

Scrum - eine Methode der agilen Softwareentwicklung

Merkmale:

- Iterativer Prozess (kurze Zyklen)
- Autonomes Entwicklerteam
- kurze/kleine Teilaufgaben
- schnelle Reaktion auf neue Anforderungen
- Regelmäßige Verbesserung des Prozesses



Rollen:

Product Owner (m/w) (PO):

Der Product Owner trägt die Verantwortung für den Erfolg des Projektes und definiert die Anforderungen. Er/Sie hat die Entscheidungsgewalt über den Inhalt des Projektes und sollte vor allem Motivator und Strategie des Projektes sein.

Er/Sie ist meist kein „Fachmann“ und muss nicht exakt spezifizieren können.

Implementierungsdetails sind für ihn/sie nicht relevant. Er/Sie interessiert sich nur dafür, dass das fertige Produkt seinen/ihren Vorstellungen entspricht.

Entwicklerteam (Dev-Team):

Das Entwicklerteam besteht meist aus Spezialisten verschiedener Programmierbereiche (z.B. DB-Management, Netzwerk, GUI, ...). Es hat als Hauptaufgabe die Umsetzung der vom PO definierten Ziele. Dabei nimmt es die formale Spezifikation vor und ist verantwortlich für die Umsetzung des Programms. Hierfür schätzt das Team den Arbeitsaufwand einzelner Aufgaben ab und identifiziert kritische Bereiche und kommuniziert diese nach außen. Das Team testet Funktionalitäten und bringt die verschiedenen Versionen raus (Releases). Es muss sich nicht um äußere Faktoren organisatorischer Art kümmern (z.B. Raum, Material).

Der Scrum-Master (m/w) (ScM):

Der Scrum-Master soll das Entwicklerteam zum Erfolg führen. Er ist vor allem beratend tätig und vermittelt zwischen PO und Dev-Team (z.B. bei Konflikten). Der ScM hilft dem Dev-Team bei der Spezifikation der Aufgaben (Backlog) und hilft als Coach dem Team sein volles Potential zu entfalten. Er ist der Moderator aller Meetings zwischen PO und Dev-Team und sollte immer über alle Vorgänge Bescheid wissen.

Der ScM muss kein Experte in Teilgebieten sein und gibt die technische Verwaltung/Wartung meist an andere ab. Er ist nicht verantwortlich für Budget/Fortschritt/Produkterfolg und sollte kein Entwickler im Dev-Team sein (sonst Gefahr des Verlustes der Überparteilichkeit/Vermittlerrolle).

Hilfsmittel:

Die Productvision

Der PO und das Dev-Team erarbeiten am Anfang eine gemeinsame Vorstellung des Produktes. Dabei werden Zielgruppe, Benötigte Ressourcen, Produktmerkmale und Wert des Produktes für das Unternehmen ausgeführt. Die Productvision sollte tief durchdacht und vor allem stabile Grundlage für das Projekt sein (z.B. kein Wechsel von Chat-App zu Video-Streaming-App). Die PV leitet die Selbstorganisation des Dev-Teams.

Die User Story

Die User Story ist ein zentrales Element beim Erstellen des Product Backlogs. Der Product Owner (oder auch Mitglieder der Entwickler) kann in jeder Story ein „Nutzer-Erlebnis“ mit dem Programm aufschreiben. Dabei wird keine Fachsprache oder „Code“ verwendet (z.B. Programm schließt beim klicken des „x“). Aus den User Stories werden dann vom Entwicklerteam konkrete Programmieraufgaben extrahiert, die dann in das Product Backlog aufgenommen werden.

Das Product Backlog

Im Product Backlog werden die einzelnen Anforderungen an das Produkt ihrer Priorität nach sortiert. Dabei ist es Aufgabe des ScM und des Dev-Teams die einzelnen Einträge in ungefähr gleich aufwendige Items zu zerteilen (Product Backlog Items „PBI“). Der PO sortiert die Items nach Priorität und ist verantwortlich für die Reihenfolge (Ausnahme: aufeinander Aufbauende Aufgaben müssen vom Dev-Team kommuniziert werden!) Es bietet sich an anfangs zwischen den Prioritätsstufen Raum für weitere Stufen zu lassen (z.B. Prio 5, Prio 10, Prio 15), damit später Aufgaben zwischengeschoben werden können. Jede Prioritätsstufe wird nur einmal vergeben!

Das Sprint Backlog

Aus dem Product Backlog nimmt das Dev-Team pro iterativen Entwicklungsprozess ein Product Backlog Item heraus und spezifiziert dieses für die Definition der Aufgaben im Team. Ziel eines Sprints ist es die Aufgaben des PBI zu erledigen. Am Ende des Sprints sollten die ausgewählten PBIs zu einem potentiell fertigem Produkt zusammengebaut werden (d.h. nach einem Sprint hat man einen immer weiter verbesserten Prototypen).

Wenn der Scrum Master Probleme im Sprint erkennt, kann er diese im Product Backlog spezifizieren und für alle sichtbar machen.

Ablauf:

1. Definition der Product Vision (z.B. mit User Stories)
2. Spezifizierung des Product Backlogs (Erstellen konkreter Anforderungen aus User Stories)
3. Sprint:
 - Dauer: meist eine Woche
 - Auswahl der PBIs für den Sprint
 - 3.1. täglicher Scrum:
 - tägliche 5-Minuten-Besprechung (Vorbereitet, kurz, prägnant)
Inhalt: Was haben wir geschafft? Was machen wir heute? Wo sind Probleme?
 - Aktualisierung des Sprint Backlog
 - 3.2 Ende des Sprints: Product Backlog überprüfen, ob sich neues ergibt
 - Muss Aufwand neu geschätzt werden?
 - Muss ein PBI zerkleinert werden?
 - Alternativen besprechen
 - Risiken kommunizieren
4. Sprint Rückblick
 - Was ging gut, was schlecht? (Glad/Sad Liste)
 - Wo haben wir länger/kürzer gebraucht? (Timeline)
 - Wie machen wir es nächsten Sprint besser? (Pragmatisch: Problemlösen NICHT Schuldsuche!) (Keep/Drop Liste)
 - Scrum Master legt Ziele für nächste Woche fest.
5. Zurück zu neuem Sprint, bis das Produkt fertig zum Release ist