

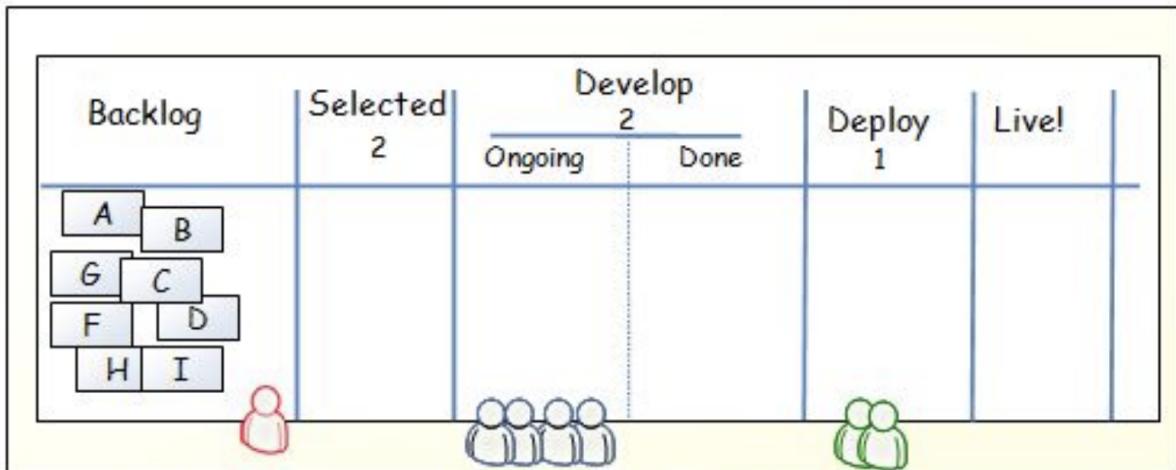
Agil LA2 Kanbanbilder

by Niklas Brandtner 18.03.2024

Interpretieren und beschreiben Sie jedes Bild aus Kanban Sicht (Welche Prozessschritte, wie viele Tasks, Task Zustände?, was passiert gerade, welche Kanban Prinzipien stehen dahinter, ...).

Reflektieren Sie das Gezeigte und beschreiben Sie sinnvolle **Verbesserungen**.

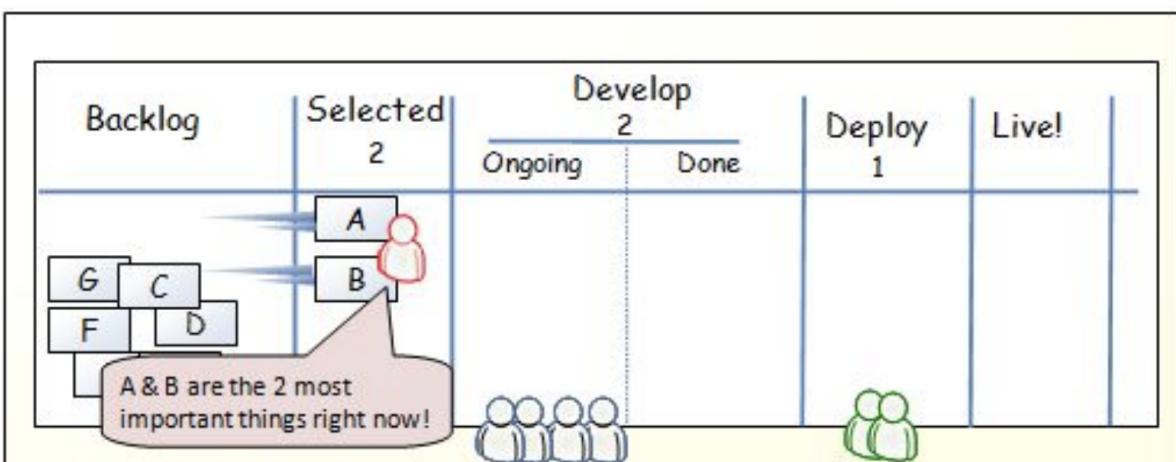
Bild 1



Man sieht ein typisches Kanbanboard wobei alle Tasks im Backlog liegen, außerdem wurden Limits für die Selected, Develop und Deploy Phasen festgelegt.

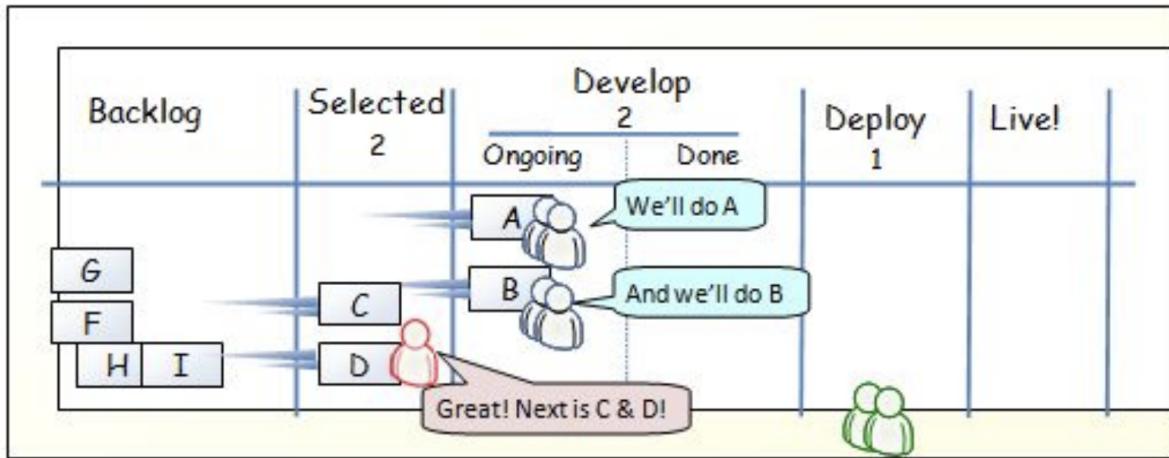
Die Develop Phase hat auch eine Unterteilung in Ongoing und Done was dafür sorgt dass sich falls das Deployment verzögert kein Bottleneck entsteht.

Bild 2



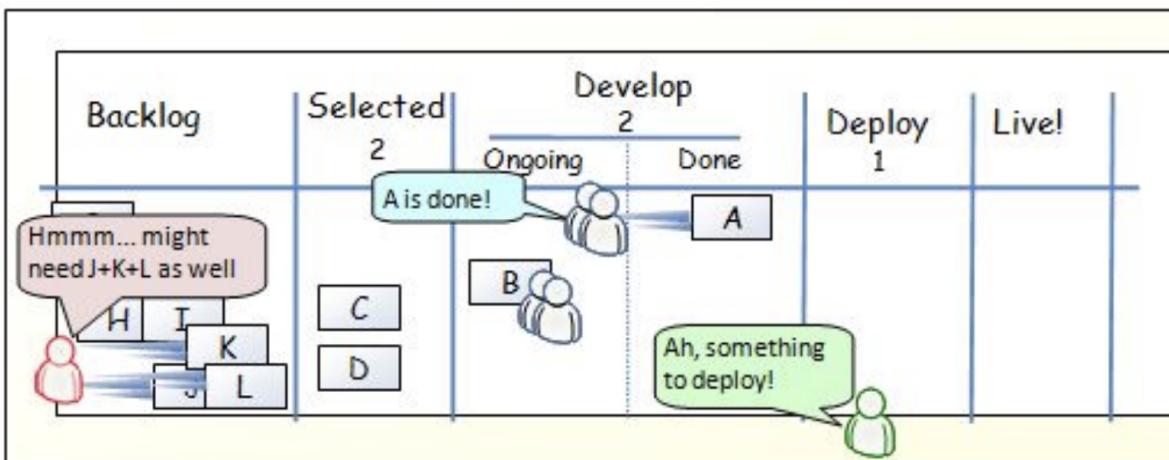
Rot schiebt als erstes die Tasks A & B in den Selected Zustand. Damit ist dieser voll und Blaue Mitarbeiter/Teams können sich daraus Tasks wählen welche sie erledigen.

Bild 3



Blau hat sich in 2 Teams geteilt wobei Team 1, Task A übernimmt und Team 2, Task B. Außerdem werden von Rot die Tasks C & D in Selected gepusht.

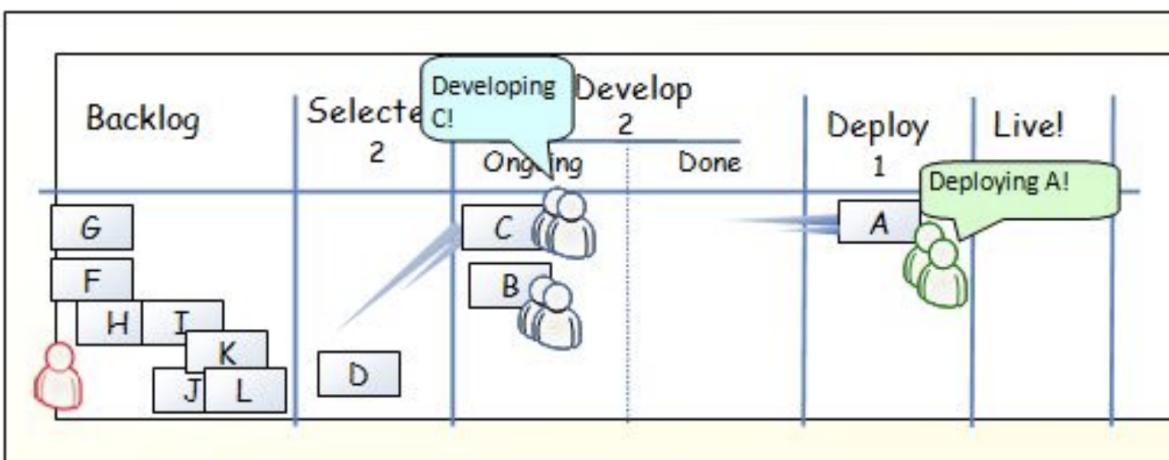
Bild 4



In diesem Schritt ist Team 1 mit dem Development von Task A fertig geworden und Grün freut sich auch arbeiten zu dürfen.

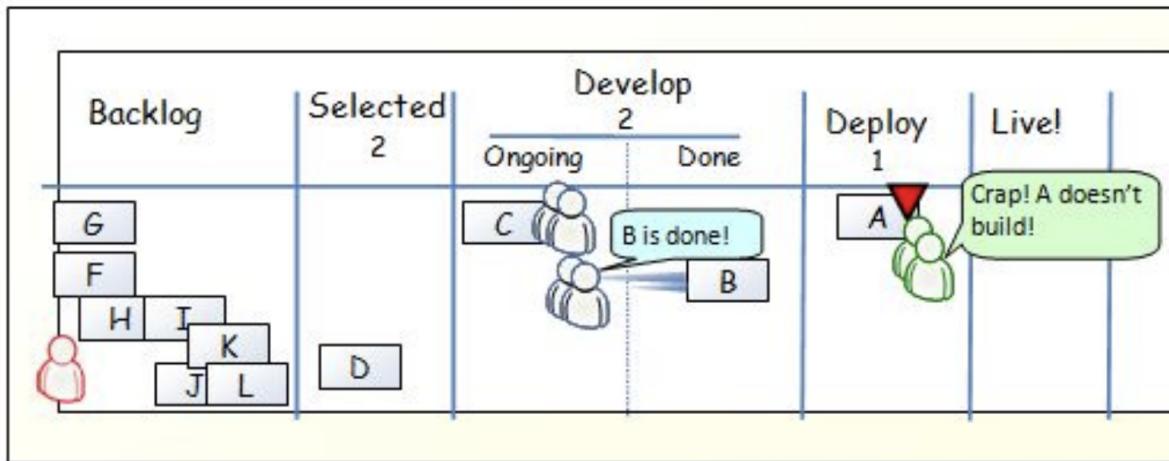
Rot überlegt laufend mit welche Tasks als nächstes priorisiert werden sollten.

Bild 5



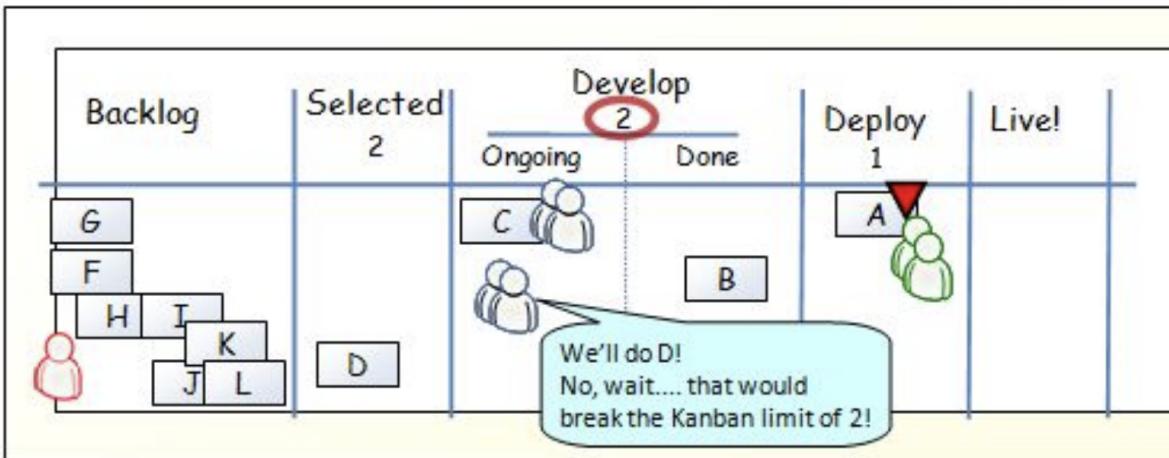
Rot ist auf Standby, Blau Team 1 nimmt sich jetzt als nächstes Task C während Team 2 noch an B arbeitet. Grün fängt an Task A zu Deployen.

Bild 6



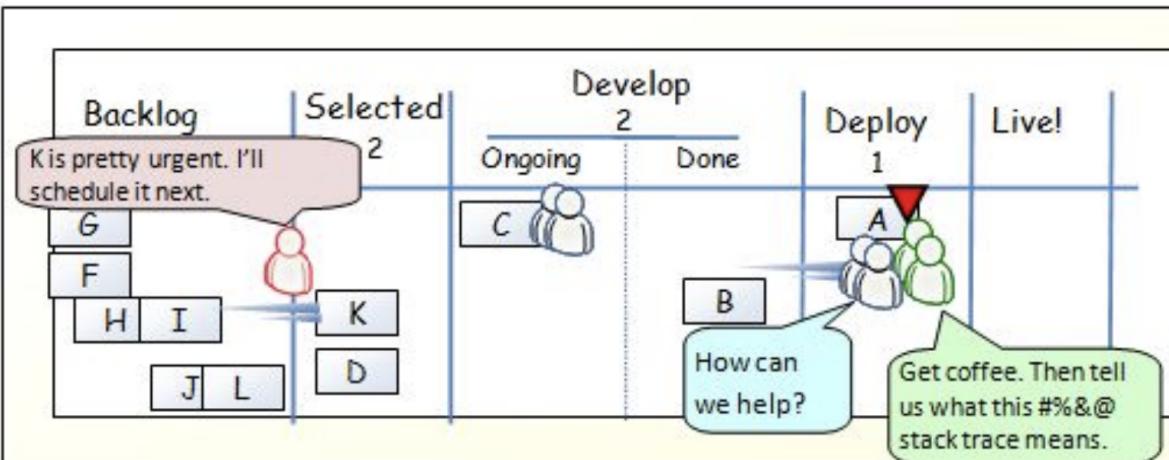
Blau Team 2 ist mit Task B fertig und pusht es ins Done Feld. Grün findet heraus das Task A beim Builden ein Problem hat.

Bild 7



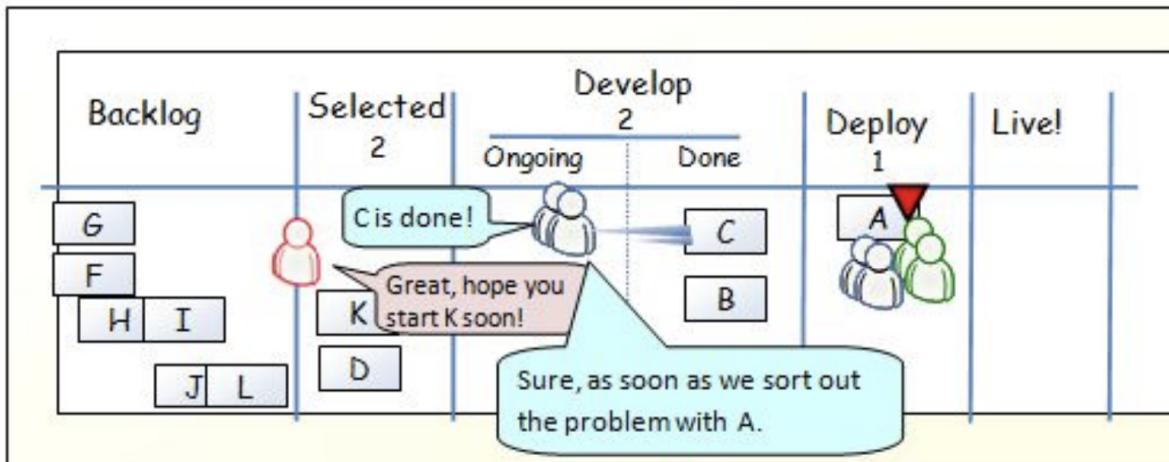
Während Grün an Task A und Team 1 von Blau und Task C arbeitet hat Team 2 jetzt ein Problem, Task B ist fertig fürs Deployment aber Grün ist noch mit A beschäftigt also sind zwei Tasks in der Development Phase so dass sie sich den nächsten Task nicht holen können um daran zu arbeiten.

Bild 8



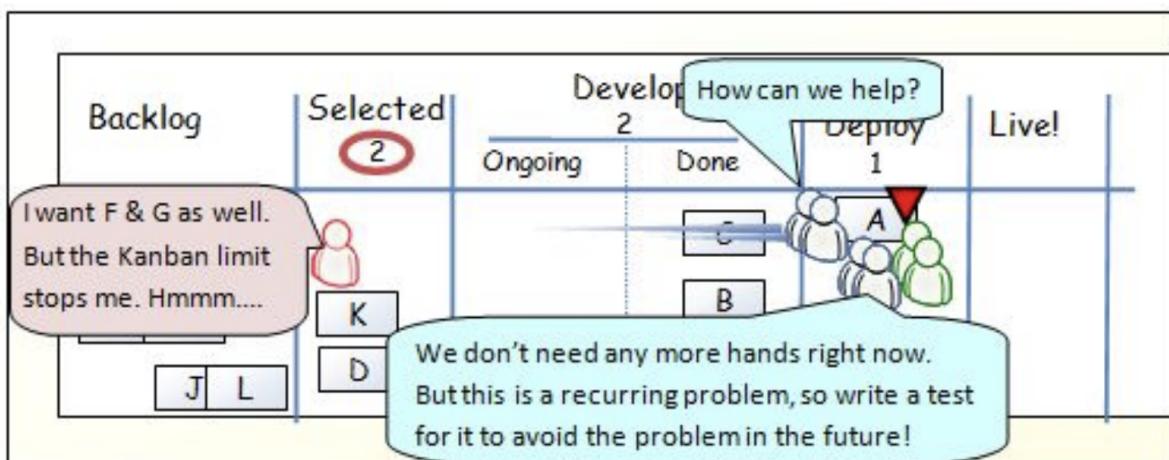
Rot entscheidet sich für den nächsten Task welcher in Selected gepusht wird. Team 2 von Blau kann nicht weiterarbeiten also wollen sie Grün helfen um das Bottleneck zu lösen.

Bild 9



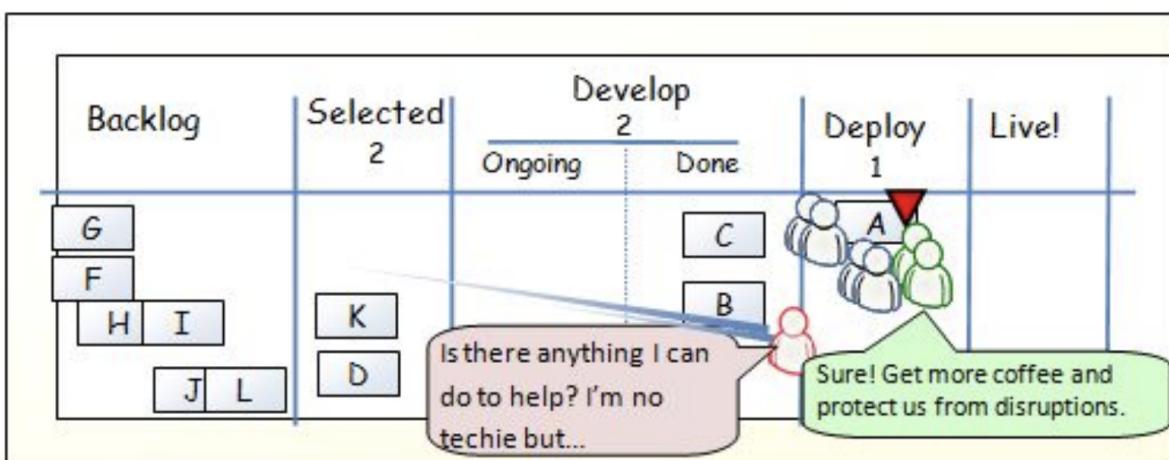
Team 1 ist mit Task C fertig und kann aber erst den nächsten Task annehmen wenn das Bottleneck gelöst wurde.

Bild 10



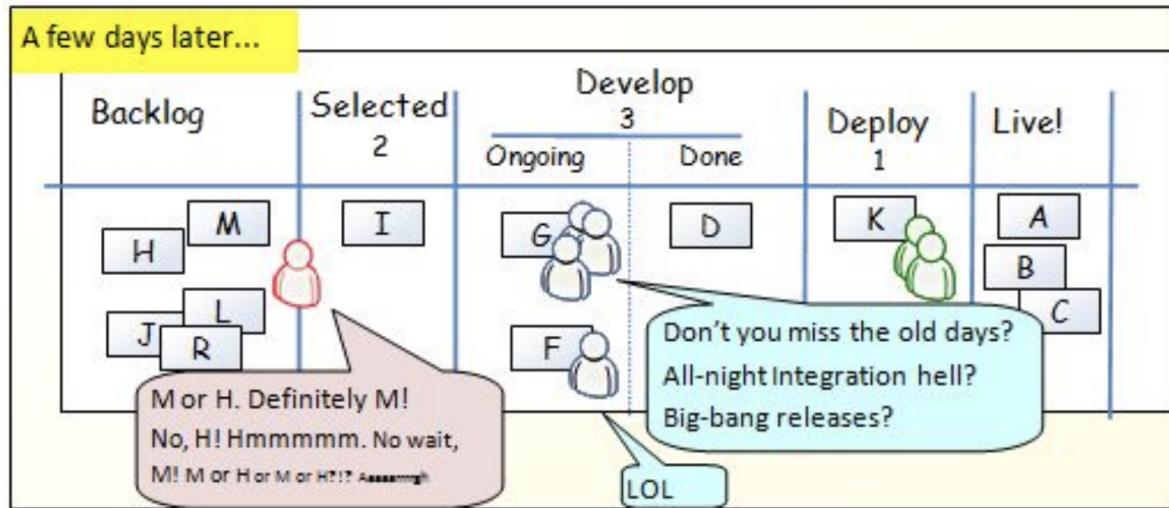
Rot wird ungeduldig. Team 1 will Grün ebenfalls helfen und automatisiert jetzt Tests für das aufgetretene Issue.

Bild 11



Rot mischt sich ein und wird aber weggeschickt :)

Bild 12



Das Bottleneck Problem wurde teilweise gelöst indem das Development Task-Limit erhöht wurde und automatisierte Tests auf die Build-Issues erstellt wurden.