

Anwendungspraktikum aus JAVA

(Rechnerpraktikum aus Programmierung - Vertiefungskurs I Informationswirtschaft)

Albert Weichselbraun

15. Oktober 2008

Inhaltsverzeichnis

1	Information zur Lehrveranstaltung	2
1.1	Einstiegsvoraussetzungen	2
1.2	Beurteilungsschema	2
1.3	Anwesenheitspflicht	2
2	Ablauf und Inhalte der Lehrveranstaltung	3
3	Organisatorisches	3
3.1	Benutzeraccount	3
3.2	Wichtige Termine	3
4	Projekt	4
4.1	Bisherige Projekte	4
4.2	Anforderungen	4
4.2.1	Problemdefinition	5
4.2.2	Projektplan (Zeitplan)	5
4.2.3	Modellierung	5
4.2.4	Implementierung	5
4.2.5	Test	5
4.2.6	Abnahme und Präsentation	6
4.2.7	Ablauf der Abnahme und Präsentation	6
5	Projektwerkzeuge	6
5.1	Modellierung	6
5.2	Programmierungsumgebung (IDE)	7
5.3	Subversion Web Zugriff	7
5.4	Projekt Wiki	7
A	Anhang	8
A.1	Literatur	8
A.2	Ressourcen	8
A.3	Folien zur Lehrveranstaltung	9
A.4	Code Beispiele	9
A.4.1	Aus der Lehrveranstaltung	9
A.4.2	Musterlösungen	11
A.5	Druckversion	11

1 Information zur Lehrveranstaltung

- **Leitung:** Dr. Albert Weichselbraun
- **Sprechstunde:** Montag 13:00-15:00
- **Termin und Ort:** Dienstag 10:00-11:30, im PC7 in der 4. Ebene des UZAI.

1.1 Einstiegsvoraussetzungen

Positiver Abschluss der Lehrveranstaltungen “Grundzüge der Modellierung” und “Grundzüge der Programmierung”. Die entsprechenden Kenntnisse werden nach einer Eingangsphase von drei Unterrichtseinheiten im Zuge eines Eingangstests überprüft.

Bitte beachten sie, dass im Zuge des Projektes von jedem Teilnehmer ein Projekt zu modellieren und auch der entsprechende Programmcode zu erstellen ist. Es ist daher unerlässlich, dass sie spätestens zum Eingangstest über die entsprechenden Kenntnisse verfügen.

1.2 Beurteilungsschema

In der Lehrveranstaltung können durch Hausübungen, Mitarbeit, Referat und Projekt in Summe 54 Punkte erreicht werden. Für das positive Absolvieren der Lehrveranstaltung sind mindestens 27 Punkte *und* ein abgeschlossenes Anwendungsprojekt notwendig.

- Mitarbeit: 10 Punkte
- Test: 12 Punkte
- Anwendungsprojekt: 32 Punkte
 - Dokumentation: 15 Punkte
 - Ausführung: 17 Punkte

1.3 Anwesenheitspflicht

Bitte beachten Sie, dass es sich um eine prüfungsimmanente Lehrveranstaltung mit Anwesenheitspflicht handelt. Sie dürfen somit *maximal* in zwei Einheiten fehlen.

2 Ablauf und Inhalte der Lehrveranstaltung

Selbständige Erweiterung eines bestehenden Software Projektes unter Einsatz von Projektmanagement und Software-Engineering Technologien in Kleinstgruppen (maximal zwei Personen).

Im Rahmen der Lehrveranstaltung werden folgende Punkte behandelt:

- Wiederholung und Überprüfung der Grundlagen.
- Erklärung der Programmierumgebung und Tools (Eclipse, Subversion, ...)
- Software Projekt (siehe Abschnitt 4.2)
 - Problemdefinition
 - Projektplan
 - Modellierung (Projekt und Interface)
 - Implementierung
 - Test
 - Abnahme und Präsentation

Das Projekt ist in Kleinstgruppen (maximal zwei Personen) zu absolvieren. Im Rahmen des Projektes soll ein Rollenspiel um weitere Funktionalität erweitert werden.

3 Organisatorisches

3.1 Benutzeraccount

Für die Dauer des Projektes steht Ihnen der institutseigene PC-Benutzerraum (Mo-Fr: 8:00-21:00, Sa: 9:00-17:00) zur Verfügung. Wenn Sie zum ersten Mal an einer Lehrveranstaltung im PC7 teilnehmen, werden die Zugangsdaten für den PC7 vor der Lehrveranstaltung an Ihren Powernet-Account gemailt.

3.2 Wichtige Termine

- **14. Oktober 2008:** Organisatorisches, Wiederholung von Grundlagen der Modellierung und Programmierung
- **21. Oktober 2008:** Java GUIs mit Swing
- **28. Oktober 2008:** Vorstellung des Projektes; Java 2D; Übungs- und Fragestunde

- **4. November 2008:** Test, Themenvergabe
- **11. November 2008:** Abgabe der Problemstellung
- **25. November 2008:** Abgabe von Zeitplan und Modellierung; Präsentation der Interface Modellierungen
- **27. Jänner 2009** *letzter* möglicher Abgabetermin für das Gesamtprojekt; Präsentation der Projekte

4 Projekt

Das Hauptaugenmerk dieser Veranstaltung liegt am Projekt. Dieses ist primär *in* den Unterrichtseinheiten zu erstellen und erfordert (i) die Projektplanung, (ii) Modellierung und (iii) Implementierung eines Plug-ins für ein Rollenspiel. Dabei kommen Projektwerkzeuge wie zum Beispiel eine Subversion-basierende Projektverwaltung, ein Projekt Wiki und Projekt-Reporting zur Anwendung.

4.1 Bisherige Projekte

Die folgenden Projekte wurde in den vorhergehenden Lehrveranstaltungen erstellt und können als Anregungen für weitere Projektideen fungieren:

- Teilnehmer und Projekte SS 2008¹
- Teilnehmer und Projekte WS 2007²
- Teilnehmer und Projekte SS 2007³
- Teilnehmer und Projekte WS 2006⁴
- Teilnehmer und Projekte SS 2006⁵
- Teilnehmer und Projekte WS 2005⁶

4.2 Anforderungen

Folgende Punkte sind im Zuge des Projektes zu bearbeiten:

¹teilnehmer2008s.html

²teilnehmer2007w.html

³teilnehmer2007s.html

⁴teilnehmer2006w.html

⁵teilnehmer2006s.html

⁶teilnehmer2005w.html

4.2.1 Problemdefinition

Die Anforderungen an das zu erstellende Programm sind verbal festzulegen und im Pflichtenheft festzuhalten. Das Pflichtenheft sollte die Fragen beantworten *worum* es bei dem Projekt geht und *was* genau realisiert wird. Nachträgliche Änderungen am Pflichtenheft sind nicht zulässig.

Die Abgabe des Pflichtenhefts erfolgt im Verzeichnis `public_html/ap/projekt`.

4.2.2 Projektplan (Zeitplan)

Der Zeitplan ist als GANTT-Diagramm auszuführen. Notieren sie auch während des Projektes sämtliche Arbeitszeiten, sodass am Ende ein Soll/Ist Vergleich möglich ist.

4.2.3 Modellierung

Im Zuge der Modellierung sind folgende Diagramme zu entwerfen:

- Use-Case Diagramm
- Klassendiagramm
- Aktivitätsdiagramm
- Sequenzdiagramm

4.2.4 Implementierung

Die Applikation ist in Java zu implementieren und muss auf den Schulungsraumserver laufen!

Neben dem lauffähigen Programm muss zusätzlich eine Javadoc-Dokumentation vorliegen.

4.2.5 Test

Getestet werden Funktionalität und Plattformunabhängigkeit. Achten sie auch auf Abhängigkeiten vom Java Runtime Environment, Betriebssystem und Verzeichnisstrukturen.

Es ist ein Testprotokoll mit allen durchgeführten Tests (Beschreibung im Checklistenformat) abzugeben.

4.2.6 Abnahme und Präsentation

Das komplette Projekt ist *spätestens* in der letzten Vorlesungseinheit abzugeben und zu präsentieren.

Sämtliche andere Schritte sind zuvor abzuschliessen. Die Java-Applikation und Dokumentation muss sich in ihrem Homeverzeichnis auf dem Schulungsraumserver im Verzeichnis `public_html/ap` befinden und via http über die URL `http://xmdimrill.ai.wu-wien` abrufbar sein.

Innerhalb des Verzeichnisses `public_html/ap` muss folgende Verzeichnisstruktur eingehalten werden:

- `projekt`: dieses Verzeichnis enthält die *gesamte* Projektdokumentation (Pflichtenheft, Projektplan, Modellierung (Diagramme) sowie die Testergebnisse) im *html* oder *pdf*-Format.
- `java`: in diesem Verzeichnis befindet sich die java-Applikation mit sämtlichen dazugehörigen `.java`-files.
- `javadoc`: enthält die mittels *javadoc* generierte Programmdokumentation.

4.2.7 Ablauf der Abnahme und Präsentation

- in der Präsentation soll das Programm (Funktion) in *maximal* 5 min vorgestellt werden.
- anschließend werden Programm (es wird 1-2 Fragen zum Quellcode geben) und Diagramme besprochen werden.

5 Projektwerkzeuge

Die folgenden Tools werden im Zuge des Projektes eingesetzt.

5.1 Modellierung

Für die Modellierung wird Dia⁷ - eine freie Software zum Erstellen von UML Diagrammen, empfohlen. Für den Zeitplan können sie Planner⁸ verwenden.

⁷<http://www.gnome.org/projects/dia/>

⁸<http://live.gnome.org/Planner>

5.2 Programmierumgebung (IDE)

Es wird Eclipse⁹ mit dem Subclipse¹⁰ plug-in für den Zugriff auf das Subversion Repository verwendet. Hinweise zur Installation von Subclipse finden sie hier¹¹.

Das Subversion Repository für das Sommersemester 2008 ist wie folgt zu konfigurieren:

1. **Benutzername:** j+Matrikelnummer
2. **Passwort:** Subversion Passwort
3. **URL:** `http://xmdimrill.ai.wu-wien.ac.at/svn/projects/java`
4. **Projekt:** Moelltal

5.3 Subversion Web Zugriff

Auf die Subversion Repositories kann auch via Web Interface zugegriffen werden:

- Wintersemester 2008¹².
- Sommersemester 2008¹³, Wintersemester 2007¹⁴, Sommersemester 2007¹⁵, Wintersemester 2006¹⁶.

5.4 Projekt Wiki

Das Projekt Wiki¹⁷ enthält Informationen zum Framework; Das Ändern, Erweitern und Hinzufügen von Information ist erwünscht und wird ihnen in Form von Zusatzpunkten refundiert; dafür bitte mit ihrem Namen unter Benutzerkonto anlegen¹⁸ ein neues Konto anlegen und mit diesem Änderungen vornehmen.

⁹<http://www.eclipse.org>

¹⁰<http://svn.collab.net/subclipse/help/index.jsp>

¹¹<http://subclipse.tigris.org/install.html>

¹²<http://xmdimrill.ai.wu-wien.ac.at/websvn/listing.php?repname=Projects&path=\%2Fjav>

¹³<http://xmdimrill.ai.wu-wien.ac.at/websvn/listing.php?repname=Java+AP+2008s&path=\%2Fjav>

¹⁴<http://xmdimrill.ai.wu-wien.ac.at/websvn/listing.php?repname=Java+AP+2007w&path=\%2Fjav>

¹⁵<http://xmdimrill.ai.wu-wien.ac.at/websvn/listing.php?repname=Java+AP+2007s&path=\%2Fjav>

¹⁶<http://xmdimrill.ai.wu-wien.ac.at/websvn/listing.php?repname=Java+AP+2006w&path=\%2Fjav>

¹⁷<https://java.semanticlab.net>

¹⁸<https://java.semanticlab.net/index.php?title=Spezial:Userlogin&type=signup&returnt>

A Anhang

A.1 Literatur

- Hanspeter Mössenböck: “Sprechen sie Java?”, ISBN: 3-89864-117-1
- Joshua Bloch: “Effektiv Java programmieren”, ISBN: 3-8273-1933-1
- Hendrik Schreiber: “Performant Java programmieren”, ISBN: 3-8273-2003-8
- Douglas Bell, Mike Parr: “Java für Studenten”, 3. überarbeitete Auflage, ISBN: 3-8273-7045-0
- Java API¹⁹
- Java Documentation²⁰
- A very good Java Swing Tutorial²¹
- Java Manuals in der Virtual University²²
- Version Control with Subversion²³ - ein freies Buch über Subversion.

A.2 Ressourcen

- UML-Diagramm der Mölltal API²⁴
- Java Software Development Kit (JDK)²⁵
- Eclipse²⁶ - eine sehr gute und leistungsfähige Java IDE
- ArgoUML²⁷ - eine Open Source UML Software (Java Web Start)
- Dia²⁸ - eine freie Software zum Erstellen von UML Diagrammen
- WinSCP²⁹ - ein Freeware SFTP und SCP Client für Windows

¹⁹<http://java.sun.com/javase/6/docs/api/>

²⁰<http://java.sun.com/j2se/learning/index.html>

²¹<http://www.apl.jhu.edu/~hall/java/Swing-Tutorial/>

²²<http://vu.wu-wien.ac.at/dyn/virlib/components/doquery?query=DC\%2ESubject\%20\%3A>

²³<http://svnbook.red-bean.com/>

²⁴<images/MoelltalAPI.png>

²⁵<http://java.sun.com/javase/downloads/>

²⁶<http://www.eclipse.org>

²⁷<http://argouml-downloads.tigris.org/jws/argouml-latest-stable.jnlp>

²⁸<http://www.gnome.org/projects/dia/>

²⁹<http://winscp.net>

- Planner³⁰ - Ein Projekt Management Tool zum erstellen von GANTT Diagrammen.
- Ultima 4 Information Page³¹

A.3 Folien zur Lehrveranstaltung

- 1. Einheit (Einführung, Java Collections, Übungsbeispiel³²)
- 2. Einheit (Swing und Java 2D³³)
- 3. Einheit (JList & JavaIO³⁴)
- 4. Einheit (Subversion³⁵)
- 5. Einheit (XMLRPC und Timer³⁶)
- 6. Einheit (RMI³⁷)
- 7. Einheit (Häufige Probleme beim Erstellen von jar-Files³⁸)
- 8. Einheit (Java Deployment³⁹)
- Javadoc - Kurzbeschreibung und Style Guide⁴⁰ | Essentielle Javadoc Infos⁴¹ von Sun

A.4 Code Beispiele

A.4.1 Aus der Lehrveranstaltung

- 1. Einheit
 - Collections Beispiel 1⁴²

³⁰<http://live.gnome.org/Planner>

³¹<http://www.moongates.com/u4/Tech.asp>

³²pdf/unit1.pdf

³³pdf/unit2.pdf

³⁴pdf/unit3.pdf

³⁵pdf/unit4.pdf

³⁶pdf/unit5.pdf

³⁷pdf/unit6.pdf

³⁸pdf/unit7.pdf

³⁹pdf/unit8.pdf

⁴⁰pdf/unit9_doku_guide.pdf

⁴¹<http://java.sun.com/j2se/javadoc/writingdoccomments>

⁴²java/collections/SimpleCollection.java

- Collections Beispiel 2: Adresse Klasse⁴³, Telefonbuch Klasse⁴⁴
- 2. Einheit
 - Swing Einführungsbeispiel 1⁴⁵
 - Swing Einführungsbeispiel 2⁴⁶
 - Java2D; Zeichenfläche Klasse⁴⁷, Java2DDemo Klasse⁴⁸ | starten⁴⁹.
- 3. Einheit
 - JList Beispiel 1⁵⁰
 - JList Beispiel 2⁵¹ (dieses Beispiel zeigt zusätzlich die Verwendung von beschrifteten Gruppierungen mittels JPanel)
 - JList Beispiel 3 (mit eigenem ListModel): Telefonbucheintrag Klasse⁵², Telefonbuch Klasse⁵³, TelefonbuchGUI Klasse⁵⁴ | starten⁵⁵ (Hinweis: das Speichern/Laden von Daten funktioniert nicht, da das JAR-File nicht signiert ist).
- 5. Einheit
 - XMLRPC und Timer; ChatServer Klasse⁵⁶, ChatGui Klasse⁵⁷
- 6. Einheit
 - RMI; Serverinterface⁵⁸, Server Klasse⁵⁹, Sender Klasse⁶⁰, Security Policy⁶¹

⁴³ java/collections/Adresse.java

⁴⁴ java/unit2/Telefonbuch.java

⁴⁵ java/swing/SimpleGUI2.java

⁴⁶ java/swing/MyGUI.java

⁴⁷ java/unit4/Zeichenflaeche.java

⁴⁸ java/unit4/Java2DDemo.java

⁴⁹ <http://www.ai.wu-wien.ac.at/~aweichse/anwendungsprojekt/jar/Java2DDemo.jnlp>

⁵⁰ java/unit3/JListExample.java

⁵¹ java/unit3/JListExample2.java

⁵² java/unit3/Telefonbucheintrag.java

⁵³ java/unit3/Telefonbuch.java

⁵⁴ java/unit3/TelefonbuchGUI

⁵⁵ <http://www.ai.wu-wien.ac.at/~aweichse/anwendungsprojekt/jar/TelefonbuchDemo.jnlp>

⁵⁶ java/unit5/ChatServer.java

⁵⁷ java/unit5/ChatGui.java

⁵⁸ java/unit6/ServiceInterface.java

⁵⁹ java/unit6/Server.java

⁶⁰ java/unit6/Sender.java

⁶¹ java/unit6/policy.txt

A.4.2 Musterlösungen

- Beispiel Einausgabenrechnung (Klasse Buchung⁶², Einnahme⁶³, Ausgabe⁶⁴, Ein-AusgabenRechnung⁶⁵)
- Beispiel Wechselkursrechner (Klasse WechselKursRechner⁶⁶ | starten⁶⁷)
- Beispiel Würfel (Klasse JWuerfel⁶⁸, WuerfelDemo⁶⁹ | starten⁷⁰)

A.5 Druckversion

Die Druckversion dieser Seite finden sie auf ([pdf/anwendungsprojekt.pdf](#)).

⁶² java/beispiele/1/Buchung.java

⁶³ java/beispiele/1/Einnahme.java

⁶⁴ java/beispiele/1/Ausgabe.java

⁶⁵ java/beispiele/1/EinAusgabenRechnung.java

⁶⁶ java/beispiele/2/WechselKursRechner.java

⁶⁷ java/beispiele/2/Wechselkursrechner.jnlp

⁶⁸ java/beispiele/3/JWuerfel.java

⁶⁹ java/beispiele/3/WuerfelDemo.java

⁷⁰ java/beispiele/3/Wuerfel.jnlp