
Programmieren in Java

<http://proglang.informatik.uni-freiburg.de/teaching/java/2011/>

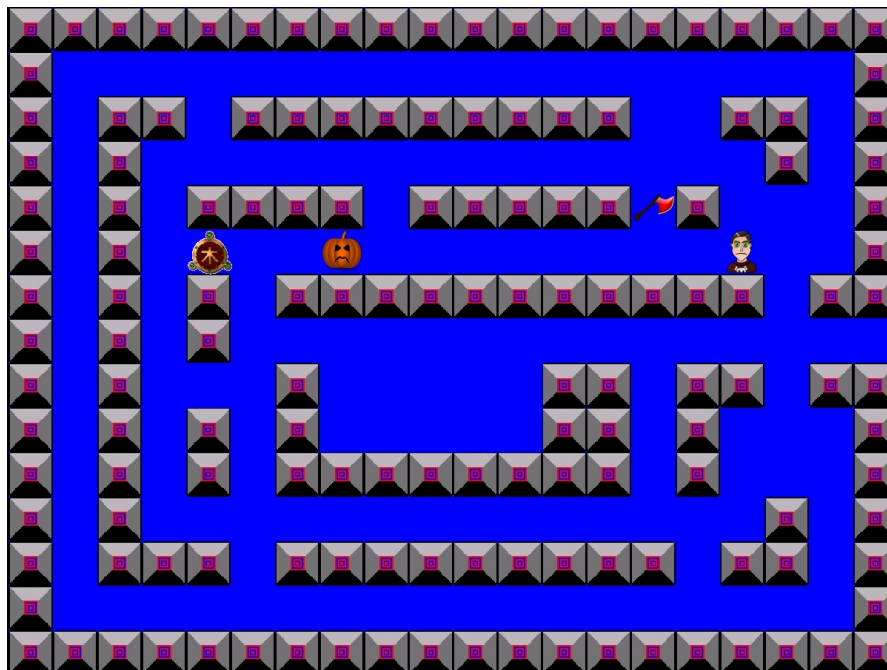
Java-Projekt: Peter and the Splendid Charm of Badenshire: Ein Interaktives 2D-Rollenspiel

2011-07-11

Hinweise

- Schreiben Sie Identifier *genau so*, wie sie auf dem Blatt stehen (inklusive Groß- und Kleinschreibung), nicht nur ungefähr.
- Committen Sie die Änderungen an Ihrem Code während des Projekts in regelmäßigen Abständen (z.B. täglich) in das bereit-
- gestellte Repository.
- Zur Abgabe committen Sie Ihren letzten Stand bis zum Abgabedatum in Ihr Repository.
- Geben Sie die PDF-Folien für Ihre Präsentation per Email bei `geffken at informatik.uni-freiburg.de` ab.

Abgabe: Freitag, 29. Juli 2011, um 9.00 Uhr.



Peter and the Splendid Charm of Badenshire

Inhaltsverzeichnis

1	Einleitung	3
2	Projekttablauf	3
2.1	Betreuung	3
2.2	Entwicklungsrechner	3
2.3	Projektphasen	3
3	Vorbereitungen	4
3.1	Subversion	4
3.2	Android-Projekt	4
3.2.1	Android-Leihgeräte	4
3.2.2	Installation	4
3.2.3	Android SDK und ADT Plugin	4
3.2.4	Hello Android Ausführen	4
3.2.5	Auf dem Android-Handy Testen	5
3.2.6	WorldLibAndroid	5
3.2.7	Tipps	5
3.3	Desktop-Projekt	5
3.3.1	TwoDeeDoo	5
4	Aufgabenbeschreibung	5
4.1	Rollenspiel-Klon	5
4.1.1	Spielprinzip	6
4.1.2	Anforderungen	7
4.2	Präsentation	7
4.3	Hinweise	7

1 Einleitung

Herzlich Willkommen beim Projekt zur Vorlesung Programmieren in Java. In diesem Projekt werden Sie in kleinen Gruppen selbständig eine überschaubare Entwicklungsaufgabe innerhalb eines gegebenen Zeitrahmens mit Java umsetzen. Sie sollen dabei das in der Vorlesung erworbene bzw. noch zu erwerbende Wissen einsetzen, um ein anspruchsvolles Problem mit den vermittelten Werkzeugen in Java umzusetzen.

Insbesondere werden Sie die Möglichkeiten objektorientierter Programmierung nutzen, um dem gegebenen Problem Herr zu werden und lernen eine nichttriviale Anwendung zu testen. Nebenbei lernen Sie auch das Arbeiten mit einer Versionsverwaltung kennen.

2 Projektablauf

2.1 Betreuung

Auch während des Projektes findet die Betreuung wie gewohnt durch die Tutoren im Pool 082-029 statt. Bitte sprechen Sie Ihren Tutor bei allen aufkommenden Fragen zum Projekt an. Weiterhin haben wir mit `java2011 at informatik.uni-freiburg.de` ein Emailalias für alle Projektteilnehmer eingerichtet, über das Sie auch mit anderen Gruppen kommunizieren können.

2.2 Entwicklungsrechner

Für die Teilnehmer des Android-Projekts und des Desktop-Projekts gibt es keine Änderungen im Vergleich zur Bearbeitung der Übungsaufgaben. Sie können sowohl auf den Pool-Rechnern arbeiten, als auch Ihren eigenen Rechner benutzen.

2.3 Projektphasen

Sie haben drei Wochen Zeit für die Umsetzung der Projektaufgabe. In der vierten Woche findet eine Vorstellung der Projekte statt. Der Ablauf des Projekts gliedert sich wie im Folgenden beschrieben.

1. Woche Das Projekt beginnt am Montag, 11. Juli 2011, also nach der Abgabe von Blatt 9, mit einer kleinen Einführung in die Aufgabe im Rahmen der Vorlesung. Im Anschluss daran bekommen die Gruppen, die das Android-Projekt gewählt haben die Leihgeräte ausgehändigt.

2. Woche Bis zu diesem Zeitpunkt sollen Sie sich ein grobes Design für Ihr Projekt überlegt haben, welches Sie in der zweiten Woche in den Übungsgruppen mit Ihren Tutoren besprechen sollen. Dabei verlangen wir keine formale Abgabe von Designdokumenten, aber Sie sollen in der Lage sein die geplante Umsetzung der Aufgabe mit ihrem Tutor zu besprechen. In erster Linie dient diese Besprechung als Hilfestellung für Ihre Gruppe. Nutzen Sie diese Gelegenheit so gut wie möglich!

3. Woche Bis zur dritten Woche sollen Sie ein voll funktionsfähiges Projekt entwickelt haben, zu dem jeder Projektteilnehmer einen signifikanten Teil beigetragen hat. Sie müssen Ihr Projekt in dieser Woche in der Übungsgruppe ihrem Tutor vorstellen. Dabei muss jeder Projektteilnehmer Fragen zum Code beantworten können. Diese Vorstellung ist Voraussetzung zum Bestehen des Kurses! Danach haben Sie noch bis Ende der Woche Zeit etwaige kleinere Fehler in Ihrem Projekt zu beseitigen. Die eigentliche Entwicklung muss jedoch schon vorher passiert sein. Die Deadline für die "Abgabe" Ihres Projektes ist Freitag, 29. Juli 2011, um 9:00 Uhr.

4. Woche Am Montag, 01. August 2011 findet im Rahmen der Vorlesung eine feierliche Vorstellung der Projekte statt. Jede Gruppe stellt einen Vertreter, der das Projekt der Gruppe kurz vorstellt und präsentiert ihr Spiel.

3 Vorbereitungen

3.1 Subversion

Wir verwenden die Versionsverwaltung *Subversion*. Ein Subversion-Server hält zeitliche Schnappschüsse eines Quellcodebaums als Versionen fest, wobei jede Version einen Änderungskommentar hat. Versionen können auch miteinander verglichen werden, und man kann sich Dateien aus einer früheren Version holen, wenn man etwas verschlechtert hat.

Jede Gruppe erhält von uns ein Subversion-Repository. Auf der Vorlesungsseite finden Sie unter *Projekt* eine Anleitung zum Auschecken Ihres Repositories.

3.2 Android-Projekt

3.2.1 Android-Leihgeräte

Sie bekommen von uns im Rahmen dieses Projektes Handys gestellt, auf denen Sie Ihr Android-Projekt testen können. Es handelt sich dabei um das Motorola MOTODEFY MB525 mit Android 2.1. Sie erhalten die Geräte beim Lehrstuhl für Programmiersprachen gegen eine Unterschrift auf einem entsprechenden Leihvertrag.

3.2.2 Installation

In dieser Installationsanleitung gehen wir davon aus, dass Sie ein funktionierendes Eclipse auf Ihrem Entwicklungsrechner installiert haben und dieses zur Entwicklung benutzen.

3.2.3 Android SDK und ADT Plugin

Sie benötigen für die Android-Entwicklung folgende Komponenten:

- Android SDK (<http://developer.android.com/sdk/index.html>)
- ADT Plugin (<http://developer.android.com/sdk/eclipse-adt.html>)

Auf den Pool-Rechnern ist das Android SDK bereits unter `/usr/local/android-sdk-linux-x86` installiert. Sie müssen allerdings noch das ADT-Plugin installieren. Folgen Sie den Installationsanleitungen unter

<http://developer.android.com/sdk/installing.html> bzw.
<http://developer.android.com/sdk/eclipse-adt.html#installing>.

Stellen Sie sicher, dass Sie eine aktuelle Version der Android-Tools haben und die SDK Platform Android 2.1 installiert ist. Diese benötigen Sie, um für das im Projekt verwendete Gerät zu entwickeln. Auf den Pool-Rechnern sollte beides bereits der Fall sein.

Falls bei der Installation des ADT-Plugins Probleme bei der Installation von Abhängigkeiten auftauchen, müssen Sie noch unter **Window - Preferences - Install/Update - Available Software Sites** die "Default Eclipse Software Site" aktivieren (z.B. <http://download.eclipse.org/releases/galileo> für Eclipse 3.5).

3.2.4 Hello Android Ausführen

Jetzt können Sie überprüfen, ob das Android SDK und das ADT-Plugin korrekt installiert sind. Folgen Sie der Anleitung unter

<http://developer.android.com/resources/tutorials/hello-world.html>,

um eine minimale Android-Applikation zum Laufen zu bringen.

3.2.5 Auf dem Android-Handy Testen

Um die von Ihnen entwickelte Anwendung auf Ihrem Handy zu testen, befolgen Sie die Schritte

1. "Declare your application as "debuggable" in your Android Manifest."
2. "Turn on "USB Debugging" on your device."
3. "Setup your system to detect your device."

auf der Seite

<http://developer.android.com/guide/developing/device.html>.

Den Abschnitt "3. Setup your system to detect your device." können Sie überspringen, wenn Sie auf den Pool-Rechnern arbeiten. Die notwendigen Anpassungen sind bereits vorgenommen und Ihr Android-Gerät sollte bereits erkannt werden.

3.2.6 WorldLibAndroid

Wir stellen Ihnen ein Framework für die 2D-Spieleentwicklung unter Android zur Verfügung, die `WorldLibAndroid`. `WorldLibAndroid` weist große Ähnlichkeiten zum Ihnen bekannten `TwoDeeDoo`-Frameworks auf. Insbesondere gibt es wie in `twodeedo` die Interfaces `IWorld` und `ICanvas`. Laden Sie sich die `WorldLibAndroid` von der Vorlesungsseite herunter.

3.2.7 Tipps

Beispielapplikation Wie Sie mit der `WorldLibAndroid` eine Android-Anwendung erstellen können, sehen Sie am besten, wenn Sie sich das `WorldLibAndroid`-Projekt von der Vorlesungsseite herunterladen und das Paket `com.WorldLibAndroid.Demo` betrachten.

Ausrichtung der Applikation auf dem Bildschirm Sie können verhindern, dass Ihre Anwendung zwischen Landscape und Portrait wechselt, indem Sie im `AndroidManifest.xml` das Attribut `android:screenOrientation="portrait"` hinzufügen.

Entwicklungsrechner Für das Android-Projekt empfehlen wir Ihnen auf einem möglichst leistungsstarken Entwicklungsrechner (Host) zu arbeiten, da die Simulationsgeschwindigkeit des Android Virtual Devices als relativ langsam eingestuft werden kann. Daher sollten Sie im Zweifelsfall einen schnellen eigenen Laptop gegenüber einem langsameren Pool-Rechner bevorzugen.

3.3 Desktop-Projekt

Für das Desktop-Projekt benötigen Sie keine Tools außer der Eclipse-Entwicklungsumgebung, wie Sie es bereits aus der Vorlesung bzw. den Übungen kennen.

3.3.1 TwoDeeDoo

Für das Projekt können Sie wieder auf das bekannte `twodeedo`-Framework zurückgreifen. Laden Sie sich dieses von der Vorlesungsseite herunter.

4 Aufgabenbeschreibung

4.1 Rollenspiel-Klon

Sie sollen in diesem Projekt als Team eine interaktive Applikation mit Java entwickeln. Es handelt sich dabei um ein einfaches 2D-Rollenspiel, bei dem der Spieler mit seinem Avatar eine Spielwelt erkundet.

4.1.1 Spielprinzip

In “Peter and the Splendid Charm of Badenshire” bewegt der Spieler einen Avatar durch (labyrinthähnliche) Räume, die in der Vogelperspektive dargestellt werden. Zur Steuerung des Avatars können Tasten oder Gesten eingesetzt werden. Die Räume bestehen aus quadratischen Kacheln fester Größe. Es gibt begehbare und nicht begehbare Kacheln. Eine Kachel kann eine Wand oder einen begehbaren Untergrund darstellen, sie kann aber auch eine Tür sein, die man mit einem zuvor erhaltenen Gegenstand (z.B. Werkzeug(e), Schlüssel) öffnen kann. Eine geöffnete Tür ist eine begehbare Kachel, eine geschlossene Tür ist entsprechend eine nicht begehbare Kachel. Durch das Öffnen von Türen kann man in Bereiche des Raumes gelangen, die auf anderem Weg möglicherweise nicht erreichbar sind.

Zum Aufschließen einer Tür kann es notwendig sein, dass neben dem vorhandenen Werkzeug noch eine Geste oder Tastenkombination (oder Abfolge von Tastatureingaben) angewandt werden muss, damit sich die Tür öffnet.

Zu jedem Zeitpunkt im Spiel wird nur ein Raum dargestellt. Der Rand eines Raumes besteht überwiegend aus Wänden. Eine begehbare Randkachel bildet den Übergang zu einem neuen Raum. Dorthin gelangt man durch das Betreten dieser Kachel. Die Spielfigur wird dann an einer bestimmten Position im neuen Raum dargestellt und es gilt diesen nun zu entdecken.

Eine begehbare Kachel kann bestimmte Eigenschaften haben. Beispielsweise kann diese eine Falltür darstellen, durch die man in einen anderen Raum gelangt, was auch das Ende des Spiels bedeuten kann. Oder der eigene Avatar verliert eine bestimmte Menge Energie pro Zeiteinheit, wenn er sich auf dieser Kachel aufhält.

Es lauern Monster in den Räumen, die es Ihrem Avatar schwer machen, sein Ziel zu erreichen. Die Monster können an gewissen Positionen stehen oder im Raum umherwandern. Wenn der Avatar auf sie trifft, kann er mit Ihnen kämpfen. Das kann durch Gesten implementiert werden (hauen, stechen, schütteln, je nach Waffe, die der Avatar gerade hat). Einige Monster sind jedoch unbesiegbar und können daher bestenfalls umgangen werden. Der Avatar kann nur mit einem Monster kämpfen, wenn es sich auf einer benachbarten (oder derselben) Kachel befindet. Ein Monster fügt Ihrem Avatar Schaden zu, wenn er sich auf einer benachbarten Kachel befindet.

Auf begehbaren Kacheln können Gegenstände liegen, die vom Spieler eingesammelt werden können. Gegenstände können Werkzeuge, Schlüssel, Waffen oder wertvolle Artefakte sein. Artefakte können beispielsweise die Widerstandsfähigkeit Ihres Avatars, seine Laufgeschwindigkeit oder den Ablauf der Kämpfe beeinflussen.

Ziel des Spiels Das Ziel des Spiels ist es, ein reich verziertes Amulett, den “Splendid Charm of Badenshire”, zu finden. Durch seine ausgeprägte Schönheit, weiß man, dass man ihn gefunden hat, sobald man ihn erblickt.

Auf dem Weg zum “Splendid Charm of Badenshire” müssen jedoch einige Kämpfe gekämpft, Gegenstände eingesammelt, Türen geöffnet und unbekannte Räume erkundet werden.

Grafische Darstellung Die Darstellung der Bewegungen des Avatars und der Monster sollen mindestens so genau sein, wie das Raster der Kacheln (diskrete Sprünge von einer Kachel zu einer benachbarten). Natürlich dürfen Sie die Bewegungen auch genauer modellieren. Monster können dem Avatar den Weg versperren: Der Avatar darf sich nicht auf Kacheln bewegen, die bereits von einem Monster belegt sind. Ob Monster über ihren Avatar laufen dürfen, bleibt Ihnen überlassen.

Die Größe der Kacheln soll “20x20” Pixel für die Android-Variante und “40x40” Pixel für die Desktop-Variante des Spiels betragen. Wählen Sie für die Desktop Variante eine Auflösung, die sich sinnvoll auf den meisten Bildschirmen anzeigen lässt, z.B. “640x480” oder “800x600”. In der grafischen Gestaltung der Kacheln, Monster und des Avatars sind Sie frei. So kann es beispielsweise geheime Gänge geben, die sich optisch nicht von Wandkacheln unterscheiden. Denkbar sind auch (fast) unsichtbare Gegenstände oder Monster.

4.1.2 Anforderungen

Sie sollen in Ihrer Anwendung mindestens die folgenden Anforderungen abdecken:

- Es soll mindestens eine Art von unbesiegbarem Monster geben, das im Raum umherwandert und mindestens eine Art von feststehendem Monster mit dem man kämpfen kann.
- Es soll mindestens zwei Arten von Waffen geben.
- Es soll mindestens vier unterschiedliche Schlüssel oder Werkzeuge geben.
- Es soll mindestens zwei wertvolle Artefakte geben.
- Es soll mindestens vier unterschiedliche Arten von Türen (zum Beispiel unterschiedliche Schlüssel und/oder Gesten akzeptierend) geben.
- Es soll mindestens zwei unterschiedliche Arten von Bodenfliesen geben.
- Es soll mindestens vier Räume geben, die zum Gewinnen des Spiels betreten werden müssen.
- Es soll ein Inventar für die Gegenstände (Waffen, Schlüssel, etc.) des Avatars geben.

Wenn Sie es schaffen sinnvolle Interfaces für die oben genannten Bausteine zu entwerfen, dürfen Sie diese benutzen, um (ausschließlich) die Implementierung dieses Interfaces mit einer anderen Gruppe auszutauschen und so maximal die Hälfte der geforderten Bausteine abzudecken. Beide Gruppen dokumentieren diese Tatsache an der betroffenen Klasse. Vorsicht: Nicht für alle Spielbausteine sind eigene Klassen sinnvoll, manchmal genügt eine Klasse und mehrere Instanzen dieser Klasse, um unterschiedliches Verhalten abzudecken.

4.2 Präsentation

Zum Abschluss des Projekts fassen Sie Ihre Ergebnisse in Form eines 5- bis 10-minütigen Vortrags mit 5 PDF-Folien zusammen und senden Sie diese per Email bis Freitag, 29. Juli 2011, um 9:00 Uhr an `geffken at informatik uni-freiburg de`. Jede Gruppe wählt einen Vertreter, der bei der Abschlussveranstaltung in wenigen Worten die im Projekt erzielten Ergebnisse zusammenfasst.

Außerdem sollen Sie Ihr Spiel demonstrieren. Bringen Sie dazu bitte (soweit vorhanden) ein Gerät mit, auf dem Sie Ihr Spiel vorführen können. Für die Desktop-Variante versuchen wir außerdem, dass Sie das Spiel auch auf dem großen Schirm präsentieren können.

4.3 Hinweise

Nachdem Sie sich ein grobes Design überlegt haben, sollten Sie zunächst versuchen eine minimale Version der Anwendung zu erstellen. Dann erweitern Sie Ihre Applikation nach und nach mit den geforderten Funktionen.

Beginnen Sie zum Beispiel mit einem einfachen Raum und einem Avatar, der sich durch diesen Raum bewegen kann. Behalten Sie dabei aber im Hinterkopf, dass es später mehrere Räume geben wird. Spätestens bei der Definition zweiten Raums wird es sinnvoll sein, sich ein Format überlegen, mit dem man Räume kompakt als Raster repräsentieren kann, z.B. in einem String. Außerdem kommt nun der Übergang von einem Raum in einen anderen ins Spiel.

Nutzen Sie die Ihnen bekannten Techniken aus der Vorlesung: Die Aufgabe bietet viele Möglichkeiten Klassen einzusetzen (`Door`, `Floor`, `Wall`, `Room`, etc.). Nutzen Sie auch abstrakte Basisklassen um Codeduplizierung zu vermeiden. Setzen Sie Interfaces sinnvoll ein. Wie bei den Übungsaufgaben sollte Ihr Code vollständig dokumentiert sein (aktivieren Sie das Checkstyle-Plugin).