Dr. Johannes Mayer
Hans Braxmeier
Ralph Guderlei
Abteilung Angewandte Informationsverarbeitung

23. Juni 2005
Blatt 8

Design und Architektur von Softwaresystemen (SS 2005)

Abgabetermin: 08. Juli 2005

1 Web– und Swing–GUI (20 Punkte)

Stellen Sie sich eine Klassenbibliothek für Benutzungsschnittstellen vor, die mehrere GUIs wie z.B. Web–GUIs, Swing–GUIs oder AWT–GUIs unterstützt. Unterschiedliche GUIs sehen zwar etwa unterschiedlich aus, sind aber vom Prinzip her im Wesentlichen dasselbe. Um zwischen verschiedenen GUIs portierbar zu sein, sollte sich eine Anwendung nicht auf eine GUI festlegen müssen. Dies geht, wenn man konsequent nur gegen Schnittstellen und nicht gegen irgend welche Klassen programmiert.

Teilaufgabe a) (Abgabe bis zum 1. Juli, 10 Punkte)

Definieren Sie aufbauend auf dem letzten Übungsblatt Schnittstellen für alle wichtigen GUI-Elemente (*Window*, *Panel*, *Button*, etc.). Danach implementieren Sie mit den konkreten Klassen *WWindow*, *WPanel* etc. jeweils die entsprechende Schnittstelle. Damit der Klient (also die konkrete GUI) nicht bei der Erzeugung der einzelnen Elemente einen Konstruktoraufruf machen muss, definieren Sie eine *abstrakte Fabrik*, also eine Schnittstelle *GUIFactory* mit Methoden zur Erzeugung der einzelnen GUI-Elemente (also *constructWindow()*, *constructPanel()*, etc.). Diese Schnittstelle implementieren Sie dann mit einer konkreten Klasse *WebFactory*, mit der Sie dann die Objekte von *WWindow*, *WPanel*, etc. bei den entsprechenden Methoden erzeugen. Der Klient wird *von außen* (!) mit einer *GUIFactory* konfiguriert, arbeitet nur mit GUI-Schnittstellen und erzeugt die Objekte durch Methodenaufrufe bei der *GUIFactory* (von der er nicht weiss – da Schnittstelle – was für ein Typ von GUI sich dahinter verbirgt).

Im letzten Übungsblatt wurde das Spiel Tic–Tac–Toe in Form einer webbasierten Anwendung (Servlet) mit einer Web–GUI (GUI mit ereignis–orientierter Verarbeitung) vorgestellt. Passen Sie nun Ihr (oder das vorgestellte) Tic–Tac–Toe–Spiel so an, dass es mit den Schnittstellen wie oben beschrieben funktioniert.

Teilaufgabe b) (Abgabe bis zum 8. Juli, 10 Punkte)

Implementieren Sie nun das Spiel Tic–Tac–Toe in Form einer swingbasierten Anwendung. Verwenden Sie dabei die in Teilaufgabe a) vorgestellte Architektur. Dazu müssen Sie konkrete Implementierungen der Schnittstellen *SButton*, *SWindow*, etc. machen, die eine Swing-GUI erzeugen. (Tipp: Ein SButton enthält einen JButton, d. h. es liegt eine Komposition vor.) Erstellen Sie außerdem eine passende Fabrik für die Swing-GUI-Elemente. Nun sollte Ihre Tic–Tac–Toe–Anwendung aus der letzten Teilaufgabe *ohne Modifikation* sofort als Swing-GUI verwendbar sein.

Viel Erfolg!