Freie Universität Berlin, Institut für Informatik, Arbeitsgruppe Software Engineering
Lutz Prechelt, Linus Ververs, Nils Goldmann, Alexander Stage

Übungsblatt 12 **Prozessmodelle & Projektmanagement** zum 2025-07-07

Aufgabe 12-1: Auswahl von Prozessmodellen

a) Fassen Sie noch einmal kurz die Unterschiede zwischen den drei Prozessmodellarten *wasserfallartig*, *evolutionär*, und *inkrementell* zusammen.

b) Wählen Sie für jedes der folgenden zu bauenden Systeme die am besten geeignete Prozessmodellart (aus den in **a)** genannten) und begründen Sie Ihre Wahl.

1. Ein Terminal als digitaler, interaktiver Ersatz für Papierfahrpläne in größeren Bahnhöfen (für Ankunft- und Abfahrtszeiten).
2. Eine Steuerungseinheit für das Antiblockiersystem (ABS) in PKWs.
3. Ein System zur Verwaltung von Lehrveranstaltungen, wie etwa unser KVV.

Aufgabe 12-2: Wasserfall vs. Agile Prozesse

a) In den letzten fünfzehn Jahren wurde das Wasserfallmodell stark kritisiert und mehr „Agilität“ gefordert. Zuvor war jedoch ein wasserfallartiges Vorgehen (fast) immer als das ideale Vorgehen beschrieben worden. Erklärung Sie diese Entwicklung:

1. Warum kann man das Wasserfallmodell einerseits durchaus als *ideal* bezeichnen?
2. Und warum hat man sich andererseits dennoch davon gelöst? Nennen Sie mindestens zwei Punkte, an denen Wasserfall-Projekte scheitern können, und die bei agilen Projekten zumindest stark abgefedert werden.

b) Füllen Sie den folgenden Lückentext mit passenden Begriffen. Falls nötig, nutzen Sie dazu die auf der Vorlesungswebseite angegebenen Quellen.

Zeitlich ist ein Projekt, das mit Extreme Programming (Abk. _____) durchgeführt wird, in _____ eingeteilt. Diese enden jeweils mit einer neuen Version des Softwaresystems. Am _____ einer jeden Iteration besprechen der Kunde und die Entwickler gemeinsam, welche Funktionalitäten realisiert werden sollen. Der Kunde formuliert dabei seine Wünsche auf den _____ (engl.). Während der gesamten Entwicklung ist der Kunde _____. Er definiert zudem _____, um am _____ jeder Iteration die Funktionalität testen zu können. Für die Entwickler gibt Extreme Programming zusätzlich eine Reihe von Praktiken vor, wie etwa _____, _____ oder die gemeinsame Verantwortung.

Aufgabe 12-3: Projektplanung ★

Sie leiten ein kleines Software-Unternehmen und haben jüngst einen Auftrag über eine Produktentwicklung sichern können. Bei diesem Projekt handelt es sich um die Entwicklung einer cloud-basierten Smartphone-App, die Sie komplett ohne externe Dienstleister entwickeln wollen.

Ihr Unternehmen ist noch relativ jung, dafür haben Sie aber eine Reihe teamfähiger Mitarbeiter/innen, die in der Lage sind, miteinander und in jedem Aufgabengebiet produktiv zu arbeiten.

Die für Planung wichtige Aufgabenzerlegung und -abschätzung (in Wochen) haben Sie bereits abgeschlossen:

ID	Beschreibung	Dauer	Vorgänger	erforderliche Personen
A	Anforderungserhebung Server	3		2
B	Anforderungserhebung Client	2		2
C	Implementierung Client	6	B	1
D	Implementierung Server	3	A	1
E	Testpersonen rekrutieren	3		1
F	Server-Hardware beschaffen	2		2
G	Nutzertests und Nachbesserungen	4	C, E	2
H	Lasttest auf echter Hardware	4	D, F	1
I	Deployment automatisieren	2	H	1
J	Going Live	3	G, I	1

Die Aufgaben sollen hier nicht weiter aufgeteilt werden (weder inhaltlich noch zeitlich); jede/r Entwickler/in arbeitet an höchstens einer Aufgabe. Aufgaben mit können nicht mit weniger als den „erforderlichen Personen“ begonnen und bearbeitet werden; zusätzliche Personen beschleunigen die Arbeit nicht.

Nun geht es um die Zeit- und Ressourcenplanung.

a) Erstellen Sie einen *Netzplan*, der die logischen Abhängigkeiten der Aufgaben untereinander sichtbar macht. Stellen Sie jede der o.g. Aufgaben als Rechteck dar und tragen Sie die *ID*, die *Aufgabendauer* und den *frühstmöglichen* Start ein.

b) Erstellen Sie ein *Gantt-Chart*, das die zeitlichen Abhängigkeiten der Aufgaben sichtbar macht.

c) Ermitteln Sie den/die *kritischen Pfad/e*, die *kürzeste Projektlaufzeit*, und für alle Aktivitäten jeweils die *Pufferzeit (slack time)*.

Nachdem Sie sich einen Überblick über die logischen Abhängigkeiten verschafft haben, wenden Sie sich nun den „Ressourcen“, in diesem Falle also dem Personal, zu.

Im Gegensatz zu Netzplänen und Gantt-Charts geht es hier um eine konkrete Zuweisung von Aktivitäten auf Entwickler/innen. Um den Überblick nicht zu verlieren, sollten Sie eine geeignete Darstellung wählen, etwa eine wie in Abb. 1.

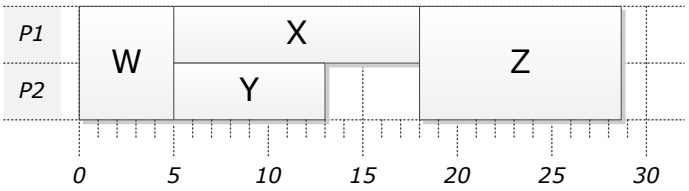

Abbildung 1: Grafische Darstellung einer Aufgabenverteilung bei zwei Personen: Zunächst arbeiten P1 und P2 gemeinsam an Aufgabe W, bevor sie sich dann für die Aufgaben X und Y aufteilen. Aufgabe Y dauert insgesamt kürzer als Aufgabe X, allerdings kann P2 noch nicht mit Aufgabe Z anfangen, weil diese wieder zwei Personen erfordert.

Bearbeiten Sie folgende Aufgaben:

d) Nehmen Sie hypothetisch an, dass Sie den Projektablauf (bei gegebenen Abhängigkeiten) bestmöglich parallelisieren wollen, also die Projektlaufzeit minimieren wollen. Wie hoch wäre der Personalbedarf P_{max}, also die größte sinnvolle Teamgröße ab der weitere Personen keine Beschleunigung mehr ergeben?

Stellen Sie eine mögliche Aufgabenverteilung für $n = P_{max}$ graphisch dar (etwa wie in Abb. 1). Geben Sie für diese Verteilung auch jeweils die Pufferzeiten aller Aufgaben, sowie die Projektlaufzeit an.

e) Nehmen Sie an, Sie könnten nur zwei Entwickler/innen für dieses Projekt abstellen. Was wäre hier die kürzeste Projektlaufzeit?

Stellen Sie eine mögliche Aufgabenverteilung für $n = 2$ graphisch dar. Geben Sie auch für diese Verteilung die Pufferzeiten und die Gesamtlaufzeit an.

f) Stellen Sie Aufgabenverteilungen für alle weiteren möglichen Teamgrößen n (d.h. also $2 < n < P_{max}$) graphisch dar, sodass jeweils die Projektlaufzeit möglichst kurz ist. Geben Sie wiederum für jede Verteilung die Projektlaufzeit und die Pufferzeiten an.

g) Betrachten Sie Ihre möglichen Projektabläufe (also für alle Teamgrößen von 2 bis einschließlich P_{max}) und fällen Sie eine Entscheidung: Wie viele Entwickler/innen setzen Sie nun auf dieses Projekt an?

Erläutern Sie Ihre Entscheidung: Vergleichen Sie Ihre Optionen explizit nach mindestens drei Gesichtspunkten.