MONASH University
Information Technology

FIT2024
Software engineering practice

Unit Guide

Semester 1, 2010

The information contained in this unit guide is correct at time of publication. The University has the right to change any of the elements contained in this document at any time.

Last updated: 25 Feb 2010

Table of Contents

FIT2024 Software engineering practice - Semester 1, 2010

Chief Examiner:

Dr David Squire
Senior Lecturer
Phone: +61 3 990 59013
Fax: +61 3 990 55159

Lecturer(s) / Leader(s):

Caulfield

Dr David Squire
Senior Lecturer
Phone: +61 3 990 59013
Fax: +61 3 990 55159

Introduction

Welcome to FIT2024 Software Engineering Practice. This 6 point unit is core to the Bachelor of Software Engineering and the BITS Systems Development specialization.

Unit synopsis

This unit develops students understanding of and skills in professional Software Engineering practices at the personal level. Students experience work in a team environment and extend their programming skills by learning a new object oriented language and maintaining a system that is larger than their experience in prior units. Students develop skills in estimating, monitoring, reviewing and reporting on practical projects.

Learning outcomes

At the completion of this unit students will have -
A knowledge and understanding of:

- the Personal Software Process and its benefits, including the need for planning, estimation, recording time, product and defect metrics, reviews, and reflection;
- the importance of, and the relationship between, a quality process and a quality product;
- reinforceing and extending their knowledge of OO programming concepts by learning how they are implemented in another programming language;
- the Software Engineers role in software development and maintenance and working with large systems;
- the Team Software Process and how it relates to the Personal Software Process.

Developed attitudes that enable them to:

- develop a positive professional attitude;
- recognise the importance of adhering to software engineering principles in designing and implementing systems;

Developed the skills to:

- make personal estimates and work plans, produce work logs and diaries, produce product and defect metrics, and participate in technical review meetings;
- monitor, reflect upon, and improve their own productivity and effectiveness;
- use a new object oriented programming language to construct systems consisting of many interacting classes;
- analyse, debug and perform maintenance on large existing object-oriented programs.

Demonstrated the communication skills necessary to:

- be able to produce reports on their plans, progress, and reviews;
- be able to work effectively in small teams, and cooperatively with other teams.

Contact hours

2 hrs lectures/wk, 2 hrs laboratories/wk

Workload

Workload commitments are:

- two-hour lecture and
- two-hour tutorial (or laboratory) (requiring advance preparation)
- a minimum of 2-3 hours of personal study per one hour of contact time in order to satisfy the reading and assignment expectations.

You will need to allocate up to 5 hours per week in some weeks, for use of a computer, including time for newsgroups/discussion groups.

Unit relationships

Prerequisites

FIT1007 or FIT1008 or FIT1015 or CSE1303 or CSE1203 or BUS2011 or FIT2034

Prohibitions

CSE2201, GCO3811, FIT3037

Teaching and learning method

Teaching approach

Each week there will be a two-hour lecture, and a two-hour tutorial class. During lectures, the lecturer will present the theory underlying the topics in the unit, and illustrate this with concrete examples. During the tutorial classes, students will work on the tutorial exercises provided each week, as well as receiving feedback on their work from the previous week. As the semester progresses, tutorial time will also be devoted to working on, and receiving feedback on, the major assignment. Students are expected to complete most of their assignment work outside tutorial times, so that the tutorial time itself can be used efficiently on the supporting tutorial exercises and getting feedback from the tutor.

Timetable information

For information on timetabling for on-campus classes please refer to MUTTS, http://mutts.monash.edu.au/MUTTS/

Tutorial allocation

On-campus students should register for tutorials/laboratories using the Allocate+ system: http://allocate.its.monash.edu.au/

Unit Schedule

Week	Date*	Topic	References/Readings	Key dates
1	01/03/10	Introduction to Eiffel	Pressman Ch 8.4; Meyer; Weiner; Walden; Jezequel	
2	08/03/10	SE & Quality; Eiffel	Pressman Ch 1, 26, 32; Meyer, Weiner, Walden, Jezequel	Exercise 1 due
3	15/03/10	SE Process, PSP; Eiffel	Pressman Ch 2, 3, 4, 13.7; Somerville Ch 4, 13.7, 28; Humphrey (1990); Humphrey (1995); Humphrey (1996); Humphrey (1998)	Exercise 2 due
4	22/03/10	Specification; Configuration Management	Pressman Ch 9, 27; Somerville Ch 29; Meyer; Weiner; Walden; Jezequel	Exercise 3 due
5	29/03/10	Project & Quality Management; Testing;	Pressman Ch 13-14, 21-27; Somerville Ch 5, 23, 27; Pfleeger Ch 3, 8, 9; Peters, Ch 4, 12; ISO9000; TickIT	Exercise 4 due; Hand out Assignment
Mid semester break				
6	12/04/10	Writing Test Plans and Test Reports		

				Exercise 5 and PIP due
7	19/04/10	Software Metrics	Pressman Ch 15,22; Somerville pp 208-210, Ch 26; Peters Ch 13, 14; Fenton	Stage 1-Planning due
8	26/04/10	Software Maintenance; Software Design	Pressman Ch 31.2; Somerville pp 492-498; Peters Ch 18; Pfleeger Ch 11; Arthur; Meyer	Stage 1-Implementation due
9	03/05/10	SQA & Reviews (PSP2)	Pressman Ch 15.1, 26; Somerville Ch 27	
10	10/05/10	Software Reuse	Pressman p 32, Ch 9.5, 30.4.3, 30.5.2; Somerville Ch 18	Stage 2-Planning due
11	17/05/10	Software Performance	Pressman Ch 13.6.4, 20.9; Pfleeger Ch 9.3	
12	24/05/10	SE Tools	Pressman pp 54 & sidebars (see lecture notes); Maciaszek Ch 3; Parkinson Ch 1-2	Stage 2-Implementation due
13	31/05/10	Semester Review		

*Please note that these dates may only apply to Australian campuses of Monash University. Off-shore students need to check the dates with their unit leader.

Improvements to this unit

The workload for the assignment will be reduced, particularly the heavy-weight documentation requirements.

Unit Resources

Prescribed text(s) and readings

Pressman, R. S., *Software Engineering: A Practitioner's Approach*, 6th Edition, McGraw-Hill, 2005. Text books are available from the Monash University Book Shops. Availability from other suppliers cannot be assured. The Bookshop orders texts in specifically for this unit. You are advised to purchase your text book early. Other texts are available in the library.

Recommended text(s) and readings

The most highly recommended texts for this unit are:

- Pressman, R. S., *Software Engineering: A Practitioner's Approach*, 6th Edition, McGraw-Hill, 2005. Text books are available from the Monash University Book Shops, and slso available in the Monash library. A later edition may also be used, but will not necessarily match the chapter references given in the lecture slides.
- Meyer, B., Object-Oriented Software Construction, 2nd Edition, Prentice-Hall, 1997.

Other texts you might find useful include:

- Pfleeger, S. L., Software Engineering Theory and Practice, 2nd Edition, Prentice Hall, 2001.
- Sommerville, I., Software Engineering 7th edition, Addison-Wesley, 2004.
- Humphrey, W.S., Introduction to the Personal Software Process, Addison Wesley 1997.
- Humphrey, W.S., A Discipline for Software Engineering, Addison Wesley 1995.
- Humphrey, W.S., Managing the Software Process, Addison Wesley 1990.
- Humphrey, W.S., Lovelace, M., Hoppes, R., Introduction to the Team Software Process, Addison Wesley, 1998.
- Peters, J. S., Pedrycz, W., Software Engineering, An Engineering Approach, J Wiley, 2000.
- Wiener R.S., Introduction to Computer Science with Eiffel, Prentice Hall 1996.

Further Reading:

- Bennett S, Skelton and Lunn, Schaum's Outlines UML, McGraw-Hill, 2001.
- Larman C, Applying UML and patterns, Prentice-Hall, 1999.
- Jacobson I, Booch G and Rumbaug J, The Unified Software Development Process , Addison-Wesley.
- Covey, S.R., The 7 Habits of Highly Effective People, Pocket Books, 1999.
- Brooks, F., The Mythical Man Month, Addison-Wesley, 1995.

Required software and/or hardware

You will need access to:

- EiffelStudio
    - Available for MS Windows, Linux, and Mac OS X
- TortoiseSVN (for MS Windows only)
    - or any other Subversion client
- A tool for creating UML diagrams, such as Visual Paradigm

Software may be downloaded via the FIT2024 MUSO website.

On-campus students may use this software which is installed in the computing labs. Information about computer use for students isavailable from the ITS Student Resource Guide in the Monash University Handbook.

Equipment and consumables required or provided

Students may use the facilities available in the computing labs. Information about computer use for students is available from the ITS Student Resource Guide in the Monash University Handbook. You will need to allocate up to 8 hours per week for use of a computer, including time for newsgroups/discussion groups.

Study resources

Study resources we will provide for your study are:

The FIT2024 web site on MUSO, where lecture slides, weekly tutorial requirements, assignment specifications, sample solutions and supplementary material will be posted.

Other useful web sites are:

- Interactive Software Engineering (ISE) Eiffel Site: http://www.eiffel.com/
- Eiffel newsgroup: comp.lang.eiffel

Assessment

Overview

Examination (2 hours): 40%; In-semester assessment: 60%

Faculty assessment policy

To pass a unit which includes an examination as part of the assessment a student must obtain:

- 40% or more in the unit's examination, and
- 40% or more in the unit's total non-examination assessment, and
- an overall unit mark of 50% or more.

If a student does not achieve 40% or more in the unit examination or the unit non-examination total assessment, and the total mark for the unit is greater than 50% then a mark of no greater than 49-N will be recorded for the unit.

In addition to the Faculty policy, to pass this unit a student must:

- Attain at least 40% of the available marks in the Individual Exercises
- Attain at least 40% of the available marks for the Software Engineering Team Assignment
- Attend *at least* 10 out of the 13 tutorials.

If a student does not meet these criteria then a mark of no greater than 49-N will be recorded for the unit.

Assignment tasks

Assignment coversheets

Assignment coversheets are available via "Student Forms" on the Faculty website:
http://www.infotech.monash.edu.au/resources/student/forms/
You MUST submit a completed coversheet with all assignments, ensuring that the plagiarism declaration section is signed.

Assignment submission and return procedures, and assessment criteria will be specified with each assignment.

- **Assignment task 1**

  Title:
  Individual Exercises
  Description:
  Five (5) programming exercises will be set in weeks 1 through 5. Each of the first four exercises are due at the following week's tutorial, where they will be briefly assessed. Students must attend tutorials to submit your work and receive feedback. The exercises are designed to improve object-oriented programming and testing skills, and knowledge of Eiffel, prior to the commencement of the major assignment work. The Personal Software Process (PSP) will be introduced as part of these exercises, and a Process Improvement Proposal (PIP) will be due with the final exercise is assessed in week 6.
  Weighting:
  10%

Due date:

Each exercise is due one week after the tutorial in which it is set.

- **Assignment task 2**

Title:

Team Project

Description:

Teams will be provided with an existing system written in Eiffel. The project has two (2) stages. In the first stage, they will be required to debug the existing code, and extend it to implement some new functionality. In the second stage further requirements will be added, and students will have to design and implement changes to the system to meet these new requirements. All work must be planned, measured and tested according to the Personal Software Process (PSP). Each stage consists of a software engineering planning and design component, and an implementation component. These components are submitted and marked separately:

- ♦Stage 1 Planning and Design, 10%, week 7
- ♦Stage 1 Implementation and Testing, 10%, week 8
- ♦Stage 2 Planning and Design, 15%, week 10
- ♦Stage 2 Implementation and Testing, 15%, week 12

Weighting:

10 + 10 + 15 + 15 = 50%

Due date:

As above.

Examination

- **Weighting:** 40%
  Length: 2 hours
  Type (open/closed book): Closed book

See Appendix for End of semester special consideration / deferred exams process.

Due dates and extensions

Please make every effort to submit work by the due dates. It is your responsibility to structure your study program around assignment deadlines, family, work and other commitments. Factors such as normal work pressures, vacations, etc. are not regarded as appropriate reasons for granting extensions. Students are advised to NOT assume that granting of an extension is a matter of course.

Students requesting an extension for any assessment during semester (eg. Assignments, tests or presentations) are required to submit a Special Consideration application form (in-semester exam/assessment task), along with original copies of supporting documentation, directly to their lecturer within two working days before the assessment submission deadline. Lecturers will provide specific outcomes directly to students via email within 2 working days. The lecturer reserves the right to refuse late applications.

A copy of the email or other written communication of an extension must be attached to the assignment submission.

Refer to the Faculty Special consideration webpage or further details and to access application forms: http://www.infotech.monash.edu.au/resources/student/equity/special-consideration.html

Late assignment

Assignments received after the due date and time will be subject to a penalty of:

- 15% for work handed in late, but within 24 hours of the due date and time;
- Zero (0) marks for work handed in more than 24 hours late.
- Work handed in after 24 hours will not normally be accepted, so students will not be assessed on it.

This policy is strict because comments or guidance will be given on assignments as they are returned, and sample solutions may also be published and distributed, after assignment marking or with the returned assignment.

Return dates

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

Appendix

Please visit the following URL: http://www.infotech.monash.edu.au/units/appendix.html for further information about:

- Continuous improvement
- Unit evaluations
- Communication, participation and feedback
- Library access
- Monash University Studies Online (MUSO)
- Plagiarism, cheating and collusion
- Register of counselling about plagiarism
- Non-discriminatory language
- Students with disability
- End of semester special consideration / deferred exams