# COMP2050

## Software Engineering

Session 2, In person-scheduled-weekday, North Ryde 2026

*School of Computing*

## Contents

# General Information

Unit convenor and teaching staff
Unit Convenor
Dr. Xiao Cheng
xiao.cheng@mq.edu.au
By Appointment

Unit Lecturer
Dr. Lachlan Patrick
lachlan.patrick@mq.edu.au
By Appointment

Credit points
10

Prerequisites
60cp at 1000 level or above including COMP1010

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.

Learning in this unit enhances student understanding of global challenges identified by the United Nations Sustainable Development Goals (UNSDGs) Quality Education; Decent Work and Economic Growth; Industry, Innovation and Infrastructure;

## 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

## Requirements to pass the unit

**To pass this unit you must:**

1. Attempt all assessments, and
2. Achieve a total mark equal to or greater than 50

## Assessments Release Dates

- Assessment 1 – Problem Analysis : Specifications to be released no later than Monday, July 27, 2026.
- Assessment 2 – Design Proposal : Specifications to be released no later than Monday, September 21, 2026.
- Assessment 3 – Final Exam : Specifications to be released during Exam Period.

## Late Assessment Submission and Penalties

- 5% Penalty – If you submit your Assessment late, 5% of the Total Possible Marks would be deducted for each day (including weekends) up to 7 days.
    - Example 1 (Out of 100) – If you score 85 / 100 but submit 20 hours late, you would lose 5 Marks and receive 80 / 100.
    - Example 2 (Out of 30) – If you score 27 / 30 but submit 1 day late, you would lose 1.5 Marks and receive 25.5 / 30.
- After 7 Days – Submissions more than 7 days late would receive a Mark of 0.
- Extensions
    - 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.
    - Automatic Short Extension – Some Assessments are eligible for Automatic Short Extension. You can only apply for an Automatic Short Extension before the due date.
    - Special Consideration – If you need more time owing to serious issues and for any Assessments that are not eligible for Automatic Short Extension, you must apply for Special Consideration.

**Assessment details:**

- Problem Analysis: Late penalties apply to the entire submission.
- Design Proposal: Late penalties apply to the entire submission.
- Final Exam: The exam is an on-campus invigilated exam. In the event of ilness or misadventure, apply for special consideration and, if approved, arrangements for a suplementary exam will be made to run **during the supplementary exam period**.

## 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 https://connect.mq.edu.au/.

## Generative AI Usage

Assessments 1 (Problem Analysis) and 2 (Design Proposal) are AI Open while Assessment 3 (Final Exam) is AI Observed.

The university has taken a ‘two lane’ approach with GenAI: assessments can be AI Open (meaning students complete the assessment in their own time and may use GenAI) and Observed (students complete assessments in supervised environments, where use of AI is controlled or not permitted). More information on the use of GenAI in Assessment can be found here.

# Assessment Tasks

| Name | Weighting | Hurdle | Due | Groupwork/Individual | Short Extension | AI Approach |
|---|---|---|---|---|---|---|
| Final Exam | 40% | No | Exam Period | Individual | No | Observed |
| Problem Analysis | 30% | No | 13/09/2026 | Individual and Group | No | Open |

| Name | Weighting | Hurdle | Due | Groupwork/Individual | Short Extension | AI Approach |
|---|---|---|---|---|---|---|
| Desgin Proposal | 30% | No | 01/11/2026 | Individual and Group | No | Open |

# Final Exam

Assessment Type [1]: Examination
Indicative Time on Task [2]: 24 hours
Due: **Exam Period**
Weighting: **40%**
Groupwork/Individual: Individual
Short extension [3]: No
AI Approach: Observed

You will undertake a final examination during the formal examination period.

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

# Problem Analysis

Assessment Type [1]: Written Submission
Indicative Time on Task [2]: 20 hours
Due: **13/09/2026**
Weighting: **30%**
Groupwork/Individual: Individual and Group
Short extension [3]: No
AI Approach: Open

This group assignment uses requirements engineering and test analysis approaches, and will require students to interact with other students within their own group (as well as their client stakeholders) to understand, analyse, and document the needs of the system being proposed for development. Students will complete a portfolio of activities that will be submitted together.

This is a group project that will have individual marks.

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

## Desgin Proposal

Assessment Type [1]: Experiential task
Indicative Time on Task [2]: 30 hours
Due: **01/11/2026**
Weighting: **30%**
Groupwork/Individual: Individual and Group
Short extension [3]: No
AI Approach: Open

This group assignment uses design and implementation approaches, and will require students use a given set of requirements to come up with a design that meets those requirements. Students will complete a portfolio of activities that will be submitted together.

This is a group project that will have individual marks.

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

---

[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
- Academic Success 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.

[3] An automatic short extension is available for some assessments. Apply through the Service Connect Portal.

# Delivery and Resources

## When do classes start for COMP2050?

- **Lectures** are on Mondays from 11am to 1pm in 23WW P.G.Price Lecture Theatre starting from **Week 1**.
- **Practical classes start in week 2. However, there is some pre-work which should be done at home during week 1 in place of a week 1 practical class.**
- The scheduled practical classes will build on the concepts and curated expereince from the lecture in the week prior to the practical class. So, the week 2 practical class will build on the week 1 lecture; The week 3 practical class will build on the week 2 lecture; and so on...

## Methods of communication for COMP2050

COMP2050 staff will be communicating with you via the provided forums in iLearn, and your offical student email if needed. It is your responsibility to monitor offical communications from either of these modes.

You can communicate with staff using this same approach. There is a private forum in iLearn where you can communicate with lecture staff and where lecturer staff can communicate with you. If, however, you need to communicate via email, you are expected to send all emails to COMP2050 staff using your official student email account. Emails from other accounts may be filtered by the university infrastructure and not reach the inbox of the intended recipient.

## Unit Resources

All resources including lecture slides, practical class materials, weekly exercises, assessments, and unit readings will be provided through the COMP2050 iLearn page which is available at https://ilearn.mq.edu.au upon enrolment into the unit.

# Unit Schedule

| Week | Lecture concepts | Practical class and assessment |
|---|---|---|
| 1 | Introduction to COMP2050, large-scale software development, complex computing problems, and software development life cycles. | No practical class. Complete the preparation task before Week 2. |
| 2 | Project management, teamwork, communication, meetings, version control, and SDLC selection. | Review the Week 1 task and identify potential group members. |
| 3 | Requirements elicitation, analysis, specification, validation, and management. | Form project teams and set up project infrastructure. Teams normally include four students from the same practical class. |
| 4 | System modelling, use cases, user stories, data flows, requirements validation, and testable criteria. | First client interview, document development, retrospective, and workload planning. |

| Week | Lecture concepts | Practical class and assessment |
|---|---|---|
| 5 | Non-functional requirements, usability, standards, compliance, and security. | Second client interview, requirements refinement, retrospective, and workload planning. |
| 6 | Requirements traceability and consistency checking. | Final client interview, integration of findings, retrospective, and workload planning. |
| 7 | Moving from analysis to design, software architecture, design options, data flows, and definition of done. | Finalise the analysis document. The System Analysis Project is due Sunday at 11:55 pm through iLearn. |
| 8 | Software quality, test planning, testing strategies, test data, equivalence classes, boundary values, regression testing, and usability testing. | Hand over the analysis document to a paired group and organise design proposal tasks. A two-week teaching break follows. |
| 9 | Human-computer interaction, UI/UX design, prototyping, storyboarding, and usability testing. | Work with test data, equivalence classes, testing scenarios, and workload planning. |
| 10 | Software security, confidentiality, integrity, availability, maintenance, versioning, CI/CD, and automation. | Evaluate user interfaces using measurable criteria and plan maintenance and deployment activities. |
| 11 | Software evolution, change management, re-engineering, release planning, and system end-of-life. | Develop maintenance, deployment, and release-management strategies. |
| 12 | Ethics, professional responsibility, professional judgement, and the ACS and Engineers Australia codes of ethics. | The System Design Project is due Sunday at 11:55 pm through iLearn. |
| 13 | Unit review, exam preparation, future technologies, and software engineering career pathways. | Final exam held during the university examination period. It is handwritten, closed-book, and does not allow notes, dictionaries, or calculators. |

# Policies and Procedures

Macquarie University policies and procedures are accessible from Policy Central (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). 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.e

du.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/

## Academic Success

Academic Success provides resources to develop your English language proficiency, academic writing, and communication skills.

- Ask a Study Coach
- 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 https://students.mq.edu.au/support/technology/service-desk.

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.

---

Unit information based on version 2026.02 of the Handbook