

Inhaltsverzeichnis

Vorbemerkungen	13
Executive Summary	14
Navigation durch die Kapitel	17
1 Gewinne statt Business-Esoterik	19
1.1. Agile Wolkenschlösser	19
1.2. Führungskraft Scrum Master	20
1.3. High Performer für Wachstum	20
1.4. Triple-A: Automation, Agility, Acting	21
2 Die Essenz der Agilität	23
2.1. Agile Beschleuniger	23
2.1.1. Offen für die Zukunft	23
2.1.2. Phasen	24
2.1.3. Iteration	25
2.1.4. Zyklus	26
2.1.5. Takt	28
2.1.6. Feedback	28
2.1.7. Shift Left	29
2.1.8. Agiles Manifest	31
2.2. Scrum	33
2.2.1. Taskforce	33
2.2.2. Captainless	34
2.2.3. Events und Artefakte	36
2.2.4. Definition of Done	37
2.3. SAFe	40
2.4. Kanban	41
2.5. Frameworks	43
2.5.1. Frameworks für Teams	43
2.5.2. Strukturierte Skalierungsframeworks	44
2.5.3. Flexible Baukasten-Frameworks	45

2.6. Continuous Value	46
2.6.1. Wert schaffen	46
2.6.2. Produkt-Ziel	48
2.6.3. Sprint-Ziel	51
2.6.4. Teamwert	54
3 Automation First	55
3.1. Automatisierung ist die Basis	55
3.1.1. Run and Change	56
3.1.2. Permanenter Wandel	57
3.1.3. Kanban und Scrum	57
3.1.4. Von DevOps zu Opsless	58
3.1.5. Das Prinzip der Containerisierung	60
3.1.6. ContinuousX	60
3.1.7. Testabdeckung	63
3.2. Ingenieurskunst	65
3.2.1. Menschen und Werkzeuge	65
3.2.2. Produkt und Service	66
3.3. Wissensorganisation	67
3.3.1. Technical Backlog	67
3.3.2. Expert Hub	68
3.4. Eingebettete Einführung der Automatisierung	70
4 Stärke durch Qualität	73
4.1. Zero Tolerance	74
4.1.1. Das magische Dreieck lösen	74
4.1.2. Zero Bug	75
4.1.3. Zero Downtime	76
4.1.4. Integrationsumgebung	77
4.1.5. Der Dreiklang der Qualitätssicherung	78
4.2. Routineprozesse	79
4.2.1. Hindernisse beseitigen	79
4.2.2. Report to Action	81
4.2.3. Best Practices	83
4.2.4. Routinen und Checklisten	85
4.2.5. On- und Offboarding	86
4.2.6. Robustheit durch Routine	87
4.3. Performance Management	87

4.3.1. Die drei Säulen	88
4.3.2. Software Delivery Manager	88
4.3.3. Performance Excellence Center	89
5 Agilität 2.0	91
5.1. Arbeitsteilung	91
5.1.1. Kognitive Entlastung	92
5.1.2. Organisation ist Kommunikation	93
5.1.3. Handover	94
5.2. Skalierung	95
5.2.1. Skalierung durch Zeit	95
5.2.2. Skalierung durch Teambildung	96
5.2.3. Skalierung durch mehrere Teams	96
5.2.4. Feature versus Komponente	97
5.2.5. Software skalieren	99
5.3. Dynamische Zusammenarbeit	99
5.3.1. Kooperation und Koordination	100
5.3.2. Auftrag statt Dirigismus	101
5.3.3. Planwirtschaft versus Marktwirtschaft	102
5.3.4. Zusammenarbeit statt Autarkie	102
5.3.5. Kapselung statt Silo	104
5.3.6. Hierarchie und Entscheidung	105
5.3.7. Struktur und Dynamik	106
5.4. Software-Architektur für Menschen	108
5.4.1. Architektur strukturiert Anforderungen	108
5.4.2. Denkmuster prägen Architekturen	108
5.4.3. Architektur und Zusammenarbeit	109
5.4.4. Team Topologies	110
5.4.5. Integration und Inklusion	112
5.4.6. Standard und Vielfalt	113
5.4.7. Zentral versus Dezentral	114
5.4.8. Abhängigkeiten schaffen Werte	114
6 Erfolgsfaktor Projekt	119
6.1. Alles ist ein Projekt	119
6.1.1. Projekt und Produkt	119
6.1.2. Project Coordinator	121
6.1.3. Wandel durch Projekte	124

6.1.4.	Liquid Organisation	125
6.2.	Budget gibt es nicht geschenkt	126
6.2.1.	Verantwortlichkeiten	127
6.2.2.	Akquisition	127
6.2.3.	Kalkulation	128
7	Der pragmatische Scrum Master	131
7.1.	Zwei Seiten eines Teams	131
7.1.1.	Die helle Seite	132
7.1.2.	Die dunkle Seite	132
7.2.	Scrum Master Routinen	133
7.2.1.	Tägliche Tickethygiene	133
7.2.2.	Meetings entlasten	134
7.2.3.	Kapazitätsplanung	135
7.2.4.	Embedded HR	136
7.2.5.	Stakeholder	137
7.2.6.	Vertretungsregelungen	137
7.2.7.	Servant Leadership	138
7.3.	Product Owner lotsen	139
7.3.1.	Drei Sätze	139
7.3.2.	Das Produkt verstehen	139
7.3.3.	Mikromanagement	141
7.4.	Developer ermächtigen	142
7.4.1.	Overall Code Control	142
7.4.2.	Selbstmanagement	143
7.5.	Stakeholder managen	144
7.6.	Das Set-up	145
7.6.1.	Das Toolset	146
7.6.2.	Das Skillset	146
7.6.3.	Das Mindset	147
7.6.4.	Das richtige Mindset	148
7.7.	Eingebettete Einführung von Scrum	149
8	Von der Last zur Leistung	153
8.1.	Inkasso technischer Schulden	153
8.1.1.	Softwarefehler	156
8.1.2.	Architekturschulden	158
8.1.3.	Arbeitsschulden	160

8.1.4.	Einwandbehandlung	161
8.1.5.	Schuldenmanagement	162
8.1.6.	Technische Retrospective	163
8.1.7.	Geplante Schulden	164
8.2.	Mit Routine Krisen managen	166
8.2.1.	Eskalation und Skill Pump	167
8.2.2.	Vor die Lage kommen	167
8.2.3.	Cutover	168
8.2.4.	Konflikte und Schwächen	169
9	Acting Organisation	171
9.1.	Design und Disruption	173
9.1.1.	Der innere Wert	173
9.1.2.	Design driven Innovation	174
9.1.3.	Die Disruption	176
9.2.	Raum für Innovation	178
9.2.1.	Backfeed	178
9.2.2.	Sprint Perspective und Sprint Pitch	179
9.2.3.	SAFe Innovation Iteration	180
9.2.4.	Icebreaker Event	181
9.2.5.	Skill Building	182
9.2.6.	Psychologische Unsicherheit	183
9.3.	Innovation generieren	184
9.3.1.	Kreativität	184
9.3.2.	Innovation Backlog	185
9.3.3.	Continuous Exploration	185
9.3.4.	Experiment-driven Development	186
9.3.5.	ProgressiveX	187
9.3.6.	Märkte formen	188
9.4.	Agierende Methoden	190
9.4.1.	Erfolg	190
9.4.2.	Passgenauigkeit	191
9.4.3.	Beschleunigung	191
9.4.4.	Shift Ahead	191
9.4.5.	Extreme Thinking	192
9.4.6.	Entscheiden	192
9.4.7.	Vereinfachen	195
9.4.8.	Descaling	195

9.5.	Team Building	198
9.5.1.	Tandem Worker	198
9.5.2.	Quarterback Team	199
9.5.3.	Product Founder	200
9.6.	Acting Scrum	201
9.6.1.	Definition	201
9.6.2.	Artefakte	202
9.6.3.	Events	202
9.7.	Eingebettete Einführung von Acting	203
10	Triple-A	205
10.1.	Der Kontext	205
10.2.	Das Framework	206
10.2.1.	Automatisierung	207
10.2.2.	Agilität	207
10.2.3.	Agieren	208
10.3.	Die Aspekte	208
10.4.	Schlüsselindikatoren	209
10.4.1.	Indikatoren für Automatisierung	210
10.4.2.	Indikatoren für Agilität	211
10.4.3.	Indikatoren für Acting	212
10.4.4.	Technische Bilanz	212
10.5.	Die Aufstellung	214
10.6.	Die Dynamik	214
Epilog		216
Autor		218