

# Praxis der Software-Entwicklung 2009/10

## Handyprogrammierung mit Java

Prof. Dr. Gregor Snelting – Dennis Giffhorn – Matthias Braun  
Lehrstuhl für Programmierparadigmen  
Universität Karlsruhe (TH)

Stand: 02.11.2009

## 1 Eine kurze Bemerkung vorab

Dies ist IHR Projekt. Dieses Dokument ist kein Katalog von Aufgaben, der Punkt für Punkt abgearbeitet werden muss, um nachher den Schein zu bekommen, sondern lediglich eine Reihe von Hinweisen, was wir erwarten. Wie IHR Programm nachher aussieht, müssen SIE selbst entscheiden.

## 2 Motivation

Die Programmiersprache Java existiert nicht nur in ihrer Standardausführung, der Java Standard Edition, sondern auch in einigen verschiedenen Variationen für spezielle Einsatzgebiete, wie etwa Java Card für Smart Cards oder die Java Micro Edition (JavaME) für mobile, vernetzte Geräte wie z.B. Handys. JavaME bietet dazu verschiedene Konfigurationen, um verschiedene Plattformen möglichst gut auszunutzen.

Ihre Aufgabe ist die Implementierung eines Rubik-Würfels für Handys mit Hilfe von JavaME, oder alternativ mit Android, einer anderen Java-Variation für mobile Geräte. Der Rubik-Würfel ist ein in den 80er Jahren sehr populäres Denkspiel, für den es mittlerweile eine Reihe von Lösungsalgorithmen gibt, die auf der Gruppentheorie basieren. Ihr Programm soll einen Rubik-Würfel auf einem Handy simulieren und als Hilfestellung für den Spieler einen Algorithmus Ihrer Wahl verwenden.

## 3 Aufgabenstellung

### 3.1 Bestandteile

Das von Ihnen zu erstellende System soll eine Model-View-Controller Architektur verwenden:

- ein Model, welches
  - eine Repräsentation des Würfels
  - den Lösungsalgorithmus
  - eine Funktion, die einen neuen Würfel für ein neues Spiel kreiertenthält
- als View ein GUI, das einen Würfel anzeigt und das manuelle sowie automatische Lösen des Würfels erlaubt und dem Benutzer Hilfestellung für den nächsten Zug geben kann. Hier ist Ihre Kreativität gefragt, da verschiedenste Darstellungsformen möglich sind, zB.
  - einen frei rotierbaren Würfel verwenden
  - "Spiegel" verwenden, um die Rückseite anzuzeigen
  - Anzeige der Hilfestellung
  - etc.
- einen Controller für die Tasteneingaben des Handys, der eine 'undo-redo'-Funktionalität bietet

Das Programm soll in der mit JavaME mitgelieferten Simulationsumgebung `KToolbar` des Sun Wireless Toolkit, bzw. in der Simulationsumgebung von Android laufen. Diese Umgebung kann ein Handy emulieren, in dem Ihr Programm ablaufen soll. Beachten Sie, dass damit die Eingabe auf die Handytastatur beschränkt ist.

### 3.2 Algorithmus

Zur Lösung des Rubik-Würfels existiert mittlerweile eine große Anzahl von Algorithmen. Sie können sich Ihren Lösungsalgorithmus frei auswählen. Berücksichtigen Sie bei Ihrer Auswahl, dass als Funktionalität Ihres Programms eine Schritt-für-Schritt-Hilfestellung verlangt ist.

Sie können darüber hinaus auch mehrere Algorithmen implementieren.

### 3.3 Zu verwendende Konfigurationen

JavaME gliedert sich in sogenannte Konfigurationen und Profile.

Konfigurationen definieren Minimalanforderungen bezüglich der angestrebten Zielgeräte und machen dementsprechend Einschränkungen und Modifikationen an der Java Virtual Machine und an dem Sprachumfang selbst. Zur Zeit existieren zwei Konfigurationen für JavaME, die *Connected Limited Device Configuration* (CLDC) und die *Connected Device Configuration* (CDC).

Profile definieren zusätzliche APIs für eine speziellere Kategorie von Geräten. Sie werden mit einer Konfiguration kombiniert und ermöglichen so eine Art Feinabstimmung.

Sie sollen mit der CLDC-Konfiguration und dem *Mobile Information Device Profile* (MIDP) arbeiten. Dies stellt eine weit verbreitete Kombination dar. CLDC und MIDP geben Ihnen vor, welche APIs Sie zur Verfügung haben.

### 3.4 Minimale Leistungsmerkmale

Folgende minimale Leistungsmerkmale soll Ihr Programm erfüllen:

- muss in der Simulationsumgebung KToolbar, bzw. in der Simulationsumgebung von Android laufen
- komplett über die dort simulierte Handytastatur bedienbar sein
- eine modulare Architektur nach dem Model-View-Controller-Prinzip besitzen
- ein GUI anzeigen, in dem es möglich ist, einen gegebenen Würfel
  - manuell
  - vollautomatisch
  - sowie halbautomatisch (Schritt-für-Schritt durch Hilfefunktion)zu lösen
- eine Funktion zum zufälligen Erstellen neuer Spielsituationen anbieten
- eine Zugrücknahme anbieten (undo-redo-Funktionalität)

## 4 Organisatorisches

Für jede Phase des Praktikums muss ein Phasendokument abgegeben werden. Dieses Dokument ist Grundlage für das Kolloquium am Ende jeder Phase, in dem die Gruppe die Ergebnisse der Phase vorträgt.

### Zeitlicher Ablauf

- 02.11. - 06.11. Erstes Gruppentreffen mit Betreuer
- 09.11. - 20.11. Pflichtenheft
- 23.11. - 18.12. Entwurf
- 11.01. - 05.02. Implementierung
- 08.02. - 19.02. Klausurpause
- 22.02. - 12.03. Validierung
- 15.03. - 19.03. Interne Abnahme
- 22.03. - 26.03. Abschlussveranstaltung

## 5 Bewertung

Die Benotung Ihres Systems richtet sich nach folgenden Kriterien:

- Qualität aller abgegebenen Dokumente
- Qualität der Kolloquien
- Qualität der Abschlusspräsentation
- Erfüllung der minimalen Leistungsmerkmale (s.o.)
- *sinnvolle* Erweiterungen über diese Merkmale hinaus
- Robustheit des erstellten Programms

Diese Liste hat keine Reihenfolge, die einer Gewichtung entspricht. Es gibt sicherlich weitere Punkte, die als selbstverständlich gelten und sich bei Nichterfüllen negativ auswirken (z.B. ist die Geschwindigkeit des implementierten Algorithmus sekundär, aber jedem sollte klar sein, dass das Errechnen einer Lösung nicht 10 Minuten dauern darf).