



COMP4050

Software Engineering Practices

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>	8
<u>Changes from Previous Offering</u>	10

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 and lecturer

Ansgar Fehnker

ansgar.fehnker@mq.edu.au

Lecturer

Kate Stefanov

kate.stefanov@mq.edu.au

Credit points

10

Prerequisites

200cp at 1000 level or above including (COMP2050 or COMP255) and (COMP3010 or COMP333)

Corequisites

Co-badged status

Unit description

This unit provides an opportunity for students to practice and demonstrate their software engineering skills within the context of a team. Students will work together to understand, design, modify, test and deliver non-trivial software using practices that are in use in the information technology industry.

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: perform the stages of a modern software development process to achieve non-trivial outcomes

ULO2: apply principles of software project management, particularly relating to teamwork, roles and responsibilities

ULO3: use modern software development tools such as version control systems and issue trackers

ULO4: communicate progress and results of the software development process

General Assessment Information

COMP4050 applies an agile method to the engineering of features in a non-trivial software project. Students work as a team on a project. They are able and encouraged to use technology with which they are already familiar or that has a relatively short learning curve. The assessment is focused on the software processes and tools that a team uses, as well as individual contributions to the team effort. The items will be assessed continuously with feedback being provided frequently by the unit convenor to teams as well as to each student. Peer feedback will be used to inform the assessment by the convenor.

The deliverables for the basis of the reflection include group presentation, peer group reviews, personal logs, and a personal reflection. Based on the supplied evidence and the convenor's observations, these assessment items will be assessed according to the following standards:

- Distinction/High Distinction: An extremely valuable team member who makes many key contributions to many different aspects of the software that is developed by the team.
- Credit: A team member who makes useful contributions to multiple aspects of the team's software.
- Pass: A team member who makes a small number of contributions to the team's software or focuses on a single area of contribution.

The team contribution assessment item captures contributions to the functioning of the team rather than a specific contribution to the development of the software artefact (which is captured by the other three assessment items). E.g., a high team contribution mark might be earned by a student who often comes up with useful suggestions in planning discussions for features proposed by other team members or who volunteers to present some useful technology to the team so everyone can learn about it. Students are encouraged to think of ways in which they would like to contribute but the unit convenor will also suggest opportunities. Students should plan to attend all classes since they are vital software team meetings.

Participation will be assessed according to the following standards:

- Distinction/High Distinction: An extremely valuable team member who contributes strongly in many, varied ways to the smooth and efficient functioning of the team.
- Credit: A team member who makes multiple varied contributions to the team's operations, not just in a single particular way.
- Pass: A team member who makes a small number of contributions to the way the team operates or whose contributions have a single focus.

The assessment during the semester will be formative. The student and teams are expected to document in their final submissions what measures they have taken in response to formative feedback.

Requirements to pass the unit

- Achieve a total mark equal to or greater than 50%.

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 the final reflection is not submitted, up until the 7th day (including weekends). After the 7th day, a mark of '0' will be awarded even if the assessment is submitted. 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.

Submission time is set at Sundays, **11:55 pm**. A 1-hour grace period is provided to students who experience technical problems.

Special Consideration

The [Special Consideration Policy](#) aims to support students impacted by short-term circumstances or events that are serious, unavoidable, and significantly disruptive, 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 ask.mq.edu.au.

Assessment Tasks

Name	Weighting	Hurdle	Due
Team contribution	25%	No	Throughout
Development	25%	No	Throughout
Planning and Estimation	25%	No	Throughout
Review	25%	No	Throughout

Team contribution

Assessment Type ¹: Project

Indicative Time on Task ²: 10 hours

Due: **Throughout**

Weighting: **25%**

The participation assessment item captures overall contribution to the functioning of the team

rather than specific contributions to the software artefact (which is captured by the other three assessment items).

On successful completion you will be able to:

- perform the stages of a modern software development process to achieve non-trivial outcomes
- apply principles of software project management, particularly relating to teamwork, roles and responsibilities
- use modern software development tools such as version control systems and issue trackers
- communicate progress and results of the software development process

Development

Assessment Type ¹: Project

Indicative Time on Task ²: 25 hours

Due: **Throughout**

Weighting: **25%**

This assessment measures student contribution to the team's software development activities, including development of tests for a proposed feature, coding the feature, debugging, ensuring that tests pass and submission to the team's version control repository.

On successful completion you will be able to:

- perform the stages of a modern software development process to achieve non-trivial outcomes
- apply principles of software project management, particularly relating to teamwork, roles and responsibilities
- use modern software development tools such as version control systems and issue trackers
- communicate progress and results of the software development process

Planning and Estimation

Assessment Type ¹: Project

Indicative Time on Task ²: 25 hours

Due: **Throughout**

Weighting: **25%**

This assessment item measures student contributions to team planning and estimation activities. Included are activities such as soliciting and developing client input, proposing new features or fixes, estimating how much functionality can be achieved in a development period, including taking into account review of experiences in previous periods.

On successful completion you will be able to:

- perform the stages of a modern software development process to achieve non-trivial outcomes
- apply principles of software project management, particularly relating to teamwork, roles and responsibilities
- use modern software development tools such as version control systems and issue trackers
- communicate progress and results of the software development process

Review

Assessment Type ¹: Project

Indicative Time on Task ²: 25 hours

Due: **Throughout**

Weighting: **25%**

This assessment item measures student contribution to team review activities. Included are activities such as code and test review of features developed by other team members, merging features into the baseline version of the system, and reflection on achievements in a development period.

On successful completion you will be able to:

- perform the stages of a modern software development process to achieve non-trivial outcomes
- apply principles of software project management, particularly relating to teamwork, roles and responsibilities
- use modern software development tools such as version control systems and issue trackers
- communicate progress and results of the software development process

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

The workshops will start in week 1.

Weekly Activities

- Workshop including lecture (3 hours)

Workshops

The focus of the unit is on development periods of a few weeks duration - so-called sprints - where milestones such as new features, bug fixes and the like are proposed, planned, delivered and reviewed. Students should not develop code by themselves or in pairs or focus narrowly on one or two aspects. Instead, we expect students to participate in each aspect of the project rather than focus on one or two aspects in isolation. In particular, students will be asked to review each other's code and participate in reviews of each software milestone.

Tools such as distributed version control and milestone planning software will be used throughout to coordinate activities.

Each week there is a three-hour class during which the teams will meet to discuss expectations, plans and progress. Usually, the first hour or so will be devoted to general topics of interest to the whole class, while the second and third hours will be used by teams to work together on their activities. Students are expected to attend all classes since they are vital meetings of the software team.

Methods of Communication

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

COVID Information

For the latest information on the University's response to COVID-19, please refer to the Coronavirus infection page on the Macquarie website: <https://www.mq.edu.au/about/coronavirus-faqs>. Remember to check this page regularly in case the information and requirements change during semester. If there are any changes to this unit in relation to COVID, these will be

communicated via iLearn.

Unit Schedule

	Week	Topic	Activity	Deliverable
Sprint 0	1	Welcome	Team formation	
	2	Agile refresher	Meet the peers	Proposal
	3		Team Presentation	Slides, log
Sprint 1	4	GitHub for project management		
	5	CI/CD and DevOps		
	6	Working in teams		
	7		Sprint review and retrospective	Log, peer feedback
Break				
Sprint 2	8		Mid-term presentation	Slides
	9	Delivering effective technical presentations		
	10		Sprint review and retrospective	Log, peer feedback
Sprint 3	11	Protecting your IP		
	12	Advanced software engineering		Repo
	13		Final presentation and demo	Slides, reflection, log

The schedule is tentative, and may be subject to changes if circumstances require.

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

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 we received from student after the previous offering, this unit will focus during the semester on formative feedback, allowing the team and individual students to demonstrate that they can continually improve processes and performance.

Unit information based on version 2024.02 of the [Handbook](#)