

Scrum Agile Software Development

Volker G. Göhler

TU Bergakademie Freiberg

2022/05/18

Scope



Abbildung 1: Software Development [shutterstock.com]

klassische Software Entwicklungsmethodik

Wasserfall



Abbildung 2: Romker Wasserfall by Unknown author – This image is available from the United States Library of Congress's Prints and Photographs division under the digital ID ppsca.00551. Public Domain, <https://commons.wikimedia.org/w/index.php?curid=2170305>

Pros and Cons

was funktioniert gut und was nicht?

Agilere Softwareentwicklung

Agile and Lean

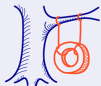
- Vorsicht Buzzword-bingo Alarm
- viele Systeme, die viel ähnlich machen aber dennoch unterschiedlich sind
- Wo Eingesetzt? Projektplanung oder im kompletten Softwareprojekt?
- nur dem Namen nach?



Agile Manifesto (<https://agilemanifesto.org>)



Individuals and
interactions



Working
software



customer
collaboration



Responding
to change

OVER

processes
and tools



comprehensive
documentation



contract
negotiation



following
a plan



Abbildung 3: Agile Manifesto on Behance -

<https://www.behance.net/gallery/46545091/Agile-Manifesto>

Kanban (Taiichi 88 [1] ; in Toyota introduced in 53)

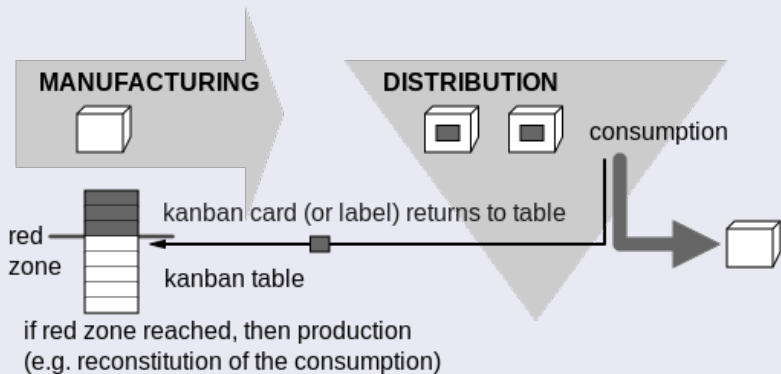


Abbildung 4: P-Kanban by Kanban_principles.jpg: Jean-Baptiste Waldner
 derivative work: Jbarta - This file was derived from: Kanban principles.jpg:, CC BY-SA 2.5,
<https://commons.wikimedia.org/w/index.php?curid=24883253>

Kanban in der agilen Projektleitung

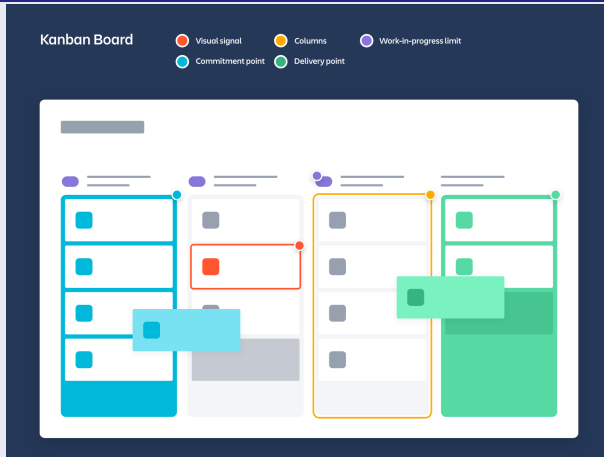


Abbildung 5: Elements of a Kanban Board – Atlassian Agile Coach

<https://www.atlassian.com/agile/kanban/boards>

Extreme Programming



Abbildung 6: Extreme Programming by DonWells – Own work based on:
XP-feedback.gif, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=27448045>

SCRUM als Software-Projektierungs/Entwicklungs-Methodik

SCRUM – The Art of Doing Twice the Work in Half the Time [2]



Abbildung 7: A Rugby Scrum –

<https://www.rugby.com.au/news/2016/12/06/01/50/kepu-spring-tour-scrum>

Teams

- **Transcendent:** Beflissenheit, Moral
- **Autonomous:** Self-Organizing and Self-Managing
- **Cross-Functional:** Brauchen alle Fähigkeiten die für das Projekt notwendig sind

- Definition of Done
- Blame is Stupid

Scrum Master

- Bulldozer des Teams, Probleme für das Team aus dem Weg räumen
- “Wie können wir das besser machen was wir tun?”
- “Was können wir an unserer Arbeitsweise ändern?”
- “Was ist unsere beste Eigenschaft? Und Warum?”

Product Owner

- Schnittstelle zum Kunden
- Vision für das Produkt, Risiken und Machbares abschätzen

Team Member

- alle Anderen
- keine Spezialrollen!
- vier Personen

Der Sprint

- grundlegende Einheit von SCRUM
- ein bis drei Wochen



Abbildung 8: by Pierre-Yves Beaudouin / Wikimedia Commons, CC BY-SA 4.0,
<https://commons.wikimedia.org/w/index.php?curid=36739114>

Daily Stand-Up

- Was hast du gestern getan?
- Was willst du heute tun?
- Was blockiert dich?
- 5 mins sharp



Abbildung 9: by Klean Denmark -
Daily Sprint Meeting, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=19838566>

cleverer Arbeiten – Vermeide Verschwendung

- Arbeite nur an einer Sache gleichzeitig!
- Halb fertig ist gar nicht fertig!
- Make it Right the First Time!
- Hart zu arbeiten, macht mehr Arbeit!
- Sei Vernünftig!

Plan
Do
Check
Act

Am Start des Sprints

The Wall

- Backlog
- Sprint
- Doing
- Done

- Transparenz
- Feedback
- Hilfe

Stories

Stories

Als Kunde, will ich Artikel durchsuchen, um die zu finden, die mir gefallen!

- Wer?
- Warum?
- Zu welchem Zweck?

wo kommen die her?

- Product Owner bringt die Stories vom Kunden mit
- einige auch vom Team (Bugs, Features)
- werden im Backlog vom Product Owner und SCRUM Master priorisiert
- und vom Team akzeptiert (im Sprint Planning)

Sprint Planning – Velocity

- **Storyvalue** ist der Wert der Stories der beim Sprint Planning vom Team mit Planning Poker festgelegt
- **Velocity** ist die Menge an Stories die das Team im Sprint schafft
- Die Velocity dient als Wert zur Auswahl der Anzahl der Stories für den jeweiligen Sprint

Velocity

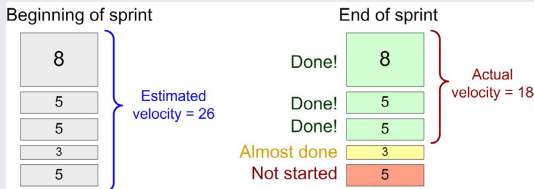


Abbildung 10: <https://pacroy.github.io/scrum/>

Rituale während des Sprints:

- Daily Sprint Meetings und
- Backlog Refinement
 - Product Owner und SCRUM Master
 - ordnen die Stories
 - erzeugen feingranulare Stories wenn nötig
 - nur so viel tun wie nötig!

... Am Ende des Sprints

Sprint Review

- dem Kunde wird das Produkt und der Fortschritt vorgeführt
- eine lauffähige Version sollte nach jedem Sprint existieren
- jeder Entwickler stellt seine Arbeit vor (nicht nur für den Kunden auch fürs Team)

Retrospektive

- was hat funktioniert
- was sollte besser gemacht werden
- velocity
- Kaizen Item

Anwendungen in der Bildung

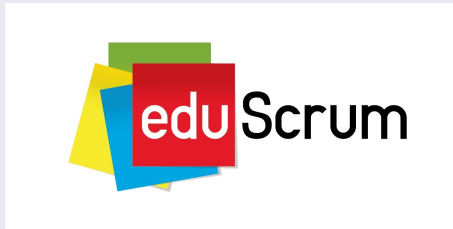
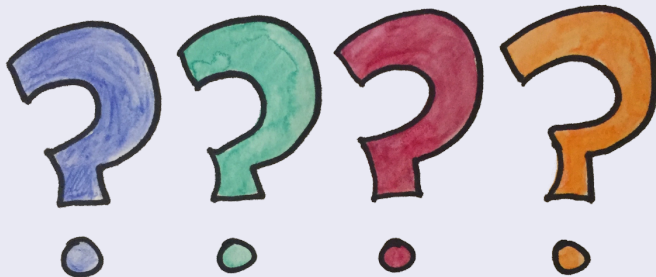


Abbildung 11: <https://eduscrum.org>



Abbildung 12: Agile Schule,
Brichzin et al [3]

Questions?



Reference

- [1] O. Taiichi, *Toyota Production System: Beyond Large-Scale Production*. Cambridge, MA: Productivity Press, 1988.
- [2] J. Sutherland, *SCRUM The Art of Doing Twice the Work in Half the Time*. Random House Business Books, London, 2015.
- [3] P. Brichzin, *Agile Schule, Methoden für den Projektunterricht in der Informatik und darüber hinaus*. hep verlag ag, Bern, 2019.