



COMP1050

Introduction to the Study of Software Engineering

Session 1, In person-scheduled-weekday, North Ryde 2025

School of Computing

Contents

| | |
|---|---|
| <u>General Information</u> | 2 |
| <u>Learning Outcomes</u> | 2 |
| <u>General Assessment Information</u> | 3 |
| <u>Assessment Tasks</u> | 4 |
| <u>Delivery and Resources</u> | 6 |
| <u>Unit Schedule</u> | 7 |
| <u>Policies and Procedures</u> | 8 |
| <u>Changes from Previous Offering</u> | 9 |

Disclaimer

Macquarie University has taken all reasonable measures to ensure the information in this publication is accurate and up-to-date. However, the information may change or become out-dated as a result of change in University policies, procedures or rules. The University reserves the right to make changes to any information in this publication without notice. Users of this publication are advised to check the website version of this publication [or the relevant faculty or department] before acting on any information in this publication.

General Information

Unit convenor and teaching staff

primary convenor and lecturer

Kate Stefanov

kate.stefanov@mq.edu.au

co-convenor and lecturer

Carl Svensson

carl.svensson@mq.edu.au

Credit points

10

Prerequisites

Admission to BEng

Corequisites

Co-badged status

Unit description

This unit covers the fundamentals of software engineering, including understanding system requirements, finding appropriate engineering compromises, learning software engineering culture, forming camaraderie, understanding basic methods of design, coding, and testing, team software development, and the application of engineering tools.

Important Academic Dates

Information about important academic dates including deadlines for withdrawing from units are available at <https://www.mq.edu.au/study/calendar-of-dates>

Learning Outcomes

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

ULO1: Work with version control, configuration management, unit/regression testing, issue tracking, and debugging tools.

ULO2: Create a project plan.

ULO3: Create and analyse design models.

ULO4: Make engineering tradeoffs.

ULO5: Demonstrate an understanding of software engineering culture and form camaraderie.

General Assessment Information

The assessment tasks include an individual submission, a group written submission and a final exam. These are unlikely to be successfully completed unless students actively participate in lectures, workshops and the group project.

Working in teams on the Group-based Software Engineering Project mostly takes place in the weekly workshops. You are required to participate in the workshop tasks, either individual or group, and contribute to the submission at the end of each workshop. Attendance will be taken, and your name will be associated with the weekly submissions only for the weeks that you have attended the workshop.

Requirements to Pass

To pass the unit, you will have to achieve a total mark equal to or greater than 50%.

Late Assessment Submission Penalty

The Individual Software Engineering Assignment and the Final report for the Group-based Software Engineering Project are to be uploaded by the due date. Unless a Special Consideration request has been submitted and approved, a 5% penalty (of the total possible mark of the task) will be applied for each day a written report or presentation assessment is not submitted, up until the 7th day (including weekends). After the 7th day, a grade of '0' will be awarded even if the assessment is submitted. The submission time for all uploaded assessments is **11:55 pm**. A 1-hour grace period will be provided to students who experience a technical concern. For any late submission of time-sensitive tasks, such as scheduled tests/exams, performance assessments/presentations, and/or scheduled practical assessments/labs, please apply for [Special Consideration](#). For example, if the assignment is worth 8 marks (of the entire unit) and your submission is late by 19 hours (or 23 hours 59 minutes 59 seconds), 0.4 marks (5% of 8 marks) will be deducted. If your submission is late by 24 hours (or 47 hours 59 minutes 59 seconds), 0.8 marks (10% of 8 marks) will be deducted, and so on.

Assessments where Late Submissions will be accepted

- Individual Software Engineering Assignment – YES, Standard Late Penalty applies unless Special Consideration has been granted
- Final report for the Group-based Software Engineering Project – YES, Standard Late Penalty applies unless Special Consideration has been granted

If you fail to sit the Final exam, you could apply for Special Consideration and if granted, you will be invited to sit a Supplementary exam.

Checkpoints

Both the Individual Software Engineering Assignment and the Final report for the Group-based Software Engineering Project have half-way checkpoints as an opportunity for students to receive feedback on their work. The due dates for the checkpoints are:

- Part 1 of Individual Software Engineering Assignment: Week 6, Wednesday 2 April 23:55

- Interim report for the Group-based Software Engineering Project: Week 7, Wednesday 9 April 23:55

While there are no marks associated with these submissions, late submissions may not receive feedback, which is likely to have a negative impact on the final mark for that assessment.

Release dates

- Individual Software Engineering Assignment: No later than Wednesday 5 March, Week 2
- Group-based Software Engineering Project: No later than Wednesday 5 March, Week 2

Special Consideration

The [Special Consideration Policy](#) aims to support students who have been impacted by short-term circumstances or events that are serious, unavoidable and significantly disruptive, and which may affect their performance in assessment. If you experience circumstances or events that affect your ability to complete the assessments in this unit on time, please inform the convenor and submit a Special Consideration request through <http://connect.mq.edu.au/>.

Assessment Tasks

| Name | Weighting | Hurdle | Due |
|--|-----------|--------|---------------------------------|
| Group-based Software Engineering Project | 30% | No | Week 13, Wednesday 4 June 23:55 |
| Final Exam | 40% | No | Exam period |
| Individual Software Engineering Assignment | 30% | No | Week 12, Wednesday 28 May 23:55 |

Group-based Software Engineering Project

Assessment Type ¹: Project

Indicative Time on Task ²: 23 hours

Due: **Week 13, Wednesday 4 June 23:55**

Weighting: **30%**

Students work in groups to develop week-by-week a software engineering project primarily during their scheduled class. Students will both learn the principles and practices of software engineering, and gain an overview of a wide range of software engineering areas (that are studied in much greater depth in individual units later in their program). All marks are for group work.

On successful completion you will be able to:

- Work with version control, configuration management, unit/regression testing, issue tracking, and debugging tools.
- Create a project plan.
- Create and analyse design models.
- Make engineering tradeoffs.
- Demonstrate an understanding of software engineering culture and form camaraderie.

Final Exam

Assessment Type ¹: Examination

Indicative Time on Task ²: 19 hours

Due: **Exam period**

Weighting: **40%**

An invigilated examination of the unit's content and the students' experiences of the individual and group activities.

On successful completion you will be able to:

- Work with version control, configuration management, unit/regression testing, issue tracking, and debugging tools.
- Create a project plan.
- Create and analyse design models.
- Make engineering tradeoffs.
- Demonstrate an understanding of software engineering culture and form camaraderie.

Individual Software Engineering Assignment

Assessment Type ¹: Problem set

Indicative Time on Task ²: 30 hours

Due: **Week 12, Wednesday 28 May 23:55**

Weighting: **30%**

An opportunity to demonstrate the learning achieved in the second half of the unit's lectures

On successful completion you will be able to:

- Work with version control, configuration management, unit/regression testing, issue tracking, and debugging tools.

- Create a project plan.
 - Create and analyse design models.
 - Make engineering tradeoffs.
-

¹ If you need help with your assignment, please contact:

- the academic teaching staff in your unit for guidance in understanding or completing this type of assessment
- the [Writing Centre](#) for academic skills support.

² Indicative time-on-task is an estimate of the time required for completion of the assessment task and is subject to individual variation

Delivery and Resources

Week 1 Classes

Both lectures and workshops start in Week 1.

Weekly Workshops

There are weekly workshops with set tasks and time allocated to working on the assessment task *Group-based Software Engineering Project*. You will need to conscientiously attend the workshops and engage with the work, with your TA and fellow students. We expect you to be involved every week. Attendance will be taken, and your name will be associated with the weekly submissions only for the weeks that you have attended the workshop.

Lectures

This course includes 2-hour weekly lectures. Lecture attendance is highly recommended as they introduce the concepts that will be used in the workshops and will be covered in the assessments in the unit.

Assignments

Assignments play a crucial role in evaluating student understanding. They are based on the lecture material, workshop activities and weekly tutorial material, and require students to integrate what they have learned, think critically and creatively.

Final exam

The final exam is designed to test student understanding of the course content and the application of the concepts to a number of scenarios or problem statements.

The exam is held during the exam period.

Unit Resources

Although there is no required textbook, we suggest that students read:

- David Farley, *Modern Software Engineering: Doing What Works to Build Better Software Faster*, Addison-Wesley Professional, 2021.

This is a valuable resource for understanding the day-to-day tasks of software engineers. The book is available in the library in both online and ebook formats.

The workshop exercises and the software engineering project are outlined in the workshop manual. Additionally, lecture notes and recordings will be made available after each lecture.

Methods of Communication

We will communicate with you via your university email, through announcements on iLearn and in lectures. Queries to convenors can either be placed on the iLearn discussion board or sent to the unit convenor via the contact email on iLearn.

Unit Schedule

Students are expected to attend a two-hour lecture and a two-hour workshop every week. The topics covered week by week are outlined below (subject to change):

| Week | Lectures | Deadlines |
|-------------|--|--|
| 1 | Introduction, What is Software Engineering | |
| 2 | SDLC and Software Processes | Individual and group assignments out |
| 3 | Requirements and epics, Issue Tracking | |
| 4 | Version control and debugging | |
| 5 | Software Testing, Code Review | |
| 6 | Software Testing (Unit and Regression) | Part 1 Individual Assignment Checkpoint (for feedback) |
| 7 | Testing in the SDLC, Project Management | Interim group project report (for feedback) |
| | mid-semester break | |
| | mid-semester break | |
| 8 | Software Architecture and Design Models | |
| 9 | Software modelling and Software Quality | |
| 10 | High Level Software Architecture | |
| 11 | Software Traceability and Configuration Management | |
| 12 | Professionalism and Ethics | Individual Assignment |
| 13 | Review and project final presentation | Group project report |
| Exam Period | | Final exam |

Policies and Procedures

Macquarie University policies and procedures are accessible from [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au). Students should be aware of the following policies in particular with regard to Learning and Teaching:

- [Academic Appeals Policy](#)
- [Academic Integrity Policy](#)
- [Academic Progression Policy](#)
- [Assessment Policy](#)
- [Fitness to Practice Procedure](#)
- [Assessment Procedure](#)
- [Complaints Resolution Procedure for Students and Members of the Public](#)
- [Special Consideration Policy](#)

Students seeking more policy resources can visit [Student Policies \(https://students.mq.edu.au/support/study/policies\)](https://students.mq.edu.au/support/study/policies). It is your one-stop-shop for the key policies you need to know about throughout your undergraduate student journey.

To find other policies relating to Teaching and Learning, visit [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au) and use the [search tool](#).

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: <https://students.mq.edu.au/admin/other-resources/student-conduct>

Results

Results published on platform other than [eStudent](#), (eg. iLearn, Coursera etc.) or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in [eStudent](#). For more information visit connect.mq.edu.au or if you are a Global MBA student contact globalmba.support@mq.edu.au

Academic Integrity

At Macquarie, we believe [academic integrity](#) – honesty, respect, trust, responsibility, fairness and courage – is at the core of learning, teaching and research. We recognise that meeting the expectations required to complete your assessments can be challenging. So, we offer you a range of resources and services to help you reach your potential, including free [online writing and maths support](#), [academic skills development](#) and [wellbeing consultations](#).

Student Support

Macquarie University provides a range of support services for students. For details, visit <http://students.mq.edu.au/support/>

The Writing Centre

The [Writing Centre](#) provides resources to develop your English language proficiency, academic writing, and communication skills.

- [Workshops](#)
- [Chat with a WriteWISE peer writing leader](#)
- [Access StudyWISE](#)
- [Upload an assignment to Studiosity](#)
- [Complete the Academic Integrity Module](#)

The Library provides online and face to face support to help you find and use relevant information resources.

- [Subject and Research Guides](#)
- [Ask a Librarian](#)

Student Services and Support

Macquarie University offers a range of [Student Support Services](#) including:

- [IT Support](#)
- [Accessibility and disability support](#) with study
- Mental health [support](#)
- [Safety support](#) to respond to bullying, harassment, sexual harassment and sexual assault
- [Social support including information about finances, tenancy and legal issues](#)
- [Student Advocacy](#) provides independent advice on MQ policies, procedures, and processes

Student Enquiries

Got a question? Ask us via the [Service Connect Portal](#), or contact [Service Connect](#).

IT Help

For help with University computer systems and technology, visit http://www.mq.edu.au/about_us/offices_and_units/information_technology/help/.

When using the University's IT, you must adhere to the [Acceptable Use of IT Resources Policy](#). The policy applies to all who connect to the MQ network including students.

Changes from Previous Offering

Assessments

The number of assessments has changed. The current offering has three assessments and the

hurdles have been removed.

The workshops and the software project

While the software engineering content is the same as previous years, the order in which topics are covered, the exercises, and the nature of the project may change.

Lectures

While the topics of the lectures remain the same, the order in which they are covered, and the application examples may change.

Repeating Students

Students who retake this unit are expected to fully participate in all learning activities, just as first-time students do. Previous participation in the unit does not exempt you from any requirements.

Unit information based on version 2025.03 of the [Handbook](#)