

# Projekt Management Basic

Fabian Gröger

24.05.2018

## Inhaltsverzeichnis

<b>Projekt Management Basic</b>	<b>2</b>
1 Projekt Management Grundlagen . . . . .	2
1.1 Lernziele . . . . .	2
1.2 Grundbegriffe . . . . .	2
1.3 Projekterfolg und -misserfolg . . . . .	3
1.4 Vorgehensmodelle . . . . .	4
1.5 Projektorganisationsformen . . . . .	14
1.6 SODA . . . . .	18
2 Projektplanung . . . . .	19
2.1 Lernziele . . . . .	19
2.2 Projektziele . . . . .	19
2.3 Projektresultate . . . . .	20
2.4 Arbeitspakete . . . . .	20
2.5 Meilensteine . . . . .	22
2.6 Scope . . . . .	23
2.7 Anforderungen . . . . .	23
3 Projektplanung Klassisch . . . . .	24
3.1 Projektstandsitzung . . . . .	24
3.2 Projektstrukturplan . . . . .	24
3.3 Ablaufplan . . . . .	24
3.4 Aufwandschätzung . . . . .	26
3.5 Ressourcenplanung . . . . .	27
3.6 Projektplanung Hybrid . . . . .	28
4 Projektkommunikation . . . . .	39
4.1 Projektsitzung . . . . .	39
4.2 Projektbericht . . . . .	39
4.3 Kommunikation mit Stakeholder . . . . .	40
5 Risiko Management . . . . .	40
5.1 Risiken . . . . .	40
5.2 Risiko Identifizieren . . . . .	41
5.3 Risiko analysieren . . . . .	41
5.4 Risiko priorisieren . . . . .	42

5.5 Massnahmen erarbeiten . . . . .	42
5.6 Risiko überwachen . . . . .	42
5.7 Zusammenfassung . . . . .	44
6 Projektcontrolling . . . . .	44
6.1 Lernziele . . . . .	44
6.2 Klassische Projektleitung . . . . .	44
6.2 Agile Projektleitung . . . . .	46
7 Business Case . . . . .	48

---

# Projekt Management Basic

## 1 Projekt Management Grundlagen

### 1.1 Lernziele

1. LZ1: Sie kennen die Grundbegriffe im Projektmanagement und können Projekte von anderen Unterfangen unterscheiden.
2. LZ2: Sie kennen wesentliche Faktoren die zum Projekterfolg beitragen und wissen, aus welchen Gründen Projekte scheitern.
3. LZ3: Sie kennen die aktuellen Kategorien der Vorgehensmodelle und können deren wesentliche Elemente und die Unterschiede erklären.
4. LZ4: Sie kennen die verschiedenen Projektorganisationsformen und typische Rollen und Gremien der Projektinstitution.
5. LZ5: Sie kennen die Grundzüge des SoDa Vorgehensmodells.

### 1.2 Grundbegriffe

LZ1: Sie kennen die Grundbegriffe im Projektmanagement und können Projekte von anderen Unterfangen unterscheiden.

#### 1.2.1 Projekt

##### Was ist ein Projekt?

Projekte sind in sich abgegrenzte, komplexe und / oder komplizierte Aufträge, deren Erfüllung eine Organisation bedingt, die für die Umsetzung der Tätigkeiten eine Projektmethode anwendet, mit der alle anfallenden Arbeiten geplant, gesteuert, durchgeführt und kontrolliert werden können.

##### Was sind Merkmale eines Projektes?

- Begrenzte Ressourcen (Zeit, Aufwand, Leistung)
- Einmalig

- Begrenzte Dauer
- Komplexe Aufgabenstellung
- Bedarf einer Organisation (Projektmethode anwenden)
- Sollte messbare Resultate haben

### 1.2.2 Komplex vs. Kompliziert

**Komplexe Probleme** müssen durch probieren, erkennen und reagieren, iterativ gelöst werden.

**Komplizierte Probleme** können durch erkennen, analysieren und reagieren, linear gelöst werden.

### 1.2.3 Teufelsquadrat

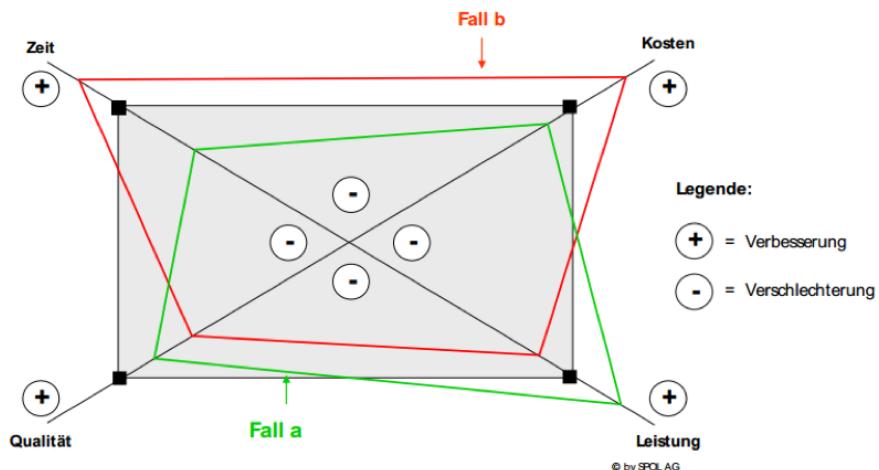


Abbildung 1: Teufelsquadrat

### 1.2.4 Projektmanagement

Das Projekt führen, koordinieren, steuern und kontrollieren.

## 1.3 Projekterfolg und -misserfolg

LZ2: Sie kennen wesentliche Faktoren die zum Projekterfolg beitragen und wissen, aus welchen Gründen Projekte scheitern.

Wie äussert sich ein Projekterfolg und -misserfolg?

**Projekterfolg** wird gleichgesetzt mit dem Erreichen des vorgegebenen Ziels/Produkts, ohne den festgelegten Zeit- und Kostenrahmen überschritten zu haben.

**Projekterfolg** kann auch gegeben sein, wenn aus dem Projekt einen Nutzen gezogen werden kann, unabhängig davon ob sich das Projekt verspätet oder teurer wird. Jedoch wird der Nutzen (nach Abschluss des Projektes) eher selten oder fast gar nicht gemessen.

**Projektmisserfolg** ist in Umkehr dieser Aussage, das Überschreiten von Zeit oder Kosten oder die Nichterfüllung der Produktfunktionalität.

### Welche Schlüsselfaktoren tragen zum Projekterfolg bei?

- Unterstützung von der GL (Finanziell oder Motivation)
- Emotionale Reife (man muss führen können, man muss geduldig sein und die Leute mögen)
- Benutzer involvieren (Zukünftige Benutzer involvieren)

### Aus welchen Gründen scheitern Projekte?

- Fehlende Erfahrung in Projektleitung
- Unzureichende Projektplanung
- Fehlende Projekt Methodiken \*Fehlendes Projekt Controlling

### Was ist das Magische Dreieck?

Um bei einer Aufgabe, die Erwartungen an ein Projekt so weit wie möglich zu erfüllen, bewegt man sich zwischen drei Größen:

- **Zeit:** Projektdauer und Termine
- **Kosten:** Ressourcen (Budget, Personal)
- **Leistung:** Inhalt (Umfang und Qualität)

Diese drei Größen stehen in Zielkonkurrenz zueinander, eine Änderung an einer Größe führt zu Änderungen an einer oder beiden anderen Größen.

## 1.4 Vorgehensmodelle

LZ3: Sie kennen die aktuellen Kategorien der Vorgehensmodelle und können deren wesentliche Elemente und die Unterschiede erklären

Die Projektführung hat in fast allen Projekten die gleichen Aufgaben aber mit unterschiedlichen Ausprägungen. Bei der Projektdurchführung gibt es je nach Projektart oder Vorgehensmodell Unterschiede.

Unter einem Vorgehensmodell versteht man eine projektübergreifende Regelung, wie Aktivitäten oder Ergebnisse umgesetzt bzw. bearbeitet werden.

**Nenne die Vorgehensmodelle des klassischen Projekts-Managements inkl. deren Eigenheiten**

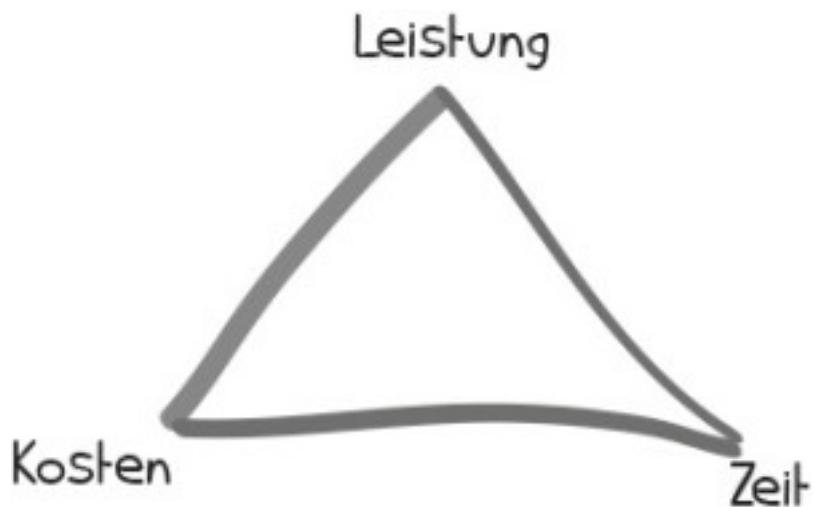


Abbildung 2: Magisches Dreieck

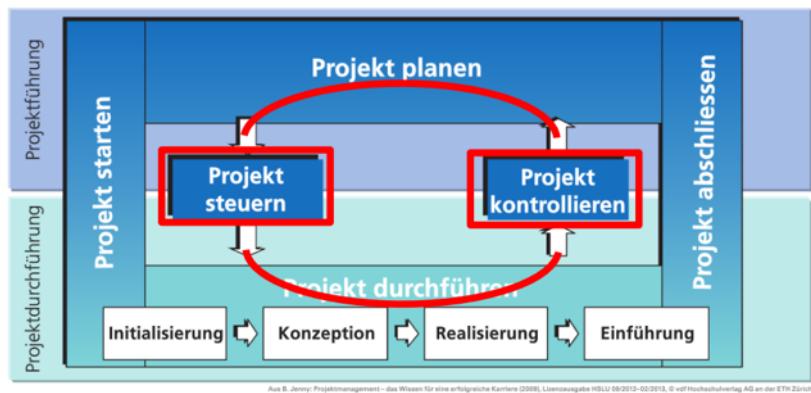


Abbildung 3: Projektführung und Projektdurchführung

- Wasserfallmodell
  - Sehr starres Vorgehen
  - Wenig Flexibilität
  - Bei geänderten Anforderungen müssen diese in einen komplett neuen Zyklus integriert werden
- V-Modell
  - Starres Vorgehen
  - Wird beim Abnahmetest festgestellt, dass die Anforderungen unzureichend spezifiziert wurden, kann dies zu einem Projektabbruch führen
  - Anpassungen der Anforderungen verursacht hohe Kosten

#### 1.4.1 Klassisch

##### Wasserfall Modell

- Initialisierung
- Analyse
- Entwurf (Design)
- Realisierung
- Einführung
- Nutzung

Treten neue oder geänderte Anforderungen an die Software auf, müssen diese in einem komplett neuen Zyklus umgesetzt werden. Dies ist ein sehr starres Vorgehen, welches wenig Flexibilität bietet. Kritisch wird dieses Vorgehen, wenn in einer Phase erkannt wird, dass die vorherige Phase unvollständige Ergebnisse geliefert hat.

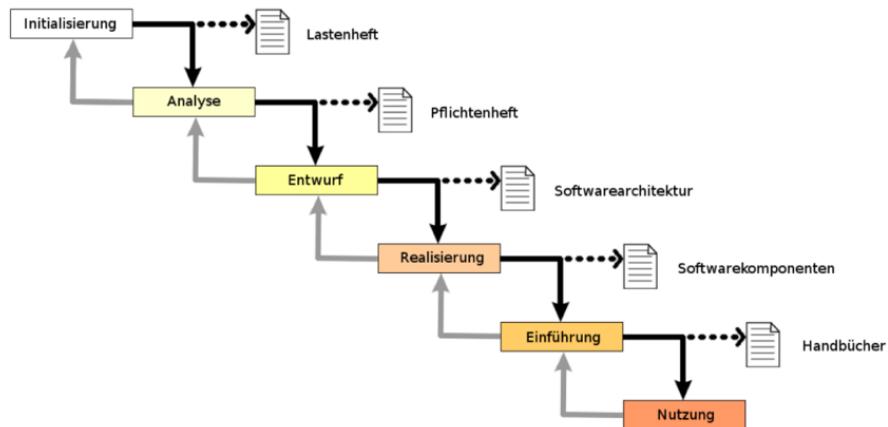


Abbildung 4: Wasserfall Modell

##### V-Modell

Die Phasen laufen von links oben nach unten und dann von rechts unten nach rechts oben. Zu jeder Phase auf der Linken-Seite gibt es die entsprechende Test-Seite auf der rechten Seite.

Wird allerdings in Phase 6 bei der Abnahmeprüfung festgestellt, dass die Anforderungen in Phase 1 unzureichend verstanden wurden, dürfte dies in den meisten Fällen zum Projektabbruch und damit zum Scheitern des Projektes führen. Es ist dabei zu beachten, dass in großen Projekten zwischen Analyse (Phase 1) und Systemabnahme (Phase 6) durchaus mehrere Jahre liegen können.

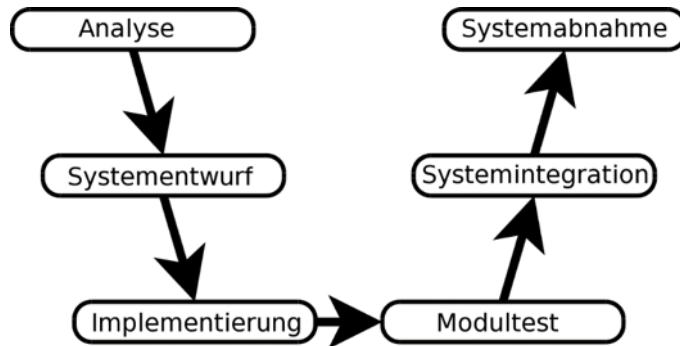


Abbildung 5: V-Modell

### Nachteile

Als Nachteil eines V-Modells oder eines Wasserfallmodells, ist die mangelnde Flexibilität anzusehen. Auf ändernde Anforderungen kann nur schlecht reagiert werden, was hohe Kosten verursacht.

#### 1.4.2 Agil

##### Was sind wesentliche Elemente von einer agilen Software Entwicklung?

- Agil: Neue, geänderte Anforderungen können eingebunden werden
- Iterativ: Liefert Produkte in regelmässigen Abständen (potenziell lieferbare Software)
- Inkrementell: Software wird stufenweise aufgebaut, Gelegenheiten für Feedback wird maximiert

##### Ab wann ist das agile Vorgehensmodell empfehlenswert?

- Mitarbeiter Anzahl  $\leq 10$
- komplexe Produkte

**Was ist Scrum und wie ist es aufgebaut?** Scrum ist agil, iterativ und inkrementell. Scrum stützt sich auf die folgenden drei Säulen:

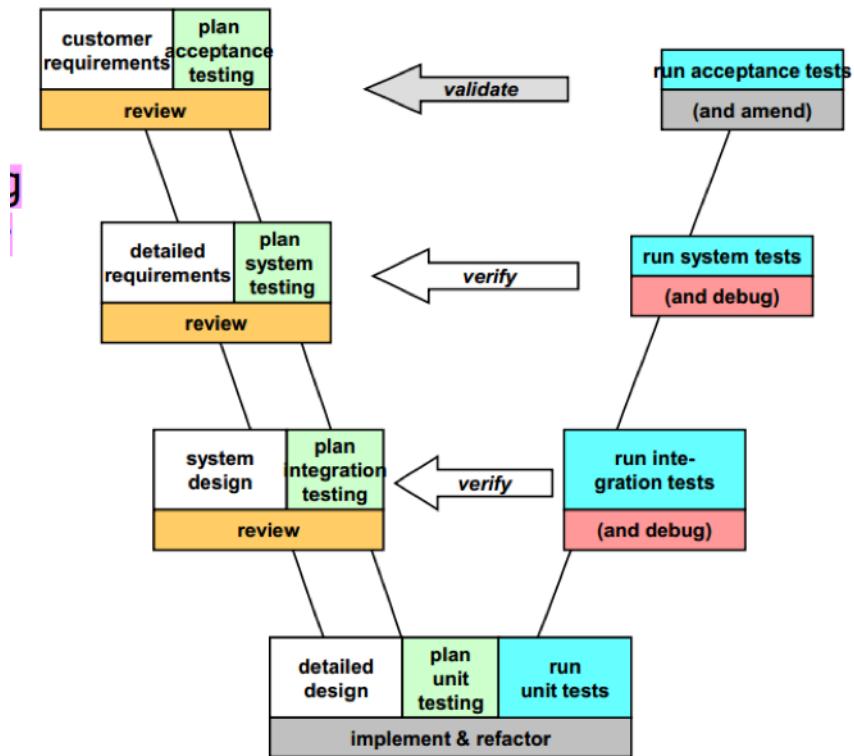


Abbildung 6: V-Modell

- Transparenz (z.B. gegenüber dem Kunden)
- Überprüfung
- Anpassung

Scrum ist vor allem für die Entwicklung komplexer Produkte gedacht.

Die Scrum Teams liefern Produkte in regelmässigen Abständen (iterativ) und stufenweise erweiternd (inkrementell). Dadurch werden auch die Gelegenheiten für Feedbacks maximiert. Die inkrementelle Lieferung der Produkte stellt sicher, dass immer eine potentielle gebrauchsfähige und nutzbare Version des Produktes zur Verfügung steht. So kann diese dem Kunden jederzeit präsentiert werden.

### **Aus welchen Elementen besteht ein Sprint?**

Der Sprint ist das Herz von Scrum. Er besteht aus:

- Sprint Planung
- Daily Meetings
- Entwicklungsarbeit
- Sprint - Abnahme
- Sprint - Retrospektive

Die Sprints haben während des Projektes eine gleichmässige, feste Dauer (time Boxing). Während eines Sprints wird ein fertiges, getestetes, nutzbares und potentiell auslieferbares Produkt inkrementell entwickelt.

### **Was ist das Produkte Backlog?**

Die Produkte Backlog Verwaltung enthält alle Anforderungen in einer geordneten und zielführenden Fertigstellungsreihenfolge.

Das ProductBacklog ist eine priorisierte Liste aller aktuell bekannten Epics bzw. Stories. Es ist ein lebendes Artefakt, jeder Eintrag enthält eine Beschreibung, die Priorität, die Aufwandschätzung und den Wert (Nutzen). Die Detailplanung erfolgt rollend im SprintBacklog!

Das Produkte Backlog muss laufend gepflegt werden („Grooming“). Diese Anforderung hätte im Sprint 2 abgehandelt werden sollen, aber es hat nicht mehr gereicht, also wird diese in den Sprint 3 eingeplant. Der Product Owner entscheidet über das Grooming ob und bis wann eine Anforderung im Projekt realisiert wird. Die Sprintplanung findet zu Beginn eines Sprintes statt.

Die Sprint Planung wird zusammen mit dem Entwicklungsteam verhandelt damit Zielkonflikte aufgelöst werden können.

Der Sprint-Abschluss welcher am Ende jedes Sprints stattfindet, wird ermittelt welche Funktionen akzeptiert und freigeben werden.

Software entwickelt sich von Sprint zu Sprint inkrementell. Ganzes Projekt wird in Sprints unterteilt und stellt den Arbeitszyklus dar. Kurze Iterationen, bei denen inkrementell entwickelt wird. Durch kurze Iterationen und kurze Release-Zyklen wird schnelles Feedback beim Kunden über den Fortschritt und

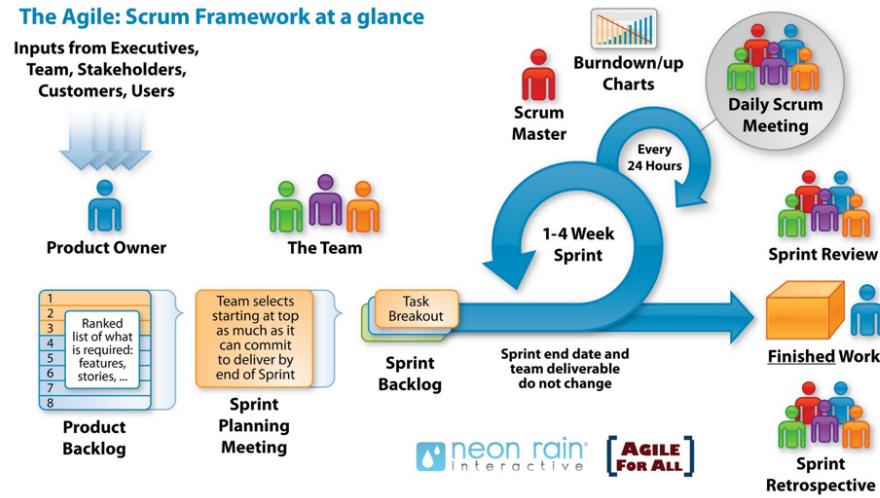


Abbildung 7: Scrum aus SLGP

Funktionsumfang eingeholt. Dieses schnelle Feedback dient zur Steuerung der weiteren Entwicklung.

#### Rollen:

- Product Owner:
  - Vertritt die Anwender des Produkts oder Stakeholder
  - Verantwortlich für das Produkt
  - Erstellt Anforderungen
  - Erstellt Release Plan für das Produkt
  - Priorisiert Anforderungen aus dem Backlog
- Scrum Master:
  - Stellt sicher, dass die Regeln und Schritte im Projekt eingehalten werden
  - Hilft bei methodischen Problemen
- Scrum Team:
  - Zwischen 5 bis 10 Mitarbeiter im Best Case

#### User Stories:

Enthält Anforderung aus Benutzersicht, z.B. als registrierter Reisende sehe ich in einer Übersicht alle meine gebuchten Reisen.

#### Product Backlog:

Quelle wo alle Anforderungen enthalten sind. Enthält:

- Anforderungen (User Storys)
- Unkritische Fehler (Kritische wandern in das Sprint Backlog)

- Forschungsaufträge

### **Sprint Backlog:**

Anforderungen welche erfüllt werden müssen um ein Sprint umzusetzen, Fehler die schnell behoben werden müssen. Die Fehler kommen aus dem Product Backlog.

### **Ticket:**

Befinden sich im Sprint, dass sind die einzelnen Anforderungen basierend auf den Story Cards, welche ausgeführt werden müssen.

### **Iterativ:**

Beschreibt ein Vorgehen in nacheinander folgenden Iterationen, ein Verfahren der schrittweisen Annäherung an die exakte oder endgültige Lösung. Eine Iteration ist eine zeitlich und fachlich in sich abgeschlossene Einheit.

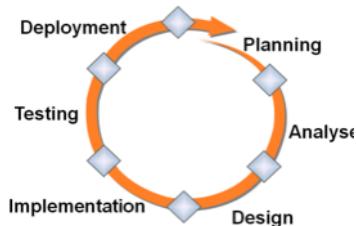


Abbildung 8: Iteratives Vorgehen

Dabei werden in jeder Iteration alle Projektphasen durchlaufen. Jede abgeschlossene Iteration liefert einen funktionierenden Zwischenstand des Produkts – einen inkrementellen Release.

### **Inkrementell:**

Inkrementell (kleine Stufe der Zunahme) bedeutet, dass mit jeder Iteration zusätzliche Funktionalität hinzugefügt wird.

### **Scrum im Überblick:**

#### 1. Anforderungen in Story Cards

Aus dieser ersten, aber anschaulichen und verständlichen Vision lassen sich die einzelnen Elemente und Merkmale des Produkts ableiten, die zu entwickeln sind. Das sind die konkreten Anforderungen und einzelne Funktionalitäten, die in sogenannten Story Cards in den Worten der Anwender formuliert werden - nicht im Jargon der Techniker. Damit wird auch sichtbar, welche Ressourcen das Projekt braucht (Qualifikationen im Projektteam und grober Aufwand für die Umsetzung).

#### 2. Arbeitspakete im Product Backlog

Aus diesen Anforderungen des Produkteigners wird ein sogenanntes Produkt-Backlog zusammengestellt. Das ist eine Sammlung sämtlicher Funktionen und Merkmale, die das Produkt haben soll. Am Anfang ist diese Zusammenstellung noch grob, doch im Projektverlauf wird sie immer genauer.

### 3. Prioritäten

Im nächsten Schritt werden Prioritäten vergeben. Welche Elemente und Funktionen sind am wichtigsten im Sinne von: Das sichert die Zufriedenheit der Anwender essenziell. Andere Anforderungen sind nicht so wichtig, können aussortiert werden, werden mit anderen zusammengelegt, sind technisch nicht realisierbar oder werden verschoben. Sie werden dann bei der Überarbeitung oder bei der Erweiterung des Produkts behandelt.

### 4. Sprint Planung

Zu jedem Beginn der Sprint-Planung erstellt der Product-Owner gemeinsam mit dem Team das neue Sprint-Ziel. Es wird geprüft, wie viele der höchst priorisierten Backlog-Items in das Sprint-Backlog übernommen werden können. So dass:

- Ein sinnvolles Sprintziel entsteht
- Mit den verfügbaren Ressourcen
- Via Timeboxing Verfahren durchgeführt werden kann.

### 5. Aufgaben im Sprint Backlog

Eine einzelne Aufgabe, die dann zu erledigen ist, wird Ticket genannt. Alle Tickets sind im sogenannten Sprint-Backlog aufgeführt. Das ist gewissermaßen der Maßnahmenplan und der Arbeitsvorrat für das Entwickler-Team für den nächsten Arbeitszyklus, der sogenannte Sprint. Jedes Teammitglied übernimmt eigenverantwortlich einzelne Ticket (Verpflichtungserklärung).

### 6. Arbeitsprozesse im Sprint

Im Sprint arbeiten die Teammitglieder an ihren Aufgaben, den Tickets. Hier werden Teillösungen gemäß dem Sprint-Backlog (Aufgabenplan) entwickelt.

### 7. Besprechungen im Scrum

Während des täglichen, 15 minütigen Treffens, dem sogenannten Scrum, berichtet jeder der Reihe nach: Was er seit dem letzten Scrum gemacht hat, was er bis zum nächsten Scrum tun wird und was ihn bei seiner Arbeit behindert. Der Scrum-Master muss diese Hindernisse aufgreifen und helfen, dass sie beseitigt werden. In einem Sprint Burndown Chart wird sichtbar gemacht, wie das Projekt voranschreitet.

### 8. Fortschrittskontrolle im Sprint Review Meeting

Jeder Sprint wird durch ein Sprint Review Meeting abgeschlossen. Das Team stellt die Ergebnisse dem Produkteigner vor. Er muss sie akzeptieren

und abnehmen. In gesonderten Treffen können auch die Teammitglieder ihre Zusammenarbeit überprüfen und festhalten, was intern verbessert werden sollte. Die Erkenntnisse sollten für zukünftige Scrum Projekte genutzt werden; so wird ein Lernprozess unterstützt.

#### 9. Projektabschluss

Sind alle Sprints durchlaufen, wird das fertige Produkt an den Produkteigner geliefert. Das Projekt kann abgeschlossen werden.

#### Kritik:

- Nur bis ca. 10 Mitarbeiter geeignet
- Oft macht es mehr Sinn die Anforderungen zu spezifizieren als User Storys niederzuschreiben

### 1.4.3 Hybrid

#### Was ist ein hybrides Vorgehen in einem Projekt?

Eine Kombination aus verschiedenen Vorgehensmodellen bezeichnet man als hybrides Vorgehensmodell. Als Beispiel Gesamtprojektsteuerung und Teilprojekte "X" und "Y" wurden nach dem klassischen Vorgehensmodell arbeiten. Das Teilprojekt "Z" wurde dagegen mit dem agilen Vorgehensmodell "SCRUM" durchgeführt.

### 1.4.4 Klassisch vs. Agil

#### Gegenüberstellung Klassisch / Agil, was ist bei wem fix und was ist bei wem variabel (Leistung und Aufwand)?

Je nach dem um welches Projekt es sich handelt muss das entsprechende Vorgehensmodell gewählt werden.

Beim klassischem Modell ist die Leistung fix. Als Beispiel: die Einführung einer Lagersoftware im Dezember. Aufgrund der erwarteten Funktionen, also welche Funktionen werden benötigt, rechnet man dann die Zeit und der Aufwand.

Beim agilen Modell ist die Zeit und der Aufwand fix. Als Beispiel: Neues Produkt, Entwicklung einer Labor Software. Die Leistung ist also variabel und es kann auch möglich sein, dass wen die Zeit und der Aufwand zu Ende geht, nicht alle Leistungen im Projekt enthalten sind.

Klassisch:

- Alle Anforderungen sind vollständig definiert und festgeschrieben
- Zeitraum ist geplant
- Aufwand muss evtl. erhöht werden um die Fertigstellung zu garantieren

Hybrid / Agiles:

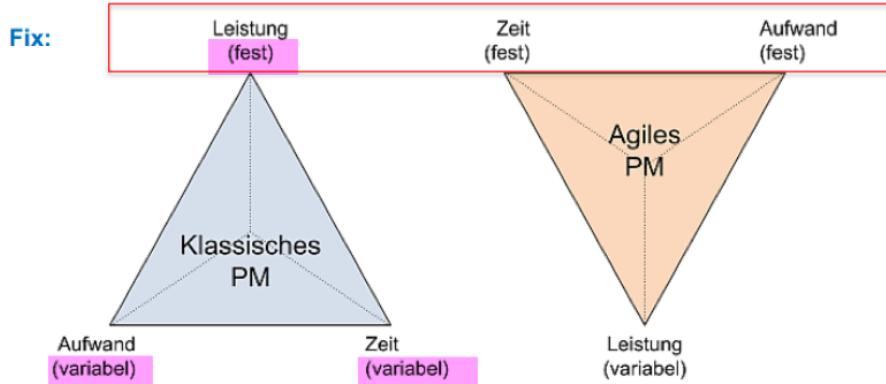


Abbildung 9: Klassisch vs. Agil

- Wichtigsten Anforderungen sind grob beschrieben und priorisiert, vollständige Umsetzung ist jedoch nicht festgeschrieben
- Anforderungen werden laufend in den Sprints konkretisiert und umgesetzt, so entsteht praktisch nach jedem Sprint eine lauffähige Software
- Aufwand ist festgelegt und wird gesteuert

## 1.5 Projektorganisationsformen

LZ4: Sie kennen die verschiedenen Projektorganisationsformen und typische Rollen und Gremien der Projektinstitution

### Welche Projektorganisationen gibt es?

- Reine Projektorganisation
- Matrixprojektorganisation
- Stablinien Projektorganisation

Es wird also zusätzlich zum Organigramm eine Projektorganisation gebildet, die die Zusammenstellung der Rollen, Befugnisse und Schnittstellen eines Projektes beschreibt. Und damit ist die Projektorganisation für den Projekterfolg von zentraler Bedeutung.

Denn nur, wenn alle beteiligten Personen wissen, welche Aufgaben, Verantwortungen und Befugnisse sie innerhalb eines Projektes haben (und wie sich diese zur Stammorganisation verhalten), können sie sich auch dementsprechend (richtig) verhalten.

**Eine Projektorganisation** enthält Aufbau- und Ablauforganisation eines bestimmten Projektes. Problem einer Projektorganisation: Es ist meist ungenau oder gar nicht definiert welche Verantwortung (finanziell, fachlich, administrative) ein Projektleiter hat.

### **1.5.1 Reine Projektorganisation oder Linien Projektorganisation**

#### **Definition:**

Alle Projektmitarbeiter werden aus ihren jeweiligen Abteilungen abgezogen und vollständig der Projektorganisation zugeordnet.



Abbildung 10: Reine Projektorganisation

#### **Vorteile:**

- Klare Befugnisse, Verantwortungen
- Einfache Kommunikationswege

#### **Nachteile:**

- Schwächung der Stammorganisation, da die Mitarbeiter komplett in die Projekte eingegliedert werden
- Leerlaufzeiten, es können im Projekt einzelne Mitarbeiter zu wenig oder zu viel ausgelastet sein

#### **Projektleiter Aufgaben:**

- Fachliche Weisungsbefugnis
- Disziplinarische Weisungsbefugnis
- Volle Zeilverantwortung

### **1.5.2 Matrix Projektorganisation**

#### **Definition:**

Hier bleiben die Mitarbeiter in ihren jeweiligen Abteilungen und die Weisungsbefugnisse werden zwischen Stamm- und Projektorganisation aufgeteilt. Jeder Abteilungsleiter behält die disziplinarische Weisungsbefugnis über seine Mitarbeiter. Der PL hat bis zu einem bestimmten Prozentsatz Zugriff auf die Mitarbeiter. Muss aber auf das Tagesgeschäft innerhalb den Abteilungen schauen. Dadurch

ist der Abstimmungs-/Kommunikationsbedarf zwischen Projektleiter und Abteilungsleiter, aber auch mit den Mitarbeitern deutlich höher als bei der autonomen Projektorganisation.

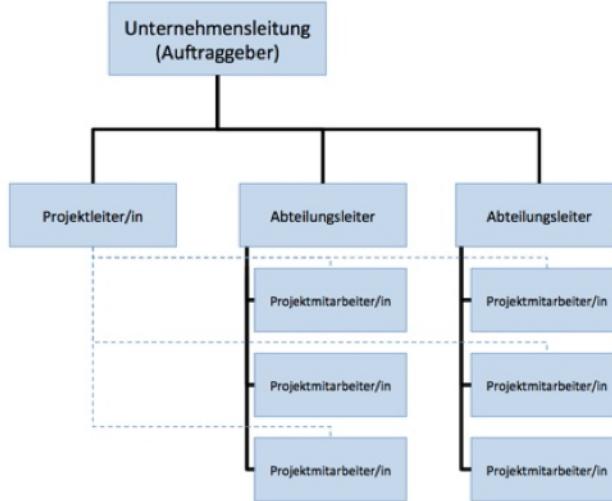


Abbildung 11: Matrix Projektorganisation

#### Vorteile:

- flexiblen Ressourceneinsatz (Mitarbeiter können am Tagesgeschäft oder Projekt arbeiten)
- Wissensaustausch zwischen Projekt- und Stammorganisation möglich
- Keine Konflikte durch spätere Reintegration in die Stammorganisation

#### Nachteile:

- Gefahr der Mitarbeiter Überlastung
- Hoher Abstimmungsaufwand zwischen Abteilung und Projekt

#### Projektleiter Aufgaben:

- fachliche Weisungsbefugnis
- keine disziplinarische Weisungsbefugnis
- volle Zielverantwortung

### 1.5.3 Stablinien Projektorganisation

#### Definition:

Bei der Stablinien Projektorganisation ist der Projektleiter nur koordinierend und beratend tätig. Er wird in einer sogenannten Stablinie zwischen Unternehmens-

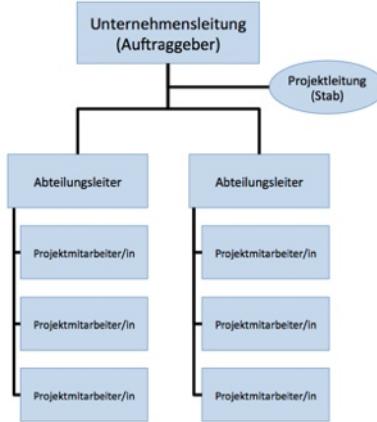


Abbildung 12: Stablinien Projektorganisation

und Abteilungsleitung angesiedelt. Deshalb heisst diese Organisationsform auch Stablinien Projektorganisation.

#### Vorteile:

- Einfacher Wissensaustausch
- Keine organisatorische Anpassung notwendig

#### Nachteile:

- Sehr hoher Koordinationsaufwand
- Starke Belastung Stammorganisation

#### Projektleiter Aufgaben:

- Keine fachliche Weisungsbefugnis
- Keine disziplinarische Weisungsbefugnis
- dadurch keine Zieleverantwortung

### 1.5.4 Rollen und Gremien

#### Was sind die typischen Rollen im agilen / hybriden Vorgehen?

- Projektleiter
- SCRUM Master
- Entwicklungsteam
- Product Owner

#### Was sind die typischen Rollen beim klassischen Vorgehen?

- Projektleiter
- Fachgremium
- Projekt Mitarbeiter

- projekt Office

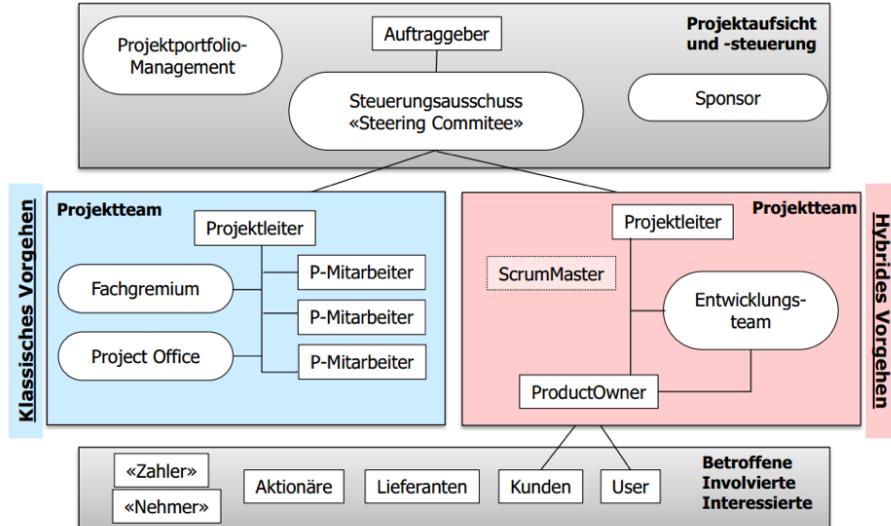


Abbildung 13: Rollen klassisches / hybrides Vorgehen

## 1.6 SODA

LZ5: Sie kennen die Grundzüge des SoDa Vorgehensmodells

**Welche Phasen sind beim SoDa agil / welche klassisch und für welches Vorhaben eignet sich das SoDa Modell?**

- Agil: Konzeptions- und Realisierungsphase
- Klassisch: Einführung, Nutzung

Beim SoDa Modell verläuft Konzeptions- und Realisierungsphase wie bei Scrum. Die anderen Phasen werden eher wie beim klassischen Modell gehalten. Somit ist während der Entwicklung ein agiles Vorgehen möglich. Jedoch sind die anderen Phasen geordnet.

**Grundlagen SoDa** SoDa basiert auf dem Vorgehensmodell Scrum und bindet dieses in einen Projektmanagement-Rahmen ein.

Es gibt 4 verschiedene Rollen:

- Scrum Team
- Product Owner
- Scrum Master
- Projektleiter

**Für welche Projekte eignet sich SoDa?** Im Fokus sind Software-Entwicklungsprojekte mit kleineren Teams über einige Monate.

## 2 Projektplanung

### 2.1 Lernziele

1. LZ1: Sie können für ein kleines bis mittleres Projekt eine Projektplanung vollständig durchführen.
2. LZ2: Sie sind in der Lage dies für ein klassisches Projekt basierend auf Arbeitspaketen und für ein agiles respektive hybrides Projekt zu tun.
3. LZ3: Sie sind in der Lage, aus der Beschreibung der Ausgangslage und dem Anforderungskatalog eines Projekts Teilresultate und Resultate für ein Projekt zu definieren.

### 2.2 Projektziele

**Definition Projektziel** Ein Projektziel ist ein angestrebter zukünftiger Zustand, welcher nach Inhalt, Mass und Zeit genau bestimmt wird.

Eine Anforderung ist eine Aussage über eine Eigenschaft oder Leistung eines Produktes.

**Projektziele sollten S.M.A.R.T. formuliert sein:**

- Spezifisch
  - Was soll genau erreicht werden?
  - Wo soll genau angesetzt werden?
  - Welche spezifischen Eigenschaften hat das zu erreichende Ziel?
  - Ist das Ziel interpretationsfrei?
  - Ist der Kontext des Ziels ausreichend berücksichtigt?
- Messbar
  - Wie möchte ich das messen?
  - Woran mache ich die Zielerreichung fest?
  - Wie viel davon genau?
- Angemessen
  - Besteht ein Bedarf für die geplante Maßnahme?
  - Ist das Ziel angemessen?
  - Ist das Ziel attraktiv?
  - Ist das Ziel motivierend genug?
- Realistisch
  - Ist das Ziel generell machbar?
  - Ist die Zielerreichung durch dich (und dein Team) beeinflussbar?
  - Ist das Ziel mit gegebenen Ressourcen in der gegebenen Zeit erreichbar?

- Terminiert
  - Bis wann soll das Ziel erreicht sein?
  - Gibt es Zwischentermine?

Beispiele:

Schlecht: Ich will weniger rauchen. Besser: Ab dem 1.5. Rauche ich keine einzige Zigarette mehr, bis zum Rest meines Lebens.

Schlecht: Die Stakeholder sollen informiert werden. Besser: Alle in der Stakeholder-Analyse mit Priorität A gekennzeichneten Stakeholder erhalten jeweils zum Monatsende einen 2-seitigen Statusbericht mit Angaben über Soll- und Ist-Stand des Projektes und aktuelle Aktivitäten per Mail zugesendet.

Schlecht: Bessere Nutzerfreundlichkeit. Besser: Die Usability der Software wird von mindestens 90% der Teilnehmer eines Usability-Tests mit „Sehr gut“ bewertet.

## **2.3 Projektresultate**

### **Definition Projektresultate**

- Projektresultaten meint man die Arbeitsergebnisse, welche das Projekt abzuliefern hat. Man beschreibt, was der Auftraggeber oder die Benutzer „in die Hand“ kriegen
- Projektresultate präzise und vollständig definieren und beschreiben!
- Resultate sind nicht Aktivitäten!

### **Eigenschaften von Projektresultate, werden gebraucht für**

- Werden gebraucht für die Erstellung von Arbeitspaketen, Projekt-Controlling und Projekt-Reporting
- Klassisch: Hauptresultate werden unterteilt in Teilresultate. Teilresultate können dann 1-n Arbeitspakete enthalten.
- Agil: Hauptresultate werden in n-Sprints unterteilt und 1 Sprint enthält im Nachhinein n Sprint-Backlog-Items

### **Welche Ergebnisse liegen in einem IT-Projekt vor:**

- Applikationssoftware
- Applikationsdokumentation
- Prozesse
- (Migration)
- Projekt-Management Ergebnisse

## **2.4 Arbeitspakete**

### **Definition Arbeitspaket**



Abbildung 14: Projektplanung

Ist die kleinste Planungseinheit und ist eine in sich geschlossenen Aufgabe welche bis zu einem festgelegten Zeitpunkt, mit einem definierten Ergebnis und Aufwand vollbracht werden kann.

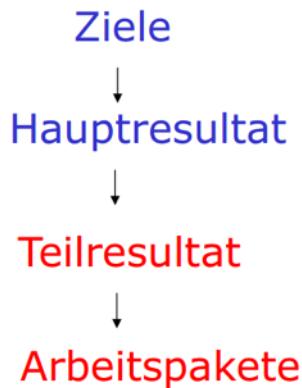


Abbildung 15: Arbeitspaket

Ein Arbeitspaket sollte zwischen 5 bis 20 Personentage Aufwand haben.

#### Arbeitspaket Inhalt

- Titel
- Beschreibung
- Aufwand von ca. 2 bis 20 Arbeitstage (Arbeitspakte die weniger als 2 Tage dauern, sollen zusammengefasst werden)
- Sollte einem MA zugewiesen sein (eine MA sollten nicht mehr als 2 Arbeitspakte zeitgleich zugeordnet sein)
- Ein MA nicht mehr als 2 Arbeitspakte gleichzeitig

#### Arbeitspaket erstellen

Nach dem die Ziele definiert wurden, daraus Anforderungen erstellt sind und die Resultate festgelegt wurde, können Arbeitspakte erstellt werden.

- Anforderung (entsteht aus Ziel) werden auf die Resultate gemappt
- Aus diesem Mapping kann in der Regel ein Arbeitspaket generiert werden
- Benennung Arbeitspaket
- Schätzung des Aufwandes Arbeitspaket
- Vorgänger / Nachfolger definieren (festlegen der logischen Abläufe im Ablaufplan)

## 2.5 Meilensteine

### Definition Meilenstein

Ein geplanter Punkt im Projektablauf an dem vorher festgelegte, messbare Ergebnisse vorliegen, die es erlauben den Projektfortschritt festzustellen.

### Was gehört zu jedem Meilenstein?

Zu jedem Meilenstein gehören Artefakte:  
 \* Prototypen / Prereleases / Releases  
 \* Testprotokolle  
 \* Entwurfsdokumente

### Wann ist der Meilenstein erreicht?

Wenn die entsprechenden Artefakte vorliegen und ihre Prüfung (Review / Test) erfolgreich war.

### Meilensteine beim klassischen Model

Bei einem klassischen Modell sind die Meilensteine praktisch durch die Phasen vorgegeben.

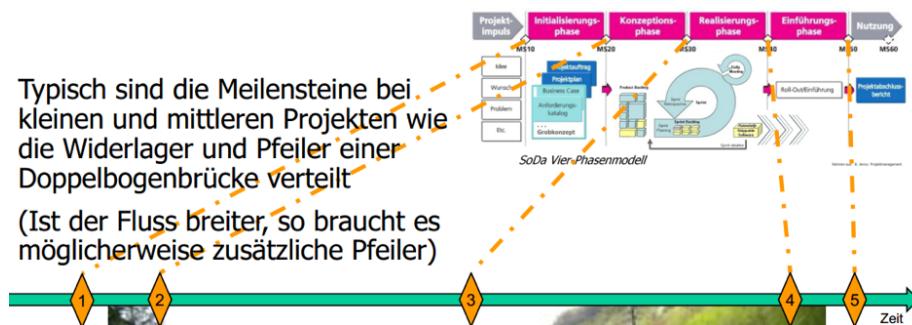


Abbildung 16: Meilensteine beim klassischen Model

### Meilensteine beim hybriden Model

Bei einem hybriden Modell muss der Meilensteinplan beim Kick-Off vorliegen. Damit der Product Owner die Einteilung in die jeweiligen Sprints durchführen

kann.

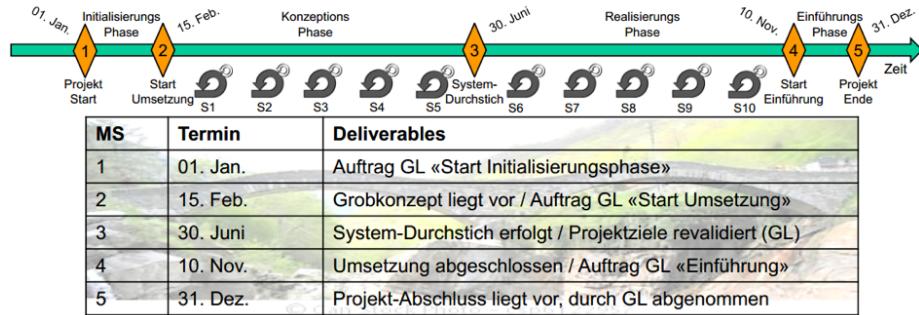


Abbildung 17: Meilensteine beim hybriden Modell

## 2.6 Scope

### Definition Scope

Beschreibt einerseits was nicht zum Projekt (OUT of scope) gehört und andererseits was die Einschränkungen sind.

Vielfach wird auch teilweise beschrieben, was zum Scope (IN scope) gehört. Trotzdem dies eigentlich mit der Beschreibung der Resultate nicht notwendig wäre.

## 2.7 Anforderungen

### Definition Anforderung

Eine Anforderung ist eine Aussage über eine Eigenschaft oder Leistung eines Produktes, eines Prozesses oder der am Prozess beteiligten Personen.

**Anforderung / Backlog-Item besteht aus:**

- Titel (kurzer Aussagekräftiger Name)
- Beschreibung (So viel wie nötig, so wenig wie möglich)
- Priorisierung
- Status (Klassisch: Entgegengenommen, Geprüft und Akzeptiert, Geprüft aber auf nächsten Release verschoben/ Agil: Alle Anforderungen im Backlog haben Status proposed «Geprüft aber auf nächsten Release verschoben» erst im Sprint haben sie «Geprüft und Akzeptiert»)
- Source: Person von dessen die Anforderung stammt
- Komplexität: (Klassig: Wird vom PL geschätzt, Agil: Erst wenn die Anforderung im Sprint-Backlog ist, wird dieser auf Initiative des Product Owners mit dem Team geschätzt)

- Benefit: Beschreibung des Nutzens der Anforderung (wird für Prio genutzt)

<b>Klassisches Vorgehen</b>	<b>Hybrides Vorgehen</b>
Anforderungskatalog	Produktbacklog
Ziel: möglichst vollständige Liste aller Anforderungen. Priorisierung fix. Änderungen restriktiv und nur über einen Change Management Prozess zugelassen.	Ziel: wichtigste (ca. 80%) Anforderungen grob erfasst. Priorisierung wird laufend nachgeführt. Änderungen können jederzeit einfließen.
Konkretisierung durch Bildung von -> Arbeitspaketen	Konkretisierung durch herunterbrechen in -> UserStories

Abbildung 18: Anforderung

Priorität ist beim klassischen Model fixiert. Priorität wird beim hybriden Model laufend festgesetzt.

### 3 Projektplanung Klassisch

#### 3.1 Projektstandsitzung

- Garantiert einen regelmässigen Informationsfluss innerhalb des Projektes
- Sie bietet eine Plattform um über Probleme zu sprechen und Entscheide zu fällen
- Es findet das Reporting der Projekt Mitarbeiter oder Teilprojektleiter gegenüber dem Projektleiter statt

#### 3.2 Projektstrukturplan

Eine resultatorientierte Gliederung der Gesamtaufgabe in Teilresultate / Arbeitspakete.

Der Projektstrukturplan ist die vollständige Darstellung aller Elemente eines Projekts und ihrer Beziehungen. Dabei werden die Elemente werden hierarchisch gegliedert, so dass eine Baumstruktur entsteht. Der PSP beinhaltet keine zeitliche Aussage über die Abarbeitung des Projektes, sondern gibt lediglich die Strukturierung wider. Der PSP wird auch als WBS (Work Breakdown Structure) bezeichnet.

#### 3.3 Ablaufplan

##### Definition Ablaufplan

Hier werden die logischen Abhängigkeiten der einzelnen Arbeitstakte festgelegt.

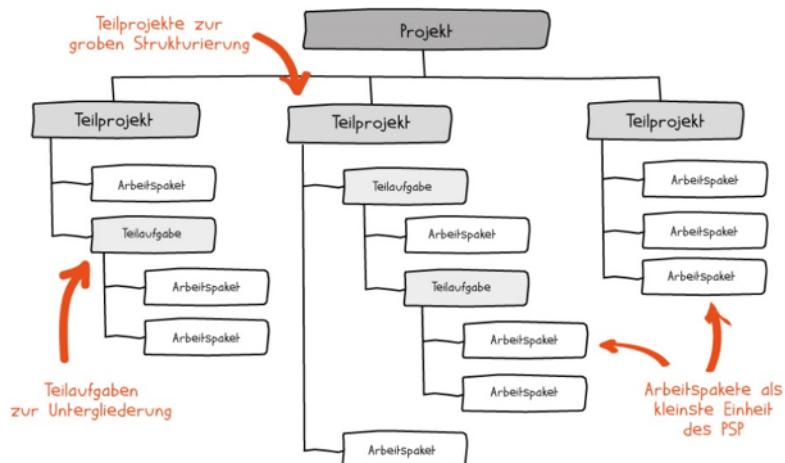


Abbildung 19: Projektstrukturplan 1

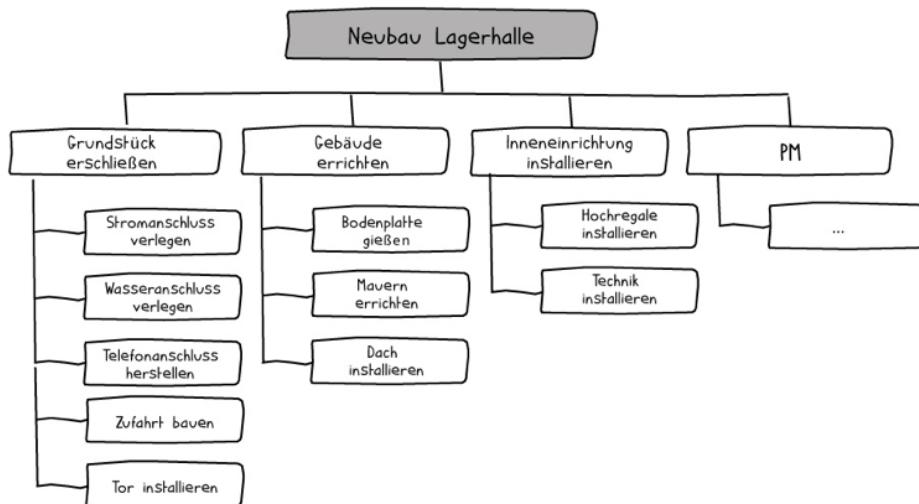


Abbildung 20: Projektstrukturplan 2

Grundlegend können die Arbeitspakete in Phasen unterteilt werden. Die Arbeitspakete innerhalb einer Phase sollten parallel bearbeitbar sein.

- Arbeitspakete möglichst parallel laufen lassen
- Abhängigkeiten nur dort wo wirklich notwendig

### **Definition Kritischer Pfad im Projekt**

Wenn keine Pufferzeit mehr zur Verfügung steht, Aneinander gereihte abhängige Arbeitspakete welche keine Pufferzeit mehr haben.

### **Warum wird beim hybriden Vorgehen ohne Zeitreserven geplant?**

Beim hybriden Vorgehen wird ohne Zeitreserven geplant, es werden dank dem Pull-Prinzip in jedem Sprint so viele der höchst priorisierten Stories wie möglich umgesetzt.

## **3.4 Aufwandschätzung**

### **Vorgehen Aufwandschätzung**

1. Zuerst Aufwandschätzung mittels Nettomethode. Mittels Nettomethode ein Arbeitspacket nach Aufwand schätzen, ohne äusserliche Einflüsse oder Risiken
2. Vorgehen Aufwandschätzung Bruttomethode: Mittels Bruttomethode ein Arbeitspacket nach Aufwand schätzen, mit äusserlichen Einflüssen und Risiken
3. Gesamtes Projekt schätzen

### **Welche Schätzmethode wenden wir an? Und wie geht die?**

Es gibt verschiedene Schätzmethoden. Wir lernen die «Individual Expert Judgment» Methode kennen.

Dabei werden die Arbeitspakete von einem Experten geschätzt.

Personen die das Projekt durchführen sollen den Aufwand schätzen. Bei Unsicherheit kann zur Gesamtschätzung einen Aufwand von 10 bis 20 % addiert werden.

### **Alternative Schätzmethoden**

Alternative Methode «Delphi Wideband»: Schätzung von Experten innerhalb einer Gruppe

Iterationen werden mit dem Time-Boxing Verfahren geplant.

Beim Time-Boxing-Verfahren schätzt man keinen Aufwand, sondern gibt einen maximalen Aufwand (timeboxing) vor. Nach Erreichen des Aufwandes werden die Arbeiten gestoppt. fixe Dauer, flexibler Inhalt

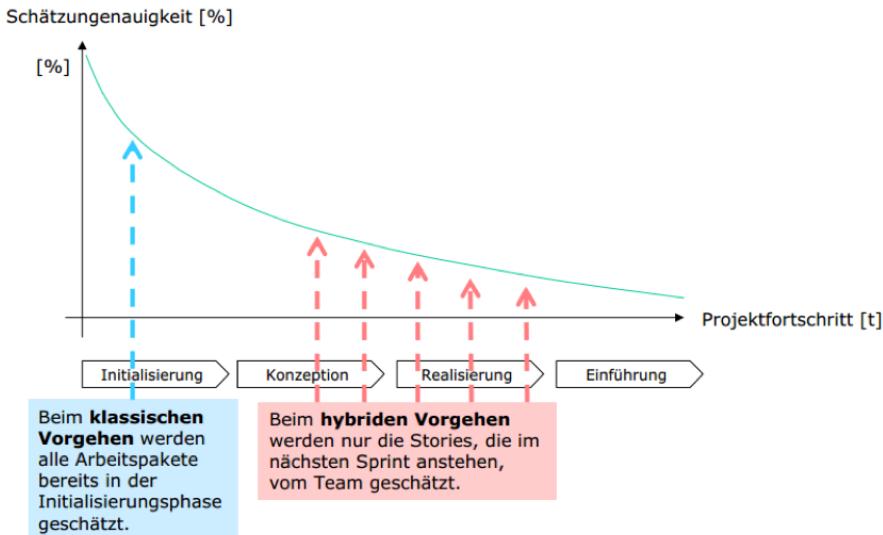


Abbildung 21: Aufwandschätzung

Das Proof of Concept wird im Time-Boxing Verfahren geschätzt, damit das Artefakt in einer relativ kurzen Zeit fertiggestellt werden kann.

### 3.5 Ressourcenplanung

#### Definition Ressourcenplanung

Zuordnung der Mitarbeiter zu den Arbeitspaketen und die Verteilung der Arbeitspakete auf die Zeitachse.

Input: Alle Arbeitspakete inkl. Aufwandplanung und Ablaufplan

Ziel: Zuteilung der Arbeitspakete an die Mitarbeiter und die Dauer der Arbeitspakete bzw. des Projektes ermitteln

Output: Dauer des Projektes und als Lieferobjekt der Ressourcen-Plan

#### Vorgehen

- Festlegung der anhand der Arbeitspakete benötigte Skillprofile und Zuordnung der Arbeitspakete an die Profile
- Verfügbare Mitarbeiter anschauen und Zuordnung der Arbeitspakete an die Mitarbeiter
- Verteilung der Aufwände pro Arbeitspaket und Mitarbeiter auf die Zeitachse unter Beachtung der Auslastung der Mitarbeiter.

**Warum wird beim agilen Vorgehen ohne Zeitreserven geplant?**

Es werden dank dem Pull-Prinzip in einem Sprint so viele der höchst priorisierten User-Storys umgesetzt.

Wie wird der Personalbedarf gerechnet?

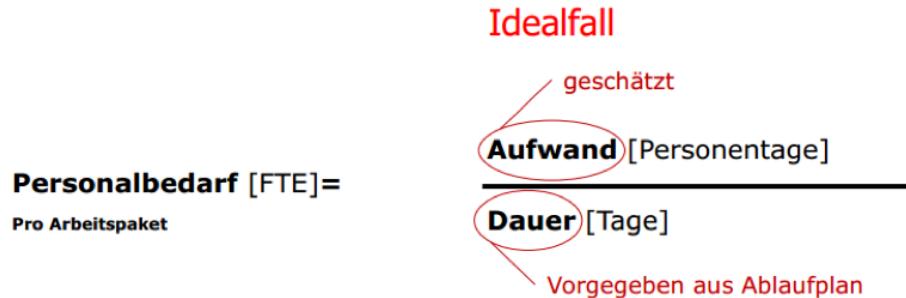


Abbildung 22: Personalbedarf Idealfall

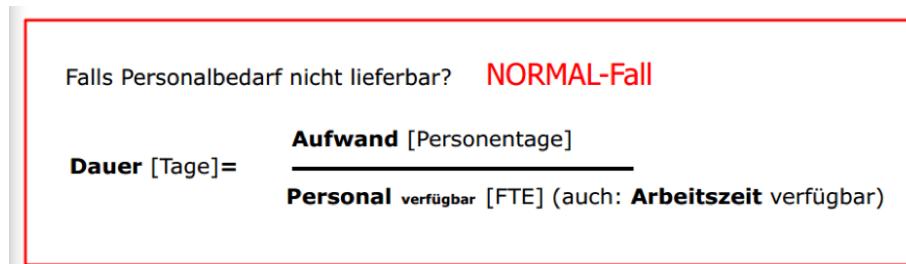


Abbildung 23: Personalbedarf Normalfall

### 3.6 Projektplanung Hybrid

Unterschied Klassisch vs. Agil

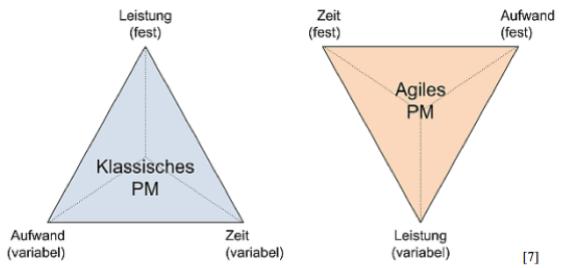
- Anforderungen
- Zeitplan

Welche Phasen werden beim Scrum / agilen Vorgehen wiederholt?

- Konzeptions- und Realisierungsphase

Beim SCRUM wird iterativ vorgegangen. Es ist kaum möglich von Anfang an alle Anforderungen zu erfassen. Das SCRUM Vorgehen erlaubt, dass die Konzeptions- und Realisierungsphase wiederholt wird, so werden die Anforderungen ergänzt und die Software immer wieder aufs Neue angepasst. Software wird so schrittweise entwickelt.

Welche Phasen sind beim SoDa agil, welche klassisch?



- Alle Anforderungen (welche die Leistung beschreiben) sind vollständig definiert und festgeschrieben
  - Der Zeitraum ist geplant
  - Der Aufwand (= Ressourcen wie Kosten und Mitarbeiter) muss allenfalls erhöht werden, um die Fertigstellung zu erreichen.
- Die wichtigsten Anforderungen sind grob beschrieben und priorisiert, die vollständige Umsetzung aber nicht festgeschrieben.
  - Es werden laufend in Sprints die wichtigsten Anforderungen konkretisiert und umgesetzt. Die Software wird - wo sinnvoll - freigegeben.
  - Der Aufwand (= Ressourcen wie Kosten und Mitarbeiter) ist festgelegt und wird gesteuert.

Abbildung 24: Unterschied Klassisch vs. Agil

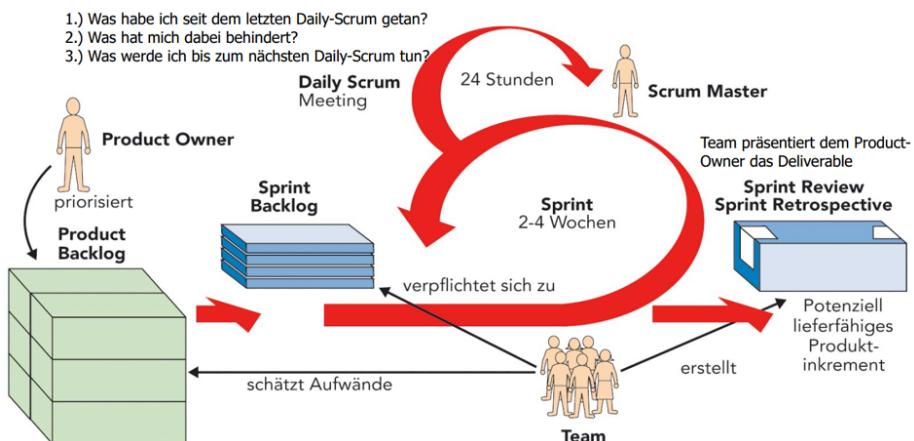


Abbildung 25: Projektplanung Agil

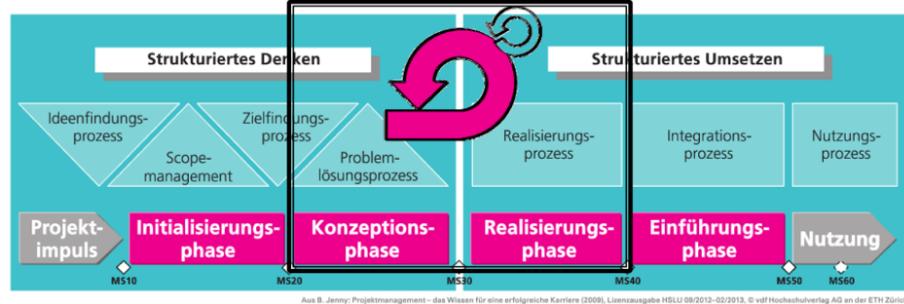


Abbildung 26: Iterativ-Inkrementelles Vorgehen

- Agil: Konzeptions- und Realisierungsphase
- Klassisch: Initialisierung (mit Projektauftrag, Projektplan, Anforderungskatalog) und Einführung

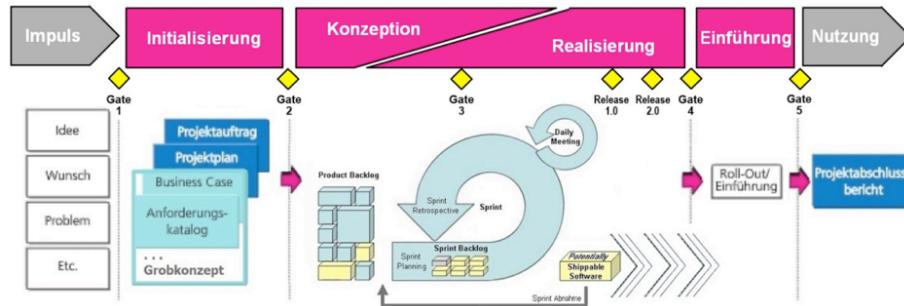


Abbildung 27: Projektplanung SoDa

Dieses Modell trägt einerseits dem Bedürfnis von Auftraggebern und übergeordneter Organisation nach geordneter Projektdurchführung mit standardisierten Phasen und Meilensteinen Rechnung. Andererseits wird sichergestellt, dass die Software iterativ-inkrementell entwickelt wird und sich agil an den Bedürfnissen und Möglichkeiten orientiert.

### 3.6.1 Planung

Beim Hybriden Modell wird nicht von Anfang an die ganze Planung für das ganze Projekt erstellt. Die Planung und Umsetzung sind viel dichter miteinander verknüpft.

#### Wie sieht die Projektplanung beim agilen Modell aus?

- Es beginnt nicht wie beim klassischen Modell in der Initialisierungsphase
- Es wird nur einmal die Meilensteinplanung gemacht und

- Dann innerhalb der Meilensteine werden die Dauer der Sprints für die Konzeptions- und Realisierungsphase festgelegt.

### Mit welchem Verfahren werden die Iterationen / Sprints geplant?

Innerhalb der Meilensteinplanung werden nun die Iterationen via Time-Boxing Verfahren festgelegt.

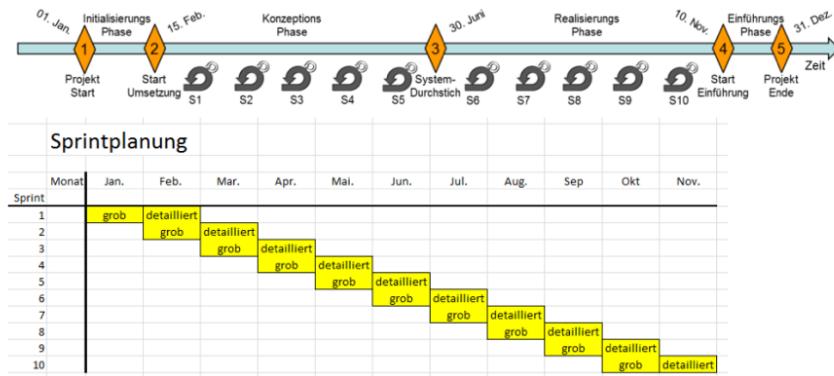


Abbildung 28: Meilensteinplanung

### 3.6.2 Ablaufplan und Ressourcenplanung

Wie sieht der Ablauf- und Ressourcenplan beim agilen Vorgehen aus und wer ist dafür verantwortlich?

Für den Ablauf und Ressourcenplan ist der Product Owner verantwortlich. Die Ablaufplanung und Ressourcenplanung erfolgt auf zwei Ebenen.

1. Produkt-Backlog: Initiale Erstellung und laufende Pflege,
  - Erfassen der Einträge im Backlog
  - Ordnen der Einträge in eine zielführende Fertigstellungsreihenfolge
  - Transparenz für alle Beteiligten sichern (alle haben Zugriff auf das Backlog)
  - Fortlaufende Pflege (Grooming) des Produktes Backlogs
2. Sprint-Backlog
  - Auf Basis der Rahmenplanung im Backlog, die Anforderungen mit der höchsten Priorität in die nächste Sprint-Backlog einfügen, sinnvolles Sprint-Backlog-Ziel festlegen
  - Ausgewählte Backlog-Items konkretisieren und Aufwand schätzen

### 3.6.3 Produkt Backlog und Backlog Items

Ist das Produkte-Backlog in der Initialisierungsphase abgeschlossen?

Nein, das Produkte-Backlog lebt während der Konzeptions- und Realisierungsphase weiter, es wird fortlaufend gepflegt und angepasst (Grooming).

**Im Gegensatz zum Klassischen Modell ist das Produkte – Backlog am Ende der Initialisierungsphase:**

- Nicht vollständig
- Die Anforderungen sind zu 80% identifiziert.
- Anforderungen sind priorisiert
- User-Storys für den 1. Sprints werden ausgewählt

**Status der Anforderungen im Produkte Backlog und im Sprint-Backlog?**

- Die Anforderungen sind im Status «proposed»
- Agil: Alle Anforderungen im Backlog haben Status proposed «Geprüft aber auf nächsten Release verschoben» erst im Sprint haben sie «Geprüft und Akzeptiert»
- Anforderungen sind nicht endgültig Priorisiert

**Was ist ein Epic und was hat es mit der Initialisierungsphase zu tun?**

In der Initialisierungsphase werden die Anforderungen im Produkte-Backlog meist grobgranular als Epics beschrieben. Ein Epic kann ein kompletter Arbeitsablauf sein, welcher danach noch in einzelne User-Storys heruntergebrochen wird.

### Userstory Definition

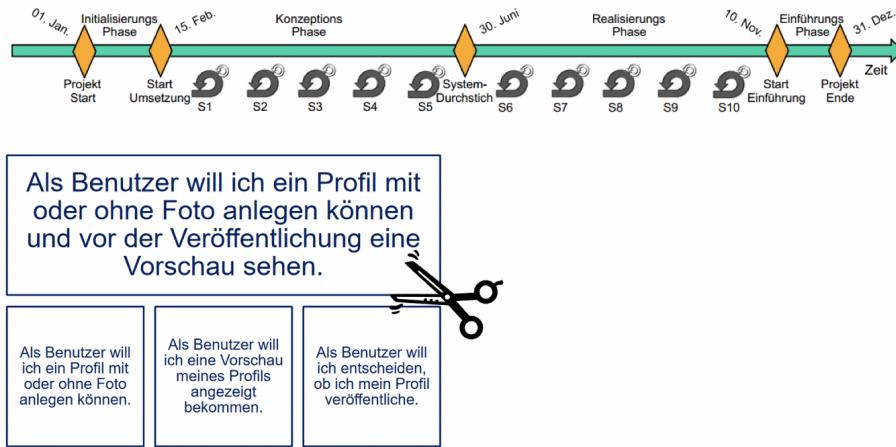
Eine UserStory ist eine eigenständige Arbeitseinheit, die von den Entwicklern und den Stakeholdern vereinbart wurde. Stories sind das Herz von Scrum und die Bausteine der Sprints.

**Was ist ein other Workitem?**

- Other Workitems: technische Aufgaben die nötig sind um die Anforderungen zu entwickeln (stabilisieren der Architektur)

### Definition Grooming

- Grooming: Mindestens 1 Mal pro Sprint wird ein Grooming vom Product Owner durchgeführt, es werden:
  - Neue Einträge hinzugefügt (Neue Anforderungen)
  - Nicht mehr benötigte Einträge gelöscht
  - Die Priorisierung der Anforderungen aktualisiert
  - Aufwände korrigiert / ergänzt
  - Produkt-Backlog Grooming Meeting in der Mitte des Sprints angesetzt inkl. Scrum Team
- D.E.E.P:
  - Ein guter Produkte-Backlog hat die D.E.E.P Eigenschaften:
    - Detailed: Angemessen Detailiert, direkt vor dem Sprint so detailliert dass er umsetzbar ist
    - Estimated: Team schätzt Aufwand eines Backlog-Item



**Epic:**  
Mitarbeitende können ein eigenes Passwort für ihren NSA Account festlegen.

**Offene Fragen:**  
Wie gelangen Benutzer zur Registrationsseite?  
Gibt es Sicherheitsanforderungen bei der Passwortwahl?

**Stories:**  
1.) Der Administrator muss per Mail einen Zugriffslink zur Registrationsseite verschicken können.  
2.) Registrierte Mitarbeitende können im Profil ihr Passwort neu setzen.  
3.) Die Passwortstärke soll den NSA-Sicherheitsanforderungen genügen

Abbildung 29: Workitem

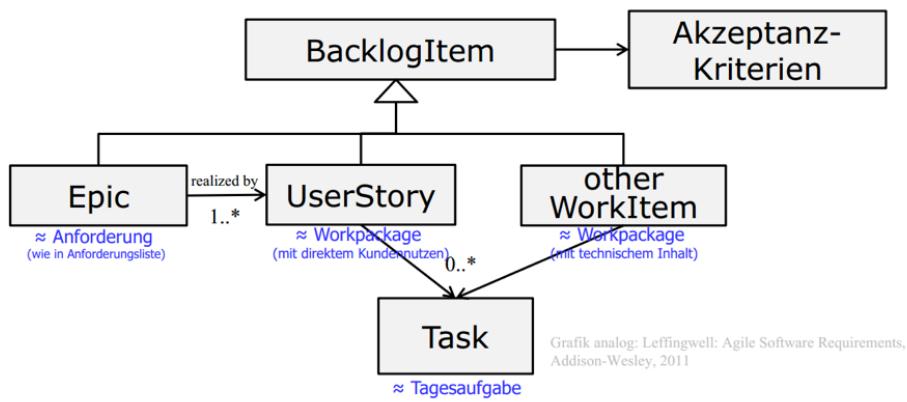


Abbildung 30: Backlog Item

- Emergent (Entwickelt sich, wird gepflegt): Wird lauffend gepflegt, der PO pflegt zusammen mit dem Team das Produkte-Backlog (grooming)
- Prioritised: Jedes Backlog-Item muss eine Priorität haben, damit man weiß ob es in den nächsten Print reinkommt oder nicht.

### 3.6.4 Sprintplanung

#### Definition Sprintplanung

In der Sprintplanung geht es darum, wie viele der höchst priorisierten Backlog-Items in den nächsten Sprint aufgenommen wird. Der Product Owner erarbeitet mit dem Team das neue Sprint-Ziel.

#### Was soll erreicht werden?

Das Sprint-Ziel (in sich geschlossen, testbar, potentially shippable) soll erreicht werden. Mit den verfügbaren Ressourcen im gewünschtem Time-Boxing.

1. Product Owner geht mit dem Team die höchst priorisierten Back-Log-Items durch.
2. Epics die fachlich noch nicht präzise genug formuliert wurden, werden in mehrere User-Storys herunter gebrochen (Story-Spliting)
3. In der Sprint-Planung werden die User-Storys ergänzt. Zu jedem User Story gehören Akzeptanzkriterium.
4. Besprechung der Eigenschaften, Abhängigkeiten sodass eine Aufwandschätzung der einzelnen Items gemacht werden kann.
5. Es werden dann nur so viele Stories in den Sprint-Backlog übertragen wie das Team potentiell liefern kann.

#### Wie ist eine User Story formuliert?

sollte folgendermassen formuliert sein: Als möchte ich um .

Eine UserStory ist **I.N.V.E.S.T:** **Independent:** Möglichst unabhängig **Negotiable:** möglichst zerlegbar, verhandelbar **Valuable:** Hat wirtschaftlichen Wert **Estimated:** So klar das es vom Team geschätzt werden kann **Small:** klein genug um in einem Sprint entwickelt werden zu können **Testable:** klare Akzeptanzkriterien

### 3.6.5 Aufwandschätzung

Damit am Ende eines jeden Sprints eine potentiell lieferbare Software entsteht müssen alle Aufwände geschätzt werden.

#### Welche Elemente gehören in eine Aufwandschätzung bzw. müssen für jeden Sprint geplant und geschätzt werden?

- Entwurf & Dokumentation
- Implementation & Unitests
- Integration & Integrationstest

Beispiel für eine User Story:

- Als Student/in will ich mich online zu einer Vorlesung anmelden, sodass ich einen Platz in der Übung erhalte.

Beispiele für dazu passende Akzeptanzkriterien:

- Wenn sich ein/e Student/in einer anderen Fakultät anmeldet, muss eine Fehlermeldung erscheinen.
- Wenn der/die Studierende bereits angemeldet ist, muss eine Fehlermeldung erscheinen.
- Studierende können sich bei einer Vorlesung mit begrenzter Teilnehmerzahl nur anmelden, wenn noch nicht alle Plätze vergeben sind.

Abbildung 31: Beispiel User Story

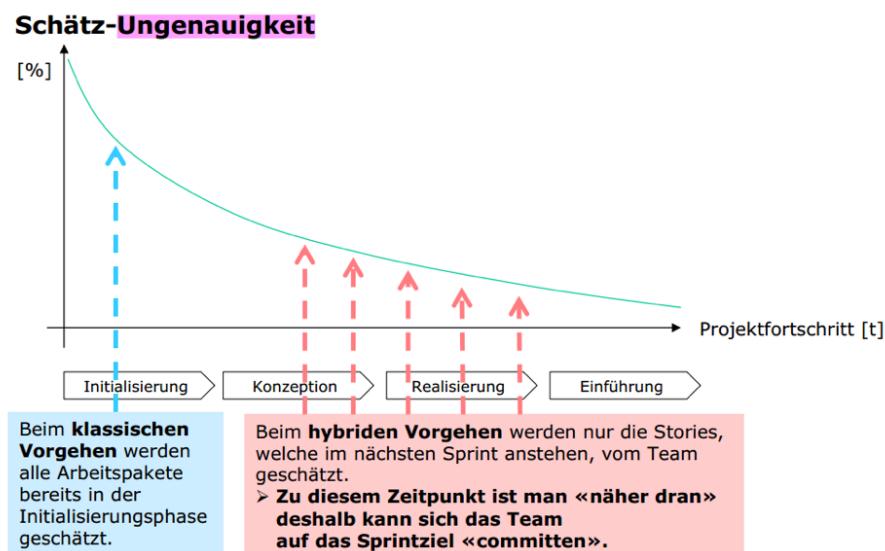


Abbildung 32: Schätz-Ungenauigkeit

- Sprintabnahme gemäss Akzeptanzkriterien

Es sollen alle Team-Mitglieder mit schätzen, damit ein Kommittent auf das Sprintziel entsteht.

### 3.6.6 Ressourcenplanung & Scrumboard

Story	To Do	In Process	To Verify	Done
As a user, I... 8 points	Code the... 9    Test the... 8 Code the... 2    Code the... 8 Test the... 8    Test the... 4	Code the... DC 4 Test the... SC 8	Test the... SC 6	Code the... DC 6    Test the... SC 6 Test the... SC 6    Test the... SC 6 Test the... SC 6
As a user, I... 5 points	Code the... 8    Test the... 8 Code the... 4    Code the... 6	Code the... DC 8		Test the... SC 6 Test the... SC 6 Test the... SC 6

<https://www.mountaingoatsoftware.com/agile/scrum/scrum-tools/task-boards>

Abbildung 33: Scrum Board

### Was lässt sich mit dem Taskboard / ScrumBoard überprüfen?

Mittels Taskboard lässt sich jederzeit überprüfen, welche Backlog-Items im Sprint umgesetzt werden oder wie der Status der einzelnen User-Storys oder Tasks sind. Eine User-Story wird in mehrere Tasks (To Do) unterteilt. Da sie meist zu gross sind und oder unterschiedliche Skills benötigen, welche nicht alleine durch einen Mitarbeiter abgearbeitet werden kann. Mittels dem Pull-Pattern von Kanban ziehen die Mitglieder im Daily Scrum nach jedem abgeschlossenen Task einen neuen Task.

### 3.6.7 Projektkosten

#### Kosten und Aufwendungen – Zusammenhang?

Alle Kosten sind zu gleich Aufwendungen, aber nicht alle Aufwendungen zählen zu den Kosten.

#### Aufwand sind Kosten, wenn...

Alle Bedingungen müssen erfüllt sein, damit ein Aufwand zu Kosten gehört:

- Betriebsbezogen: Aufwand muss mit betrieblichen Leistungen verbunden sein

klassisch	$\diamond$	hybrid	Bemerkungen
Ziele & Hauptresultat	=	Ziele & Hauptresultat	Auf Ebene der Ziele & Hauptresultate unterscheiden sich die klassisches und hybrides Vorgehen nicht.
Teilresultate	$\sim$	Sprintziele	Teilresultate werden zu Beginn festgelegt, Sprintziele erst vor dem jeweiligen Sprint
Anforderungskatalog	$\approx$	ProductBacklog	Backlog nicht vollständig, Einträge und Priorisierung können ändern > Grooming
Workpackage	$\sim$	UserStory	Workpackages werden zu Beginn vom PL definiert und geschätzt, UserStories erst vor dem jeweiligen Sprint vom PO gemeinsam mit dem Team.
Ablaufplanung	$\diamond$	Grooming / Sprintplanung	APs werden zu Beginn den Phasen zugeordnet und innerhalb nach Abhängigkeit wenn möglich parallelisiert. Backlog wird laufend nachgeführt (> Grooming) höchst-priorisierte Einträge werden in Stories gesplittet und diese raschmöglichst in einem Sprint geplant.
Ressourcenplanung	$\diamond$	Sprintplanung / TaskBoard & StandUp	Ressourcen werden zu Beginn entsprechend Skills und Verfügbarkeit den Arbeitspaketen zugeordnet. Das Team organisiert sich im Sprint selbst und jede/r zieht entsprechend Skills und Verfügbarkeit die nächste Task. Im StandUp werden Behinderungen gelöst.

Abbildung 34: Klassisch Hybrid

- Periodenbezogen: Aufwand muss in laufender Periode stattfinden
- Ordentlich: Aufwand darf nicht zu ungewöhnlichen Ereignissen zählen. (Unfälle)

**Projektkosten** sind einmalig

Beispiele:

- Hardware kauf von
  - Server
- Software
  - Lizizenzen
- Personal
  - Schulungen

**Betriebskosten** sind laufende Kosten

Beispiele:

- Hardware
  - Abschreibungen
- Software
  - Abschreibungen
- Personal
  - Schulung neuer MA

### Unterschiede Projektplanung Agil und Klassisch

Projektplanung wird beim agilen Modell nicht bereits in der Initialisierungsphase gemacht

Phase / Objekt	Klassisch	Agil
Basis für Planung	Projektziel -> (Anforderungen) -> Hauptresultat -> Teilresultate -> Arbeitspacket	Projektziel -> Hauptresultat -> Sprintziel [n] -> Sprint-Backlog-item
Anforderungskatalog	Anforderungskatalog	Produkt-Backlog
Meilensteinplanung	Meilensteinplanung	Meilensteinplanung - Dauer der Sprints für Konzeptions- und Realisierungsphase
Projektstrukturplan	Projektstrukturplan	-
Ablaufplan	Logische Abhängigkeiten der <u>Arbeitspakte</u> , Arbeitspakte in Phasen unterteilt, Auf Zeitachse gelegt	Ablaufplan – und Ressourcenplanung - Produkt-Backlog - Sprint-Backlog
Aufwandschätzung	- Individual <u>Expert Judgement</u>	Erfolg auf Basis der User-Stories
Ressourcenplanung	Ressourcenplanung	Findet im: - Produkte-Backlog - Sprint- Backlog statt
Ende Initialisierungsphase	Alle Anforderungen sind identifiziert und geplant	Hier trennen sich die Wege zwischen klassischem und Agilen Model

Abbildung 35: Unterschiede Projektplanung Agil und Klassisch

## **4 Projektkommunikation**

**Welches sind die wichtigsten Instrumente / Mechanismen welches einem PL zur Verfügung stehen?**

- Projektcontrolling
- Projektreporting
- Projektsitzung
- Test / Stichproben

**Welches sind die wichtigsten Fähigkeiten eines Projektleiters?**

- Leadership
- Kommunikation mit Mitarbeiter
- Verhandlungen
- Problemlösungen

**Welches sind die wichtigsten Kommunikations-Instrumente in einem Projekt?**

- Projektsitzung
- Projektreporting

### **4.1 Projektsitzung**

**Zwecke einer Projektsitzung / Daily Stand Up Meeting:**

- Garantiert einen regelmäßigen Informationsfluss
- Bezuglich Projektresultate übt einen gewissen Druck auf MA auf
- Projektreporting für PL

**Agenda Projektsitzung:**

- Begrüßung
- Letztes Protokoll
- Personelles / Abwesenheiten
- Fachliches
- Pendenzen
- Open Issues

### **4.2 Projektreport**

Wichtigstes Kommunikationsmittel gegen außen. Für Empfänger Auftraggeber, wichtigste Stakeholder. Die wichtigsten Inhalte sind:

- IST / Soll der Projektziele
- Drohende Risiken / Probleme
- Massnahmen und Gründe der Abweichung
- Erzielte Erfolge

- Zeitpunkt des Projektberichts

#### **4.3 Kommunikation mit Stakeholder**

Definition Stakeholder: Anspruchsgruppen die unmittelbar Einfluss auf den Projektfortschritt haben und vom Projektziel direkt / indirekt betroffen sind.

Stakeholder Analyse Vorgehen:

1. Stakeholder identifizieren
2. Beziehungsanalyse
3. Kommunikationsplan

### **5 Risiko Management**

#### **Definition Risiko Management:**

Risiko ist eine kalkulierte Prognose eines möglichen Schadens. Jenes Risiko muss minimiert werden welches am häufigsten eintritt / eintreten kann.

Nach dem Abschluss des Meilensteines / Arbeitspaket soll das Projektrisiko neu beurteilt werden.

#### **Ziel von Risikomanagement:**

- Risiken erkennen
- Massnahmen ergreifen
- Risiken kontrollieren
- Gutes Projektmanagement verhindert Risiken

#### **5.1 Risiken**

#### **Klassische Risiken:**

- **Produktrisiken:** Welche Gefahren für Mensch & Umwelt treten bei Bau, Nutzung, Entsorgung des Produktes auf. Wie verhält sich das Produkt bei einem Teilausfall?
- **Projektrisiken:** Was hindert mich daran mein Projekt erfolgreich abzuschliessen?
  - Technische Risiken: zu wenig Know-How über die Technologie, Patente von Mitbewerbern
  - Implementationsrisiken: Architektur & Design nicht korrekt, Anforderungen ändern sich,
  - Wirtschaftliche Risiken: Mitbewerber stellen bessere Produkte her, Ressourcen stehen nicht zur Verfügung, Änderungen am Markt

#### **Fünf Schritte zum Risikomanagement Prozess:**

1. Risiko identifizieren
2. Risiko Analysieren (Ursachen, Eintrittswahrscheinlichkeit, Schadenmass)
3. Risiko Priorisieren (Risiko-Einstufung, Verantwortung festlegen)
4. Massnahmen erarbeiten (Risikovermeidung, Schadensreduzierung)
5. Risiko Überwachen (Regelmässig bewerten)

## 5.2 Risiko Identifizieren

**Hilfsmittel:**

- Checklisten
- Brainstorming
- Übertragung von früheren Erfahrungen

## 5.3 Risiko analysieren

**Risiko-Eintritt:** Ein Risiko hat eine Eintrittswahrscheinlichkeit. Beim Risikoeintritt materialisiert sich das Risiko (Schaden in Zeit oder Geld). Risikoeintritt ist Startsignal zur Schadensbegrenzung. **Eintrittsindikatoren:** Zeigen evtl. Frühzeitig das Eintreten des Risikos an.



### Würfelspiel

Risikoauslösendes Ereignis	den Würfel werfen
Risiko	Geld verlieren, wenn keine 6 gewürfelt wird
Eintrittswahrscheinlichkeit:	$5/6 = 83\%$
Auswirkungen	10 CHF verloren

Abbildung 36: Risiko Analysieren

## 5.4 Risiko priorisieren

Risiko Landkarte:

Visualisieren aller Projektrisiken

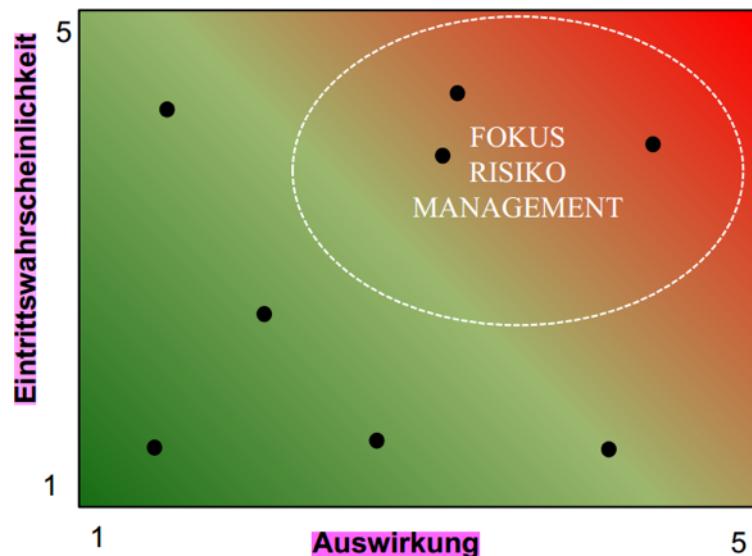


Abbildung 37: Risiko Landkarte

Verschiedene, unabhängige Gruppen sollen die Risiken quantifizieren. Nie die Schlüsselpersonen alleine das Risiko bewerten lassen.

## 5.5 Massnahmen erarbeiten

4 Risikomassnahmen:

- **Vermeiden:** Eintrittswahrscheinlichkeit reduzieren
- **Behandeln:** Zusätzlicher Aufwand um Eintrittswahrscheinlichkeit zu reduzieren. Z. B Prototypen erstellen
- **Begrenzen:** Aktionen zur Abschwächung, diese müssen stetig kontrolliert werden
- **Ignorieren**

## 5.6 Risiko überwachen

Laufend überwachen: hat sich etwas geändert? Greifen die Aktionen zur Abschwächung...

## Muster für eine Risikobeschreibung

Risikoformular	
Kurzname:	Risikokontrollnummer:
Urheber:	Tag der Entdeckung:
Beschreibung:	
Eintrittswahrscheinlichkeit:	Auswirkung:
Potenzielle Kosten:	Mögliche Verzögerung:
Ausgewählter Eintrittsindikator:	
Risikoplanung:	
Risikoklasse:	Erfordert Verminderung [ ]
	Erfordert Eventualfallplanung [ ]
	Akzeptiert (kein Plan erforderlich) [ ]
Vermeidung:	
Abschwächung:	
Behandlung:	

Abbildung 38: Beispiel Risikobeschreibung

Durch konsequente Überwachung und Steuerung des Projektes kann das Risiko minimiert werden.

## 5.7 Zusammenfassung

**Eintrittswahrscheinlichkeit** sollen minimiert werden durch vorbeugende Massnahmen z.B durch Prototypen und riskante Tätigkeiten früh genug angehen. Auswirkungen von Ereignissen minimieren (Eventuelpfad, alternative Lieferanten)

# 6 Projektcontrolling

## 6.1 Lernziele

- LZ1: Sie können erläutern, warum Projektcontrolling betrieben wird.
- LZ2: Sie sind in der Lage, auf Grundlage einer Liste von Arbeitspaketen ein sauberes Controlling für ein Software-Projekt aufzusetzen.
- LZ3: Sie kennen die Projektcontrolling-Instrumente in der agilen Entwicklung.

## 6.2 Klassische Projektleitung

### Warum wird Projekt-Controlling durchgeführt?

Projektcontrolling umfasst alle Aktivitäten um projektbezogene Abweichungen zwischen dem Sollzustand und Ist-Zustand aufzudecken.

### Ziele von Projekt-Controlling

- Risikoeintrittsindikatoren frühzeitig erkennen (technische Risiken, Fehlerhafter Zeitplan)
- Überprüfung ob alle Ressourcen ausgelastet sind
- Post-Mortem-Analyse um wichtige Erkenntnisse für die nächsten Projekte zu sammeln

### Was ist Projekt-Controlling?

Regelmäßiges Abgleichen des Projektes zwischen Sollzustand und Ist-Zustand.

### Was dient als Grundlage für das Projekt-Controlling? Der Projektplan

### Auf welcher Ebene wird das Projekt-Controlling durchgeführt?

- Projektziele
- Projektresultate
- Teilresultate
- Arbeitspakete

Auf der Ebene von Arbeitspaketen. Denn wenn unten die Arbeitspakete auf Grün sind, sind die oberen Ziele ebenfalls auf grün. Wenn jedoch das Projektziel grün ist, muss es nicht heißen, dass die Arbeitspakete ebenfalls grün sind.

### **Welche Stati können Arbeitspakete im klassischen Projekt-Controlling haben?**

- Offen, noch nicht begonnen
- Begonnen aber noch nicht fertig
- Fertig

### **Welche Kriterien müssen erfüllt sein, damit ein Arbeitspaket „fertig“ ist (absoluter Fertigstellungsgrad)?**

- Code muss integriert und getestet sein
- Freigabeprozess (Review) für Dokumentation beendet

### **Was bedeutet RAS?**

Rest-Aufwand-Schätzung: So viel Aufwand werden wir voraussichtlich noch brauchen.

### **Wie wird das wahrscheinliche Gesamtergebnis eines Arbeitspaketes berechnet?**

- Soll-Ist-RAS = Gesamtergebnis
  - Positiv = Vorsprung
  - Negativ = Verzug
- Rest-Budget = Soll-Ist

### **Was ist die Meilenstein-Trend-Analyse?**

Sie verdeutlicht auf einen Blick den Stand und Planungsverlauf des Projektes anhand der Meilensteine.

### **Regelkreis des Projekts-Controllings**

1. **Zieldefinition:** Die Projektziele werden nach S.M.A.R.T definiert. (Bsp. Gewinnmaximierung, Kundenzufriedenheit, Neue Software im Unternehmen)
2. **Planung** Wie werden die Ziele erreicht? Welche Massnahmen / Arbeitspakete müssen abgearbeitet werden um die Ziele zu erreichen?
3. **Datenerfassung** Regelmäßige Ist-Daten erheben
4. **Soll-Ist-Vergleich** Beschränkung auf den Vergleich ohne vorschnell Entschlüsse zu fassen
5. **Abweichungsanalyse** Warum sind Abweichungen aufgetreten? Sind diese vernachlässigbar, systematisch?
6. **Massnahmen** Nach dem Analysieren der Abweichungen werden geeignete Massnahmen definiert
7. **Erfolgskontrolle** der eingeleiteten Massnahmen

## 6.2 Agile Projektleitung

### Unterschied Klassisch und Hybrid

Klassisch: Die Anforderung 1 wird im Sprint 1 abgehandelt. Die Anforderung 2 wird im Sprint 2 angehandelt usw...

Hybrid: Dier erst Priorisierten nehmen wir in den ersten Sprint, was übrigbleibt und nicht erledigt werden konnten, in den Backlock zurück. Grooming wieder priorisieren und wir gehen in dem nächsten Sprint.

### Was ist das Problem beim agilen Projekt-Controlling?

Gefahr: Es kann sein, dass beim letzten Sprint, die Anforderung 10 im Backlock bleibt. Die Zeit ist fix. Der Aufwand ist fix. Aber die Leistung ist nicht fix.

Grooming = Wieder Priorisieren

### Wer bestimmt was in den nächsten Sprint kommt?

Der Product Owner hat das Ziel die Anforderungen zu Priorisieren. Der Product Owner repräsentiert den Kunden, und geht zu ihm und weiss, welche Anforderungen eine Priorität haben.

Nach dem Sprint geht der Product Owner zurück zum Kunden und bespricht die Anforderungen mit dem Kunden.

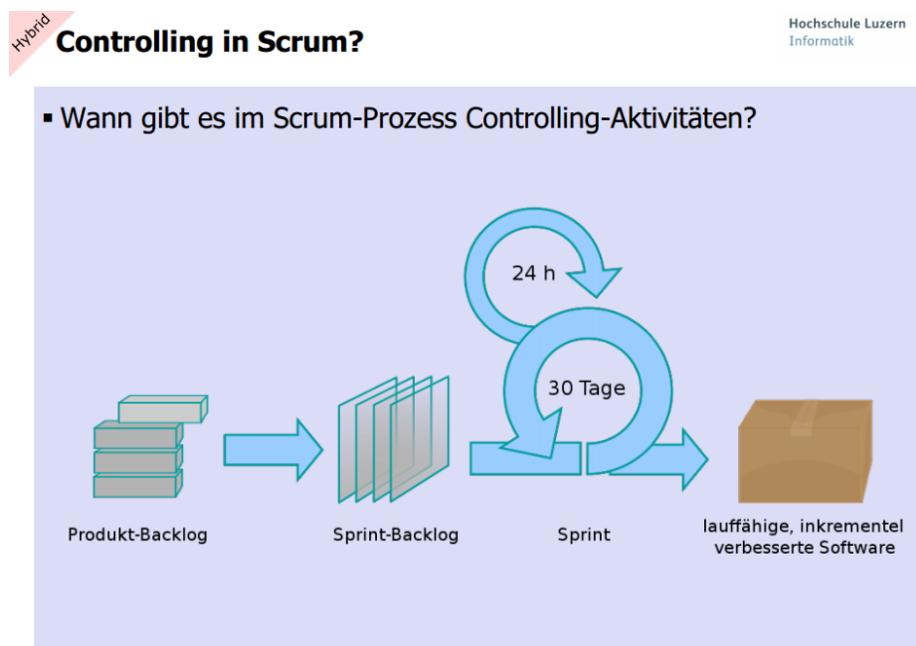


Abbildung 39: Controlling Scrum

- Nach jedem Sprint (haben wir alles erreicht?)
- Daily Meeting
- Produkt-Backlog kontrollieren (Punkteverbage pro Aufgabenstellung) Wir haben 12 Sprint und müssen somit 10 Punkte pro Sprint abarbeiten damit das Projekt fertig wird

**Was ist ein Scrum-Board? Nenne die wesentlichen Punkte.**

- Sprint Fortschrittskontrolle
- Wird beim Daily Meeting aktualisiert
- Enthält alle User-Storys welche zu diesem Sprint gehören, enthält den Fertigstellungsgrad einer User-Story

**Was ist eine Burn-Down-Chart? Nenne die wesentlichen Punkte.**

- Eine Sprint-Fortschrittsanalyse
- Aufzeigen wie viele Arbeit bis zum Sprint-Ende noch geleistet werden muss
- X-Achse Tage bis zum Sprint Ende, Y-Achse zu leistende Arbeitsstunden
- Basiert auf RAS und Schätzungen der Sprint-Planung

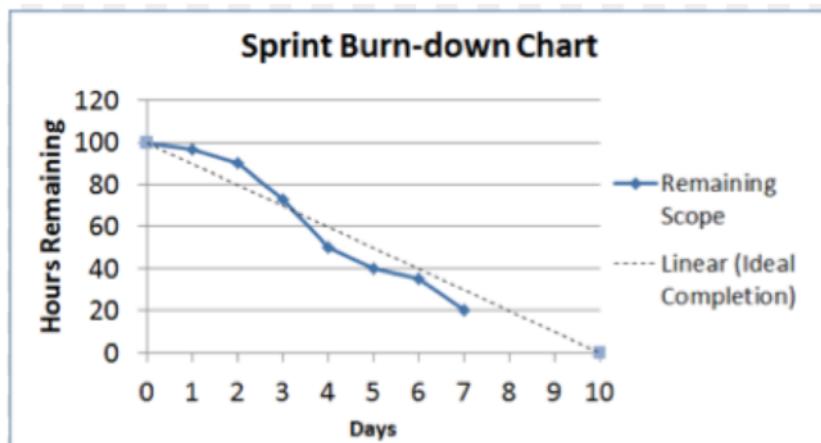


Abbildung 40: Sprint Burn-down Chart

**Was ist ein Sprint-Review-Meeting?**

- Nach jedem Sprint werden die Sprint-Ergebnisse validiert
- Ganzes Scrum-Team nimmt teil
- Product-Owner passt das Produkte-Backlog an wenn nötig (Grooming)

**Sprint-Retroperspektive**

- KVP (Kontinuierlicher Verbesserungsprozess)

**Problematik Restaufwand und agiles Projekt**

- In einem agilen Projekt lässt sich der Restaufwand nur anhand des aktuellen Sprints klar bestimmen. Das Produkte-Backlog kann sich ja ändern innerhalb des Sprints
- Deshalb muss nach jedem Sprint-Ende die noch zu realisierenden Produkt-Backlog-Items ins eine Burndown Chart eingetragen werden.

## **Projektcontrolling - Zusammenfassung**

Hochschule Luzern  
Informatik

- klassisches Projektcontrolling
  - Ist das Projekt in der Zeit?
  - Kann das Budget eingehalten werden?
  - Sind die nächsten Meilensteine erreichbar?
- Agiles Projektcontrolling
  - Was sind die wichtigsten und wertvollsten Anforderungen?
  - Haben wir diese richtig umgesetzt?
  - Was ist im nächsten Produktinkrement neu enthalten?
  - Können wir das Produkt in den noch verbleibenden Sprints fertig entwickeln? Wie viele Sprints braucht es noch?



Abbildung 41: Projektcontrolling Zusammenfassung

## **7 Business Case**

### **Was ist ein Business Case?**

Ein Business Case ist eine Entscheidungsvorlage für ein Vorgaben, die eine qualitative und quantitative Begründung für ein Vorhaben liefert.

Ein Business Case untersucht ein bestimmtes Geschäftsszenario hinsichtlich dessen Rentabilität einer Investitionsmöglichkeit. Er dient zur Darstellung und Abwägung der prognostizierten finanziellen und strategischen Auswirkungen der Investition.

### **Aus welchen zwei Teilen besteht ein Business-Case?**

- Sachliche (qualitative) Begründung eines Vorhabens
  - Umfang und Abgrenzung
  - Kosten für Anpassungen

- Projektkosten
- Management Kosten
- Wirtschaftliche (quantitativ) Begründung des Vorhabens
  - Welchen Nutzen und Kosten bringt mir dieses Projekt?
  - Erparnisse / Verkürzung Time-to-Market
  - Kann weggelassen werden, wenn eine neue Geschäftsidee umgesetzt werden soll. Bsp. Online Werbung Anfang der 90er Jahre. Es gab noch keine Erfahrungen auf diesem Gebiet, deshalb kann der Nutzen auch nicht beurteilt werden. Oder beispielsweise lässt sich nur schwer messe, wie viele % meiner Kunden nach dem Projekt glücklicher sein werden.

**Welche Hauptnutzenarten für IT-Systeme gibt es:**

Zeitersparnis, Effizienz, Übersicht, Ablösung anderer Versionen, Automatisieren, Schnittstellen (neue Schnittstellen einführen)

**Welche Hauptkostenarten für IT-Systeme gibt es?**

Wartung, Entwicklung, Lizizenzen, Einführung, Schulung und Infrastruktur

**Wie ist ein Business Case gegliedert?**

1. Einführung und Überblick (Thematik und Zielsetzung)
2. Nutzenargumentation in Prosa, Umfang und Abgrenzung
3. Optional: Risiko- und Sensitivitätsanalyse
 

qualitative Begründung
4. Annahmen und Methoden (Wichtigste Annahmen, Finanzkennzahlen, Kostenmodell)
5. Betriebswirtschaftliche Auswirkungen (Cash-Flow-Berechnung, Ergebnisanalyse)
 

quantitative Begründung
6. Fazit und Zusammenfassung



Abbildung 42: Business Case Gliederung

**Nennen Sie eine Methode zur Wertbestimmung einer Investition**

- Kapitalwertmethode
- Cashflows die in der Zukunft liegen werden auf heute zurück gerechnet (diskontiert)

## Sample Business case template

- Migration of technology stacks is an IT ask
  - Challenge: Creating a business case with quantitative and qualitative benefits/outcomes with supporting data and a cost model with the projected cost savings.
- Core value proposition: Increase the availability and resiliency and reduce development and operational costs.
- Our Business case:
  - Quantitative:
    - Reduce ongoing development and operational costs by 15%
    - Reduce hardware costs by 30%
    - Reduce downtime support requests by 25%
    - Reduce software costs by 40%
  - Qualitative:
    - Improve reliability, availability and scale of customer-facing portals
    - Standardize technology stack with a full fledged integrated platform
    - Increase agility, productivity and reusability
    - Embrace open source with abundant skill-set availability
  - Strategic:
    - *Cloud Application Platform – Start the journey to the PaaS*



7

Confidential

vmware®

Abbildung 43: Business Case Beispiel