



COMP2050

Software Engineering

Session 2, Special circumstance 2020

Department of Computing

Contents

General Information	2
Learning Outcomes	2
General Assessment Information	3
Assessment Tasks	4
Delivery and Resources	7
Unit Schedule	9
Policies and Procedures	9
Changes from Previous Offering	11
Grading	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.

Notice

As part of [Phase 3 of our return to campus plan](#), most units will now run tutorials, seminars and other small group learning activities on campus for the second half-year, while keeping an online version available for those students unable to return or those who choose to continue their studies online.

To check the availability of face-to-face and online activities for your unit, please go to [timetable viewer](#). To check detailed information on unit assessments visit your unit's iLearn space or consult your unit convenor.

General Information

Unit convenor and teaching staff

Lecturer and Convenor

Michael Johnson

michael.johnson@mq.edu.au

Contact via michael.johnson@mq.edu.au

After each live lecture

Lecturer

James Zheng

james.zheng@mq.edu.au

Contact via james.zheng@mq.edu.au

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

A major part of all software engineering projects is being able to report accurately and on time. For all your assignments, and for your professional life in the future, you are encouraged to

- set your personal deadline earlier than the official deadline
- keep backups of all your important files
- make sure that no-one else has access to your files or documents

Late work will not be accepted. Develop good working habits and manage your time well. If your contributions are seriously affected by illness or misadventure do your utmost to contact the convenor **before** the due date. If the severity of the disruption warrants it, you should consider applying for special consideration.

If you receive special consideration for the final exam, a supplementary exam will be scheduled for a time in December 2020. By making a special consideration application for the final exam you are declaring yourself available to sit during the supplementary examination period and you will not be eligible for a second special consideration approval based on pre-existing commitments. Please ensure you are familiar with the policy prior to submitting an application. Approved applicants will receive an individual notification approximately one week prior to the exam with the exact date and time of the supplementary examination.

Finally a remark about **hurdle** assessments:

Students are required to perform satisfactorily to meet the hurdle requirement. If a student makes a serious attempt, but fails to meet the hurdle requirement for a particular item of assessment, then he or she will be offered one further opportunity to meet the hurdle requirement. In this unit we expect students to reach a pass level (50%) in these assessment items, but, recognising that they are designed as an aid to learning (they are formative), the hurdle requirement will be merely 40%. Students obtaining between 30% and 40% on an assessment item will be given a further assessment task to allow them another opportunity to demonstrate the performance required to meet the hurdle.

In other words, in this unit we define 'satisfactory' as at least 40%, and 'serious attempt' as at least 30%.

Assessment Tasks

Name	Weighting	Hurdle	Due
Final Exam	50%	No	Formal University Examination Period
Class Participation	10%	No	Weekly, starting in week 2
Assignment 2	15%	Yes	Thursday, 22nd October, 5pm
Weekly problems	10%	Yes	Weekly, starting in week 1
Assignment 1	15%	Yes	Thursday, 17th September, 5pm

Final Exam

Assessment Type ¹: Examination

Indicative Time on Task ²: 25 hours

Due: **Formal University 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

Class Participation

Assessment Type ¹: Participatory task

Indicative Time on Task ²: 24 hours

Due: **Weekly, starting in week 2**

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

Assignment 2

Assessment Type ¹: Design Implementation

Indicative Time on Task ²: 25 hours

Due: **Thursday, 22nd October, 5pm**

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.

No late submission.

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

Weekly problems

Assessment Type ¹: Participatory task

Indicative Time on Task ²: 12 hours

Due: **Weekly, starting in week 1**

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.

No late submissions.

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: 25 hours

Due: **Thursday, 17th September, 5pm**

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.

No late submission.

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

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

Learning Locations

COMP2050 is taught in 2020 via on-line lectures and synchronous, including some face-to-face, Workshops (mixed classes) which are two hours long and take place in various locations generally on level 1 of 9WW (formerly E6A) at a time assigned to each student individually.

Textbook

The textbook for COMP2050 is *Software Engineering: A practitioner's approach* by Roger S. Pressman and Bruce Maxim, McGraw Hill, 2020. You may also consider using earlier editions of this text which, while not quite as up-to-date, are frequently cheaper. Additional unit content will be presented in lectures and practical classes. There will also be some other assigned readings, and there are many books that can be consulted for reference material.

Lectures

Digital recordings of parts of lectures are expected to be available. Note that some of the planned lecture activities are interactive, and it is unlikely that you would be able to understand what was happening if you were not present for the original lecture. In addition, some confidential case-studies may be presented in lectures but will not be recorded. So lectures will overall be presented as a combination of pre-recorded lecture material and interactive discussions of that material each week. You need to keep yourself available to interact at the appointed lecture times which are on Tuesdays at 1pm for one hour and on Fridays from 11am to 1pm. More information will be provided in the first lecture, at 1pm on Tuesday July 28.

Website

Extra course materials, some online discussions, and the submission of assessable work, will be carried out via iLearn.

The unit will make use of discussion boards hosted within iLearn. Feel free to post questions there -- the discussion boards will be monitored by the staff teaching the unit. Everyone is encouraged to engage with the discussion and provide questions and/or answers. Such discussions will generally be about the material treated in comp2050, but other aspects of Software Engineering may be discussed too. We are a "community of interest" and we are here

to learn about software engineering (including the software engineering taught in comp2050, but not only the software engineering taught in comp2050!).

If you have a personal question it is usually more appropriate to email the lecturer teaching during that week or the unit convenor.

Teaching and Learning Strategies

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.

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 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 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, practical and tutorial material and require students to bring together what they have been learning, and to think creatively and rigorously.

Exams will be designed to test your understanding of the course content and your application of the concepts to a number of scenarios or problem statements.

Each week you should:

- **Attend lectures**, take notes, ask questions
- **Complete the weekly questions** and submit them via ilearn
- **Attend your workshop** 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

- **Start working on any assignments** immediately after they have been released.

Unit Schedule

The topic titles are indicative, and may be subject to change.

Week	Topic	Lecturer
1	Software Architecture and Distributed Systems	Mike
2	Requirements Definition and Requirements Specification	Mike
3	Modelling Domains and Modelling Systems	Mike
4	Unified Modelling Language (UML)	Mike
5	Software Development Methods, especially Agile	James
6	Version Control, DevOps and Software Refactoring	James
7	Software Testing and Quality	James
	Semester break	
8	Embedded Systems	James
9	Real Time Applications	James
10	Design, Implementation and Deployment	Mike
11	Software Maintenance	Mike
12	Software Security Engineering	Mike
13	More Security Engineering	Mike

Policies and Procedures

Macquarie University policies and procedures are accessible from [Policy Central \(https://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central\)](https://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central). 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](#)
- [Grade Appeal Policy](#)
- [Complaint Management Procedure for Students and Members of the Public](#)
- [Special Consideration Policy](#) (**Note:** *The Special Consideration Policy is effective from 4 December 2017 and replaces the Disruption to Studies Policy.*)

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

If you would like to see all the policies relevant to Learning and Teaching visit [Policy Central](http://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central) (<http://staff.mq.edu.au/work/strategy-planning-and-governance/university-policies-and-procedures/policy-central>).

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: <https://students.mq.edu.au/study/getting-started/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

Student Support

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

Learning Skills

Learning Skills (mq.edu.au/learningskills) provides academic writing resources and study strategies to help you improve your marks and take control of your study.

- [Getting help with your assignment](#)
- [Workshops](#)
- [StudyWise](#)
- [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

Students with a disability are encouraged to contact the [Disability Service](#) who can provide appropriate help with any issues that arise during their studies.

Student Enquiries

For all student enquiries, visit Student Connect at ask.mq.edu.au

If you are a Global MBA student contact globalmba.support@mq.edu.au

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

The unit comp2050 changes every year as the software engineering programs at Macquarie are further developed and as we take account of student feedback on previous offerings. While the intended outcomes remain substantially the same as offerings of COMP255, students should not assume that any particular aspects of comp2050 this year will be in any particular ways similar to those aspects of previous years' offerings of COMP255.

Changes for this year (Semester 2, 2020):

- the experimental assignment 1 tried in 2019 will not be repeated
- assignment 1 and assignment 2 are co-dependent -- students will need to participate in assignment 1 in order to complete assignment 2
- there will be lecture material emphasising the purpose and importance of the unit for students' likely future careers
- hurdles have been maintained because of their demonstrated effectiveness in aiding student learning, but adjusted in the light of COVID-19.

Grading

At the end of the session, you will receive a mark that reflects your achievement in the unit.

The final mark for the unit will be calculated by combining the marks for all assessment tasks according to the percentage weightings shown in the assessment summary.

No late submissions for the weekly problems and the assignments.