

Inhaltsverzeichnis

Vorwort von Kent Beck	15
Über die Fachkorrektoren	16
Danksagungen	17
Einleitung	19
Teil I Der Einstieg	23
1 Ein Überblick	25
1.1 Was ist eine User Story?	26
1.2 Wo sind die Details?	27
1.3 »Wie viel muss ich schreiben?«	29
1.4 Das Kundenteam	30
1.5 Wie soll der Prozess aussehen?	30
1.6 Releases und Iterationen planen	32
1.7 Was sind Akzeptanztests?	34
1.8 Warum Veränderung?	35
1.9 Zusammenfassung	36
1.10 Fragen	37
2 User Stories schreiben	39
2.1 Independent (unabhängig)	39
2.2 Negotiable (verhandelbar)	40
2.3 Valuable (werthaltig) für Käufer oder User	42
2.4 Estimatable (schätzbar)	44
2.5 Small (klein)	45
2.5.1 Stories aufteilen	45
2.5.2 Stories kombinieren	48
2.6 Testable (testbar)	48
2.7 Zusammenfassung	49
2.8 Verantwortung des Entwicklers	50
2.9 Verantwortung des Kunden	50
2.10 Fragen	50

3	Modellieren von Nutzerrollen	51
3.1	Nutzerrollen	51
3.2	Schritt für Schritt zur Rollenmodellierung	53
3.2.1	Ideen für einen Initialsatz an Nutzerrollen sammeln	53
3.2.2	Initialsatz an Nutzerrollen ordnen	54
3.2.3	Nutzerrollen konsolidieren	55
3.2.4	Nutzerrollen verfeinern	56
3.3	Zwei weitere Methoden	57
3.3.1	Personas	58
3.3.2	Extreme Charaktere	59
3.4	Was, wenn ich lokale User habe?	59
3.5	Zusammenfassung	60
3.6	Verantwortung des Entwicklers	60
3.7	Verantwortung des Kunden	60
3.8	Fragen	61
4	User Stories sammeln	63
4.1	Datenerhebung sollte verboten sein	63
4.2	Ein bisschen reicht doch, oder nicht?	64
4.3	Methoden	65
4.4	Interviews mit Usern	65
4.4.1	Offene und kontextunabhängige Fragen	66
4.5	Fragebögen	67
4.6	Beobachtung	68
4.7	Story-Workshops	69
4.8	Zusammenfassung	72
4.9	Verantwortung des Entwicklers	73
4.10	Verantwortung des Kunden	73
4.11	Fragen	74
5	Arbeiten mit User Proxys	75
5.1	Manager	75
5.2	Entwicklungsleiter	77
5.3	Vertriebsleute	77
5.4	Domänenexperten	77
5.5	Marketinggruppe	79
5.6	Ehemalige User	79
5.7	Kunden	79
5.8	Schulungsleiter und technischer Support	80

5.9	Business oder Systemanalysten	81
5.10	Was beim Arbeiten mit einem User Proxy zu tun ist.....	81
5.10.1	Wenn es zwar User, aber nur begrenzten Zugang zu ihnen gibt	81
5.10.2	Wenn es wirklich keinen User gibt	82
5.11	Können Sie es selbst machen?	83
5.12	Das Kundenteam zusammenstellen	83
5.13	Zusammenfassung	84
5.14	Verantwortung des Entwicklers	85
5.15	Verantwortung des Kunden	85
5.16	Fragen.....	85
6	User Stories für Akzeptanztests.....	87
6.1	Erst Tests schreiben, dann programmieren	88
6.2	Der Kunde legt die Tests fest	89
6.3	Testen ist Teil des Prozesses	89
6.4	Wie viele Tests sind zu viele?	90
6.5	Das Framework for Integrated Test (FIT)	90
6.6	Testmethoden	91
6.7	Zusammenfassung	92
6.8	Verantwortung des Entwicklers	93
6.9	Verantwortung des Kunden	93
6.10	Fragen.....	93
7	Leitfaden für gute User Stories	95
7.1	Mit Ziel-Stories beginnen.....	95
7.2	Den Kuchen in Stücke schneiden	95
7.3	Geschlossene Stories schreiben.....	96
7.4	Beschränkungen auf Story Cards festhalten	97
7.5	Story an den geplanten Zeitraum anpassen	98
7.6	UI so lange wie möglich heraushalten	99
7.7	Manche Dinge sind keine Stories	100
7.8	Nutzerrollen in Stories einbinden	100
7.9	Für einen einzelnen User schreiben	101
7.10	Im Aktiv schreiben	101
7.11	Kunden schreiben lassen	101
7.12	Story Cards nicht nummerieren.....	101
7.13	Nicht den Zweck vergessen	102
7.14	Zusammenfassung	102
7.15	Fragen.....	103

Teil II	Schätzen und Planen	105
8	User Stories schätzen	107
8.1	Story Points	107
8.2	Schätzen im Team	108
8.3	Schätzen	108
8.4	Triangulieren	110
8.5	Mit Story Points arbeiten	111
8.6	Was ist mit Pair Programming?	112
8.7	Einige Gedächtnisstützen	113
8.8	Zusammenfassung	114
8.9	Verantwortung des Entwicklers	114
8.10	Verantwortung des Kunden	115
8.11	Fragen	115
9	Ein Release planen	117
9.1	Wann wollen wir das Release?	117
9.2	Was hätten Sie denn gerne?	118
9.3	User Stories priorisieren	119
9.4	Gemischte Prioritäten	120
9.5	Riskante User Stories	120
9.6	Infrastrukturelle Anforderungen priorisieren	121
9.7	Iterationslänge auswählen	122
9.8	Von Story Points zu voraussichtlicher Dauer	123
9.9	Die anfängliche Velocity	123
9.9.1	Velocity erraten	124
9.10	Releasepläne erstellen	124
9.11	Zusammenfassung	125
9.12	Verantwortung des Entwicklers	126
9.13	Verantwortung des Kunden	126
9.14	Fragen	126
10	Iterationen planen	127
10.1	Übersicht über die Iterationsplanung	127
10.2	Stories besprechen	127
10.3	In Tasks zerlegen	128
10.3.1	Leitlinien	130
10.4	Verantwortung übernehmen	130
10.5	Schätzen und Bestätigen	131

10.6	Zusammenfassung	132
10.7	Verantwortung des Entwicklers	133
10.8	Verantwortung des Kunden	133
10.9	Fragen.	133
11	Velocity messen und überwachen	135
11.1	Velocity messen	135
11.2	Soll- und Ist-Velocity	137
11.3	Iteration Burndown Charts.	138
11.4	Burndown Charts während einer Iteration	141
11.5	Zusammenfassung	143
11.6	Verantwortung des Entwicklers	143
11.7	Verantwortung des Kunden	144
11.8	Fragen.	144
Teil III Häufig diskutierte Themen	147
12	Was User Stories nicht sind	149
12.1	User Stories sind nicht IEEE 830.	149
12.2	User Stories sind keine Use Cases.	153
12.3	User Stories sind keine Szenarios	157
12.4	Zusammenfassung	159
12.5	Fragen.	160
13	Warum User Stories?	161
13.1	Verbale Kommunikation.	161
13.2	User Stories sind verständlich	164
13.3	User Stories haben die richtige Größe für die Planung.	164
13.4	User Stories eignen sich für iterative Softwareentwicklung	165
13.5	User Stories ermutigen dazu, Details zurückzustellen	166
13.6	User Stories unterstützen opportunistische Softwareentwicklung	167
13.7	User Stories unterstützen partizipatorisches Design	168
13.8	User Stories bauen implizites Wissen auf.	169
13.9	Warum keine User Stories?	169
13.10	Zusammenfassung	170
13.11	Verantwortung des Entwicklers	171
13.12	Verantwortung des Kunden	171
13.13	Fragen.	171

14	Mängelliste für User Stories	173
14.1	User Stories sind zu klein	173
14.2	Voneinander abhängige User Stories	173
14.3	Goldplating	174
14.4	Zu viele Details	175
14.5	Zu früh Details zur Benutzeroberfläche aufnehmen	175
14.6	Zu weit vorausdenken	176
14.7	Ständiges Aufteilen von Stories	176
14.8	Kunden haben Probleme mit dem Priorisieren	177
14.9	Kunden wollen Stories weder schreiben, noch priorisieren	178
14.10	Zusammenfassung	178
14.11	Verantwortung des Entwicklers	179
14.12	Verantwortung des Kunden	179
14.13	Fragen	179
15	User Stories und Scrum	181
15.1	Scrum ist iterativ und inkrementell	181
15.2	Scrum – Die Grundlagen	182
15.3	Das Scrum-Team	182
15.4	Der Product Backlog	183
15.5	Das Sprint Planning Meeting	184
15.6	Das Sprint Review Meeting	186
15.7	Das Daily Scrum Meeting	187
15.8	Stories in Scrum einfügen	189
15.8.1	User Stories und der Product Backlog	189
15.8.2	User Stories im Sprint Planning Meeting	189
15.8.3	User Stories im Sprint Review Meeting	190
15.8.4	User Stories und das Daily Scrum Meeting	190
15.9	Eine Fallstudie	190
15.10	Zusammenfassung	192
15.11	Fragen	192
16	Weitere Themen	193
16.1	Nichtfunktionale Anforderungen behandeln	193
16.2	Papier oder Software?	195
16.3	User Stories und die Benutzerschnittstelle	197
16.4	User Stories aufbewahren	200
16.5	User Stories für Fehlerberichte	201
16.6	Zusammenfassung	202
16.7	Verantwortung des Entwicklers	202

16.8	Verantwortung des Kunden	203
16.9	Fragen	203
<hr/>		
Teil IV Ein Beispiel		205
<hr/>		
17	Die Nutzerrollen	207
17.1	Das Projekt	207
17.2	Wer ist der Kunde?	207
17.3	Erste Nutzerrollen festlegen	208
17.4	Konsolidieren und eingrenzen	208
17.5	Rollenmodellierung	211
17.6	Personas hinzufügen	213
18	Die User Stories	215
18.1	User Stories für Teresa	215
18.2	User Stories für Kapitän Müller	218
18.3	User Stories für einen Segelanfänger	219
18.4	User Stories für nicht segelnde Geschenkeinkäufer	219
18.5	User Stories für einen Nutzungsstatistik-Beobachter	220
18.6	Einige User Stories zum Administrator	220
18.7	Zusammenfassung	221
19	User Stories schätzen	223
19.1	Die erste Story	224
19.2	Erweiterte Suche	226
19.3	Buchbewertung und -besprechung	227
19.4	Konten	228
19.5	Schätzungen beenden	229
19.6	Alle Schätzungen auf einen Blick	230
20	Der Releaseplan	233
20.1	Velocity schätzen	233
20.2	User Stories priorisieren	234
20.3	Der fertige Releaseplan	235
21	Die Akzeptanztests	237
21.1	Tests für die Suche	237
21.2	Tests für den Warenkorb	238
21.3	Bücher kaufen	239
21.4	Kundenkonten	240

21.5	Verwaltung	241
21.6	Beschränkungen testen	242
21.7	Eine letzte Story	242
 Teil V Anhänge		 245
A	Extreme Programming – Ein Überblick	247
A.1	Nutzerrollen	247
A.2	Die zwölf Techniken	248
A.2.1	Kurze Releasezyklen	249
A.2.2	Das Planning Game	249
A.2.3	Refactoring	250
A.2.4	Testen	250
A.2.5	Pair Programming	251
A.2.6	Nachhaltiges Tempo	252
A.2.7	Collective Ownership	252
A.2.8	Coding-Standards	253
A.2.9	Einfaches Design	253
A.2.10	Metapher	253
A.2.11	Continuous Integration	254
A.2.12	Kunde vor Ort	254
A.3	Die XP-Werte	254
A.4	Die XP-Prinzipien	255
A.5	Zusammenfassung	256
B	Antworten auf die Fragen	257
B.1	Kapitel 1 – Ein Überblick	257
B.2	Kapitel 2 – User Stories schreiben	258
B.3	Kapitel 3 – Modellieren von Nutzerrollen	259
B.4	Kapitel 4 – User Stories sammeln	260
B.5	Kapitel 5 – Arbeiten mit User Proxys	261
B.6	Kapitel 6 – User Stories für Akzeptanztests	261
B.7	Kapitel 7 – Leitfaden für gute User Stories	262
B.8	Kapitel 8 – User Stories schätzen	263
B.9	Kapitel 9 – Ein Release planen	264
B.10	Kapitel 10 – Iterationen planen	264
B.11	Kapitel 11 – Velocity messen und überwachen	264
B.12	Kapitel 12 – Was User Stories nicht sind	266

B.13	Kapitel 13 – Warum User Stories?	267
B.14	Kapitel 14 – Mängelliste für User Stories	267
B.15	Kapitel 15 – User Stories und Scrum	268
B.16	Kapitel 16 – Weitere Themen	269
C	Literaturverzeichnis	271
C.1	Bücher und Artikel	271
C.2	Englischsprachige Websites	274
	Stichwortverzeichnis	275