**Eine Übersicht**

**Scrum – Kanban**

**1 Scrum**

Scrum ist eine agile Alternative zu den klassischen Projektmanagement-Methoden, und wird seit den frühen 1990er Jahren eingesetzt. Im Gegensatz zu den traditionellen Vorgehensmodellen wird bei Scrum das Projekt nicht von Anfang an komplett durchgeplant und dann Phase für Phase umgesetzt, sondern in kleinere Arbeitspakete zerlegt. Diese werden dann gebündelt und der Priorität nach in sog. Sprints (fixe Zeitperioden von typischerweise 2 bis 4 Wochen) abgearbeitet. Bei ständigem Kontakt mit dem Auftraggeber entstehen so in regelmäßigen Abständen lauffähige Software-Inkremente und bei Abweichungen von dem gewünschten Weg kann schnell korrigiert und gegengesteuert werden.

**1.1 Rollen**

Bei Scrum werden 3 Rollen unterschieden: Product Owner (PO) , Entwicklungsteam und Scrum Master. Der Product Owner ist derjenige, der dem Entwicklungsteam gegenüber die zu lösende Implementierungsaufgabe definiert, er ist dem Team gegenüber der Auftraggeber. Er ist verantwortlich für die Eigenschaften und den wirtschaftlichen Erfolg des Produktes. Der Product Owner kann der tatsächlicher Auftraggeber der Software sein (ein externer oder interner Kunde), oder es ist jemand, dem diese Aufgabe delegiert wurde. Primäre Aufgaben des Product Owner sind die Anforderungen zu definieren, zu priorisieren und der fertig gestellten Funktionalitäten abzunehmen. Dem Entwicklerteam kommt die Aufgabe zu, das Softwareprodukt gemäß den Anforderungen zu implementieren und dabei auch die geforderten Qualitätsstandards einzuhalten. Das Team organisiert sich und seine Arbeit selbst, dazu besteht es aus Mitgliedern verschiedener Fachgebiete, z. B. Architektur, Entwicklung, Testen, Dokumentation und Datenbank. Der Scrum Master tritt als Schiedsrichter auf, er trägt dafür Sorge, dass die vereinbarten (Scrum-)Regeln von allen Beteiligten eingehalten werden. Bei den regelmäßigen Meetings wirkt er als Moderator. Treten Probleme auf, z.B. externe Blockaden, versucht der Scrum Master diese zu beheben um zu gewährleisten, dass das Team in seiner Arbeit nicht ausgebremst wird. Vor allem bei der Einführung von Scrum und ersten Projekten ist der Scrum-Master wichtig für das Gelingen. Später, wenn PO und Team bereits Erfahrung mit Scrum haben, kann er sich mehr und mehr zurückziehen und sich auf die Moderation der Meetings beschränken.

**1.2 Der Scrum-Prozess**

Am Anfang der Software-Erstellung stehen die Anforderungen des Kunden. Diese sind oft noch unvollständig oder vage. Der Kunde bzw. der Product Owner formuliert seine Anforderungen daher erst mal grob soweit sie bekannt sind und fasst sie als Ansammlung von User Stories im Product Backlog zusammen. Die wichtigsten Funktionen werden entsprechend priorisiert und müssen nun noch hinreichend genau spezifiziert werden, so dass das Entwicklungsteam den Aufwand schätzen kann. Am Anfang eines Scrum-Entwicklungszyklus, dem sog. Sprint, schätzt das Team im Sprint Planning Meeting die User Stories mit der höchsten Priorität und zerlegt sie in kleinere Teilaufgaben. Es werden dann vom Team nach absteigender Priorität so viele Aufgaben für den aktuellen Sprint Backlog zur Implementierung ausgewählt, wie mit den gegebenen Ressourcen bearbeitet werden können. Das Team verpflichtet sich, diese Aufgaben bin zum Ende des Sprints umzusetzen (commitment). Teilweise wird anstatt Verpflichtung auch der Begriff Prognose verwendet, da die Erfahrung zeigt, dass die Qualität des Produktes leidet, wenn das Team verpflichtet ist, gegen Ende des Sprints noch unbedingt alle Aufgaben fertigstellen zu müssen. Im Anschluss an das Sprint Planning werden die Aufgaben aus dem Sprint Backlog auf Karteikarten oder kleine Zettel geschrieben und im neuen (bereinigten) Task Board in der Spalte ToDo platziert. Innerhalb eines Sprints organisiert das Team seine Arbeit selbst, der PO und der Scrum Master dürfen keine Anweisungen geben. Jeden Tag trifft sich das Entwicklungsteam in einem kurzen Daily Standup Meeting in dem jedes Teammitglied über seine Arbeit (aktuelle, erledigte und nächste Aufgaben, Probleme) berichtet. Am Ende des Sprints wird im Sprint Review Meeting das Produkt-Inkrement dem PO bzw. dem Kunden präsentiert und die Implementierungen werden abgenommen (oder zurückgewiesen und wieder in das Product-Backlog aufgenommen). In der den Sprint abschließenden Sprint-Retrospektive wird der Verlauf des Sprint kritisch im Rückblick überprüft. Es nehmen nur das Team und der Scrum Master teil und untersuchen, was für Probleme es gab und was besser gemacht werden kann.

**1.3 Vorteile und Nachteile von Scrum**

Die weite Verbreitung von Scrum deutet bereits darauf hin, dass es als Werkzeug funktioniert und viele Projekte zur Zufriedenheit der Stakeholder abgeschlossen werden konnten. Bei der Einführung von Scrum in einer klassisch gemanagten Organisation sind die klaren Vorgaben und Handlungsanweisungen von Scrum ein Vorteil. Diese strenge Struktur kann sich aber auch im Verlauf der Zeit als Nachteil herausstellen.

Vorteile:

• schafft Transparenz für alle Beteiligten, vermeidet bösen Überraschungen zum Ende des Projektes

• fördert Teamgedanken

• fordert Eigenverantwortung des Teams

• Entlastung des Managements / der Projektleitung

• integrierter Verbesserungsprozess durch Retrospektiven

• geeignet für verteilte Produktentwicklung in großen Projekten mit Scrum of Scrums

Nachteile:

• geringere Planbarkeit: zum Projektstart kaum belastbare Aussagen zum Aufwand möglich

• Trägheit: Änderungen und neue Anforderungen erst im nächsten Sprint umsetzbar (bis zu 2 Sprint-Perioden Verzögerung)

• Arbeit innerhalb des Sprint ist wenig strukturiert

• gegen Ende des Sprints steigt die Belastung des Entwicklerteams

• Supportaufgaben können nicht parallel bearbeitet werden ohne im Sprint zurück zu fallen, oder es muss für Support eine fixer Zeitanteil reserviert werden

• lange Zeitspanne zwischen Auftreten eines Problems und dessen Bearbeitung in der Sprint-Retrospektive

• hoher Zeitbedarf für die regelmäßigen Meetings

**2 Kanban**

Kanban wurde als Methode der Produktionsplanung und -steuerung zuerst in der japanischen Automobil-Industrie der Kriegs- und Nachkriegszeit entwickelt und eingeführt. Die Firma Toyota versuchte zu der Zeit, den Abstand zu den amerikanischen Wettbewerbern zu verringern. Hierzu sollten die Kosten der Produktion verringert und die Qualität der Produkte verbessert werden. Wegen den damals sehr begrenzten Ressourcen galt es jede Art von Verschwendung zu vermeiden. Dies gilt nicht nur für Rohstoffe, sondern auch für Zeit, Personal, Kapital. Die seit dieser Zeit entwickelten Methoden sind heute auch als Lean Production, just in Time und Kontinuierlicher Verbesserungsprozess (KVP)bekannt. Das Wort Kanban besteht aus den zwei Zeichen 看 ( kan=sehen) und 板 (ban=Tafel, Brett) und läßt sich etwa mit Sichttafel, Aushängeschild oder auch Pendelkarte übersetzen. Diese Kanban-Karten sind ein zentraler Bestandteil im Kanban-Prozess. Im industriellen Produktionsprozess steht jede Kanban-Karte für einen Behälter einer bestimmten Grösse, der eine festgelegt Anzahl von Bauteilen enthält. Die Anzahl von Kanban-Karten für ein Bauteil oder eine Bauteilgruppe ist begrenzt, auf diese Weise soll verhindert werden, dass zu viel auf Lager produziert wird, denn große Zwischenlager binden Ressourcen. Der gesamte Produktionsprozess wird betrachtet als eine Aneinanderreihung von Quellen und Senken von Produktionsgütern, mit Zwischenlagern als Puffer. Eine Senke nimmt sich einen Behälter aus dem Zwischenlager (Pull-Prinzip), verarbeitet alle Teile darin und füllt (als Quelle) das nachgelagerte Zwischenlager. Hat das nachfolgende Zwischenlager einen bestimmten Höchststand überschritten, darf nicht weiter produziert werden. Wird dagegen ein bestimmter Mindeststand unterschritten, so muss wieder Nachschub produziert werden. Auf diese Weise werden Probleme oder Engpässe schnell sichtbar und es können entsprechende Gegenmaßnahmen unternommen werden. Andererseits können durch die mehrstufigen Zwischenlager Schwankungen bei Nachfrage, Zulieferung oder Personalstärke in gewissen Grenzen ausgeglichen werden. Die Zwischenlager der einzelnen Produktionsstufen werden auf der Plantafel (auch Kanban-Tafel) visualisiert, welche an zentraler Stelle für alle Beteiligten gut sichtbar platziert wird. Für jede Produktionsstufe gibt es auf der Plantafel eine Spalte oder 6 Zeile mit festen Plätzen für die Kanban-Karten. Die Karten der leeren Behälter werden hier für jeden sichtbar plaziert, so dass auf einen Blick der Bestand der Zwischenlager erkennbar wird.

**2.1 Kaizen**

Eine wesentliche Idee bei Kanban ist Kaizen, der kontinuierliche Verbesserungsprozess. Durch Kaizen soll der Arbeitsfluss ständig optimiert werden, Probleme sollen an ihrem Ursprung gelöst werden, Verschwendung/Vergeudung (engl. Waste) und Überlastung vermieden bzw. minimiert werden. Alle Beteiligten sind aufgefordert, sich bei diesem Kontinuierlichen Verbesserungsprozess einzubringen. Wenn durch eine Veränderung eine Verbesserung im Prozess erreicht wird, so wird diese Änderung zum neuen Standard erklärt. Auch übermäßige Meeting sind im Sinne von Kaizen Vergeudung von wertvollen Ressourcen, daher sollte der Zeitbedarf für Meetings optimiert werden. Manche Teams kommen ganz ohne regelmäßige Meetings aus, andere Teams machen Daily Standup Meetings, regelmäßig oder nach Bedarf Nachschubmeetings und ReleasePlanungsmeetings. Meetings sollten auch abgehalten werden, wenn dafür akuter Bedarf besteht z.B. weil eine Blockade beseitigt werden muss oder weil ein Engpass besteht. Als Ergebnis ist dann eine Verbesserung des Prozesses zu erwarten. Auf diese Weise kann auch das WIP-Limit bei Bedarf angepasst werden, oder auch die Kanban-Tafel umgestaltet werden, z.B. durch mehr oder weniger Spalten, durch Einführung von Swimlanes, Ticket-Typen oder Service-Klassen. Für eine objektive Beurteilung der Verbesserungsmaßnahmen bedarf es Kennzahlen. Wichtige Kennzahlen bei Kanban sind die Durchlaufzeit der Tickets (Lead time), der Durchsatz an Tickets und die Fehlerrate. Für aussagekräftige Kennzahlen ist es wichtig, dass die Kanbankarten von Art und Umfang vergleichbar sind. Große Aufgaben sollten daher in mittlere und kleine Aufgaben zerlegt werden, bevor sie bearbeitet werden, kleine Aufgaben sollte man Bündeln und dann an einem Stück bearbeiten. Es empfielt sich, getrennte Kennzahlen für verschiedene Aufgaben-Typen zu ermitteln, z.B. Support-Aufgaben und Implementierungs-Aufgaben zu trennen. Die Kennzahlen sollten laufend erfasst und in Diagrammen dargestellt werden, um Probleme zu erkennen und Verbesserungen zu erfassen. Bei einem Cumulative Flow Diagram beispielsweise werden in regelmäßigen Abständen (z.B. wochenweise) die Anzahl der Tickets in den einzelnen Spalten der Kanban-Tafel kummulativ, also übereinander gestapelt eingetragen.

**2.2 Vorteile und Nachteile von Kanban**

Aufgrund der wenigen Regeln kann Kanban gut in bestehende Prozessen und Strukturen integriert werden. Es muss nicht eine plötzliche große Umstellung vorgenommen werden, sondern Kanban kann auch evolutionär Schritt für Schritt eingeführt werden.

Die wichtigsten Vorteile von Kanban sind:

• Transparenz für alle Stakeholder durch Visualisierung des Workflows

• deckt Kapazitätsengpässe im Prozess auf

• schnelle Reaktion auf Veränderung möglich

• Flexibilität: an spezifische Gegebenheiten anpassbar

• Skalierbarkeit: nutzbar für Einzelperson, kleine, mittlere und große Teams oder für eine Hierarchie

• Kontinuierliche Lieferung von Software-Inkrementen möglich (Continuous Integration)

• Vielseitigkeit: einsetzbar bei Entwicklung, Betrieb, Support, DevOps , Continuous Delivery

Auch Kanban hat natürlich Nachteile, die beachtet werden müssen:

• geringe Planbarkeit bei Umsetzung großer Projekte

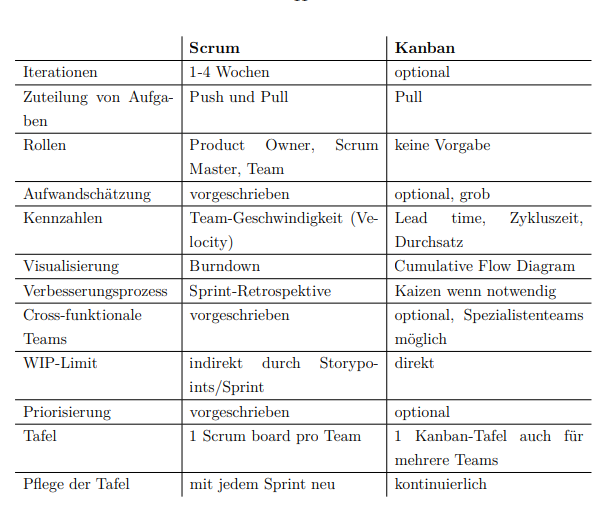
• geringe Übersichtlichkeit der Kanban-Tafel bei großen Projekten und großen Teams

• längere Durchlaufzeiten bei vielen und grossen Zwischenpuffern

• externe Kunden erfordern i.d.R. eine Aufwandschätzung

• Kanban-Tafel schwierig zu implementieren bei externen Teams oder Teammitgliedern (g.g.f. elektronisch)

**3 Vergleich Scrum und Kanban**

Scrum und Kanban haben viele Gemeinsamkeiten und sind sich in manchen Punkten ähnlich. Beide sind agil und schlank (lean) und verwenden das Pull-Prinzip, haben eine gewisse Beschränkung der WIP und unterstützen einen kontinuierlichen Verbesserungsprozess durch Schaffung von Transparenz. Scrum und Kanban setzen beide auf sich selbst organisierende Treams und auf die eigenverantwortliche Mitarbeit und teilen die Arbeite in kleinere Arbeitspakete auf. Jedoch macht Scrum sehr viel mehr Vorgaben und erlaubt kaum Variationen. Kanban ist leichtgewichtiger und anpassbar, nicht das Vorgehensmodell steht im Mittelpunkt sondern die zu erledigenden Aufgaben in ihrer angestammten Umgebung.

Jonas Richter

Infos und Grafiken zusammengetragen aus Scrumban Hochschule für Ökonomie & Management Essen