# COMP2050

## Software Engineering

Session 2, In person-scheduled-weekday, North Ryde 2022

*School of Computing*

## Contents


| | |
|---|---|
| General Information | 2 |
| Learning Outcomes | 3 |
| General Assessment Information | 3 |
| Assessment Tasks | 3 |
| Delivery and Resources | 7 |
| Unit Schedule | 8 |
| Policies and Procedures | 10 |
| Changes from Previous Offering | 12 |



**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

Unit Convenor
Ansgar Fehnker
ansgar.fehnker@mq.edu.au

Lecturer
James Zheng
james.zheng@mq.edu.au

Tutor
Bharosha Poudel
bharosha.poudel@mq.edu.au

Tutor
Yuzhe Tian
yuzhe.tian@mq.edu.au

Tutor
Yao Deng
yao.deng@mq.edu.au

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 a 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.

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 – NO, unless Special Consideration is granted
- Class participation - NO, unless Special Consideration is granted

# Assessment Tasks

| Name | Weighting | Hurdle | Due |
|---|---|---|---|
| Weekly problems | 10% | Yes | Weekly |

| Name | Weighting | Hurdle | Due |
|---|---|---|---|
| Assignment 1 | 15% | Yes | week 7 |
| Assignment 2 | 15% | Yes | Week 12 |
| Class Participation | 10% | No | Weekly |
| Final Exam | 50% | No | Examination Period |

## Weekly problems

Assessment Type [1]: Participatory task
Indicative Time on Task [2]: 12 hours
Due: **Weekly**
Weighting: **10%**
**This is a hurdle assessment task (see assessment policy for more information on hurdle assessment tasks)**

Each week from week 1 to week 12, some questions/problems will be posted in iLearn, usually related to the lecture content for that week.

By the end of each week you are required to submit your responses in the manner specified in the questions for that week. Your answers will need to demonstrate appropriate depth of understanding.

Students are expected to complete all 12 submissions satisfactorily, but must complete at least 8 of 12 to meet the hurdle requirement.

On successful completion you will be able to:

- Use good software engineering practices to design code including unit testing and quality documentation
- 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: **week 7**
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: **Week 12**
Weighting: **15%**
**This is a hurdle assessment task (see assessment policy for more information on hurdle assessment tasks)**

Assignment 2 will be a written 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

## Class Participation

Assessment Type [1]: Participatory task
Indicative Time on Task [2]: 24 hours
Due: **Weekly**
Weighting: **10%**

**You are expected to actively participate in the weekly workshop.** Your active **participation will be marked**. A roll will be taken to record your attendance (online or in person). You must attend at your assigned weekly workshop time.

Workshops will involve a range of activities, some individual, some in pairs, some in groups. Be

prepared to present your ideas.

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

## Final Exam

Assessment Type [1]: Examination
Indicative Time on Task [2]: 24 hours
Due: **Examination 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

## Teaching and Learning Strategies

in 2021, COMP2050 will have 2-hour lectures delivered on-line each teaching week. Workshops (starting from week 2) have offerings in various locations (generally on the ground or first floor of 4 Research Park Drive). There are also some workshops which are online-only and intended to be available for those who are overseas or not in a position to travel to campus during this COVID-19 impacted period. Workshops are scheduled in 2-hour blocks and cover a wide variety of activities, exercises, and case studies which will invovled some individual work and some group collaboration.

Each week, you should:

- Attend lectures, take notes, ask questions
- Complete the weekly questions and submit them via ilearn
- Attend your workshop, engage in the activities and discussions, and seek feedback from your tutor on your work
- Read assigned reading material, add to your notes and prepare questions for your lecturer or tutor

You are encouraged to start working on any assignments immediately after they have been released - as they require careful thought and insight which is better done over longer periods of time rather than at the last minute.

## Feedback

The feedback that you receive plays a crucial role in your learning. You have many opportunities to seek and receive feedback. During lectures, you are encouraged to ask the lecturer questions to clarify anything you might not be sure of. To ensure you don't miss feedback make sure that you 1) review your marked assignments, 2) read unit emails and information at the unit website and 3) attend lectures which are often used to point out what was expected in assessable work and to provide examples of good solutions and/or examples of common errors.

## Unit Delivery

### Textbook

Aspects of COMP2050 will follow from Software Engineering:  A practitioner's approach (Nineth Edition) by Roger S. Pressman and Bruce Maxim, McGraw Hill, 2020. We will also be posting related chapters for another text book - Software Engineering (Tenth Edition) by Ian Sommerville, Pearson, 2015 if you have a copy of this textbook. You may also consider using earlier editions, while not quite as up-to-date, are frequently cheaper (chapter numbers and contents may vary from edition to edition).  Additional unit content will be presented in lectures and workshop classes. There may also be some other assigned readings, and there are many books that can be consulted for reference material.

## Lectures and related weekly tutorial questions

Lectures are used to motivate engagement with and reinforcement of the unit's subject matter. While most lectures are planned to be recorded, some activities such as role plays and dramas conducted in the lecture will not be captured properly in the recording. Volunteers to participate in role plays may be requested in the lecture, please consider whether you can contribute in this way. 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 tutor 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.

Note carefully that lecture material and workshop material have only some overlap, and each learning experience is designed to help you to learn different things.  You need to pay careful attention to, and take notes from, both kinds of classes throughout the semester.

## Assignments

Assignments will play a key role in providing formative evaluation so that students and the teachers can gauge levels of understanding. Assignments will be related to the lecture material, workshop activities and weekly tutorial material and require students to bring together what they have been learning, and to think creatively and rigorously.

## Exam

A written exam (held withing 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.

# Unit Schedule

| Week | Block Theme | Lecturer | Topic | Related Chapters | Weekly Submissions | Workshops | Assignment Notes |
|---|---|---|---|---|---|---|---|
| 1 | Software and Systems Analysis | Ansgar | Complexity, Software Architectures, and Distributed Systems | Sommerville: Ch 19, 17 Pressman: Ch 1, 10 | Due by Friday 5pm | No workshops in week 1 | |
| 2 | | Ansgar | Complex Systems, Requirements Definition and Requirements Specification | Sommerville: Ch 4, 12 Pressman: Ch 7 | Due by Friday 5pm | The worst webpage. | |

| Week | Block Theme | Lecturer | Topic | Related Chapters | Weekly Submissions | Workshops | Assignment Notes |
|---|---|---|---|---|---|---|---|
| 3 | | Ansgar | Modelling Domains and Modelling Systems | Sommerville: Ch 20 Pressman: Ch 8 | Due by Friday 5pm | | |
| 4 | | Ansgar | Communicating requirements for system structure, behaviour, data, and usage (including UML) | Sommerville: Ch 5 Pressman: Ch 8 | Due by Friday 5pm | | |
| 5 | Software and System Design and Implementation | Ansgar | Software Development Methods, especially Agile | Sommerville: Ch 2, 3 Pressman: Ch 2, 3 | Due by Friday 5pm | | |
| 6 | | James | Software Project management, especially Version Control | Sommerville: Ch 7, 23 Pressman: Ch 22 | Due by Friday 5pm | | |
| 7 | | James | DevOps and Software Configuration Management | Sommerville: Ch 5 Pressman: Ch 20, 22 | Due by Friday 5pm | | Assignment 1 Due |
| Mid Semester Teaching Break | | | | | No submission due this week | | |
| 8 | Software and System Testing, Verification and Validation | James | Software Testing and Quality | Sommerville: Ch 8, 24 Pressman: Ch 15, 19 | Due by Friday 5pm | | |
| 9 | | James | Software Testing and Quality | Sommerville: Ch 8, 24 Pressman: Ch 20, 23 | Due by Friday 5pm | | |
| 10 | | James | Software Verification and Validation | Sommerville: Ch 20, 24 Pressman: Ch 12, 23 | Due by Friday 5pm | | |
| 11 | Software and System Evolution Professional Responsibility | Ansgar | Software Maintenance and Software Evolution | Sommerville: Ch 9 Pressman: Ch 27 | Due by Friday 5pm | | |
| 12 | | Ansgar | Ethics and professional responsibilities in Software Engineering | Sommerville: Ch 1, 10 Pressman: Ch 30 | Due by Friday 5pm | | Assignment 2 Due |

| Week | Block Theme | Lecturer | Topic | Related Chapters | Weekly Submissions | Workshops | Assignment Notes |
|---|---|---|---|---|---|---|---|
| 13 | Consolidation | Ansgar& James | Software Engineering roundup and COMP2050 revision | | No submission due this week | | |
| Exam Period | Week 1 | | | | | | |
| | Week 2 | | | | | | |
| | Week 3 | | | | | | |

# 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.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 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

This years unit will be offered on campus.