

Übungen zur Vorlesung Softwaretechnologie

- Wintersemester 2018/19 -
Dr. Günter Kniesel

Übungsblatt 7

Zu bearbeiten bis: 30.11.2018, 16 Uhr

Bitte fangen Sie **frühzeitig** mit der Bearbeitung an, damit wir Ihnen bei Bedarf helfen können. Checken Sie die Lösungen zu den Aufgaben in Ihr Repository ein, „Erklärungen“ bitte als Textdatei. Fragen zu Übungsaufgaben/Vorlesung können Sie auf der Mailingliste swt-tutoren@lists.iai.uni-bonn.de, bzw. swt-vorlesung@lists.iai.uni-bonn.de stellen.

Aufgabe 1. *Design by Contract* (11 Punkte)

- a) (1,5 Punkte) Erläutern Sie die drei Kernkonzepte von „Design by Contract“.
- b) (1,5 Punkte) Wie können diese Konzepte in Java durch die Verwendung von `assert()`-Anweisungen¹ umgesetzt werden? Beschreiben Sie zu jedem Konzept kurz die Umsetzung.
- c) (1 Punkt) Wie würde man mit einem `AssertionError` umgehen, der im Fehlerfall ausgelöst wird? Würde man ihn abfangen und falls ja, wo? (Direkt nach der Assertion oder an der Stelle, an der die Methode aufgerufen wird, in der sich die Assertion befindet?) Begründen Sie Ihre Antwort.
- d) (4 Punkte) Implementieren Sie für den Entwurf auf der letzten Seite des Übungsblatts die Methoden
 - o `Veranstaltung.addStudent(Student s)`,
 - o `Veranstaltung.removeStudent(Student s)` und
 - o `Veranstaltungsangebot.eintragen(Veranstaltung v)`.

Garantieren Sie dabei durch die Verwendung geeigneter `assert`-Ausdrücke, dass:

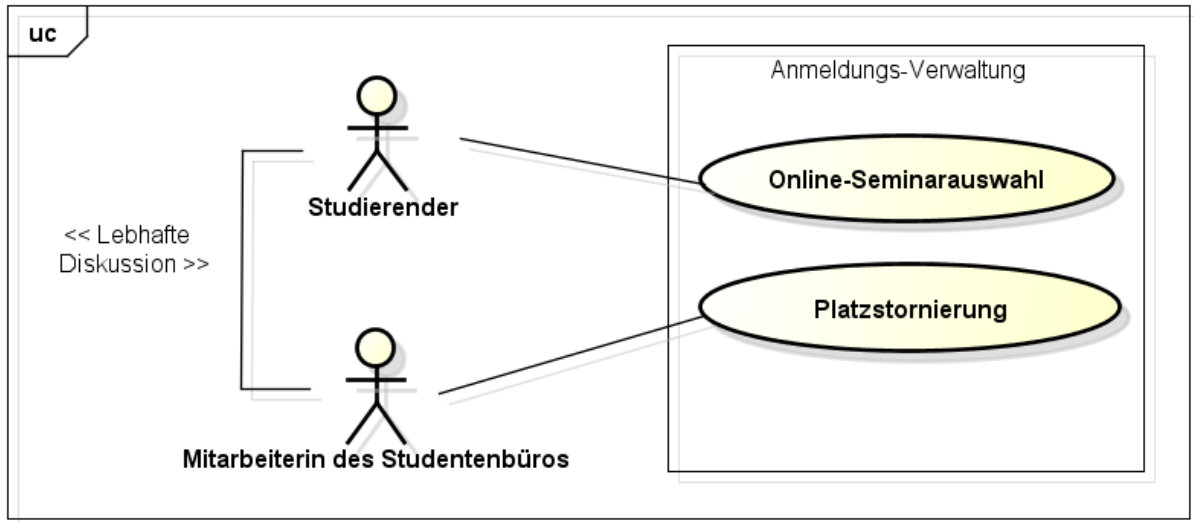
- a. ein Student nicht mehrfach als Teilnehmer einer Veranstaltung eingetragen sein kann,
 - b. ein Student sich nur von einer Veranstaltung abmelden kann, für die er eingetragen ist,
 - c. der Titel einer Veranstaltung im System zur Raumreservierung eindeutig ist (d.h. es darf keine zwei Veranstaltungen mit demselben Titel geben).
- e) (2 Punkte) Wie würden Sie Vorbedingungen in Java ohne Assertions realisieren? Nennen Sie zwei unterschiedliche Herangehensweisen. Beschreiben Sie die Unterschiede mit Beispiel-Code unter Abwandlung eines der Codestücke aus der vorherigen Teilaufgabe.
 - f) (1 Punkt) In *Teilaufgabe e*) haben Sie festgestellt, dass es sehr einfach ist, DBC in Java zu realisieren, selbst ohne spezielle Sprachunterstützung. Weshalb haben sich Ihrer Meinung nach die Entwickler von Java trotzdem dazu entschlossen, die Sprache um Assertions zu erweitern?

1

siehe <http://docs.oracle.com/javase/7/docs/technotes/guides/language/assert.html>

Aufgabe 2. Analysemodell (10 Punkte)

Sie sind mit dem Design eines Systems zur Seminaerauswahl im Studienbüro beauftragt. Als Vorgabe erhalten Sie die zwei unten eingezeichneten Anwendungsfälle:



Die verbale und nonverbale Kommunikation zwischen Studierender und Mitarbeiterin ist hier nur durch die Beziehung <<Lebhaftes Diskussion>> angedeutet. Sie wird nicht weiter modelliert, denn sie läuft nicht über das System.

Die Beschreibungen der Anwendungsfälle finden Sie auf der nächsten Seite.

- (4 Punkte) Identifizieren Sie für jeden der beiden Anwendungsfälle jeweils Boundary-, Controller- und Entity-Objekte sowie die beteiligten Akteure. Es reicht, sie pro Anwendungsfall textuell aufzuzählen.
- (4 Punkte) Setzen Sie die Ergebnisse aus *Teilaufgabe a)* in einem Kommunikationsdiagramm um, das lediglich die beteiligten Objekte und ihre Beziehungen erfasst. Nachrichten sind in diesem Schritt nicht erforderlich.
- (2 Punkte) Erstellen Sie ein neues Diagramm in dem Sie die Analyseobjekte aus *Teilaufgabe b)*, die ähnliches tun bzw. konzeptuell identisch sind, zusammenfassen.

Hinweis: Wenn Ihnen die Vorgehensweise nicht klar ist, sehen Sie sich die Folien zur Objektmodellierung im Analyseworkflow aus der Vorlesung an.

Beschreibungen der Anwendungsfälle zu *Aufgabe 2*:

<p>Name des Anwendungsfalls: Online-Seminarplatzwahl</p> <p>Akteure: Studierender</p> <p>Anfangsbedingung: Dieser Anwendungsfall beginnt, wenn der Studierende die URL der Online-Seminarwahl aufruft.</p> <p>Ereignisfluss:</p> <ol style="list-style-type: none">1. Studierender sucht Seminar im System.2. Studierender wählt ein Seminar aus.3. System zeigt Verfügbarkeit von Plätzen für das ausgewählte Seminar an und öffnet eine Eingabemaske für die Anmelde-Daten.4. Studierender gibt Studiengang, Semesterzahl und Matrikelnummer an.5. System reserviert Platz.6. System druckt Bestätigung aus <p>Endbedingung: Studierender muss die Reservierungs-Bestätigung mitnehmen, um sie vorzeigen zu können.</p> <p>Sonderfälle: keine.</p> <p>Spezielle Anforderungen: Dem Studierenden steht ein Drucker zur Verfügung.</p>
<p>Name des Anwendungsfalls: Platzstornierung</p> <p>Akteure: Studierender, Mitarbeiterin des Studienbüros</p> <p>Anfangsbedingung: Dieser Anwendungsfall beginnt, wenn der Studierende das Studienbüro betritt. Der Studierende besitzt bereits eine Bestätigung für ein Seminar.</p> <p>Ereignisfluss:</p> <ol style="list-style-type: none">1. Mitarbeiterin nimmt die Reservierungs-Bestätigung des Studierenden entgegen.2. Mitarbeiterin sucht Seminar im Stornierungs-System.3. Mitarbeiterin wählt den Platz des Studierenden zur Freigabe aus4. Mitarbeiterin bestätigt Platz-Freigabe5. System gibt reservierten Platz wieder frei.6. Mitarbeiterin vernichtet die Reservierungs-Bestätigung. <p>Endbedingung: Reservierungs-Bestätigung ist vernichtet.</p> <p>Sonderfälle: keine.</p> <p>Spezielle Anforderungen: keine.</p>

Aufgabe 3. *Kommunikationsdiagramm* (5 Punkte)

Extrahieren Sie für den Anwendungsfall **Online-Seminarauswahl** die Nachrichten des Ereignisflusses und notieren Sie sie in einem zugehörigen Kommunikationsdiagramm. Nutzen Sie als Grundlage das Ergebnis der *Aufgabe 2c*).

Entwurf zu Aufgabe 1:

