Rheinische Friedrich-Wilhelms-Universität Bonn
Institut für Informatik III

Übungen zur Vorlesung Softwaretechnologie

- Wintersemester 2019/20 -

Dr. Günter Kniesel

Übungsblatt 6

Zu bearbeiten bis: 15.11.2019, 16 Uhr

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

Praktische Aufgaben

Aufgabe 1. *Anwendungsfalldiagramm* **15 Punkte**

Sie befinden sich im Jahr 2003 und die bekannte Spielefirma *Pipe* möchte eine Online-Vertriebsplattform namens *Ice* für Computerspiele entwickeln.
Dazu nennt Ihnen die Firma einige Punkte, die Sie gerne umsetzen möchten:

- Neue Nutzer sollen sich zuerst einloggen
- Da im Jahr 2003 die Internetverbindung noch bescheiden ist, wird im Einloggen-Fenster immer der Login-Server-Status angezeigt
- Erst nach dem Einloggen wird dem Nutzer die Shop-Startseite angezeigt
- Nach dem Login behandeln wir den Nutzer als eingeloggten Nutzer
- Eingeloggte Nutzer können den Shop durchstöbern, ihre persönliche Spielesammlung anschauen oder einfach ein bereits erworbenes Spiel spielen
- Falls der Nutzer beim Shop durchstöbern Interesse an einem Spiel hat und den *Kaufen*-Knopf drückt, so wird der Kaufprozess eingeleitet
- Der Kauf erfolgt über eine Onlinezahlung
- Der Nutzer hat die Möglichkeit ein Spiel zurückzugeben und ihm wird durch einen Administrator automatisch der Kaufpreis als Guthaben erstattet
- Sowohl die Onlinezahlung als auch die Rückerstattung sind Aktionen um den Guthabenstand zu verändern

Ihre Aufgabe ist es nun, zu den oben genannten Anforderungen ein passendes Anwendungsfalldiagramm zu erstellen.

Aufgabe 2. *Domain Object Model* **16 Punkte**

In dieser Aufgabe orientieren wir uns an Inhalten der Vorlesung *Techniken des wissenschaftlichen Arbeitens* aus dem ersten Semester.

Genauer soll folgender Sachverhalt zur Struktur einer Tagung in ein *Domain Object Model* umgesetzt werden:

- Auf Tagungen werden an einem Ort an wenigen Tagen eine Vielzahl von Vorträgen gehalten
- Eine Tagung hat eine begrenzte mögliche Teilnehmeranzahl sowie (durch zeitliche Vorgaben) nur eine begrenzte Anzahl an Vorträgen
- Diese Vorträge gehören allesamt einem bestimmten Fachgebiet an, welche das Thema der Tagung bilden
- Ein Vortrag hat einen Titel und eine Dauer, welche etwa im Zeitrahmen von 15 Minuten liegen soll
- In der Regel gibt es zu jeder Tagung einen Tagungsband, welcher zu jedem Vortrag einen tiefergehenden Forschungsartikel enthält
- Die Vorträge werden von Wissenschaftlern gehalten und von anderen Wissenschaftlern gehört
- Im Vorfeld der Tagung müssen alle Teilnehmer, die einen Vortrag halten möchten, ihren Beitrag zur Begutachtung einsenden
- Das Programmkomitee begutachtet die eingesandten Beiträge inhaltlich und entscheidet, ob diese akzeptiert, zur Revision gegeben oder abgelehnt werden
- Als weitere Formen einer Tagung gibt es Symposien und Workshops
- Ein Symposium richtet sich nach einem spezifischen Thema
- Ein Workshop hat neben einem spezifischeren Thema noch eine kleinere Teilnehmerzahl
- Zu einer Tagung gibt es ein Organisationskomitee, welches sich aus folgenden Teilen zusammensetzt:
    - Eine gesamtverantwortliche Tagungsleitung
    - Eine Programmleitung, welche für die Besetzung und Arbeit des Programmkomitees verantwortlich ist
    - Eine Leitung zur Auswahl von Workshops, welche die Haupttagung ergänzen
    - Eine Leitung für lokale Belange, welche z.B. Tagungsräume anmietet, Hotelunterbringungen durchführt usw.

Ihre Aufgabe ist nun:

a) **(9 Punkte)** Erstellen Sie eine Textanalyse nach Abbots Technik.

b) **(4 Punkte)** Entwickeln Sie nun aus Ihren Ergebnissen aus (a) ein vollparametrisiertes Klassendiagramm. Gehen Sie dabei zunächst strikt mit den Entsprechungen nach Abbot. Verbessern Sie Redundanzen oder andere Ihnen sinnlos erscheinenden Stellen noch nicht, sondern laden Sie Ihre Zwischenlösung so ins Repository!

c) **(3 Punkte)** Überarbeiten Sie nun Ihr Klassendiagramm aus (b), sodass es möglichst minimal ist und laden Sie es als separate Datei ins Repository.

Aufgabe 3. *CRC-Karten* **10 Punkte**

Wir betrachten noch mal das Beispiel aus Aufgabe 1. Wir stellen eine Klasse, die das *Programmkomitee* und eine Klasse, die die *Wissenschaftler* modellieren soll. Wir beschreiben diese Klassen nun etwas genauer:

- Ein Wissenschaftler sendet im Voraus einer Tagung seinen Beitrag, zu welchem er einen Vortrag halten will, an das Programmkomitee
- Wissenschaftler können vor und nach ihrem Vortrag den anderen Wissenschaftlern zuhören und Fragen stellen
- In den Kaffeepausen kann er zusätzlich mit den anderen Wissenschaftlern Smalltalk halten
- Im Falle, dass der Beitrag des Wissenschaftlers in Revision gegeben wurde, überarbeitet er diesen
- Das Programmkomitee überprüft alle eingesandten Beiträge und gibt diese entweder in Revision, akzeptiert sie oder lehnt sie ab
- Außerdem legt das Programmkomitee fest, in welcher Reihenfolge und zu welchen Zeiten die Vorträge gehalten werden

Erstellen Sie zu den Klassen *Wissenschaftler* und *Programmkomitee* jeweils eine CRC-Karte. Falls Sie auf neue Klassen stoßen, geben Sie eine kurze textuelle Beschreibung zu diesen an.

Theoretische Aufgaben

Aufgabe 4. *Ersetzbarkeit* **10 Punkte**

a) **(1 Punkte)** Wann ist ein Typ ein Subtyp eines anderen Typen?

b) **(1,5 Punkte)** Was bedeuten Vor- und Nachbedingungen im Kontext von statischen Programmiersprachen? Was ist der Unterschied bei DBC?

c) **(0,5 Punkte)** Was bedeutet Inhärenz?

d) **(1 Punkt)** Was ist eine Kapselungseinheit?

e) **(1 Punkte)** Was ist Information Hiding?

f) **(0,5 Punkte)** Was besagt das Gesetzt von Demeter?

g) **(2 Punkte)** Was bedeutet es, wenn eine Methode überladen bzw. überschrieben wird?

h) **(2,5 Punkte)** Nennen Sie die fünf Prinzipien zu Subtyping und Ersetzbarkeit.

Aufgabe 5. *Anforderungserhebung* **10,5 Punkte**

a) **(2 Punkte)** Geben Sie die Schritte der Anforderungserhebung wieder.
b) **(2 Punkte)** Geben Sie die wichtigsten Konzepte der Anforderungserhebung wieder.
c) **(2 Punkte)** Welche Arten von Szenarien gibt es?
d) **(3 Punkte)** Erklären Sie die Arten der Anforderungserhebung.
e) **(1,5 Punkte)** Welche Anforderungsarten gibt es? Erklären Sie ihre Bedeutung.

∑ 61,5,5 Punkte