

Softwaretechnik SoSe 2011·Übungsblatt 5: Statische und dynamische Analyse
Bearbeitung zum Tutorium in KW 20

Aufgabe 5-1 (Wissensfragen)

Erläutern Sie anhand konkreter, selbst entworfener Beispiele aus dem Bereich Gesundheitskarte die Unterschiede zwischen...

1. **Problembereichsklassen** (application domain classes) und **Lösungsklassen** (solution domain classes)
2. **Geschäftsobjekten** (entity classes), **Grenzklassen** (boundary classes) und **Steuerklassen** (control classes)

Aufgabe 5-2 (Statisches Objektmodell)

Das Statische Objektmodell beschreibt die statischen Eigenschaften des Systems z.B. durch Klassen- und Objektdiagramme.

1. Überarbeiten Sie ggf. das im letzten Tutorium basierend auf Aufgabe 4-3 erstellte Klassendiagramm Ihrer zu entwickelnden Software.
Sollten Sie in diesem Tutorium nicht anwesend gewesen sein, bearbeiten Sie vorher die folgende Aufgabe:
 - Identifizieren Sie anhand der in der Vorlesung vorgestellten Heuristiken die partizipierenden Objekte Ihres Anwendungsfalls aus Aufgabe 4-3. Bedenken Sie, dass es sich hier nur um Objekte des Anwendungsbereichs handeln soll.
 - Klassendiagramm erstellen: bilden Sie das entsprechende statische Analysemodell, indem Sie mithilfe der Methode von Abbott Klassen sowie deren Attribute und Operationen und danach deren Assoziationen identifizieren.
 - Überlegen Sie sich, zu welchem **Typ** die jeweiligen Klassen gehören (**Grenz-, Steuerungs- oder Entitätsobjekte**)
2. Entwerfen Sie ein Ihrem Klassendiagramm entsprechendes Objektdiagramm.

Notieren Sie bei der Bearbeitung von Aufgabe 5-2 und 5-3 stets um welchen Typ von Klasse / Objekt es sich jeweils handelt (Entitäts-, Grenz- oder Steuerungsobjekt). (siehe auch Hinweis und Wiederholung auf der nächsten Seite)

Aufgabe 5-3 (Dynamisches Objektmodell)

Das dynamische Objektmodell beschreibt die dynamischen Eigenschaften des Systems z.B. durch Zustands- und Sequenzdiagramme.

Ein Sequenzdiagramm

- beschreibt das dynamische Verhalten zwischen mehreren Objekten, und zwar die Reihenfolge von Nachrichten, die zwischen den Objekten ausgetauscht werden (Beispiele für dynamisches Verhalten: Interaktion zwischen Web-Browser und Web-Server über HTTP, allgemein: jedes Protokoll) *Quelle: B. Bruegge, Einführung in die Informatik II, UML*
- verbindet Anwendungsfälle mit Objekten. Es zeigt, wie das Verhalten eines Anwendungsfalls (oder Szenarios) auf die partizipierenden Objekte verteilt wird.

1. Erstellen Sie zu Ihrem Anwendungsfall aus Aufgabe 4-3 ein dynamisches Modell in Form eines Sequenzdiagramms.

Hinweis: Es ist zu beachten, dass das Objektmodell und das dynamische Modell noch keine aktuellen Softwareklassen oder -komponenten darstellen, sondern Konzepte auf Ebene des Anwendungsbereichs.

Wiederholung: Das Objektmodell besteht aus Entitäts-, Grenz- und Steuerungsobjekten.

Entitätsobjekte

- repräsentieren Informationen, die vom System verwaltet werden und die persistent sind.
- Grundfunktionalität (ändert sich eher selten)

Grenzobjekte (boundary object)

- stellen die Interaktionen zwischen den Akteuren und dem System dar.
- ändern sich häufiger
- z.B. Benutzerschnittstelle

Steuerungsobjekte

- sind verantwortlich für die Verwirklichung der Anwendungsfälle (repräsentieren Aktivitäten)
- Grundfunktionalität (ändert sich eher selten)

Diese Vorgehensweise mit drei Objekttypen ergibt kleinere und spezialisiertere Objekte. Das führt auch zu Modellen, die bei Änderungen belastbarer sind.

Es gibt jedoch keine drei verschiedenen Typen von Klassen in UML, sondern man nutzt in diesem den sog. Stereotyp - Mechanismus um zusätzliche Informationen für Klassen bereitzustellen.