

Einführung in Software Engineering WS 10/11

Fachbereich Informatik

Dr. Michael Eichberg

eichberg@informatik.tu-darmstadt.de

Assistent: Ralf Mitschke

mitschke@st.informatik.tu-darmstadt.de



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Übungsblatt 2: Projektmanagement und Anforderungsanalyse

Abgabeformat: Reichen Sie ihre Lösung per SVN ein. Jede Übung muss in einem eigenen Ordner **ex<Nummer>** (<Nummer> = 01, 02, ...) in Ihrem Gruppenverzeichnis eingereicht werden. Während der Übungsbearbeitung können Sie Ihre Lösungen beliebig oft in das SVN hochladen (per Commit). Wir prüfen die Zeit der Einreichung Ihrer Lösungen unter der Benutzung des SVN Zeitstempels.

Erstellen Sie für Lösungen der Aufgaben, die keinen Quelltext erfordern, eine PDF-Datei mit dem Dateinamen **solution.pdf**. Auch UML-Diagramme die Sie erstellen, müssen als Bilder eingebunden werden. Vorgaben für Java-Aufgaben werden als Eclipse-Projekt zu Verfügung stehen. Ihr eigener Code muss entsprechend in den dafür vorgesehenen Verzeichnissen (**/src** oder **/test**) erstellt werden.

Abgabetermin: 10.11.2010 - 24:00 Uhr

Aufgabe 1 (3 Punkte)

Ziel: *Verständnis häufig auftretender Probleme beim Management von Software-Projekten*

Im SVN (<https://cage.st.informatik.tu-darmstadt.de/eise/public/lecture/additions>) befinden sich zwei Texte zum Thema Software-Projektmanagement bzw. Erfolg von Software-Projekten. Lesen Sie die Texte **“2010 IT PROJECT SUCCESS RATES”** und **“CRITICAL SUCCESS FACTORS IN SOFTWARE PROJECTS”**.

Schreiben Sie einen Essay über Ihre Erfahrungen bei von Ihnen durchgeführten Software-Projekten. Dies kann zum Beispiel das Programmierpraktikum sein, welches im Rahmen der Veranstaltung Grundzüge der Informatik I durchgeführt wurde. Andere relevante Projekte sollten mindestens einen Rahmen von einer Woche (Vollzeit) und ein Team von zwei bis drei Personen aufweisen. Ihr Essay soll das Projekt im Hinblick auf Management Faktoren wie z.B. Projektfortschritt, Teamzusammensetzung oder Projekterfolg reflektieren. Die vorgegebenen Texte bieten wichtige Anhaltspunkte, um ein Verständnis für Problematiken bei Software-Projekten zu entwickeln. Reflektieren Sie in Ihrem Essay zwei bis drei der in den Texten erwähnten Faktoren, die Ihnen wichtig erscheinen.

Ihr Essay muss mindestens 200 Worte umfassen und darf eine Obergrenze von 300 Worten nicht überschreiten. Erstellen sie nur einen Essay gemeinsam als Team und reichen sie diesen ein. Der beste Essay wird belohnt werden.

Sollten kein Mitglied Ihrer Gruppe ein Software-Projekt durchgeführt haben (auch nicht das GDI Programmierpraktikum) wenden sie sich an Ihren Tutor.

Aufgabe 2 (2 Punkte)

Ziel: Bewertung der Aussagekraft von nicht-funktionalen Anforderungen

Betrachten Sie die gegebenen nicht-funktionalen Anforderungen. Bewerten Sie ob die Anforderungen geeignet sind, um eine fundierte Aussage darüber abzugeben, wann die ausgedrückte Anforderung erfüllt ist. Begründen Sie Ihre Einschätzung kurz. Diskutieren Sie gegebenenfalls zusätzliche Rahmenbedingungen oder Angaben, die eine nicht-funktionalen Anforderung prüfbar machen.

Flashcard-Anwendung (Nicht-funktionale Anforderungen)

- (I) „Die Flashcard-Anwendung muss als Inhalt der Karteikarten alle gängigen Bildformate unterstützen.“
- (II) „Die Karteikarten können im Rahmen einer Webanwendung im Internet mit Freunden ausgetauscht werden. Die online gestellten Karten dürfen keine Schimpfwörter und Kraftausdrücke enthalten.“
- (III) „Die Flashcards können im Rahmen einer Webanwendung online gelernt werden. Das Laden der nächsten zu lernenden Flashcard aus dem Internet und die Anzeige im Browser dürfen höchstens 50 Millisekunden dauern.“
- (IV) „Die Registrierung als Benutzer für die Flashcards-Webanwendung soll einfach sein.“

Hinweis: Bei Ihrer Einschätzung der nicht-funktionalen Anforderungen ist es hilfreich, wenn Sie sich legen wie genau die gegebenen nicht-funktionalen Anforderungen sichergestellt werden können.

Aufgabe 3 (4 Punkte)

Ziel: Einordnung und Formulierung von nicht-funktionalen Anforderungen

Betrachten Sie die in der Vorlesung vorgestellte, unten stehende, Übersicht (Taxonomie) über verschiedene Kategorien von nicht-funktionalen Anforderungen.

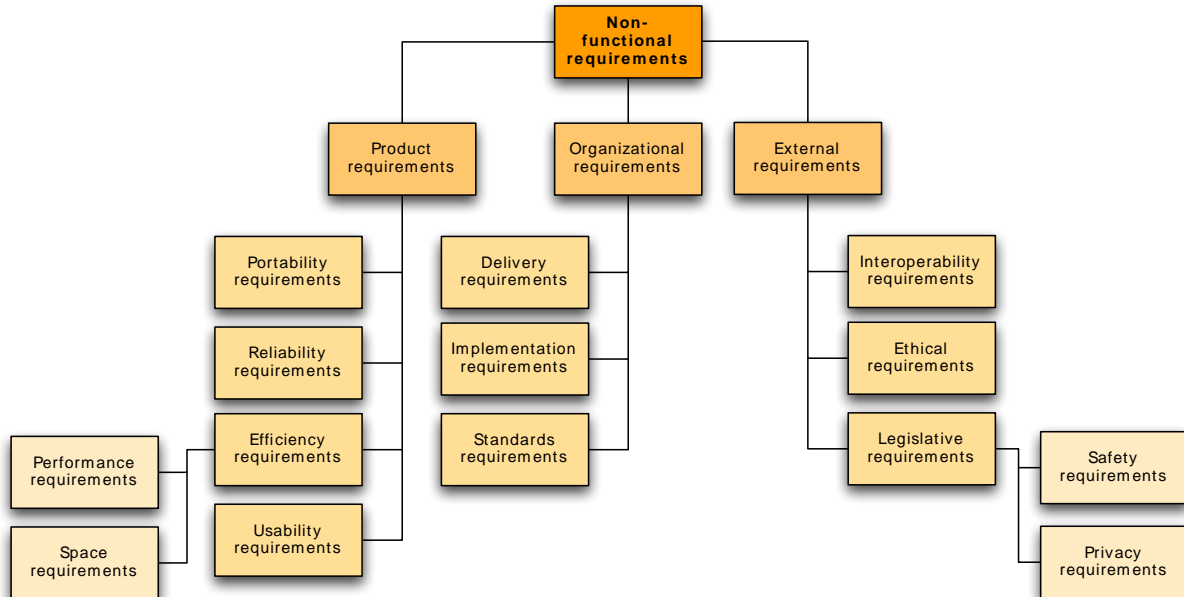


Abbildung 1: Taxonomie - nicht-funktionale Anforderungen

a) Einordnen von nicht-funktionalen Anforderungen (1 Punkt)

Ordnen Sie jeder, der in Aufgabe 2 gegebenen nicht-funktionalen Anforderungen, eine Kategorie aus der Übersicht der nicht-funktionalen Anforderungen zu.

b) Formulieren von nicht-funktionalen Anforderungen (3 Punkte)

Überlegen Sie sich zu jeder Kategorie im Bereich **Product Requirements**, **Organizational Requirements** und **External Requirements** eine sinnvolle, erfüllbare und aussagekräftige nicht-funktionale Anforderung aus der Domäne der Flashcards-Anwendung. Für Kategorien, welchen in Teilaufgabe a) bereits Anforderungen zugeordnet wurden, müssen Sie keine neue Anforderungen stellen. Es ist nicht notwendig getrennte Anforderungen für alle Kategorien der 4ten Ebene (z.B. Performance und Space Requirements) zu erstellen.

Um nicht-funktionale Anforderungen zu formulieren, können sie folgende drei Szenarien bei Ihren Überlegungen in Betracht ziehen.

Die Lernkartei als Webanwendung: Karten können auf einem zentralen Server gespeichert werden, getauscht werden und online gelernt werden

Die mobile Lernkartei auf einem Smartphone: Karten können auf dem Smartphone überall gelernt werden und Inhalte können z.B. über Portale des Smartphones gekauft werden.

Die Lernkartei als eine Anwendung der TU: Die TU bietet für alle Studierenden ein Lernkartei-Anwendung, z.B. als Webanwendung, oder als herunterladbare Desktop-Applikation.