Inhalt

1 Einleitung 1
1.1 Ziele des Buches 2
1.2 Orientierungshilfe 2
1.2.1 Schritt 1: „Grundlagen“ 3
1.2.2 Schritt 2: „Wirkung“ 3
1.2.3 Schritt 3: „Anwendung“ 4
1.3 Leseanleitung 4

2 Einführung in das Risikomanagement 7
2.1 Was ist Risikomanagement? 8
2.1.1 Begriffe des Risikomanagements 8
2.1.2 Exkurs „Risiko ist weder gut noch schlecht!“ 12
2.1.3 Risikomanagementsystem 13
2.1.4 Exkurs: Gründe nicht Alarm zu schlagen 18
2.1.5 Risikomanagement-Techniken 20
2.2 Risikomanagement in der Softwareentwicklung 31
2.2.1 Ist-Situation der Softwareentwicklung in Deutschland 32
2.2.2 Probleme von Softwareprojekten 44
2.2.3 Exkurs: Die 5 Stufen der Unwissenheit 48
2.2.4 Probleme in der Umsetzung des Risikomanagements 51

3 Prinzipien wirkungsvollen Risikomanagements 57
3.1 Die Mission eines wirkungsvollen Risikomanagements 58
3.2 Ausblick auf die weiteren Kapitel 59
3.3 Der Risikomanagementkontext 60
3.4 10 Thesen für wirkungsvolles Risikomanagement 60
3.4.1 These 1: „Verteilen Sie Risikobewusstsein und Risikoverantwortung!“ 61
3.4.2 These 2: „Betreiben Sie Projektmanagement als Risikomanagement und umgekehrt!“ 67

3.4.3 These 3: „Fördern Sie eine offene und professionelle Risikokommunikation und -kultur!“ ... 71
3.4.4 These 4: „Lernen Sie von anderen!“ ... 80
3.4.5 These 5: „Passen Sie den Risikomanagement-prozess an und benutzen Sie ihn!“ ... 83
3.4.6 These 6: „Integrieren Sie den Risikomanage-mentprozess in Ihre vorhandenen Prozesse!“ ... 86
3.4.7 These 7: „Nutzen Sie bewusst das Prinzip der Evolution für sich!“ ... 87
3.4.8 These 8: „Lernen Sie schätzen!“ ... 90
3.4.9 These 9: „Überblicken Sie die Wirtschaftlichkeit!“ ... 95
3.4.10 These 10: „Risikomanagement handelt von den Menschen!“ ... 99

4 Ein Risikomanagementsystem nach Art des Hauses ... 103
4.1 Vor Projektbeginn: Projektbewertung mit Hilfe des Risikomanagements ... 104
4.2 Die Risikomanagementorganisation ... 107
4.2.1 Die Ausgangbasis definieren ... 108
4.2.2 Die organisatorischen Rahmenbedingungen festlegen ... 109
4.3 Der Risikomanagementprozess ... 111
4.3.1 Risikoidentifikation ... 113
4.3.2 Der Risikoprozess ... 127
4.3.3 Monitoring und der kontinuierliche Verbesserungsprozess ... 145
4.3.4 Der Projektrückblick am Ende des Projektes ... 148
4.4 Risikokommunikation und -kultur ... 149
4.5 Fazit – Starten Sie durch! ... 150

5 Integration des Risikomanagements in Vorgehensmodelle der Softwareentwicklung ... 153
5.1 Risikomanagement beginnt nicht auf der grünen Wiese ... 153
5.2 Checkliste: Wie integrieren Sie Risikomanagement und Vorgehensmodell? ... 154
5.2.1 Schritt 1: Bestimmung der Ist-Situation ... 154
5.2.2 Schritt 2: Beschreibung der Soll-Situation ... 155
5.2.3 Schritt 3: Veränderung vom Ist zum Soll ... 157

5.3 Integration des Risikomanagements in verbreitete Vorgehensmodelle ... 158
5.3.1 Das Wasserfallmodell ... 158
5.3.2 Rational Unified Process (RUP) ... 161
5.3.3 V-Modell XT ... 163
5.3.4 Exkurs: Gemeinsames Risikomanagement – Risikomanagement und die Auftraggeber-Auftragnehmer-Beziehung ... 166
5.3.5 Extreme Programming (XP) und Agile Methoden ... 169
5.3.6 Risikomanagement und Kleinprojekte ... 171
5.4 Fazit ... 175

6 Risiken beschreiben und das Risikolexikon einsetzen ... 177
6.1 Die Risikoschablone des Risikolexikons ... 177
6.1.1 Risikoname ... 178
6.1.2 Kategorie ... 178
6.1.3 Projektphasen ... 179
6.1.4 Risikosatz und Risikobeschreibung ... 179
6.1.5 Indikatoren ... 180
6.1.6 Risikomatrix und Auswirkungen im magischen Viereck ... 180
6.1.7 Maßnahmen ... 182
6.1.8 Beziehungen: Vorgänger und Nachfolger ... 183
6.1.9 Rollen: Entdecker und Überwacher ... 183
6.2 Erweiterungen der Risikoschablone für den Einsatz im Risikomanagementprozess ... 184
6.3 Was bleibt ... 185

7 Risikolexikon ... 187
7.1 Übersicht ... 188
7.2 Phase: Planung ... 191
7.2.1 Unklare Zielvorstellung ... 191
7.2.2 Beschaffung Falscher Software für Anwender ... 193
7.3 Phase: Analyse ... 195
7.3.1 Hohe Systemkomplexität ... 195
7.3.2 Ungenügende Benutzerbeteiligung ... 197
7.3.3 Mangelhaftes Design der Gebrauchstauglichkeit (Usability) ... 199
7.3.4 Mangelhafte Anforderungsqualität ... 201
7.3.5 Mangelhafte Definition der funktionalen Anforderungen ... 204

7.3.6 Mangelhafte Definition der nichtfunktionalen Anforderungen/Qualitätsanforderungen 206
7.3.7 Steigender Anforderungsumfang (Anforderungsexplosion) 209
7.3.8 Verändernde Anforderungen..... 210
7.3.9 Second-System-Effekt..... 214
7.3.10 Verwendung von Legacy-Software 216
7.3.11 Datenschutzrisiko 218
7.4 Phase: Design..... 220
7.4.1 Mangelhaftes Domänenmodell 220
7.4.2 Erzwungener Architekturwechsel..... 222
7.4.3 Ungeeigneter Softwareentwurf 224
7.4.4 Mangelhafte Internationalisierung 226
7.4.5 Implementierung startet ohne Design 228
7.4.6 Einsatz neuer Technologien 230
7.5 Phase: Implementierung 233
7.5.1 Softwareerosion über die Zeit 233
7.5.2 Entwicklung unnötiger Funktionen 236
7.5.3 Mangelhafte Schnittstellen zwischen Anwendungen..... 238
7.5.4 Ungenügende Einhaltung der Programmierrichtlinien 241
7.5.5 Entwickler vernachlässigen Test..... 243
7.5.6 Komplexe Datenmigration..... 246
7.5.7 Mängel in Echtzeitverhalten 248
7.6 Phase: Test 250
7.6.1 Nichtausreichende Testinfrastruktur..... 250
7.6.2 Ungenügende Testmentalität/-motivation 252
7.6.3 Nichtausreichendes Testmanagement und Testreporting 255
7.7 Phase: Auslieferung..... 258
7.7.1 Fehlende Benutzerakzeptanz..... 258
7.7.2 Starkes Schattensystem vorhanden..... 260
7.7.3 Mangelhafte Umgebung für das System 262
7.7.4 Mangelhafte Definition des Abnahmeverfahrens..... 264
7.7.5 Unausgereifte Auslieferungsstrategie 266
7.7.6 Mangelhafte Anwenderschulung 269
7.8 Übergreifende Risiken: Prozesse und Projektmanagement 271
7.8.1 Mangelhaftes Betriebskonzept..... 271
7.8.2 Mangelhafte Projektkontrolle 273
7.8.3 Mangelhafte Vertragsgestaltung 275
7.8.4 Unrealistische Zeitplanung..... 278

7.8.5 Unrealistisches Budget ... 281
7.8.6 Projektleiterausfall/Projektleiterwechsel ... 283
7.8.7 Unzureichende Aufgabenidentifikation ... 285
7.8.8 Kein eindeutig definiertes Vorgehensmodell ... 288
7.8.9 Mangelhafte Definition von Qualitätsmerkmalen im Vorgehensmodell ... 290
7.8.10 Keine Nutzung/Akzeptanz des Vorgehensmodells ... 292
7.8.11 Unzureichende Reviewzyklen ... 295
7.8.12 Mangelhafter Werkzeugeinsatz oder -verfügbarkeit ... 297
7.8.13 Mangelnde Hardware- und Ressourcenverfügbarkeit ... 300
7.8.14 Unzureichendes Konfigurationsmanagement ... 302
7.8.15 Nicht geeignete Neueinführung eines Werkzeugs ... 304
7.8.16 Mangelhaftes Sicherheitskonzept ... 306
7.8.17 Ungenügende Akzeptanz Projektleiter/Mitarbeiter ... 308
7.8.18 Unzureichende Einarbeitung neuer Mitarbeiter ... 311
7.8.19 Technologischer Fortschritt zur Projektlaufzeit ... 313
7.9 Übergreifende Risiken: Organisation und Politik ... 315
7.9.1 Machtkämpfe ... 315
7.9.2 Kommunikationsrisiko außerhalb des Projektes ... 318
7.9.3 Unzureichende Unterstützung der Geschäftsleitung ... 321
7.9.4 Unzureichende Unterstützung durch Auftraggeber ... 323
7.9.5 Budgetproblematik ... 325
7.9.6 Mangelnde Mitarbeiterverfügbarkeit ... 327
7.9.7 Unrealistische Zielvorstellung ... 330
7.9.8 Veränderliche Projektabgrenzung/-ziele ... 333
7.10 Übergreifende Risiken: Mensch ... 336
7.10.1 Sprachbarrieren ... 336
7.10.2 Kommunikationsrisiko innerhalb des Projektes ... 338
7.10.3 Mangelhaftes Teamwork ... 340
7.10.4 Unpassendes Know-how der Mitarbeiter ... 343
7.10.5 Mitarbeiterfluktuation ... 345
7.10.6 Ungenügende Erfahrung des Projektleiters ... 347
7.10.7 Unmotivierte Mitarbeiter ... 350

7.11 Übergreifende Risiken: Externe Produkte und Lieferanten 353
7.11.1 Abhängigkeiten vom Zulieferer 353
7.11.2 Mängel in zugelieferten Produkten/Komponenten 355
7.11.3 Mangelnde Unterstützung durch Produkt-/Komponentenanbieter 357
7.11.4 Mängel in externen Dienstleistungen 358
7.12 Verwendete Literatur im Risikolexikon 360

Literatur 363

Über die Autoren 367

Sachverzeichnis 369

http://www.springer.com/978-3-540-30024-3

IT-Risikomanagement leben!
Wirkungsvolle Umsetzung für Projekte in der Softwareentwicklung
Ahrendts, F.; Marton, A.
2008, XVII, 371 S., Hardcover
ISBN: 978-3-540-30024-3