IAPM

WEISSBUCH

HYBRID PROJECT MANAGEMENT

iapm
INTERNATIONAL ASSOCIATION OF PROJECT MANAGERS

IAPM
INTERNATIONAL ASSOCIATION OF PROJECT MANAGERS

Die IAPM begann 1997 als loser Zusammenschluss von internationalen Projektmanagern, die alle ein Ziel hatten: Projektmanagement fördern, modernisieren und jungen Projektmanagern Rüstzeug an die Hand geben, um erfolgreich arbeiten zu können. Seitdem fanden in jährlichem Rhythmus mehrere „International Project Manager Meetings“ (IPMM) statt. Schon 1998 brachte die IAPM den Vorläufer des heutigen „PM Guide 2.0“ heraus, die „IAPM By-laws of Project Management“. Diese bildeten den Grundstein für den 2010 komplett überarbeiteten, an die Bedürfnisse und den realen Alltag des modernen Projektmanagers angepassten „PM Guide 2.0“. Im Zuge dessen folgte 2010 der komplette Relaunch der IAPM. Im März 2011 erschien der Scrum Guide 1.0, Vorläufer des aktuellen, 2016 überarbeiteten Agile PM Guide 2.0. Im Jahr 2017 erschien dann der Hybrid PM Guide 2.0.

Im Jahr 2012 wurden erstmals zwei Awards verliehen, zum einen der Award „Project Manager of the Year“ und zum anderen der Award „Book of the Year“.

Der Award „Project Manager of the Year“ liegt der IAPM besonders am Herzen, denn damit zeichnet sie denjenigen IAPM Senior Project Manager aus, der herausragende Leistungen im Projektmanagement erbracht hat.

Mit dem Award „Book of the Year“ werden Werke ausgezeichnet, die sich mit dem Themenkomplex Projektmanagement beschäftigen und in deutscher und englischer Sprache herausgegeben werden. Das kann in Form von besonders innovativer Erfahrungs- und Wissensvermittlung sein, ein (auto)biographisches Werk oder ein Lehrbuch, das den Einstieg in das Thema Projektmanagement bietet.

Die IAPM ist Träger einer unabhängigen Zertifizierungsinstitution, die Fachwissen und Kompetenz der Zertifikatsanwärter durch ein umfassendes, faires und neutrales Prüfungssystem online bewertet. Das Zertifizierungssystem ist somit ganz speziell auf die herausfordernde Lebenswelt des Projektmanagers des 21. Jahrhunderts zugeschnitten.

INHALT

06 Einleitung Hybrides Projektmanagement

07 **Die drei wichtigsten Ansätze für das Projektmanagement**

07 **Klassisches Projektmanagement**

08 Mögliche Projektphasen eines Entwicklungsprojekts

10 Die Projektorganisation und die Rollen im Projekt

11 Vor- und Nachteile des klassischen PM

12 **Lean Management**

12 Die Prinzipien des Lean Management

13 Lean Methoden für das Projektmanagement

13 Kanban

14 MVP – Minimum Viable Product

15 Lean Startup

16 **Agiles Projektmanagement**

16 Das Fundament des Agilen PM

17 Die wichtigsten Vertreter des Agilen PM

17 Scrum

21 IT-Kanban

25 Extreme Programming (XP)

29 Design Thinking

31 FDD Feature Driven Development

35 **Die Ausgangsbasis für hybrides Projektmanagement**

35 Grundsätzliche Formen des Hybriden PM

36 Die richtige Wahl der Partnerelemente

36 Einfach, kompliziert oder komplex?

37 Anforderungen, die sich aus dem Projekt, der Branche und dem Team ergeben

40 Das Organisationsumfeld und die kulturellen Aspekte des Wandels

42 Die Motivation für Hybrides PM

45 **Der Weg zum Hybriden PM-Modell**

45 Strukturen des Hybriden PM im Projektkontext

48 Herausforderungen der Hybriden Kultur

49 **Schlussbemerkungen**

50 Impressum

HYBRIDES PROJEKTMANAGEMENT

hy·b·rid: etwas gebündeltes, gemischtes oder durch Kreuzung entstandenes

Mit dem „Hybrid PM Guide 2.0“ wird das Thema hybrides Projektmanagement beleuchtet, seine Ursprünge und Formen, seine Möglichkeiten und Grenzen aufgezeigt.

1. EINLEITUNG HYBRIDES PROJEKTMANAGEMENT

Üblicherweise wird unter „hybridem Projektmanagement“ die Verwendung von agilen Ansätzen in einem klassischen, phasengetriebenen Projektumfeld verstanden. Das beschreibt aber nur einen Teilbereich der hybriden Varianten. Ausgehend von den drei Managementkulturen „klassisches Projektmanagement“, „Lean Management“ und „agiles Projektmanagement“ ergeben sich unterschiedlichste Kombinationen, die jeweils ihren eigenen Charme und gleichzeitig ihre Begrenzungen haben. Die Kombination kann von den Unternehmen in einfacher Weise zur Gestaltung einer adäquaten, firmenspezifischen Projektmanagementkultur genutzt werden. Selbst Fertigungsbereiche, eine bisher reine Domäne des klassischen Projektmanagements, finden inzwischen in agilen Fertigungskonzepten eine funktionierende Alternative und Ergänzung. Die Motivation hybrides Projektmanagement einzusetzen oder den Einsatz in Erwägung zu ziehen, liegt neben dem Wunsch nach effizienteren und wirksameren Werkzeugen, häufig in einer traditionellen Unternehmensorganisation, die sich mit der Transformation in eine agilere Organisation nicht anfreunden kann oder einen kontrollierten, stufenweisen Prozess bevorzugt. Zunächst werden die unterschiedlichen Projektkulturen und ihre Charakteristiken gegenübergestellt, um anschließend die möglichen Kombinationen zu beleuchten.

2. DIE DREI WICHTIGSTEN ANSÄTZE FÜR DAS PROJEKTMANEGEMENT

2.1 KLASSISCHES PROJEKTMANAGEMENT (WASSERFALLMODELL)

Beim klassischen Projektmanagement wird der gesamte Projektablauf in sequentielle Phasen untergliedert. Den Abschluss jeder Phase bildet ein Meilenstein bei dem die zeit-, leistungs- und aufwandsgerechte Umsetzung der in der jeweiligen Phase festgelegten Arbeitsschritte in einem Review geprüft und wenn möglich abgenommen wird. Erst nach der erfolgten Abnahme wird die nächste Phase begonnen. Eine Rückkopplung ist nur zwischen aufeinanderfolgenden Phasen eingeschränkt möglich, bedeutet aber u. U. einen erhöhten Aufwand. Mit fortschreitender Projektlaufzeit steigen mögliche Änderungskosten als Konsequenz des sequentiellen Vorgehens häufig exponentiell an. Um diesem Problem zu begegnen, wird in der Entwurfsphase des Projekts ein hoher Aufwand für eine möglichst umfassende und genaue Planung betrieben.

Die Notwendigkeit, zu Beginn des Projekts die vereinbarten Anforderungen einzufrieren, ist häufig problematisch. Bei langen Projektlaufzeiten besteht die Gefahr, dass der erzeugte Projektgegenstand zwar dem Anforderungsprofil zu Projektbeginn entspricht aber tatsächlich bei Fertigstellung veraltet ist.

Umfangreiche Vertragswerke mit Änderungskonzepten schaffen hier zwar Abhilfe, können aber nicht verhindern, dass im Eintrittsfall Kosten- und Terminabweichungen auftreten.

2.1.1 MÖGLICHE PROJEKTPHASEN EINES ENTWICKLUNGSPROJEKTS

Anforderungsphase

- Ausarbeitung eines Business Case
- Durchführung einer Stakeholder- und Umfeldanalyse
- Erstellung einer Anforderungsanalyse und -priorisierung
- Ausarbeitung und Übergabe eines umfassenden Lastenheftes mit allen Anforderungen durch den Auftraggeber

Entwurfsphase (Planung)

- Festlegung Projektart
- Vereinbaren der internen Projektorganisation
- Festlegung der Kommunikationsstruktur
- Festlegung der Rollenverteilung
- Festlegung der Projektziele und Priorisierung
- Erstellung des Pflichtenheftes durch den Auftragnehmer
- Freigabe des Umsetzungskonzeptes
- Erarbeitung der projektspezifischen Phasen
- Festlegung Meilensteine inkl. der Abnahmekriterien für den Phasenwechsel
- Ausarbeitung eines Projektstrukturplans inkl. kritischem Pfad
- Risikoanalyse
- Maßnahmen zur Fehlervermeidung bzw. Fehlererkennung definieren
- Festlegung Änderungsmanagement
- Entwicklungswerkzeuge, Technologien und Arbeitsumgebungen definieren
- Dokumentationsstrategie festlegen

Entwicklungsphase

- Eventuell in mehreren Umsetzungsstufen (Prototyp, Endprodukt) inkl. einer jeweils nachgeschalteten Testphase
- Umsetzung aller im Pflichtenheft gelisteten Anforderungen
- Review der Projektziele und Einhaltung der Vorgaben
- Proaktives Stakeholdermanagement
- Anforderung sind eingefroren - Änderungen müssen begründet und dokumentiert werden. Sie bedürfen Zustimmung des Auftragsgebers
- Erstellung der erforderlichen Dokumentationen

Testphase und Finalisierung

- Testszenario entwerfen
- Testen des Endproduktes
- Fehlererkennung, -bewertung und -behandlung
- Produktpräsentation erstellen
- Benutzerhandbücher, Installationsanweisungen etc. erstellen
- Interner Projektreview

Auslieferung/Einsatz/Betrieb

- Übergabe des Produktes an Auftraggeber
- Produktpräsentation
- Gemeinsamer Review
- Wartungsstrategie festlegen

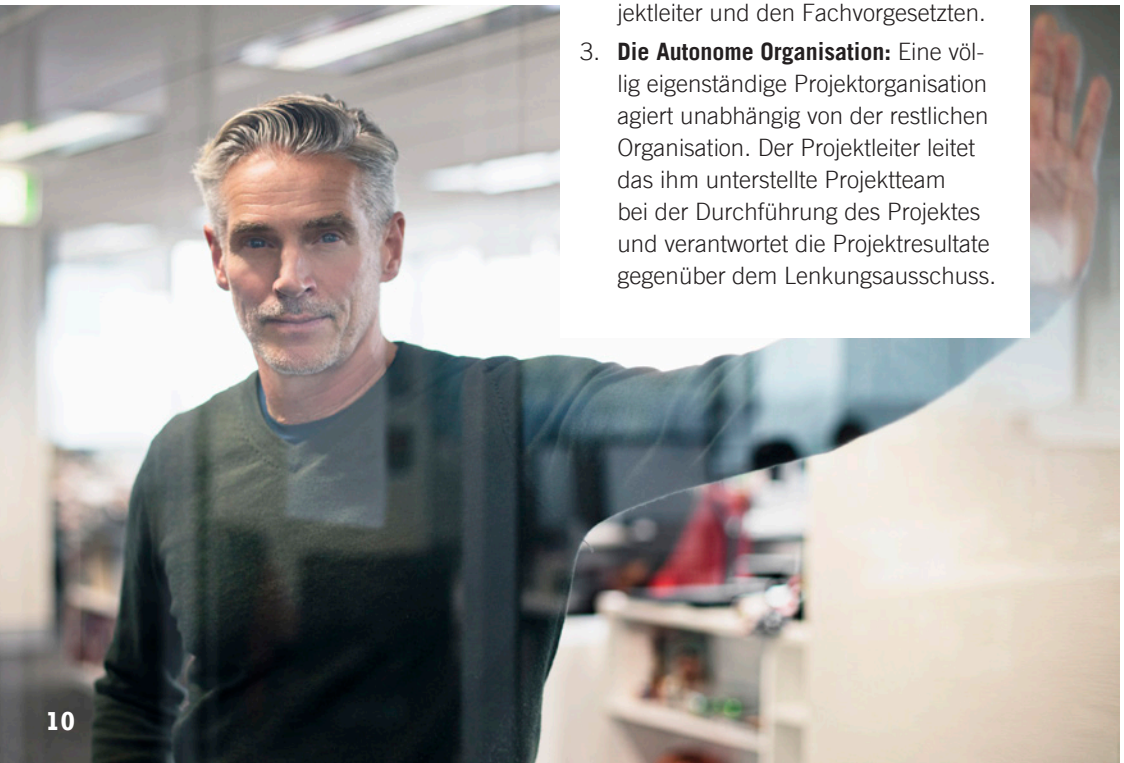
2.1.2 DIE PROJEKTORGANISATION UND DIE ROLLEN IM PROJEKT

Im klassischen Projektmanagement kann man vier Rollen identifizieren, die je nach Organisationsform unterschiedliche Befugnisse und Verantwortlichkeiten haben.

1. Projektleiter
2. Projektteam
3. Lenkungsausschuss
4. Stakeholder – alle, die vom Projekt oder dem Projektergebnis mehr oder weniger beeinflusst sind.

Das klassische Projektmanagement kennt drei Aufbauorganisationsformen:

1. **Die Stabsorganisation:** Der Projektleiter hat hier nur eine sehr eingeschränkte Handlungsfreiheit und keine Weisungsbefugnis. Diese bleibt bei der Linie. Seine Funktion ist dabei auf den informellen Austausch beschränkt und er kann somit auch keine Verantwortung übernehmen.
2. **Die Matrixorganisation:** Bei der Matrixorganisation überschneiden sich die Weisungsbefugnisse der Linie mit denen der Projektleitung. D.h. die Projektbeteiligten haben immer zwei gleichwertige Vorgesetzte: Den Projektleiter und den Fachvorgesetzten.
3. **Die Autonome Organisation:** Eine völlig eigenständige Projektorganisation agiert unabhängig von der restlichen Organisation. Der Projektleiter leitet das ihm unterstellte Projektteam bei der Durchführung des Projektes und verantwortet die Projektresultate gegenüber dem Lenkungsausschuss.



2.1.3 VOR- UND NACHTEILE DES KLASSISCHEN PROJEKTMANAGEMENTS IM ÜBERBLICK

Vorteile

- Hohe Planungssicherheit bzgl. Kosten, Terminen und Leistungen
- Klare Phasenabgrenzung
- Einfach abzuarbeitende Kontrollszenarien
- Klassisches Projektmanagement bietet bei stabilen und klaren Anforderungen und bekannten Umsetzungsmöglichkeiten einen effektiven Handlungsrahmen
- Festgelegte, klare Verantwortlichkeiten
- Einfache Skalierung

Nachteile

- Strikte Phasengrenzen sind oft unrealistisch
- Planung ist schwierig bei unbekannter technischer Umsetzung
- Inflexibilität bei Änderungen
- Zu Beginn des Projektes gibt es häufig hohe Kostenunsicherheit und am Ende des Projektes ausufernde Kosten bei Änderungen
- Auftraggeber ist gezwungen umfangreiche Anforderungen zu einem frühen Zeitpunkt abzugeben ohne sich an Ergebnissen oder Vorstellungen orientieren zu können.
- Die Projektleitung und der Lenkungsausschuss, als Verantwortliche, sollten sich fachlich mit ihrem Team auf Augenhöhe befinden oder sie treffen wichtige Entscheidungen ohne einen entsprechenden Hintergrund.

2.2 LEAN MANAGEMENT

2.2.1 DIE PRINZIPIEN DES LEAN MANAGEMENT

Das oberste Ziel des „Lean Management“ war es ursprünglich, den Herstellungsprozess physikalischer Produkte auf allen Ebenen zu optimieren. Seither hat sich der der „leane“ Gedanke weiterentwickelt und kann heute in jeder Branche und für jedes Produkt Anwendung finden. Das Lean Management ist dabei, anders als das agile und das klassische Projektmanagement, kein speziell für die Projektarbeit geschaffenes Managementsystem, sondern ein Bündel von Maßnahmen und Herangehensweisen, die im Projektmanagement Anwendung finden können.

Die wichtigsten sind:

- Just-in-time, das Richtige, zur richtigen Zeit in der richtigen Qualität und Menge am richtigen Ort
- Kanban (Schildchen) zur Sicherstellung von Just-in-time und Einführung des Pull-Prinzips
- Vermeidung von Mura (Unausgeglichenheit), Muri (Überbeanspruchung) und Muda (Verschwendung)
- Kaizen als evolutionärer, fortlaufender Verbesserungsprozess in kleinen Schritten
- Kaikaku, ein revolutionärer, drastischer Verbesserungsprozess und Kulturwandel im Unternehmen
- Jidoka, die autonome Automation zur Fehlererkennung
- Heijunka, die Reduktion der Losgrößen zur Flexibilisierung und Nivellierung der Fertigung
- Die 5W-Fragen – das mehrfache (nicht notwendigerweise fünf Mal) Stellen der Frage warum, um eine Ursache hinreichend zu erforschen
- Andon, die Schaffung von einfachen, visualisierten Statuskontrollen (Ampelfarben)
- Genchi genbutsu – hingehen und nachschauen als effektive Hilfe bei Problemstellungen

Das Kernziel lässt sich dabei in der folgenden Aussage zusammenfassen:

„Die Erzeugung eines möglichst großen Kundennutzens durch einen optimalen Wertschöpfungsprozess, bei dem kein unnützer Aufwand betrieben wird.“

Aus diesen Maßnahmen lassen sich **Lean Prinzipien** generieren, die schon früh den Weg Richtung Agile weisen und im Projektumfeld anwendbar sind:

- Der Fokus auf die Menschen (Kunde, Mitarbeiter)
- Vermeidung von Verschwendung
- Die Weitergabe von Wissen
- Die Reflektion, des eigenen Verhaltens
- Transparenz in der Arbeit
- Beständiges Lernen
- Gemeinsame Ziele vereinbaren
- Streben nach bestmöglicher Qualität
- Die Arbeit begrenzen
- Arbeit im Fluss halten

Trotz vieler Gemeinsamkeiten ist es wichtig lean und agil nicht zu vermischen, da es zwar wenige, aber doch deutliche Unterschiede in der grundsätzlichen Ausrichtung gibt.

1. Lean sucht die Standardisierung – Agiles die Individualisierung/Flexibilisierung
2. Lean konzentriert sich auf den Produktionsprozess, Agile auf das Produkt
3. Lean möchte die Fehler und damit Verschwendung vermeiden, Agile begreift den Fehler als Teil eines Lernprozesses

2.2.2 LEAN METHODEN FÜR DAS PROJEKTMANAGEMENT

2.2.2.1 KANBAN

Der Begriff Kanban entstammt der japanischen Sprache und steht für „Karte“ oder „Beleg“ und steht für ein nach dem „Pull-Prinzip“ funktionierendes Steuerungsverfahren zur Produktivitätssteigerung. Kanban findet hier Erwähnung, weil es die Basis des heute im Projektmanagement verwendeten IT-Kanban ist. Die Unterschiede sind allerdings beträchtlich so, dass IT-Kanban, als eine nahezu reine agile Methode, im folgenden agilen Abschnitt beschrieben wird.

2.2.2.2 MVP – MINIMUM VIABLE PRODUCT

MVP steht für ein verkaufsfähiges Produkt mit minimalen Anforderungen und Eigenschaften. Ziel ist es möglichst schnell ein Produkt mit nur den notwendigsten Features beim Kunden zu platzieren.
Dessen Feedback wird dann genutzt um das Produkt zu erweitern bzw. um überhaupt das Kundeninteresse zu evaluieren. Alle weiterführenden Funktionen und Features sollen sich dabei auf den Zweck des Produktes konzentrieren. Unnötiges und Zweckfremdes wird vermieden. Gerade im Start-up Bereich arbeitet man häufig mit einer überschaubaren Kapitaldecke bei gleichzeitig hohem Risiko für den „Return on Investment“ (RoI). Mit Hilfe des MVP ist es möglich sehr zügig Marktlücken zu entdecken und besetzen.

Durch die Beschränkung auf das Notwendigste wird die Projektlaufzeit beim MVP minimalisiert. Lange Produktentwicklungszeiten bergen immer das Risiko, an einem sich schnell wandelnden Markt, vorbei zu entwickeln und dem Kundenbedürfnis nicht mehr gerecht zu werden.

Vorteile

- Kurze Laufzeit
- Geringes Risiko
- Schnelle Prüfung von Marktchancen

Nachteile

- Beschränkt auf weniger komplexe Produkte und Dienstleistungen
- Kunde ist „Tester“ ob das „Minimum“ schon erreicht ist
- Produkt reift beim Kunden

2.2.2.3 LEAN STARTUP

Lean Startup basiert auf den Lean Management Methoden und beschreibt ein Vorgehen zur Unternehmensgründung mit möglichst geringem Risiko. Sichergestellt wird das durch die Minimalisierung des Aufwands, um ein Produkt oder eine Dienstleistung in den Markt zu bringen. Dabei geht es nicht nur um die Arbeitsergebnisse, auf den Prüfstand kommen auch das Geschäftsmodell, die Vertriebskanäle, die Preisgestaltung und die Kundengewinnung.

Vorteile

- Geringes Risiko
- Schnelle Prüfung von Marktchancen
- Unternehmenswachstum an RoI anpassbar

Nachteile

- Kunde ist „Tester“ der Produkte und Beta-Versionen
- Misserfolge und der folgende Lernprozess sind Bestandteile des Startup-Konzeptes

Entscheidend sind folgende Elemente:

- Enge kommunikative Einbindung des Auftraggebers (falls vorhanden)
- Geringst möglicher Kapital- bzw. Ressourceneinsatz
- Kurzer Entwicklungszyklus ohne lange Vorplanung
- Wenn möglich mit Beta-Versionen oder Prototypen im Markt schon sehr frühzeitig das Kundenfeedback abholen und in ein Produkt-Update umsetzen.
- Einbinden des Endanwenders und kontinuierliches Einholen des Kunden-Feedbacks
- Learning by doing und iterative Entwicklung der Produkte und Dienstleistungen
- Oft in Verbindung mit dem MVP

2.3 AGILES PROJEKTMANAGEMENT

2.3.1 DAS FUNDAMENT DES AGILEN PROJEKTMANAGEMENTS

Agiles Projektmanagement wird häufig fälschlicherweise mit Scrum gleichgesetzt. Tatsächlich ist agiles Projektmanagement mittlerweile ein bunter Strauß an unterschiedlichen Vorgehensweisen und Methoden, Werkzeugen und Regelwerken, von denen Scrum eines der bekanntesten ist. Der symbolische Startschuss des agilen Projektmanagements, als Kontrapunkt zum Wasserfallmodell, fiel 2001 mit dem Erstellen des „Agilen Manifest“ – eigentlich „Agile Software Development Manifesto“.

Mit dem agilen Manifest wurden Prinzipien formuliert, um die zunehmend komplexere Softwareentwicklung im Umfeld der New Economy in den Griff zu bekommen, nachdem phasengetriebene Projekte häufig scheiterten.

Das Agile Manifest besagt:

- Individuen und Interaktionen sind wichtiger als Prozesse und Werkzeuge
- Funktionierende Software ist wichtiger als eine umfassende Dokumentation
- Zusammenarbeit mit dem Kunden ist wichtiger als Vertragsverhandlungen
- Reagieren auf Veränderung ist wichtiger als das Befolgen eines Plans

Das „Agile Manifest“ darf nicht missverstanden werden, denn die Aussagen der rechten Seite (rot) sollten niemals durch die Werte der linken Seite (grün) gänzlich ersetzt werden – es wird nur bewusst mehr Augenmerk auf die Aussagen der linken Seite gelegt.

Seit der Veröffentlichung des „Agilen Manifests“ hat sich die Familie der agilen Vorgehensweisen zwar in vielen Varianten weiterentwickelt und vergrößert aber dennoch tragen sie alle seine Grundideen in sich. Einige der bekanntesten Vertreter werden im folgenden Kapitel erläutert.

2.3.2 DIE WICHTIGSTEN VERTRETER DES AGILEN PROJEKTMANAGEMENTS

2.3.2.1 SCRUM

Scrum ist keine Methode, sondern ein Framework und zeigt dabei, ähnlich dem Regelwerk einer Sportart, die Handlungsgrenzen auf und legt Regeln fest. Es trifft keine Aussagen oder macht keine Vorgaben wie etwas zu entwickeln ist. Es beinhaltet genauso wenig eine Strategie an der man sich orientieren kann. Sein Ziel ist es in kurzen, iterativen Schritten ein auslieferbares, also fertiges, Produkt mit Hilfe eines selbstorganisierenden Teams zu erzeugen oder zu betreuen. Die iterativen Entwicklungszyklen werden Sprints genannt. Die Basis von Scrum sind die Grundwerte Commitment (Selbstverpflichtung), Courage (Mut), Focus (Konzentration), Openness (Offenheit) und Respect (Respekt). Sie bilden das Fundament der drei Säulen, die Scrum tragen:

Transparency

Die wichtigen Aspekte des Arbeitsprozesses muss für alle Beteiligten durchschaubar und klar sein, um ein gemeinsames Verständnis für das Vorgehen und die Ergebnisse teilen zu können.

Inspection

Regelmäßig werden die Arbeitsprozesse, Artefakte (Von Menschen erzeugte oder veränderte Dinge) und die Ergebnisse im Hinblick auf das Erreichen des gemeinsamen Zieles von den Beteiligten inspiziert.

Adaption

Erkannte Abweichungen oder Impediments (Hindernisse) in den Prozessen oder Ergebnissen werden umgehend behoben bzw. beseitigt, um den negativen Einfluss auf Kosten, Leistung oder Zeit zu minimieren.

Scrum besteht aus drei Gruppen von Elementen, die fest mit Regeln verknüpft sind.

A. Die Rollen des Scrum Teams

1. Product Owner

- Verantwortet den RoI und ist vollumfänglicher Vertreter des Auftraggebers
- Legt fest „was“ und „warum“ etwas entwickelt wird
- Nimmt die Arbeitsergebnisse im Sprint Review ab

2. Development Team

- Interdisziplinäres, selbstentscheidendes Team, mit maximal 9 Mitgliedern
- Entscheidet „wie“ die Arbeitsergebnisse erzeugt werden
- Überprüft seine Arbeitsprozesse in der Sprint Retrospective

3. Scrum Master

- Sorgt für die Einhaltung der Scrum Regeln
- Beseitigt Impediments im Arbeitsprozess
- Moderator und Unterstützer des Scrum Teams

B. Die Artefakte

Bei Artefakten handelt es sich um Arbeitsunterlagen und Dokumente, Charts und Arbeitsergebnisse.

1. Product Vision

Die Product Vision dient für alle Projektbeteiligten als Leuchtfeuer für die Ziele und den Zweck des Projekts.

2. Product Backlog

Das Product Backlog ist eine priorisierte, üblicherweise unvollständige Liste, der im Produkt umzusetzenden Themen und Wünsche – im Scrum Sprachgebrauch User Stories genannt.
Neupriorisierungen, das Hinzufügen, Entfernen oder Ersetzen von User Stories im Product Backlog sind üblich und erlauben auf diesem Weg die Anpassung der Entwicklung an sich ändernde Umfeldbedingungen.

3. Sprint Backlog

Das Sprint Backlog enthält die Tasks (Aufgaben), die notwendig sind, um die aus dem Product Backlog gewählten hochprioren User Stories im nächsten Sprint in ein Product Increment umzusetzen.

4. Product Increment

Am Ende jedes Sprints, steht ein auslieferbares, getestetes Product Increment, das vom Product Owner abgenommen werden muss.

5. Sprint Burndown Chart

Über das Sprint Burndown Chart wird der Arbeitsfortschritt im Sprint visualisiert.

6. Release Burndown Chart

Das Release Burn Down Chart dokumentiert den Arbeitsfortschritt von Sprint zu Sprint und erlaubt damit eine Release Planung für den Product Owner.

7. Impediment Backlog

Eine vom Scrum Master zu erstellende Liste der Impediments für das Scrum Team, die er abarbeitet.

C. Die Scrum Events (4 Meetings & Sprint)

Alle Events im Scrum sind time-boxed, d. h. es existiert eine feste Zeitvorgabe, in der das Event abgeschlossen sein muss.

1. Der Sprint

Der Sprint ist ein maximal 4 Wochen langer Entwicklungszeitraum, in dem das Product Increment erstellt wird. Als Arbeitsgrundlage dient dabei das Sprint Backlog.

2. Sprint Planning

Meeting bei dem es darum geht, ein Verständnis zwischen Product Owner und Development Team bzgl. der umzusetzenden User Stories im Product Backlog herzustellen. In einem zweiten Schritt wird dann eine vereinbarte Menge an User Stories vom Development Team in Tasks für den nächsten Sprint übersetzt und im Sprint Backlog platziert.

3. Daily Scrum

Ein 15-minütiges, tägliches Stand-Up Meeting bei dem sich das Development Team im Sprint gegenseitig einen kurzen, formalisierten Statusbericht gibt.

4. Sprint Review

Im Sprint Review stellt das Development Team dem Product Owner das Product Increment vor.
Der Product Owner prüft anhand der von im Sprint Planning vereinbarten Kriterien das erzeugte Product Increment und nimmt es bei Erfüllung der Anforderungen ab.

5. Sprint Retrospective

Im Anschluss an das Sprint Review trifft sich das Development Team mit dem Scrum Master, um ihre Zusammenarbeit zu überprüfen und gegebenenfalls Vorschläge zur Verbesserung aufzunehmen und später umzusetzen.

D. Die Scrum Elemente im Ablauf

Ausgehend von einer Product Vision erstellt der Product Owner ein erstes Product Backlog. Dieses wird mit dem Development Team im ersten Teil des Sprint Planning diskutiert und wenn notwendig angepasst. Im zweiten Teil des Sprint Planning werden dann die hochprioren Themen in das Sprint Backlog übernommen, in Tasks (Aufgaben) überführt und im folgenden Sprint abgearbeitet. Täglich tauschen sich die Development Teammitglieder während des Sprints im Daily Scrum aus. Am Ende des Sprints werden im Sprint Review und der Sprint Retrospective Arbeitsergebnis und Arbeitsweise überprüft und wenn notwendig angepasst. Dann beginnt der Kreislauf mit dem nächsten Sprint Planning von vorne. Dieser Prozess wird solange wiederholt, bis der Product Owner die Aufgabe für erfüllt erklärt oder das Product Backlog vollständig umgesetzt wurde.

E. Vor- und Nachteile von Scrum

Vorteile

- Wenige und leicht verständliche Regeln
- Überschaubare Kommunikationskanäle
- Hohe Flexibilität und leichte Anpassung an sich ändernde Anforderungen
- Effektivität durch Selbstorganisation der Development Teams
- Große Transparenz der Prozesse und Ergebnisse
- Schnelle Umsetzung von Produktideen erlaubt dem Auftraggeber frühes Feedback
- Kontinuierlicher Lern- und Verbesserungsprozess

Nachteile

- Mitarbeiter müssen mehr Verantwortung übernehmen – das findet nicht bei allen Zustimmung
- Scrum gibt keine Handlungsempfehlungen
- Scrum ist leicht zu verstehen aber schwer umzusetzen
- Bei der Skalierung von Scrum werden häufig bestimmte Regeln missachtet und die Vorteile von Scrum damit potentiell aufgegeben. Beispielhaft ist dabei das Sammeln von Product Increments über mehrere Sprints hinweg um eine mit anderen Development Teams synchronisierte, gebündelte Auslieferung zu realisieren.
- Scrum ist potentiell mit traditionellen Unternehmensstrukturen schwer vereinbar und Bedarf der Einführung neuer organisatorischer Praktiken (z. B. Mitarbeiterbewertung bei unabhängigen Teams)

2.3.2.2 IT-KANBAN

IT-Kanban ist die Verknüpfung des Konzepts des ursprünglichen Kanban mit Prinzipien aus den Bereichen der Lean Production, des Lean Development und der „Theory of Constraints“ (Engpasstheorie), die besagt, dass der Durchsatz eines Systems ausschließlich von einem begrenzenden Faktor bestimmt wird. Das Ergebnis ist ein Vorgehensmodell für die Softwareentwicklung in der IT, bei dem die Wertschöpfungskette auf einem Kanban Board visualisiert wird. Durch die Begrenzung von parallelen Arbeiten (WiP - Work in Progress) wird ein kontinuierlich, optimaler Arbeitsfluss („Flow“) erreicht.

Die Weitergabe der Arbeiten von Station zu Station findet im Pull-Mode statt, d. h. wenn eine Station Arbeitskapazität zur Verfügung hat, holt sie sich neue Aufgaben, solange der festgelegte WiP-Level noch nicht erreicht ist. Obwohl ursprünglich für die IT-Entwicklung konzipiert, lässt sich IT-Kanban in sehr einfacher Weise an beliebige Branchen und Umgebungen anpassen. Das ist möglicherweise einer der Gründe, warum in der Literatur IT-Kanban meistens nur noch als Kanban bezeichnet wird und damit die Differenzierung zum originären Kanban verwischt wird.

A. DIE PRINZIPIEN DES IT-KANBAN

Visualisierung der Wertschöpfungskette

Das Kanban-Board ist das zentrale Element der Visualisierung des Arbeitsflusses und der Identifikation von Engpassstellen. Ganz links ist die Spalte „To Do“, die alle anstehenden Aufgaben enthält. Eine Aufgabe durchläuft danach verschiedene Stationen entsprechend der notwendigen Prozessschritte, wie z. B. Analyse, Design, Coding, Testing etc., von links nach rechts. Jede Station ist dabei noch einmal in zwei Spalten unterteilt: In Progress (In Bearbeitung) und Done (Fertiggestellt). Jede Station kann sich bei ihrer Vorgängerstation neue Aufgaben aus der „Done“-Spalte ziehen (Pull-Prinzip).

Begrenzung des Work in Progress (WiP)

Die Aufgaben, die pro Station gleichzeitig bearbeitet werden dürfen, werden limitiert. Damit wird möglichen Flaschenhälsen begegnet und die Arbeit im Fluss gehalten. Durch diese Maßnahme in Zusammenhang mit einem Pull-Prinzip hält die Effizienz auf einem hohen Level.

Steuerung des Workflows und Analyse der Metriken

Die Priorisierung der Aufgaben erlaubt die Möglichkeit Aufgaben mit unterschiedlichen Dringlichkeitsstufen über das Kanban Board zu schicken. In Ver-

bindung mit der Analyse der Metriken kann das System optimiert und die Planung verlässlicher gestaltet werden. Typische Metriken sind:

- Lead Time (Zeit, die eine Aufgabe braucht, um die komplette Wertschöpfungskette zu durchlaufen)
- Cycle Time (Zeit, die eine Aufgabe braucht, um von einem Punkt A zu einem Punkt B zu gelangen)
- Throughput (Der mittlere Durchsatz von Aufgaben in einer gegebenen Zeit)
- Cumulative Flow Diagrams
- Queues (Warteschlangen)
- Blockers (Blockaden – evtl. Aufgaben, die externer Unterstützung bedürfen)
- WiP (Work in Progress)

Formulierung expliziter Prozessregeln

Ziel ist die Schaffung einer objektiven Basis um sicherzustellen, dass alle Prozessbeteiligten mit einem gemeinsamen Verständnis arbeiten und gleichen Regeln folgen. Zu diesem Prozessregeln gehören z. B. folgende Vereinbarungen:

- Wann ist eine Aufgabe „done“, das heisst erledigt?
- Wie werden Aufgaben gezogen?
- Nach welchen Regeln werden neue Aufgaben gezogen (Priorität)
- Wer wählt die Aufgabe?
- Wie werden Rückläufer aufgrund von fehlerhafter Umsetzung gehandhabt?

Förderung von Führungsqualitäten und kontinuierliche Verbesserung

Die kontinuierliche Verbesserung bedarf des Inputs der unmittelbar betroffenen Mitarbeiter „vor Ort“.
Um diese Kommunikation zwischen den Organisationsebenen zu ermöglichen ist es notwendig echte Führungsqualitäten zu fördern und die Kommunikation auf Augenhöhe zu begrüßen. Die Fähigkeit zu Lernen und das Gelernte umzusetzen ist ein essentieller Bestandteil des Verbesserungsprozesses (sog. Kaizen).

B. DIE EINFÜHRUNG VON IT-KANBAN MIT FLIGHT LEVELS

Flight Level 1

IT-Kanban wird in einer kleinen Organisationseinheit, z. B. bei einem cross-funktionalen Team eingesetzt, um den Prozess im Team zu verbessern. Auf die Schnittstellen nach außen hat das keinen Einfluss. Die Stakeholder, die dem Team ihre Aufgaben zuführen, stimmen sich nicht ab und priorisieren ihre Aufgaben nicht. Deshalb verbessert sich das Gesamtsystem nur begrenzt.

Flight Level 2

Im zweiten Flight Level stimmen sich die Stakeholder und das Team in sogenannten Queue Replenishment Meetings ab. Die anstehenden Aufgaben werden priorisiert und es werden ihnen Service-Klassen mit strikten Regeln zugeordnet, die darüber entscheiden wie schnell eine Aufgabe erledigt werden soll und welche Aufmerksamkeit ihr geschenkt wird. Auf diesem Weg findet eine Effektivitätssteigerung statt, weil die zur Zielerreichung notwendigen Ergebnisse bevorzugt werden.

Flight Level 3

Im nächsten Flight Level wird das IT-Kanban System auf die anderen Teams ausgedehnt und damit rückt die Zusammenarbeit der Teams in den Fokus. Durch die schrittweise Optimierung der ganzen Wertschöpfungskette werden Engpässe sichtbar, nach deren Auflösung die Wartezeiten an den Schnittstellen abnehmen und der Arbeitsfluss sich zunehmend verbessert.

Flight Level 4

Im letzten Flight Level 4 wird das Unternehmensportfolio durch den Einsatz von IT-Kanban gemanagt. Die Wertströme von mehreren Produkten und Projekten werden optimiert und das „Machbare“ mit den Anforderungen ausbalanciert.

C. VOR- UND NACHTEILE VON IT-KANBAN

Vorteile

- Leicht in beliebigen Umgebungen und Prozessen, auch bei kleinsten Teams einsetzbar
- Direkte Visualisierung des Prozesses
- Kanban Board ist eine gute Kommunikationsgrundlage für die Teamsynchronisation
- WiP Limit verhindert schädliches Multitasking
- Die Aufgabengröße ist anders als bei Scrum nicht limitiert (Sprintlänge)

Nachteile

- Ein physikalisches Board ist nur bei kleinen, lokalen Teams handhabbar
- Bei großen Projekten wird das physikalische Kanbanboard unübersichtlich
- Dokumentation und Archivierung machen den Einsatz von Softwarelösungen notwendig
- Kanban entwickelt sein ganzes Potential erst, wenn die komplette Lieferkette von der Anforderung bis zur Auslieferung konsequent miteinbezogen wird (Continous delivery)

2.3.2.3 EXTREME PROGRAMMING (XP)

XP ist eine iterative Vorgehensweise zur Entwicklung von Software. Es setzt sich in der ursprünglichen Fassung aus den aufeinander aufbauenden Elementen Values (Werte), Principles (Prinzipien) und Practices (Techniken) zusammen.

XP eignet sich für kleine bis mittlere Teams mit bis zu 15 Personen und ist ein strenger und disziplinierter Entwicklungsprozess, der einen starken Fokus auf die Kundeneinbindung legt.

A. DIE ROLLEN IM XP

Die Rollen im XP sind nicht in Stein gemeißelt und dienen nur der anfänglichen Orientierung. Ziel ist, dass jeder im XP Team im Rahmen seiner Möglichkeiten, auch rollenübergreifend, zum Gesamterfolg beiträgt.

1. Coach

Der Coach promotet die XP Grundlagen, hilft dem Team bei der Auswahl der effektivsten Werkzeuge und sorgt für die Einhaltung der Werte, Prinzipien und Praktiken.

2. Tester

Die Tester stellen die automatisierten Tests gemäß der Kundenanforderungen bereit und verantworten die Qualität des fertigen Produktes

3. Interaction Designer

Interaction Designer bilden die Brücke zwischen Entwickler und User (Anwender). Sie unterstützen die Kunden bei der Erstellung der User Stories und begleiten ihre Verwirklichung.

4. Architect

Architects sind verantwortlich für die innere Struktur der Systeme und optimieren diese bei Bedarf.

5. Project Manager

Der Project Manager organisiert die Kommunikation zwischen dem XP-Team und den Kunden. Er ist zuständig für die Fortschrittsverfolgung und die Planung der nächsten Aufgaben.

6. Executives (Geschäftsführung)

Die Geschäftsführung trägt und mitverantwortet das Projekt

7. Technical Writer

Der Technical Writer beschreibt die Systemfunktionen und sammelt im Dialog mit den Usern die Erfahrungen um sie in das Endprodukt zu integrieren

8. User (Anwender)

Die User erstellen die Systemanforderungen in Form von User Stories. Idealerweise bildet der User ein breites Spektrum an zukünftigen Anwendern ab.

9. Programmer

Die Programmer entwickeln das System und überführen die User Stories in funktionierende Software und Programmfunktionen.

B. WERTE, PRINZIPIEN UND TECHNIKEN IM XP

Values (Werte)

Die Werte (Values) Communication (Kommunikation), Courage (Mut), Feedback, Respect (Respekt) und Simplicity (Einfachheit) stehen bei XP im Vordergrund und bilden die Basis, auf der die Prinzipien aufbauen.

Principles (Prinzipien)

- Es gibt 14 Prinzipien, die eine Brücke bilden zwischen den eher abstrakten Values (Werten) und den direkt zur Anwendung kommenden Techniken „Accepted Responsibility“ (Verantwortung übernehmen)
- „Assume Simplicity“ (Einfachheit anstreben - sowohl beim Entwicklungsprozess als auch beim Produkt)
- „Concrete Experiments“ (gezielte Experimente zeigen frühzeitig Fehlentwicklungen auf und bestätigen andererseits funktionierende Ansätze)
- „Embracing Change“ (Veränderung wollen - wer sich gegen Veränderungen stemmt kann nicht agil werden und sich an beständig ändernde Umfeldbedingungen anpassen)
- „Honest Measurement“ (ehrliches Messen - kein Schönreden von Arbeitsergebnissen)
- „Incremental Change“ (inkrementelle Veränderung)
- „Local Adaptions“ (an örtliche Gegebenheiten anpassen, d.h. sowohl die Arbeitsmethoden als auch das erzeugte Produkt)
- „Open, Honest Communication“ (offene, aufrichtige Kommunikation)
- „Play To Win“ (auf Sieg spielen)
- „Quality Work“ (Qualitätsarbeit – nicht „quick and dirty“)
- „Rapid Feedback“ (unmittelbares Feedback hält Kosten niedrig und vermeidet Terminverzögerungen)
- „Small Initial Investment“ (geringe Anfangsinvestition)
- „Teach Learning“ (Lernen lehren – fortlaufendes Lernen ist ein essentieller Bestandteil agiler Vorgehensweisen)
- „Travel Light“ (mit leichtem Gepäck reisen, d.h. unnötige und umständliche Arbeitsweisen und Werkzeuge vermeiden)
- „Work With People’s Instincts, Not Against Them“ (Instinkte des Teams nutzen, nicht dagegen arbeiten)

Practices (Techniken)

- „On-site Customer“ (Kunde ist vor Ort)
- „Planning Game“ (Planungsspiel zu Beginn der Entwicklungs-Iteration, um den Arbeitsumfang und die technische Umsetzung abzustimmen)
- „Short Releases“ (kurze Releasezyklen zum Einholen des Kundenfeedbacks)
- „Coding Standards“ (gemeinsam vereinbarte Programmierstandards)
- „Continuous Integration“ (fortlaufende Integration)
- „Collective Ownership“ (alle tragen eine gemeinsame Verantwortung für das Erstellte)
- „Metaphor“ (Metapher - User Stories werden allgemeinverständlich beschrieben)
- „Sustainable Pace“ (nachhaltiges Tempo - die Vermeidung von Überlastung der Teammitglieder durch zu große Arbeitslasten)
- „Pair Programming“ (Programmieren zu zweit mit regelmässigem Rollentausch)
- „Refactoring“ (das Arbeitsergebnis wird kontinuierlich in kleinen, funktionserhaltenden Schritten optimiert)
- „Simple Design“ (einfaches Design)
- „Testing“ (Testen)

C. ITERATIONSSCHLEIFEN BEIM XP

Ein XP Projekt beginnt mit der Formierung des Projektteams, das entsprechend dem Projektthema multifunktional aufgestellt sein muss. Der User entwickelt zusammen mit dem Team seine Anforderungen in Form von User Stories. Der Architekt generiert einen ersten Systementwurf. Nach der Planung der Releases starten im nächste Schritt die Iterationsschleifen zur Entwicklung des Systems. Die Releases werden eingehend geprüft, eventuelle Fehler behoben und an den User übergeben.

1. **Sekundentakt:** Durch das Pair Programming wird der generierte Code während der Entstehung von einem zweiten Programmierer inspiziert und Fehler oder Designschwächen werden unmittelbar behoben.
2. **Minutentakt:** Im XP wird testgetrieben entwickelt. Die vor dem Coding generierten UNIT-Tests erlauben eine permanente, hohe Testabdeckung.
3. **Stundentakt:** Durch die Technik des Continuous Integration werden die entwickelten Komponenten mehrmals täglich in das lauffähige Gesamtsystem integriert. Durch dieses Vorgehen werden Fehler frühzeitig erkannt und teure Spätfolgen vermieden.
4. **Tagestakt:** Ähnlich dem Daily Scrum beim Scrum Framework findet auch beim XP ein tägliches Stand-Up Meeting statt, bei dem das Entwicklungsteam den Projektfortschritt erörtert und Probleme reflektiert. Durch die Meetings im Stehen und der damit verbundenen Unbequemlichkeit versucht man eine stärkere Fokussierung und damit möglichst kurze Besprechungsdauer zu forcieren.
5. **Wochentakt:** Dem Kunden werden in kurzen Abständen lauffähige Systemversionen zur Verfügung gestellt, um seine Anforderungen an ihnen zu spiegeln. Gegebenenfalls kann so eine Fehlentwicklung im Anfangsstadium vermieden und gegengelenkt werden. Weiterhin können Ergänzungen zu den Anforderungen für eine spätere Realisierung aufgenommen werden.
6. **Monatstakt:** Releases werden im Gegensatz zu phasengetriebenen Projektmodellen nicht einmalig am Projektende, sondern möglichst monatlich ausgeliefert. Damit wird es dem Auftraggeber ermöglicht schon frühzeitig einen Teil des Geschäftswertes zur wirtschaftlichen Verfügung zu haben.

D. VOR- UND NACHTEILE VON XP

Vorteile

- Die Techniken sind leicht kombinierbar mit anderen agilen Ansätzen
- Große Flexibilität bei der Umsetzung
- Hohe Qualität der Ergebnisse

Nachteile

- Als Gesamtpaket eher für kleine bis mittlere IT-Teams geeignet
- Fordert eine starke Kundeneinbindung, die nicht immer einfach darstellbar ist
- Weniger geeignet für verteilte Teams
- Sehr IT-lastig und deshalb schwerer auf andere Branchen übertragbar

2.3.2.4 DESIGN THINKING

Design Thinking geht in seinem Ansatz weit über das klassische Gestalten hinaus und rückt, im Gegensatz zu den technischen Disziplinen, bei denen die Machbarkeit im Vordergrund steht, den Anwender und seine Wünsche in den Mittelpunkt. Die multidisziplinären Entwicklerteams begeben sich in die Rolle des Anwenders und versuchen mit „seinen Augen“ das Problem zu analysieren. Design Thinking ist dabei keiner Branche zugeordnet, sondern eignet sich als systematische Lösungsstrategie für viele komplexe Problemstellungen, Entwicklung von Serviceleistungen und innovativer Produkte aus unterschiedlichsten Bereichen. Gefundene Lösungen und Ideen sollen in der Form von Prototypen möglichst früh erstellt werden, um ein entsprechendes Feedback vom Anwender zu ermöglichen. Die erzielten Ergebnisse und innovativen Problemlösungen vereinen am Ende die drei essentiellen Komponenten technologische Machbarkeit, Wirtschaftlichkeit und die Erfüllung eines Bedürfnisses oder Wunsches.

DIE DREI ELEMENTE DES ARBEITSUMFELDES

Der Erfolg von Design Thinking beruht im wesentlichen auf einem gemeinschaftlichen Arbeits- und Denkprozess, der durch die folgenden drei Elemente gefördert wird:

Multidisziplinäre Teams

Mehrere multidisziplinäre, heterogene Teams von jeweils fünf bis sechs Personen arbeiten mit der Unterstützung eines methodisch ausgebildeten Coaches an einem Thema. Ziel ist es immer, anfassbare, begreifbare Ergebnisse zu erzielen und diese mit den anderen Teams auszutauschen, um möglichst viele Perspektiven auf die Problemlösung zu erhalten.

Variable Arbeitsräume

Die Arbeitsräume und Möblierungen sind variabel gestaltet, um den kreativen Prozess zu unterstützen. Eine Vielzahl von Präsentationsmöglichkeiten (Whiteboard, Flipchart, Post-It, etc.) und Werkzeugen für den Prototypenbau (Papier, Lego, Stoffe, etc.) bieten Raum für den Austausch zwischen den Teammitgliedern und die spielerische Generierung und Umsetzung von Ideen.

Der Design Thinking Prozess

Design Thinking hat seinen Ursprung im Silicon Valley, in einer Umgebung, die von pragmatischem und schnellem unternehmerischen Handeln geprägt ist. Beim Design Thinking wird der Übergang von der Idee zu einem Prototypen, an dem die Anforderungen gespiegelt werden können, zügig vollzogen. Dies geschieht iterativ in sechs Phasen, wobei Rückkopplungen in vorherige Phasen möglich sind. In Phase 1, der „Verstehensphase“ versuchen sich die Teammitglieder mit Hilfe von Befragungen in die Nutzerperspektive hineinzuversetzen, um dann in Phase 2 „Beobachten“ zu verfolgen, wie Anwender mit den jeweiligen Problemstellungen umgehen. Basierend auf diesen Beobachtungen werden in der dritten Phase Sichtweisen definiert, die gewonnenen Erkenntnisse zusammengetragen und verdichtet und verschiedene Anwenderperspektiven herausgearbeitet, damit ein ganzheitlicher, 360 Grad Blick auf das Problem entsteht. Nach der Phase des „Einfühlens“ schließt sich die Ideenfindungsphase 4 an. Bei diesem Suchprozess spielt die Arbeitsumgebung und ihre Möglichkeiten für kreative Prozesse eine große Rolle. In der vorletzten Phase 5 fokussiert man sich auf die geeignetsten Ideen und überführt diese in in aufwandsarme Prototypen. Final werden diese in der sechsten Phase auf ihre Brauchbarkeit hin untersucht.

VOR- UND NACHTEILE VON DESIGN THINKING

Vorteile

- Das frühzeitige Prototyping erlaubt schnelle Rückschlüsse auf die Brauchbarkeit einer Lösung
- Design Thinking erlaubt die Integration von vielen Kreativitätstechniken
- Multidisziplinäre Teams erlauben umfassendere Perspektiven auf das Problem

Nachteile

- Erfahrene Moderation ist Voraussetzung
- Kreative Prozesse im Bereich der Ideenfindung sind nicht immer greifbar und Ergebnisse sind nicht zwingend gegeben
- Benötigt aufwendige Räumlichkeiten und ist zeitintensiv

2.3.2.5 FDD FEATURE DRIVEN DEVELOPMENT

Der agile Prozess FDD genießt nicht die gleiche Popularität wie Scrum und XP, hat aber Vorzüge, die das FDD für das hybride Projektmanagement interessant machen. Die Domäne von Scrum und XP sind kleine bis mittlere Teams von hochqualifizierten Spezialisten. Aus diesem Grund fällt die Skalierung schwer und geht häufig auf Kosten der ursprünglichen Agilität. FDD wurde von vorneherein für große Projekte konzipiert und schafft den Spagat von kleinen bis großen Teams deshalb ohne Umstrukturierung und Anpassungsprozesse. Anders als seine agilen Pendants Scrum und XP besitzt FDD eine ausgeprägtere Planungsphase im Vorfeld der Iterationen und wird deshalb häufig, zu Unrecht, mit einem phasengetriebenen Projektmanagementansatz verglichen.

A. DIE ROLLEN IM FDD

1. Project Manager

Er stimmt die Verteilung der Ressourcen ab und ist verantwortlich für die Terminierung des Projektes. An der eigentlichen Programmierarbeit hat der Project Manager keinen Anteil. Er ist verantwortlich für die administrativen Aufgaben im Projekt.

2. Chief Architect

Er behält den Überblick über die Architektur der gesamten Software und der zentralen Modelle.
Der Chief Architect unterstützt Entwickler und Kunden bei der gemeinsamen Entwicklung neuer Softwarekomponenten. Bei kleinen Projekten mit einer geringen Personaldecke verschmelzen die Funktion des Chief Architects und die Funktion des Chefprogrammierers zu einer Rolle.

3. Development Manager

Er ist der Moderator des Tagesgeschäftes und löst Ressourcenprobleme. Bei manchen Projekten wird die Rolle des Development Managers mit der des Chief Architects oder des Project Managers verknüpft.

4. Chief Programmer

Bei größeren Projekten führen erfahrene Chief Programmer die einzelnen Entwicklerteams, erstellen Feature Listen, planen und steuern deren Umsetzung und verantworten die ihnen zugeordneten Geschäftstätigkeiten.

5. Class Owners

Class Owners sind Entwickler, die in kleinen Teams für die Umsetzung der Features sorgen und sich mit ihren Chief Programmern und dem Chief Architect an deren Entwurfsplanung beteiligen. Sie sind den Key-Classes zugeordnet.

6. Domain Experts

Domain Experts sind Anwender, Kunden, Sponsoren, Geschäftsanalysten oder eine Kombination daraus. Sie haben eine profunde Kenntnis der Aufgabe (Domänenwissen), die das zu entwickelnde Produkt haben muss.

7. Unterstützende Roles

Neben den genannten, gibt es noch andere Rollen, die bei Bedarf eine unterstützende Funktion haben können: Language Guru, Release Manager, Build Engineer, System Administrator, Tool Smith.

8. Zusätzliche Roles

Diese sind Testers, Deployers und Technical Writers.

B. DIE PROZESSE IM FDD

1. Prozess: Entwicklung eines Gesamtmodelles

- Bildung des Modelling Teams aus Chief Programmern und Domain Experts
- Weitergabe einer Übersicht über den zu gestaltenden Domänenbereich durch den Domain Experten
- Durchsicht vorhandener Anforderungsdokumente und Referenzen
- Entwicklung von Modellen in kleinen Gruppen
- Zusammenführung der kleinen Modelle zu einem Team-Modell
- Iterative Verbesserung des Team-Modells
- Schriftliche Fixierung von wichtigen Anmerkung zum Team-Modell
- Festlegung der Key-Classes und Verifizierung des Modells in Form eines Assessments

2. Prozess: Erstellung einer Feature-Liste

- Formierung des Feature List Teams aus den Cief Programmern des Modelling Teams
- Erstellung der Feature Listen der einzelnen Systembereiche unter Leitung der Chefprogrammierer
- Die Ergebnisse aus Prozess 1 werden zuerst in Major Feature Sets (Themenbereiche) und dann in Business Aktivitäten (Feature Sets) untergliedert. Jeder Schritt (Step) in einer Aktivität wird als Feature identifiziert. Das Resultat dieser Zerlegung sind hierarchisch angeordnete, kategorisierte Feature Listen.
- Die Realisierung von Features darf den voraussichtlichen Zeitrahmen von 2 Wochen nicht überschreiten, ansonsten muss das Feature in Unterfeatures zerlegt werden.
- Verifizierung der Feature Listen in Form eines Assessments.

3. Prozess: Planen der Features

- Formierung des Planning Teams aus Project Manager, Development Manager und Chief Programmers
- Festlegung der Entwicklungssequenz
- Zuteilung der Business Aktivitäten (Feature Sets) zu den Chief Programmern
- Erstellung einer Class Owner List, in der die Developer als Betreuer der bekannten Key-Classes festgelegt werden.
- Erstellung eines Umsetzungsplanes inklusive Terminplanung unter Berücksichtigung von Abhängigkeiten und Komplexität der Features und der Verfügbarkeit der Ressourcen.
- Verifizierung der Ergebnisse in Assessment Format.

4. Prozess: Entwerfen der Features

- Formierung des Feature Teams durch den Chief Programmer
- Der Domain Expert gibt einen Überblick über die Domäne der zu realisierenden Features.
- Durchsicht der vorhandenen Anforderungsdokumente und Referenzen
- Entwicklung der Sequenz Diagramme
- Erstellung der Feature Modelle
- Inspektion des Designs

5. Prozess: Konstruieren der Features

- Implementierung der vorbereiteten Features aus Prozess 4
- Code-Inspektion
- Unit-Tests

Die Prozesse 4 und 5 werden in Schleifen geführt, bis alle Features umgesetzt sind und das Projekt abgeschlossen ist. Falls es erforderlich ist, können aufgrund der Erkenntnisse in der Umsetzung dem Gesamtpaket mehr Features in Schritt eins hinzugefügt werden.

C. VOR- UND NACHTEILE DES FDD

Vorteile

- Leicht auf größere Projekte und Teams skalierbar
- Besser für Teams, bei denen die Teammitglieder unterschiedliche Erfahrung mitbringen, geeignet
- Bietet definierte Projektverfolgungsmöglichkeiten
- Unterstützt das parallele Arbeiten von mehreren Teams

Nachteile

- Obwohl FDD ein Projekt mit einem vereinbarten Funktionsumfang darstellt, trägt es eine Unsicherheit bei komplexen Projektthemen in sich. Trotz sorgfältiger Vorplanung und Abschätzung können während des Entwickelns vorher unabsehbare Probleme oder Risiken auftreten, die die bislang gültige Leistungs-, Termin- und Ressourcenplanung hinfällig werden lassen.
- Der Programmcode wird nicht einem Team (shared ownership) sondern Einzelpersonen zugeordnet.
- Die Iterationsschritte sind undefiniert.

3. DIE AUSGANGSBASIS FÜR HYBRIDES PROJEKTMANAGEMENT

3.1 GRUNDSÄTZLICHE FORMEN DES HYBRIDEN PROJEKTMANAGEMENTS

Hybrides Projektmanagement entsteht durch die Erweiterung einer vorhandenen Basis mit neuen Elementen oder durch die Auswahl verschiedener Konzepte und die Installation ihrer Kombinationen.

Die folgenden Betrachtungen beschränken sich auf die drei wichtigsten hybriden Grundformate:

1. Die klassische Projektmanagementkultur wird durch agile Konzepte erweitert (Klassisch-Agil)
2. Eine Kombination aus agilen Konzepten (Agil-Agil)
3. Lean Projektmanagement ergänzt mit agilen Elementen (Lean-Agil)

3.2 DIE RICHTIGE WAHL DER PARTNERELEMENTE

Um ein geeignetes hybrides System zu erstellen ist es nötig, sich mit den möglichen Optionen und Anforderungen der Komponenten zu befassen und ausreichendes Wissen über sie zu erlangen. Im Folgenden sind die Auswahlkriterien näher beschrieben, um die Wahl zu vereinfachen. Außerdem sollte ein Prinzip des XP an dieser Stelle befolgt werden: „Travel Light“, d.h. sich auf das Notwendige beschränken und nicht mehr Systeme miteinander verknüpfen als für die zu bewältigende Aufgabe unbedingt erforderlich sind.

3.2.1 EINFACH, KOMPLIZIERT ODER KOMPLEX?

Projektvorhersagen unterliegen häufig einem hohen Unsicherheitsfaktor. Das Konzept der „Cone of Uncertainty“ beziffert die mögliche Spanne der tatsächlichen Projektaufwände auf das bis zu 4-fache des vor Projektbeginn abgeschätzten Wertes. Das ist umso deutlicher, je größer der Anteil von Forschung und Entwicklung am Projektergebnis sind. Grund ist die mangelhafte Planbarkeit dynamischer und komplexer Vorhaben. Klassisches Projektmanagement fordert die Vorabplanung des Projektes vor dem eigentlichen Arbeitsbeginn. Das ist machbar, wenn die Problemstellung einfach und das Projektleistungsziel weitgehend statisch ist. Auch komplizierte Problemstellungen lassen sich mit entsprechendem Wissen und Planungsaufwand in einem klassischen Projektumfeld ohne große Unsicherheiten lösen. Anders ist es bei komplexen und/oder dynamischen Projektzielen.

Während sich komplizierte Sachverhalte durch lineare Zusammenhänge auszeichnen, haben komplexe Themen interagierende Elemente, die weitreichende Vorhersagen unmöglich machen und sich einer planerischen Erfassung entziehen. Ist das Projektziel zusätzlich „beweglich“ d.h. das Projektziel wird erst über die Projektlaufzeit präziser abbildbar, kann man das Projekt immer nur „auf Sicht“ planen und inkrementell abarbeiten und folgt damit dem Ansatz des Lean Management oder agilem Projektmanagement.

Entscheidungsmatrix für den zu wählenden Ansatz bei einem Projekt oder Projektteil

Projektziel	Umsetzung des Projektziels	Vorteilhafter Projektmanagementansatz
weitgehend statisch	einfach	klassisch
weitgehend statisch	kompliziert	klassisch oder agil
weitgehend statisch	komplex	klassisch, agil oder lean
dynamisch	einfach	agil oder lean
dynamisch	kompliziert	agil oder lean
dynamisch	komplex	agil oder lean

3.2.2 ANFORDERUNGEN, DIE SICH AUS DEM PROJEKT, DER BRANCHE UND DEM TEAM ERGEBEN

1. PROJEKTART UND BRANCHE

Während das klassische Projektmanagement die Projektarten Organisations-, Investitions- und Entwicklungsprojekt in allen Branchen abdeckt, sind viele der agilen Pendants ihrem Ursprung nach spezifisch auf Entwicklungsprojekte in der IT Branche ausgelegt. Deshalb gilt es zu prüfen, ob der Einsatz entsprechender agiler Vorgehensweisen machbar und sinnvoll ist. In einigen Fällen ist die Übertragbarkeit völlig problemlos, in anderen Fällen sind Anpassungsvorgänge zwingend erforderlich, manchmal ist es aber eben auch nicht möglich bestimmte Vorgehensweisen außerhalb ihrer „Heimatbranche“ anzuwenden.

Beispiel 1: XP Techniken wie z. B. „Pair Programming“ sind rein für die Erstellung von Software konzipiert und können bestenfalls von ihrer Grundidee in andere Branchen übernommen werden.

Dem Original entsprechen sie dann jedoch nicht mehr und der Anwendernutzen dieses Vorgehens ist im Einzelfall zu

prüfen. Im Fall „Pair Programming“ gibt es inzwischen adoptierte Anwendungsbeispiele aus dem juristischen („Pair Contracting“) und gestalterischen („Pair Designing“) Bereich.

Beispiel 2: Events wie z.B. der „Daily Scrum“ im Regelwerk „Scrum“ sind einfach auf andere Branchen übertragbar und sorgen in beliebigen Umgebungen für eine Verbesserung des Informationsaustausches.

2. PROJEKTUMFANG UND TEAMGRÖSSE

Jeder Projektmanagementansatz hat seinen eigenen optimalen Arbeitsbereich und sollte auch in einer hybriden Struktur entsprechend angewandt werden. Das soll nicht heißen, dass ein gewählter Ansatz außerhalb seiner Komfortzone nicht funktioniert, aber bestimmte Einschränkungen und zusätzliche Aufwände können durchaus entstehen.

Beispiel 1: Klassisches Projektmanagement erfordert einen umfassenden, organisatorischen Aufwand, der bei Kleinstprojekten unnötig und deshalb ineffizient ist.

Beispiel 2: Das Scrum Regelwerk eignet sich sehr gut für kleine bis mittlere Aufgaben und Teams. Bei Großprojekten muss Scrum skaliert werden. Dafür gibt es verschiedene Skalierungsansätze, die aber aufgrund der potentiell ausufernden Kommunikationskanäle und Herausforderungen bei der Synchronisation der einzelnen Scrum Teams und deren Ergebnisse, in der Realität häufig an Agilität verlieren (z.B. bei der Auslieferung von Inkrementen).

Vorschlagsmatrix für den zu wählenden Ansatz bei einem Projekt oder Projektteil

Projekt- bzw. Teamgröße	Vorteilhafter Projektmanagement-Ansatz
klein - mittel	lean oder agil
mittel - groß	klassisch, lean, agil
groß - sehr groß	klassisch, FDD, Kanban oder Design Thinking

3. KOMMUNIKATIONSKULTUR UND STAKEHOLDERANBINDUNG

Verteilte Projektteams und Stakeholder stellen eine zusätzliche Herausforderung dar. Gerade agile Vorgehensweisen fokussieren sehr stark auf die kommunikative Einbindung der Beteiligten und ein häufiger und intensiver Austausch innerhalb der Teams, wie auch zum Kunden und den Stakeholdern gehört zum Pflichtprogramm. Zur hybriden Gestaltung der Projektmanagementmethode gehört neben der Auswahl der Komponenten auch die Bereitstellung entsprechender Kommunikationsplattformen, um den Austausch zwischen den Teams untereinander, als auch dem Kunden und anderer Stakeholder, sicherzustellen.

4. WAS WIRD BENÖTIGT? EIN WERKZEUG, EINE METHODE ODER EINE KOMPLETTE STRUKTUR?

Abhängig von dem zu lösenden Problem ist festzulegen, ob man ein komplettes Projektmanagement-Regelwerk benötigt oder ob eine Methode, ein philosophischer Ansatz oder auch nur ein Werkzeug zur Unterstützung der vorhandenen Struktur vollkommen ausreichend ist. Dabei ist es durchaus denkbar Teile aus den vorhandenen Projektmanagement-Ansätzen herauszulösen und in die eigene Projektlandschaft zu integrieren. Voraussetzung ist eine breite Kenntnis der unterschiedlichen zur Verfügung stehenden Möglichkeiten.

Beispiel 1: In einem klassischen Projekt-Umfeld kommt es zu Interaktions- und Kommunikationsproblemen, die den Projektablauf verzögern. Der Projektmanager entscheidet sich zusammen mit seinem Team IT-Kanban im Flight-Level 1 einzuführen, um die Wertschöpfungskette zu visualisieren und das gegenseitige Verständnis im Team zu fördern. Abhängig vom Erfolg des hybriden Ansatzes erwägt man die Einbeziehung weiterer Teams.

Beispiel 2: In einem Scrum Projekt entscheidet sich das Development Team für den Einsatz von XP Techniken zur Durchführung der Sprints, um eine hohe Qualität der erzeugten Software sicherzustellen.

3.2.3 DAS ORGANISATIONSUMFELD UND DIE KULTURELLEN ASPEKTE DES WANDELS

Üblicherweise sind Abläufe und Regeln innerhalb eines Unternehmens so angelegt, dass sie den inneren Aufbau der Organisation bestärken. Gleichzeitig steht das Zusammenspiel der hybriden Struktur in engem Zusammenhang mit der sie einschließenden Unternehmenskultur.

Während sich die klassischen Projektmanagementstrukturen sehr gut in hierarchischen Strukturen einfügen, leben leane Elemente von einer starken Beteiligung der Projektmitarbeiter. Viele agile Komponenten verzichten schließlich nahezu vollständig auf hierarchische Strukturen, verlangen Selbstorganisation und setzen auf Führung statt auf Management. Agiles Arbeiten kann nicht befohlen werden („sei agil!“) sondern entwickelt sich durch die Bereitstellung eines geeigneten Projektumfelds.
Vor diesem Hintergrund müssen strukturelle Anpassungsprozesse gestartet werden, um die gewählten hybriden Elemente optimal einsetzen zu können.

1. Klassisches Projektmanagement

- Hierarchische Organisationsstruktur
- Top-Down Zuweisung von Projektbeteiligten
- Projektmitarbeiter werden von Personalverantwortlichem geführt
- Vorgesetzter muss die Arbeiten zuweisen und die Ergebnisse abholen
- In der Matrixorganisation haben Projektmitarbeiter mehr als einen Vorgesetzten

2. Lean Management

- Hierarchische Organisation
- Top-Down Zuweisung von Projektbeteiligten
- Projektmitarbeiter werden von Personalverantwortlichem geführt
- Aufgaben werden zusammen mit dem Vorgesetzten kooperativ festgelegt
- Meinungen und Verbesserungsvorschläge der Teammitglieder werden mit Vorgesetzten diskutiert und bei Zustimmung verwirklicht.

3. Agiles Projektmanagement

- Selbstorganisierte Teams
- Top-down Zuweisung von Mitarbeitern soll vermieden werden
- Führung der Teams durch Zielvorgaben
- Mitarbeiter müssen mit der Selbstorganisation und der damit einhergehenden Übernahme von Verantwortung einverstanden sein.
- Mitarbeiter machen ihre Arbeitsleistung transparent.
- Einsatz von Moderatoren um die Teammitarbeiter in ihrer Arbeit zu unterstützen und Konfliktsituationen aufzulösen.
- Installation von neuen internen Prozessen notwendig, die in hierarchischen Organisationsformen von disziplinarischen Vorgesetzten übernommen werden. Beispiele dafür sind Personalbewertung und -entwicklung, Weiterbildung und Urlaubsplanung.

3.3 DIE MOTIVATION FÜR HYBRIDES PROJEKTMANAGEMENT

Die Motivation für den Einsatz von hybridem Projektmanagement prägt in entscheidender Weise die zu bewältigenden Transformationsprozesse und die erreichbaren Ziele. Generell ist es wichtig, dass die Zielvorstellungen direkt gekoppelt sind mit den Möglichkeiten, die ein hybrider Ansatz bzw. das Hinzufügen von Elementen bieten kann.

1. Auflösung einer akut problematischen Situation in einem klassischen Projektumfeld

- Derzeitiger Projektansatz vermag es nicht, das Projektziel unter Einhaltung der Vorgaben zu erreichen. Ein entsprechender Leidensdruck ist bereits entstanden.
- Eine Konfliktsituation bzgl. des Projektfortschrittes bzw. -ergebnisses zwischen Kunde und Auftragnehmer verschärft sich.
- Im Vorfeld eines Projektes wird den Projektbeteiligten klar, dass das Projekt ganz oder in Teilen schwer zu planen ist.
- Dem Auftraggeber fällt es schwer das Projektziel präzise und vollständig zu formulieren.

Ziel des hybriden Ansatzes: Durch Einsatz leaner und agiler Elemente die Kundenanbindung zu verstärken, inkrementelle Entwicklung der für den Kunden hochprioren Teilziele vorantreiben und nach Rücksprache mit dem Kunden möglicherweise vorerst Beschränkung auf die wichtigsten Ziele.

2. Hybrides Projektmanagement als Übergangsprozess von klassischem zu agilem Projektmanagement

- Das Unternehmen verspricht sich deutliche Vorteile von einer neuen Unternehmensstruktur und einem agileren Ansatz in der Projektbearbeitung.
- Der „Sprung ins kalte Wasser“ ist für das Unternehmen kein geeignetes Mittel der Wahl, da man etwaige Risiken der Umorganisation nicht abschätzen kann.

Ziel des hybriden Ansatzes: Dieses Vorgehen gibt Sicherheit und schafft für die Beteiligten Raum, um sich an die neuen Prozesse und Regeln anzupassen, bevor man eine endgültige und vollständige Transition vollzieht. Die hybriden Zwischenstationen müssen vorab festgelegt und nach der Einführung auf Zweckmäßigkeit geprüft werden.

3. Verbesserung der Kundenanbindung in einer klassischen Projektumgebung

- Die derzeitige Kommunikation mit dem Kunden ist nicht immer geeignet dessen Wünsche adäquat im Projektergebnis abbilden zu können.
- Das ist mehreren Gründen geschuldet: Unterschiedliche „Sprache“ zwischen Kunde und Auftragnehmer/ Entwickler und dem vagen Kundenwunsch.

Ziel des hybriden Ansatzes: Einbindung von Praktiken und Kommunikationswerkzeugen, um besser auf den Kunden eingehen zu können, ohne den zentralen geplanten Projektmanagementansatz zu verlieren. Eventuell Aufgliederung des Projektansatzes in eine agile Vorentwicklung und eine traditionelle Umsetzung in die Breite.

4. Einem Trend folgen und agile Elemente in eine vorhandene klassische Projektumgebung einführen

- Agile Projektansätze werben mit einer Vielzahl von Vorteilen gegenüber bestehenden klassischen Strukturen.
- In vielen Publikationen werden Berichte die positiven Erfahrungen von Unternehmen, die agile Projektansätze eingeführt haben, veröffentlicht.
- Die ständige Wiederholung von agilen Begriffen verstärkt das Gefühl „dabei sein“ zu müssen.

Ziel des hybriden Ansatzes: Im Projektmanagement den Anschluss an neue Führungsmethoden und sich wandelnde Arbeitswelten nicht zu verlieren.

Wichtig: Immer wieder werden agile Projektansätze aber mit Adjektiven versehen, denen sie in der Realität nicht gerecht werden können. Bevor man einen hybriden Ansatz wählt, ist es zwingend erforderlich sich mit ausreichend Wissen und einen breiten Überblick über die möglichen Partnerelemente zu informieren um die eigene Organisation vor Fehlentwicklungen zu schützen. Agil heißt nicht schneller, sondern beweglicher, anpassungsfähiger!

5. Verbesserung der Performance

- Die Leistungsfähigkeit des bestehenden Systems erscheint nicht ausreichend.
- Qualitätsmängel Kommunikationsprobleme und Silodenken
- Es sind keine geeigneten Werkzeuge, Methoden und Vorgehensweisen definiert

Ziel des hybriden Ansatzes: Einführung von Elementen, die die eigenen Strukturen effektiver und effizienter gestalten, die Qualität und das Miteinander zu stärken und eines Arbeitsumfeld zur Verfügung zu stellen.

6. Andocken an andere Projektkulturen

- Mitunter arbeiten Partnerfirmen oder Kunden mit anderen Projektansätzen, die mit der eigenen Projektkultur inkompatibel sind.
- Gesetzliche Vorgaben und Regelungen empfehlen oder verlangen bestimmte Projektschemata einzuhalten
- Die Zusammenarbeit mit Mitarbeitern, Partnern und Kunden aus dem internationalen Umfeld verlangt explizit ein kulturell angemessenes Projektmanagement.

Ziel des hybriden Ansatzes: Setzen von Andockpunkten, um eine zweckmäßige Projektschnittstelle zum Partner anbieten und Vorgaben einzuhalten zu können. Die gewählten Projektmanagementansätze sollen kulturverträglich ausgewählt werden um die Integration internationalen Mitarbeiter in die Projektorganisation zu ermöglichen und die erwarteten Arbeitsergebnisse erzielen zu können.

4. DER WEG ZUM HYBRIDEN PROJEKTMANAGEMENT-MODELL

Es gibt keine fertigen Lösungen, keine Best Practice Blaupausen, sondern bestenfalls Konzepte, die man empirisch testen und verfeinern kann und muss. Jede Organisation, ihre Mitarbeiter und ihre Projekte sind im Zusammenhang einzigartig und brauchen eine individuelle, maßgeschneiderte Lösung.

4.1 STRUKTUREN DES HYBRIDEN PROJEKTMANAGEMENTS IM PROJEKTKONTEXT

Betrachtet werden nicht nur die Möglichkeiten für eine einzelnes Projekt sondern auch die organisatorische Aufstellung für einen Projektpool bzw. eine Umgebung für Multiprojektmanagement.

1. Incubator

Separate „behütete“, Projektumgebung innerhalb einer Organisation um unter unabhängigen Bedingungen einen neuen Projektmanagementansatz testen zu können. Ziel ist das Benchmarking mit Projektergebnissen außerhalb des Incubators. Bei erfolgreichem Abschluss kann das Projektteam die Keimzelle für die Übertragung der angewandten Regelwerke,Methoden und Werkzeuge in andere Projekte bilden.

2. Containerlösung

Einige der Projektansätze bilden einen Rahmen aus Regeln oder Vorschriften, die aber noch mit Inhalten gefüllt werden können. Bei der Containerlösung bettet man in diesen vorhandenen Rahmen ein zweites System, wobei beide in sich selbst unabhängig voneinander existieren und sich gegenseitig nicht beeinträchtigen.

Beispiel 1 (Klassisch-Agil): In einer klassisch geprägten Projektumgenbung wird IT-Kanban zur Kommunikationsverbesserung, Steuerung des WiP und der Risikominimierung eingeführt.

Beispiel 2 (Agil-Agil): Ein Development Team in einem Scrum Projekt beschließt in ihren Sprints XP einzusetzen um auf einen Pool gemeinsamer Techniken bei der Softwareentwicklung zurückgreifen zu können.

3. Partielle Assimilation

Ein bestehendes Projektmanagementkonzept wird durch Bestandteile eines zweiten Konzeptes erweitert. Die neuen, hinzugefügten Strukturen werden vollständig im bestehenden Projektmanagement integriert.

Beispiel 1 (Klassisch-Agil): In einem klassischen Projektansatz soll die Selbstreflektion und Kommunikation des Teams gestärkt werden. Der Projektleiter beschließt mit dem Team im zweiwöchigen Rhythmus „Reviews und Retrospektiven" abzuhalten, um sowohl die Arbeitsergebnisse, als auch die Arbeitsmethoden kritisch zu beleuchten.

Beispiel 2 (Klassisch-Agil): Ein klassisch geprägtes Unternehmen trifft mit einem Kunden die Übereinkunft das Lastenheft zusammen mit einem Teil des Projektteams zu entwerfen, um schon im Vorfeld die Wünsche des Kunden besser verstehen zu können und möglicherweise dem Kunden neue Wege aufzeigen zu können.

4. Synthese

Im Gegensatz zur Containerlösung werden hier zwei unterschiedliche Projektmanagementansätze zu einem neuen hybriden Ansatz verschmolzen, um etwaige Defizite der Einzelsysteme aufzuheben.

Beispiel (Agil-Agil): Eines der bekanntesten Systeme ist Scrumban, dass aus einer Verbindung von Scrum mit IT-Kanban entstanden ist. Bei Scrumban wird die zyklische Struktur in einen kontinuierlichen Arbeitsfluss überführt. Die Planung für den Sprint und die Bevorratung von Themen im Product Backlog werden nicht mehr zu fest vorgegebenen Zeitpunkten, sondern bedarfsorientiert durchgeführt. Die Scrum üblichen Burndown Charts werden durch Kanban Metriken abgelöst.

5. Parallelwelten

Vor Projektstart werden je nach dem Bedarf der Teilprojekte oder der Arbeitspakete entsprechende Projektansätze zugeordnet und danach individuell bearbeitet. Voraussetzung ist, dass die zu bearbeiteten Module voneinander unabhängig entwickelt werden können.

Beispiel (Klassisch-Agil): Bei einem Projekt werden Hardware- und Softwareanteil unterschiedlich entwickelt. Dazu werden im Vorfeld bestimmte einzuhaltende Schnittstellenbedingungen für die Integration der Module am Projektende definiert. Im weiteren Verlauf werden die Entwicklungsstränge unabhängig geführt. Die Hardwareentwicklung traditionell, die Softwareentwicklung mit einem agilen Ansatz. Nach der Fertigstellung werden die Ergebnisse zusammengeführt.

6. Reißverschluss-Lösung

Im Projektverlauf gibt es Teilprojekte oder Arbeitspakete, die in Abhängigkeit zu anderen Teilprojekten/Arbeitspaketen stehen, aber vom Projektansatz her einer unterschiedlichen Behandlung bedürfen. In diesem Fall bietet sich ein Reissverschluss-Verfahren an. Der hybride Ansatz ähnelt dem Vorgehen „Parallelwelten" aber die Vereinbarkeit der Ergebnisse der Teilstränge wird durch Synchronisationsmaßnahmen sichergestellt.

Beispiel (Klassisch-Agil): Bei einem Projekt werden Hardware- und Softwareanteil unterschiedlich entwickelt. Da die beiden Teilsysteme miteinander eng verknüpft sind, gibt es Mitarbeiter, die in beiden Teams eingegliedert sind und deren Aufgabe es ist die synchrone Gestaltung der Teilergebnisse sicherzustellen. Damit werden die Entwicklungsstränge in loser Abhängigkeit voneinander geführt und an Synchronisationspunkten die Vereinbarkeit der Ergebnisse geprüft. Die Hardwareentwicklung verläuft traditionell, die Softwareentwicklung mit einem agilen Ansatz. Nach der Fertigstellung werden die Ergebnisse zusammengeführt. Dieser hybride Ansatz, macht nur dann Sinn, wenn davon ausgegangen werden kann, dass der Umfang und die Anzahl der Änderungen auf der Seite der Hardwareentwicklung den traditionellen Ansatz nicht überstrapazieren.

7. Gekoppelte Systeme

Eine weitere Anordnung von Projektansätzen zu einem neuen hybriden Modell sind gekoppelte Systeme. Während manche hybriden Ansätze dazu gedacht sind, sich in eine neue Projektmanagementkultur hineinzutasten, ist der gekoppelte Ansatz ein sehr präzises Werkzeug, um bestimmte Ergebnisse zu erzielen. Dazu sind eine genaue Kenntnis der unterschiedlichen Projektansätze und ihrer Wirkungen Pflicht.

Beispiel (Agil-Lean-Agil): Ziel ist ein risikoarmer, schneller und agiler Start-Up Prozess für ein neues Produkt. Ein Kanon aus Projektmanagement-Varianten soll diese Anforderungen sicherstellen und die Entwicklung von der Produktidee bis zur Markteinführung einer Serie von Produktfeatures begleiten. Die Lösung ist ein gekoppeltes Sytem aus Design Thinking, Minimum Viable Product (MVP) und Scrum. Mit Hilfe von Design Thinking wird eine Produktidee in einen ersten testbaren Prototypen übersetzt und mit dem Kunden abgeglichen. Der Prototyp wird mit Hilfe von MVP in ein Erstes, auf das Notwendigste beschränktes Produkt überführt und die Marktreife geprüft. Bei Erfolg der Markteinführung dient ein Scrum Ansatz dazu, die Produktpalette inkrementell zu erweitern.

4.2 HERAUSFORDERUNGEN DER HYBRIDEN KULTUR

Die Einführung einer neuen Projektmanagementkultur bringt zunächst viele Erschwernisse mit sich und der damit verbundene Aufwand, soll neben den zu erwartenden positiven Effekten, nicht vergessen werden. Es ist wie bei der Gesundung eines Patienten. Bevor es besser wird, wird es häufig erst einmal schlechter. Eine Veränderung zu einem neuen Projektmanagementansatz und damit einhergehend evtl. auch einem Wechsel in der Unternehmenskultur erfordert Zeit.

Organisationen sind häufig resistent gegenüber Veränderungen. Mitarbeiter müssen mitgenommen und geschult werden. Leider muss man sich auch manchmal von Mitarbeitern trennen. Nicht jeder fühlt sich in einer neu strukturierten Projekt- und Organisationswelt willkommen und gebraucht. Die rechtliche Rahmenbedingungen sind zu prüfen und interne Unterlagen anzupassen und bestehende Qualitätsvorgaben müssen auf den neuen Stand gebracht werden (z.B. ISO 9001). Schließlich wird zu prüfen sein, ob die hybriden Systeme den an sie gestellten Anforderungen genügen oder ob Nachbesserungen oder ein Alternativansatz notwendig sind.

5. SCHLUSSBEMERKUNG – WAS KOMMT DANACH?

Am Ende gilt es zu prüfen ob das gefundene hybride Gesamtkonzept mehr ist als die Summe seiner Einzelkomponenten, denn nur dann war man bei der Einführung wirklich erfolgreich. Was danach folgt hängt im Einzelnen von den Führungsgremien der Unternehmen ab. Wählt man einen langsamen evolutionären Übergang von einer klassisch geprägten Projektwelt in eine agile/leane Projektwelt oder vollzieht man den Paradigmenwechsel in Form einer Revolution? Schreitet man inkrementel von einer hybriden Lösung zur nächsten um sich jeweils die positivsten gefundenen Ansätze in seinem Werkzeugkasten einzuverleiben oder konsolidiert man sich auf dem neu gefundenen hybriden Projektmanagementansatz? Vermutlich wird es immer weniger möglich sein, sich einer fortwährenden Veränderung unserer Projektkultur zu entziehen, da sich auch unsere Arbeitswelt in einem beständigen Wandel in kürzer werdenden Zyklen befindet. Hybrides Projektmanagement wird dabei ein wertvoller Beitrag sein, diese Veränderung in unseren Projekten angemessen zu begleiten.

IMPRESSUM

www.iapm.net

1. Auflage
Copyright © IAPM 2017

Fotografie: www.istockphoto.com

Herausgegeben von der
IAPM International Association of Project Managers™
in Liechtenstein

ISBN: 978-3-941739-31-4

Qualitätsmanagement-System
Das Qualitätsmanagement-System der
IAPM International Association of Project Managers™
genügt den Anforderungen der ISO 9001.

Markenschutz
IAPM International Association of Project Managers™
ist eine geschützte EU-Marke – Nr. 9539354 –

Die IAPM ist ein im Fürstentum Liechtenstein eingetragener ideeler Verein:
Nr. FL-0002.353.470-6

iapm
INTERNATIONAL ASSOCIATION OF PROJECT MANAGERS

DIE VORTEILE

EINER IAPM-ZERTIFIZIERUNG

1

Wettbewerbsvorteile & Karrieresprungbrett

- Nachweisliche Kompetenz/Erfahrung
- Wettbewerbsvorteile für Unternehmen und Einzelpersonen
- Standardisierung von Begriffen und Methoden
- Externe, objektive Bestätigung von Wissen

2

Onlineprüfverfahren

- Keine Anreisekosten
- Kein Zeitdruck in der Vorbereitung
- Durchführbar an jedem beliebigen PC und Tablet

3

Keine Rezertifizierung notwendig

- Kein Verfallsdatum des Zertifikats
- Keine erneuten Kosten

4

Faire Preise

- Kosten berechnen sich nach dem Bruttoinlandsprodukt des Landes, dessen Staatsbürgerschaft der Zertifikant innehat

5

Personenneutrale Zertifizierung

- Keine subjektive Bewertung des Zertifikanten
- Keine „Durchfallerquote“
- Keine Diskriminierung des Zertifikanten