

Übungen zu Software Engineering

Wintersemester 2014/15

Blatt 1

Übungsbetrieb

Donnerstags in der Vorlesung wird ein Aufgabenblatt verteilt, das bis einschließlich Donnerstag (letztes Testat um 17:30 Uhr) der darauffolgenden Woche zu bearbeiten ist. Die Aufgabenblätter finden sich auch auf der Veranstaltungsw Webseite (<http://www-lehre.inf.uos.de/~swe>).

Die Übungen finden Freitags zu zwei unterschiedlichen Terminen (08:00 - 10:00 Uhr, Raum 69/118 und 10:00 - 12:00 Uhr, Raum 69/118) mit dem gleichen Inhalt statt. In den Übungen werden das neue Aufgabenblatt besprochen und die Lösungen zum alten Aufgabenblatt diskutiert.

Testatbetrieb

Begleitend zur Veranstaltung finden wöchentliche, 30-minütige Testate bei den Tutoren der Veranstaltung statt. Das aktuelle Aufgabenblatt ist bis zum Testattermin zu bearbeiten und dem jeweiligen Tutor zum entsprechenden Zeitpunkt in Raum 31/E53 vorzulegen sowie in das unten genannte Web-System hochzuladen.

Achten Sie bei der Bearbeitung der Aufgaben darauf, diese *vor* dem Testat einmal auf einem Uni-Rechner in Raum 31/E53 getestet zu haben, der zum Bearbeiten der Aufgaben zur Verfügung steht.

Die Testate erfolgen in Zweiertteams, zu denen man sich von Freitag, 17.10. 09:00 Uhr, bis Montag, 20.10. 11:00 Uhr, unter <https://swe-testate.informatik.uni-osnabrueck.de> mit dem RZ-Login anmelden kann.

Fragen, Antworten und Ankündigungen

Um Fragen oder Probleme untereinander und mit den Tutoren diskutieren zu können, steht auf der Webseite <http://www.piazza.com/uni-osnabrueck.de/fall2014/swe14> ein Q&A-System zur Verfügung. Die Nutzung wird nachdrücklich empfohlen, ist aber freiwillig. Wichtige Ankündigungen werden dort, aber auch in Stud.IP bekannt gegeben.

Scheinvergabe

Um die Zulassung zur Klausur zu erhalten, müssen alle bis auf eins der ausgegebenen Übungsblätter erfolgreich bearbeitet (mindestens 50% der Punkte) und dem Tutor präsentiert werden. Zum Abschluss der Veranstaltung entscheidet eine Klausur über die Scheinvergabe.

Bearbeitung der Aufgaben

Lösen Sie sämtliche Aufgaben immer schriftlich. Quellcode oder Speicherstände der verschiedenen Entwicklungswerkzeuge können Sie auf das Testatverwaltungstool vor dem Testat hochladen. Zeichnungen müssen im pdf-Format hochgeladen und auch ausgedruckt abgegeben werden.

Aufgabe 1.1: Grundlagen: Software-Krise und Software Engineering (40 Punkte)

Bearbeiten Sie folgende Teilaufgaben:

1. Geben Sie einen kurzen Überblick über die Ursachen, die zur Software-Krise geführt haben.
2. Erläutern Sie die Bestrebungen, die als Gegenmaßnahmen gegen die Auswirkungen der Software-Krise unternommen werden.
3. In der Vorlesung wurde als Metrik zur Bestimmung der Software-Größe LOC (*Lines of Code*) genannt. Nennen Sie Probleme, die in Zusammenhang mit dieser Maßeinheit auftreten können.

Aufgabe 1.2: Objektorientierter Entwurf (60 Punkte)

Das Zahlenrätsel Sudoku besteht aus einem Spielfeld mit 9x9 Feldern, die teilweise mit je einer Ziffer zwischen 1 und 9 gefüllt sind. Aufgabe des Spielers ist es, die übrigen leeren Felder mit Ziffern zwischen 1 und 9 zu füllen, so dass in jeder Zeile, jeder Spalte und jedem Unterfeld alle Ziffern zwischen 1 und 9 genau einmal vorkommen. Ein Unterfeld besteht aus 3x3 Feldern. Es gibt 9 disjunkte Unterfelder in einem Sudoku.

Entwerfen Sie die Geschäftslogik des Sudoku-Spiels als UML-Klassendiagramm, so dass sie in Java implementiert werden kann. Es sollen keine primitiven Datentypen oder Arrays genutzt werden. Gehen Sie dafür davon aus, dass mehrwertige Beziehungen mit Listen umgesetzt werden. Modellieren Sie nicht nur das Spielfeld, seine Zeilen, Spalten und Unterfelder, sondern auch den Spieler und die Züge des Spielers in einem Spiel. Jeder Spielzug sollte eine Referenz auf den vorherigen Spielzug besitzen, so dass Züge rückgängig gemacht werden können. In einem Spielzug soll zudem die Zahl, die gesetzt wurde, sowie das Spielfeld referenziert sein.

Für die Modellierung können Sie ein Programm Ihrer Wahl, wie beispielsweise DIA, verwenden. Auf die Angabe von Methoden und Sichtbarkeiten können Sie im gesamten Diagramm verzichten. Geben Sie aber bei jeder Beziehung Kardinalitäten, Rollen und Namen an. Machen Sie das Diagramm außerdem navigierbar.