MONASH University
Information Technology

FIT5151
Object-oriented business application development

Unit Guide

Semester 1, 2014

Last updated: 20 Feb 2014

Table of Contents

FIT5151 Object-oriented business application development - Semester 1, 2014

FIT5151 will aim at capitalising on what students have learned in FIT9017 Foundations of programming (or equivalent). The unit covers more in-depth material to enable students to build business applications that follow good Software Engineering principles of maintainability, reusability and expandability. The emphasis will be on helping students acquire solid object-oriented programming knowledge and skills for building business applications. Popular object-oriented design patterns will be introduced whenever appropriate to illustrate effective design process in building larger systems.

Mode of Delivery

Gippsland (Off-campus)

Workload Requirements

Minimum total expected workload equals 12 hours per week comprising:

(a.) Contact hours for on-campus students:

- 2 hours of lectures
- One 2-hour laboratory

(b.) Study schedule for off-campus students:

- Off-campus students generally do not attend lecture and tutorial sessions, however should plan to spend equivalent time working through the relevant resources and participating in discussion groups each week.

(c.) Additional requirements (all students):

- A minimum of 8 hours independent study per week for completing lab and project work, private study and revision.

Unit Relationships

Prerequisites

FIT5131 or FIT9017 or equivalent

Chief Examiner

Associate Professor Judithe Sheard

Campus Lecturer

Gippsland

Dengsheng Zhang

Your feedback to Us

Monash is committed to excellence in education and regularly seeks feedback from students, employers and staff. One of the key formal ways students have to provide feedback is through the Student Evaluation of Teaching and Units (SETU) survey. The University's student evaluation policy requires that every unit is evaluated each year. Students are strongly encouraged to complete the surveys. The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied and areas for improvement.

For more information on Monash's educational strategy, see:

www.monash.edu.au/about/monash-directions/ and on student evaluations, see:
www.policy.monash.edu/policy-bank/academic/education/quality/student-evaluation-policy.html

Previous Student Evaluations of this Unit

Based on previous student evaluations, we decided:

1. To provide more examples in the lecture to help with the understanding of concepts.
2. To improve the sequencing of certain topics.
3. To focus on programming skills rather than the implementation of prototype features in all practical assignments.
4. To align the assessment requirements with students' undertanding of concepts rather than prototype features. The alignment will be done based on weekly learning objectives.

If you wish to view how previous students rated this unit, please go to
https://emuapps.monash.edu.au/unitevaluations/index.jsp

Academic Overview

Learning Outcomes

On successful completion of this unit, you should be able to:

- design and implement well-run, well-tested and well-documented object-oriented software by following the solid software engineering principles of maintainability, reusability and expandability;
- explain and apply a variety of popular object-oriented design patterns within the greater context of an application as a whole;
- identify and analyse business-related problems and develop object-oriented solutions to these problems;
- evaluate the successfulness of an object-oriented solution through the use of thorough software testing strategies;
- demonstrate the communication skills needed to explain the overall design and technical aspects of a completed object-oriented solution in both a face-to-face manner and through well documented source code.

Unit Schedule

Week	Activities	Assessment
0		No formal assessment or activities are undertaken in week 0
1	Programming Concepts and Java - variables, operators, expressions, control structures, recursion, class structure, collections, primitive types, object types, exceptions, I/O, file I/O	This week's exercise is a revision of prerequisite knowledge.
2	Object oriented Concepts - classes, objects, methods, attributes, message passing, constructors, inheritance, polymorphism, encapsulation, visibility, abstraction, packages, interacting classes, association, aggregation, composition	
3	Inheritance - subclasses, subtyping, substitution, overriding, types of inheritance, access modifiers	
4	Inheritance - abstract classes, multiple inheritance, interfaces, inner classes, enumerations	
5	Testing, testing tools (JUnit)	
6	GUI - event handling, components, layout, AWT and Swing libraries	
7	Persistence: Java database connectivity, more file I/O	Assignment 1 due Week 7 (Friday 11:55pm Australian EST)
8	Program Design - design techniques (responsibility driven design), Parnas' principles, design representation (UML)	
9	Program design - coupling and cohesion, Law of Demeter, Design by Contract, Assertions, Refactoring	
10	Design Patterns - decorator, singleton, factory, observer, etc., frameworks	
11	Software development methodologies, agile methods	
12	Bringing it all together - Revision	Assignment 2 due Week 12 (Friday 11:55pm Australian EST)
	SWOT VAC	No formal assessment is undertaken in SWOT VAC
	Examination period	LINK to Assessment Policy: http://policy.monash.edu.au/policy-bank/academic/education/assessment/assessment-in-coursework-policy.html

*Unit Schedule details will be maintained and communicated to you via your learning system.

Teaching Approach

Online recorded lectures, work sheets and other resources

Each week students will be expected to view the lecture recording, do the recommended reading and

then complete the practical exercises. Students will be supported by online discussion forums and the lecturer who will be available for guidance via email and forums.

Assessment Summary

Examination (3 hours): 60%; In-semester assessment: 40%

Assessment Task	Value	Due Date
Assignment 1	20%	Week 7 (Friday 11:55pm Australian EST)
Assignment 2	20%	Week 12 (Friday 11.55pm Australian EST)
Examination 1	60%	To be advised

Assessment Requirements

Assessment Policy

Faculty Policy - Unit Assessment Hurdles
(http://intranet.monash.edu.au/infotech/resources/staff/edgov/policies/assessment-examinations/assessment-hurd

Academic Integrity - Please see resources and tutorials at
http://www.monash.edu/library/skills/resources/tutorials/academic-integrity/

Assessment Tasks

Participation

- **Assessment task 1**

  Title:
  Assignment 1
  Description:
  The design and implementation of an application employing the principles introduced in the early to mid part of the semester.

  Details will be made available in the assignment specification.
  Weighting:
  20%
  Criteria for assessment:
  This is an individual assignment and must be entirely your own work.

  Assessment for the assignment will be via interview.

  Submissions will be judged on their compliance with the stated functional requirements, code and design principles presented in the unit. You will be asked to demonstrate your system during an interview and can also expect to be asked to explain your system, your code, your design, discuss design decisions and alternatives and modify your code / system as required. Marks will not be awarded for any section of code or functionality that a student cannot explain or modify satisfactorily (the marker may delete excessive comments in code before a student is asked to explain that code).

  Further details on the tasks and requirements will be made available in the assignments' specifications. Arrangements regarding interviews will also be outlined in the assignments' specifications.
  Due date:
  Week 7 (Friday 11:55pm Australian EST)

- **Assessment task 2**

  Title:
  Assignment 2
  Description:
  The design and implementation of an application employing the principles introduced in the mid to later part of the semester.

Details will be made available in the assignment specification.

Weighting:

20%

Criteria for assessment:

This is an individual assignment and must be entirely your own work.

Submissions will be judged on their compliance with the stated functional requirements, code and design principles presented in the unit.

Assessment of this assignment will be by interview.

Due date:

Week 12 (Friday 11.55pm Australian EST)

Examinations

- **Examination 1**

Weighting:

60%

Length:

3 hours

Type (open/closed book):

Closed book

Electronic devices allowed in the exam:

None

Learning resources

Reading list

The following may provide useful extra reading for this unit. Copies of these are available in the Library (on reserve, one day loan or in the normal circulation).

Java Foundations, Lewis, De Pasquale & Chase, (Pearson Education), 2011

Big Java (4th Edition), Cay Horstman (John Wiley & Sons), 2010

Java Programming - from Problem Analysis to Program Design (5th Edition), D. S Malik (Thomson), 2012

Thinking in Java (4th Edition), Eckell (Prentice Hall), 2006

Absolute Java (5th Edition), Savitch (Addison Wesley), 2013

Monash Library Unit Reading List (if applicable to the unit)
http://readinglists.lib.monash.edu/index.html

Faculty of Information Technology Style Guide

Feedback to you

Examination/other end-of-semester assessment feedback may take the form of feedback classes, provision of sample answers or other group feedback after official results have been published. Please check with your lecturer on the feedback provided and take advantage of this prior to requesting individual consultations with staff. If your unit has an examination, you may request to view your examination script booklet, see http://intranet.monash.edu.au/infotech/resources/students/procedures/request-to-view-exam-scripts.html

Types of feedback you can expect to receive in this unit are:

- Graded assignments with comments
- Interviews
- Other: Staff responses to queries posted on discussion forums

Extensions and penalties

Submission must be made by the due date otherwise penalties will be enforced.

You must negotiate any extensions formally with your campus unit leader via the in-semester special consideration process: http://www.monash.edu.au/exams/special-consideration.html

Returning assignments

Students can expect assignments to be returned within two weeks of the submission date or after receipt, whichever is later.

Assignment submission

It is a University requirement (http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-managing-pla for students to submit an assignment coversheet for each assessment item. Faculty Assignment coversheets can be found at http://www.infotech.monash.edu.au/resources/student/forms/. Please check with your Lecturer on the submission method for your assignment coversheet (e.g. attach a file to the online assignment submission, hand-in a hard copy, or use an online quiz). Please note that it is your responsibility to retain copies of your assessments.

Online submission

If Electronic Submission has been approved for your unit, please submit your work via the learning system for this unit, which you can access via links in the my.monash portal.

Required Resources

Please check with your lecturer before purchasing any Required Resources. Limited copies of prescribed texts are available for you to borrow in the library, and prescribed software is available in student labs.

Prescribed Software

You must have the Java SE 6 (or later) Software Development Kit (called the JDK) installed on your computer. This software can be downloaded for free from the internet by going to http://www.oracle.com/technetwork/java/javase/downloads/index.html and clicking on the 'download' button in the JDK column.

Prescribed text(s)

Limited copies of prescribed texts are available for you to borrow in the library.

Barnes and Kolling. (2012). *Objects First with Java: A Practical Introduction Using BlueJ*. (5th Edition) Prentice Hall (ISBN: 9780132492669).

Recommended Resources

Useful Software

Whilst the JDK provides the compiler and runtime interpreter for the Java language, you will most likely want to make use of an Integrated Development Environment (IDE). You may use any IDE that you are comfortable with. Examples of IDE are BlueJ, Eclipse, JCreator and NetBeans.

Recommended text(s)

Stuart Reges and Marty Stepp. (2011). *Building Java Programs: A Back to Basics Approach*. (2nd Edition) Addison Wesley (ISBN: 0-136-09181-4).

Other Information

Policies

Monash has educational policies, procedures and guidelines, which are designed to ensure that staff and students are aware of the University's academic standards, and to provide advice on how they might uphold them. You can find Monash's Education Policies at: www.policy.monash.edu.au/policy-bank/academic/education/index.html

Key educational policies include:

- Student Academic Integrity Policy and Student Academic Integrity: Managing Plagiarism and Collusion Procedures ; http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-policy.h
- Assessment in Coursework Programs; http://www.policy.monash.edu/policy-bank/academic/education/assessment/assessment-in-coursework-po
- Special Consideration; http://www.policy.monash.edu/policy-bank/academic/education/assessment/special-consideration-policy.ht
- Grading Scale; http://www.policy.monash.edu/policy-bank/academic/education/assessment/grading-scale-policy.html
- Discipline: Student Policy; http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-discipline-policy.html
- Academic Calendar and Semesters; http://www.monash.edu.au/students/dates/
- Orientation and Transition; http://intranet.monash.edu.au/infotech/resources/students/orientation/
- Academic and Administrative Complaints and Grievances Policy; http://www.policy.monash.edu/policy-bank/academic/education/management/complaints-grievance-policy.h

Faculty resources and policies

Important student resources including Faculty policies are located at http://intranet.monash.edu.au/infotech/resources/students/

Graduate Attributes Policy

http://www.policy.monash.edu/policy-bank/academic/education/management/monash-graduate-attributes-policy.ht

Student Charter

www.opq.monash.edu.au/ep/student-charter/monash-university-student-charter.html

Student services

The University provides many different kinds of support services for you. Contact your tutor if you need advice and see the range of services available at http://www.monash.edu.au/students. For Malaysia see http://www.monash.edu.my/Student-services, and for South Africa see http://www.monash.ac.za/current/.

Monash University Library

The Monash University Library provides a range of services, resources and programs that enable you to save time and be more effective in your learning and research. Go to www.lib.monash.edu.au or the library tab in my.monash portal for more information. At Malaysia, visit the Library and Learning Commons at http://www.lib.monash.edu.my/. At South Africa visit http://www.lib.monash.ac.za/.

Disability Liaison Unit

Students who have a disability or medical condition are welcome to contact the Disability Liaison Unit to discuss academic support services. Disability Liaison Officers (DLOs) visit all Victorian campuses on a regular basis.

- Website: http://www.monash.edu/equity-diversity/disability/index.html
- Telephone: 03 9905 5704 to book an appointment with a DLO; or contact the Student Advisor, Student Commuity Services at 03 55146018 at Malaysia
- Email: dlu@monash.edu
- Drop In: Equity and Diversity Centre, Level 1, Building 55, Clayton Campus, or Student Community Services Department, Level 2, Building 2, Monash University, Malaysia Campus