

Dies ist keine „Musterlösung“, sondern eine gute von vielen möglichen Lösungen. Kommentare, die nicht Teil der Lösung sind, sind kursiv gesetzt.

Aufgabe 12-1:

- a. Prozess ist die Summe der Aktivitäten zur Erstellung eines Softwareprodukts, Prozessmodell ist eine Abstraktion davon (d.h. ein vereinfachtes Modell, wie ein Prozess grob aussieht oder aussehen soll) und Prozessmanagement ist die Anpassung eines Prozesses an aktuelle Umstände (Prozessverbesserung) oder an die Einhaltung eines vorgeschriebenen Prozessmodells (Prozesseinhaltung)
- b. Schon ab Stufe 3 ist z.B. der Umfang der Testphase abhängig von aktuellen Testmaßen. Richtig gesteuert wird der Prozess aber erst ab Stufe 4.
- c. Für TQM hat der Qualitätsaspekt für jede Tätigkeit jedes einzelnen Mitarbeiters oberste Priorität. Ziel ist es, den Kunden (das kann auch der eigene Kollege sein!) zufrieden zu stellen.
- d. Evolutionär wächst eine Software, wenn sie an allen Stellen/Teilfunktionalitäten stetig wächst. Inkrementell wächst eine Software, wenn erst die eine Funktionalität komplett und dann die andere komplett realisiert wird. (Beispiel: Menschen wachsen evolutionär, Überraschungseierfiguren beim Zusammenstecken inkrementell). *Beide nennt man iterativ (als Oberbegriff).*
- e. Prototypen werden gebaut, um das Risiko einer funktionalen oder nicht-funktionalen Anforderung abschätzen zu können: Diese Anforderung wird isoliert realisiert. *Danach wird der Prototyp (der Theorie nach) weggeworfen!*
- f. Die Phasen sind dann gleich den Workflows (=Aktivitäten) ohne Iteration.
- g. Sie verdeutlicht, dass auf jeder Ebene die Modelle links herangezogen werden, um die Realisierungen rechts zu validieren.
- h. Das grundsätzlich iterative Vorgehen (d.h. immer wieder Zwischenversionen) und die ständige Rückmeldung mit dem Kunden (durch Präsentation der Zwischenversionen)
- i. Manche extreme Verhaltensweisen sind nicht mal so locker durchzuführen, z.B. Test First oder Pair Programming oder Refactoring. Typischerweise ist man zu faul für solche Praktiken.

Aufgabe 12-2:

- a. Hier ist das Wasserfallmodell geeignet, denn es ist eine enge Koordination mit Technik nötig (also Planbarkeit). Die Anforderungen bei diesem eher kleinen, kritischen System können klar definiert werden, so dass die Nachteile eines Wasserfallmodells kaum auftreten.
- b. Ein evolutionäres Vorgehen ist hier angesagt, da sich die Anforderungen mit der Zeit vermutlich immer mal wieder ändern werden, d.h. insbesondere soll die Software vermutlich mit der Zeit immer mehr können (und nicht nur Prüfungsverwaltung).
- c. Ein inkrementelles Vorgehen wäre hier in sofern angesagt, als das ein hohes Risiko die Bedienbarkeit des Systems ist. Also wird man erstmal die Benutzungsoberfläche bauen und sie ausführlich testen wollen.

Aufgabe 12-3:

Ein Projekt, das mit Extreme Programming (Abk. XP), durchgeführt wird, läuft grundsätzlich in Iterationen ab. Jede Iteration endet mit einer neuen Softwareversion, wobei einige neue Funktionalitäten zu dem System hinzugefügt wurden. Welche dies sind, schreibt der Kunde auf so genannte Story Cards (engl.). Am Anfang einer jeden Iteration besprechen der Kunde und die Entwickler, welche Funktionalitäten in ihr realisiert werden sollen. Auch während der Entwicklung ist der Kunde immer ansprechbar. Er definiert zudem Akzeptanztestfälle, um am Ende jeder Iteration die Funktionalität zu testen. Auf Entwicklerseite gibt es auch einige besondere Praktiken, z.B. das frühe Testen, die Paarprogrammierung oder die gemeinsame Verantwortung. *Andere Praktiken außer frühes Testen oder Paarprogrammierung sind möglich.*

Aufgabe 12-4:

Das Wasserfallmodell ist eigentlich ideal: Man kann alles gut Planen, denn die Anforderungen sind fest, daraufhin kann man eine Systemzerlegung machen und die Aufwände abschätzen, die Ressourcen planen und ein Festpreisangebot machen. Das Vorbild war z.B. der Hausbau, wo dies üblich ist. Mit der Zeit ist man vom Wasserfallmodell abgerückt, weil es schlicht nicht funktionierte, denn (a) sind unsere Aufwandsabschätzungsmethoden zu ungenau und (b) ändern sich die Anforderungen (ganz im Sinne der Kunden!) ständig (zumindest werden sie präzisiert). Deswegen wurde man agiler und sagte: O.k., lass es uns agil machen, d.h. ich kann Dir kein Festpreisangebot machen (nur einen Kostenvoranschlag), dafür hast Du Kunde aber noch viel Einfluss auf den Verlauf des Projektes.