

COIT13229 *Applied Distributed Systems*

Term 1 - 2026

Profile information current as at 21/04/2026 07:50 pm

All details in this unit profile for COIT13229 have been officially approved by CQUniversity and represent a learning partnership between the University and you (our student). The information will not be changed unless absolutely necessary and any change will be clearly indicated by an approved correction included in the profile.

General Information

Overview

Modern computer systems commonly rely on distributed and cloud computing systems, which enable seamless access to networked resources such as processors, data stores, sensor networks, cyber-physical systems, and multimedia services. In this unit, you will gain foundation knowledge, and an understanding of the system models and their enabling technologies. You will learn the core concepts of distributed computing, such as interprocess communication, concurrency and synchronisation, transactions and concurrency control as well as distributed databases and files. This unit also exposes you to distributed system characteristics of scalability, heterogeneity, security, and failure handling. The areas of mobile and ubiquitous computing and the social impact arising from the ubiquity of distributed systems will also be explored. You will learn to develop secure and reliable distributed computing applications and web services that can perform concurrent operations across multiple computers.

Details

Career Level: *Undergraduate*

Unit Level: *Level 3*

Credit Points: 6

Student Contribution Band: 8

Fraction of Full-Time Student Load: 0.125

Pre-requisites or Co-requisites

Prerequisites: COIT12200 and COIT11237.

Important note: Students enrolled in a subsequent unit who failed their pre-requisite unit, should drop the subsequent unit before the census date or within 10 working days of Fail grade notification. Students who do not drop the unit in this timeframe cannot later drop the unit without academic and financial liability. See details in the [Assessment Policy and Procedure \(Higher Education Coursework\)](#).

Offerings For Term 1 - 2026

- Brisbane
- Cairns
- Melbourne
- Online
- Rockhampton
- Sydney

Attendance Requirements

All on-campus students are expected to attend scheduled classes – in some units, these classes are identified as a mandatory (pass/fail) component and attendance is compulsory. International students, on a student visa, must maintain a full time study load and meet both attendance and academic progress requirements in each study period (satisfactory attendance for International students is defined as maintaining at least an 80% attendance record).

Website

[This unit has a website, within the Moodle system, which is available two weeks before the start of term. It is important that you visit your Moodle site throughout the term. Please visit Moodle for more information.](#)

Class and Assessment Overview

Recommended Student Time Commitment

Each 6-credit Undergraduate unit at CQUniversity requires an overall time commitment of an average of 12.5 hours of study per week, making a total of 150 hours for the unit.

Class Timetable

Regional Campuses

Bundaberg, Cairns, Emerald, Gladstone, Mackay, Rockhampton, Townsville

Metropolitan Campuses

Adelaide, Brisbane, Melbourne, Perth, Sydney

Assessment Overview

1. Practical Assessment

Weighting: 35%

2. Written Assessment

Weighting: 20%

3. Reflective Practice Assignment

Weighting: 45%

Assessment Grading

This is a graded unit: your overall grade will be calculated from the marks or grades for each assessment task, based on the relative weightings shown in the table above. You must obtain an overall mark for the unit of at least 50%, or an overall grade of 'pass' in order to pass the unit. If any 'pass/fail' tasks are shown in the table above they must also be completed successfully ('pass' grade). You must also meet any minimum mark requirements specified for a particular assessment task, as detailed in the 'assessment task' section (note that in some instances, the minimum mark for a task may be greater than 50%). Consult the [University's Grades and Results Policy](#) for more details of interim results and final grades.

CQUniversity Policies

All University policies are available on the [CQUniversity Policy site](#).

You may wish to view these policies:

- Grades and Results Policy
- Assessment Policy and Procedure (Higher Education Coursework)
- Review of Grade Procedure
- Student Academic Integrity Policy and Procedure
- Monitoring Academic Progress (MAP) Policy and Procedure - Domestic Students
- Monitoring Academic Progress (MAP) Policy and Procedure - International Students
- Student Refund and Credit Balance Policy and Procedure
- Student Feedback - Compliments and Complaints Policy and Procedure
- Information and Communications Technology Acceptable Use Policy and Procedure

This list is not an exhaustive list of all University policies. The full list of University policies are available on the [CQUniversity Policy site](#).

Previous Student Feedback

Feedback, Recommendations and Responses

Every unit is reviewed for enhancement each year. At the most recent review, the following staff and student feedback items were identified and recommendations were made.

Feedback from Student Evaluations and Unit Coordinator Reflections

Feedback

The volume of content was excessive and the code in the slides is difficult to understand.

Recommendation

Some advanced topics may be removed: causal consistency may no longer be covered, and sequential consistency can be taught conceptually rather than at the code level.

Feedback from Student Evaluations

Feedback

The assessment specifications were unclear.

Recommendation

The final project may be broken down into stages and additional scaffolding provided to help students work through the project's intended ambiguity and conflicting requirements.

Unit Learning Outcomes

On successful completion of this unit, you will be able to:

1. Evaluate techniques for creating secure, reliable, and efficient distributed systems by examining the technology needs and social impact arising from ubiquitous distributed systems
2. Design and develop secure distributed applications in a distributed service-oriented architecture
3. Analyse the use of algorithms and mechanisms to execute coordinated concurrent processes in distributed systems
4. Implement software applications that enable concurrent access to distributed databases and transactions.

Australian Computer Society (ACS) recognises the Skills Framework for the Information Age (SFIA). SFIA is a widely used and consistent definition of ICT skills. SFIA is increasingly being used when developing job descriptions and role profiles. ACS members can use the tool MySFIA to build a skills profile at

<https://www.acs.org.au/professionalrecognition/mysfia-b2c.html>

This unit contributes to the following workplace skills as defined by SFIA 8. The SFIA code is included:

- Solution architecture (ARCH)
- Software Design (SWDN)
- Systems integration and build (SINT)
- Programming/Software Development (PROG),
- Database Design (DBDS)
- User Experience Evaluation (USEV)
- Testing (TEST)
- System Installation and Removal (HSIN)

Alignment of Learning Outcomes, Assessment and Graduate Attributes

— N/A Level  Introductory Level  Intermediate Level  Graduate Level  Professional Level  Advanced Level

Alignment of Assessment Tasks to Learning Outcomes

Assessment Tasks

Learning Outcomes

1 2 3 4

Assessment Tasks**Learning Outcomes**

	1	2	3	4
1 - Written Assessment - 20%	•			
2 - Practical Assessment - 35%		•		•
3 - Reflective Practice Assignment - 45%	•	•	•	•

Alignment of Graduate Attributes to Learning Outcomes**Graduate Attributes****Learning Outcomes**

	1	2	3	4
1 - Communication	•		•	•
2 - Problem Solving		•	•	
3 - Critical Thinking	•		•	
4 - Information Literacy	•			
5 - Team Work			•	•
6 - Information Technology Competence		•	•	•
7 - Cross Cultural Competence				•
8 - Ethical practice	•			•
9 - Social Innovation		•		•
10 - First Nations Knowledges				
11 - Aboriginal and Torres Strait Islander Cultures				

Textbooks and Resources

Textbooks

There are no required textbooks.

IT Resources

You will need access to the following IT resources:

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- Visual Studio Code (latest version)
- Python 3.10 (or higher)
- Python PyPI (pip) packages including zmq, flask, boto3, hdfs, black, pylint and requests
- Java JDK 25 or higher

Referencing Style

All submissions for this unit must use the referencing style: Harvard (author-date)

For further information, see the Assessment Tasks.

Teaching Contacts

Jamie Shield Unit Coordinator
j.shield@cqu.edu.au

Schedule

Week 1 - 09 Mar 2026	Chapter	Events and Submissions/Topic
Module/Topic 1 Introduction	Refer to the unit website for the readings. • van Steen & Tanenbaum, Chps 1 & 2 • Bobrov, Chp 2	
Week 2 - 16 Mar 2026	Chapter	Events and Submissions/Topic
Module/Topic 2 Threads	• van Steen & Tanenbaum, Chp 3 • Lanaro, Ngyuen & Sakis, Chp 10	
Week 3 - 23 Mar 2026	Chapter	Events and Submissions/Topic
Module/Topic 3 Communication	• van Steen & Tanenbaum, Chp 4 • Bitterman & Deb, Chp 4	W3 Practical (12%)
Week 4 - 30 Mar 2026	Chapter	Events and Submissions/Topic
Module/Topic 4 Linearisability	• van Steen & Tanenbaum, Chp 7 • Verma et al., Chp 10	W4 Practical (12%)
Week 5 - 06 Apr 2026	Chapter	Events and Submissions/Topic
Module/Topic 5 Sequential Consistency		W5 Practical (11%)
Week 6 - 13 Apr 2026	Chapter	Events and Submissions/Topic
Module/Topic 6 Peer-to-Peer Sequential Consistency	• van Steen & Tanenbaum, Chp 8 • Haloi, Chp 2 • Tankariya & Parmar, Chp 12	
Vacation Week - 20 Apr 2026	Chapter	Events and Submissions/Topic
Week 7 - 27 Apr 2026	Chapter	Events and Submissions/Topic
Module/Topic 7 Coordination	van Steen & Tanenbaum, Chp 5	
Week 8 - 04 May 2026	Chapter	Events and Submissions/Topic
Module/Topic 8 Naming	van Steen & Tanenbaum, Chp 6	
Week 9 - 11 May 2026	Chapter	Events and Submissions/Topic
Module/Topic 9 Security and Web	• van Steen & Tanenbaum, Chp 9 • Gaspar & Stouffer, Chp 6 • Lanaro, Chp 6 • Relan, Chp 1 • Peralta, Chp 4	Project Due: Week 9 Thursday (14 May 2026) 11:45 pm AEST
Week 10 - 18 May 2026	Chapter	Events and Submissions/Topic
Module/Topic 10 Review		
Week 11 - 25 May 2026	Chapter	Events and Submissions/Topic
Module/Topic No lecture		In-class assessment (45%)

Term Specific Information

Unit Coordinator: Jamie Shield, Cairns, j.shield@cqu.edu.au
Oncampus students must attend the Week 11 tutorial. Distance students must organise their own location and supervisor which will be vetted by the unit coordinator.

Assessment Tasks

1 Practicals

Assessment Type
Practical Assessment

Task Description

Your task for this assignment is to design, implement, test and document small client/server and peer-to-peer distributed systems. The purpose of this assignment is to assess your competency in distributed networking, multi-threading and consistency. You will evaluate techniques for creating secure, reliable and efficient distributed systems.

AI Collaboration: You may use AI to assist with specific tasks such as drafting text, refining and evaluating your work. You must critically evaluate and modify any AI-generated content you use.

Re-attempts of this assignment are at the discretion of the unit coordinator.

Assessment Due Date

Due in weeks 3, 4 and 5. This assessment is exempt from the 72-hour submission grace period and must be completed by the stated submission date/time.

Return Date to Students

Immediate feedback

Weighting
35%

Assessment Criteria

This assessment consists of small programming activities. Each question will be marked on aspects such as functionality, coding style, documentation of code and testing, error handling, variable naming, code reuse and referencing.

Referencing Style

- [Harvard \(author-date\)](#)

Submission

Online

Submission Instructions

Complete the practicals each week on the unit website.

Learning Outcomes Assessed

- Design and develop secure distributed applications in a distributed service-oriented architecture
- Implement software applications that enable concurrent access to distributed databases and transactions.

2 Project

Assessment Type
Written Assessment

Task Description

You will develop a distributed system including analysis of requirements, design, implementation and testing. You will apply the distributed systems technologies to manage non-Byzantine faults such as latency and network partitions. You will record a video that explains the most important aspects of your system. You will create requirements and design(s) which will assess your ability to design and develop secure distributed systems; your code and testing reports will assess your ability to implement concurrent software applications.

Group Work

You may work alone or in groups of 2 or 3 people for this assignment. All group members must be identified in the group work artefacts. All group members must submit via the unit website. The moderation process might allocate group members different marks. Sharing of artefacts, for example, code, between groups is not permitted.

Repository

Create a private code repository and invite your tutor and the unit coordinator. One code repository is to be used by all

group members.

AI

To improve learning, you are encouraged to manually draft your artefacts and then (responsibly) use AI to check and improve your work. To protect yourself, you must identify which parts of your submissions were created by which tools. AI Collaboration: You may use AI to assist with specific tasks such as drafting text, refining and evaluating your work. You must critically evaluate and modify any AI-generated content you use.

Code Reuse

In this unit, you are encouraged to search for and reuse good quality code and to use generative AI tools responsibly. All sources of code should be referenced. Deprecated code should not be used. Sharing of artefacts, for example, code, outside of your group is not permitted.

Re-attempts of this assignment are at the discretion of the unit coordinator.

Assessment Due Date

Week 9 Thursday (14 May 2026) 11:45 pm AEST

This assessment is not exempt from the 72-hour submission grace period.

Return Date to Students

Week 11 Friday (29 May 2026)

Weighting

20%

Assessment Criteria

You will be marked on aspects such as quality and correctness of requirements and design including diagrams ($\approx 33\%$), code and testing ($\approx 33\%$), individual presentation ($\approx 33\%$) and use of code repository tools.

Referencing Style

- [Harvard \(author-date\)](#)

Submission

Online

Submission Instructions

Submit a single Word document and code to both a private code repository and to the unit website. Submit links to your private repository and individual video to the unit website. All group members must submit.

Learning Outcomes Assessed

- Evaluate techniques for creating secure, reliable, and efficient distributed systems by examining the technology needs and social impact arising from ubiquitous distributed systems

3 Invigilated, no-AI

Assessment Type

Reflective Practice Assignment

Task Description

You are required to complete an invigilated, no-AI assessment during your Week 11 tutorial. The assessment may include tasks requiring you to analyse, design, code and evaluate distributed systems, together with reflective critique and comparative discussion of your project work.

Oncampus students: you must attend the Week 11 tutorial. Distance students: prior to Week 10, you must organise your own exam including the location and supervisor to be vetted by the unit coordinator.

No AI: You must not use AI at any point during the assessment. You must demonstrate your core skills and knowledge.

Re-attempts of this assignment are at the discretion of the unit coordinator.

Assessment Due Date

Oncampus: inclass Week 11 tutorial. Distance: in Week 11 by negotiation with unit coordinator. This assessment is exempt from the 72-hour submission grace period and must be completed by the stated submission date/time.

Return Date to Students

Feedback will be provided by the certification of grades.

Weighting

45%

Minimum mark or grade

You must obtain 15/45 to pass the unit.

Assessment Criteria

You will be marked on the quality and correctness of your responses, with marks balanced across weekly topics,

requirements analysis, designs, code and supporting diagrams.

Referencing Style

- [Harvard \(author-date\)](#)

Submission

Offline

Submission Instructions

Oncampus: attend your Week 11 tutorial. Distance: by negotiation with unit coordinator.

Learning Outcomes Assessed

- Evaluate techniques for creating secure, reliable, and efficient distributed systems by examining the technology needs and social impact arising from ubiquitous distributed systems
- Design and develop secure distributed applications in a distributed service-oriented architecture
- Analyse the use of algorithms and mechanisms to execute coordinated concurrent processes in distributed systems
- Implement software applications that enable concurrent access to distributed databases and transactions.

Academic Integrity Statement

As a CQUniversity student you are expected to act honestly in all aspects of your academic work.

Any assessable work undertaken or submitted for review or assessment must be your own work. Assessable work is any type of work you do to meet the assessment requirements in the unit, including draft work submitted for review and feedback and final work to be assessed.

When you use the ideas, words or data of others in your assessment, you must thoroughly and clearly acknowledge the source of this information by using the correct referencing style for your unit. Using others' work without proper acknowledgement may be considered a form of intellectual dishonesty.

Participating honestly, respectfully, responsibly, and fairly in your university study ensures the CQUniversity qualification you earn will be valued as a true indication of your individual academic achievement and will continue to receive the respect and recognition it deserves.

As a student, you are responsible for reading and following CQUniversity's policies, including the [Student Academic Integrity Policy and Procedure](#). This policy sets out CQUniversity's expectations of you to act with integrity, examples of academic integrity breaches to avoid, the processes used to address alleged breaches of academic integrity, and potential penalties.

What is a breach of academic integrity?

A breach of academic integrity includes but is not limited to plagiarism, self-plagiarism, collusion, cheating, contract cheating, and academic misconduct. The Student Academic Integrity Policy and Procedure defines what these terms mean and gives examples.

Why is academic integrity important?

A breach of academic integrity may result in one or more penalties, including suspension or even expulsion from the University. It can also have negative implications for student visas and future enrolment at CQUniversity or elsewhere. Students who engage in contract cheating also risk being blackmailed by contract cheating services.

Where can I get assistance?

For academic advice and guidance, the [Academic Learning Centre \(ALC\)](#) can support you in becoming confident in completing assessments with integrity and of high standard.

What can you do to act with integrity?



Be Honest

If your assessment task is done by someone else, it would be dishonest of you to claim it as your own



Seek Help

If you are not sure about how to cite or reference in essays, reports etc, then seek help from your lecturer, the library or the Academic Learning Centre (ALC)



Produce Original Work

Originality comes from your ability to read widely, think critically, and apply your gained knowledge to address a question or problem