



COMP2050

Software Engineering

Session 2, In person-scheduled-weekday, North Ryde 2024

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>	7
<u>Unit Schedule</u>	8
<u>Policies and Procedures</u>	9
<u>Changes from Previous Offering</u>	11

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

Convenor, Lecturer

Carl Svensson

carl.svensson@mq.edu.au

Contact via iLearn forum, otherwise email

By appointment

Lecturer

James Zheng

james.zheng@mq.edu.au

Contact via iLearn forum, otherwise email

By appointment

Credit points

10

Prerequisites

60cp at 1000 level or above including COMP1010 or COMP125

Corequisites

Co-badged status

Unit description

This unit introduces engineering principles and practices to all stages of the software development lifecycle to ensure a systematic, quality-focused and quantifiable approach to the management, design, development, maintenance, verification and validation of [large and complex] software products, projects and processes. Problem formulation and solving are emphasised. Topics covered include: requirements gathering and specification; object-oriented modelling using the Unified Modeling Language (UML); process management; and software design, testing and evolution.

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: Use good software engineering practices to design code including unit testing

and quality documentation

ULO2: Effectively use modern software development techniques and tools

ULO3: Demonstrate knowledge of requirements elicitation techniques and the ability to apply those techniques to a range of problem domains

ULO4: Discuss the life cycle of software systems development and the impact of implementation issues on various phases of the life cycle

ULO5: Explain the principles, practices, ethics and responsibilities of Software Engineering

General Assessment Information

Late Assessment Submission Penalty

Unless a Special Consideration request has been submitted and approved, a 5% penalty (of the total possible mark) will be applied each day one of the two written 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. Submission time for all written assessments is set at **11:55 pm**. A 1-hour grace period is provided to students who experience a technical concern.

The weekly tasks should be completed before the following workshop. If they are submitted after the workshop, they will receive at most a passing grade.

For any late submission of time-sensitive tasks, such as scheduled tests/exams, performance assessments/presentations, and/or scheduled practical assessments/labs, students need to submit an application for [Special Consideration](#).

Assessments where Late Submissions will be accepted

- Assignment 1 – YES, Standard Late Penalty applies
- Assignment 2 – YES, Standard Late Penalty applies
- Weekly problem – YES, Late submissions will receive at most a passing grade.
- Workshop participation - NO, unless Special Consideration is granted

Hurdle Assessment

- **Assignment 1 (10%):** The assignment has 3 components: A report, a prepared presentation and a viva. The first two components are group exercises. The last is an individual assessment. The viva for assignment 1 is a hurdle tasks, to ensure engagement of all student in the group project. Student need at least a passing mark for this assignment to clear the hurdle.
- **Weekly Problem (20%):** To meet the hurdle requirement students must complete at least 8 of 12 weekly problems before the weekly deadline and participate in the workshop of that week.

Students who fail the hurdle requirement - by missing deadlines, or absence from workshops - are required to complete at least 8 of 12 weekly problems at a satisfactory level, followed by a viva on the content of the weekly problems at the end of the session.

Requirements to pass the unit

To pass this unit you must:

- Achieve a total mark equal to or greater than 50%, and
- Pass the hurdle requirement for the weekly tasks.
- Participate in the viva for Assignment 1.

Special Considerations

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.

Written Assessments: If you experience circumstances or events that affect your ability to complete the written assessments in this unit on time, please inform the convenor and submit a Special Consideration request through ask.mq.edu.au.

Weekly Tasks: To pass the unit you need to complete at least 8 of 12 weekly problems before the weekly deadline and participate in the workshop of that week. If you miss a weekly practical class due to a serious, unavoidable and significant disruption, contact the teaching staff of your registered class ASAP and catch up with your team.

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. Note that a Special Consideration should only be applied for if you miss more than three of the weekly practical classes.

Assessment Tasks

Name	Weighting	Hurdle	Due
Weekly Problem	20%	Yes	Weekly
Assignment 1	15%	Yes	Multiple in week 7 & 8 (see iLearn page for details)
Assignment 2	15%	No	W12 Sunday 11:55pm Submission, W13 class presentation
Final Exam	50%	No	Exam Period

Weekly Problem

Assessment Type ¹: Practice-based task

Indicative Time on Task ²: 36 hours

Due: **Weekly**

Weighting: **20%**

This is a hurdle assessment task (see [assessment policy](#) for more information on hurdle assessment tasks)

Each week from week 1 to week 12, problems or small tasks will be posted, Students must complete at least 8 of 12 of these tasks to be completed before the next workshop and attend the workshop where the tasks is discussed.

Students who fail the hurdle requirement - by missing deadlines, or absence from workshops - are required to complete at least 8 of 12 weekly problems at a satisfactory level, followed by a viva on the content of the weekly problems at the end of the session.

On successful completion you will be able to:

- Use good software engineering practices to design code including unit testing and quality documentation
- Effectively use modern software development techniques and tools
- Demonstrate knowledge of requirements elicitation techniques and the ability to apply those techniques to a range of problem domains
- Discuss the life cycle of software systems development and the impact of implementation issues on various phases of the life cycle
- Explain the principles, practices, ethics and responsibilities of Software Engineering

Assignment 1

Assessment Type ¹: Case study/analysis

Indicative Time on Task ²: 20 hours

Due: **Multiple in week 7 & 8 (see iLearn page for details)**

Weighting: **15%**

This is a hurdle assessment task (see [assessment policy](#) for more information on hurdle assessment tasks)

Assignment 1 will allow you to demonstrate the development of your understanding and your ability to apply the things that you have learned in the first part (weeks 1 to 6) of the unit.

On successful completion you will be able to:

- Use good software engineering practices to design code including unit testing and quality documentation
- Effectively use modern software development techniques and tools
- Demonstrate knowledge of requirements elicitation techniques and the ability to apply those techniques to a range of problem domains
- Explain the principles, practices, ethics and responsibilities of Software Engineering

Assignment 2

Assessment Type ¹: Design Implementation

Indicative Time on Task ²: 20 hours

Due: **W12 Sunday 11:55pm Submission, W13 class presentation**

Weighting: **15%**

Assignment 2 will be a assignment that will allow you to demonstrate the development of your understanding and your ability to apply the things that you have learned in the second part (weeks 7 to 12) of the unit.

On successful completion you will be able to:

- Use good software engineering practices to design code including unit testing and quality documentation
- Effectively use modern software development techniques and tools
- Discuss the life cycle of software systems development and the impact of implementation issues on various phases of the life cycle
- Explain the principles, practices, ethics and responsibilities of Software Engineering

Final Exam

Assessment Type ¹: Examination

Indicative Time on Task ²: 24 hours

Due: **Exam Period**

Weighting: **50%**

The final examination will be held during the usual University examination period and can cover all topics.

On successful completion you will be able to:

- Use good software engineering practices to design code including unit testing and quality documentation
 - Effectively use modern software development techniques and tools
 - Demonstrate knowledge of requirements elicitation techniques and the ability to apply those techniques to a range of problem domains
 - Discuss the life cycle of software systems development and the impact of implementation issues on various phases of the life cycle
 - Explain the principles, practices, ethics and responsibilities of Software Engineering
-

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

In week 1 there will be a lecture. Workshops start in week 2.

Textbook

This unit will be aligned with content from Software Engineering (Tenth Edition) by Ian Sommerville, Pearson, 2015, and Software Engineering: A practitioner's approach (Ninth Edition) by Roger S. Pressman and Bruce Maxim, McGraw Hill, 2020.

Lectures and related weekly practical questions

Lectures are used to motivate engagement with and reinforcement of the unit's subject matter. Lectures will of course include a significant amount of learning material, but even more importantly they include contextual material and learning activities that "make meaning" of the subject matter.

Workshops (Starting in week 2)

Workshops are smaller group classes which give you the opportunity to interact with your peers and with a teacher who has sound knowledge of the subject. Workshops will require working in small groups and sometimes involve reporting back to the class. The classes will focus on

reinforcing understanding of the concepts and their practical applications to problems. It is important that you participate in the activities and make some notes from them to assist you with revision of the material. Contribution to the weekly workshop and completion of the weekly task is a hurdle requirement, to ensure continued engagement throughout the unit.

Assignments

Assignments will play a key role in providing evaluation so that students and the teachers can gauge levels of understanding. Assignments will be related to the lecture material, workshop activities and weekly practical material and require students to bring together what they have been learning, and to think creatively and rigorously. The assignments are group assignment combined with individual reflection and individual vivas. The viva for assignment 1 is a hurdle tasks, to ensure engagement of all student in the group project.

Exam

A written exam (held within the university examination period) is designed to test your understanding of the course content and your application of the concepts to a number of scenarios or problem statements.

Methods of Communication

We will communicate with you via your university email and through announcements on iLearn. 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

Carl	Introduction to Software Engineering	No workshops in week 1
Carl	Complex Systems, Requirements Definition and Requirements Specification	Requirements Team selection
Carl	Modelling Domains and Modelling Systems	Design Team Selection
Carl	Communicating requirements for system structure, behaviour, data, and usage (including UML)	
James	Software Development Methods, especially Agile	
James	Software Project management, especially Version Control	
James	DevOps and Software Configuration Management	Assessment 1 (presentation and report due)

James	Software Testing and Quality	Assessment 1 (viva)
James	Software V&V	
Carl	Software Maintenance and Software Evolution	
Carl	Project Management	
Carl	Quality Management	Assessment 2 (report due)
Carl & James	Review	Assessment 2 (presentation)

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 ask.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 [AskMQ](#), 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

We value student feedback to be able to continually improve the way we offer our units. As such we encourage students to provide constructive feedback via student surveys, to the teaching staff directly, or via the FSE Student Experience & Feedback link in the iLearn page.

In response to feedback by students, and the advent of generative AI the assignments and weekly tasks have been changed. The assignments are group assignment with an individual component, instead of fully individual assignments. Written reports will be accompanied by presentations and vivas.

Workshops will also link more closely with the assignment tasks to allow for more face to face time with group activities relating to the assignments.

Unit information based on version 2024.03 of the [Handbook](#)