MARIO NEUMANN

PROJEKT SAFARI 2

Das Handbuch für AGILES Projektmanagement

campus

Mario Neumann

PROJEKT-SAFARI 2

Das Handbuch für agiles Projektmanagement

Campus Verlag
Frankfurt/New York

ISBN 978-3-593-51684-4 Print
ISBN 978-3-593-45337-8 E-Book (PDF)
ISBN 978-3-593-45594-5 E-Book (EPUB)

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt.
Jede Verwertung ist ohne Zustimmung des Verlags unzulässig. Das gilt
insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen
und die Einspeicherung und Verarbeitung in elektronischen Systemen.
Trotz sorgfältiger inhaltlicher Kontrolle übernehmen wir keine Haftung
für die Inhalte externer Links. Für den Inhalt der verlinkten Seiten sind
ausschließlich deren Betreiber verantwortlich.
Copyright © 2023. Alle Rechte bei Campus Verlag GmbH, Frankfurt am Main.
Umschlaggestaltung: Haus am Meer
Umschlagmotiv: © istockphoto.com/Jafar Setiawan
Illustrationen im Innenteil: © Natalie Neumann
Satz: Publikations Atelier, Dreieich
Gesetzt aus der Sabon Next und der Motiva Sans
Druck und Bindung: Beltz Grafische Betriebe GmbH, Bad Langensalza
Beltz Grafische Betriebe ist ein klimaneutrales Unternehmen (ID 15985-2104-1001).
Printed in Germany
www.campus.de

Inhalt

Vorwort 10
Einleitung 11

Und plötzlich steht alles Kopf 13
Die geänderte Sicht auf die Welt der Projekte
Zwei zunehmend unversöhnliche Lager 14
Zielorientiert? Oder doch zweckorientiert? 15
Die Umkehr des magischen Dreiecks 15
Das Beste aus zwei Welten 17

Ist Scrum etwas für uns? 22
Annäherung an eine neue Idee
Selbstorganisation statt fester Strukturen 23
Neue Abläufe – so einfach ist Scrum 24
Die fehlende Einbindung des Kunden 25
Scrum ist leicht, aber nicht einfach 26

ETAPPE 1
Aufbruch ins Ungewisse 31

1.1 Klassisch oder agil? 33
Die Wahl der richtigen Methode
Es muss nicht immer der Wasserfall sein 34
Die Komplexität bleibt – in jedem Fall 35
Kompliziert oder komplex – ein wichtiges Entscheidungskriterium 36
Die Stacey-Matrix – eine bewährte Entscheidungshilfe 37

1.2 Jede Reise braucht ein Ziel 42
Auftragsklärung – wie man für Klarheit sorgt
Die Sehnsucht nach dem weiten, endlosen Meer 42
Die Formulierung der Produktvision 43
Der Projektleiter als Aufklärer 44
Das Product-Vision-Board 45

1.3 Kurs bestimmen und Segel setzen 50
Gut vorbereitet ins Projektabenteuer starten
Nicht voreilig starten! 51
Farbe bekennen – Zahlen auf den Tisch! 51
Der Projektauftrag – das Fundament für die Projektarbeit 52
Die Projektstrategie – die Magie der Five Bold Steps 53
Schritt 1: Rundumblick – den Kontext erfassen 54
Schritt 2: Zukunftsbild – die Vision entwerfen 54
Schritt 3: Strategie – die Hauptschritte bestimmen 55

1.4 Den Schiffbruch verhindern 60
Wie man Schwierigkeiten von vorneherein vermeidet
Warnsignal 1: Falsche Erwartungshaltungen 60
Warnsignal 2: Komplexe Anforderungen 62
Warnsignal 3: Mangelnde Professionalität 63

ETAPPE 2
Große Ziele, kleine Schritte 69

2.1 Das Abenteuer solide beginnen 71
Gerade agile Projekte brauchen ein stabiles Fundament
Die Tücken der Anforderungsanalyse 72
Der Product Owner als Architekt 73
Das Fundament absichern 73
Eine Architekturskizze für agile Projekte 74
Die vier Säulen einer Lösungsarchitektur 75
Mit der Lösung am Ziel vorbei 78

2.2 Der Heilige Gral 82
Konzeption und Handhabung des Product Backlogs
Inhalt und Konzept des Product Backlog 83
Initiativen, Epics und User Storys 84
Die Struktur des Product Backlog 85

2.3 Alles unter Kontrolle 92
Die kontinuierliche Arbeit mit dem Product Backlog
Fünf typische Anti-Patterns 93
Die Kontrolle über das Product Backlog 94
Komplexität beherrschbar machen 96

2.4 Böse Überraschungen 102
Der bewusste Umgang mit Risiken im Projekt
Die kollektive Risikoignoranz 102
Den echten Risiken auf die Spur kommen 103
Die Risiken konkretisieren und bewerten 105
Den Risiken entschieden begegnen 107
Risiken ernst nehmen – auch in agilen Projekten 109

ETAPPE 3
Das Abenteuer beginnt 113

3.1 Das agile Dreigestirn 115
Schlagkräftig aufgestellt in agilen Projekten
Manchmal geht es gar nicht ohne Projektleiter 115
Der Projektleiter – verantwortlich für das Projekt 116
Der Product Owner – verantwortlich für das Produkt 118
Der Scrum Master – verantwortlich für das Team 120

3.2 Ein funktionierendes Team 125
Worauf es bei der Selbstorganisation ankommt
Die Dysfunktionen eines Teams 125

Mythos 1 – Selbstorganisation braucht keine Führung 127
Mythos 2 – Selbstorganisierte Teams wollen keine Führung 129
Mythos 3 – Selbstorganisation entsteht, wenn ich mich raushalte 130
Die 10 Gebote der Selbstorganisation 131

3.3 Letzte Vorkehrungen 136
Die wichtigsten Scrum-Abläufe installieren
Das Product Backlog – Notwendige Vorbereitungen treffen 138
Die Sprint-Planung – Die nächsten Wochen fest im Blick 139
Der Daily Scrum – Meetings in agilen Teams 140
Das Burn-Down-Chart – Die Ergebnisse kontrollieren 141
Das Sprint Review – Die Ergebnisse präsentieren 143
Die Retrospektive – Das Ziel der kontinuierlichen Verbesserung 144
Nach dem Sprint ist vor dem Sprint 145

3.4 Jetzt geht's los 149
Die Gestaltung des Kick-off-Meetings
Die Vorgespräche – Erst mal die Lage sondieren 149
Ziel und Aufgaben des Kick-off-Meetings 150
Die Kick-off-Rede – das Team begeistern 151
Das gegenseitige Kennenlernen 152
Dem Projektteam Orientierung geben 153
Der Team-Kodex – Regeln für die Zusammenarbeit 154
Wichtige Artefakte und Events 155
Und was kommt dann? 157

ETAPPE 4
Trotz Gegenwind auf Kurs 161

4.1 Vielfalt statt Einfalt 163
Die richtigen Projektmitarbeiter auswählen
Auch Teamgeist kann man planen 163
Der Mythos vom perfekten Team 164
Belbin und die sozialen Rollen im Team 164
Typische Probleme in agilen Teams 166
Die Kenntnis der verschiedenen Teamrollen 167

4.2 Von Null auf Hundert 174
Der Weg zum High-Performance-Team
In vier Stufen zur Top-Leistung 175
Scrum-Teams – eine besondere Herausforderung 176
Forming: Das Team lernt sich kennen 177
Storming: Es kommt zu ersten Reibereien 177
Norming: Die Spielregeln werden gesetzt 178
Performing: Das Team bringt Top-Leistungen 179
Auf alle vier Stufen kommt es an 180

4.3 Wer hat hier das Sagen? 184
Der Umgang mit Rangordnungen in agilen Teams
Eine Hierarchie gibt es immer 185
Das Prinzip der Hackordnung 185
Eine Hackordnung etablieren 186
Warum der Alpha die Betas braucht 187
Der Umgang mit Low Performern 188
Ursachen im Einzelgespräch differenzieren 189

4.4 Die Gratwanderung 193
Zusammen mit dem Scrum Master die Teamprozesse etablieren
Die schwierige Rolle des Scrum Masters 193
Schnell zu guten Entscheidungen kommen 194
Hindernisse aus dem Weg räumen 195
Der Umgang mit Impediments 197
Konflikte im Team zügig beilegen 197
Afterwork-Aktivitäten ja, Zwang nein! 199

ETAPPE 5
Hindernisse überwinden 203

5.1 Miserable Meeting-Kultur 204
Sprint Meetings statt Marathon-Sitzungen
Scrum Master – Moderator und Sekretärin? 206
Die vier Grundprinzipien agiler Meetings 207
Timeboxing – Wer hat an der Uhr gedreht? 207
Teamwork – Was macht die Gruppendynamik? 208
Fokussierung – Wie gelingt die richtige Balance? 209
Machbarkeit – Is done better than perfect? 209
Das Gesetz der zwei Füße 210

5.2 Sand im Getriebe 213
Reibungsverluste im agilen Handeln vermeiden
Die Sprint-Planung – Grundlage für einen reibungslosen Sprint 214
Daily Scrum – Und täglich grüßt das Murmeltier 215
Burnout – Teams im Crunch Mode 217
Reviews – Von wegen Grande Finale 218
Der Verstoß gegen die Las-Vegas-Regel 219
Am Ende sind das alles Impediments 220

5.3 Die Mühen der Ebene 224
Wie man Hindernisse aus dem Weg räumt
Es liegt nicht am fehlenden Mindset 225
Hindernis 1: Simulierte Agilität 225
Hindernis 2: Starre Strukturen 226
Hindernis 3: Fehlendes Vertrauen 227

ETAPPE 6
Am Rande des Abgrunds 231

6.1 Möge die Macht mit dir sein 233
Wie man sich Macht und Einfluss sichert
Die Machtlosigkeit des agilen Projektleiters 234
Macht und Einfluss muss man sich verdienen 235
Projektkrisen erfordern ein machtvolles Vorgehen 236
Weniger Kontrolle bedeutet mehr Kreativität 237

6.2 Wüste, Hölle oder Arktis 241
Wann Sie in einem agilen Team eingreifen sollten
Wenn die Stimmung im Team zum Problem wird 241
Irgendwann muss jemand die Reißleine ziehen 242
Jetzt braucht es doch einen Projektleiter 243

Schritt 1: Die Lage erst einmal beruhigen 243
Schritt 2: Gezielt intervenieren 244
Schritt 3: Die Arbeitsfähigkeit wiederherstellen 244
Schritt 4: Ein neues Wir-Gefühl schaffen 245
Eine Teamveränderung ist die Ultima Ratio 245

6.3 Houston, wir haben ein Problem 248
Wann Sie ein agiles Projekt vorübergehend leiten müssen
Die Projektkrise als Krise wahrnehmen 249
Jemand muss jetzt die Verantwortung übernehmen 250
Schritt 1 – Analyse:
Die Wahrheit, nichts als die Wahrheit 250
Schritt 2 – Planung: Alles zurück auf Anfang 251
Schritt 3 – Verhandlung: Die Beteiligten müssen Farbe bekennen 251
Schritt 4 – Neustart: Mit Vollgas aus der Krise 252
Schritt 5 – Rückzug: Die Anderen wieder machen lassen 252

ETAPPE 7
Kurz vor dem Gipfel 255

7.1 Nicht auf der letzten Rille 257
Wie der Release-Sprint eine Etappe abschließt
Mit letzter Kraft das Ziel erreichen 258
Alles wird auf Hochglanz poliert 259
Das Produkt zur Verfügung stellen 259
Rendezvous mit den Stakeholdern 260
Agil bedeutet keineswegs planlos 261

7.2 Die letzten Meter im Projekt 264
Wie man die Abschlussarbeiten im Projekt meistert
Was zu einem geregelten Projektende gehört 264
Die Abnahme – Der inhaltliche Projektabschluss 265
Abschlussbericht – Projektergebnisse optimal darstellen 266
Die interne Übergabe – den laufenden Betrieb sicherstellen 267
Die externe Übergabe – die Ergebnisse präsentieren 267
Das Projekt ist zu Ende – mit dem Produkt geht es weiter 268
Das Werk ist vollbracht – jetzt wird gefeiert 268

7.3 Ein letzter Blick zurück 271
Wie man aus dem Projekt die richtigen Lehren zieht
Durch Manöverkritik auf Effizienz getrimmt 272
Lessons Learned – Aus Fehlern in Projekten lernen 272
Die vier Blickwinkel eines Rückblicks 273
Die richtigen Schlüsse daraus ziehen 275

Anhang 279
Die Protagonisten 279
Literatur 284
Über den Autor 287

Vorwort

Seit dem Erscheinen des ersten Bandes meiner *Projekt-Safari* sind etwa zehn Jahre vergangen, und ich habe sehr viel positives Feedback zu diesem Buch bekommen. Besonders hat es mich gefreut, dass einige Experten es als »Must-have« für jeden Projektleiter bezeichnen oder das *Hamburger Abendblatt* das Buch seinerzeit in die Top Ten des Jahres gewählt hat.

Doch die Entwicklung im Projektmanagement hat nicht Halt gemacht – und es war nun an der Zeit, den Klassiker zu ergänzen. Zwischenzeitlich haben neue, agile Prozesse wie Scrum in die Welt der Projekte Einzug gehalten. Ich kam und komme immer häufiger in die Verlegenheit, den praktischen Nutzen von agilen Methoden zu erklären. Das wohl auch, weil im ersten Band der »Projekt-Safari« kein Wort darüber zu lesen ist – denn dort habe ich den herkömmlichen (klassischen) Projektansatz beschrieben.

Der Wert agiler Methoden erschließt sich keineswegs sofort, und in vielen Unternehmen ist noch wenig bekannt darüber, welche Auswirkungen diese Methoden auf den Verlauf eines Projektes haben. Ursprünglich stammt das Konzept aus der Softwareentwicklung. Daraus hat sich dann im Laufe der Jahre allgemein ein »agiles Projektmanagement« entwickelt, denn nicht nur Softwareprojekte lassen sich agil gestalten und steuern.

Der zweiten Band der *Projekt-Safari* führt in diese neue agile Welt ein. Er möchte einen Beitrag zum besseren Verständnis agiler Projektmethoden im Allgemeinen und des Scrum-Ansatzes im Besonderen leisten. Das Buch versteht sich damit nicht als reines Scrum-Werk, das die Werkzeuge und Methoden im Detail vorstellt. Vielmehr knüpft es an den ersten Band an und stellt wieder praktische Hinweise in den Vordergrund: Was ist der Unterschied zum klassischen Projektmanagement? Wann sind agile Methoden sinnvoll? Wie lassen sich diese Methoden im Projektalltag einsetzen? Worauf ist dabei zu achten? Das Buch richtet sich deshalb in erster Linie an die Projektverantwortlichen, nämlich Projektleiter, Product Owner und Scrum Master.

Ein abschließendes Wort sei mir zum Gendern in diesem Buch gestattet. Mir ist die Relevanz einer gendergerechten Sprache bewusst, jedoch wollte ich vermeiden, dass die Lesbarkeit meines Buches unnötig leidet. Ich konnte mir nicht vorstellen, dass der Text durchgängig mit Sonderzeichen durchbrochen wird. Bei Rollenbezeichnungen wie beispielsweise »Projektleiter« habe ich mich deshalb für das generische Maskulin als geschlechtsneutrale Personenbezeichnung entschieden. In den Erzählungen aus den Projekten spielt Gendern keine Rolle, weil ich nah an den handelnden Einzelpersonen erzähle. Wenn in den praktischen Beispielen eine Frau gemeint ist, dann heißt es eben auch »Projektleiterin« und nicht »Projektleiter«.

Mario Neumann

Einleitung

Vor mehr als 20 Jahren, vom 11. bis 13. Februar 2001, trafen sich 17 Softwareentwickler in den amerikanischen Rocky Mountains, um ein paar Tage Ski zu fahren. Die Lodge am Little Cottonwood Creek liegt am Fuße des Skigebietes Snowbird und bietet ihren Gästen einen atemberaubenden Blick auf die tief verschneiten Wasatch Mountains von Utah.

Die Protagonisten – unter ihnen namhafte Größen wie Kent Beck, Ken Schwaber, Jeff Sutherland und Alistair Cockburn – hatten sich nicht nur getroffen, um tagsüber Ski zu fahren. Sie wollten auch zusammen essen, miteinander reden – und über berufliche Erfahrungen diskutieren. Vermutlich ahnten sie nicht, welche Auswirkung ihre Zusammenkunft auf die Softwareentwicklung und in der Folge auf das Projektmanagement haben würde.

Was die Männer mittleren Alters dort in den Bergen von Utah verband, war eine tiefe Frustration über die etablierten Entwicklungsmethoden der 1990er Jahre. Trotz großer Differenzen waren sie sich in einem Punkt einig: In der Softwareentwicklung waren dringend neue Ansätze notwendig, weil man mit den damals typischen, sehr schwergewichtigen Methoden nicht mehr weiterkam. Zwischen den Kundenwünschen auf der einen Seite und der Bereitstellung von Ergebnissen auf der anderen Seite lag eine enorme Zeitspanne, was ein ums andere Mal ins Desaster führte. Rund ein Drittel aller IT-Projekte scheiterten vorzeitig. Da war Projektmanagement nicht nur ein Abenteuer, sondern mitunter das reinste Himmelfahrtskommando. Auch betriebswirtschaftlich war das alles völlig inakzeptabel. Auf der einen Seite wuchs der Bedarf an leistungsfähiger Software; gleichzeitig endete aber ein Teil der Projekte im Fiasko.

Vor diesem Hintergrund diskutierten die Protagonisten in Snowbird viele widersprüchliche, aber auch sich ergänzende Perspektiven zur Softwarentwicklung. Statt sich daran zu versuchen, die einzig wahre Softwareentwicklungsmethode und -technik zu definieren, legten sie schließlich den Schwerpunkt auf gemeinsame Erkenntnisse, die sie in folgenden vier »agilen Werten« zusammenfassten: Menschen und ihre Interaktionen sollten mehr Gewicht bekommen als Prozesse und Werkzeuge, eine funktionierende Software sollte mehr zählen als eine umfassende Dokumentation, die Zusammenarbeit mit dem Kunden wichtiger sein als die Vertragsverhandlung – und vor allem sollte das Eingehen auf Veränderungen Priorität haben vor der starren Befolgung eines Plans.

Im »Agilen Manifest« wurden die vier Kernaussagen dann näher ausgeführt: Die Kundenzufriedenheit sei durch eine frühzeitige und regelmäßige Auslieferung von nützlicher Software sicherzustellen, Änderungswünsche des Kunden seien willkommen – und das Projektteam solle auf täglich aktualisierter Basis

zusammenarbeiten. Am Anfang des agilen Projektmanagements, so belegt das Dokument, stand die Ablehnung des traditionellen Projektmanagements, das auf intensive Planung, Überwachung und Steuerung setzt.

Das ist jetzt mehr als zwei Jahrzehnte her. Die Verfasser des Manifests hatten sich wohl kaum vorstellen können, welche Folgen ihr Wochenendausflug auf das Projektmanagement haben würde. Während ihrer Zusammenkunft in den Rocky Mountains wollten sie nichts revolutionär Neues schaffen. Sie waren frustriert und wollten provozieren – wollten auf die Probleme in der Softwareentwicklung hinweisen.

Dieses Buch lädt Sie zu einer neuen Projekt-Safari ein. Wieder erwarten Sie sieben Etappen, doch die Vorgehensweise orientiert sich jetzt am agilen Mindset jener Männer in den Bergen von Utah. Stand im ersten Band der klassische Projektmanagementansatz im Mittelpunkt, möchte dieses Buch das Rüstzeug geben, um mit agilen Arbeitsweisen die vielfältigen Abenteuer zu meistern, die einem Projektmanager begegnen. Das Buch enthält wieder zahlreiche Tipps, berichtet über Erlebnisse von Projektleitern – und gibt Ihnen einfache Modelle und wirksame Werkzeuge an die Hand, mit denen Sie ein agiles Projektabenteuer erfolgreich managen können.

Die *erste Etappe* leitet uns von der Idee zum konkreten Projektauftrag. Auch wenn Agilität das Projektgeschehen bestimmt, so bleibt es doch eine Reise – meist durch unbekanntes Gelände. Bevor wir das Navigationssystem in Betrieb nehmen, sollten wir wissen, welches Ziel wir eingeben.

Das Product Backlog ist gewissermaßen der Heilige Gral der agilen Vorkämpfer. Es tritt in der *zweiten Etappe* an die Stelle der klassischen Projektplanung: Wir formulieren unsere Anforderungen, bleiben aber offen gegenüber neuen Einsichten oder Ideen.

Mit der *dritten Etappe* stehen wir unmittelbar vor dem Aufbruch: Wir beschäftigen uns mit den Hauptakteuren und sorgen für eine funktionsfähige Projektorganisation. Außerdem treffen wir letzte Vorkehrungen und starten mit einem Kick-off-Workshop in das Projekt.

In der *vierten Etappe* bringen wir das Projekt ins Rollen und schaffen die Voraussetzungen für agiles Arbeiten. Es geht es darum, hilfreiche Strukturen und agile Methoden zu etablieren – denn auch bei so klar definierten Prozessen wie Scrum kann im Projektalltag eine Menge schiefgehen.

Nur eine starke Mannschaft an Bord sichert den gewünschten Erfolg. Das gilt auch für agile Projekte. In der *fünften Etappe* sorgen wir deshalb für Freiräume, in denen Projektmitarbeiter agil und selbstorganisiert arbeiten und aktiv Verantwortung übernehmen.

Nun gilt es, das Projekt auf Kurs zu halten. In der *sechsten Etappe* klären wir, wie das auch trotz heikler Konfliktherde gelingt, mit denen immer zu rechnen ist. Hier kommt es vor allem auf das Führungsgeschick des Projektleiters an.

Mit der *siebten Etappe* biegen wir auf die Zielgerade ein – wobei wir das in agilen Projekten nicht nur einmal machen, sondern am Ende jedes der zahlreichen Zwischenziele. Das Projektteam trifft sich jedes Mal, um die zurückliegende Projektphase Revue passieren zu lassen. So profitiert der folgende Projektabschnitt von den Erfahrungen des vorhergehenden. Man lernt noch während des Projekts aus Fehlern und kann Erfolge wiederholen.

In *Projekt-Safari 2* treffen wir einen alten Bekannten wieder,

der uns durch alle sieben Etappen begleiten wird: den Projektleiter Tom. Der mittlerweile 45-jährige Diplom-Ingenieur hatte im ersten Band ein anspruchsvolles Projekt erfolgreich gemeistert und seine Erfahrungen mit uns geteilt. Die Geschäftsführung hat ihm jetzt ein neues Projekt übertragen, das für das Familienunternehmen mit weltweit rund 6000 Mitarbeitern wieder von großer Bedeutung ist. Das Vorhaben hat seine besonderen Tücken, was Tom dazu veranlasst, dieses Mal auf agile Methoden setzen. In einem Tagebuch notiert er seine Erlebnisse und lässt uns so an seinen Schlussfolgerungen teilhaben. Dabei vergleicht er den agilen Ansatz mit der klassischen Vorgehensweise – und kommt zu bemerkenswerten Erkenntnissen.

Und plötzlich steht alles Kopf

Die geänderte Sicht auf die Welt der Projekte

If the only tool you have is a hammer,
you tend to see every problem as a nail.

Abraham Maslow, Psychologe

»Der Helikopter landet auf einer freien Fläche mitten im Urwald. Eine Gruppe von Menschen wird herausgestoßen; geblendet vom grellen Sonnenlicht blicken sie sich unsicher um, ohne zu wissen, was sie erwartet. Erst vor wenigen Minuten haben sie das Ziel ihrer Expedition erfahren – und schon hebt der Helikopter wieder ab und lässt sie im Dschungel zurück. Ihr Abenteuer beginnt.«

Mit diesen Worten begann das Buch *Projekt-Safari*, das vor mehr als zehn Jahren erschien. Charakteristisch für dieses klassische Projektabenteuer war ein hohes Maß an Standardisierung mit einer strikten Planung und klaren Vorgaben, wie welche Projektziele erreicht werden sollten. Ressourcen wurden klar zugeteilt, Kosten schon im Voraus abgeschätzt. Ebenso wurde ein Endtermin festgelegt, der dem Auftraggeber eine gewisse Sicherheit gab.

Das wusste auch der Leiter jener Gruppe, die sich nun plötzlich im Urwald wiederfand. Anstatt gleich in den Dschungel einzudringen, ließ er seine Leute am Landungsplatz die Zelte aufschlagen. Gemeinsam mit ihnen erarbeitete er einen Plan, sondierte die größten Risiken, besprach Notfallpläne und vereinbarte Regeln, die für die Zeit der Expedition galten. Erst dann brach die Gruppe auf. Trotz all dieser Vorbereitungen blieb es eine Reise ins Ungewisse.

Diesem traditionellen Ansatz steht nun das agile Projektmanagement gegenüber. Wieder beginnt das Abenteuer auf einer freien Fläche mitten im Urwald, doch im Gegensatz zu damals wird am Anfang der Expedition nicht alles im Detail festgelegt.

Der Grundgedanke: Menschen können im Voraus nicht alle Einzelheiten überschauen. Selbst ein Auftraggeber weiß zu Beginn eines Vorhabens noch nicht im Detail, was er genau will und welche Möglichkeiten bestehen. Und so entscheidet man sich für eine Expedition der kleinen Etappen: Abschnitt für Abschnitt schlägt sich die Gruppe einen Pfad durch den Dschungel. Sie dringt vergleichsweise schnell in den Dschungel ein, hält aber nach jeder Etappe inne, überprüft das Vorgehen und, wenn nötig, verbessert es. So kann die Gruppe dynamisch auf die Herausforderungen des Dschungels reagieren.

Im ersten Schritt, so zeigt die Erfahrung, wird vieles noch nicht ideal gelöst. Dem trägt diese iterative Vorgehensweise Rechnung, indem Fehler im nächsten Schritt korrigiert werden. Überspitzt ausgedrückt: Der erste Wurf ist für die Tonne, aber wir können daraus lernen und darauf aufbauen.

Diese Flexibilität und Anpassungsfähigkeit sind für das agile Projektmanagement charakteristisch. Die Projektziele werden erreicht, indem sich die Beteiligten durch Feedback kontinuierlich auf dem Laufenden halten und ihr Vorgehen immer wieder neu justieren. Entscheidend für den Erfolg ist nicht ein perfekter Plan, sondern die Orientierung an den Wünschen der Kunden. Wenn sich deren Anforderungen ändern, bietet das agile Projektmanagement genügend Flexibilität, um die erforderlichen Ziel- und Planänderungen schnell und einfach einzuarbeiten.

Zwei zunehmend unversöhnliche Lager

Klassisch oder agil – zwei Methoden im Projektmanagement, wie sie unterschiedlicher kaum sein könnten. Jeder Projektleiter möchte sein Projekt zum Erfolg führen, doch mit welchem Ansatz? Die gegensätzlichen Ideen und Vorgehensweisen der beiden Ansätze äußern sich nicht zuletzt in hitzigen Debatten darüber, welche Methode denn nun die »bessere« sei. Dabei besitzen beide Ansätze ihre (un-)bestrittenen Vorteile und haben zweifelsohne ihre Daseinsberechtigung.

Klassisches Projektmanagement entstand im 20. Jahrhundert – in einer Zeit, als Sicherheit und die Verhinderung von Risiken im Mittelpunkt standen. Es galt das Motto: Safety first! Um Fehler, Risiken und Probleme in den Griff zu bekommen, wurden Projekte von Anfang bis Ende so detailliert wie möglich geplant. Gegen Ende der 1990er Jahre uferten diese Pläne derart aus, dass sie in ihrer ganzen Pracht selbst DIN-A0-Plotter an ihre Grenzen bringen konnten.

Im 21. Jahrhundert kam mit der Agilität die Forderung auf, sich auf die eigentlichen Werte zurückzubesinnen: den Kunden und seine Anforderungen. Darin lag keine hellseherische Genialität, sondern die schlichte Erkenntnis, dass man über das Ziel hinausgeschossen war und jetzt eine Gegenbewegung nötig war, um wieder zur sinnvollen Mitte zu gelangen. Mit dem Hype der letzten Jahre drängt sich allerdings der Eindruck auf, dass das klassische Projektmanagement zunehmend durch agile Methoden abgelöst wird. Agilität gilt als »trendy«, während das klassische Projektmanagement als nicht mehr zeitgemäß abgetan wird.

Auch wenn die mehrere hundert Seiten starken Bibeln des klassische Projektmanagements, etwa der PMBoK-Guide oder die ICB, sicherlich übertrieben sind und die Kritik daran berechtigt ist: Die Hoffnung kann doch nicht sein, mit agilen Ansätzen jetzt sämtliche Herausforderungen des Projektalltags lösen zu können. Agile Frameworks mögen zwar vereinfachte Vorgehensweisen hervorgebracht haben, sie simplifizieren aber nicht die Herausforderungen, die im jeweiligen Projekt stecken. Und doch gibt es agile Coaches, die davor die Augen verschließen und so lange mit ihren Werkzeugen hantieren, bis das Unternehmen kapituliert: »Agil haben wir probiert. Das hat bei uns nicht funktioniert.«

Wer als Werkzeug nur den Hammer kennt, sieht in jedem Problem einen Nagel. Viel zu lange hat diese Weisheit die beiden Lager getrennt. Es wird Zeit, dass wieder die Sache über das richtige

Werkzeug entscheidet – und nicht die Vorliebe des Projektleiters oder gar die Vorgabe eines Bereichsleiters. Wie bei jedem Werkzeug kommt es darauf an, wer es in der Hand hält. Manchmal muss der Hammer anders geführt werden oder es braucht eben ein ganz anderes Werkzeug. Letztlich ist es eine rein pragmatische Frage, welche Methode im konkreten Fall zum Zuge kommen sollte. Um ein Projekt von Anfang an auf den richtigen Weg zu bringen, gilt daher die Maxime: Pragmatismus vor methodischer Dogmatik!

Bleibt festzuhalten: Im Projektmanagement haben beide Methoden ihre Berechtigung. Wie die praktische Erfahrung der letzten Jahre gezeigt hat, sollte man bedarfs- und fallorientiert zwischen den beiden Projektmanagementmethoden unterscheiden und die jeweils sinnvollste Variante nutzen. Vermeiden Sie es also, sich im Paradigmen-Streit blindlings auf eine der beiden Seiten zu schlagen! Schließlich gilt für beide Lager gleichermaßen: Ein Projekt ist dann erfolgreich, wenn es seine Ziele – Ergebnisqualität, Termintreue, Budgettreue – erreicht oder übertroffen hat.

Zielorientiert? Oder doch zweckorientiert?

Klassisch oder agil – in der Diskussion um die beiden gegensätzlichen Methoden wird immer wieder die Gültigkeit einer eisernen Regel des Projektmanagements angezweifelt: Ein Projektziel muss klar, spezifisch und messbar sein. Wenn aber, wie in agilen Vorhaben propagiert, die Ziele noch nicht vollständig bekannt sind, wie sollen sie dann klar, spezifisch und messbar sein?

Wenn Anhänger agiler Methoden so argumentieren und deshalb glauben, auf klare Projektziele verzichten zu können, übersehen sie eines: Projektziele müssen nicht zwingend Ergebnis- oder Leistungsziele sein, es können auch Vorgehensziele sein. Klassisches Projektmanagement definiert in der Regel klare Ergebnis- oder Leistungsziele, während agiles Projektmanagement sich auf Vorgehensziele konzentriert.

Ein konventionelles Projekt ist beispielsweise der Bau eines Flugzeugs. Das Ergebnis, nämlich das funktionsfähige Flugzeug, ist vorgegeben und wird dementsprechend präzise im Projektziel spezifiziert. Der klassische Projektansatz lässt sich demzufolge als zielorientiert beschreiben. Die Vorgehensziele werden dagegen oftmals eher schwammig formuliert und in der Verantwortung des Projektleiters und seiner Ingenieure belassen.

Agiles Projektmanagement kehrt dieses Muster um. Hier bleiben die Ergebnis- oder Leistungsziele über weite Strecken des Projektes bis zu einem gewissen Grad flexibel. Beispielsweise liegt es bei der Entwicklung einer App nahe, die Funktionalitäten nicht von vorneherein zu spezifizieren, sondern im Laufe des Projektes immer wieder weiterzuentwickeln. Dafür sind hier aber die Vorgehensziele strikt festgelegt: Das Vorgehen muss spezifisch und messbar sein, um Budget- und Terminvorgaben trotz flexibler Leistungsziele einhalten zu können. Die agile Vorgehensweise könnte man deshalb auch als sehr zweckorientiert bezeichnen.

Die Umkehr des magischen Dreiecks

Egal ob klassisch oder agil, ein Projektleiter kämpft immer gleichzeitig an drei Fronten: Umfang, Zeitraum und Aufwand. Er hat

mit den Inhalten, den Terminen und den Kosten des Projekts zu tun – drei Größen, die sich nur schwer miteinander vereinbaren lassen. An diesem Grundprinzip des Projektmanagements ändern auch agile Vorgehensweisen nichts.

Setzt ein Projektleiter inhaltlich die Messlatte zu hoch, laufen Termine und Aufwand aus dem Ruder. Ist dagegen die Zeit zu knapp bemessen, schießen die Kosten in die Höhe und die Qualität der Ergebnisse leidet. Das eine ist also nur auf Kosten des anderen zu haben, die Größen konkurrieren miteinander. Dieses Phänomen ist auch als »Magisches Dreieck des Projektmanagements« bekannt. Die Kunst liegt darin, alle drei Größen während der gesamten »Projektreise« im Auge zu behalten und erfolgreich zu managen.

Mit der Umkehr der Muster im agilen Projektmanagement ändert sich der Blick auf das Magische Dreieck. Beim Wechsel vom traditionellen zum agilen Vorgehen tauschen Fixpunkte und verhandelbare Aspekte ihre Rollen und stellen das Dreieck quasi auf den Kopf.

In einem konventionellen Projekt ist der Umfang fest vorgegeben, während die Kosten und die verfügbare Zeit variabel sind. Für den Bau eines Flugzeuges würde dies bedeuten, dass zu Beginn des Projekts die Produktanforderungen in Form von Lasten- und Pflichtenheften ausführlich spezifiziert werden. Die benötigten Kosten und geplanten Termine sind hingegen variabel und müssen anhand des festgelegten Umfangs eingeschätzt und verhandelt werden.

Das Magischen Dreieck gibt dem Projektteam Orientierung und kann Grundlage sein, mit dem Auftraggeber über Kompromisse zu verhandeln. Wenn etwa beim Bau des Flugzeugs unerwarteten Schwierigkeiten auftreten und der Projektleiter nach der Hälfte des Projekts feststellt, dass der Fertigstellungstermin nicht einzuhalten ist, kann er nur zwei Variablen beeinflussen:

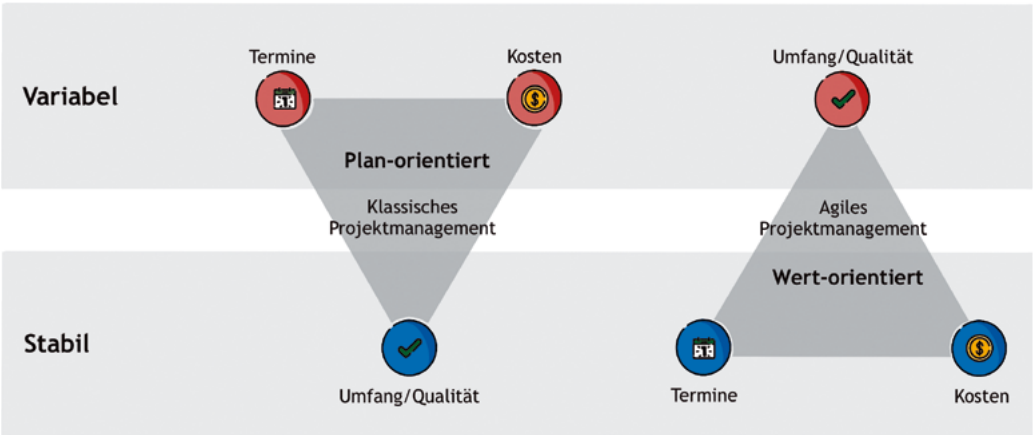

Abbildung 1: Die Umkehr des Magischen Dreiecks

- **die Zeit** – Er kann mit dem Auftraggeber über einen späteren Fertigstellungstermin verhandeln.
- **die Kosten** – Er kann zusätzliche Mitarbeiter zum Projekt hinzuziehen, wodurch die Kosten steigen.

Auch beim agilen Projektmanagement geht es um die Frage, welche Elemente festgelegt sind und wo es sich nur um Schätzungen handelt. Das Ergebnis ist hier jedoch ein anderes,

weshalb das Magische Dreieck auf dem Kopf steht: Anders als beim traditionellen Projektmanagement sind bei agilen Projekten der Zeitrahmen und die Kosten festgelegt, während der Umfang veränderlich ist.

Der Begriff »Umfang« ist bei beiden Vorgehensmodellen gleich definiert: Ein bestimmtes Flugzeug oder eine bestimmte Software soll gebaut oder entwickelt und ausgeliefert werden. Anders als beim Flugzeug werden bei der Software jedoch nicht gleich zu Anfang ausführliche und detaillierte Anforderungen formuliert. Konkrete Inhalte werden nur für die nächste Iteration verhandelt, Kosten und Termine zur Umsetzung dieses Schritts sind jedoch bekannt. So kann das Projektteam leichter auf Veränderungen reagieren. Es kann Anpassungen vornehmen, ohne seine Pläne zu verwerfen.

KLASSISCH	AGIL
Umfang ist fest und klar definiert	Umfang ist flexibel und wird verhandelt
Zeit und Aufwand werden geschätzt	Zeit und Aufwand sind vorab festgelegt
Linearer Projektprozess	Iterativer Projektprozess
Prozess ist fest vorgegeben	Prozess wird fortlaufend verbessert
Einfluss von Stakeholdern sinkt im Verlauf des Projekts	Einfluss von Stakeholdern bleibt konstant im Verlauf des Projekts
Anforderungen sind zu Beginn bekannt	Anforderungen anfangs eher unscharf
Anforderungen werden nur am Anfang erfasst (Lasten- und Pflichtenheft)	Anforderungen werden kontinuierlich erfasst und priorisiert (Product Backlog)
Ergebnisse werden nur am Endes des Projekts geliefert und bewertet	Ergebnisse werden im Projektverlauf regelmäßig geliefert und bewertet
Änderungen werden eher nicht erwartet	Änderungen werden ständig erwartet
Änderungen sind im Projektverlauf schwierig umzusetze	Änderungen werden im Projektverlauf fest eingeplan

Abbildung 2: Klassisch oder agil – Die Unterschiede

Ungeachtet der Unterschiede zwischen dem klassischen und dem agilen Ansatz gibt es beim Anwenden des Magischen Dreiecks kein Richtig oder Falsch. Das Magische Dreieck soll lediglich helfen, während des Projektabenteuers die besten Entscheidungen und Kompromisse zu finden.

Das Beste aus zwei Welten

Im Projektmanagement scheinen zwei Welten einander gegenüberzustehen: auf der einen Seite das traditionelle, auf der anderen Seite das agile Projektmanagement. Jenes steht für einen akribischen Plan, dieses für ein kundennahes und teamorientiertes Vorgehen. Eine Brücke zwischen den beiden Ansätzen soll das *hybride Projektmanagement* schlagen. Im Grunde handelt es sich dabei um eine Mischform der beiden Methoden. Anstatt den

Fokus auf die Unterschiede zu setzen, werden Verknüpfungen zwischen den Ansätzen gesucht. Hybrides Projektmanagement kombiniert die Stärken klassischer und agiler Methoden miteinander.

Für einen Projektleiter öffnen sich damit neue Türen. So kann er Schwächen des klassischen Projektmanagements umgehen, indem er einzelne Vorgänge agil durchführt. Hierfür lohnt es, sich den Nutzen des agilen Projektmanagements vor Augen zu führen. Er liegt insbesondere in folgenden Aspekten:

- Die Mitarbeiter sollen sich selbst organisieren und damit weniger überlastet sein.
- Es gibt sowohl Eigen- als auch Teamverantwortung. Wer wofür verantwortlich ist, ist jedoch klar definiert.
- Im Vordergrund steht eine iterative Arbeitsweise. Dabei wird zunächst eine vorläufige Version oder ein Prototyp entworfen, für den sich das Team ein Feedback einholt. Unter Einbindung des Feedbacks entsteht eine Nachfolgeversion.
- Es werden so viele Feedback- und Nachbesserungsschleifen wiederholt, bis der Auftraggeber mit dem Ergebnis zufrieden ist.
- Das Team muss auf den Input des Auftraggebers reagieren, was eine besonders flexible Arbeitsweise voraussetzt beziehungsweise überhaupt erst ermöglicht.
- Das Risiko eines kompletten Scheiterns des Projekts wird durch die iterative Arbeitsweise deutlich reduziert.

Was sich mit der agilen Vorgehensweise nicht ändert, sind die klaren Projektziele: Der Auftraggeber erwartet eine termingerechte Lieferung in einer vorab definierten Anzahl von Tagen, in einem festgelegten Kostenrahmen und mit einem exzellenten Ergebnis. Agile Methoden beschreiben nur, wie man dieses Ziel erreichen möchte – sie dürfen aber nicht zum Mittelpunkt des Vorhabens werden.

Natürlich sind Zeit, Kosten und Qualität Kriterien aus dem traditionellen Projektmanagement. Aber nur weil man solche Zielsetzungen »traditionell« nennt, sind sie nicht überholt. Die Stärken des klassischen Projektmanagements – Verbindlichkeit, Zuverlässigkeit und Sicherheit – sollte man nicht leichtfertig über Bord werfen, nur weil »Agilisten« sie gerne verteufeln.

Als Projektleiter oder Projektleiterin sollten Sie sich nicht gezwungen fühlen, entweder nach der einen oder nach der anderen Methode zu arbeiten. Sehen Sie stattdessen Planung und Agilität als zwei Pole an, zwischen denen sich ein breites Spektrum an Möglichkeiten entfaltet. Die Herausforderung liegt darin, diejenigen Möglichkeiten herauszufinden, die zu guten Ergebnissen führen und die Projektarbeit erleichtern. Sowohl klassische wie auch agile Wege können hier gleichermaßen gute Antworten geben.

Ein Blick in Toms Tagebuch

Montag, 5. Februar

Der närrische Endspurt steht vor der Tür – mit Weiberfastnacht, Rosenmontag und Faschingsdienstag geht der Karneval seinem Höhepunkt entgegen. Die ganze Stadt fiebert dem Großereignis entgegen. An mir geht das alles vorbei, nach einer jecken Zeit ist mir nicht zumute. Ich muss nämlich persönlich verkünden, dass wir den Produktivstart der neuen Steuerungssoftware erneut verschieben müssen.

Meinen Leuten kann ich keinen Vorwurf machen. Sie legen sich eigentlich immer mächtig ins Zeug. Wenn es sein muss, arbeiten sie Tag und Nacht, und normalerweise bekommen sie es immer hin. Noch vor Kurzem waren sie zuversichtlich, dass wir die Probleme in zwei Monaten in den Griff bekommen würden. Aber in diesem Projekt ist einfach der Wurm drin!

So bin ich heute zum Firmensitz unseres Unternehmens gereist, habe im Hotel eingecheckt und meine Klamotten aufs Zimmer gebracht. Anschließend habe ich mir an der Hotelbar ein Bier bestellt. Seitlich von mir saß ein Mann an der Theke und prostete mir zu. Er meinte, ich sähe so aus, als wolle ich mir Mut antrinken. Da musste ich doch lachen! Das Eis war gebrochen, und wir kamen ins Gespräch. Sein Name ist Niklas. Er arbeitet als Softwareentwickler in der Spieleindustrie. Niklas besucht gerade eine agile Konferenz in der Stadt und hat heute einen Vortrag zu »Scrum« gehalten. Natürlich habe ich schon von Scrum gehört, aber mich nie wirklich ernsthaft damit beschäftigt. So fragte ich ihn, was es damit auf sich hat.

So erfuhr ich, dass Scrum zur Gruppe sogenannter agiler Methoden gehört. Niklas beklagte, dass der Begriff »agil« ziemlich missverständlich sei, weil er oft mit »schnell« assoziiert wird. Zwar könne man mit agilen Methoden auch schneller entwickeln, aber das sei nicht das Hauptziel. Viel wichtiger sei es, dass man »flexibler« wird. Der Abend hat mich zwar einige Bier gekostet, die ich uns spendiert habe, brachte aber doch auch einige wichtige Erkenntnisse über agile Methoden.

Einige Dinge, die ich heute Abend gelernt habe:

- Das Wichtigste an Scrum ist, dass man offen bleibt gegenüber neuen Einsichten oder Ideen – den eigenen genauso wie denen des Kunden.
- Feedback ist entscheidend für ein erfolgreiches Produkt. Je eher ich Feedback bekomme, umso leichter kann ich auf Fehlentwicklungen reagieren.
- Man muss bereit sein, die eigenen Prioritäten und damit auch das Projektergebnis (Produkt) anzupassen.

Dienstag, 6. Februar

Heute war Vorstandssitzung. Dass es so hart kommen würde, hatte ich nicht erwartet. Die Vorstände haben mich regelrecht auseinandergenommen. Erst nach langer Diskussion ist es gelungen, vom Vorstand eine allerletzte Chance zu bekommen: Wir haben drei weitere Monate, aber keinen Tag länger. Dann muss es fertig sein!

Ich habe ich keine Ahnung, ob unsere Steuerungssoftware bis dahin wirklich fertig wird. Vielleicht könnten wir, so überlegte ich

nach der Sitzung, den Vorstand besänftigen, indem wir regelmäßig ein neues Release unserer Steuerungssoftware mit einem eingeschränkten Funktionsumfang ausliefern. Das wäre auch eine gute Möglichkeit, von den Produktionsingenieuren Feedback zu erhalten. Natürlich würde ich sagen, dass es nur eine Beta-Version ist, damit keine falschen Erwartungen entstehen.

Am Abend traf ich Niklas im Hotelrestaurant. Er winkte mir zu und fragte, ob ich mit ihm zu Abend essen möchte. Gerne nahm ich an. Ich erzählte ihm von meiner Überlegung, die Steuerungssoftware stückweise auszuliefern. Niklas dachte einen Moment darüber nach: »Das wäre ein möglicher Ansatz. Du solltest allerdings darauf achten, nur Funktionalitäten zu liefern, die schon ausreichend gut funktionieren. Wenn der erste Eindruck einer neuen Funktion schlecht ist, dann musst du nachher doppelt so hart arbeiten, um diesen Eindruck wieder zu revidieren. Ich würde ihnen deshalb auch nicht sagen, dass es ein Beta-Relase ist. Das klingt so, als wäre die Qualität nicht besonders gut.«

Und wieder haben wir uns ausführlich über Scrum unterhalten – naja, Niklas hat erzählt, und ich habe andächtig gelauscht. Ich finde immer mehr Gefallen an der Methode, denn sie vertraut auf den gesunden Menschenverstand der Personen, die sich letztendlich erfolgreich darin bewegen sollen.

Der Abend war erneut sehr lehrreich. Langsam aber sicher bekomme ich eine Idee, was ich tun muss, um mein Projekt doch noch zu retten. Neben mir liegt ein Bierdeckel, den ich mit aufs Zimmer genommen habe. Niklas hat darauf den Ablauf von Scrum skizziert und behauptet: Scrum sei so einfach, dass es auf einen Bierdeckel passt.

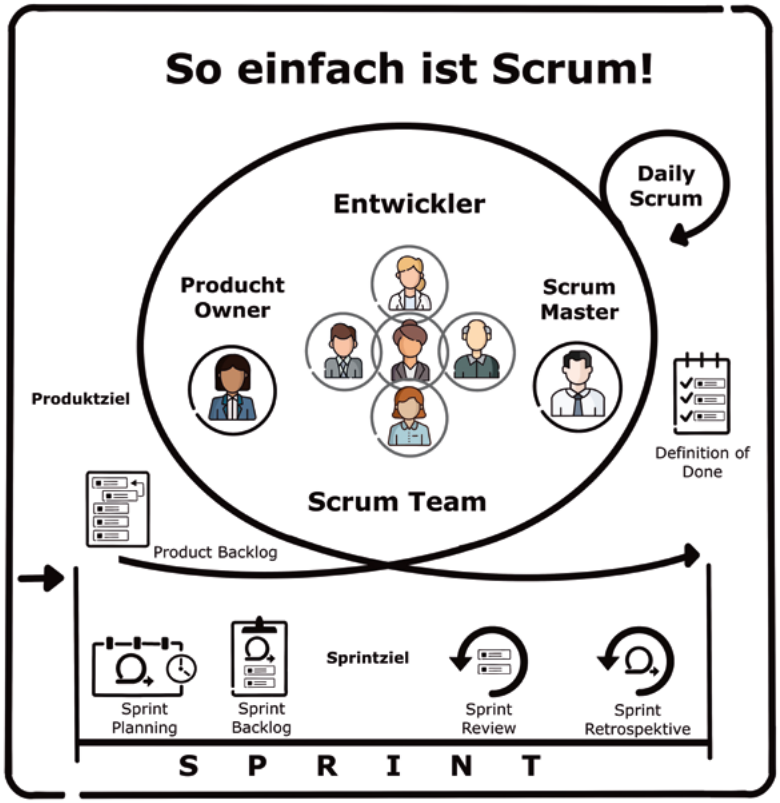

Abbildung 3: Scrum auf einem Bierdeckel

Unversöhnlicher Dogmatismus

Agilität ist in den letzten Jahren zum allgegenwärtigen Schlagwort geworden und nimmt manchmal fast schon religiöse Züge an. Werden Vorgehensmodelle in Projekten allerdings allzu dogmatisch diskutiert, besteht die Gefahr, dass Tools und Methoden sklavisch eingehalten werden (müssen) – was so ziemlich das Gegenteil von Agilität darstellt.

So wappnen Sie sich

- Denken Sie daran: Dogmatismus schadet der Agilität. Projektmanagement muss immer an das Projekt angepasst werden und ist deshalb meistens hybrid.
- Heißen Sie Veränderungen willkommen. Nehmen Sie Veränderungen immer auf Basis von neuen Einsichten vor. Wenn Sie viele Veränderungen erwarten, stellen Sie sich frühzeitig darauf ein.
- Achten Sie darauf, dass Vorgehensweisen im Projekt nicht falsch etikettiert werden. Nur weil »agil« drauf steht, können sich trotzdem falsch verstandene Modelle dahinter verbergen.
- Machen Sie es sich nicht zu leicht. Es wäre einfach, eine extreme Position zu »klassisch« und »agil« einzunehmen. Das polarisiert aber und baut unnötige Feindbilder auf.
- Bleiben Sie objektiv. Die Berichterstattung ist (immer noch) sehr einseitig. Agilität gilt als »gut«, andere Methoden werden diskreditiert und für das Scheitern verantwortlich gemacht.

Ist Scrum etwas für uns?

Annäherung an eine neue Idee

Scrum will help you to fail in thirty days or less.

Ken Schwaber, Softwareentwickler

Im April 2019 wird die 32-Millionen-Dollar-Klage des US-amerikanischen Autovermieters Hertz gegen seinen IT-Dienstleister Accenture publik. Die digitalen Pioniere des t3n-Magazins titelten seinerzeit: »Accenture blamiert sich mit Redesign einer Website«. Die 16-seitige Anklageschrift lässt tief blicken. Sicherheitslücken, fehlende Features und verpasste Deadlines – beim Redesign der Website für den Autovermieter sind der renommierten Beraterfirma Accenture offenbar viele Fehler unterlaufen. Nun wollte der Kunde sein Geld zurück und klagte auf Entschädigung: mehr als 32 Millionen US-Dollar, und zusätzlich weiteres Geld, das benötigt wird, um das von Accenture hinterlassene Chaos aufzuräumen.

Bereits im August 2016 hatte Hertz dem IT-Dienstleister den Auftrag erteilt. Die Anforderungen klangen nicht allzu herausfordernd: Ein neuer Onlineauftritt sollte es sein, dazu eine passende App für das Smartphone. Spätestens im Dezember 2017 sollte alles fertig sein. Diese erste Deadline wurde jedoch ebenso verfehlt wie eine zweite (Januar 2018) und dritte (April 2018). Offenbar wurde bis zum Zeitpunkt der Klage im April 2019 keines der beiden Projekte an den Kunden ausgeliefert.

Die Anklageschrift beginnt mit einem Rückblick auf die Zielsetzung des Projektes: »*Anfang 2016 startete Hertz das ambitionierte Projekt, seine digitale Identität zu transformieren. Das Hauptziel des Projekts ist, die Customer Experience auf den digitalen Plattformen von Hertz neu zu definieren, indem eine marktführende Website und ergänzend mobile Anwendungen entwickelt werden …*«

In dieser Aussage deutet sich schon an, was später zum Scheitern des Projekts führen sollte: Die vermeintliche Anpassung auf Kundenbedürfnisse wuchs sich zu einem Mammutprojekt aus. Eigentlich hätte der Autovermieter wissen können, dass man in der Softwareentwicklung keine Millionenbeträge mehr in traditionelle Wasserfallprojekte investiert, sondern besser auf agile Projektmanagementmethoden wie Scrum setzt. Es mag verlockend sein, Schadenfreude darüber zu empfinden, dass auch große Firmen wie Hertz und Accenture ein millionenschweres IT-Projekt in den Sand setzen können. Aber das einzige, was das Scheitern dieses Projektes besonders macht, ist die Tatsache, dass es auf so spektakuläre Weise publik wurde.

Wie im Falle des Autovermieters wird bei Projekten mitunter viel geplant und dokumentiert, aber am Ende werden die Ziele trotzdem nicht erreicht. Auf den ersten Blick mag das überraschen: Die bisherige Website des Autovermieters war ja schließlich auch durch eine gute Planung erfolgreich online gegangen. Warum sollte das gleiche Prinzip nicht auch beim Relaunch greifen? Der Grund liegt in einer stetig steigenden Komplexität der Anforderungen, die es verlangt, schneller und flexibler auf Veränderungen zu reagieren.

Scrum und andere Methoden des agilen Projektmanagements versprechen da Abhilfe. Der Kerngedanke dahinter ist denkbar einfach: Wenn die ausgeklügelten Methoden, Werkzeuge und

Vorgehensweisen des traditionellen Projektmanagements in die Sackgasse führen, lässt man sie doch einfach sein – und vertraut darauf, dass ein erfahrenes Projektteam die Sache schon irgendwie schaukeln wird. Einfach mal machen lassen!

Selbstorganisation statt fester Strukturen

Wenn herkömmliche Methoden nicht funktionieren, warum es dann nicht auf eine ganz andere Weise versuchen? Diese Frage beschäftigte die beiden US-Amerikaner Ken Schwaber und Jeff Sutherland. Herausgekommen ist ein neues Vorgehensmodell – die Scrum-Methode.

Der englische Begriff »Scrum« stammt aus dem Rugby und bedeutet übersetzt soviel wie »dichtes Gedränge«. Gemeint ist das Gedränge, das entsteht, wenn sich im Rugby die Spieler um den Ball versammeln. Hinter der Rugbymetapher steht die bittere Erkenntnis, dass Einzelkämpfer in starren Strukturen mit ihren festen Plänen den heutigen Anforderungen von Schnelligkeit und Flexibilität nicht mehr gewachsen sind. Wenn hingegen ein Team gemeinsam als Einheit einen Weg zurücklegt und sich dabei wie beim Rugby den Ball hin- und herspielt, besteht eine gute Aussicht auf Erfolg.

Ken Schwaber und Jeff Sutherland stellten 1995 auf der OOPSLA, einer jährlich abgehaltenen Forschungskonferenz, erstmals eine Definition von Scrum vor. Im Kern beinhaltet sie, starre Projektpläne durch einfache Rollen, Regeln und Methoden der Zusammenarbeit zu ersetzen.

> Scrum ist ein Rahmenwerk, innerhalb dessen Menschen komplexe adaptive Aufgabenstellungen angehen können. Es versetzt sie in die Lage, produktiv und kreativ Produkte mit dem höchstmöglichen Wert auszuliefern.

Scrum wird heute meist von Softwareentwicklern genutzt, doch das Prinzip und die gewonnenen Erkenntnisse treffen auf alle Arten von Teamwork zu. Darin liegt einer der Gründe, weshalb die Methode so beliebt ist.

Man darf Scrum nicht rein mechanisch als Prozess verstehen, dem man blind zu folgen hat. Im Unterschied zum klassischen Projektmanagement, das sich als fester Prozess versteht, bietet Scrum ein Framework an – also einen Rahmen, der es ermöglicht, die agilen Prinzipien in der Projektarbeit zu leben. Die ursprüngliche Definition kann man deshalb wie folgt abwandeln:

> Scrum ist ein Rahmenwerk für die Zusammenarbeit von Teams. Es basiert auf einer Definition von Rollen, Meetings und Werkzeugen, die einem Team Struktur und einen klar definierten Arbeitsprozess geben. Dieser wiederum basiert auf agilen Prinzipien.

Diese Abkehr von einem fest vorgegebenen Prozess hat vor allem einen entscheidenden Vorteil: Man bleibt offen gegenüber neuen Einsichten oder Ideen – den eigenen genauso wie denen des Kunden. Anstatt vorab definierte Pläne abzuarbeiten, ermöglicht es das Scrum-Prinzip, Prioritäten zu ändern und damit auch das Projektergebnis anzupassen.

Neue Abläufe – so einfach ist Scrum

Der Kern der Scrum-Methode lässt sich so beschreiben: Das Team kann sich selbst organisieren. Seine Mitglieder setzen sich interdisziplinär zusammen, damit möglichst viele Kompetenzen abgedeckt werden. Kennzeichnend für den Ablauf sind kurze, vorhersagbare Entwicklungszyklen, an deren Ende jeweils eine neue funktionierende Version des Produkts steht. Auf diese Weise steigert man Schritt für Schritt nicht nur die Qualität des Produkts, sondern gewinnt auch das Vertrauen des Kunden.

Die Scrum-Methode kommt mit nur wenigen Regeln aus. Im Gegensatz zur Steuererklärung passt der generelle Ablauf tatsächlich auf einen Bierdeckel, wie Niklas dem erstaunten Tom an der Hotelbar demonstrierte (Abbildung 3).

Am Anfang steht eine Vision des Produkts, das entstehen soll – etwa einer neuen Steuerungssoftware wie im Projekt von Tom. Aus der Vision werden konkrete Eigenschaften des Produkts abgeleitet, die der »Product Owner« im »Product Backlog« auflistet und priorisiert. Da er für den ökonomischen Erfolg des Produkts verantwortlich ist, geht er dabei nach wirtschaftlichen Gesichtspunkten vor. Das heißt vor allem: Er legt fest, in welcher Reihenfolge die Einträge im Product Backlog abgearbeitet werden.

Die Produktentwicklung erfolgt iterativ, das heißt in festen Abschnitten von gleicher Länge, meist zwei bis vier Wochen. Diese Abschnitte werden »Sprints« genannt, die sich quasi die Klinke in die Hand geben: Das Ende des einen bildet den Anfang des nächsten Sprints.

Für die Umsetzung der im Product Backlog gesammelten Produkteigenschaften ist das »Entwicklungsteam« zuständig. Der Product Owner bestimmt somit, *was* bei einem Sprint entwickelt werden soll, während sich das Entwicklungsteam darum kümmert, *wie* es entwickelt wird.

Damit dieses Zusammenspiel klappt, braucht es eine enge Abstimmung zwischen dem Product Owner und dem Entwicklungsteam. Zu Beginn jedes Sprints gibt es deshalb eine »Sprintplanung«, bei der die anstehenden Aufgaben diskutiert und der voraussichtliche Aufwand geschätzt werden. Moderiert wird die Planungssitzung vom »Scrum Master«, dessen Aufgabe es ist, das Team und den Product Owner zu begleiten und zu unterstützen.

Auf Grundlage der Sprint-Planung trifft der Product Owner eine Kosten-Nutzen-Abwägung und priorisiert die anstehenden Aufgaben. Damit ist es möglich, die Aufgaben mit der höchsten Priorität in die Aufgabenliste des nächsten Sprints, das sogenannte »Sprint Backlog«, zu übertragen: Der Product Owner gibt die Reihenfolge der Aufgaben vor, und das Entwicklungsteam entscheidet, wie viele dieser Aufgaben in den nächsten Sprint passen. Das Team bestimmt auch, wie die Aufgaben konkret umgesetzt werden sollen.

Nun startet der Sprint. Das Team muss in der vorgegebenen Zeit die im Sprint Backlog aufgelisteten Aufgaben abarbeiten. Die Teammitglieder gehen dabei weitgehend selbstorganisiert vor, entscheiden zum Beispiel eigenständig darüber, wer als nächstes welche Aufgabe angeht. Unterstützt werden sie dabei vom Scrum Master. Seine Funktion ist es, dem Team bei Unklarheiten und Hindernissen beizustehen und dafür zu sorgen, dass es fokussiert auf das Sprint-Ziel hinarbeitet.

Zu Beginn eines Arbeitstages trifft sich das Entwicklungsteam

zu einem kurzen, maximal viertelstündigen Meeting, dem »Daily Scrum«. Um nicht abzuschweifen, sondern sich wirklich auf die wichtigen Punkte zu konzentrieren, wird das Treffen bevorzugt im Stehen abgehalten. Das Team bespricht kurz den Fortschritt im Sprint und organisiert die Arbeit für den bevorstehenden Tag. Das Daily Scrum dient damit vor allem der Selbstorganisation im Team. Darüber hinaus hat es die Funktion, Teamgeist und Motivation zu fördern, indem die Teilnehmer über ihre Fortschritte berichten und sich gegenseitig durch ihre Erfolge bestärken.

Während das Entwicklungsteam den aktuellen Sprint absolviert, wendet sich der Product Owner wieder dem Product Backlog zu und bereitet schon den nächsten Sprint vor. Dabei stimmt er sich im »Refinement Meeting« auch mit dem Entwicklungsteam ab, sodass die anstehenden Aufgaben vor der nächsten Sprintplanung bereits einmal durchdacht und für die Teammitglieder nicht völlig neu sind. Zudem steht der Product Owner dem Entwicklungsteam für Rückfragen zur Verfügung. Möglicherweise priorisiert er auch Aufgaben neu, sofern sich während des Sprints neue Erkenntnisse ergeben und eine Umplanung nahelegen.

Der Sprint schließt mit dem »Sprint Review«, bei dem das Entwicklungsteam ein funktionsfähiges Zwischenprodukt präsentiert – das »Product Increment«. Ganz gleich ob der Product Owner diesen Zwischenstand noch zurückhalten oder bereits ausliefern möchte: In jedem Fall muss die Qualität des Product Increment so gut sein, dass es ausgeliefert werden könnte.

Zum Sprint Review sind neben dem Product Owner auch relevante Stakeholder wie etwa Kunden, Anwender oder das Management eingeladen. Das Team stellt die neuen Produkteigenschaften vor und erhält von den Anwesenden ein Feedback zu den erledigten Aufgaben. Auch für den Product Owner ist das Feedback der Stakeholder wichtig, denn in der Regel führt es auch zu Änderungen im Product Backlog: Es entstehen neue Einträge, existierende Einträge werden neu priorisiert, erledigte Aufgaben entfernt.

Nach dem Sprint-Review findet noch die »Sprint Retrospektive« statt, in der alle Beteiligten den letzten Sprint Revue passieren lassen und überlegen, wie sie die Zusammenarbeit und den Entwicklungsprozess weiter verbessern können. Im Anschluss daran beginnt bereits der nächste Sprint – mit der Sprint-Planung.

Die fehlende Einbindung des Kunden

Sieht man sich den Scrum-Ablauf an, fällt vor allem eine Besonderheit auf: das intensive Feedback, das die Entwickler erhalten. Die Scrum-Methode setzt auf eine echte Zusammenarbeit mit dem Kunden. Nach jedem Sprint gibt der Kunde den Entwicklern die Rückmeldungen, die sie für den Fortgang ihrer Arbeit brauchen.

Genau hier dürfte ein wesentlicher Grund liegen, an dem Projekte wie das von Hertz und Accenture scheitern. Zwar werben große Beratungsfirmen wie Accenture gerne mit einer kontinuierlichen Beziehung zu ihren Kunden, doch im Falle dieses Projekts scheint es genau daran gefehlt zu haben. Während der mehrjährigen Projektlaufzeit konnte Accenture mehrfach die Erwartungen der Kunden nicht erfüllen. Die Klageschrift wirft dem IT-Dienstleister zum Beispiel vor, die notwendigen Tests der

Software vernachlässigt zu haben. Die Scrum-Methode mit ihren regelmäßigen Sprint-Reviews hätte geholfen, sich enger miteinander abzustimmen und das Desaster womöglich zu verhindern.

Die Ursache für das Scheitern alleine bei Accenture zu suchen, wäre zu einfach. Um den Projekterfolg abzusichern, hätte der Auftraggeber zumindest die Rolle des Product Owners definieren und durch eigene Leute besetzen sollen. Da Hertz das versäumte, lag die Verantwortung für Produkteigenschaften nicht in der eigenen Hand, sondern in der des Dienstleisters. Die Folge davon war eine unzulängliche Kommunikation zwischen Hertz und Accenture, was eine der Hauptursachen für das Scheitern des Projekts gewesen sein dürfte. Denn hätte der Autovermieter die Verantwortung für die Produkteigenschaften innegehabt, hätte er sich zwangsläufig mit den Entwicklern von Accenture auseinandersetzen müssen – und es wäre ein häufiger und enger Kontakt zustande gekommen.

Auf den Punkt gebracht lässt sich das Relaunch-Desaster an einem Begriff festmachen, der aus dem klassischen Projektmanagement stammt: dem »Big-Bang-Deployment«. Bei einer Big-Bang-Strategie wird die neue Software vollständig zu einem klar definierten Zeitpunkt in Betrieb genommen. Hertz und Accenture hatten den Big Bang geplant – zu dem es dann nie kam. Und egal, ob Hertz ihn gefordert hat oder nicht, Accenture hätte sich niemals darauf einlassen dürfen. Es wäre sinnvoller gewesen, nach dem Scrum-Prinzip zu verfahren: in kurzen Iterationen Produkteigenschaft für Produkteigenschaft zu entwickeln und kontinuierlich auszurollen.

Scrum ist leicht, aber nicht einfach

Scrum wirbt damit, leichtgewichtig zu sein und auf den Ballast »traditioneller« Vorgehensweisen in Projekten zu verzichten. Dass sie sich auf einem Bierdeckel skizzieren lassen, ist ein zentraler Aspekt agiler Prozesse und Methoden: Sie sind einfach zu verstehen und können von jedem Team nach einer kurzen Eingewöhnungsphase umgesetzt werden. Gleichzeitig ist Scrum in gewisser Weise auch streng und verlangt von allen Beteiligten viel Disziplin. Wenn beispielsweise nicht alle Elemente von Scrum eingesetzt werden, sondern nur die »bequemen« Teile, verliert die Methode schnell ihre Durchschlagskraft.

Die Leichtigkeit, mit der Scrum daherkommt, mag zwar verlockend sein. Die Einführung und Integration dieser Methode in eine bestehende, durch hierarchische Strukturen definierte Organisation sind dagegen alles andere als einfach. Sie erfordern die Bereitschaft aller Beteiligten, Prozesse, Arbeits- und Denkweisen im Unternehmen konsequent zu verändern.

Ein Blick in Toms Tagebuch

Mittwoch, 7. Februar

Heute Morgen bin ich früh aufgewacht. Ich habe gut geschlafen. Neben mir auf dem Nachttisch lag der Bierdeckel, auf dem Niklas gestern Abend den »Scrum Flow« skizziert hat. Noch im Bett liegend griff ich nach ihm und dachte erneut darüber nach.

Eines ist klar: Egal, welches Projekt ich bisher gemanagt habe – der Weg von einer Idee bis zu ihrer Realisierung hat bei uns immer mehrere Monate, manchmal sogar Jahre gedauert. Und am Ende gab es immer wieder Ärger, weil der Kunde die Software so nicht nutzen wollte oder mit ihr unzufrieden war. Das ist ja auch kein Wunder. Wenn wir Monate brauchen, um ein Produkt auf den Markt zu bringen, dann ist es doch schon veraltet, noch bevor es richtig in Betrieb genommen wird. Gott sei Dank hat die Konkurrenz das gleiche Problem, sonst wären wir mit unserer Steuerungssoftware längst weg vom Fenster …

Niklas hat mir klar gemacht, dass wir da etwas ändern müssen!

In unserem Unternehmen erfassen wir das Kundenfeedback über separate Abteilungen – entweder über das Callcenter oder über den Vertrieb. Das Problem ist nun: Dieses Wissen um die Kundenbedürfnisse findet keinen Eingang in unsere Steuerungssoftware! Deshalb werfen wir erst mal die gesamte Maschinerie des Projektmanagements an: Wie könnte eine (neue) Lösung aussehen? Wie sieht der (veränderte) Projektplan aus? Wann stehen die (schon verplanten) Ressourcen zur Verfügung? Wer entwickelt wann die notwendige Softwareänderung?

Ideen und Umsetzungspläne müssen dann von allen möglichen Fachabteilungen abgesegnet und genehmigt werden. Dahinter steckt das Selbstverständnis, dass nur ein gut durchdachtes Produkt beziehungsweise Projekt erfolgreich sein kann. Und ich gebe es ja offen zu, auch ich habe bisher so gedacht – schließlich hat man das als Projektleiter ja so gelernt. Dieser Weg kostet allerdings unglaublich viel Zeit. Zwischen einer Idee und dem Beginn der Umsetzung liegen auf diese Weise schnell mehrere Wochen. Und bis der Plan dann tatsächlich umgesetzt ist, sind bereits Monate, schlimmstenfalls Jahre vergangen.

Einige wichtige Erkenntnisse:

- Wir sorgen im Unternehmen durch unsere Organisationsstrukturen praktisch selbst dafür, dass Informationen über veränderte Kundenwünsche sehr spät oder schlimmstenfalls gar nicht in der Softwareentwicklung ankommen.
- Wir sollten als Firma in der Lage sein, aus Kundenfeedback erheblich schneller Änderungen für unsere Software abzuleiten.
- Das schnellste Feedback würden wir als Team erhalten, wenn der Softwarenutzer selbst Teil des Projektteams wäre.
- Die Wahrscheinlichkeit, dass ein Softwareprodukt nicht zu den Bedürfnissen des Kunden passt, steigt mit der Dauer, die es braucht, um das Produkt zu entwickeln.

Vielleicht treffe ich Niklas gleich noch beim Frühstück. Ich möchte mich bei ihm für die zahlreichen Impulse und hilfreichen Erläuterungen der letzten Tage bedanken. Anschließend noch kurz in die Zentrale und dann ab nach Hause, bevor hier in der Stadt der närrische Ausnahmezustand beginnt.