
Praktikumsanleitung – Versuch 4

Modellierung von Anforderungen

Praktikumstermine: Mo, 26.04.2004 / Di, 27.04.2004 / Do, 22.04.2004

Einführung

In der Informatik gibt es verschiedene Möglichkeiten zur Darstellung von Anforderungen. Eine konstruktive Darstellung durch Modelle ist heute das Mittel der Wahl zur Spezifikation der von einem System zu verwaltenden Daten, sowie der damit verbundenen Funktionalität und des zeitlichen Verhaltens des Systems.

In diesem Versuch wird die objektorientierte Spezifikationssprache UML (Unified Modeling Language) verwendet, um die Anforderungen an das System «Börsencafé» zu modellieren.

Zunächst werden mit Hilfe von Klassenkarten (siehe Einführung CRC) die für die Systembenutzer relevanten Klassen, deren Aufgaben und Verantwortlichkeiten identifiziert. Anschliessend wird mit Hilfe der UML-Notation ein auf den Klassenkarten aufbauendes Klassenmodell entwickelt. Zum Schluss wird dieses Modell durch die Spezifikation des Systemverhaltens angereichert. Für die Erstellung des Anforderungsmodells steht Ihnen ein Entwicklungswerkzeug (together) zur Verfügung. In Versuch 5 werden die erstellten Versuchsergebnisse dann einem technischen Review durch ein anderes Team unterzogen.

Vorbereitung des Versuchs (Heimarbeit)

Neben der obligaten thematischen Vorbereitung ist eine Einarbeitung in die UML-Notation und in die Bedienung des Entwicklungswerkzeuges notwendig. Schwerpunkt hierbei sind die grundlegenden Notationselemente (Klassen, Objekte, Subsysteme, div. Verhaltensbeschreibungselemente).

Falls noch nicht in Versuch 3 erfolgt, sind die Anwendungsfälle aufgrund der Befunde des Kunden entsprechend in Heimarbeit nachzuführen.

Durchführung des Versuchs

Aufgabe 4.1: Klassenkartenanalyse

Eine Klassenkarte (engl. Classes Responsibilities and Collaborators, kurz CRC) ist ein Dokument, das für eine Klasse deren Aufgabe (Zweck), Verantwortlichkeiten und Verbindungen zu anderen Klassen innerhalb des Systems definiert. Die Beschreibung der Aufgabe und der Verantwortlichkeiten erfolgt in der Regel durch Prosatext. Identifizieren Sie in einem ersten Schritt auf Basis der aktualisierten Anwendungsfälle (Versuch 3) für das zu erstellende System relevante Klassen. Erstellen Sie anschliessend für diese Klassen jeweils eine Klassenkarte (siehe hierzu [Joos97]).

Denken Sie daran, dass Sie für die Vorführung Ihres Programms am Ende des Softwarepraktikums eine Simulationsklasse implementieren müssen, welche automatisch Bestellungen generiert!

Aufgabe 4.2: Klassenmodell

In dieser Aufgabe soll ein Klassenmodell des geplanten Systems erstellt werden (Systemmodell). Das Klassenmodell umfasst alle für das System relevanten Klassen, deren Attribute sowie die Beziehungen zwischen den Klassen.

Basis für das Klassenmodell sind die in Aufgabe 4.1 bestimmten Klassen. Es können aber noch weitere, von der Klassenkartenanalyse nicht erfasste Klassen hinzugefügt werden. Achten Sie bei der Modellierung der Beziehungen von Klassen auf die Angabe der Kardinalitäten. Grenzen Sie das System gegen seine Umgebung ab und zeigen Sie die Systemanbindung auf.

Aufgabe 4.3: Verhaltensmodell

Erstellen Sie für die zentralen Klassen des Klassenmodells ein Verhaltensmodell in Form von Zustandsautomaten (engl. state diagram) oder Statecharts (siehe [Glin96] Kapitel 4.3). Achten Sie bei der Erstellung der Zustandsautomaten / Statecharts auf eine ausreichende Dokumentation der Zustandsübergänge (Bedingung und Aktion kann wiederum mit Hilfe von Prosatext beschrieben werden).

Hinweise:

- Vom Entwicklungsteam getroffene Annahmen sind explizit anzugeben.

Abgabe

Ein (!) sinnvoll gegliedertes .doc oder .pdf Dokument "Spezifikation" enthaltend:

- CRC
- Klassenmodell des geplanten Börsencafé-Systems
- Verhaltensmodell des Börsencafé-Systems

Die Abgabe erfolgt im OLAT Gruppenordner in einem von Ihnen anzulegenden Verzeichnis mit der Bezeichnung „Versuch04_Lsg“.

Abgabetermin

- Klasse Montag: Do, 29.04.04, 24:00
- Klasse Dienstag: Fr, 30.04.04, 24:00
- Klasse Donnerstag: So, 25.04.04, 24:00

Unterlagen in OLAT:

- Intro UML (aus Versuch 2)
- Intro together (aus Versuch 2)
- Einführung in die CRC-Analyse

Referenzen:

- [Fowl97] M. Fowler, K. Scott: UML Distilled: Applying the Standard Object Modeling Language. Wokingham: Addison-Wesley.
- [Glin98] M. Glinz: Vorlesungsskript Software Engineering I, WS 98/99, Kapitel 7
- [Glin96] M. Glinz: Vorlesungsskript Informatik, Teil 2, Band 2, Modellierung, SS 96, Kapitel 4, 5, 6