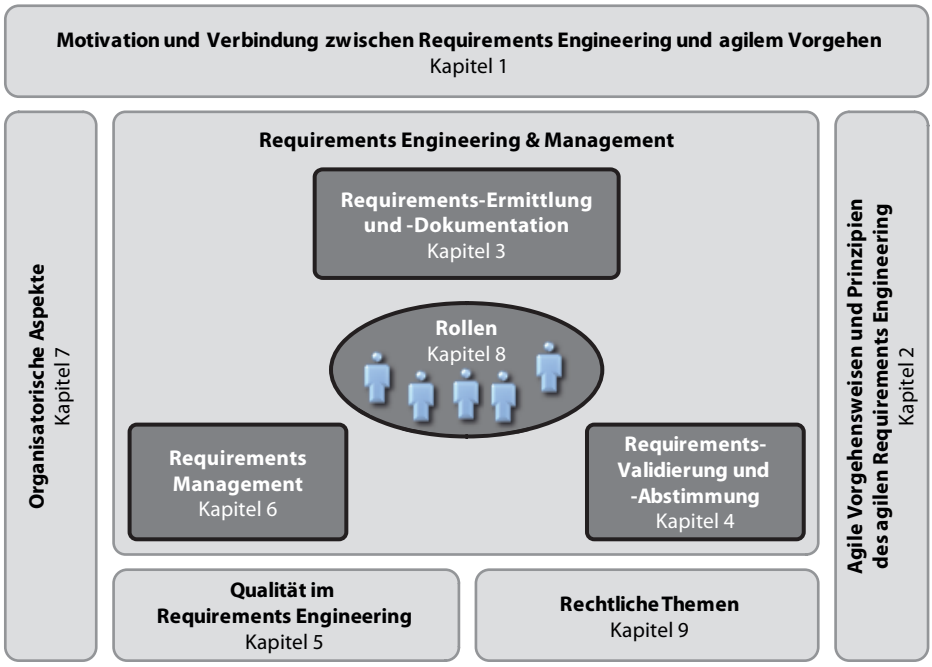
für die agile Softwareentwicklung

Methoden, Techniken und Strategien

→ Unter Mitwirkung von Markus Unterauer

Optimal für IREB® RE@Agile

dpunkt.verlag


Motivation und Verbindung zwischen Requirements Engineering und agilem Vorgehen
Kapitel 1
Organisatorische Aspekte
Kapitel 7
Requirements Engineering & Management
Requirements-Ermittlung und -Dokumentation
Kapitel 3
Rollen
Kapitel 8
Requirements Management
Kapitel 6
Requirements-Validierung und -Abstimmung
Kapitel 4
Qualität im Requirements Engineering
Kapitel 5
Rechtliche Themen
Kapitel 9
Agile Vorgehensweisen und Prinzipien des agilen Requirements Engineering
Kapitel 2

Johannes Bergsmann hat technische Informatik studiert und arbeitete ca. 11 Jahre als Softwareentwickler, Projektleiter, Technischer Leiter, Architekt, Produktmanager und Berater in einem internationalen Systemhaus und als selbstständiger Unternehmer. Im März 2003 gründete er »Software Quality Lab« und begleitet seither als Berater und Trainer viele Unternehmen im Bereich Requirements Engineering und Prozessgestaltung.

Johannes Bergsmann ist zertifizierter Scrum Master, Sachverständiger für Informatik bei Gerichten, als Lehrbeauftragter an Fachhochschulen im Bereich Softwarequalitätsmanagement tätig, ist Autor vieler Fachartikel und hält Fachvorträge bei verschiedenen Veranstaltungen und Konferenzen.

Unter Mitwirkung von Markus Unterauer:

Markus Unterauer hat Wirtschaftsinformatik studiert. In seiner Berufspraxis war er in vielen Bereichen der Softwareentwicklung wie Architektur, Entwurf, Entwicklung, Testen, Testautomatisierung bis zu Deployment tätig. Er lernte dabei sowohl klassische als auch agile Projekte und Methoden intensiv kennen.

Seit 2012 arbeitet Markus Unterauer bei Software Quality Lab als Berater und Trainer. Er ist zertifizierter Scrum Master und hat sich auf die Bereiche Softwareprozesse und Anforderungsmanagement spezialisiert. Markus Unterauer ist auch als Vortragender in diesen Themenbereichen immer wieder auf Konferenzen tätig.

Johannes Bergsmann

Requirements Engineering für die agile Softwareentwicklung

Methoden, Techniken und Strategien

Unter Mitwirkung von Markus Unterauer

3., überarbeitete und aktualisierte Auflage

dpunkt.verlag

Johannes Bergsmann
johannes.bergsmann@software-quality-lab.com

Markus Unterauer
markus.unterauer@software-quality-lab.com

Lektorat: Christa Preisendanz
Lektoratsassistenz: Julia Griebel
Copy-Editing: Ursula Zimpfer, Herrenberg
Layout & Satz: Birgit Bäuerlein
Herstellung: Stefanie Weidner
Umschlaggestaltung: Helmut Kraus, *www.exclam.de*
Druck und Bindung: mediaprint solutions GmbH, 33100 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:
Print 978-3-86490-929-0
PDF 978-3-96910-953-3
ePub 978-3-96910-954-0
mobi 978-3-96910-955-7

3., überarbeitete und aktualisierte Auflage 2023

Wieblinger Weg 17
69123 Heidelberg

Teile dieses Buches orientieren sich am Lehrplan RE@Agile des IREB e.V. Der Besitz und das Urheberrecht dieses Lehrplans und Studienleitfadens liegt bei IREB e.V. und den Autoren: Lars Baumann, Stefan Gärtner, Peter Hruschka, Kim Lauenroth, Markus Meuten, Sacha Reis, Gareth Rogers, Francois Salazar, Hans-Jörg Steffe, Thorsten Weyer.

Hinweis:
Dieses Buch wurde auf PEFC-zertifiziertem Papier aus nachhaltiger Waldwirtschaft gedruckt. Der Umwelt zuliebe verzichten wir zusätzlich auf die Einschweißfolie.

Schreiben Sie uns:
Falls Sie Anregungen, Wünsche und Kommentare haben, lassen Sie es uns wissen: *hallo@dpunkt.de*.

5 4 3 2 1 0

Vorwort

Viele Projekte werden aus verschiedenen Gründen nicht so effizient und effektiv abgewickelt, wie dies möglich wäre: Zum Beispiel wenn die Fachexperten zwar wissen, dass sie mit den Entwicklerinnen täglich kommunizieren sollten, dies jedoch nicht können,

- weil sie im Tagesgeschäft schon zu stark involviert und überlastet sind,
- weil sie von ihrer Persönlichkeit her keine aktiv kommunizierenden Typen sind,
- weil sie mitten im Projekt andere Aufgaben zugewiesen bekommen,
- weil sie die Abteilung oder Firma wechseln,
- weil es schwierig ist, das Wissen in den Köpfen der Beteiligten bewusst zu machen und an andere zu kommunizieren,
- weil Kommunikation zwischen zwei Personen immer auch mit einer Interpretation und evtl. mit einer Veränderung der Information einhergeht oder
- aus verschiedensten anderen Gründen.

Als Berater habe ich in meiner bisherigen beruflichen Tätigkeit mehr als 200 verschiedene Projekte begleitet oder auch selbst in unterschiedlichen Rollen (Entwickler, Tester, Projektleitung, Architekt, Produktmanager, Analytiker, Coach, Berater etc.) mitgearbeitet. Viele dieser Projekte – vor allem in den letzten 20 Jahren – waren agile Projekte oder Projekte, in denen die Mitwirkenden zumindest versuchten, einige der agilen Prinzipien umzusetzen.

Am erfolgreichsten und effizientesten waren in dieser langen Zeit immer diejenigen Projekte, bei denen ich agile Vorgehensweisen mit Elementen aus dem klassischen Requirements Engineering und Projektmanagement ergänzte und so die Stärken jeder Methodik voll ausnutzen konnte.

Man könnte alle diese Softwareprojekte mit einer Autofahrt von München nach Rom vergleichen. Im Idealfall fahren wir auf gerader Strecke mit konstanter und optimaler Geschwindigkeit mit unserem Auto alleine auf der Straße. In der Praxis aber hat die Straße Kurven, es gibt Verkehrsbeschränkungen, in den Bergen ist evtl. auch Eis auf der Straße, es gibt Staus, andere Autofahrer, die rücksichtslos unterwegs sind und nur ihr eigenes Ziel im Blick haben, unser Auto hat eine Panne etc.

Auf alle diese individuellen Situationen sollten wir vorbereitet sein und unser Vorgehen der jeweiligen Situation entsprechend anpassen, damit wir unser Ziel auch erreichen. Generell werden wir grob planend vorgehen, z. B. den Startzeitpunkt bestimmen, den ungefähren Zeitrahmen der Fahrt abschätzen und den Streckenverlauf z. B. für die Alpenüberquerung über den Brenner planen. Wir sollten auch ungefähr wissen, welches Wetter zu erwarten ist, und abhängig davon die Reifen, den Frostschutz, die Klimaanlage etc. entsprechend vorbereiten. Im Verlauf der Fahrt wird es vielleicht zu Änderungen kommen, z. B. wenn der Brennertunnel wegen eines Unfalls gesperrt ist. In diesem Fall werden wir agil darauf reagieren müssen und die Umleitungsstrecke über den Pass nehmen (schließlich haben wir ja Google Maps dabei ☺). Wenn wir im Vorfeld in der »Architektur« unseres Autos diese Situation mangels vorausschauender Planung nicht berücksichtigt haben und beim ersten kurzen Anstieg der Passstraße feststellen, dass wir ohne Winterreifen und Schneeketten auf der verschneiten Straße nicht weiterkommen, müssen wir wiederum agil reagieren und nun einen großen Umweg über die Tauernautobahn nehmen. Beide Vorgehensweisen haben daher ihre Berechtigung. Als Softwareentwicklerin, Projektleitung, Scrum Master, Product Owner – oder welche Verantwortlichkeit auch immer wir im Projekt innehaben – sollten wir viele unterschiedliche Methoden im Köcher haben und diese für unser Projekt zur optimalen Vorgehensweise kombinieren.

Das Agile Manifest

Das Agile Manifest [Agile Manifesto] (siehe auch Abschnitt 1.2) beschreibt in seinen vier Leitsätzen und zwölf Prinzipien die Eckpfeiler, an denen sich praktisch alle agilen Vorgehensweisen orientieren. Darin wird unter anderem festgehalten, dass funktionierende Software und Zusammenarbeit mit den Kunden wichtiger sind als umfassende Dokumentation und Vertragsverhandlungen, wobei im Zusatz angeführt ist, dass umfassende Dokumentation und Vertragsverhandlungen auch als wichtig angesehen werden. So sollten Projekte idealerweise ablaufen. Wenn man die Praxis in vielen Softwareprojekten erlebt hat, wird man dem Agilen Manifest begeistert zustimmen und die Aussage und Sichtweise uneingeschränkt unterstützen.

Die aus dem Agilen Manifest entstandenen verschiedenen Vorgehensweisen wie z. B. Scrum greifen bestimmte Aspekte und Themen daraus auf und lassen andere Themen bewusst offen und unkonkret. Unter Berücksichtigung der Projektsituation und Rahmenbedingungen müssen diese offenen Themen selbst passend definiert werden.

Wenn Requirements zum Problem werden

Sehr oft wurde und wird in verschiedenen Stadien eines Projekts das Requirements Engineering und Requirements Management zum Thema, z.B. wenn …

- … eine externe Kundschaft schon bei Projektstart einen Festpreis vereinbaren will und seine Juristen oder Einkäufer vorab wissen wollen, was denn schlussendlich für den vereinbarten Preis geliefert wird.
- … bei der Abnahme eines Produkts durch die Kundschaft plötzlich wichtige Personen fehlende Funktionen bemängeln und nicht klar ist, ob dies vom Lieferanten noch als Auftragsbestandteil geschuldet wird oder ob das nun ein kostenpflichtiger Change Request ist.
- … im Laufe des Projekts eine auf Basis von User Stories (siehe Abschnitt 3.4.2) entwickelte Eingabemaske schon zum x-ten Mal über mehrere Iterationen hinweg wieder und wieder angepasst und verändert wird, weil der Kunde sich immer wieder etwas anderes als Ergebnis vorstellt.
- … Entwicklungsverantwortliche, die auf die Frage »Was kann denn das Produkt, das Sie entwickeln, nun eigentlich alles?« nur sagen können, dass sie Tausende User Stories in ihrem Request-Tool dokumentiert haben und dort nachzulesen ist, was umgesetzt wurde.
- … das Projektteam zwar weiß, was es die letzten zwei, drei Iterationen entwickelt hat und was es die kommenden zwei Iterationen entwickeln wird, jedoch keinen Überblick mehr darüber hat, welche Funktionalität das Produkt insgesamt hat.
- … »der Wald vor lauter Bäumen nicht mehr erkannt« wird und die Zusammenhänge für die Beteiligten möglicherweise schon verloren gegangen sind.

Alle diese geschilderten Fälle sind primär auf mangelndes Requirements Engineering und Requirements Management zurückzuführen. Tendenziell treten solche Probleme in Projekten auf, in denen ein Vorgehensmodell gewählt wurde, das viele thematische Freiheiten bietet, die offengelassenen Teile aber nicht vorab zwischen den beteiligten Personen festgelegt wurden oder das Modell nicht an die gegebene Situation angepasst wurde.

Gerade agile Vorgehensweisen überlassen die konkrete Ausgestaltung des Requirements Engineering (RE) zum Großteil der Entscheidung des Teams. RE wird in den verschiedenen agilen Methoden nur sehr grob beschrieben. In der Begeisterung der ersten Stunde möchten viele Teams möglichst rasch starten und erste Erfolge zeigen und beginnen mit einem sehr intuitiven Ansatz, wie z.B. einer einfachen Liste von User Stories. User Stories sind eine sinnvolle Technik aus den agilen Vorgehensweisen und ich baue in vielen Abschnitten dieses Buches auf dieser Technik auf. Es gibt jedoch auch viele Techniken aus dem Requirements Engineering, die in agilen Vorgehensweisen angewendet werden können. Aus meiner Er-

fahrung bietet die Kombination der verschiedenen Herangehensweisen gute Lösungen für die Requirements-Herausforderungen, nicht nur in agilen Projekten.

In diesem Buch werden daher viele verschiedene Techniken und Methoden aus den agilen Vorgehensweisen vorgestellt, mit klassischen Methoden verknüpft und ergänzt und in einen strukturierten und systematischen Zusammenhang mit Requirements Engineering gebracht.

Über dieses Buch

In meiner Beratungspraxis habe ich über viele Jahre hinweg Management und Teams hinsichtlich Requirements Engineering beraten und gecoacht und gesehen, wie schwierig es den Beteiligten in vielen Projekten fällt, agil und RE in Einklang zu bringen. Daher entschloss ich mich im Jahr 2014 dazu, die erste Auflage dieses Buches zu schreiben.

Der Fokus dieses Buches liegt darauf, gute Methoden und Techniken – egal aus welchem Zeitalter oder mit welcher Ausrichtung – unvoreingenommen aufzugreifen und sie nicht von vornherein auszuschließen, nur weil es sich dabei um »plangetriebene« oder »agile« Methoden oder Techniken handelt. Aussagen wie »klassische plangetriebene Methoden sind überfrachtet, ineffizient und schlecht« oder »agile Methoden sind was für Individualisten oder Dokumentationsverweigerer« gehören für mich nicht zu einer weitsichtigen und nachhaltigen Denkweise.

Jede genannte Methode und Technik stellt ein Werkzeug im Werkzeugkasten der Softwareentwicklung dar und soll hier mit ihren Vorteilen und Nachteilen in bestimmten Projektsituationen betrachtet werden. Es soll klar werden, wo und wie diese Methoden und Techniken in agilen Projekten eingesetzt werden können, um einen Nutzen für das Projekt zu stiften. Es werden daher auch sinnvolle Anwendungsmöglichkeiten und Fallstricke der einzelnen Methoden und Techniken dargestellt und im Kontext eines nachhaltigen, systematischen und praxisorientierten Requirements Engineering in der agilen Softwareentwicklung betrachtet.

Ziel ist es, einen Überblick und einen Werkzeugkasten anzubieten, der zeigt, welche Methoden und Techniken zusätzlich zu den sehr oft angewendeten User Stories noch sinnvoll sind und welche Hindernisse und Problemstellungen im Zusammenhang mit Requirements Engineering in agilen Projekten auftreten können.

Abgerundet wird das Thema durch die Behandlung von Qualitätsaspekten für Requirements, durch organisatorische Aspekte, einen Blick auf die Zusammenhänge der RE-Techniken und durch rechtliche Aspekte sowie durch Tipps und Tricks bei der konkreten Anwendung von Requirements-Themen im agilen Umfeld.

Das Buch richtet sich an Product Owner, Produktverantwortliche, Projektleitung, Softwareauftraggeber, Scrum Master, Entwicklerinnen, Tester und alle anderen Personen, die sich mit nachhaltigem und systematischem Requirements Engineering und Requirements Management in der agilen Softwareentwicklung beschäftigen oder davon betroffen sind.

Dieses Buch ist keine komplette Einführung in agile Methoden. Kapitel 1 und 2 geben zwar einen kurzen Überblick über das Agile Manifest, wesentliche agile Konzepte und Scrum, als den am häufigsten eingesetzten Vertreter agiler Methoden. Der Hauptfokus des Buches liegt jedoch auf dem Requirements Engineering und Requirements Management.

Hilfreich beim Lesen ist ein grundlegendes Verständnis im Bereich Requirements Engineering und insbesondere die Kenntnis der allgemein anerkannten Begriffe, Techniken und Vorgehensweisen dieses Themenbereichs. Die Inhalte und die Begriffe, die das International Requirements Engineering Board (IREB®) [IREB] für die Ausbildung zum »Certified Professional for Requirements Engineering (CPRE) – Foundation Level« in seinem Lehrplan [IREB CPRE FL] vorgibt, stellen den derzeitigen Stand der Technik zum Thema Requirements Engineering dar, der als Grundlage für das Lesen dieses Buches empfohlen wird.

Das Buch soll Einführungslektüre sowie Praxisleitfaden sein und ist nicht als wissenschaftlich komplette Abhandlung über das Thema Requirements Engineering in der agilen Softwareentwicklung gedacht. Es kann daher sequenziell oder auch auszugsweise gelesen werden. Als Autor ist es mir natürlich am liebsten, wenn Sie dieses Buch ständig an Ihrem Arbeitsplatz griffbereit haben und bei Fragen oder Unklarheiten zu Requirements-Engineering-Techniken oder -Vorgehensweisen darin einen schnellen Rat und brauchbare Infos finden.

In der Praxis hat man leider oft nicht die Zeit, beim Auftauchen einer Frage ein ganzes Buch z. B. zu Use Cases, User Stories oder Behavior Driven Development zu lesen. Ich habe daher auf eine möglichst große Unabhängigkeit der einzelnen Themen und Kapitel geachtet, sodass das Buch auch als Nachschlagewerk in der täglichen Arbeit verwendet werden kann. Mit entsprechendem Vorwissen in agilen Methoden und Requirements Engineering kann es auszugsweise gelesen und verstanden werden. Für die Leserschaft aus der Praxis müssen das Wichtigste und die Zusammenhänge in Kürze ersichtlich sein. Daher gibt es in vielen Kapiteln eine tabellarische Übersicht über wesentliche Punkte und verschiedene Überblicksgrafiken, die die Zusammenhänge auf einen Blick darstellen.

Auch wenn in vielen Kapiteln Begriffe aus Scrum verwendet oder zitiert werden, so wurde darauf geachtet, die einzelnen Themenbereiche möglichst allgemeingültig zu halten. Das Buch adressiert nicht »Requirements Engineering in Scrum-Projekten«, sondern behandelt generell das Thema Requirements Engineering im agilen Umfeld. Die beschriebenen Techniken und Methoden können auch in anderen agilen Vorgehensweisen angewendet werden.

Beispiele werden mit einem grauen Hintergrund versehen:

In diesem Buch werden verschiedene Beispiele und Formulierungen aus der Praxis und für die Praxis aufgeführt. Die meisten dieser Beispiele sind abgeleitet aus einem Projekt zum Thema »Zeiterfassung«. Es geht in diesem Beispielszenario darum, dass die Tagesarbeitszeiten inklusive der Pausen und Abwesenheitszeiten von Mitarbeitenden eines Unternehmens erfasst, ausgewertet und verwaltet werden können und dass auch die Erfassung, Auswertung und Verwaltung von Projektarbeitszeiten durchgeführt werden kann – also von einzelnen Zeitblöcken der Tagesarbeitszeit, die bestimmten Projekten zugeordnet werden.

Weiterführende Literatur zum Thema Requirements Engineering und agile Vorgehensweisen ist im Anhang E zu finden.

Ich freue mich, wenn dieses Buch für Sie als Anwenderin oder Experte in Ihrer täglichen Praxis eine Unterstützung und Anregung ist und für Sie als an der Schulung und Zertifizierung RE@Agile interessierte Person eine gute Informations- und Lerngrundlage darstellt.

Für die Weiterentwicklung dieses Themas hoffe ich natürlich auch auf zahlreiches Feedback und interessante Diskussionen (bitte direkt an *johannes@bergsmann.at*).

Johannes Bergsmann
Linz, im November2022

Hinweise zur 2. Auflage

2017 wurde der Lehrplan RE@Agile für eine Ausbildung des International Requirements Engineering Board (IREB®) herausgegeben, der die Lücke zwischen Requirements-Engineering-Methoden und agilen Methoden schließt. Sehr viele Inhalte dieses Lehrplans deckten sich bereits mit der ersten Auflage dieses Buches und die Motivation hinter diesem Lehrplan ist dieselbe, wie sie hinter meinem Buch steht.

Daher waren eine Anpassung und Überarbeitung der ersten Auflage ein logischer und sinnvoller Schritt zur zweiten Auflage dieses Buches, damit dieses nun auch als Lernunterlage für am RE@Agile interessierte Personen verwendet werden kann.

Hinweise zur 3. Auflage

Nachdem Mitte 2022 auch der Advanced Level RE@Agile in der Version 2 herausgegeben wurde, habe ich dies zum Anlass genommen, dieses Buch erneut entsprechend anzupassen, viele Bereiche auf den neuesten Stand zu bringen und einige Stellen auch wesentlich zu überarbeiten.

Die Struktur und die Inhalte dieses Buches orientieren sich am Lehrplan RE@Agile des International Requirements Engineering Board (IREB®). Damit bietet es den an dieser Ausbildung und Zertifizierung interessierten Personen nun eine praxisnahe Vorbereitung und Lerngrundlage sowie noch viele ergänzende Inhalte, die über den Lehrplan hinausgehen. In Abschnitt 1.1.3 ist angeführt, welche Kapitel zur Berücksichtigung des Lehrplans Advanced Level RE@Agile neu hinzugenommen bzw. wesentlich ergänzt wurden. Da die Ausrichtung und der Umfang der in diesem Buch behandelten Themen jedoch weiter gefasst sind, war es erforderlich, einige wenige Abschnitte in unterschiedlicher Reihenfolge zu behandeln (insbesondere der Themenblock »Skalierung von RE@Agile«) sowie zusätzliche Inhalte aufzunehmen, die für den Gesamtfokus des Buches relevant sind.

Ende 2020 wurde der Scrum Guide an einigen Stellen wesentlich angepasst und diese Änderungen wurden in dieser Auflage ebenfalls eingearbeitet.

Danksagung

Ich bedanke mich bei allen Personen, mit denen ich im Laufe der letzten Jahre zum Thema Requirements Engineering und agile Methoden diskutieren durfte und die schlussendlich dazu beigetragen haben, dass ich mich zum Schreiben der 1. Auflage und auch weiterer Auflagen dieses Buches entschlossen habe und entsprechende Sichtweisen aus der Praxis mit einfließen lassen konnte. Insbesondere auch dem IREB® und seinen Mitgliedern, die dieses Thema weiter vorantreiben, den Lehrplan stetig weiterentwickeln und anpassen, um den Anforderungen in der Praxis gerecht zu werden.

Ein besonderer Dank gilt Markus Unterauer, der mich als Koautor bei der Erstellung der ersten Auflage dieses Buches maßgeblich unterstützt hat und wesentliche Teile vor allem im Kapitel 4 »Requirements-Validierung und -Abstimmung«, Kapitel 6 »Requirements Management« und Kapitel 8 »Requirements-Engineering-Rollen« beigetragen und viele gute Anregungen gegeben hat.

Vielen Dank auch dem dpunkt.verlag – insbesondere Christa Preisendanz – für die initiale Anregung und für die wertvolle Unterstützung im Laufe der Ausarbeitung dieses Buches sowie für die Motivation zur Überarbeitung und Erstellung der Neuauflagen.

Und schlussendlich gilt ein ganz großer Dank meiner Frau Petra und meinen Kindern Beate und Barbara, die mich dadurch unterstützt haben, dass sie mir die Zeit zum Schreiben dieses Buches gegeben haben.

Inhaltsübersicht

Inhaltsverzeichnis

1 Einleitung und Motivation

1.1 Fokus dieses Buches

Der Fokus dieses Buches liegt auf einer unvoreingenommenen Betrachtung von guten Methoden und Techniken – egal welchen Alters oder welcher Ausrichtung – für die Anwendung des Requirements Engineering in agilen Projekten, um einen Nutzen für die Projektarbeit zu stiften.

Nachhaltiges, systematisches und praxisorientiertes Requirements Engineering in der agilen Softwareentwicklung steht dabei im Vordergrund und nicht die wissenschaftlich vollständige Aufarbeitung.

Abgerundet wird das Buch durch die Behandlung von Qualitätsaspekten für Requirements, organisatorische Aspekte sowie durch rechtliche Aspekte bei der Anwendung von Requirements im agilen Umfeld.

1.1.1 Zielgruppen

Das Buch richtet sich an Product Owner, Produktverantwortliche, Projektleitung, Softwareauftraggeberin, Scrum Master, Entwicklerinnen, Tester und alle anderen Personen, die sich mit nachhaltigem und systematischem Requirements Engineering und Requirements Management in der agilen Softwareentwicklung beschäftigen oder davon betroffen sind.

Folgende Zielgruppen werden primär durch den Lehrplan [IREB RE@Agile Primer] wie auch [IREB RE@Agile AL] adressiert:

- Requirements Engineers, die sich mit agiler Entwicklung befassen und ihre Techniken in dieser Umgebung erfolgreich anwenden möchten.
- Requirements Engineers, die etablierte Konzepte und Techniken agiler Ansätze anwenden und ihre Requirements-Engineering-Prozesse verbessern möchten.
- Fachkräfte für agile Entwicklungsprozesse, die die Werte und Vorteile des Requirements Engineering in agilen Projekten verstehen möchten.
- Fachkräfte für agile Entwicklungsprozesse, die die agile Entwicklung durch bewährte Requirements-Engineering-Techniken und -Methoden verbessern möchten.

- Personen aus verwandten Disziplinen – IT-Management, Tester, Entwicklerinnen, Architekten und andere Vertreter im Bereich der Entwicklung (überwiegend, aber nicht ausschließlich Softwareentwicklung) –, die verstehen möchten, wie sie Requirements-Engineering- und agile Ansätze in Entwicklungsprozessen erfolgreich kombinieren können.

1.1.2 Abbildung des Lehrplans IREB CPRE RE@Agile Primer

[IREB RE@Agile Primer]

Seit 2017 gibt es den Lehrplan CPRE RE@Agile Primer des International Requirements Engineering Board (IREB®), der die Lücke zwischen Requirements-Engineering-Methoden und agilen Methoden schließt.

Einerseits wird die Sicht des IREB-Standards auf agile Werte abgebildet und andererseits wird eine agile Sicht auf die Werte des Requirements Engineering dargestellt. Zum Inhalt des Lehrplans CPRE RE@Agile Primer gehören Klassifizierung und Beurteilung von Requirements-Engineering-Artefakten und -Techniken im Zusammenhang mit Agilität, agilen Artefakten und Techniken, Requirements Engineering und wesentlichen Prozesselementen in der agilen Produktentwicklung. Das Modul RE@Agile Primer zeigt die Motivation für die Verwendung agiler Methoden in Entwicklungsprozessen auf und betont die Synergie zwischen Requirements Engineering und Agilität.

Da dieses Buch viele ergänzende und vertiefende Inhalte umfasst, die nicht im Lehrplan CPRE RE@Agile Primer enthalten sind, ist in einigen Bereichen eine andere Strukturierung gegeben. Nachfolgend sind ein erster Überblick und einige Hinweise angeführt, wo in diesem Buch die wichtigsten Inhalte des Lehrplans zu finden sind:

Lehrplan RE@Agile Primer	LE	in diesem Buch
LE 1 **Motivation und Denkweisen**	1.1 1.2	■ Abschnitt 1.2 ■ Abschnitt 1.2.1
	1.3	■ Abschnitt 1.1.4 und 1.2.2
	1.4	■ Abschnitt 1.2.5, 1.2.6 und 1.2.7
	1.5	■ Abschnitt 2.1.3
LE 2 **Grundlagen von RE@Agile**	2.1	■ Abschnitt 2.1 und 3.2.2
	2.2	■ Abschnitt 2.1.2
	2.3 – 2.7	■ Abschnitt 2.2
LE 3 **Artefakte und Techniken in RE@Agile**	3.1	■ Abschnitt 3.1 mit den Abschnitten 3.1.1 bis 3.1.8 ■ Abschnitte 3.2.2 und 3.2.3
	3.2	■ Abschnitt 3.1 mit den Abschnitten 3.1.9 und 3.1.10 ■ Die Einleitungen von Kapitel 4 und 6

→

Lehrplan RE@Agile Primer	LE	in diesem Buch
LE 4 **Organisatorische Aspekte von RE@Agile**	4.1	■ Abschnitt 7.1
	4.2	■ Abschnitt 7.2
	4.3	■ Abschnitte 7.3 und 7.4
LE 5 **Begriffsdefinitionen, Glossar**		■ Siehe auch [IREB Glossar] und [IREB RE@Agile Glossar]

Die für den Lehrplan [IREB RE@Agile Primer] relevanten Kapitel und Stellen werden mit »[IREB RE@Agile Primer]« sowie ggf. mit der Ergänzung »LE xxx« gekennzeichnet und sind prüfungsrelevant. LE xxx referenziert auf die jeweilige Lerneinheit (LE = Lerneinheit) des Lehrplans. Bei allen anderen Unterkapiteln handelt es sich um ergänzende Informationen und Hinweise oder Anregungen für die Praxis.

Wenn die Bezeichnung »RE@Agile« als Kurzbegriff verwendet wird, so bezieht sich dies immer auf den Lehrplan [IREB RE@Agile Primer].

Weiterführende Literatur zu den Themen Requirements Engineering und agile Vorgehensweisen ist im Literaturverzeichnis im Anhang E des Buches zu finden.

1.1.3 Abbildung des Lehrplans IREB CPRE Advanced Level RE@Agile – Practitioner/Specialist

[IREB Advanced Level RE@Agile – Practitioner/Specialist]

Der Lehrplan IREB CPRE Advanced Level RE@Agile wurde in Version 2 im Sommer 2022 veröffentlicht und richtet sich an Requirements Engineers und Experten für agile Entwicklungsprozesse. Der Schwerpunkt liegt auf dem Verständnis und der Anwendung von Verfahren und Techniken aus der Disziplin des Requirements Engineering in agilen Entwicklungsprozessen sowie auf dem Verständnis und der Anwendung von Konzepten, Techniken und essenziellen Prozesselementen agiler Ansätze in Requirements-Engineering-Prozessen. Die Zertifizierung versetzt Personen mit Requirements-Engineering-Kenntnissen in die Lage, in agilen Umgebungen zu arbeiten, und Experten für agile Entwicklungsprozesse erlaubt sie, bewährte Requirements-Engineering-Verfahren und -Techniken in agilen Projekten anzuwenden.

Wie bei allen anderen Advanced-Level-Modulen des IREB CPRE-Ausbildungsmodells ist das Zertifikat CPRE FL (Foundation Level) eine Voraussetzung für die Teilnahme an den Advanced-Level-Zertifizierungsprüfungen zum RE@Agile – Practitioner und RE@Agile – Specialist. Zudem wird dringend angeraten, dass Teilnehmende mindestens über ein Zertifikat für agile Entwicklung (d.h. RE@Agile Primer oder ein Scrum-Zertifikat) verfügen oder vergleichbare Kenntnisse über agile Ansätze besitzen.

Da dieses Buch viele ergänzende und vertiefende Inhalte umfasst, die nicht im Lehrplan IREB CPRE Advanced Level RE@Agile – Practitioner/Specialist enthalten

sind, ist in einigen Bereichen eine andere Strukturierung gegeben. Nachfolgend sind ein erster Überblick und einige Hinweise angeführt, wo in diesem Buch die wichtigsten Inhalte des Lehrplans für das Modul Advanced Level RE@Agile zu finden sind:

Lehrplan Advanced Level RE@Agile – Practitioner/Specialist	LE	in diesem Buch
LE 1 **Was ist RE@Agile**	1	■ Abschnitt 1.2.3
LE 2 **Projekte erfolgreich starten**	2.1	■ Abschnitt 3.2.2
	2.2	■ Abschnitt 3.2.3
	2.3	■ Abschnitt 3.2.4
	2.4	■ Abschnitt 3.2.1
LE 3 **Umgang mit funktionalen Anforderungen**	3.1	■ Abschnitt 3.9, Einleitung
	3.2	■ Abschnitt 3.1.9, zweiter Teil
	3.3	■ Abschnitte 3.4.2 und 5.1.3
	3.4	■ Abschnitt 3.4.2, erster Unterabschnitt
	3.5	■ Abschnitt 3.4.2, zweiter Unterabschnitt
	3.6	■ Abschnitt 3.1.10
LE 4 **Umgang mit Qualitätsanforderungen und Randbedingungen**	4.1	■ Abschnitt 3.4.3
	4.2	■ Abschnitt 3.4.3.1
	4.3	■ Abschnitt 3.4.3.3
	4.4	■ Abschnitt 3.4.3.2
LE 5 **Priorisieren und Schätzen von Anforderungen**	5.1	■ Abschnitt 4.3
	5.2	■ Abschnitt 4.4, Einleitung
	5.3	■ Abschnitte 4.5.2 und 4.5.4, zweiter und dritter Unterabschnitt
LE 6 **Skalierung von RE@Agile**	6.1	■ Abschnitt 7.3.4
	6.2	■ Abschnitt 7.3.6
	6.3	■ Abschnitt 6.4.2
	6.4	■ Abschnitt 5.5
Begriffsdefinitionen, Glossar		■ Siehe auch [IREB Glossar] und [IREB RE@Agile Glossar]

Die für das Modul Advanced Level RE@Agile relevanten Kapitel und Stellen werden mit »[IREB Advanced Level RE@Agile – Practitioner/Specialist]« sowie ggf. mit der Ergänzung »LE xxx« gekennzeichnet und sind prüfungsrelevant. LE xxx referenziert auf die jeweilige Lerneinheit (LE = Lerneinheit) des Lehrplans. Bei allen anderen Unterkapiteln handelt es sich um ergänzende Informationen und Hinweise oder Anregungen für die Praxis.

Generell gilt für Verweise auf die IREB-Lehrpläne: Wenn der Verweis direkt unter einer Kapitel- oder Unterkapitelüberschrift steht, ist das ganze Kapitel oder Unterkapitel von Bedeutung. Wenn der Verweis im Fließtext steht, ist dies bis zum nächsten Verweis bzw. zur nächsten Kapitel- oder Unterkapitelüberschrift relevant.

1.1.4 Allgemeine Begriffseinordnung

[IREB RE@Agile] LE 1.3

Das umfangreiche Wissen in der Softwarebranche wird in unterschiedlichen Abstraktionsebenen beschrieben: [Meyer 2014] unterscheidet zwischen Prinzipien, Praktiken/Verfahren und Techniken. Diese Ebenen werden durch die Begriffe »präskriptiv« (vorschreibend), »abstrakt« und »falsifizierbar« (widerlegbar) unterschieden (siehe auch das Glossar im Anhang D des Buches).

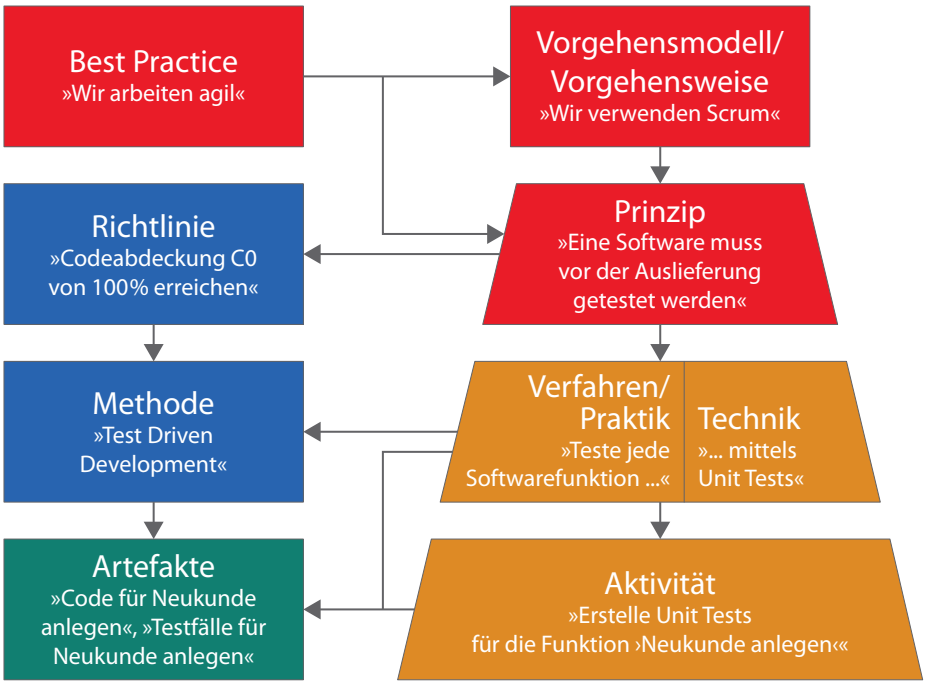


Abb. 1–1 *Abstraktionsebenen beim Wissen über die Softwareentwicklung*

- Ein »Prinzip« ist eine präskriptive Aussage, die abstrakt und falsifizierbar ist. Ein Prinzip ist z. B. »Prüfe die Requirements vor der Implementierung auf ausreichende Qualität«. Dies ist präskriptiv, da eine Aktion vorgegeben wird. Und es ist abstrakt, da keine konkreten Praktiken oder Techniken, die verwendet werden sollen, vorgegeben werden. Diesem Prinzip kann in bestimmten Situationen auch widersprochen werden. Zum Beispiel könnte es in einer Forschungssitua-

tion sinnvoll sein, ohne vorherige Requirements-Spezifikation mehrere Prototypen zu erstellen und diese gegeneinander zu validieren. Dieses Prinzip ist somit auch falsifizierbar. Die Kenntnis der Prinzipien ist wichtig, um bewusste Entscheidungen treffen zu können. Die Falsifizierbarkeit ist wichtig, um Diskussionen und Nachdenken über Prinzipien und auch Praktiken in Gang zu bringen und zu entscheiden, ob diese in einem bestimmten Kontext angewendet werden sollen.

- Ein »Verfahren« oder auch »Praktik« ist ebenfalls präskriptiv, jedoch nicht so abstrakt, sondern eine konkretere »Instanz« des Prinzips in einem bestimmten Kontext, die Hinweise auf die Umsetzung des Prinzips gibt, ohne jedoch die tatsächliche Durchführung einer Aktivität zu beschreiben. Zum Beispiel »Prüfe jedes Backlog Item gegen die ›Definition of Ready‹, bevor es in das Sprint Backlog eingefügt wird« ist eine allgemeine Praktik, die schon konkret bestimmte Artefakte (Backlog Items, DoR und Sprint Backlog) zur Verwendung anspricht (die Begriffe werden in den Kapiteln dieses Buches erläutert – siehe auch den Index im Anhang des Buches). Abhängig von der jeweiligen Situation und Kontext können auch jeweils unterschiedliche Praktiken zur Erfüllung eines Prinzips angewendet werden.
- Eine »Technik« ist eine Konkretisierung der Umsetzung des Verfahrens bzw. der Praktik wie z.B. »Teste durch die Verwendung von Unit Tests«.
- Eine »Aktivität« ist die reale Umsetzung eines Verfahrens/einer Praktik bzw. Technik wie z.B. »Erstelle die Unit Tests für die Funktion ›Neukunden anlegen‹«.

Sowohl der Lehrplan [IREB RE@Agile] als auch dieses Buch mit seinen erweiterten Ausführungen und Inhalten vermitteln Wissen von Requirements Engineering im Kontext der agilen Vorgehensweisen und stellen diese Abstraktionselemente in sachlicher Form vor bzw. regen zum Nachdenken und zur Diskussion über deren Anwendbarkeit an.

1.2 Verbindung zwischen RE und agilem Vorgehen

[IREB RE@Agile Primer] LE 1.1

Heute ist die Software und IT ein »Enabler« und Treiber in vielen Bereichen. Viele über lange Jahre etablierte plangetriebene Entwicklungsmethoden wurden nicht für ein sich schnell änderndes Umfeld ausgelegt. Agile Methoden (siehe auch Abschnitt 2.1) haben sich hier mittlerweile etabliert und schließen diese Lücke.

»Agile« oder »Agilität« wird in [Sheppard & Young 2006] wie folgt definiert: »A rapid whole-body movement with change of velocity or direction in response to a stimulus.«

Die Motivation für die Verwendung agiler Methoden ist hier definiert als Reaktion (z.B. eine schnelle Veränderung der Ausrichtung oder der Umsetzungsgeschwindigkeit) auf eine Stimulation (z.B. aus dem Projektumfeld).

Wenn man sich das Agile Manifest [Agile Manifesto] und die agilen Methodenbeschreibungen durchliest, dann geht es darin um mehr als nur Schnelligkeit. Im Grunde fokussieren alle agilen Prinzipien und Vorgehensweisen auf eine Lieferung von für die Kunden nützlichen Ergebnissen in kurzen, kontrollierten Intervallen.

Agile Methoden oder Agilität sind jedoch kein Selbstzweck. Nicht in allen Situationen passt dieses Vorgehen gleich gut. Wenn z. B. längerfristige Planung, Vorhersagbarkeit oder Nachvollziehbarkeit erforderlich ist, müssen ggf. andere Vorgehensweisen oder Anpassungen der agilen Methoden angewendet werden.

Es ist daher wichtig, im Vorfeld darüber nachzudenken und ein passendes bzw. angepasstes Entwicklungsvorgehen zu wählen. Dies kann grundsätzlich auch dazu führen, dass in unterschiedlichen Teams entsprechend dem jeweiligen Bedarf unterschiedliche Vorgehensweisen oder auch Kombinationen aus agilen und plangetriebenen Vorgehensweisen angewendet werden (z. B. wenn im Maschinenbau in der Hardwareentwicklung plangetriebenes Vorgehen und in der dazugehörenden Softwareentwicklung agiles Vorgehen in Kombination eingesetzt werden). Gartner hat dafür den Begriff »bimodale IT« geprägt. Dieses angepasste pragmatische Vorgehen ist auch einer der wesentlichen Erfolgsfaktoren der Softwareentwicklung und IT in dem heute sehr dynamischen Umfeld.

1.2.1 Denkweisen und Werte im RE und agilen Vorgehen

[IREB RE@Agile Primer] LE 1.2 und Einleitung

Requirements Engineering (RE)

In der IREB-Definition von Requirements Engineering sind Denkweisen und Werte von Requirements Engineering angegeben:

Requirements Engineering ist eine systematische und disziplinierte Herangehensweise zur Spezifikation und zum Management von Anforderungen mit den folgenden Zielen:

1. Die relevanten Anforderungen kennen, einen Konsens zwischen den Stakeholdern dieser Anforderungen erreichen, konform zu vorgegebenen Standards dokumentieren und die Anforderungen systematisch managen.
2. Die Wünsche und Bedürfnisse der Stakeholder verstehen und dokumentieren.
3. Die Anforderungen spezifizieren und managen, um ein System auszuliefern, das möglichst weitgehend den Wünschen und Bedürfnissen der Stakeholder entspricht.