

Übungsblatt 4

Software Engineering (WiSe 2016)

Abgabe: **So. 20.11.2016, 23:59 Uhr**

Besprechung: Mo. 28.11.2016 / Di. 29.11.2016 / Mi. 30.11.2016

Bitte lösen Sie die Übungsaufgabe in **Gruppen von 3-4 Studenten** und geben die Lösung über **Moodle** im Dateiformat **PDF** ab. Bitte erstellen Sie dazu ein **Titelblatt**, welches die Namen der Studenten und/oder die Matrikelnummern, sowie den Gruppennamen enthält. Bei eventuellen Unstimmigkeiten hinsichtlich des Abgabetermins zählt das in **Moodle** gegebene Datum. Beachten Sie, dass Sie jedes nicht-optionale Übungsblatt bearbeiten und bestehen müssen, um zur Prüfung zugelassen zu werden. Sollte eine Abgabe in **Moodle** aus *nachvollziehbaren* Gründen nicht möglich sein, schicken Sie Ihre Lösung per Mail *rechtzeitig* an **{pavesees,noller}@informatik.hu-berlin.de**. Das Eingangsdatum der Mail zählt in diesem Fall. Lösen Sie verschiedene Aufgaben auf verschiedenen Seiten.

Aufgabe 1 Diskussion: Acceptance Testing vs. Functional Testing (25 Punkte)

Der Funktionstest (functional testing) sowie der Akzeptanztest (acceptance testing) beschäftigen sich beide damit, das tatsächliche Verhalten mit einer Spezifikation zu vergleichen, welche verschiedene Formen annehmen kann (z.B. informale/formale Spezifikation oder Orakel). Dennoch sind die beide Arten von Tests *nicht* gleich und verfolgen tatsächlich nicht einmal das selbe Ziel.

Diskutieren Sie:

- Was sind die Unterschiede zwischen beiden Aktivitäten?
- Was sind die erwarteten Resultate der beiden Testaktivitäten?
- Was können diese entdecken?
- Was ist die Erkenntnis bei jeder dieser Techniken?
- Welche Arten von "Versagen" können diese Techniken aufdecken?

Aufgabe 2 Äquivalenzklassentests (30 Punkte)

Einer der Programmierer in Ihrer Gruppe wurde beauftragt ein Programm zu schreiben, das zwei Strings aus alphanumerischen Zeichen (eventuell mit Leerzeichen) zur Eingabe hat und prüfen soll, ob ein String ein echter Substring des anderen ist. Echte Substrings sind solche, die nicht trivial sind. Zum Beispiel ist **hallo** *kein* echter Substring von **hallo**, da es der gleiche String ist. Ähnlich ist somit der leere String auch kein echter Substring eines anderen Strings.

Entwickeln Sie ein Äquivalenzklassentestschema für das Testen dieser Funktion. Wie viele Tests haben Sie zum Schluss? Sind irgendwelche dieser Tests ausschließlich für den Akzeptanztest und können nicht auf das funktionale Testen angewendet werden?

Aufgabe 3 Angewendete Äquivalenzklassentests (45 Punkte)

Eine in Berlin liegende Logistik Firma hat uns kontaktiert, um ihre Webseite für sie zu entwerfen. Das Hauptziel dieser Webseite ist den Nutzer mit effizienten Routen innerhalb verschiedener Städte Europas für landgebundenen Transport zu versorgen. Die Firma nutzt nur einige der Straßen für den Transport der Güter. Der Einfachheit halber, ist eine Straße für sie eine unidirektionale Verbindung zwischen zwei Städten. Jede Straße hat eine assoziierte Distanz, finanzielle Kosten und eine Durchschnittszeit, die es braucht sie zu überqueren. Es kann mehr als eine Straße zwischen zwei Städten existieren.

Die Webseite, die entwickelt werden soll, wird die Funktionalität bereitstellen, bei zwei gegebenen Städten, die effizienteste Route anzugeben. Es gibt mehrere Maße der *Effizienz* für eine Route:

- Minimale Überquerungszeit;
- Minimale Distanz;
- Minimale finanzielle Kosten;
- Minimale *agony*, wobei *agony* eine Funktion von Zeit und Kosten ist (Distanz ist irrelevant für *agony*).

Um dieses Ziel zu erreichen, hat die Webseite zwei Hauptmodi: Bearbeitung und Abfrage. In dem Modus Bearbeitung kann die Karte bearbeitet werden, indem Routen und Städte hinzugefügt oder gelöscht werden. Das Abfrage Modell erlaubt dem Nutzer nach effizienten Routen zwischen zwei Städten zu fragen.

Entwickeln Sie ein Set von äquivalenzklassenbasierten Akzeptanztests für die angegebene Webseite. Diskutieren Sie Ihre Ergebnisse.