
Inhaltsübersicht

1	Einleitung und Motivation	1
2	Grundlagen	29
3	Requirements-Ermittlung und -Dokumentation	57
4	Requirements-Validierung und -Abstimmung	201
5	Qualität im Requirements Engineering	239
6	Requirements Management	259
7	Organisatorische Aspekte	291
8	Requirements-Engineering-Rollen	313
9	Rechtliche Themen	327
Anhang		355
A	Agile Methoden zur Unterstützung des Requirements Engineering	357
B	Rollenbeschreibungen – Beispiele	375
C	Abkürzungen	385
D	Glossar	387
E	Literatur	395
	Index	401

Inhaltsverzeichnis

1	Einleitung und Motivation	1
1.1	Fokus dieses Buches	1
1.1.1	Zielgruppen	1
1.1.2	Abbildung des Lehrplans IREB CPRE RE@Agile Primer	2
1.1.3	Abbildung des Lehrplans IREB CPRE Advanced Level RE@Agile – Practitioner/Specialist	3
1.1.4	Allgemeine Begriffseinordnung	5
1.2	Verbindung zwischen RE und agilem Vorgehen	6
1.2.1	Denkweisen und Werte im RE und agilen Vorgehen	7
1.2.2	Zusammenhang zwischen RE und Agile	11
1.2.3	Was ist RE@Agile	14
1.2.4	RE im Kontext des Agilen Manifests	16
1.2.5	Nutzen von RE im agilen Vorgehen	21
1.2.6	Vorurteile und Probleme beim RE im agilen Umfeld	22
1.2.7	Fallstricke bei RE@Agile	24
1.2.8	Resümee	27
2	Grundlagen	29
2.1	Methodenüberblick	29
2.1.1	Allgemeine agile Vorgehensweisen	29
2.1.2	Scrum »in a Nutshell«	31
2.1.3	Methoden zur Unterstützung des Requirements Engineering	36
2.2	Requirements Engineering im agilen Umfeld	43
2.3	Grundprinzipien des RE in der agilen Softwareentwicklung ..	46
2.4	Umfang des Requirements Engineering	53

3	Requirements-Ermittlung und -Dokumentation	57
3.1	Ein kurzer Überblick	57
3.1.1	Anforderungsarten	58
3.1.2	Requirements-Dokumente vs. Product Backlog	58
3.1.3	Granularität funktionaler Requirements	60
3.1.4	Grafische Modelle und textuelle Beschreibungen	62
3.1.5	Definition von Begriffen, Glossare und Informationsmodelle	63
3.1.6	Akzeptanz- und Abnahmekriterien	64
3.1.7	Definition of Ready & Definition of Done	66
3.1.8	Prototyp vs. Inkremente	66
3.1.9	Ermittlung	67
3.1.10	Dokumentation	69
3.1.11	Artefakte	72
3.1.12	Ein Blick auf das große Ganze	74
3.2	Übergeordnete Artefakte	77
3.2.1	Zusammenhänge und Abhängigkeiten	77
3.2.2	Vision und Ziele	78
3.2.3	Systemgrenze und Kontext	86
3.2.4	Stakeholder	91
3.2.5	Epics	95
3.2.6	Personas	100
3.3	Geschäftsprozesse und Systemverhalten	102
3.3.1	Prozesse	102
3.3.2	Use Cases	110
3.3.3	Use-Case-Szenario bzw. -Template	116
3.4	Funktionale und nicht funktionale Sicht	125
3.4.1	Features	125
3.4.2	User Stories	127
	3.4.2.1 Schneiden, Aufteilen bzw. Gruppieren von User Stories	133
	3.4.2.2 Wann sollte man aufhören zu zerlegen?	139
	3.4.2.3 Nicht funktionale User Stories	140
	3.4.2.4 Technische User Stories	141
3.4.3	Qualitätsanforderungen und Randbedingungen	141
	3.4.3.1 Qualitätsanforderungen	143
	3.4.3.2 Randbedingungen (Constraints)	154
	3.4.3.3 Abnahme und Backlog-Management	158

3.5	Benutzerschnittstelle	160
3.5.1	Wireframes	164
3.5.2	Sketchy User Interface/Sketches	164
3.5.3	Finales User Interface	166
3.5.4	Szenariobasierte UI-Spezifikation	168
3.5.5	Hinweise zur GUI-Spezifikation	170
3.6	Systemschnittstelle	172
3.7	Prototypen und Inkremente	175
3.8	Entwicklersicht	177
3.8.1	Spikes	177
3.8.2	Architektur und technisches Design	180
3.8.3	Developer Story	185
3.8.4	System szenarien	188
3.8.5	Developer Constraints	190
3.8.6	Tasks	195
3.9	Inhaltliche Strukturierungshilfsmittel	197
3.9.1	Themes	198
3.9.2	Epics und Features	199
4	Requirements-Validierung und -Abstimmung	201
4.1	Verfeinerung von Anforderungen	203
4.1.1	Backlog Refinement	203
4.1.2	Refinement-Meeting	204
4.2	Machbarkeitsanalyse	206
4.2.1	Technische und funktionale Analyse mit Spikes ..	206
4.2.2	Organisatorische und personelle Machbarkeit ...	206
4.3	Ermitteln von Geschäftswert und Nutzen	207
4.3.1	Messung des Nutzens	208
4.3.2	Das Kano-Modell	209
4.3.3	Ordnung nach relativem Nutzen	210
4.3.4	Abstrakter Geschäftswert (Business Value)	210
4.3.5	MVP – Minimum Viable Product	211
4.3.6	MMP – Minimum Marketable Product	211
4.4	Risikobewertung	212
4.4.1	Risiken identifizieren und bewerten	214
4.4.2	Maßnahmen planen	217
4.4.3	Risiken überwachen und steuern	219

4.5	Aufwands- und Kostenschätzung	220
4.5.1	Aufwandsschätzung in nicht agilen Softwareprojekten	220
4.5.2	Prinzipien agiler Schätzungen	223
4.5.3	Schätzen im Projektverlauf	227
4.5.4	Schätztechniken	227
4.5.5	Ermitteln von Aufwand und Kosten aus Story Points	232
4.6	Bewertung der Qualität der Anforderungen	235
4.7	Priorisierung	235
4.7.1	Prioritätsskala	236
4.7.2	Basis für die Priorisierung	236
5	Qualität im Requirements Engineering	239
5.1	Qualitätskriterien für Requirements	240
5.1.1	Qualitätskriterien nach IEEE 830-1998 und IREB	240
5.1.1.1	Qualitätskriterien für einzelne Anforderungen	240
5.1.1.2	Qualitätskriterien für mehrere Anforderungen	244
5.1.2	DEEP-Qualitätskriterien	245
5.1.3	INVEST-Qualitätskriterien	246
5.2	Definition of Ready (DoR)	247
5.2.1	Definition of Ready für einen einzelnen Backlog-Eintrag	247
5.2.2	Definition of Ready für eine übergreifende Prüfung	249
5.3	Definition of Done (DoD)	250
5.4	Review von Requirements	256
5.5	Produktvalidierung	258
6	Requirements Management	259
6.1	Allgemeines	259
6.2	Inhalt vs. Management des Inhalts	260
6.3	Requirements-Management-Aktivitäten	263
6.4	Planende Aktivitäten des Requirements Management	263
6.4.1	Portfolio- und Programmplanung	264
6.4.2	Produkt-Roadmap, Delivery Roadmap	266
6.4.3	Produktplanung	268

6.4.4	Releaseplanung	271
6.4.5	Sprint-Planung	273
6.4.6	Daily Scrum	275
6.5	Artefakte für das Requirements Management	275
6.5.1	Backlog	275
6.5.2	Listen	278
6.5.3	Story Maps – Story Cards	279
6.5.4	Agiles Requirements-Board	281
6.5.5	Taskboard	286
7	Organisatorische Aspekte	291
7.1	Einfluss der Organisation	291
7.2	Agile Entwicklung im nicht agilen Umfeld	292
7.2.1	Interaktion mit Stakeholdern außerhalb der Softwareorganisation	292
7.2.2	Produkt- vs. Projektorganisation	293
7.2.3	Die Rolle des Managements im agilen Kontext ...	294
7.3	Skalierung	295
7.3.1	Motivation für die Skalierung	295
7.3.2	Ansätze für das Organisieren von Teams	296
7.3.3	Ansätze für das Organisieren der Kommunikation	297
7.3.4	Frameworks für das Skalieren von RE@Agile	299
7.3.5	Einfache skalierte Entwicklungsorganisation	302
7.3.6	Kriterien für die Strukturierung von Anforderungen und Teams im Großen	305
7.3.7	Auswirkungen der Skalierung auf RE@Agile	306
7.4	Vorab- und kontinuierliche Aufgaben des Requirements Engineering im Zusammenhang mit Skalierung	307
7.4.1	Initiale Requirements-Definition	308
7.4.2	Detaillierungsgrad für Backlog Items	309
7.4.3	Validität von Backlog-Einträgen	310
7.4.4	Feedback zum Backlog und dessen Aktualisierung	311
7.4.5	Zeitlicher Ablauf des Entwicklungszyklus	311

8	Requirements-Engineering-Rollen	313
8.1	Product Owner	314
8.1.1	Product Owner als Stellvertretung des Kunden im Team	314
8.1.2	Unterschiedliche Product-Owner- Verantwortlichkeiten im skalierten Umfeld	315
8.1.3	Schwierige Ausprägungen von Product Ownern ...	316
8.2	Chief Product Owner (CPO)	318
8.2.1	Der Chief Product Owner als Dirigent mehrerer Teams	318
8.2.2	Schwierige Ausprägungen des Chief Product Owners	319
8.3	Agile Entwickler	320
8.3.1	Die Entwickler als Umsetzer und Berater des Product Owners	320
8.3.2	Schwierige Ausprägungen bei den Entwicklern ...	321
8.4	Agile Master	322
8.4.1	Agile Master als Coach und Problemlöser	322
8.4.2	Schwierige Ausprägungen von Agile Masters	323
8.5	Tester	324
8.5.1	Der Tester als Prüfer und Qualitätsberater	324
8.5.2	Schwierige Ausprägungen von Testern	325
8.6	Architekt	325
8.6.1	Der Architekt als beratende Person für das Gesamtsystem	325
8.6.2	Schwierige Ausprägung beim Architekten	326
9	Rechtliche Themen	327
9.1	Allgemeine rechtliche Aspekte	328
9.2	Vertragsbasis und Vertragserfüllungspflicht	330
9.3	Gewährleistung	337
9.4	Agile Vorgehensweisen und Festpreis	339
9.5	Das Vier-Stufen-Modell für agile Festpreisprojekte	342
9.5.1	Stufe 1: Definition der Projektziele und ersten Kundenanforderungen	342
9.5.2	Stufe 2: Agiles Erstellen der Vertragsbasis	343
9.5.3	Stufe 3: Festpreisangebot durch den Lieferanten ..	344
9.5.4	Stufe 4: Agile Projektabwicklung	345

9.6	Öffentliche Ausschreibungen	346
9.7	Standards und Normen	348
9.8	Absicherung der Auftraggeberin	351
9.9	Absicherung des Lieferanten	352

Anhang	355
---------------	------------

A	Agile Methoden zur Unterstützung des Requirements Engineering	357
A.1	Specification by Example	357
A.2	Test Driven Development	362
A.3	Behavior Driven Development	367
B	Rollenbeschreibungen – Beispiele	375
B.1	Product Owner (PO)	375
B.2	Chief Product Owner (CPO)	377
B.3	Feature & Component Owner (FO, CO)	379
B.4	Proxy Product Owner (PPO)	382
C	Abkürzungen	385
D	Glossar	387
E	Literatur	395
	Index	401