

Java Programming

(Rechnerpraktikum aus Programmierung - Vertiefungskurs I Informationswirtschaft)

Albert Weichselbraun

April 5, 2011

Contents

1	General Information	2
1.1	Requirements	2
1.2	Assessment	2
1.3	User Account	2
2	Unit Aims and Schedule	3
3	Software Project	3
3.1	Past Projects	4
3.2	Required Project Steps and Documentation	4
3.2.1	Requirement Specification	4
3.2.2	Modeling	5
3.2.3	Project Plan	5
3.2.4	Implementation and Documentation	5
3.2.5	Deployment and Presentation	5
4	Recommended Tools	5
4.1	Modeling	5
4.2	Programming Environment	6
4.3	Subversion Web Access	6
4.4	Project Wiki	7
A	Appendix	7
A.1	Literature	7
A.2	Resources	7
A.3	Slides	8
A.4	Code Examples	8
A.4.1	Lecture	8
A.4.2	Sample Solutions	9
A.5	Print Version	9

1 General Information

- **Lecturer:** PD Dr. Albert Weichselbraun
- **Counseling hours:** Monday 8:00-9:30am
- **Time:** Tuesday 10:00-12:00pm, (PC7, UZA-II)
- **Course language:** English

1.1 Requirements

Basic programming and modeling skills (completed lectures “Grundzüge der Modellierung” and “Grundzüge der Programmierung”).

1.2 Assessment

The assessment consists of lab work, a test and a programming project.

- participation and work in labs: 20%
- test: 25%
- software project: 55%
 - documentation: 20%
 - implementation: 35%

You need at least 50% and a completed programming project to pass this lecture.

1.3 User Account

You may use the PC7 lab (Monday-Friday: 8am-9pm, Saturday 9am-5pm) during the semester in which you have registered for this course. If this is the first time that you have subscribed to a class in PC7, we will e-mail you your account data to your university account.

2 Unit Aims and Schedule

Students extend an existing software project (a role-playing game) using project management and software engineering technology. The software project is either performed alone or together with another student.

This lecture covers the following topics:

- repetition and verification of the required programming and modeling skills.
- introduction to the program environment (Eclipse, Subversion, ...)
- software project (Section 3)
 - requirement specification
 - project plan
 - modeling
 - implementation
 - deployment and presentation

Schedule

- **8 March 2011:** introduction, Java Collections
- **15 March 2011:** Java GUIs with Swing
- **22 March 2011:** introduction of the project framework; Java 2D; assessment of the programming topics
- **29 March 2011:** test, discussion of the requirement specification
- **5 April 2011:** completed project plan and modelling
- **31 May 2011:** presentation of the software project

3 Software Project

The goal of the software project is to implement plug-ins for an existing role-playing game, which provide additional quests or extend the game's existing functionality.

3.1 Past Projects

- winter term 2010¹
- summer term 2010²
- winter term 2009³
- summer term 2009⁴
- summer term 2008⁵
- winter term 2007⁶
- summer term 2007⁷
- winter term 2006⁸
- summer term 2006⁹
- winter term 2005¹⁰

3.2 Required Project Steps and Documentation

3.2.1 Requirement Specification

The requirement specification outlines the plug-ins requirements and specifies

- the project's topic (what is the purpose of your plug-in) and
- and functions (what functions are you going to provide)

Please upload the requirement specification into the directory `public.html/ap/projekt`.

¹teilnehmer2010w.html

²teilnehmer2010s.html

³teilnehmer2009w.html

⁴teilnehmer2009s.html

⁵teilnehmer2008s.html

⁶teilnehmer2007w.html

⁷teilnehmer2007s.html

⁸teilnehmer2006w.html

⁹teilnehmer2006s.html

¹⁰teilnehmer2005w.html

3.2.2 Modeling

The project modeling comprises the following diagrams

- use-case diagram,
- class diagram,
- activity diagram and
- sequence diagram

3.2.3 Project Plan

A Gantt chart which illustrates the project schedule on a class level.

3.2.4 Implementation and Documentation

Your plug-in needs to be implemented in Java and runnable in the lab. The program documentation should be auto-generated using JavaDoc.

3.2.5 Deployment and Presentation

All projects are presented in the course's last unit. All implementation and documentation steps need to be finished by then. Please upload your application and documentation to the following directories:

- `public_html/ap/project`: complete project documentation (requirement specification, project plan, modeling) as *HTML* or *PDF* files.
- `public_html/ap/java`: source code of your Java application.
- `public_html/ap/javadoc`: auto-generated JavaDoc program documentation.

The students are expected to provide a short (¡ 5 minutes) presentation of their plug-in. Afterwards there will be a discussion of the program's source code and modeling.

4 Recommended Tools

4.1 Modeling

- Dia¹¹ for creating UML diagrams and

¹¹<http://www.gnome.org/projects/dia/>

- Planner¹² for the project plan.

4.2 Programming Environment

Eclipse¹³ using the Subversive¹⁴ plug-in for version control and accessing the Subversion repository. A documentation outlining the Subversive Installation Process can be found here¹⁵.

Subversion Repository Configuration:

1. User: j+matriculation number
2. Password: your PC7 password
3. URL: <https://xmdimrill.ai.wu.ac.at/svn/projects/java>
4. Project: Moelltal

Attention: Please verify that the Eclipse encoding settings are set to UTF-8 (Project -> Properties -> Resource -> Text file encoding -> UTF-8).

4.3 Subversion Web Access

- summer term 2011¹⁶.
- winter term 2010¹⁷, summer term 2010¹⁸, winter term 2009¹⁹, summer term 2009²⁰, winter term 2008²¹, summer term 2008²², winter term 2007²³, summer term 2007²⁴, winter term 2006²⁵.

¹²<http://live.gnome.org/Planner>

¹³<http://www.eclipse.org>

¹⁴<http://www.eclipse.org/subversive>

¹⁵<http://www.eclipse.org/subversive/documentation/gettingStarted/aboutSubversive/ins>

¹⁶<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Projects&path=%2Fjava%2F>

¹⁷<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Projects&path=%2Fbranches>

¹⁸<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Projects&path=%2Fbranches>

¹⁹<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Projects&path=%2Fbranches>

²⁰<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Projects&path=%2Fbranches>

²¹<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Projects&path=%2Fbranches>

²²<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Java+AP+2008s&path=%2Fjav>

²³<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Java+AP+2007w&path=%2Fjav>

²⁴<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Java+AP+2007s&path=%2Fjav>

²⁵<http://xmdimrill.ai.wu.ac.at/websvn/listing.php?reaname=Java+AP+2006w&path=%2F&sc>

4.4 Project Wiki

The Project Wiki²⁶ contains information regarding the role-playing framework; extending and improving this information is encouraged and yields additional lab points; you need to register²⁷ to change the Wiki's contents.

A Appendix

A.1 Literature

- Bruce Eckel: "Thinking in Java", Prentice Hall; 4 edition (February 20, 2006)
- Java API²⁸
- Java Documentation²⁹
- A very good Java Swing Tutorial³⁰
- Version Control with Subversion³¹ - a free subversion book.

A.2 Resources

- UML-diagram of the Mölltal API³²
- Java Software Development Kit (JDK)³³
- Eclipse³⁴
- Dia³⁵ - a program for drawing UML diagrams
- Tortoise SVN³⁶ - a subversion client which works with Windows 7
- WinSCP³⁷ - an SFTP and SCP Client for Windows

²⁶<https://www.semanticlab.net/index.php/Mlltal>

²⁷<https://www.semanticlab.net/index.php?title=Special:Userlogin&type=signup>

²⁸<http://download.oracle.com/javase/6/docs/api/>

²⁹<http://java.sun.com/j2se/learning/index.html>

³⁰<http://www.apl.jhu.edu/~hall/java/Swing-Tutorial/>

³¹<http://svnbook.red-bean.com/>

³²[moelltal-doku/moelltal-class.class.png](http://moelltal-doku.moelltal-class.class.png)

³³<http://java.sun.com/javase/downloads/>

³⁴<http://www.eclipse.org>

³⁵<http://www.gnome.org/projects/dia/>

³⁶<http://tortoisesvn.net>

³⁷<http://winscp.net>

- Planner³⁸ - a project management tool for creating Gantt diagrams.
- Ultima 4 Information Page³⁹

A.3 Slides

- Unit 1 (Introduction, Java Collections, Exercises⁴⁰)
- Unit 2 (Swing and Java 2D⁴¹)
- Unit 3 (Subversion⁴²)
- Unit 4 (Javadoc Introduction and Style Guide⁴³ — Essential Javadoc⁴⁴ by Oracle)
- Unit 5 (Java Deployment and common problems when creating JAR-Files⁴⁵)

A.4 Code Examples

A.4.1 Lecture

- Unit 1
 - Collections Example 1⁴⁶
 - Collections Example 2: Class Address⁴⁷, Class PhoneBook⁴⁸
- Unit 2
 - Swing Example 1⁴⁹
 - Swing Example 2⁵⁰
 - Java2D; MyComponent class⁵¹, Java2DDemo class⁵² — start demo⁵³.

³⁸<http://live.gnome.org/Planner>

³⁹<http://www.moongates.com/u4/Tech.asp>

⁴⁰pdf/unit1.pdf

⁴¹pdf/unit2.pdf

⁴²pdf/unit3_subversion.pdf

⁴³pdf/unit4_javadoc.pdf

⁴⁴<http://www.oracle.com/technetwork/java/javase/documentation/index-137868.html>

⁴⁵pdf/unit5_deployment.pdf

⁴⁶java/collections/SimpleCollection.java

⁴⁷java/Address.java

⁴⁸java/PhoneBook.java

⁴⁹java/swing/SimpleGUI2.java

⁵⁰java/swing/MyGUI.java

⁵¹java/MyComponent.java

⁵²java/Java2DDemo.java

⁵³<http://www.ai.wu.ac.at/weichselbraun/anwendungsprojekt/jar/Java2DDemo.jnlp>

A.4.2 Sample Solutions

- Accounting Example (class Transaction⁵⁴, Revenue⁵⁵, Expenditure⁵⁶, Accounting⁵⁷)
- Exchange rate calculator (class ExchangeRateCalculator⁵⁸ — start⁵⁹)
- Dice Example (class JDiceComponent⁶⁰, DicerDemo⁶¹ — start⁶²)

A.5 Print Version

The print version of this page is available at ([pdf/anwendungsprojekt.pdf](#)).

⁵⁴ java/beispiele/1/Transaction.java

⁵⁵ java/beispiele/1/Revenue.java

⁵⁶ java/beispiele/1/Expenditure.java

⁵⁷ java/beispiele/1/Accounting.java

⁵⁸ java/beispiele/2/ExchangeRateCalculator.java

⁵⁹ java/beispiele/2/Wechselkursrechner.jnlp

⁶⁰ java/beispiele/3/JDiceComponent.java

⁶¹ java/beispiele/3/DiceDemo.java

⁶² java/beispiele/3/Wuerfel.jnlp