MONASH University

FIT2024
Software engineering practice

Unit guide

Semester 1, 2009

Last updated : 20 Apr 2009

Table of Contents

FIT2024 Software engineering practice - Semester 1, 2009

Unit leader :

David Squire

Lecturer(s) :

Caulfield

- David Squire

Introduction

Unit synopsis

The focus in this unit is on the application of tools and techniques used by software engineers in the development of software systems. Students gain experience with the software process, particularly PSP, the Personal Software Process; estimation in project management; software metrics; software testing and debugging techniques; and software maintenance. Students also gain an appreciation of project, quality, and configuration management. Programming concepts are revisited with more advanced problems and solutions focusing on the design, development, and testing of object-oriented systems. Case study work involves the development and maintenance of a software system using the above software engineering techniques and object-oriented methods.

Learning outcomes

At the completion of this unit students should have further developed their skills as software engineers and have a deeper appreciation of software engineering principles in the development and maintenance of computer systems; have further developed their skills in problem-solving and software measurement; recognise the role of, and be able to use, specific software engineering tools and techniques such as quality plans, project plans, test plans, software metrics models, configuration management tools and development environments in the software engineering process; be able to measure their own improvement using the personal software process; have acquired an appreciation of the application of software engineering principles including software process, quality management, software metrics, configuration management, performance, testing and debugging, technical reviews, reusability and change management.

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

Before attempting this unit you must have satisfactorily completed CSE1203 or (CSE1401 and CSE1303) or BUS2011, or equivalent. You should have knowledge of basic object-oriented programming and design.

Relationships

FIT2024 is a core unit in in the Bachelor of Software Engineering, and in the System Development major of the Bachelor of ITS (Caulfield). It is a prerequisite for FIT3057 in the Bachelor of ITS.

You may not study this unit and CSE2401, GCO3811, RDT2231, SFT2201, or CSE2201 in your degree.

Continuous improvement

Monash is committed to 'Excellence in education' (Monash Directions 2025 - http://www.monash.edu.au/about/monash-directions/directions.html) and strives for the highest possible quality in teaching and learning.

To monitor how successful we are in providing quality teaching and learning Monash regularly seeks feedback from students, employers and staff. One of the key formal ways students have to provide feedback is through Unit Evaluation Surveys. The University's Unit Evaluation policy (http://www.policy.monash.edu/policy-bank/academic/education/quality/unit-evaluation-policy.html) requires that every unit offered is evaluated each year. Students are strongly encouraged to complete the surveys as they are an important avenue for students to "have their say". The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied and areas for improvement.

Faculties have the option of administering the Unit Evaluation survey online through the my.monash portal or in class. Lecturers will inform students of the method being used for this unit towards the end of the semester.

Student Evaluations

If you wish to view how previous students rated this unit, please go to http://www.adm.monash.edu.au/cheq/evaluations/unit-evaluations/

Improvements to this unit

Some topics have been updated to reflect changes in the state-of-the art in this area. Some topics have been enhanced with material previously taught in CSE3308, but not present in the new FIT3077.

Further updates will be made this semester to reflect advances in the Eiffel programming language, and the development of service-oriented software engineering.

Unit staff - contact details

Unit leader

Dr David Squire
Senior Lecturer
Phone +61 3 990 59013
Fax +61 3 990 55157

Lecturer(s) :

Dr David Squire
Senior Lecturer
Phone +61 3 990 59013
Fax +61 3 990 55157
Contact hours : Consultation hours will be given in the lectures and on Blackboard. The best way to contact me is via email.

Teaching and learning method

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.

Tutorial allocation

On-campus students should register for tutorials/laboratories using Allocate+.

Communication, participation and feedback

Monash aims to provide a learning environment in which students receive a range of ongoing feedback throughout their studies. You will receive feedback on your work and progress in this unit. This may take the form of group feedback, individual feedback, peer feedback, self-comparison, verbal and written feedback, discussions (on line and in class) as well as more formal feedback related to assignment marks and grades. You are encouraged to draw on a variety of feedback to enhance your learning.

It is essential that you take action immediately if you realise that you have a problem that is affecting your study. Semesters are short, so we can help you best if you let us know as soon as problems arise. Regardless of whether the problem is related directly to your progress in the unit, if it is likely to interfere with your progress you should discuss it with your lecturer or a Community Service counsellor as soon as possible.

Unit Schedule

Week	Topic	References/Readings	Key dates
1	Introduction to Eiffel	Pressman Ch 8.4; Meyer; Weiner; Walden; Jezequel	
2	SE & Quality; Eiffel	Pressman Ch 1, 26, 32; Meyer, Weiner, Walden, Jezequel	Exercise 1 due

3	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	Specification; Configuration Management	Pressman Ch 9, 27; Somerville Ch 29; Meyer; Weiner; Walden; Jezequel	Exercise 3 due
5	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
6	UNIT TEST		
Mid semester break			
7	Software Metrics	Pressman Ch 15,22; Somerville pp 208-210, Ch 26; Peters Ch 13, 14; Fenton	Stage 1-Planning due
8	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	SQA & Reviews (PSP2)	Pressman Ch 15.1, 26; Somerville Ch 27	
10	Software Reuse	Pressman p 32, Ch 9.5, 30.4.3, 30.5.2; Somerville Ch 18	Stage 2-Planning due
11	Software Performance	Pressman Ch 13.6.4, 20.9; Pfleeger Ch 9.3	
12	SE Tools	Pressman pp 54 & sidebars (see lecture notes); Maciaszek Ch 3; Parkinson Ch 1-2	Stage 2-Implementation due
13	Semester Review		

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

- Meyer, B., Object-Oriented Software Construction, 2nd Edition, Prentice-Hall, 1997.
- 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 Blackboard website.

On-campus students may use this software which is installed in thecomputing 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 Blackboard, 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

Library access

The Monash University Library site contains details about borrowing rights and catalogue searching. To learn more about the library and the various resources available, please go to http://www.lib.monash.edu.au.

The Educational Library and Media Resources (LMR) is also a very resourceful place to visit at http://www.education.monash.edu.au/library/

Monash University Studies Online (MUSO)

All unit and lecture materials are available through MUSO (Monash University Studies Online). Blackboard is the primary application used to deliver your unit resources. Some units will be piloted in Moodle. If your unit is piloted in Moodle, you will see a link from your Blackboard unit to Moodle (http://moodle.monash.edu.au) and can bookmark this link to access directly. In Moodle, from the Faculty of Information Technology category, click on the link for your unit.

You can access MUSO and Blackboard via the portal: http://my.monash.edu.au

Click on the Study and enrolment tab, then Blackboard under the MUSO learning systems.

In order for your Blackboard unit(s) to function correctly, your computer needs to be correctly configured.

For example:

- Blackboard supported browser
- Supported Java runtime environment

For more information, please visit: http://www.monash.edu.au/muso/support/students/downloadables-student.html

You can contact the MUSO Support by phone : (+61 3) 9903 1268

For further contact information including operational hours, please visit: http://www.monash.edu.au/muso/support/students/contact.html

Further information can be obtained from the MUSO support site: http://www.monash.edu.au/muso/support/index.html

Assessment

Unit assessment policy

Assessment in this unit has two components:

Component A: 50%

- Final Exam (30%)
- Unit Test and Exercises (20%)

Component B: 50%

- Software Engineering Team Assignment, including Planning, Programming & Documentation (50%)

To pass this unit you must:

- Attain 50% overall, **and**
- Attain at least 40% of the available marks in both Components A & B, **and**
- Attend *at least* 10 out of the 13 tutorials.

If a student does not meet these criteria then a mark of no greater than 44-N will be recorded for the unit.

Assignment tasks

- ### Assignment Task

  Title : Individual Exercises

  Description :

  Four (4) programming exercises will be set in weeks 1 through 4. 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 in week 5.
  Weighting : 5%

  Criteria for assessment :

  Each program will be assessed for correctness with respect to its functional specification, as well as for good programming style and object-oriented design. Each programming exercise is worth 1%. The PIP is also worth 1%.
  Due date : Each exercise is due one week after the tutorial in which it is set.
- ### Assignment Task

  Title : Unit Test

  Description :

  The Unit Test will be conducted online during tutorial time in week 6, using Blackboard. It will be open-book. It will consist of two sections. The first section will examine knowledge of Object-Oriented Software Engineering principles, and their implementation in Eiffel, using both multiple choice and short answer questions. For the second part, the student will have to examine an Eiffel program (provided during the test), answer questions about the code and its design, and then to extend it to provided new functionality.
  Weighting : 15%

  Criteria for assessment :

  Questions will be assessed for correctness. Submitted code will be assessed with respect to the functional specification, coding style, and design quality.
  Due date : Week 6

- **Assignment Task**

  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%

  Criteria for assessment :

  Plans will be assessed for completeness with respect to the requirements and the PSP. Designs and code will be assessed for completeness with respect to the requirements and quality of object-oriented design and coding style. All PSP metrics and test documentation will be assessed for completeness, accuracy and conformance to PSP guidelines.
  Due date : As above.

- **Assignment Task**

  Title : Team Assignment Stage 2 - Planning & Design

  Description :

  Weighting : 15%

  Criteria for assessment :

  Due date :

- **Assignment Task**

  Title : Team Assignment Stage 2 - Implementation & Team Interview

  Description :

  Weighting : 15%

  Criteria for assessment :

  Due date :

Examinations

- **Examination 1**

  Weighting : 30%

  Length : 2 hours

  Type (open/closed book) : Closed book

Assignment submission

Assignments will be submitted on-line via Blackboard.

University and Faculty policy on assessment

Due dates and extensions

The due dates for the submission of assignments are given in the previous section. 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 seldom regarded as appropriate reasons for granting extensions. Students are advised to NOT assume that granting of an extension is a matter of course.

Requests for extensions must be made to the unit lecturer at your campus **before the due date**. You will be asked to forward original medical certificates in cases of illness, and may be asked to provide other forms of documentation where necessary. A copy of the email or other written communication of an extension must be attached to the assignment submission.

Late assignment

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

- 15% for work handed in after 10 minutes into the lecture but within 24 hours;
- 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.

Assessment for the unit as a whole is in accordance with the provisions of the Monash University Education Policy at http://www.policy.monash.edu/policy-bank/academic/education/assessment/

We will aim to have assignment results made available to you within two weeks after assignment receipt.

Plagiarism, cheating and collusion

Plagiarism and cheating are regarded as very serious offences. In cases where cheating has been confirmed, students have been severely penalised, from losing all marks for an assignment, to facing disciplinary action at the Faculty level. While we would wish that all our students adhere to sound ethical conduct and honesty, I will ask you to acquaint yourself with the University Plagiarism policy and procedure (http://www.policy.monash.edu/policy-bank/academic/education/conduct/plagiarism-procedures.html) which applies to students detected plagiarising.

In this University, cheating means seeking to obtain an unfair advantage in any examination or any other written or practical work to be submitted or completed by a student for assessment. It includes the use, or attempted use, of any means to gain an unfair advantage for any assessable work in the unit, where the means is contrary to the instructions for such work.

When you submit an individual assessment item, such as a program, a report, an essay, assignment or other piece of work, under your name you are understood to be stating that this is your own work. If a submission is identical with, or similar to, someone else's work, an assumption of cheating may arise. If you are planning on working with another student, it is acceptable to undertake research together, and discuss problems, but it is not acceptable to jointly develop or share solutions unless this is specified by your lecturer.

Intentionally providing students with your solutions to assignments is classified as "assisting to cheat" and students who do this may be subject to disciplinary action. You should take reasonable care that your solution is not accidentally or deliberately obtained by other students. For example, do not leave copies of your work in progress on the hard drives of shared computers, and do not show your work to other students. If you believe this may have happened, please be sure to contact your lecturer as soon as possible.

Cheating also includes taking into an examination any material contrary to the regulations, including any bilingual dictionary, whether or not with the intention of using it to obtain an advantage.

Plagiarism involves the false representation of another person's ideas, or findings, as your own by either copying material or paraphrasing without citing sources. It is both professional and ethical to reference clearly the ideas and information that you have used from another writer. If the source is not identified, then you have plagiarised work of the other author. Plagiarism is a form of dishonesty that is insulting to the reader and grossly unfair to your student colleagues.

Register of counselling about plagiarism

The university requires faculties to keep a simple and confidential register to record counselling to students about plagiarism (e.g. warnings). The register is accessible to Associate Deans Teaching (or nominees) and, where requested, students concerned have access to their own details in the register. The register is to serve as a record of counselling about the nature of plagiarism, not as a record of allegations; and no provision of appeals in relation to the register is necessary or applicable.

Non-discriminatory language

The Faculty of Information Technology is committed to the use of non-discriminatory language in all forms of communication. Discriminatory language is that which refers in abusive terms to gender, race, age, sexual orientation, citizenship or nationality, ethnic or language background, physical or mental ability, or political or religious views, or which stereotypes groups in an adverse manner. This is not meant to preclude or inhibit legitimate academic debate on any issue; however, the language used in such debate should be non-discriminatory and sensitive to these matters. It is important to avoid the use of discriminatory language in your communications and written work. The most common form of discriminatory language in academic work tends to be in the area of gender inclusiveness. You are, therefore, requested to check for this and to ensure your work and communications are non-discriminatory in all respects.

Students with disabilities

Students with disabilities that may disadvantage them in assessment should seek advice from one of the following before completing assessment tasks and examinations:

- Faculty of Information Technology Student Service staff, and / or
- your Unit Coordinator, or
- Disabilities Liaison Unit

Deferred assessment and special consideration

Deferred assessment (not to be confused with an extension for submission of an assignment) may be granted in cases of extenuating personal circumstances such as serious personal illness or bereavement. Information and forms for Special Consideration and deferred assessment applications are available at http://www.monash.edu.au/exams/special-consideration.html. Contact the Faculty's Student Services staff at your campus for further information and advice.