

Thema, Ziele: Vorgehensmodelle

Aufgabe 1: Verständnisfragen

- a) Erklären Sie, warum das Original-Wasserfallmodell in der Praxis nicht funktioniert.
(In welchen speziellen Fällen kann es doch funktionieren?)
- b) Erklären Sie den/die Unterschied(e) zwischen dem modifizierten Wasserfallmodell (mit Rückweg zur vorhergehenden Phase) und den anderen iterativen und inkrementellen Vorgehensmodellen z.B. Spiralmodell.
Zeichnen Sie drei mögliche und naheliegende Wege auf für die Tätigkeiten a, b, c, d.
- c) Bei den Vorgehensmodellen spricht man auch von "schwergewichtigen" und "leichtgewichtigen" Prozessen. Was ist da gemeint? Nennen Sie dazu jeweils mindestens ein Vorgehensmodell.
- d) Ist eine Kombination von agilen und klassischen Vorgehensmodellen möglich? Wenn ja, auf welche Art und Weise? Wenn nein, warum nicht?
- e) Angenommen die Kombination von agilen und klassischen Vorgehensmodellen aus Aufgabe d) ist möglich. Zu welchem Zeitpunkt sollen welche Anforderungen definiert sein und wieso?
- f) Nicht alle Vorgehensmodelle eignen sich gleich gut für verschiedene Vertragsformen. Welche Vor- und Nachteile haben bestimmte Vorgehensmodelle mit einem Festpreis-Vertrag?
- g) Erklären Sie, was man unter dem Begriff "Big Bang" bei Software Projekten versteht.
Bei welcher Art von Vorgehen kann es diesen "Big Bang" Effekt am ehesten geben?
- h) Bei welchen Vorgehensmodellen kann die objektorientierte Software-Entwicklung eingesetzt werden?
- i) In der nächsten Aufgabe wird von einem Projektauftrag gesprochen. In welcher Projektphase befindet sich demnach das Projekt?

Aufgabe 2: Projekteinstieg

Bilden Sie ein Team, welches bis zum Ende des Semesters an folgendem Projekt beschäftigt sein wird. Die Aufgabenstellung sowie weitere Informationen zum Projekt finden Sie auf dem Skripte Server unter dem Ordner *910-Projekt/Projektauftrag.pdf*. Lesen Sie dieses Dokument durch und überlegen Sie Ihr Vorgehen. Bei Fragen können Sie sich an den Praktikumsleiter wenden.

Anschliessend können die gebildeten Teams dem Praktikumsleiter gemeldet werden.

Setzen Sie Ihre Entwicklungsumgebung auf, damit Sie in der folgenden Woche ohne Verzögerung mit dem Projekt beginnen können.

Die folgende Liste von Werkzeugen / Tools kann als Checkliste dienen:

- Projektplanungstool, bevorzugt MS Project oder MS Excel (mindestens einer aus der Gruppe)
- Qt Umgebung*
 - inklusive Qt Test*
 - inklusive googletest*
- Git*
- Git Repo erstellt und geklont
 - Workflow-Strategie festgelegt
- Doxygen*
- MS Word oder LaTeX

* ist bereits auf dem Linux-Image installiert

Aufgabe 3: grobe Anforderungen ermitteln

Wenn wir uns an den Projektphasen orientieren wäre die nächste Phase die Projektdefinition. Diese Phasen sind jedoch eher als grobes Raster und nicht exakte Wissenschaft zu verstehen. D.h. die Phasen können sich überlappen.

Zu diesem Zeitpunkt des Projekts sollen die groben Anforderungen ermittelt werden. Einige Anforderungen sollten aus dem Projektauftrag ersichtlich sein, andere müssen selber ergänzt werden.

Die ermittelten Anforderungen geben dem Entwickler ein besseres Bild des Produktziels und werden auch für die Aufwandschätzung genutzt. Diese Aufwandschätzungen wiederum können zugleich für ein Projektangebot genutzt werden, falls z.B. ein Festpreis-Projekt ausgeschrieben ist.

Das Einarbeiten in fachliche Themen kann zu diesem Zeitpunkt durchaus Sinn machen. Gerade bei Themen in denen die Erfahrung fehlt oder bereits kritische, komplexe Teile des Produkts bekannt sind ist eine Einarbeitung zu empfehlen. Ein gängiges Verfahren dazu ist z.B. *Rapid Prototyping* (siehe Broy Kap. 4.2.3.2).

Ermitteln Sie aufgrund des Projektauftrags die groben Anforderungen. Erstellen Sie dazu ein Kontextdiagramm mittels UML Use-Case Diagramm. Die Theorie zu Use-Case Diagrammen ist im PDF *UseCaseDiagramm.pdf* zu finden.