

## Praktikumsanleitung – Versuch 8

### Implementierung I, Software-Pflege, Pair Programming

Praktikumstermine: Mo, 24.05.2004 / Di, 25.05.2004 / Do 27.06.2004

#### Einführung

Nachdem Sie im letzten Versuch den Detailentwurf zu einem Teil des Systems «Börsencafé» ausgearbeitet haben, beginnen Sie in diesem Versuch mit der Implementierung der Börsencafé-Software. Sie sollen dabei die vorgegebenen Klassen an Ihre Architektur und damit an Ihr System Börsencafé anpassen (ausgegeben in Versuch 2, packages 'sopra', 'remote' und 'util').

Ihnen steht der von Ihnen nachdokumentierte Code zur Verfügung. Die Signatur der Preisberechnungsmethode darf **nicht** verändert werden!

Zur Qualitätssteigerung in der Software-Entwicklung wird heute vermehrt Pair Programming verwendet. Dabei teilen sich beim Programmieren zwei Personen einen Computer, wobei eine Person programmiert und die zweite den geschriebenen Code fortlaufend überprüft. Die so entwickelte Software enthält durch das konstante informelle Review typischerweise weniger Fehler, als wenn sie von einem Einzelnen geschrieben wird. Zudem gibt es zwei Personen, welche sich im Programm auskennen.

Die Implementierung erstreckt sich über mehrere Termine. Nach dem ersten Implementierungstermin werden **von Ihnen als kritisch erachtete Teile des Codes** abgegeben und in Versuch 9 von einer anderen Gruppe einem Review unterworfen. Ziel des Versuchs ist es, dass am Ende von Versuch 10 das System soweit fertig ist, dass es getestet und dann dem Kunden zur weiteren Validierung und zur Spezifikation des produktiven Systems übergeben werden kann.

#### Vorbereitung des Versuchs (Heimarbeit)

Einarbeiten in die Materie. Grundlagen der Programmierung, wie sie in der Vorlesung «Einführung in die Programmierung» und im Programmier-Praktikum vermittelt werden, müssen bekannt sein und sollten vergegenwärtigt werden. Vertiefende Literatur ist in den Referenzen angegeben: [Somm01] gibt in Kapitel 15 einen kurzen Abriss über strukturiertes Programmieren, Datenabstraktion, defensives Programmieren und Ausnahmebehandlung. [Wirt93] führt in die systematische Programmierung ein.

## Durchführung des Versuchs

### Aufgabe 8.1: Implementierung

Realisieren Sie die im Detailentwurf festgelegten Klassen, indem Sie die skizzierten Algorithmen ausarbeiten und wo nötig den Detailentwurf verfeinern und anpassen.

Dokumentieren Sie die Implementierung den Richtlinien [Bern98] entsprechend (Bezeichner/Variablennamen, selbstdokumentierende Namen, Kommentare,...).

### Aufgabe 8.2: Arbeiten an fremden Code

Die in Versuch 2 ausgegebenen Packages „remote“, „sopra“ und „util“ sollen in Ihr System angebunden werden. Dafür ist es notwendig, die Schnittstellen-Methode „transfer()“ aus der Klasse „InputDeviceEngine“ zu implementieren und an Ihr System anzupassen. Es können auch andere Änderungen im Code notwendig sein. Führen sie auch diese aus. Die Signatur der Preisberechnungsmethode darf **nicht** verändert werden!

Passen sie das GUI der vorgegebenen Komponente an „Look and Feel“ Ihrer eigenen Benutzeroberfläche an.

### Aufgabe 8.3: Pair Programming

Versuchen Sie, während der Implementierung Pair Programming einzusetzen. Dabei teilen sich zwei Gruppenmitglieder einen Computer. Eine Person (der schwächere Programmierer) schreibt den Code, die andere unterwirft das eben Geschriebene einer fortlaufenden kritischen Überprüfung. Dadurch werden Fehler im Code und Abweichungen von stylistischen Vorgaben oder Anforderungen meist sofort erkannt. Ein zusätzlicher Vorteil für Ihre Gruppe besteht darin, dass sich mehrere Personen im gleichen Stück Code auskennen und ihre Ideen einbringen können.

Wechseln Sie Schreiber und Leser periodisch ab.

### Hinweis

Achten Sie auf genügendes Team- und Projektmanagement: Die Aufteilung der Arbeitsaufteilung in der Implementierung ist entscheidend, damit Sie in der zur Verfügung stehenden Zeit mit dem System fertig werden.

## **Abgabe**

Drei Tage nach dem Termin (Versuch 8) geben Sie einen Teil Ihres Codes ab, welchen Sie als kritisch einstufen, damit eine andere Gruppe in Versuch 9 ein Review durchführen kann. Das Ergebnis dieses Reviews kommt Ihnen im späteren Verlauf der Implementierung zu Gute. Sprechen Sie sich vor der Abgabe des zu reviewenden Codes mit dem Tutor über Eignung und Umfang der Abgabe ab.

Zusätzlich geben Sie als Referenzunterlagen den in Versuch 7 geprüften und überarbeiteten Architekturentwurf ab.

Die Abgabe erfolgt in Ihrem OLAT-Gruppenordner in einem von Ihnen anzulegenden Verzeichnis „Versuch08\_Lsg“.

## **Abgabetermin zu reviewender Code**

- Klasse Montag: Do, 27.05.04 24:00
- Klasse Dienstag: Fr, 28.05.04, 24:00
- Klasse Donnerstag: So, 30.05.04 24:00

## **Unterlagen in OLAT:**

- Alle Entwurfsdokumente aus den Versuchen 6 und 7 (Architekturentwurf und begonnener Feinentwurf)
- Code der abgegebenen Klassen (aus Versuch 2)

## **Referenzen:**

- [Somm92] I. Sommerville: *Software Engineering*. Fourth Edition, Wokingham: Addison-Wesley, 1992, Kapitel 15. (--> IfI-Bibliothek, D.2 8359 und D.2 9200)
- [Wirt93] N. Wirth: *Systematisches Programmieren: eine Einführung*. Teubner Stuttgart, 1993.  
(--> HBI)