



AGILE BUSINESS BODY OF KNOWLEDGE

Agile Business Body of Knowledge (ABBoK)

Verfasst vom International Institute for
Agile Business Certification (IIABC.org)

November 2021

Stoffübersicht zum Bestehen der Prüfungen für
die folgenden Zertifikate:



INHALTSVERZEICHNIS

WESHALB DER BUSINESS BODY OF KNOWLEDGE ERSTELLT WURDE	3
AGILE	3
DIE VIER WERTE	3
DIE ZWÖLF PRINZIPIEN	4
WANN AGILE ARBEITEN UND WANN NICHT?	4
DAS EISERNE DREIECK AUF DEM KOPF	5
AGILE PROJEKTE ODER TEAMS STARTEN	6
VORBEREITUNGEN	6
DIE FORMULIERUNG DER VISION	6
DAS ERSTELLEN DES ERSTEN PRODUCT BACKLOGS	7
SCRUM	8
EMPIRISMUS	8
WERTE	8
ROLLEN	9
Product Owner	9
Developer	9
Scrum Master	9
DER SPRINT	10
Sprint Planning	11
Daily Scrum	12
Review	12
Retrospective	13
Product Backlog Refinement	13
ARTEFAKTE	14
Product Backlog	14
Sprint Backlog	14
Inkrement	15
Definition of Done	15
Fortschritt überwachen	15
KANBAN	15
PRAKTIK 1: ARBEITSFLUSS VISUALISIEREN	16
PRAKTIK 2: MENGE ANGEFANGENER ARBEIT LIMITIEREN	16
PRAKTIK 3: ARBEITSFLUSS MANAGEN	17
PRAKTIK 4: VEREINBARUNGEN EXPLIZIT MACHEN	17
PRAKTIK 5: FEEDBACKSCHLEIFEN EINBAUEN	18
PRAKTIK 6: GEMEINSAM IN KLEINEN SCHRITTEN BESSER WERDEN	18
LITERATURLISTE	19

WESHALB DER AGILE BUSINESS BODY OF KNOWLEDGE ERSTELLT WURDE

Der vorliegende Agile Business Body of Knowledge wurde vom International Institute for Agile Business Certification (iiabc.org) zusammengestellt. Das Dokument ist in folgender Hinsicht einzigartig:

1. iiabc.org zielt auf die Anwendung agiler Arbeitsweisen in der gesamten Organisation ab. So macht das IIABC aktuelle Erkenntnisse bzgl. agilen Arbeitens (sonst v.a. von IT-Teams praktiziert) für alle zugänglich.
2. Viele zertifizierende Instanzen beschäftigen sich in ihrem Body of Knowledge bis in die Tiefe mit einem spezifischen Framework. Das IIABC hingegen präsentiert eine Übersicht von (i) Agile im Allgemeinen sowie (ii) Scrum und Kanban im Besonderen. Das befähigt Teilnehmer, den Einsatz von Agile in der Organisation in seiner Gesamtheit zu betrachten. Denn:
 - a. Wissen über die Werte und Prinzipien von Agile schafft die allgemeine Basis für die Entwicklung eines zu Agile passenden Mindsets;
 - b. Kenntnis von Scrum und Kanban wird benötigt, um sowohl prozessbezogene als auch operationelle Arbeiten zu strukturieren.

Die in diesem Body of Knowledge wiedergegebene Theorie wurde größtenteils von Schlüsselfiguren der Agile-Bewegung entwickelt. Das IIABC möchte mit diesem Dokument keinesfalls den Eindruck erwecken, es behaupte, die ursprüngliche Quelle zu sein. Die Art und Weise, in der die Theorie präsentiert wird, ist jedoch einmalig: Eine Kombination von Agile, Scrum und Kanban. Und davon profitieren die Leserinnen und Leser.

AGILE

Das Gedankengut rund um Agile ist aus der Notwendigkeit heraus entstanden, dass sich Organisationen an eine schnell verändernde Umwelt anpassen müssen. Nur Organisationen, denen das gelingt, bleiben relevant und können langfristig überleben. Eine weitere Antriebsfeder waren von althergebrachten rigiden Organisationsstrukturen und Projektmanagement-Methoden hervorgerufene Frustrationen. Deshalb wurde im Jahr 2001 im Agilen Manifest eine Reihe neuer Werte und Prinzipien formuliert. Sie bilden immer häufiger die Basis für agile Organisation.

Die vier Werte

Die untenstehenden vier Werte sind aus dem ursprünglichen Agilen Manifest, frei übersetzt. Das ermöglicht eine breitere Nutzung über die Grenzen von Organisationen im IT-Bereich hinaus.

Agile Teams wertschätzen:

- **Menschen und ihre Interaktion** vor Prozessen und Hilfsmitteln;
- **Abgeschlossene (Teil-)Projekte** vor allumfassender Dokumentation oder Plänen;
- **Zusammenarbeit mit dem Kunden** vor Vertragsverhandlungen;
- **Reagieren auf Veränderung** vor Festhalten an Plänen.

Natürlich sind auch die in der Aufzählung rechts stehenden Werte von Bedeutung. Aber in agilen Organisationen liegt der Schwerpunkt auf den Werten, die links aufgeführt sind.

Die zwölf Prinzipien

Danach wurden die vier Werte in zwölf Prinzipien überführt. Gemeinsam bilden Werte und Prinzipien den Rahmen dafür, wie in agilen Organisationen und Teams zu handeln ist. Um einen Einsatz in unterschiedlichen Bereichen zu ermöglichen, sind auch die Prinzipien im Folgenden frei übersetzt:

1. Kundenzufriedenheit ist immer die oberste Priorität.
2. Änderungen bei den Projektinhalten sind immer willkommen; auch in späten Projektphasen.
3. Arbeitsergebnisse in kurzen Zyklen an den Auftraggeber übergeben.
4. Jeden Tag multidisziplinär zusammenarbeiten.
5. Projekte werden von motivierten Mitarbeitern durchgeführt. Dabei werden ihnen die Unterstützung und das Vertrauen zuteil, das sie benötigen.
6. Am effizientesten lassen sich Informationen in persönlichen Gesprächen weitergeben ("face-to-face").
7. Vollständig fertiggestellte Arbeit ist der wichtigste Indikator für Fortschritt.
8. Ein konstantes und nachhaltiges Innovationstempo anstreben, das die Organisation dauerhaft bewerkstelligen kann.
9. Kontinuierliches Augenmerk auf eine hohe Qualität
10. Einfachheit ist entscheidend. Reduziere Arbeit, die wenig Wert liefert, auf ein Minimum.
11. Die besten Produktideen stammen von selbstorganisierten Teams mit einem hohen Grad an Autonomie.
12. Zusammenarbeit und inhaltliche Expertise werden verbessert, indem in regelmäßigen Abständen Gelegenheit zur Reflexion geschaffen wird.

Ausgehend von den obenstehenden Prinzipien wurden in der Praxis viele Arbeitsmethoden und Regelwerke entwickelt. Dieser Body of Knowledge beschäftigt sich mit den beiden am häufigsten verwendeten Konzepten: Scrum und Kanban.

Wann Agil arbeiten und wann nicht?

Agile Arbeitsweisen sind kein Wundermittel. Ihr Einsatz empfiehlt sich zudem nur dort, wo sich der Kontext dafür anbietet. Die untenstehende Tabelle 2 zeigt, in welchem Kontext Agile für Organisationen zielführend sein kann.

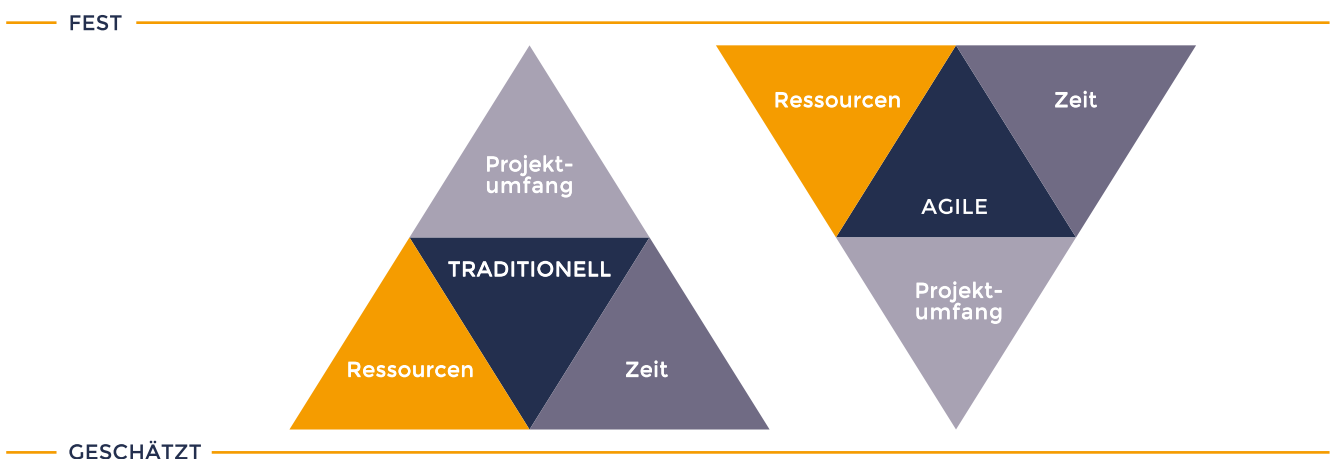
MERKMAL	GÜNSTIG FÜR AGILE	UNGÜNSTIG FÜR AGILE
Marktsituation	Wenn sich Kundenwünsche und Technologie schnell verändern bzw. entwickeln	Wenn die Umgebung der Organisation beständig ist
Einbindung von Kunden	Wenn Zusammenarbeit mit Kunden und regelmäßige Evaluation wünschenswert sind	Wenn Kundenwünsche beständig sind und auf lange Sicht relevant bleiben oder wenn Kunden für eine Zusammenarbeit nicht zur Verfügung stehen

Art der Innovationsaufgabe	Wenn eine Organisation es mit komplexen Herausforderungen zu tun hat, ohne dass eine logische Lösung im Vorhinein bekannt wäre sowie bei vagem Projekthinhalte	Wenn klar ist, wie für die Organisation der Weg in die Zukunft aussieht, detaillierte Pläne vorhanden sind sowie verlässliche Voraussagen getroffen werden können
Teilbarkeit der zu verrichtenden Arbeit	Wenn sich die Arbeit im Projektportfolio in kleine selbständige Teile unterteilen lässt	Wenn ausschließlich ein komplett fertiggestelltes Endprodukt von Kunden getestet werden kann
Machen von Fehlern	Wenn Fehler gemacht werden dürfen, um aus ihnen zu lernen	Wenn Fehler gravierende Konsequenzen für die Organisation hätten
Organisationskultur	Wenn wenig Bürokratie vorherrscht und die Organisation autonom und multidisziplinär arbeitenden Teams offen gegenübersteht	Wenn es viel Bürokratie in der Organisation gibt, die Zusammenstellung von Teams innerhalb von Silos erfolgt und vollständige Kontrolle über das Handeln der Teams gewünscht ist

Das eiserne Dreieck auf dem Kopf

Aus dem traditionellen Projektmanagement ist das "eiserne Dreieck" hervorgegangen. Dieses Dreieck zeigt auf, wie sich die Parameter Projektumfang, Zeit und Ressourcen gegenseitig beeinflussen. Ist ein Parameter begrenzt, hat das unmittelbare Folgen für mindestens einen der anderen Parameter. So kann beispielsweise eine Reduzierung der Ressourcen (Menschen) eine Verkleinerung des Projektumfangs oder reine Verlängerung der benötigten Zeit implizieren.

- **Projektumfang:** angestrebtes Arbeitsergebnis
- **Ressourcen:** Menschen und finanzielle Mittel, die benötigt werden, um das Arbeitsergebnis zu erzielen
- **Zeit:** ein Plan, der aufzeigt, was zu welchem Zeitpunkt fertiggestellt wird. In der Vergangenheit wurde der Projektumfang zu Beginn festgelegt. Davon ausgehend schätzte man ein, wie viel Ressourcen und Zeit zur Umsetzung benötigt werden. Heutzutage hat die Komplexität von Projekten stark zugenommen. Hierdurch ist es meistens nicht wünschenswert und oft auch nicht möglich, den Projektumfang vorab zu definieren. Agile Organisationen drehen das Dreieck deshalb um: Ressourcen und Zeit werden festgelegt. Beim Projektumfang handelt es sich um eine Schätzung. Im Projektverlauf wird der Projektumfang neuen Erkenntnissen entsprechend ständig optimiert.



AGILE PROJEKTE ODER TEAMS STARTEN

Vorbereitungen

Bevor mit einem agilen Projekt oder Team begonnen werden kann, sind eine Reihe von Vorbereitungen zu treffen. Die nachfolgende Liste stellt dar, was zu gewährleisten ist. Sie ist nicht für jede Situation relevant. Aber sie zeigt einige wichtige Rahmenbedingungen auf, die in den gängigen Agile-Regelwerken keine Berücksichtigung finden.

Auf den Start eines Agile Projekts oder Arbeit mit einem Agile Team

Ja/nein

Wurden ein Auftraggeber und ein Budget für das Projekt oder Team festgelegt?

Verfügt das Team über das Fachwissen, das zur Bewältigung des Projekts erforderlich ist ohne auf andere Teams oder Einzelpersonen angewiesen zu sein?

Wurden alle Beteiligten in der Agile-Methode ausgebildet, die zum Einsatz kommen soll?

Kann das Team selbstorganisierend arbeiten?

Ist ein Raum vorhanden, in dem das Team "face to face" zusammenarbeiten kann?

Wurden deutliche Vereinbarungen bzgl. der Vorgehensweise getroffen? (Beispielsweise welche Agile-Methode angewendet werden soll, erforderliche Meetings, Rollen, Formulierung von Backlog Items, Hilfsmittel, Definition of Done, Teamwerte etc.)

Wurde deutlich formuliert, welchen Mehrwert das Team schaffen wird?

Wurde unter Einbeziehung des Inputs von Stakeholdern und der Developer ein erstes Backlog erstellt?

Sind die Backlog Items verständlich, klein genug, validiert, testbar und nach ihrem Mehrwert priorisiert?

Die Formulierung der Vision

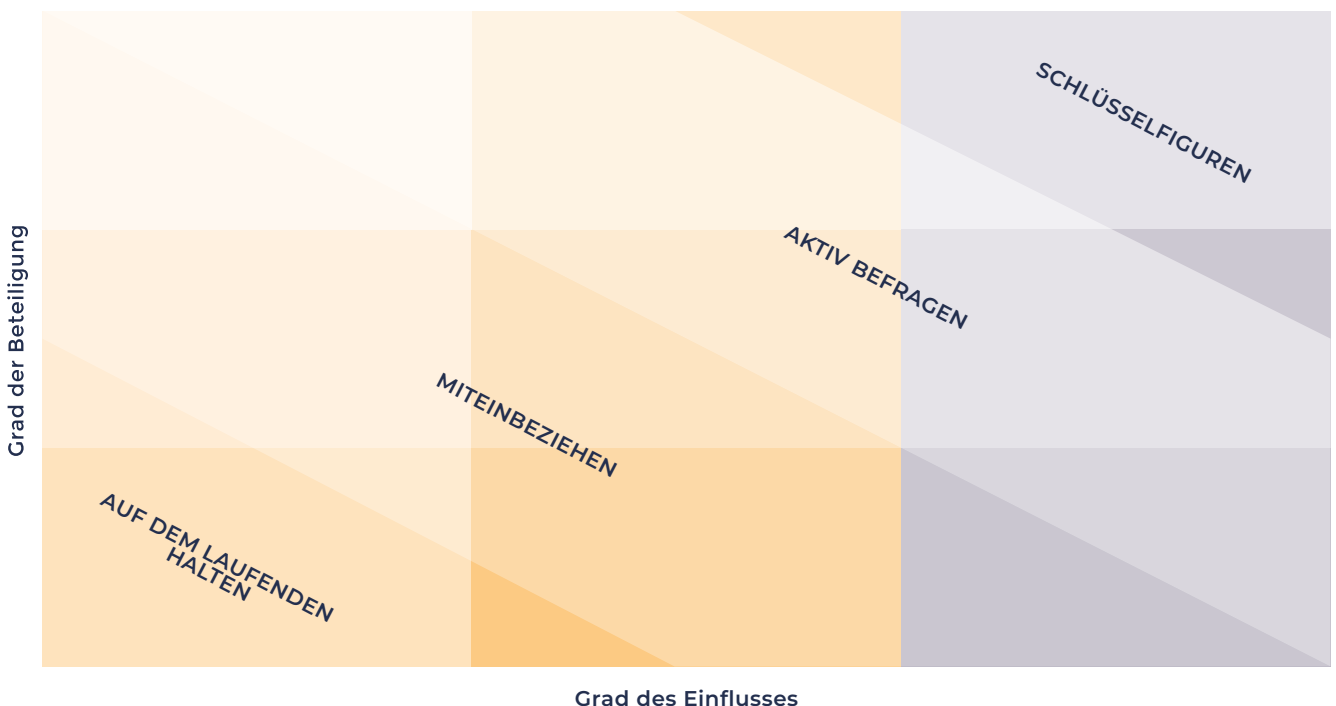
Sowohl in Projektteams als auch in operationellen Teams ist wichtig, dass alle Teammitglieder das übergeordnete Ziel vor Augen haben. Eine deutliche und sinngebende Vision führt zu intrinsischer Motivation. Hierdurch werden bessere Arbeitsergebnisse erzielt und sind die Teammitglieder zufriedener. Außerdem sorgt eine deutliche Vision für Fokus und erleichtert die Selbstorganisation.

VISION Mit welchem Ziel wird das Produkt entwickelt? Welche positiven Veränderungen soll es herbeiführen?			
ZIELGRUPPE Für welchen Markt ist das Produkt bestimmt? Wer sind die Kunden und Benutzer?	WÜNSCHE DER ZIELGRUPPE Welches Problem löst das Produkt? Welchen Vorteil liefert das Produkt?	PRODUKT Aus welchen Teilen besteht das Produkt? Was macht es einzigartig? Ist die aktuelle Organisation in der Lage, das Produkt anzufertigen?	ZIELE Wie wird das Arbeitsergebnis für die Organisation von Nutzen sein?
WETTBEWERB Wer sind die größten Wettbewerber? Welches sind ihre Stärken und Schwächen?	EINNAHMEN Wie wird das Produkt auf den Markt gebracht und lassen sich Einnahmen erzielen?	COSTS Welche Kosten entstehen, um: 1. Das Produkt zu entwickeln 2. Das Produkt zu verkaufen 3. Das Produkt instand zu halten	VERKAUFSKANÄLE Wo wird das Arbeitsergebnis vermarktet? Gibt es bereits entsprechende Kanäle?

Das Dokument, in dem die Vision kommuniziert wird, besteht aus den folgenden Elementen::

- Die Vision, formuliert in einem einzigen Satz: Erklären Sie, welche positiven Auswirkungen das Team haben wird. Mit anderen Worten: Was ist das Ziel der Zusammenarbeit im Team?
- Die Zielgruppe: Für wen wird das Projekt durchgeführt?
- Bedürfnisse der Zielgruppe: Welchen Bedürfnissen der Zielgruppe wird das Team gerecht? Löst es ein bestehendes Problem?
- Das Produkt: Welches sollen die drei bis fünf einzigartigen Merkmale des Arbeitsergebnisses des Teams sein?
- Ziele: Wie wird das Arbeitsergebnis für die Organisation von Nutzen sein?
- Wettbewerb: Wer sind die Wettbewerber und welches sind ihre Stärken und Schwächen?
- Einnahmen: Wie wird das Produkt auf den Markt gebracht und wie wird es Einnahmen generieren?
- Kosten: Auf welche Höhe werden die Kosten ungefähr geschätzt, die mit dem angestrebten Ergebnis verbunden sind?
- Verkaufskanäle: Wo wird das Arbeitsergebnis vermarktet? Gibt es bereits entsprechende Kanäle?

Die Vision kann vom Product Owner, dem gesamten Team oder von Stakeholdern formuliert werden. Die besten und am meisten unterstützten Visionen werden von möglichst vielen Stakeholdern so früh wie möglich entwickelt. Anhand der untenstehenden Matrix lässt sich entscheiden, wie spezifische Stakeholder in das Projekt oder Team miteinbezogen werden sollten.



Das Erstellen des ersten Product Backlogs

Für agile Teams ist es wichtig, in einem ersten Schritt das erste Backlog zu erstellen. Bei Scrum handelt es sich um einen Product Backlog. Bei Kanban um eine Liste mit Items. Sie können sowohl

projektbasiert als auch operationell sein. Die Vision stellt den Ausgangspunkt und wichtigsten Rahmen für die Erstellung des ersten Product Backlogs dar. Ausgehend von der Vision liefern wichtige Stakeholder sowie das agile Team Input und entwickeln so das erste Product Backlog. Es enthält genügend Elemente, damit das agile Team mit der Lieferung von Kundennutzen beginnen kann. Nach dem Scrum Framework obliegt die Verantwortung für die Erstellung und Pflege des Product Backlog dem Product Owner. Bei Kanban einigt das Team sich intern darauf, wer diese Aufgaben übernimmt.

SCRUM

Scrum wurde als Framework entworfen, in dessen Rahmen sich komplexe adaptive Probleme produktiv und kreativ lösen lassen. Mit Scrum werden Produkte entwickelt oder Projekte abgeschlossen, die Kunden einen maximalen Mehrwert liefern.

Eine vollumfassende Beschreibung von Scrum ist im Leitfaden "The Scrum Guide" zu finden. Dieses Dokument erläutert die Werte, Rollen, Besprechungen und Artefakte von Scrum vor dem Hintergrund der Softwareentwicklung. Im vorliegenden "Body of Knowledge" beschreiben wir Scrum mit Blick auf Teams, die außerhalb der IT arbeiten.

Empirismus

Scrum basiert auf Empirismus. Empirismus besagt, dass (i) Wissen aus Erfahrung entsteht und (ii) Entscheidungen auf Grundlage dessen getroffen werden müssen, was man weiß. Scrum implementiert Empirismus durch **Transparenz, Überprüfung und Anpassung**.

Transparenz wird in Scrum so erreicht:

1. Immer mit denselben Rollen, Meetings und Artefakten arbeiten. Hierdurch sprechen alle dieselbe Sprache und man versteht einander;
2. Immer den aktuellen Stand der Arbeit auf dem Product und Sprint Backlog sichtbar machen;
3. Immer mit einer gemeinsamen Definition of Done arbeiten, damit für alle Mitglieder des Scrum-Teams klar ist, wann ein Arbeitsschritt beendet ist.

Überprüfung und Anpassung werden in Scrum durch die folgenden Verhaltensweisen gewährleistet:

1. Im Daily Scrum Meeting den Sprint Backlog überprüfen und wo nötig anpassen, um sicherzustellen dass sich das Team auch wirklich dem Sprint-Ziel nähert;
2. Während des Review Meetings das Inkrement (die fertiggestellte Arbeit) überprüfen und falls nötig das Product Backlog ändern, damit ein maximaler Kundenwert gewährleistet ist;
3. Im Verlauf der Retrospective den Prozess, die Beziehungen und (Arbeits-)Mittel überprüfen und falls nötig verändern.

Werte

Scrum liegen fünf Werte zugrunde:

- **Commitment** (Einsatz): Das Team zeigt vollen Einsatz, um seine Vision zu realisieren.
- **Courage** (Mut): Das Team ist mutig genug, um an schwierigen Aufgaben zu arbeiten. Es traut sich auch, Fehler zu machen.

- **Focus:** Zu bestimmten Zeitpunkten arbeiten alle konzentriert an der jeweiligen Aufgabe, um das Sprint-Ziel zu erreichen.
- **Openness** (Offenheit): Sowohl die Stakeholder als auch das Scrum-Team kommunizieren offen über die zu erledigende Arbeit und Herausforderungen, die sich dabei ergeben.
- **Respect:** Man erkennt einander als fähige und selbständig funktionierende Teammitglieder an.

Rollen

Ein Scrum-Team setzt sich aus einem Product Owner, einem Scrum Master und den Developern zusammen. Bei einem Scrum-Team handelt es sich um ein selbstorganisiertes Team. Im Idealfall ist es multidisziplinär.

Product Owner

Das Ziel der Product Owner Rolle besteht darin, den Wert des Arbeitsergebnisses des Scrum-Teams zu optimieren. Jedes Projekt hat einen einzigen Product Owner. Der Product Owner hat das volle Mandat, im Rahmen des Projektes die Prioritäten für zu erbringende Arbeiten festzulegen.

Der Product Owner ist für die Verwaltung des Product Backlog zuständig. Der Product Owner kann bestimmte Aufgaben den Developern übertragen, bleibt aber selbst verantwortlich.

Aufgaben des Product Owners:

- Deutliches Beschreiben von Product Backlog Items;
- Vergeben von Prioritäten für Items im Product Backlog;
- Sicherstellen, dass das Product Backlog sichtbar, transparent und für alle eindeutig ist;
- Gewährleisten, dass die Developer die Product Backlog Items verstehen.

Developer

Die Developer sind für die Lieferung aller für den Sprint ausgewählten Product Backlog Items verantwortlich.

Die Developer:

- Arbeiten autonom und sind selbstorganisiert;
- Schätzen die Product Backlog Items und bestimmen selbst, wie viel Aufwand mit bestimmten Arbeiten verbunden ist;
- Legen fest, wie viel Arbeit für den nächsten Sprint ausgewählt werden soll;
- Sind multidisziplinär und verfügen über alle Fachkenntnisse, die für einen erfolgreichen Abschluss des Projektes notwendig sind;
- Ordnen Mitgliedern keine spezifischen Funktionstitel zu und haben auch keine Unterteams;
- Sind immer gemeinsam für das Ergebnis verantwortlich;
- Bestehen aus drei bis neun Mitgliedern. Der Product Owner und der Scrum Master werden nur dann mitgezählt, wenn sie auch selbst Arbeit aus dem Sprint Backlog erledigen.

Scrum Master

Der Scrum Master sorgt dafür, dass das gesamte Team mit der Theorie, üblichen Events, Regeln und Werten der Scrum-Methode vertraut ist. Der Scrum Master steht dem Scrum-Team unterstützend zur Seite. Außerdem stellt er sicher, dass Stakeholder und andere Personen außerhalb des

Scrum-Teams Scrum verstehen. Er ist um eine optimale Zusammenarbeit innerhalb des Scrum-Teams bemüht. Ziel ist, dass das vom Team gelieferte Arbeitsergebnis den maximalen Kundennutzen darstellt.

Der Scrum Master unterstützt den Product Owner bei Folgendem:

- Sicherstellen, dass die Projektziele und der Projektumfang dem Scrum-Team bekannt sind;
- Effektive Pflege des Product Backlogs;
- Dem Scrum-Team die Wichtigkeit von klaren und gut geschriebenen Product Backlog Items vermitteln;
- Förderung des Verständnisses von Produkt- und Projektplanung in einem empirischen Umfeld;
- Festlegen von Prioritäten für den Product Backlog;
- Förderung des Verständnisses von Agilität (Agile);
- Im Bedarfsfall Unterstützung der Moderation von Meetings

Der Scrum Master hilft den Developern:

- Als multidisziplinäres und selbstorganisierendes Team zu arbeiten;
- Bei der Beseitigung von Hindernissen;
- Bei der Arbeit in einer Organisation, in der Scrum (noch) nicht vollumfänglich verstanden wird;
- Im Bedarfsfall Unterstützung bei der Moderation von Meetings

Der Scrum Master hilft der Organisation, indem er:

- Der Organisation das Verinnerlichen von Scrum erleichtert;
- Mitarbeitern und anderen Stakeholdern beibringt, was Scrum und empirische Produktentwicklung bedeuten;
- Verbesserungen in der Organisation realisiert, durch die die Produktivität des Scrum-Teams zunimmt;
- Mit anderen Scrum Mastern zusammenarbeitet, um die Effektivität von Scrum in der Organisation zu steigern.

Der Sprint

Als Sprint wird eine Periode mit fester Dauer bezeichnet, in der ein fertiger und potentiell nutzbarer Teil des Projektes umgesetzt wird. (Der Zeitraum beträgt mindestens eine Woche und maximal einen Monat.) Jedes Projekt hat mehrere Sprints.

Ein Sprint:

- Beginnt unmittelbar nach Abschluss des vorherigen Sprints;
- Dauert in einem Projekt immer gleich lang;
- Besteht aus Sprint Planning, Daily Scrums, Projektarbeit, der Sprint Review und der Sprint Retrospective;
- Hat immer ein genau definiertes Sprint-Ziel.

Während eines Sprints:

- Können bzgl. der geplanten Arbeiten keine Änderungen mehr vorgenommen werden, die das Sprint-Ziel in Gefahr bringen könnten;
- Hat die Qualität des Arbeitsergebnisses immer oberste Priorität;

- Können der Projektumfang bzw. Projekthinhalte in Abstimmung mit dem Product Owner präzisiert und verändert werden.

Länger als einen Monat dauernde Sprints sollten vermieden werden. Bei einer zu großen Dauer werden Sprints zu komplex und bergen mehr Risiken. Eindeutige Sprints sorgen für Planbarkeit. Mithilfe regelmäßiger Überprüfung und ggf. Anpassung im Daily Scrum Meeting wird das Sprint-Ziel erreicht. Außerdem lassen sich finanzielle Risiken begrenzen, indem Produkte bzw. Projekte in nutzbaren kleineren Teilen fertiggestellt werden.

Der Product Owner ist der einzige, der einen Sprint vorzeitig beenden kann. Diesen Entschluss kann der Product Owner fassen, wenn das Sprint-Ziel überflüssig geworden ist. Aufgrund des kurzzyklischen Charakters von Scrum kommt es nur selten dazu, dass ein Sprint abgebrochen wird. Geschieht dies doch, werden die umgesetzten Backlog Items bewertet. Fertiggestellte (Teil-)Produkte können trotzdem in Gebrauch genommen werden, wenn der Product Owner das möchte. Noch nicht abgeschlossene Backlog Items werden wieder im Product Backlog platziert. Der Abbruch eines Sprints ist aufgrund der verlorenen Zeit sowie des Mehraufwands in Zusammenhang mit der erforderlichen Koordination nach dem Abbruch eine kostspielige Angelegenheit.

Sprint Planning

Jeder Sprint beginnt mit einem Sprint Planning Meeting. Im Sprint Planning plant das Scrum-Team die im Sprint zu erledigende Arbeit. Der Scrum Master ist dafür zuständig dafür zu sorgen, dass eine entsprechende Besprechung stattfindet und alle Teilnehmer die Ziele verstehen.

Ein Sprint Planning:

- Dauert maximal acht Stunden (Time-Box) für einen einmonatigen Sprint;
- Beantwortet die Frage "Was können wir in diesem Sprint fertig bekommen?";
- Beantwortet die Frage "Wie gehen wir die zu bewältigende Arbeit an?";

Im Sprint Planning gibt es zwei Themen:

- Thema 1: "Was können wir in diesem Sprint fertig bekommen?"
 - o Der Product Owner präsentiert das Ziel des Sprints und die Product Backlog Items, die dazu beitragen sollen, dass dieses Ziel erreicht wird.
 - o Die Developer schätzen ab, wie viel Arbeit sie während des Sprints schaffen können. Das tun sie auf Basis verfügbarer Kapazitäten und anhand von Erfahrungswerten aus der Vergangenheit. Die Developer entscheiden schlussendlich, wie viel Arbeit eingeplant werden soll.
 - o Das ganze Scrum-Team arbeitet im Print Planning zusammen. Das stellt sicher, dass alle den Arbeitsinhalt des Sprints auch wirklich verstehen.
- Thema 2: "Wie gehen wir die zu bewältigende Arbeit an?"
 - o Nachdem das Sprint-Ziel festgelegt und die Product Backlog Items ausgewählt wurden, überlegen die Developer, wie sie die Items im Sprint vollständig umsetzen können.
 - o Die Kombination aus ausgewählten Product Backlog Items und dem Plan für ihre Umsetzung wird als Sprint Backlog bezeichnet.

- o Im Sprint Planning Meeting braucht nicht jedes Product Backlog Item bis ins letzte Detail ausgearbeitet zu werden. Aber wenigstens die Items, mit denen sich das Team in den ersten Tagen beschäftigen möchte, sollten in eintägige oder noch kürzere Aufgaben zerlegt werden.
- o Die Developer sind für das Sprint Backlog verantwortlich.
- o Wenn das Team zu viel oder zu wenig Arbeit aufgenommen hat, besteht die Möglichkeit, Product Backlog Items in Abstimmung mit dem Product Owner zu verändern.
- o Die Developer können Fachleute einladen, die ihnen dabei helfen, einen Plan für den Sprint zu erstellen.

Das Sprint-Ziel:

Als Sprint-Ziel bezeichnet man den Zustand, der durch den jeweiligen Sprint erreicht werden soll. Das Sprint-Ziel wird durch vollständige Umsetzung der hierfür ausgewählten Product Backlog Items realisiert. Es gibt den Developern eine Richtung vor und sorgt für Fokus. Mit Blick auf die Erledigung der Arbeiten behält das Team aber eine gewisse Flexibilität. Solange es nur Arbeitsergebnisse liefert, die zum Sprint-Ziel passen.

Daily Scrum

Das Daily Scrum ist eine tägliche Besprechung der Developern. Jedes Daily Scrum dauert maximal 15 Minuten (Time-Box). Es wird an jedem Tag abgehalten, an dem die Developer am Projekt arbeiten. Der Scrum Master bringt dem Team bei, sich an 15 Minuten als Besprechungsdauer zu halten. Im Daily Scrum plant das Team die Arbeitsschritte für die nächsten 24 Stunden. Es wird auch besprochen, welche Arbeiten seit dem letzten Daily Scrum erledigt wurden. So optimieren die Developer ihre Zusammenarbeit und daraus resultierende Arbeitsergebnisse.

Das Daily Scrum:

- Wird täglich zur selben Uhrzeit am selben Ort abgehalten;
- Dient der Beschäftigung mit dem Sprint Backlog und dem im Hinblick auf das Sprint-Ziel realisierten Fortschritt;
- Maximiert die Wahrscheinlichkeit, dass die Developer das Sprint-Ziel erreichen;
- Hat keine fest vorgeschriebene Form und kann von den Developern selbst gestaltet werden, solange nur der Fortschritt in Richtung Sprint-Ziel im Mittelpunkt steht;
- Muss stattfinden. Dafür ist der Scrum Master verantwortlich. Er braucht aber nicht unbedingt selbst an der Besprechung teilzunehmen. Die Developer aber sehr wohl. Alle anderen Teilnehmer haben vor allem darauf zu achten, dass sie das Meeting nicht stören;
- Optimiert die Kommunikation und macht andere Besprechungen überflüssig. Das Daily Scrum Meeting dient der Identifikation von Hindernissen, wirkt sich positiv auf eine schnelle Entscheidungsfindung aus und sorgt für einen verbesserten Wissensstand der Developer.

Review

Sprint Reviews werden am Ende des Sprints abgehalten, um das Arbeitsergebnis zu überprüfen und das Product Backlog anzupassen, wenn neue Erkenntnisse dies notwendig erscheinen lassen. Während des Sprint Reviews arbeiten Stakeholder und Scrum-Team zusammen. Die Developer präsentieren die fertiggestellten Product Backlog Items. Anschließend präsentiert der Product Owner den aktuellen Stand des Product Backlog. Gemeinsam mit den Stakeholdern

wird festgelegt, welches die wertvollsten nächsten Schritte für das Projekt bzw. angestrebte Produkt sind. Der Product Owner teilt mit, für wann Stakeholder die Umsetzung erwarten können. Das Sprint Review Meeting ist informell und mehr als nur ein Status Update. Es soll für gegenseitiges Feedback sorgen und eine gute Zusammenarbeit zwischen Scrum-Team und Stakeholdern sicherstellen.

Das Sprint Review:

- Dauert maximal vier Stunden (Time-Box) für einen einmonatigen Sprint (4 Wochen);
- Muss stattfinden. Die Verantwortung dafür obliegt dem Scrum Master. Der Scrum Master sorgt auch dafür, dass alle Teilnehmer das Ziel des Reviews verstehen und die maximale Besprechungsdauer nicht überschritten wird.
- Ist das einzige Meeting, an dem sowohl die Stakeholder als auch das komplette Scrum-Team teilzunehmen haben;
- Ist beendet, sobald die Teilnehmer sich über die Prioritäten der Product Backlog Items im nächsten Sprint einig sind.

Retrospective

In der Retrospective hat das Scrum-Team Gelegenheit, sich selbst zu reflektieren und bei Bedarf Veränderungen anzustoßen. Das Ziel besteht darin, (i) zu kontrollieren, wie der letzte Sprint im Hinblick auf Zusammenarbeit, Beziehungen, Prozesse und Hilfsmittel verlaufen ist, (ii) die größten Erfolge, aber auch Verbesserungspotential zu identifizieren und (iii) einen Plan zur Umsetzung dieser Verbesserungen aufzustellen. Während der Sprint Retrospective kann das Scrum-Team die Definition of Done überdenken und bei Bedarf ändern.

Der Scrum Master stellt sicher, dass die Besprechung positiv und produktiv verläuft. Als Mitglied des Scrum-Teams nimmt auch der Scrum Master selbst am Meeting teil und achtet darauf, dass es keine Zeitüberschreitung gibt.

Die Retrospective:

- Findet nach dem Sprint Review und vor dem Sprint Planning für den nächsten Sprint statt;
- Dauert bei einem einmonatigen Sprint maximal drei Stunden;
- Muss abgehalten werden, und dafür ist der Scrum Master zuständig;
- Der Scrum Master stellt sicher, dass alle Teilnehmer das Ziel der Retrospective verstehen und die maximale Besprechungsdauer nicht überschritten wird.

Natürlich darf das Scrum-Team jederzeit Verbesserungen realisieren. Aber spätestens die Sprint Retrospective schafft für die Teilnehmer Raum, sich auf das Überprüfen und Durchführen von Änderungen zu konzentrieren.

Product Backlog Refinement

Im Verlauf des gesamten Sprints wird der Product Backlog kontinuierlich aktualisiert und weiter ausgearbeitet. Dafür ist der Product Owner verantwortlich. Aber der Product Owner braucht diese Aufgabe nicht alleine zu bewältigen.

Product Backlog Refinement beinhaltet das (i) weiter Detaillieren und Zerteilen von Product Backlog Items, (ii) Vergeben von Prioritäten für Product Backlog Items bzw. (iii) die Bitte an die

Developer, eine Einschätzung bzgl. Product Backlog Items abzugeben.

Die gängige Faustregel lautet: Für diesen Schritt wird maximal 10% der gesamten Sprintdauer reserviert. Es ist nicht unbedingt erforderlich, dass das komplette Scrum-Team an Besprechungen zum Product Backlog Refinement teilnimmt. Wann genau das Refinement stattfinden soll, kann das Scrum-Team selbst entscheiden.

Artefakte

Zu Scrum gehören verschiedene Artefakte. Sie sollen die Transparenz wesentlicher Informationen maximieren. Transparente Informationen ermöglichen dem Team das Überprüfen und Anpassen.

Das Product Backlog

Das Product Backlog ist eine nach Prioritäten geordnete Liste aller Ideen, die im Projekt umgesetzt werden können. Der Product Backlog ist die einzige Quelle derer sich das Scrum-Team bedient, um Arbeit auszuwählen und anzugehen. Für das Product Backlog ist der Product Owner verantwortlich. Das bedeutet, dass der Product Owner den Inhalt pflegt, dafür sorgt, dass Prioritäten festgelegt werden und die Liste zugänglich ist. Ein Product Backlog ist nie ganz fertig und wird ständig weiterentwickelt. Dabei fließen neue Erkenntnisse und Veränderungen des Umfelds mit ein.

Items im Product Backlog (i) verfügen über eine Beschreibung, (ii) sind nach Priorität sortiert, (iii) mit einer Einschätzung des mit ihnen verbundenen "Aufwands" versehen und (iv) haben einen Wert. Außerdem gibt es für Product Backlog Items i.d.R. verschiedene Akzeptanzkriterien. Dabei handelt es sich um spezifische Bedingungen, die ein Item erfüllen muss, bevor es als fertiggestellt, also als "done" angesehen werden kann. Product Backlog Items, die höher im Product Backlog stehen, also eine höhere Priorität haben, sind weiter ausgearbeitet als Items weiter unten auf der Liste. Die Items ganz oben auf der Liste sind so ausgearbeitet, dass sie im Prinzip im nächsten Sprint angegangen und als "done" ausgeliefert werden können. Ein Item, das darauf wartet bearbeitet zu werden, wird als "ready" bezeichnet bzw. als "fertig für den Sprint".

Das Sprint Backlog

Das Sprint Backlog besteht aus:

- Den für den aktuellen Sprint ausgewählten Product Backlog Items;
- Einem Arbeitsplan anhand dessen sich die Items umsetzen lassen, um das Sprint-Ziel zu erreichen.

Das Sprint Backlog:

- Macht alle Arbeiten deutlich, die zu erledigen sind, damit das Sprint-Ziel erreicht wird;
- Enthält mindestens eine Verbesserungsmaßnahme, die in der vorherigen Retrospective beschlossen wurde;
- Kann nur von den Developern verändert werden. Das geschieht, sobald sich im Verlauf des Sprints neue Erkenntnisse ergeben, die zu Mehr- oder Wenigerarbeit führen;
- Ist eine Übersicht der Arbeit der Developer im Sprint in Echtzeit.

Inkrement

Als Inkrement wird die Summe aller im Rahmen eines Sprints umgesetzten Product Backlog Items ergänzt um den Wert aller Inkrements vorangegangener Sprints bezeichnet. Jedes Inkrement bringt das Team dem angestrebten Ziel näher. Das Inkrement bzw. die fertiggestellten Product Backlog Items müssen immer der Definition of Done entsprechen. Die Entscheidung, ob ein Inkrement tatsächlich in Gebrauch genommen wird, obliegt dem Product Owner.

Definition of Done

Wenn ein Product Backlog Item oder Inkrement als komplett fertig ("done") gekennzeichnet wird, muss allen Beteiligten klar sein, was genau "done" bedeutet. Die Definition of Done ist eine Liste mit deutlichen Kriterien, die jedes ausgelieferte Item zu erfüllen hat.

Die Definition of Done:

- Ist für jedes Scrum-Team eine andere;
- Hilft den Entwicklern im Sprint Planning Meeting beim Bestimmen der Arbeitsmenge;
- Kann innerhalb einer Organisation (teilweise) in Standards, Richtlinien oder Konventionen festgelegt sein. Alternativ hat eine Organisation auch die Möglichkeit, eine minimale Definition of Done vorzugeben.
- Kann vom Scrum-Team selbst festgelegt werden, wenn die Organisation nichts vorschreibt;
- Ist für mehrere Scrum-Teams, die am selben Projekt arbeiten, gleich.

Fortschritt überwachen

Der Product Owner ist dafür verantwortlich, im Sprint Review Meeting zu bestimmen, wie viel Arbeit noch verbleibt, bis die Vision erreicht ist. Der Product Owner erkennt den Fortschritt, indem er die aktuell verbleibende Arbeit mit der Arbeitsmenge vergleicht, die zum Zeitpunkt des vorangegangenen Reviews noch zu erledigen war. Diese Information wird den Stakeholdern zugänglich gemacht. Hierfür verwendet der Product Owner ein Burn-Down- oder Burn-Up-Chart.

Der Product Owner überwacht die verbleibende Arbeitsmenge über mehrere Sprints hinweg. Während der Sprints tun dies die Developer selbst. Um einen Anhaltspunkt dafür zu erhalten, ob das Team im Hinblick auf das Sprint-Ziel im Zeitplan liegt, wird die noch im Sprint Backlog verbleibende Arbeit aufaddiert. Das machen die Developer mindestens zu jedem Daily Scrum Meeting.

KANBAN

Scrum ist eine ausgezeichnete Methode für die Durchführung von Projekten oder (Weiter-) Entwicklung von Produkten. Für das Organisieren operativer Arbeiten eignet Scrum sich eher wenig. Die Kanban-Methode hingegen lässt sich sowohl auf die Projektarbeit als auch auf operative Arbeiten anwenden.

Kanban zielt darauf ab, Wissensarbeit (Prozesse oder Projekte) zu definieren, managen und verbessern. Beispiele sind Dienstleistungen wie die Abwicklung von Kundenbeschwerden, Softwareentwicklung oder Entwicklung von Inhalten für eine Website. Kanban basiert auf sechs

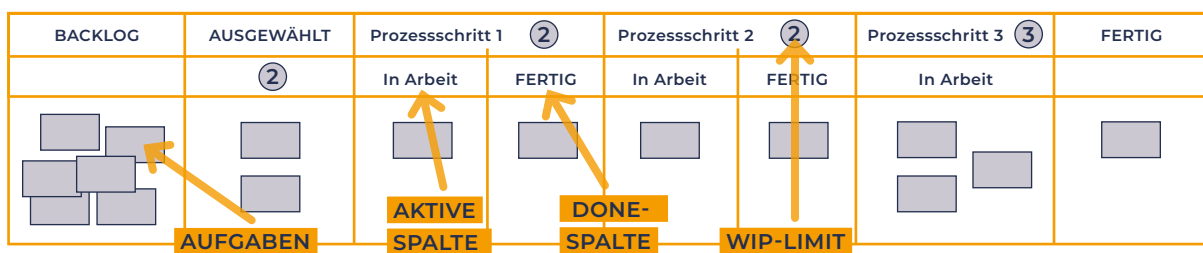
Praktiken, die im Folgenden näher erläutert werden.

Praktik 1: Arbeitsfluss visualisieren

Visualisierung steht bei Kanban zentral. Der Einsatz der Kanban-Methode beginnt mit einer Visualisierung des Arbeitsprozesses ("Workflow"). Wichtig ist, dass das Team die Visualisierung des Arbeitsprozesses gemeinsam angeht. Denn es muss Einigkeit darüber bestehen, wie die Arbeit erledigt werden soll.

In Kanban:

- Wird der Arbeitsprozess (digital oder auf Papier) auf einem Kanban-Board visualisiert. Ein Kanban-Board hat für jeden Prozessschritt eine "aktive" Spalte für Items, an denen noch gearbeitet wird sowie eine Spalte für Items, die bereits "done" sind.
- Werden für das Team getroffene Arbeitsvereinbarungen und Regeln dargestellt.
- Trifft das Team Vereinbarungen darüber, welche Informationen für ein Backlog Item benötigt werden. Wie bei Scrum auch hat jedes Item eine Beschreibung, eine Aufwandsschätzung und einen Wert.



Praktik 2: Menge angefangener Arbeit limitieren

Die Items bzw. Aufgaben, an denen das Team aktuell arbeitet, werden als "Work in Progress", kurz "WIP" bezeichnet. Es liegt in der Natur von Teams, neue Arbeit so schnell wie möglich anzugehen. Sobald sich neue Aufgaben auftun, bitten Manager und Projektleiter ihre Teams, sie direkt zu erledigen. Das ist ein Push-Modell, bei dem Arbeit sozusagen von außen nach innen gedrückt wird. Die Folge: Arbeitsprozesse laufen über. Ein Team ist mit so vielen Dingen gleichzeitig beschäftigt, dass die Durchlaufzeit der einzelnen Items stark zunimmt. Die Teammitglieder bekommen das Gefühl, die Kontrolle über ihre Arbeit verloren zu haben. Konkrete Arbeitsergebnisse bleiben hinter den Erwartungen zurück.

In Kanban:

- Wird mit einem *Pull-Modell* gearbeitet. Teammitglieder ziehen neue Arbeit in den Prozess hinein, sobald es Platz dafür gibt.
- Wird anhand eines WIP-Limits festgelegt, wie viele Items sich maximal in einem einzigen Prozessschritt befinden dürfen. Das WIP-Limit gilt je Prozessschritt für die aktive und die "done"-Spalte zusammengenommen.
- Wird das WIP-Limit für jeden Prozessschritt visualisiert.
- Wird eine Faustregel zur Abschätzung des WIP-Limits angewendet. Die Regel lautet wie folgt: "Zahl am Arbeitsschritt beteiligter Personen + 50%". Wenn drei Teammitglieder an

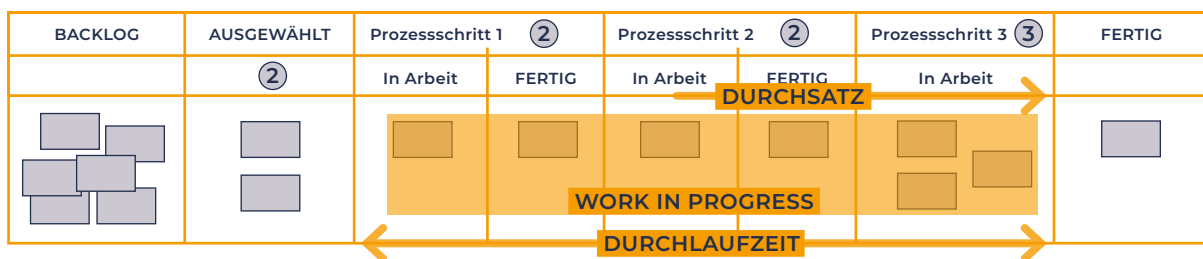
einem Prozessschritt arbeiten, dürfen für diesen Prozessschritt 4,5 Items eingetragen sein. Nach oben aufgerundet also fünf Aufgaben. Diese Formel formt den Ausgangspunkt und wird verändert, sobald neue Erkenntnisse dies nötig erscheinen lassen.

Praktik 3: Arbeitsfluss managen

Die Kanban-Methode optimiert den Arbeitsfluss. Von einem guten Fluss ist erst dann die Rede, wenn die Items auf dem Kanban-Board kaum noch zum Stillstand kommen. Wie gut der Fluss ist, lässt sich anhand der Durchlauf- oder Lieferzeit von Items ermitteln.

Im Zusammenhang mit dem Arbeitsfluss gängige Begriffe:

- **Durchlaufzeit:** zur Umsetzung eines Items benötigte Zeit (Zyklusdauer)
- **Work in Progress:** die Summe aller Arbeiten (gemessen in Items, Stunden, Story Points usw.) die derzeit im Prozess sind
- **Durchsatz:** Die Arbeitsmenge (gemessen in Anzahl der Items, Stunden oder Story Points), die ein Team je Zeiteinheit fertigstellt



Work in Progress, Durchlaufzeit und Durchsatz stehen in einer Abhängigkeit zueinander. Das Gesetz von Little besagt: Wenn zwei Variablen bekannt sind, lässt sich auch die dritte abschätzen.

Little's Law (Gesetz von Little): **Durchlaufzeit = Work in Progress / Durchsatz**

Beispielprozess: Bezahlvorgang an der Supermarktkasse. Das Bezahlen dauert 30 Sekunden. Dann beläuft sich der Durchsatz auf 2 Kunden in der Minute. Stehen sechs Personen an der Kasse an, beträgt der Work in Progress 6. Anhand des Gesetzes von Little lässt sich die Durchlaufzeit wie folgt abschätzen: $(\text{Work in Progress} / \text{Durchsatz}) = (6/2) = 3$ Minuten. Kanban-Teams, die ihren Arbeitsfluss im Griff haben, können ihren Stakeholdern gegenüber verlässliche Aussagen bzgl. der Durchlaufzeiten treffen.

Praktik 4: Vereinbarungen explizit machen

Indem sie Arbeitsvereinbarungen transparent machen, verbessern Kanban-Teams die Zusammenarbeit und steigern ihre Produktivität. Typische Vereinbarungen betreffen beispielsweise WIP-Limits, die Zahl an jedem Prozessschritt beteiligter Personen, die Unterstützung anderer Teammitglieder, wenn Zeit übrig ist, die Definition of Done oder das Angehen neuer Arbeit.

Des Weiteren gilt:

- Ein paar (nicht zu viele) wichtige Vereinbarungen treffen, damit sich das Team auf Absprachen konzentriert, die wirklich wichtig sind;
- Vereinbarungen auf dem Kanban-Board sichtbar machen;
- Vereinbarungen wird immer nachgekommen. Als selbstorganisiertes Team behalten die Teammitglieder einander diesbezüglich im Auge.
- Falls nötig, passt das Kanban-Team Arbeitsvereinbarungen an.

Praktik 5: Feedbackschleifen einbauen

Die Kanban-Methode ist darauf ausgelegt, dass sich das Team iterativ verbessert. Hierfür werden tägliche, wöchentliche und monatliche Feedbackschleifen eingebaut. Verbesserungen sind ein fester Themenpunkt in regelmäßig stattfindenden Besprechungen. Das Kanban-Meeting ist Pflicht. Weitere Besprechungen können je nach Bedarf zusätzlich eingeplant werden.

Beispiele von Meetings:

- **Kanban Meeting:** für die tägliche Koordination, ermöglicht die Selbstorganisation. Diese Besprechung ähnelt dem Daily Scrum Meeting. Es wird aber lediglich die Frage gestellt, ob ein Teammitglied durch etwas behindert oder ausgebremst wird. Das Meeting dauert rund fünf Minuten.
- **Strategie Review:** um zu besprechen, wie sich die Außenwelt im Verhältnis zu(m) Prozess(en) verändert. Je nach Ergebnis können angebotene Dienstleistungen überdacht werden.
- **Retrospective Meeting:** dasselbe wie bei Scrum
- **Service Delivery Review:** zum Beurteilen und Verbessern der Effektivität der Dienstleistung
- **Replenishment Meeting:** In diesen Meetings nimmt das Team neue Items in das Backlog auf. Die Besprechung ähnelt dem Sprint Planning bei Scrum.
- **Delivery Planning:** Das Überwachen und Planen größerer Fertigstellungen oder Meilensteine

Praktik 6: Gemeinsam in kleinen Schritten besser werden

Veränderungsprozesse beginnen i.d.R. mit einem deutlichen Bild vom Zielpunkt und konsequentem darauf Hinarbeiten. Bei Kanban geht es nicht um langfristige Planungen. Kanban beginnt im Hier und Jetzt. Als Startpunkt bedient sich die Methode dem Wissen und der Erfahrung, die über die Jahre hinweg erworben wurden.

Bei Kanban gilt folgendes:

- Das Team ist um kontinuierliche Verbesserung in kleinen Schritten bemüht;
- Veränderung ist ein ständiger Prozess, der nie aufhört;
- Verbesserungen werden von unten nach oben ("Bottom-up") realisiert. Die Verantwortung dafür liegt beim selbstorganisiertem Team;
- Kleine Experimente zum Verbessern hält das Team auf dem Kanban-Board fest. So werden und bleiben Bemühungen um Verbesserung transparent.

LITERATURLISTE

Anderson, D. J., & Carmichael, A. (2016). *Essential Kanban Condensed*. Blue Hole Press.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Kern, J. (2001). *Manifesto for agile software development*.

Pichler, R. (2010). *Agile product management with scrum: Creating products that customers love*. Addison-Wesley Professional.

Sutherland, J., & Schwaber, K. (2013). *The scrum guide. The definitive guide to scrum: The rules of the game*. Scrum.org, 268.

