

Lasse Dörjer (1583997)
Jannes Lensch (1556567)
Yunus Ahmad (1498950)

Aufgabe 6.5)

- (a) **Projektdurchführungsstrategie:** Legt die Reihenfolge fest, in der die für das Projekt relevanten Entscheidungspunkte durchlaufen werden müssen. Sie bestimmt sich anhand der Auswahl einer Projekttypvariante und der Belegung aller bedingter Projektmerkmale
- (b) In einem **Entscheidungspunkt** wird über das Erreichen einer Projektfortschrittsstufe entschieden. Diese Entscheidung wird auf Basis der zum Entscheidungspunkt vorzulegenden, fertig gestellten Produkte getroffen. Die Reihenfolge, in welcher die Entscheidungspunkte im Rahmen eines Projekts durchlaufen werden müssen, wird in der Projektdurchführungsstrategie festgelegt.
- (c) **Vorgehensbausteine** fassen Produkte, Themen und Aktivitäten eines Teilprozesses zusammen. Die Vorgehensbausteine Projektmanagement, Qualitätssicherung, Konfigurationsmanagement sowie Problem und Änderungsmanagement bilden den V-Modell-Kern, der in jedem Projekt enthalten ist. Dieser Kern wird im V-Modell XT Bund stets um den Vorgehensbaustein Wirtschaftlichkeitsbetrachtung erweitert. Alle anderen Vorgehensbausteine werden abhängig vom Ergebnis des Tailorings (Anwendungsprofil) hinzugefügt.
- (d) **Tailoring** bedeutet im Kontext des V-Modells nicht nur das "Wegschneiden" von Teilen, sondern auch das "Anpassen" des V-Modells. Die Anpassung des VModells an ein konkretes Projekt erfolgt im Normalfall über Hinzunehmen von Vorgehensbausteinen. Anpassungen innerhalb von Vorgehensbausteinen sind als Ausnahmefall anzusehen. Zusätzlich zur Auswahl der Vorgehensbausteine werden dabei die Projektdurchführungsstrategien ermittelt. Die Basis für die Auswahl der Vorgehensbausteine und der Projektdurchführungsstrategie bildet die Festlegung des Projekttyps und die Auswahl einer Projekttypvariante. Je nach Projektfortschritt wird zwischen
 - **statisches Tailoring**, das heißt Tailoring während der Projektinitialisierung und
 - **dynamisches Tailoring**, das heißt Tailoring im weiteren Projektverlauf unterschieden

Aufgabe 6.6)

- Prinzip 1: Nach jedem erreichten Meilenstein dem Kunden z.B. den funktionierenden Prototypen vorstellen
 - ➔ **Wichtigsten Use Cases als erstes Implementieren**
- Prinzip 2: Möglichst immer mehr Zeit geben als nötig als Puffer für z.B. späte Änderungen
- Prinzip 3: Nach jedem erreichten Meilenstein dem Kunden z.B. den funktionierenden Prototypen vorstellen
- Prinzip 4: Ein/e Kundenmitarbeiter*in muss im selben Raum wie Projektteam arbeiten
- Prinzip 5: Mitarbeiter mit Bedacht wählen
- Prinzip 6: Kein Home-Office → Ein Gebäude mit allein Mitarbeitern + Kundenvertretung
Daily Scrums oder Scrum Reviews
- Prinzip 7: Vorschritt des Projekts anhand der bereits fertigen Software messen z.B. durch vermehrte Tests
- Prinzip 8: Feste Arbeitszeiten
- Prinzip 9: Clean Code, Refactoring, Design vor der Implementierung festlegen und an diese halten, automatische Codechecks

Prinzip 10: Priorisierung der wichtigsten Aufgaben und diese als erstes abarbeiten, CI/CD

Prinzip 11: Kanban nutzen, Flache Hierarchien

Prinzip 12: Meetings durchführen in wiederkehrenden Intervallen z.B. alle 2 Wochen

Aufgabe 6.7)

- a) **Timeboxing:** Zeitabschnitte, die nicht überschritten werden sollen, in denen Meetings/Inkrementale ablaufen. Ziel ist eine begrenzte verfügbare Zeit für eine bestimmte Aktivität festzulegen, um die Effizienz des Projekts zu erhöhen bzw. den Projektvorschritt/Ablauf strukturiert zu planen.
Ubuntu nutzt timeboxing
- b) **Task-boxing:** Beschränkt die Anzahl der Aufgaben, an denen gleichzeitig gearbeitet oder die gleichzeitig ausgeliefert werden können. Sorgt für eine gleichmäßige Verteilung der Aufgaben und sorgt für eine erhöhte Effizienz, da sich nicht laufend neue Aufgaben genommen werden können, sondern eine Maximale Anzahl an Aufgaben vollständig abgeschlossen werden müssen bevor neue Aufgaben begonnen werden können. Es können zum Beispiel nur 3 Aufgaben gleichzeitig im Development sein, so kommt es nicht zu einer Überproduktion an Aufgaben. Es entsteht ein Produktionsfluss → Effizienz
- c) **Sprint Burndown Chart:**
- Zeigt Tagesaktuell die bereits geleisteten Aufwände bzw. noch zu leistende Aufwände im Allgemeinen pro Sprint
 - Nicht Aufgaben spezifisch sondern bezogen auf dem Gesamtaufwand des aktuellen Sprints
 - Dient der Planung des aktuellen Sprints. Liegt man noch im Zeitplan? (Timeboxing)
- Kanban Board:**
- Nicht einem bestimmten Sprint zugeordnet, sondern dem gesamten Projekt
 - Zeigt den aktuellen Zustand des Projekts bzw. die verschiedenen Arbeitsabläufe
 - Teilt Aufgaben zusätzlich in verschiedene Arbeitsschritte auf
 - Dient der Planung des Gesamtprojekts
- d) **Gemeinsamkeiten**
- Dienen beide der Planung von Aufwänden/Zeit etc.
 - Beide nutzen das Pull-Prinzip
 - Selbstorganisiertes/Autonomes arbeiten
 - Erschaffen visuellen Überblick über das Projekt
 - Beide Agil
 - Sich selbst optimierende Prozesse
 - Regelmäßige Meetings
 - Beide Inkrementelle Planung

Aufgabe 6.8)

Probleme: Unerfahrenheit + Unterschiede in der Erfahrung, Overkill für ein Studentisches Projekt (Andererseits muss man es auch irgendwie lernen/üben), Nicht realitätsnahe z.B. keinen richtigen Auftraggeber und jeder Student hat nicht 40 Stunden pro Woche Zeit für das Projekt, Tägliche Meetings nicht möglich, Leute springen ab, keine festen Arbeitszeiten

Direkt verwendbar: Backlogs → Sprints (Ziel vielleicht Nicht immer erreicht da jeder Student noch andere Fächer hat etc. → Keine 40 Stunden pro Woche pro Studenten), Sprint Reviews/Retroperspektive, Produkt Inkrement pro Sprint (nicht so groß wie in Realität aber zumindest irgendwas lieferbar)