LU18a - Arbeitspakete/Tasks

Ein sehr zentraler Aspekt bei Projekten ist das herunterbrechen des gesamten Projektes in einzelne Arbeitspakete (Work Packages) bzw. Aufgaben (Tasks).

Je nach Projektmethode heissen diese Stücke etwas anders, aber das Prinzip bleibt ungefähr das gleiche.

Wasserfall	**Agil**
• Projekt • Teilprojekt • Arbeitspaket • Arbeitsschritt	• Epic • (User) Story • Task • Bug

Für unsere Projekte werden wir in der Regel mit Epics und Stories arbeiten.

Backlog

Der Backlog enthält alle Tasks. Beim „Backlog grooming“ werden die Tasks vorbereitet, geschätzt und nach Priorität sortiert. Falls der Aufwand zu gross ist für einen Sprint, sollte der Task weiter heruntergebrochen werden. Wichtig ist es auch allfällige Abhängigkeiten zwischen Tasks zu erkennen und zu berücksichtigen.

Board

Ein Projekt-Board dient dazu den Status der einzelnen Tasks abzubilden. Besonders bei agilem Projektmanagement sind die Sprint-Boards, Kanban-Boards etc. ein zentrales Werkzeug, um einen Überblick über die Fortschritt der Iteration zu erhalten.

Die Spalten entsprechen quasi dem Status der Tasks (Ready, In Progress, Done), wobei man diverse weitere Spalten definieren kann (ToDo, Verify/Test, …). Die einzelnen Task wandern im Verlauf der Iteration (Sprint) von links nach rechts. Ein Kerngedanke bei Kanban und weiteren agilen Vorgehensmethoden ist es, nie zu viele Tasks miteinander zu bearbeiten, sondern nach Möglichkeit immer zuerst angefangene Tasks abzuschliessen.

Oftmals wird auch definiert, welche Bedingungen erfüllt sein muss, damit ein bestimmter Status gesetzt werden kann.

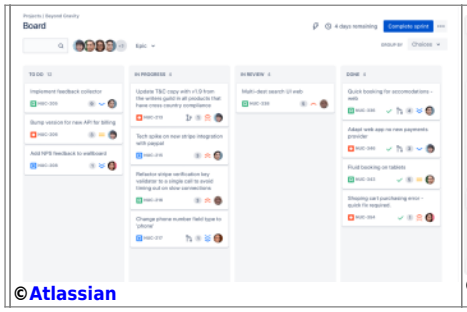
Beispiel Definition of Ready (DoR):

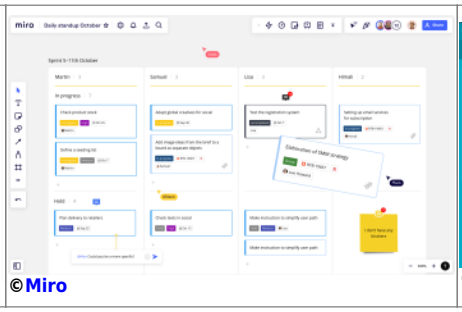
- Der Aufwand wurde vom Entwicklungsteam geschätzt.
- Mindestens ein Akzeptanzkriterium ist definiert, das beschreibt, wann die Story als erledigt gilt.
- Alle internen und externen Abhängigkeiten sind identifiziert und geklärt.

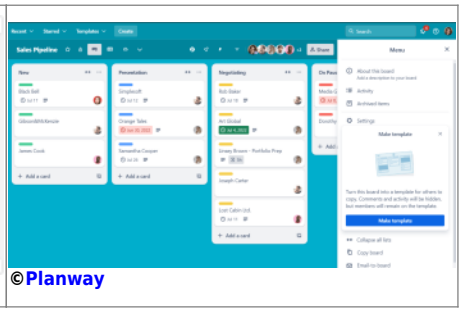
Beispiel Definition of Done (DoD):

- Der Code ist gemäss Richtlinien implementiert und formatiert.
- Die Build Pipeline ist grün, es gibt keine fehlerhaften Tests.
- Das Code-Review wurde durchgeführt.

Häufig dafür verwendete Produkte sind zum Beispiel Atlassian Jira, Miro, Trello, ...







Dokumentation

Es ist relativ fehleranfällig, wenn die Implementierung von Code und dessen Dokumentation getrennt sind. Sobald Änderungen nicht konsequent nachgetragen sind oder die Dokumentation eine kritische Grösse erreicht, wird es sehr mühsam Codeänderungen nachzuvollziehen.

Daher ist es sehr empfohlen die einzelnen Commits mit den Tasks zu verknüpfen...

T0D0

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/de/modul/ffit/3-jahr/java/learningunits/lu18/a?rev=1768849637

Last update: **2026/01/19 20:07**