

A sensible PRINCE2 practitioner should pick the best product development approach for each product or team in their project's scope.

Use the Burn Charts or Earned-Value and call planning 'Rolling Wave', 'Stages' or 'Release and Sprint' depending on what keeps everyone happy.

Also focus on 'value (agile label)' or 'benefits (PRINCE2 label)'.

A sensible PMBoK user and agile proponent should recognise that PRINCE2 brings useful structure to interface to the rest of the organisation's deep-rooted constraints in hierarchy, annualised budgeting, matrix resource structures, routine reporting and dislike of change.

Artikel

Artikel unserer Goldsponsoren

Hybrides Projektmanagement für agile Projekte in mittelständischen und großen Unternehmen

Rainer Wendt, masVenta Business GmbH

Schon seit vielen Jahren dauert eine Art Grabenkrieg zwischen den Vorgehensweisen im Projektmanagement an. Immer wieder wird die Schwarz-Weiß-Frage gestellt: Wird das Projekt klassisch oder agil abgewickelt? Leider führt diese Frage genau in die falsche Richtung!

In diesem Artikel erfahren Sie, warum klassische und agile Vorgehensweisen in Projekten nicht getrennt oder als konkurrierende Methoden, sondern als

Komponenten eines ganzheitlichen Ansatzes betrachten werden sollten. Hierzu berichte ich von meinen eigenen Erfahrungen als Projektmanager in mittelständischen und großen Unternehmen.

Der große Fehler

In den meisten mittelständischen und großen Unternehmen ist Projektmanagement etabliert, es gibt Vorgehensmodelle wie das V-Modell, Projektmanager, die nach PRINCE2- oder PMI-zertifiziert sind, und ein



Abb: Agile Vorgehensweisen in Projekten werden zum Standard

Projektmanagement-Handbuch mit klar beschriebenen und einzuhaltenden Vorgehensweisen. Vorlagen für zu liefernde Dokumente wie Kosten/Nutzen-Rechnung, Projektauftrag und -plan sind erprobt und ausgereift. Immer wieder habe ich in den letzten Jahren erlebt, dass diese klassischen Vorgehensweisen, auch als Wasserfall-Vorgehen bezeichnet, nicht ohne weiteres geändert oder erweitert werden konnten. Die Prozesse für das Projektmanagement sind meist vom Qualitätsmanagement oder der Project Management Office (PMO) offiziell verabschiedet worden und galten fortan unternehmensweit, Ausnahmen hatten eher einen „inoffiziellen“ Charakter. Wollte ein Projektmanager von diesen Regeln abweichen, indem er z.B. die Entwicklung von Prototypen eines Produktes zu erstellen bevor die vollständigen Anforderungen genehmigt worden waren, galt das als Regelabweichung, für die er vom Steuerungskreis oder seinem Linienvorgesetzten grünes Licht benötigte. Andernfalls hätte man ihm bei einem späteren Scheitern des Projektes den Vorwurf machen können, dass dies ggfs. verursacht worden sei durch die Abweichung von der Standardvorgehensweise.

In die Schublade der Regelabweichungen bzw. der Alternativen zu den klassischen Vorgehensweisen wurden üblicherweise auch die agilen Methoden gesteckt. Der Agilität haftete besonders in der frühen Phase der neunziger Jahre der Makel an, nicht regelkonform zu sein. Der Klassiker ist hier wohl das Klischee der nicht vorhandenen Dokumentation.

Durch die immense Popularität und Bedeutung, die die agilen Vorgehensweisen im Laufe der letzten zwanzig Jahre aus guten Gründen (s.u.) erlangt haben, mussten sie schlussendlich einen Platz in den Vorgehensweisen des Qualitätsmanagements und der PMOs großer Unternehmen finden. Leider haben jedoch viele Unternehmen die agilen Vorgehensweisen nicht in die bestehenden integriert sondern *neben* den klassischen Vorgehensweisen angesiedelt, so dass Projektmanager und –beteiligte stets entscheiden müssen, ob sie ein Projekt klassisch *oder* agil abwickeln. Da viele Projektmanager bevorzugen, das zu machen, was sie gelernt haben, vermeiden sie normalerweise unnötige Risiken wie die eines neuen Vorgehensmodells. Hinzu kommt, dass agile Elemente oft nur unzureichend eingeführt worden sind, so dass sie schnell in den Verruf kamen, nicht zu funktionieren. Ich habe selbst schon sehr erfahrene Architekten und Entwickler kennengelernt,

die User Stories als unzureichend betrachteten und sie deshalb nicht mehr einsetzten. User Stories waren in diesem Unternehmen sozusagen verbrannt!

So bleiben viele Projektmanager bei den klassischen Vorgehensweisen, wenn sie sich zwischen klassisch und agil entscheiden müssen. Das wird unterstützt durch die Tatsache, dass in den prägenden Papieren der populärsten agilen Methode Scrum von Ken Schwaber und Jeff Sutherland von Projektmanagern gar nicht die Rede ist. Oft hört man sogar die aus meiner Sicht vollkommen falsche Schlussfolgerung, dass agile Vorgehensweisen keine Projektmanager mehr benötigen.

Der große Fehler liegt also im Schubladendenken, in der Annahme, dass ein Projekt entweder agil oder klassisch zu sein hat.

Hybrides Vorgehen als Erfolgsfaktor

Unter hybridem Projektmanagement versteht man im Allgemeinen eine sinnvolle Mischung der agilen und klassischen Vorgehensweisen, angepasst an das Unternehmen und sogar angepasst an das partikuläre Projekt. Durch einen projektindividuellen Zuschnitt der Vorgehensweisen, man spricht hier auch von „Tailoring“, erreicht man die höchste Effizienz im Projekt, man macht die Dinge also genau *richtig*, mit diesem Team, in dieser Situation. Hierzu nimmt man das Beste aus beiden Welten und kombiniert es mit größter Sorgfalt, so wie ein guter *Blend Whisky* aus verschiedenen *Single Malt Whiskys* hergestellt wird. Dem prozessorientierten Projektmanager alter Schule wird bei einem solchen Statement – ähnlich dem Whisky-Puristen – angst und bange, schließlich klingt das danach, dass jedes Projekt unterschiedlich ist, dass Standards im Unternehmen aufgelöst werden. Hybride und angepasste Vorgehensweisen sind jedoch die beste Formel für erfolgreiche Projekte, so wie auch die Blends unter den Whiskys zu den erfolgreichsten Spirituosen zählen, um das gewählte Beispiel noch etwas weiter zu strapazieren.

Makro- und Mikromanagement im Projekt

Die Kunst bei der Einführung hybrider Vorgehensweisen liegt in der Beibehaltung bewährter Prozesse im Makromanagement, kombiniert mit grundlegenden Änderungen im Mikromanagement. Konkret bedeutet das, dass das Berichtswesen eines Projektes z.B. in das

übergeordnete Programm weitestgehend erhalten bleibt, während die Entwicklung sich radikal neu organisieren kann. Oft geht gerade von der Entwicklung das starke Bedürfnis aus, sich neu zu organisieren und agile Vorgehensweisen einzuführen, da die Vorteile für das Unternehmen auf der Hand liegen.

Das agile Mikromanagement

Die hier wohl radikalste Änderung bei der Einführung von agilem Vorgehen ist die im Gegensatz zur klassischen Vorgehensweise viel schlankere Spezifikation der Anforderungen, die zunächst nur grobe Geschäfts- und Stakeholderanforderungen auf hohem Level adressiert. Hier ein typisches Beispiel für eine sogenannte User Story, der Kontext sei ein CRM-System:

„Als Verkäufer möchte ich Umsätze und Verkaufschancen eines Kunden auf einen Blick sehen, um den Wert und das Potential eines Kunden schnell beurteilen zu können“.

Diese User Story enthält viele Informationen und lässt gleichzeitig viel Spielraum für die Implementierung einer Lösung. Das ist gut und schlecht zugleich, hier zunächst die

Vorteile der User Story:

- Die User Story beschreibt den konkreten Bedarf der Stakeholder, ist als Stakeholder Requirement sauber und strukturiert in das Anforderungsmanagement einzuordnen. Eine damit verbundene Geschäftsanforderung auf noch höherem Level, ein sog. Business Requirement, könnte lauten „Als Verkäufer möchte ich den Wert meiner Kunden beurteilen können“. Insofern beinhaltet die o.g. Story schon erste Aspekte einer abstrakten Lösung.
- Die User Story schreibt aber keine konkrete Lösung vor, sondern überlässt es den Analysten und Entwicklern, die beste und wirtschaftlichste Lösung im Kontext der Projektsituation (Dauer, Kosten, Qualität) zu finden
- Die User Story ist konkret genug, um ihre Priorität festlegen zu können und damit zu entscheiden, *wann* sie implementiert wird

- Die User Story konkret genug, um den Aufwand für die Implementierung grob abschätzen zu können (Dauer und Kosten)
- Die User Story ist leicht zu verstehen, sie kann ohne Lernaufwand oder Expertenwissen mit dem Fachbereich diskutiert und priorisiert werden.
- Die User Story kann durch einfache grafische Darstellungen und Modelle illustriert werden, was das Verständnis und die Diskussionen nochmals signifikant vereinfacht für beide Seiten, den Fachbereich und das umsetzende Entwicklungsteam.
- Und zum Schluss der vielleicht größte Vorteil: die User Story ist klein genug, um bereits nach einer kurzen, mehrwöchigen Iteration in der Entwicklung ein Produkt anfertigen zu können, das der Fachbereich real ausprobieren kann bzw. muss. Man spricht hier auch von einem Minimum Viable Product (MVP). Gefällt das Ergebnis dem Fachbereich nicht, wird die Lösung angepasst, die User Story in einer nachfolgenden Iteration ganz oder teilweise neu entwickelt. Das Risiko der Fehlentwicklung ist also sehr überschaubar, der maximale Schaden beträgt höchstens „ein paar Wochen“. Das ist der Kernvorteil der agilen Vorgehensweise.

Die Nachteile der User Story:

- Die User Story ist nicht präzise genug, sondern wirft viele Fragen auf. Eine naheliegende Frage in unserem Beispiel wäre „Was ist Definition von ‘schnell’?“. Keiner kann sagen, was der Autor der User Story sich unter ‘schnell’ vorstellt. Es gibt noch viele weitere Unklarheiten, dazu mehr im Verlauf des Textes.
- Der Vorteil, dass die User Story keine Lösung vorschreibt, kann sich in einen Nachteil umkehren. Falls die Stakeholder schon eine konkrete Erwartungshaltung bzgl. der Gestalt der Lösung haben, führt eine vollkommen andere Lösung womöglich zu Enttäuschungen, und das selbst für den Fall, dass die andere Lösung viel ‘besser’ als die erwartete Lösung ist. Hier stellt sich übrigens eine ähnliche Frage wie im ersten Punkt, was bedeutet ‘besser’?
- Wird die User Story nicht hinreichend besprochen und damit verstanden, besteht das Risiko, dass eine

funktional und qualitativ mangelhafte Lösung geliefert wird.

Das Rezept für erfolgreiche agile Vorgehensweisen ist mit diesen Erkenntnissen vom Prinzip her denkbar einfach:

1. Die Vorteile im Sinne von Chancen verstärken
2. Die Nachteile im Sinne von Risiken abschwächen oder sogar vermeiden

Wie kann das erreicht werden? Die Antwort lautet auch hier, dass eine zugeschnittene Vorgehensweise, die im Unternehmen und in der konkreten Umgebung funktioniert, der Schlüssel zum Erfolg ist. Hier ist ein Coaching durch einen erfahrenen Berater empfehlenswert, der mit den einflussreichsten Stakeholdern agile Prinzipien und Regeln im Unternehmen verankert. Auch dies möchte ich durch unser Beispiel unterstützen. Wie wir oben festgestellt haben, ist unsere User Story nicht präzise genug und wirft viele Fragen auf. Dagegen muss etwas getan werden.

Darum kümmern wir uns genau dann, wenn wir es müssen, also erst kurz vor oder zu Beginn einer Iteration. Im Dialog mit dem Fachbereich werden dazu für jede User Story sogenannte Akzeptanzkriterien erstellt, mit Hilfe derer das Akzeptieren der Software in einer Vorführung nach der Iteration (z.B. Sprint Demo oder Sprint Review genannt) nach vorher vereinbarten Kriterien und Regeln verläuft. Der Fachbereich überlegt bei der Erstellung der Kriterien ganz genau, welchen Akzeptanzkriterien zugestimmt werden kann und wie scharf z.B. Qualitätskriterien formuliert werden müssen. Das heißt, dass man sich im Fachbereich mit den funktionalen und nichtfunktionalen Requirements und der Entwicklung auseinandersetzt, man nimmt sich die Zeit dieses oder jenes genauer zu verstehen und in die Tiefe zu gehen. Alleine dieser Dialog, die Auseinandersetzung mit sinnvollen Akzeptanzkriterien, ist Gold wert; führt er doch dazu, dass man über die Produkte spricht, mindestens vor und nach jeder Iteration, d.h. mindestens alle 2 bis 4 Wochen. Üblicherweise sind Akzeptanzkriterien viel umfangreicher als die zugehörigen User Stories. Während diese meist aus einem Satz bestehen, können die passenden Akzeptanzkriterien schnell eine Seite füllen.

Bezogen auf unser Beispiel würde man in die Akzeptanzkriterien auch eine nicht-funktionale Anforderung einbauen, die beschreibt, was 'schnell' bedeutet, z.B. dass innerhalb von 3 Sekunden die Information zur Verfügung stehen muss. Ich habe in meinen agilen Projekten in der Rolle des Projektmanagers die Regel aufgestellt, dass die Akzeptanzkriterien für eine Iteration spätestens am 5. Tag der Iteration vom Fachbereich genehmigt sein müssen, ansonsten kann die Entwicklung gestoppt werden. Diese sehr strikte Regel sorgt dafür, dass man die Akzeptanzkriterien erstellen *muss*, was wiederum dafür sorgt, dass der Dialog zwischen Fachbereich und dem Entwicklungsteam sehr lebendig bleibt. Gerade die Kompromisslosigkeit bzgl. der Einhaltung dieser Regel hat unserem Team viel Lob und Anerkennung beschert, da die Sprint Review Meetings sehr geschmeidig abliefen und eine hohe initiale Qualität erreicht wurde. Schließlich wussten beide Seiten genau, mit welchen Bedingungen man zu einem 'Done', also zur hinreichenden Erfüllung aller Akzeptanzkriterien, kommen konnte.

Das große Risiko der klassischen Vorgehensweisen, in denen zunächst alle Anforderungen detailliert spezifiziert werden und erst nach Vollendung und Genehmigung aller Anforderungen mit der Entwicklung begonnen wird, ist zweifelsohne die schwache Einbindung des Fachbereiches in die Prozesse der Entwicklung. In großen Projekten sieht der Fachbereich oder der Endnutzer oft monatelang keine Zwischenergebnisse, muss vertrauen und glauben, dass am Ende schon alles passen wird. Und wenn es dann nicht passt, was sehr oft der Fall ist, wird es sehr teuer. Der CHAOS Report des renommierten US-amerikanischen Standish Institutes beziffert die Projekte, die aus dem Ruder laufen, immer noch auf über 50%. Agilität ist die Antwort auf dieses hohe Risiko und hat in den letzten 20 Jahren eine beispiellose Entwicklung durchlaufen. Während es in den Neunzigern noch als exotisch galt, agil zu entwickeln, sind diese Vorgehensweisen heutzutage Standard und in vielen Branchen schon deutlich in der Überzahl.

Das klassische Makromanagement

Das Management auf der übergeordneten Programm- oder Portfolioebene soll – wenn möglich – für alle darin enthaltenen Projekte nach den gleichen Regeln ablaufen. Hierzu zählen z.B. Standards für das

Berichtswesen, Kosten/Nutzen-Berechnungen und Möglichkeiten zur Eskalation, sollten Projekte in Schieflage geraten. Typische Gremien wie der Projektsteuerungskreis und eine Genehmigungsinstanz für Änderungen im Projekt (Change Request Board) sind für den Projektleiter meist bindend, inklusive der hier erwarteten Dokumente und Daten. Während agile Teams die Reporting-Schnittstellen wie den Projektfortschrittsbericht im Detail noch ganz passabel bedienen können (sie verwenden hier Burn-Down- und Burn-Up-Charts, die die fertiggestellten User Stories grafisch oder tabellarisch darstellen), ist eine Gesamtbetrachtung des Fortschritts nach den etablierten Prinzipien des Earned Value Managements (EVM) schwieriger, da der finale Funktionsumfang eines agilen Projektes sich erst am Ende des Projektes beziffern lässt. Hier zeigt sich der entscheidende Unterschied im Paradigma der Vorgehensweisen: agile Projekte legen vor dem Projekt die Projektdauer und das verfügbare Projektbudget fest, aber nicht den finalen Funktionsumfang, wie klassische Projekte. Agil hat deshalb den prinzipiellen Vorteil, dass Projekte „on-time“ und „on-budget“ abgeschlossen werden können. Aus Sicht des Managements ist das sicherlich hervorstechend, vor allem im Lichte der oben genannten Zahlen aus dem CHAOS Report. In klassischen Projekten versucht man hingegen zu Beginn des Projektes den Umfang bis auf Detailebene zu definieren und abzuschätzen um dann mit Hilfe dieser Schätzungen den Projektplan abzarbeiten. Änderungen bei Features und Funktionen oder fehlerhafte Schätzungen führen damit

zwangsläufig zu Änderungen des Projektplans, meist auch zu höheren Kosten und längerer Projektdauer.

Das oben angesprochene Change Control Board (CCB) hat in agilen Umgebungen nur noch selten Arbeit mit dem Projekt. Die meisten Änderungen im Projekt werden mit Hilfe der üblichen agilen Mechanismen eingearbeitet, Änderungen gelten als „normal“ und gehören zum Projektalltag. Es sei denn es handelt sich um kapitale Änderungen des Projektes, also solche, die dem Projekt eine vollkommen andere Richtung geben. Für solche Fälle muss es weiterhin eine Instanz wie den Steuerungskreis oder das CCB geben, die entscheidet, ob das Projekt neu ausgerichtet oder vielleicht sogar beendet wird, ggfs. zugunsten eines Nachfolgeprojektes.

Allerdings muss der Fachbereich oder der Endkunde bei agilen Vorgehensweisen damit einverstanden sein, dass sich der Umfang einer Lösung durch fortwährende Änderungen ebenfalls ändert – er wird kleiner, Features und Funktionen werden herunterpriorisiert und am Ende unter Umständen nicht mehr implementiert, weil Zeit und Budget erschöpft sind. Hier liegt die Crux des Paradigmenwechsels: die Features und Funktionen mit der geringsten Priorität schaffen es nicht in das finale Produkt, was jedoch meist nicht problematisch ist. Meine Erfahrung ist, dass der Fachbereich oder Endkunde damit einverstanden ist, wenn weniger wichtige Features, die eh sehr selten bis gar nicht verwendet werden, nicht produziert werden dafür aber die Kernfunktionalität den optimalen Nutzen bietet. Die fortlaufende, durch den Kunden gesteuerte Priorisierung

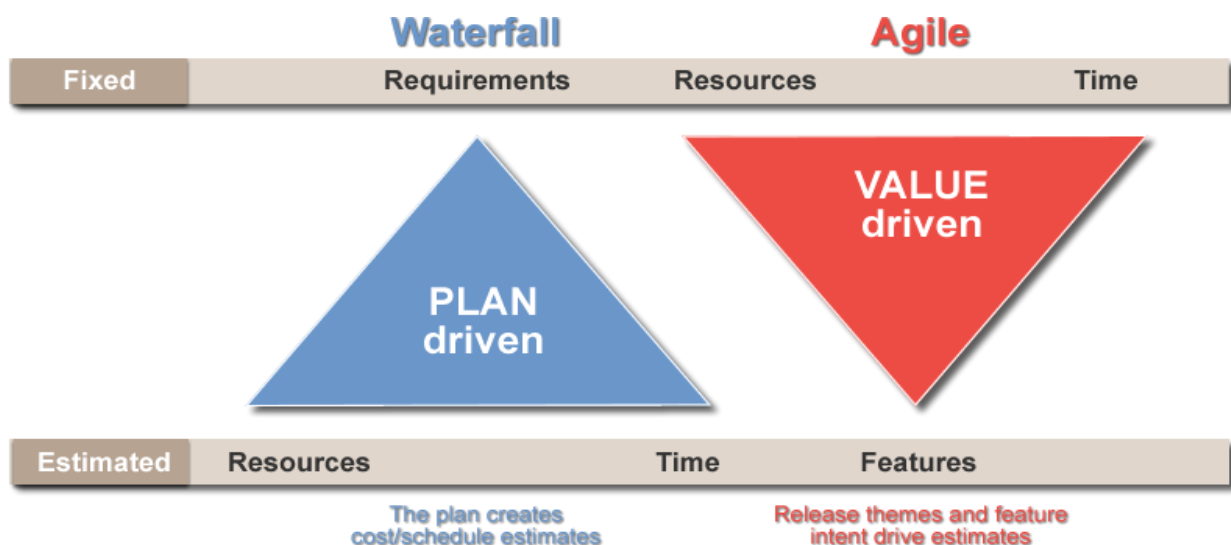


Abb: The Inverted Triangle zeigt die unterschiedlichen Paradigmen der klassischen und agilen Vorgehensweisen
Quelle: Ryan Matena, Ready for Agile, Projects@Work.com

der Anforderungen, stellt sicher, dass jeweils die Anforderungen mit der höchsten Priorität in der nächsten Iteration entwickelt werden, solange bis Zeit bzw. Ressourcen erschöpft sind. Will man auf bestimmte Low-Prio-Features nach Ende des Projektes partout nicht verzichten, gibt es immer die Möglichkeit ein zweites Projekt bzw. eine zweite Projektphase anzuschließen, für die dann natürlich neue Zeit und neues Budget gefunden werden müssen. Das passiert aus meiner Erfahrung in der Praxis relativ selten, oft können alle User Stories implementiert werden bzw. man zufrieden mit dem Erreichten und verzichtet auf 'Gold-Plating'.

Die Verbindung der beiden Welten

Wir erkennen hier bereits gut, dass die Koexistenz klassischer und agiler Elemente viel Sorgfalt erfordert. Nicht alles im Makromanagement kann so bleiben, wenn agile Vorgehen im Mikromanagement eingeführt werden, Zugeständnisse und Anpassungen sind erforderlich. So macht es beispielsweise keinen Sinn, detaillierte EVM Diagramme von agilen Projekten zu verlangen oder kleine Änderungen im CCB zu diskutieren. Stattdessen muss das Makromanagement den neuen Vorgehensweisen und vor allem natürlich den Personen in den Projekten Vertrauen entgegenbringen, und den Teams die Freiheit geben, sich entwickeln zu können. Weiterhin ist es wichtig, im Fachbereich oder beim Endkunden das Bewusstsein zu schaffen, dass sich das finale Produkt entwickelt in einer kontinuierlichen Kooperation und Kollaboration zwischen Kunde und Entwicklung, mit einem Ergebnis, das nicht zu 100% vorhersagbar ist. Diese inhärenten Eigenschaften agiler Projekte schränken die Eignung der Vorgehensweise natürlich ein. Ist von vorneherein klar, wie ein Produkt eines Projektes im Detail auszusehen hat, und kann dies auch gut geplant und abgeschätzt werden, ist die klassische Vorgehensweise der agilen vorzuziehen. Typische Beispiele sind Projekte aus dem Bauwesen, wie der Bau einer Brücke, oder andere, sehr hardwarelastige Projekte.

Letztendlich ist das Maß des Projekterfolges bestimmt durch den Nutzen, den die Produkte der Projekte für ihre Anwender bzw. den Endkunden bringen.

Fazit

Die Quintessenz aus diesen Überlegungen lautet, dass in bestehenden klassischen Strukturen Freiheiten für neue,

agile Strukturen geschaffen werden müssen, will man die immensen Vorteile der Agilität für das Unternehmen nutzen. In den agilen Prinzipien finden wir immer wieder Hinweise auf die Notwendigkeit der Empirik. Agile Teams müssen experimentieren und ständig aus Fehlern lernen, um ihre vollen Potentiale entfalten zu können. Das gilt natürlich auch für die externen Schnittstellen zum Management. Dort sollte man ebenfalls flexibler und anpassungsfähiger werden, wenn es letztendlich dem Zweck dient, wertvolle und nützliche Produkte schneller in den Markt zu bringen oder Lösungen in den Live-Betrieb. Dazu gehören auch die einen oder anderen Fehlschläge, die in einer neuen Fehlerkultur positiv bewertet werden sollten, insbesondere wenn es schnelle und überschaubare Fehlschläge sind.

Falls agile Elemente wie User Stories nicht auf Anhieb funktionieren, kann das viele Ursachen haben, die man professionell analysieren und verstehen sollte. Denn jedes schnelle Verwerfen bzw. Verurteilen einer agilen Vorgehensweise ist eine vergebene Chance, den Unternehmenswert nachhaltig zu erhöhen.

Gratis Webinar zum Thema

Am 24.3.2017 um 16:00-17:00

Hybrides Projektmanagement für agile Projekte in mittelständischen und großen Unternehmen

Anmeldung 24.3.2017:

<https://attendee.gotowebinar.com/register/4121524112520959490>

Dieses Webinar vermittelt hybride Ansätze für das Projektmanagement in mittelständischen und großen Unternehmen. Hybrid heisst hier, dass auf Makroebene gewohnte Projekt- und Portfoliomanagementstandards im Unternehmen beibehalten werden können bzw. müssen, während auf Mikroebene im Core Team agil gearbeitet wird. Die Schnittstellen zwischen Makro- und Mikroebene gilt es abzustimmen, damit neue, agile Vorgehensweisen in traditionelle Vorgehensweisen integriert werden können.

Besprochen wird auch, wie agile Vorgehensweisen zugeschnitten werden können und sollen auf Ihr individuelles Umfeld sodass Sie durch die agilen Elemente Ihre Projekte erfolgreicher machen und vor allem für mehr Wertbeitrag für den Fachbereich sorgen.

Scrum als die beliebteste und am meisten verbreitete agile Methode wird als Referenz verwendet, d.h. anhand typischer Vorgehensweisen in Scrum werden Anregungen gegeben um hybrides Projektmanagement erfolgreich zu machen.

Autoren-Info

Der Autor Rainer Wendt ist Certified Business Analysis Professional (CBAP)[®], Project Management Professional (PMP)[®], Professional in Business Analysis (PMI-PBA)[®], sowie Agile Certified Practitioner (PMI-ACP)[®]. Herr Wendt ist Geschäftsführer der masVenta Business GmbH und Präsident des Germany Chapter des International Institute of Business Analysis .



Dipl.-Ing. Rainer Wendt, CBAP, PMP,
PMI-ACP, PMI-PBA
Präsident IIBA Germany Chapter e.V.

Managing Director masVenta Business
GmbH, Germany

Herr Wendt arbeitet seit über 25 Jahren in der IT Branche. Vor der Gründung der masVenta durchlief er verschiedene Positionen im mittleren Management unter anderem in der Software Entwicklung und im Corporate Development der Siemens Tochter Cycos AG und bei der deutschen Niederlassung eines Global Players für Mobilfunk CRM und Billing Lösungen Amdocs, Israel. Herr Wendt leitete viele große Software-Integrations-Projekte wie bei SAP, BMW, Deutscher Bundestag Berlin und bei Vodafone Deutschland. Seit 2009 wurde Herr Wendt kontinuierlich als Projektleiter von Uniper Global Commodities SE (vormals E.ON) in Düsseldorf bestellt.