

**OST**

Ostschweizer  
Fachhochschule

# **Agiles Vorgehen**

## **oder: Wie in der Umsetzungsphase gearbeitet wird**

### **Digitalisieren Hands-on**

## **Summer School & Umsetzungsmodul**

Jörg Bachmann, Uwe Riss, Danilo Just, Sebastian Müller, Christoph Baumgarten

OST

## Aufbau

- Motivation – Warum verwenden wir agile Methoden
- Der agile SCRUM Prozess
- Meetings in SCRUM
- Rollen in SCRUM
- Sprints takten
- Arbeiten mit dem SCRUM Board
- Modulspezifisch:
  - Das Kickoff Meeting vorbereiten
  - Einen Review vorbereiten
  - Das Lernjournal verwenden

# Leitlinien der agilen Softwareentwicklung

Folgende **vier Werte** für das agile Vorgehen sind im [Manifest für agile Softwareentwicklung](#) beschrieben:

- **Individuen und Interaktionen** sind wichtiger als Prozessen und Werkzeugen
- **Funktionsfähige Produkte** sind wichtiger als umfangreicher Dokumentation
- **Zusammenarbeit mit dem Kunden** ist wichtiger als Vertragsverhandlungen
- **Das Reagieren auf Änderungen** ist wichtiger als das strikte Einhalten eines Plans

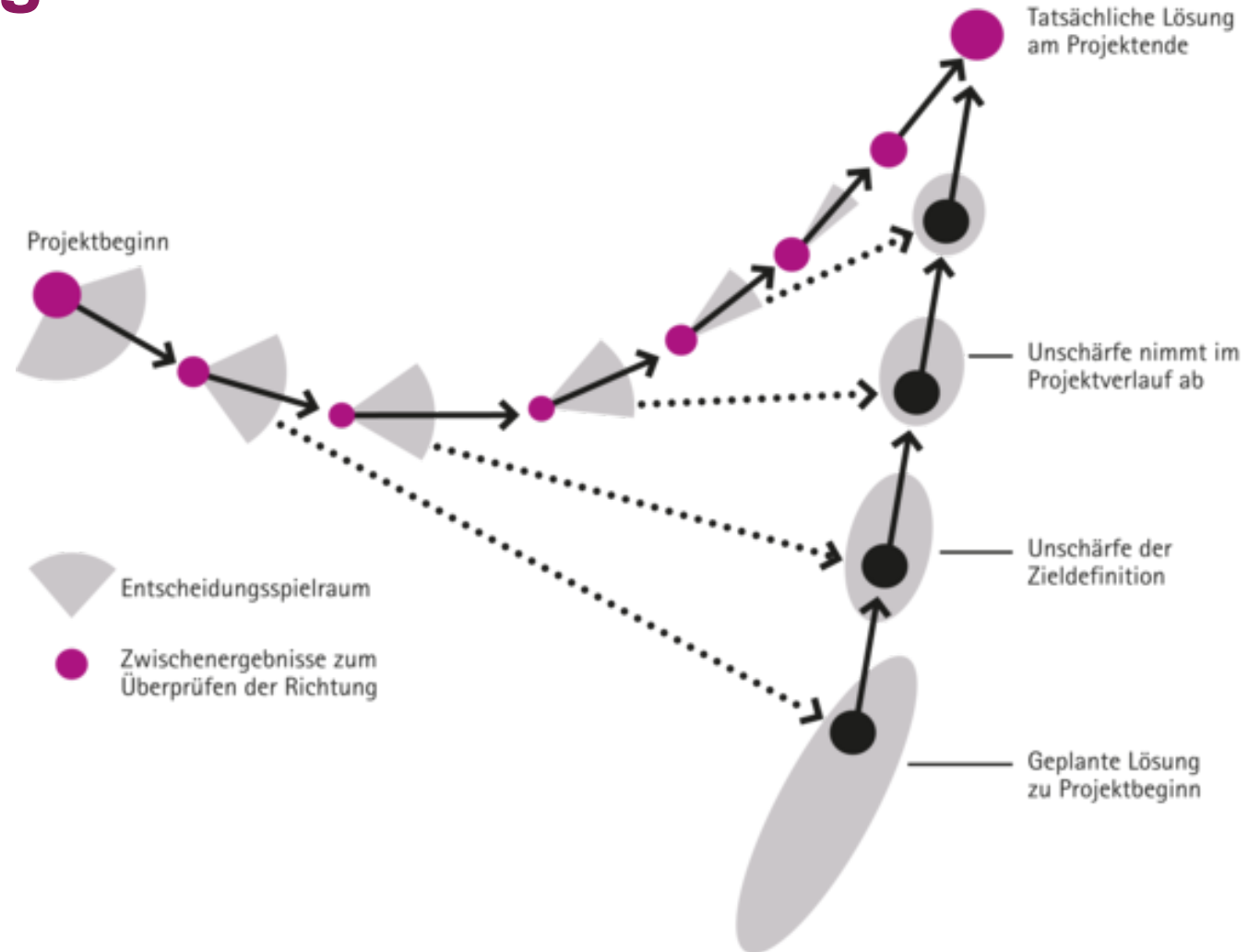
Das bedeutet allerdings nicht, dass traditionelle Qualitätskriterien, wie sie sich im Vorhandensein von Prozessen, Dokumentation, Verträgen und Plänen widerspiegeln, im agilen Projektmanagement keine Rolle spielen würden.

Der eigentliche Unterschied liegt im Denkansatz.

## Motivation: Warum agil-iterativ?

Ursprünglich vorgenommene Ziele ändern sich während der Projektlaufzeit, neue Anforderungen werden definiert, alte geändert.

Der Entscheidungsspielraum nimmt mit Fortschreiten des Projekts immer mehr ab



[[iteratec](#)]

# Prinzipien agiler Methoden

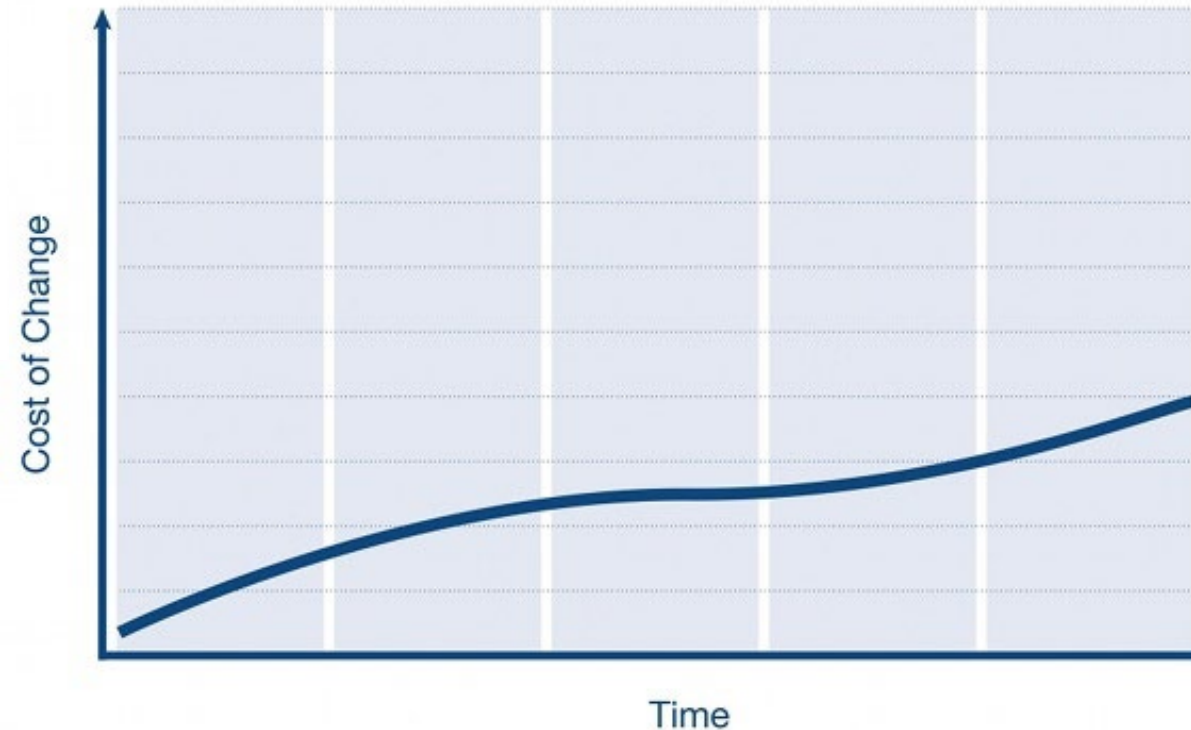
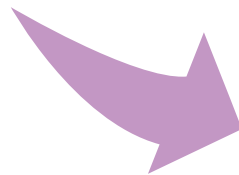
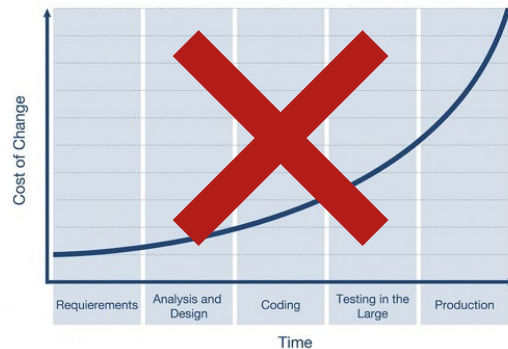
Agile Methoden sind als Antwort auf die Unzulänglichkeiten des Phasenmodells (Wasserfall) entwickelt worden:

- Grundlegende Funktionalität wird, ausgehend von ersten Anforderungen, **schnell umgesetzt**
- Fokus liegt auf der Wertschöpfung, d.h., es wird nur das umgesetzt, was einen klaren **Nutzen** beweist
- Das **Feedback** der Anwender fließt in künftige Entwicklungen ein
- Konsequenterweise **iterative** Vorgehensweise statt ausführlicher (Vor-)Planung
- Umsetzungsteams arbeiten **selbstorganisiert**

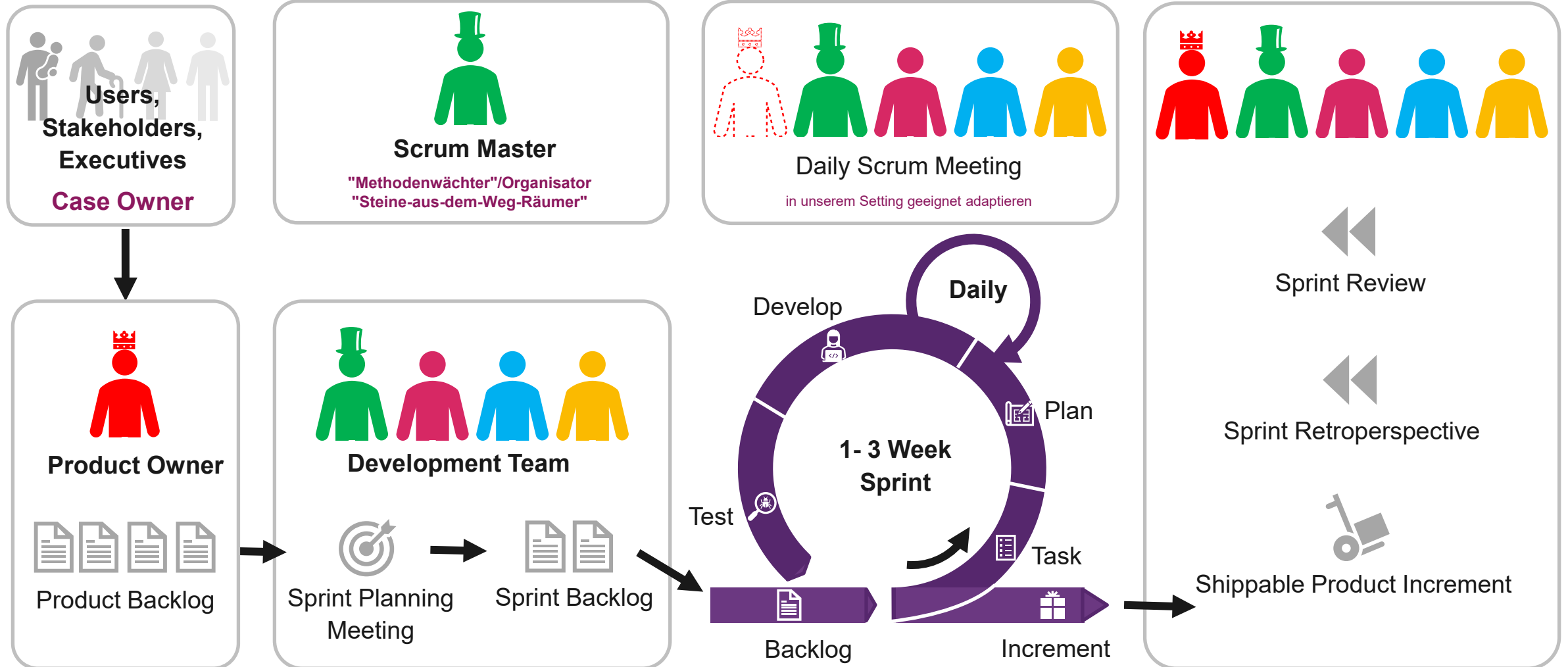
# Vorteile agiler Methoden

Durch eine frühe Inbetriebnahme des (halbfertigen) Produkts werden Fehlentscheidungen schnell aufgedeckt und sind korrigierbar («fail fast»).

Kosten für die Fehlerbereinigung sind deutlich geringer



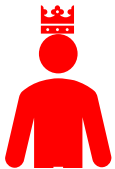
## Methoden – Agile Arbeitsweise (Scrum)



## Die Rollen im Lernlabor «Digitalisieren Hands-on»



**Case Owner / CO** : Der Dozent in den Rollen **Kunde** und **Begleitdozierender**



**Product Owner / PO**: Der Student, welcher die Schnittstelle zum Kunden übernimmt und das Product Backlog pflegt



**Scrum Master / SM**: Der Student, welcher die Sprints und Meetings plant und das Sprint Backlog pflegt

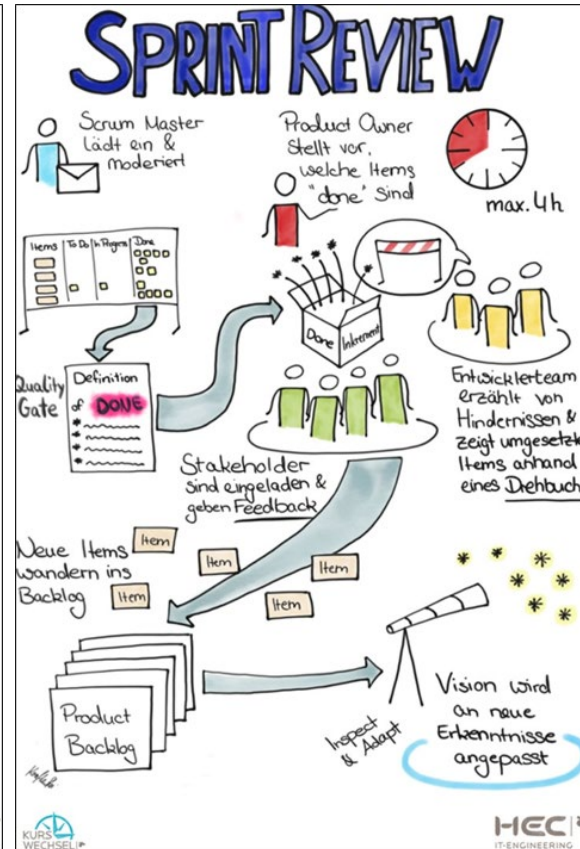
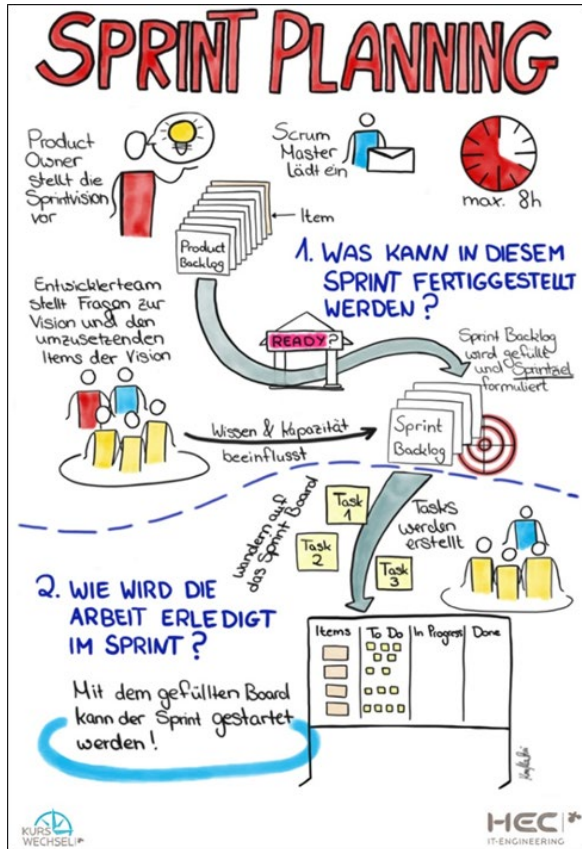


**Development Team / SCRUM Team**: Entwickeln die Projektinhalte und arbeiten an der Umsetzung

Im SCRUM Team gibt es – neben PO und SM – typischerweise (nicht zwingend – durchs Team abschliessend festzulegen) diese Rollen:

- **User Interface Designer / User Experience Designer / UX Designer**
- **Developer / Software Entwickler**
- **Qualitätssicherer / Testverantwortlicher / Tester**
- **Softwarearchitekt** (mit Fokus auf die im Team erarbeitete konkrete Lösung)

## Methoden – Agile Arbeitsweise



## Die Sprint Planung/Taktung festlegen

Variante 1													
KW38	KW39	KW40	KW41	KW42	KW43	KW44	KW45	KW46	KW47	KW48	KW49	KW50	KW51
KickOff	Sprint	Sprint	Sprint	Sprint	Sprint	Sprint	Sprint	Sprint	Sprint	Sprint	Sprint	Präsentation	Abgabe
	Backlog												
Variante 2													
KW38	KW39	KW40	KW41	KW42	KW43	KW44	KW45	KW46	KW47	KW48	KW49	KW50	KW51
KickOff	Sprint		Sprint		Sprint		Sprint		Sprint		Sprint	Präsentation	Abgabe
	Backlog												
Variante 3													
KW38	KW39	KW40	KW41	KW42	KW43	KW44	KW45	KW46	KW47	KW48	KW49	KW50	KW51
KickOff	Sprint			Sprint			Sprint			Sprint	Präsentation	Abgabe	
	Backlog												

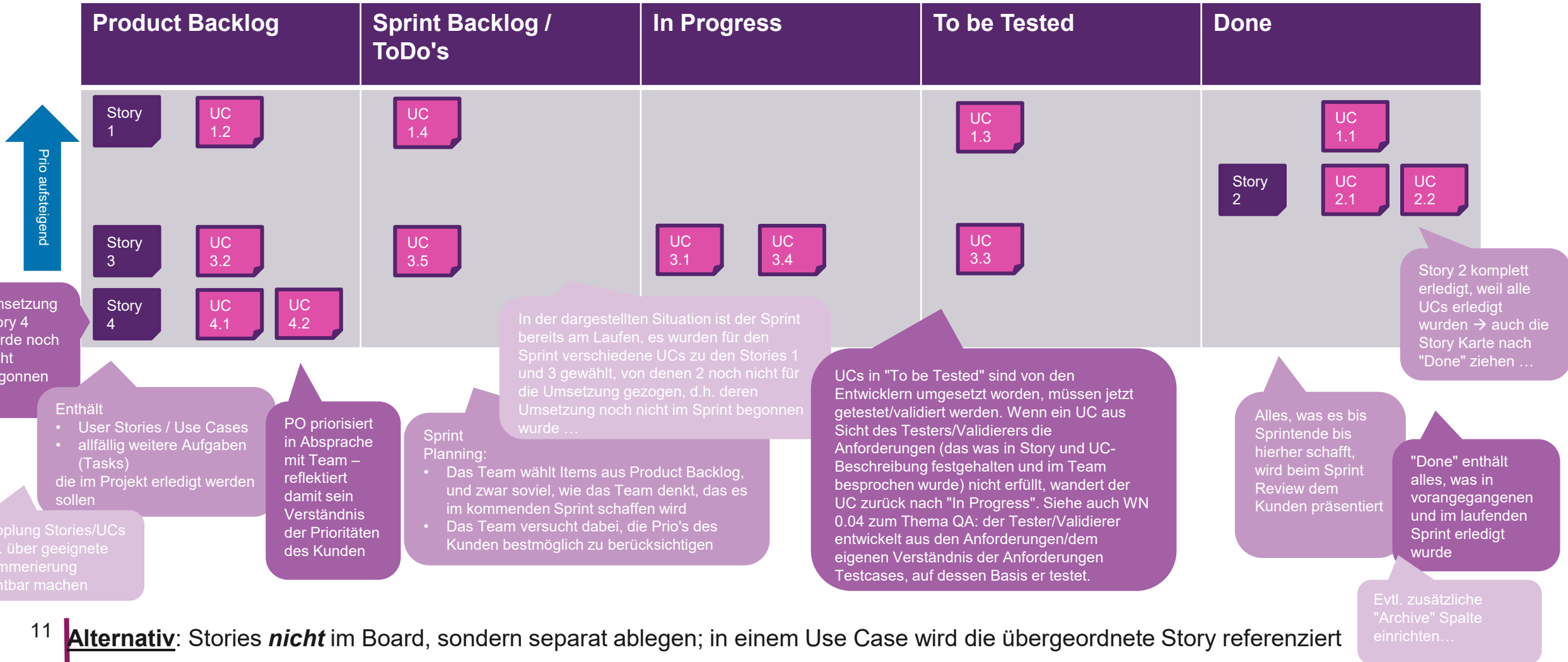
Normalfall für unser Setting: Variante 2

Achtung

## Verwendung des SCRUM Boards

Z.B. mit Hilfe des O365 Planner Tools ...

- Story = User Story
- UC = Use Case



# Vorbereitung Kickoff durch das Team

- Bereinigt euren Backlog, im Lead der Product Owner. In der Backlog Spalte sollten User Stories/Use Cases und evtl. weitere Tasks, die ihr aktuell als notwendig erachtet, enthalten sein. Die Use Cases sollten mit den übergeordneten User Stories verknüpft sein.
- Die Use Cases/Tasks sollten nach Priorität sortiert sein. Die Reihenfolge reflektiert also, was der Kunde nach eurem Verständnis als prioritär erachtet. User Stories / Use Cases allenfalls noch weiter konkretisieren, wenn noch zu wenig konkret und allenfalls aufteilen, wenn zu gross: Ein Use Case sollte innerhalb eines Sprints umgesetzt werden können.
  - Allenfalls auch einen Use Case zunächst vereinfachen und zu einem späteren Zeitpunkt mit dem eigentlichen Use Case "überspielen" ...
- Dann wählt ihr die Use Cases/Tasks aus, von denen ihr denkt, das sie im ersten Sprint angegangen werden sollten und können. Auswählen heisst: ihr bringt die Use Cases/Tasks von der Product Backlog in die Sprint Backlog Spalte. Dabei müsst ihr nicht zwingend in der Reihenfolge der Prioritäten auswählen – ihr könnt in eure Auswahl auch die Frage berücksichtigen 'Welche Use Cases/Tasks kann man gut en bloc erledigen, und wie sorgen wir dafür, dass alle Team Members ungefähr gleich ausgelastet werden?'
- Wie lange dauert ein Sprint? Hier solltet ihr euch festgelegt haben.
- Das Scrum Board sollte nur Karten in der "Backlog" Spalte haben.
- Fixiert eure initiale Rollenverteilung: Product Owner, SCRUM Master, Userinterface-Designer, Entwickler, Tester/Qualitätssicherer und evtl. weitere, die ihr für nötig erachtet.
- Die "Techies" unter euch haben bestimmt schon eine grobe Komponenten-Architektur im Hinterkopf, die reflektiert, wie die technische Lösung aufgebaut werden könnte. Den aktuellen Stand dieser Überlegungen aufmalen (eine Seite), und dann dem Team (gerade den "Nicht-Techies" unter euch!) erklären. Dieses Bild kann Teil eurer Lösungsdokumentation werden, es wird sich über die Zeit weiterentwickeln.
- Was erwartet euch im Kickoff Meeting?
  - Der Dozierende betritt das Kickoff Meeting in der Rolle des Kunden. Holt den Kunden ab: Übernimmt die Moderation des Kickoffs (könnte gut der SCRUM Master machen).
  - Ziel ist, dass ihr Klarheit darüber erhaltet, was tatsächlich im ersten Sprint gemacht wird. D.h. steigt mit der Produktvision ein (sollte die sich eurer Meinung nach noch etwas anpassen?), präsentiert sodann euren Sprint Backlog Vorschlag. Erklärt, warum ihr zu dieser Auswahl gekommen seid. Der Kunde wird euch Feedback geben, allenfalls adaptiert ihr noch ein bisschen eure Sprint Backlog Spalte.
  - Hört dem Kunden gut zu – der Product Owner kann aus der Diskussion mit dem Kunden oft schon weitere User Stories/UseCases/Tasks ableiten.
  - Der Kunde wird später in die Rolle des Begleitdozierenden wechseln und dann auch die Moderation des Meetings übernehmen. Er wird euch u.a. nochmal nach eurer Rollenverteilung und dem geplanten Zusammenarbeitsmodus fragen und mit euch die grobe Komponenten-Architektur besprechen.

# Agendavorschlag für einen Kickoff

- Case Owner als "Kunde" / Moderation durch Studierendenteam
  - Rekapitulation der Vision
  - Review Product Backlog, Planen des Sprint Backlogs zum ersten Sprint
    - Welche User Stories/Use Cases/Tasks im Product Backlog, welche im Sprint Backlog?
    - User Stories/Use Cases/Tasks priorisieren, ergänzen
    - Sprintplan: Sprint Takt abschliessend fixieren
- Case Owner als "Begleitdozierender" / Moderation durch Begleitdozierenden
  - ← *diesen Teil würde ein Kunde mit einem SCRUM Team normalerweise nicht diskutieren*
  - "Erläutert mir eure grobe Komponenten-Architektur im aktuellen Stand"
    - Welche Wissensnuggets könnten euch evtl. helfen?
  - Wie organisiert sich das Team während des nächsten Sprints; gibt es Anpassungen, die noch nicht diskutiert wurden?
    - Mit welcher Rollenverteilung wird in den nächsten Sprint gestartet (kann von Sprint zu Sprint wechseln)

# Ziel des Reviews im Hinblick auf die Rolle «Kunde»

- Vor dem Review sollte das Team diskutieren, ob die angedachte Lösung (Produkt) in der geplanten Form umsetzbar und zielführend ist.
  - Das Team sollte sich folgende Fragen stellen:
    - Haben wir noch genug Zeit und die geeigneten Mittel, um die bisher geplante Lösung zu erreichen?
    - Wenn dies nicht der Fall ist, welche Anpassungen sind notwendig?
    - Wie können wir dem «Kunden» diese Anpassungen erklären?
- Im Review hat das Team gegenüber dem Case Owner als «Kunde» folgendes Ziel:
  - Team möchte
    - dem «Kunden» ein Bild vom aktuellen Stand der Arbeit zeigen
    - dem «Kunden» notwendige/sinnvolle Anpassungen am Ziel erklären
    - Einigungen mit dem «Kunden» über solche Anpassungen erreichen (und diese festhalten)
  - Die Einigung ist Grundlage für die weitere Arbeit des Teams. Hierdurch soll verhindert werden, dass sich erst am Ende des Projekts herausstellt, dass die ursprünglich geplante Lösung nicht umsetzbar ist.
  - Dabei kann es durch die begrenzten Mittel/Zeit auch zu Vereinfachungen der Lösung kommen.
  - Am Ende muss die Lösung nicht der ursprünglichen Lösungsidee (aus der Blockwoche) entsprechen, aber dem, was das Team mit dem «Kunden» in den Reviews ausgehandelt hat.

# Agendavorschlag für den Review

- Team tritt möglichst als Ganzes auf, teilt sich Präsentationsabschnitte
- Vom Produkt wird das gezeigt, was es im Sprint bis in die "Done" Spalte geschafft hat
- Case Owner als "Kunde" / Moderation durch Studierendenteam (z.B. Scrum Master)
  - Team
    - rekapituliert nochmal die Anforderungen, die im letzten Sprint angegangen wurden und
    - präsentiert das Produkt bzw. die Produkt-Features, die im letzten Sprint dazugekommen sind
    - Schlägt vor, was im nächsten Sprint angegangen werden soll, hat dafür für sich schon mal den Sprint Backlog neu aufgefüllt
  - Kunde gibt Feedback, der Product Owner hört besonders gut zu und wird daraus weitere User Stories/UseCases/Tasks ableiten, die in den Product Backlog und u.U. auch schon in den Sprint Backlog für den nächsten Sprint einfließen
- Case Owner als "Begleitdozierender" / Moderation durch Begleitdozierenden
  - Team präsentiert den Outcome des zu Sprintende teamintern durchgeführten Sprint Retrospective Meetings
    - Was lief gut?
    - Wo haben wir Luft nach oben?
    - Was wollen wir künftig anders machen? Welche Massnahmen haben wir abgeleitet?Der Begleitdozierende wird hier Fragen stellen und Feedback geben
  - Weitere Fragen des Case Owners
    - "Wie habt ihr im Produkt ... aufgebaut?" → Komponentenarchitektur revisited ... (sofern es hier Neuigkeiten gibt)
    - Wie wurde getestet? Was kam beim Testen raus?
    - "Wird sich eure Rollenverteilung anpassen?"
    - "Erreicht das Team weiterhin eine einigermaßen gleichmäßige Auslastung der einzelnen Team Members? Was könnte man allenfalls tun?"
    - "Welche Wissensnuggets könnten euch evtl. für die nächsten Schritte helfen?"
- *Nach dem Meeting aktualisiert Product Owner / das Team Product und Sprint Backlog, der nächste Sprint wird sodann gestartet*