

SCRUM & KANBAN





SCRUM – ALLGEMEINES

- Das Wort „Scrum“ kommt aus dem Rugby und bezeichnet ein angeordnetes Gedränge: nach einem kleineren Regelverstoß wird das Spiel kurz unterbrochen, die Spieler sprechen sich in ihrem Team kurz ab und positionieren sich, danach wird der Spielverlauf neu gestartet
- Eines der bekanntesten agilen PM-Vorgehensmodelle
- Basiert auf inkrementeller Vorgehensweise mit einem Mix aus enger Zusammenarbeit, stetigem Feedback und regelmäßigen Anpassungen
- Einfache Struktur
- Klar definierte Rollen



SCRUM – ROLLEN

- **Entwicklerteam:** ohne PL, steht im Zentrum, selbstorganisiert
- **Scrum Master:** Fachmann für die Scrum-Methode, koordiniert Entwicklungsprozess und beseitigt Hindernisse, hat ein Auge auf die Regeleinhaltung durch alle Projektbeteiligten, visualisiert aktuellen Projektfortschritt
- **Product Owner (PO):** Produktverantwortlicher, der Anforderungen (requirements) definiert, priorisiert und tauscht (kann Kunde bzw. AG sein), hat Wissen in seinem Gebiet, hat die Befugnis, Entscheidungen zu treffen, verantwortlich für das finale Produkt
- weitere **Stakeholder** als Beobachter, Ratgeber und am Endprodukt Interessierte



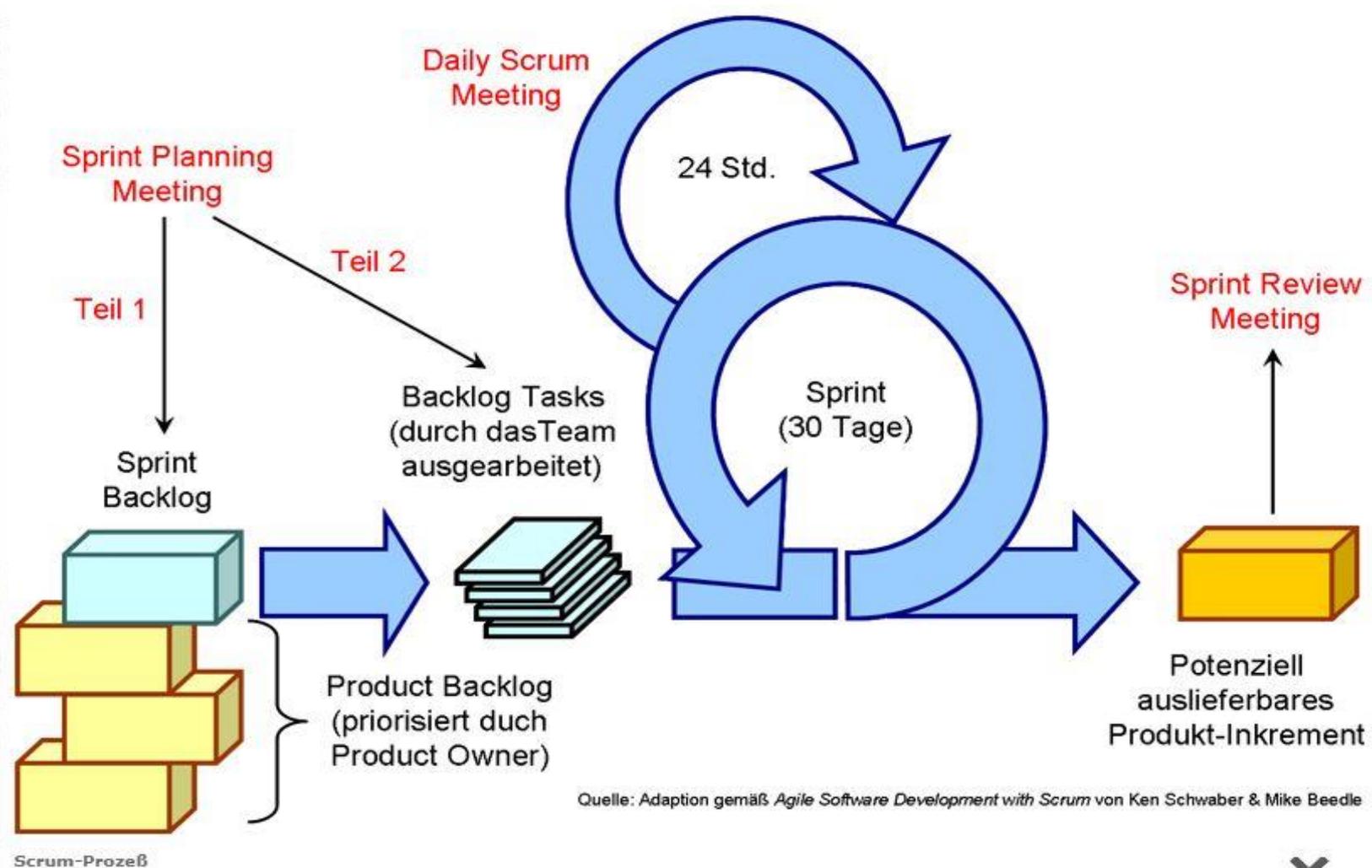
SCRUM – ZIEL

- Aufgrund teilweise großer Änderungen während des Projektverlaufs ist das Ziel und die Planung nicht konkret von Anfang an dargestellt, sondern man nähert sich diesem schrittweise im Laufe der Umsetzung
 - skizzierte **Ziele (epics)** = grobe Beschreibung des grundsätzlichen Ziels
 - Unterteilung der epics in **user stories (Arbeitspakete)**
 - Unterteilung der Arbeitspakete in **story items** (dies aber erst kurz vor der Bearbeitung des jeweiligen epics)



SCRUM – ABLAUF

EIGENE GRAPHIK ERSTELLEN!



Quelle: Adaption gemäß *Agile Software Development with Scrum* von Ken Schwaber & Mike Beedle



SCRUM – ABLAUF

- Das Entwicklerteam arbeitet in immer gleich langen Zyklen (**Sprints**, 1 Sprint = 1-4 Wochen). Während eines Sprints wird dieses nicht vom PL oder durch Änderungen unterbrochen.
- Während eines Sprints sammelt der PO **Anforderungen** an das System (**requirements**) und an **Arbeitspakete** als auch an die weiteren Entwicklungsschritte. Diese trägt er in die Liste des **Product Backlogs** ein. Der PO fügt durchgehend neue Anforderungen und Arbeitspakete hinzu und priorisiert diese ständig neu.
- Vom Product Backlog werden **Arbeitspakete** in das **Sprint Backlog** verlegt, wo sie nochmals in kleinere Arbeitspakete (**tasks, user items**) heruntergebrochen und bestimmten Team-Mitgliedern zugeteilt werden. Diese Tasks werden im nächsten Sprint bearbeitet.



SCRUM – ABLAUF

- Täglich findet ein **Daily Scrum** mit allen Team-Mitgliedern statt. Hier berichtet jede Person über den aktuellen Stand, was sie am Tag zuvor getan bzw. was sie dabei behindert hat und was sie bis zum nächsten Daily Scrum machen wird. Neben der Transparenz des Projektverlaufs dient das Daily Scrum auch der Vermeidung von Abstimmungsfehlern.
- Am Ende eines Sprints erfolgt ein **Sprint Review**, in dem das Team dem Product Owner und den weiteren interessierten Stakeholdern den aktuellen Stand des Produkts vorstellt. Außerdem wird ein neues **Arbeitspaket** mit der höchsten Priorisierung herausgenommen und im nächsten Sprint bearbeitet.
- Im **Sprint Retrospective** werden Erfahrungen des letzten Sprints analysiert und Verbesserungsaspekte für die Zukunft gesammelt.



SCRUM – VORTEILE

- Der Product Owner kann regelmäßig den Plan und die Umsetzung beeinflussen und kurzfristig auf Fehlentwicklungen oder Probleme reagieren
- Vermeidung von Diskussionen über zeitlich weit entfernte, stark ins Detail gehende Anforderungen an das Produkt bei Projektstart → Prozess entwickelt sich durch Eigendynamik → Team-Mitglieder erarbeiten Inhalte und teilweise Vorgaben selbst → Produktivität steigt → schnellerer und größerer Erfolg
- Vermeidung von Fehleinschätzungen und hohen Korrekturkosten
- Kein genaues Projektende → groben Zeitrahmen oder Maximalkosten als Projektende definieren
- Besseres Zeitmanagement durch kleine Arbeitspakete, Sprints und Daily Scrums (Stand Up Meetings)
- Höhere Motivation der Mitarbeiter aufgrund von mitarbeiterfreundlichen Prinzipien, Transparenz und sichtbarem Projektfortschritt



KANBAN

Methode, welche bei SCRUM verwendet wird

Entwickelt durch David Anderson (US-Amerikanischer Unternehmensberater), 2007 vorgestellt

Kan = Signal
Ban = Karte } „Signalkarte“

→ Signalkarten verwendet man bei der Lean-Produktion für eine optimale Steuerung des Materialflusses

Exkurs: **Lean Management von Toyota**

- Unterbindung von Verschwendung jeglicher Art durch ...
 - Muda: Arbeit, die keinen zusätzlichen Wert für das Produkt liefert
 - Muri: Überlastung der Mitarbeiter und Maschinen
 - Mura: Unregelmäßigkeit von Prozessen

→ effiziente Gestaltung der Wertschöpfungsprozesse/Produktion von Gütern



FUNKTION

- Kanban verläuft nach dem **Pull-Prinzip (Hol-Prinzip)**, d.h. Team-Mitarbeiter bedienen sich selbst von den Arbeitspaketen
 - weniger Vorgaben bei Arbeitspaketen → auf eigene Bedürfnisse anpassbar
- Fokus bzw. Ziel: Fluss-Optimierung der Aufgaben
- zudem Verhinderung von Engpässen (bottlenecks) und Begrenzung der Kapazitäten durch Angabe an maximalen Arbeitspaketen pro Person



STORY / FEATURE

- Element der Kanban-Methode
- **Arbeitspakete/Liefereinheiten** mit Aufgabenbeschreibung, Nennung des Verantwortlichen und zeitlicher Frist (due date)
- werden von links nach rechts über das Kanban-Board geschoben
- die **Spalte**, in der sich die Stories befinden, zeigen den Workflow bzw. den IST-Zustand



STORY / FEATURE

Aufbau (Beispiel):

Datum der Aufnahme am
Kanban-Board

Deadline

**Beschreibung des
Arbeitspakets:**

-
-

Priorisierung:

★ = Priorität

★★ = Panik

Verantwortliche Person (Sticker oder
farbiger Magnet pro Person)



PRIORISIERUNG DER STORIES

Die Stories werden anhand der folgenden Reihenfolge nach ihrer Wichtigkeit der Bearbeitung priorisiert:

1. Panik-Stories

- Haben höchste Priorität und werden immer als erstes bearbeitet
- Werden möglichst von allen bearbeitet und sind ständig in Bewegung
- In dringenden Fällen ggf. andere Aufgaben unterbrechen und Regel bzgl. der maximalen Anzahl an Stories brechen

2. Priorität

- nach Zeit und Wichtigkeit einer Story

3. Deadlines

4. Alter

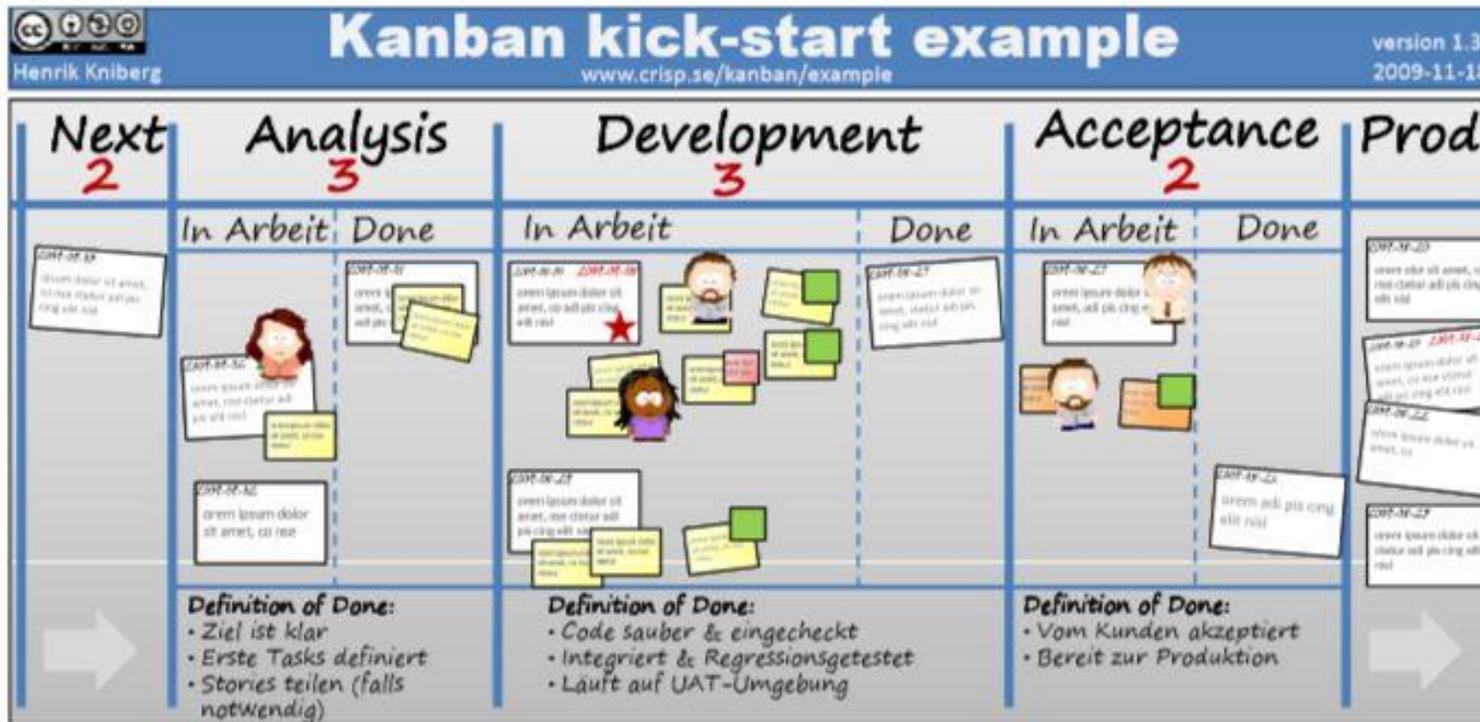


TASK UND DEFEKT

- Zusätzliche Markierung der Stories
- Zeigen den **aktuellen Zustand des Arbeitspakets**, in der sich eine Story befindet, oder die notwendige Aktivität, die zur Implementierung der Story benötigt wird
- Fließen nicht über das Kanban-Board wie die Stories
- Zustand durch Farbe unterstützt
 - **Grün** = done / fertig gestellt
 - **Orange** = defekt
 - **Rot** = blockiert
- Sobald eine Story einen grünen „Done-Status“ erhält, werden alle Tasks entfernt
- Wenn Fehler bemerkt werden: **Defekt-Notiz**



KANBAN-BOARD (BEISPIEL)



Graphik
beibehalten?

<http://agilecoach.de/kanban/Kanban-example-German.pdf>

Feature / Story

Datum der Aufnahme am Brett

2009-08-20 2009-09-30

(Beschreibung)

Harte Deadline (falls zutreffend)

★ = Priorität
★★ = Panik

Wer analysiert / testet gerade

Task / Defekt

Task = Task Defekt = Defekt

fertig gestellt

blockiert

wer arbeitet gerade daran

Was zuerst ziehen

1. **Panik-Features** ★★
(sollte von allen bearbeitet und in Bewegung gehalten werden. Unterbreche andere Arbeiten und verletze WIP-Limits falls notwendig)
2. **Priorität-Features** ★
3. **Harte Deadline Features** (nur wenn Deadline in Gefahr)
4. **Älteste Features**