# COIT12200 *Software Design & Development*

## Term 1 - 2023

Profile information current as at 23/07/2025 10:46 am

All details in this unit profile for COIT12200 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

In this unit you will learn both theoretical and practical aspects of software engineering. The theory will focus on software processes, requirements engineering, modelling, architectural design, object-oriented design, software development methodology, testing, and quality assurance. You will also be introduced to the principles of software re-use, software evolution and configuration management. You will develop multi-tiered software application having presentation, application and data persistence tiers. You will also learn test-driven software application development using appropriate tools, thus building solid foundations for software development.

### Details

Career Level: *Undergraduate*
Unit Level: *Level 2*
Credit Points: *6*
Student Contribution Band: *8*
Fraction of Full-Time Student Load: *0.125*

### Pre-requisites or Co-requisites

Pre-requisites: (COIT11134 and COIT11237) OR (COIT11134 and COIT12167).
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 - 2023

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

### 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 and Written Assessment**
Weighting: 25%
2. **Practical and Written Assessment**
Weighting: 35%
3. **Project (applied)**
Weighting: 40%

## 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 feedback

**Feedback**

Need a clearer provision about each student contribution to the final paired project - Assessment Item 3.

**Recommendation**

We will update the marking criteria related to teamwork part in the final project, which will set a clear provision or clause related to each team member's contribution to the project.

### Feedback from Student feedback

**Feedback**

The current textbook emphasises software development theories. Weekly Lecture slides have too many pages.

**Recommendation**

We are considering reducing the number of lecture slides. The adopted textbook is a classic book in the field of software development but indeed weekly lecture slides have too many pages that need to be reduced.

### Feedback from Self-reflection

**Feedback**

We should add some contents in Week 12 related to security engineering. In Assessment Item 3 (Project), we add the requirement of Authentication.

**Recommendation**

We are considering adding the Authentication requirement for Assessment Item 3 in the next offering. The study materials or tutorial materials in Week 12 will add SHA1/256 algorithm for encrypting the password that can be stored in a table in a MySQL database.

## Unit Learning Outcomes

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

1. Prepare software requirements specification following requirements engineering
2. Apply modelling techniques to document architectural and system models as per the requirements specification
3. Design and implement a multi-tiered software application consisting of presentation, application and data persistence tiers.
4. Conduct test-driven development, validation and verification testing
5. Discuss quality management in a range of different design and development scenarios
6. Work independently and collaboratively in small teams.

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 7. The SFIA code is included:
Requirements Definition and management (REQM)
User Experience Analysis (UNAN)
Software Design (SWDN)
System Integration and Build (SINT
Programming/Software Development (PROG)
Database Design (DBDS)
Testing (TEST)
Quality assurance(QUAS)
Quality management (QUMG).

## 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** | **5** | **6** |
| **1 - Practical and Written Assessment - 25%** | ● | ● | | | | |
| **2 - Practical and Written Assessment - 35%** | | | ● | | | |
| **3 - Project (applied) - 40%** | ● | ● | ● | ● | ● | ● |

### Alignment of Graduate Attributes to Learning Outcomes

| **Graduate Attributes** | **Learning Outcomes** | | | | | |
|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** |
| **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 - Aboriginal and Torres Strait Islander Cultures** | | | | | | |

### Alignment of Assessment Tasks to Graduate Attributes

| **Assessment Tasks** | **Graduate Attributes** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **1 - Practical and Written Assessment - 25%** | ▒ | ▒ | ▒ | ▒ | | ▒ | | | | |
| **2 - Practical and Written Assessment - 35%** | | ▒ | ▒ | | | ▒ | | | | |

| Assessment Tasks | Graduate Attributes | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 3 - Project (applied) - 40% | ● | ● | ● | ● | ● | ● | ● | ● | | |

## Textbooks and Resources

### Textbooks

COIT12200

**Prescribed**

**Software Engineering**
Global Edition (10e) 2015 (2015)
Authors: Ian Sommerville
Pearson
Harlow , Essex , United Kingdom
ISBN: 9781292096131
Binding: Paperback

**Additional Textbook Information**

BOTH paper and eBook versions can be purchased at the CQUni Bookshop here: http://bookshop.cqu.edu.au (search on the Unit code)

**View textbooks at the CQUniversity Bookshop**

### IT Resources

**You will need access to the following IT resources:**

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- Modelling software such as Microsoft Visio
- JDK 11 or a higher version, NetBeans IDE (Java EE version) 11 or a higher version, MySQL Database 5 or a higher version, GlassFish Application Server 5 or a higher version
- MySQL Community Server 8.06 or higher (https://dev.mysql.com/downloads/windows/installer/8.0.html)
- JUnit 5
- NetBeans 12.4 or higher version

## Referencing Style

All submissions for this unit must use the referencing style: Harvard (author-date)
For further information, see the Assessment Tasks.

## Teaching Contacts

**Gopi Akella** Unit Coordinator
g.akella@cqu.edu.au

## Schedule

**Week 1 - 06 Mar 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|

|  |  |  |
|---|---|---|
| Introduction<br>Software Processes | Sommerville, Chapter 1 & 2 | Tutorial exercises<br>Tutorial workshop<br>Introduction to MySQL Server 8.0 |

**Week 2 - 13 Mar 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Requirement Engineering | Sommerville, Chapter 4 | Tutorial exercises<br>Tutorial workshop<br>Installing MySQL driver for NetBeans |

**Week 3 - 20 Mar 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| System Modelling | Sommerville, Chapter 5 | Tutorial exercises<br>Tutorial workshop<br>Case Study: Patient(MVP) UML Class diagram |

**Week 4 - 27 Mar 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Architectural Design | Sommerville, Chapter 6<br>JDBC Overview | Tutorial exercises<br>Tutorial workshop<br>Case Study - MentCare System |

**Week 5 - 03 Apr 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Design & Implementation | Sommerville, Chapter 7<br>Introduction to java GUI building<br>GUI Building in the Netbeans IDE | Tutorial exercises<br>Tutorial workshop<br>Case Study - Patient Book<br>**Assignment-1 Due (Friday, 7 April 2023)**<br>**Assignment 1** Due: Week 5 Friday (7 Apr 2023) 11:45 pm AEST |

**Vacation Week - 10 Apr 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Mid-term break |  |  |

**Week 6 - 17 Apr 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Agile Software Development | Sommerville, Chapter 3 | Tutorial exercises<br>Tutorial Workshop<br>Case study: Address Book |

**Week 7 - 24 Apr 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Software Testing | Sommerville, Chapter 8 | Tutorial exercises<br>Tutorial Workshop<br>JUnit 5 |

**Week 8 - 01 May 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Software Evolution & Reuse | Sommerville, Chapter 9 & 15 | Tutorial exercises<br>Tutorial Workshop |

**Week 9 - 08 May 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|
| Quality Management | Sommerville, Chapter 24 | Tutorial exercises<br>Tutorial Workshop |

**Week 10 - 15 May 2023**

| Module/Topic | Chapter | Events and Submissions/Topic |
|---|---|---|

| | | |
|---|---|---|
| Configuration Management | Sommerville, Chapter 25 | Tutorial exercises<br>Tutorial Workshop<br>**Assignment-2 Due (Friday, 19 May 2023)**<br>**Assignment 2** Due: Week 10 Friday (19 May 2023) 11:45 pm AEST |
| **Week 11 - 22 May 2023** | | |
| **Module/Topic** | **Chapter** | **Events and Submissions/Topic** |
| Dependable Systems | Sommerville, Chapter 10 | Tutorial exercises<br>Tutorial Workshop |
| **Week 12 - 29 May 2023** | | |
| **Module/Topic** | **Chapter** | **Events and Submissions/Topic** |
| Security Engineering | Sommerville, Chapter 13 | Tutorial exercises<br>Tutorial Workshop |
| **Review/Exam Week - 05 Jun 2023** | | |
| **Module/Topic** | **Chapter** | **Events and Submissions/Topic** |
| Project Submission Due | | **Project** Due: Review/Exam Week Friday (9 June 2023) 11:45 pm AEST |
| **Exam Week - 12 Jun 2023** | | |
| **Module/Topic** | **Chapter** | **Events and Submissions/Topic** |

## Assessment Tasks

### 1 Assignment 1

**Assessment Type**
Practical and Written Assessment

**Task Description**
**Assignment 1**
Due date: Friday of Week 5 (7th April 2023) 11.45 pm AEST
Weighting: 25%
Length: NA
**Objectives**
The purpose of this assessment item is to assess your skills attributable to the following learning outcomes.

- Prepare software requirements specification following requirements engineering.
- Apply modelling techniques to document architectural and system models as per the requirements specification.

**Assessment Due Date**
Week 5 Friday (7 Apr 2023) 11:45 pm AEST

**Return Date to Students**
Week 6 Friday (21 Apr 2023)

**Weighting**
25%

**Assessment Criteria**
Your task in this assessment is to analyse the given case study, understand the user requirements, develop the necessary requirement specification, and design documentation for a software system. Case study will be available on Unit Moodle site.
**ASSESSMENT COMPLETION GUIDELINES**
You should submit a report containing the following sections as listed below.
1. **System Requirements**: Provide functional and non-functional system requirements.
2. **System Architecture**: Provide a package diagram for the MVP or MVC design showing all the classes involved.
3. **Use Case Diagrams**: Use UML to develop use case diagrams clearly showing the users (actors) and the main

functions of the system
4. **UML Class Diagrams**: Identify the classes based on the descriptions and the functions required. Show clearly public, private access and associations between classes.
5. **Sequence Diagrams**: Provide two sequence diagrams with annotations for update and search operations. Develop the sequence diagrams clearly showing the user interactions and the different layers of the system.
6. **Data Design**: Provide database table design with column names and types
7. **GUI design**: Develop the GUI design based on the use cases and the sequence diagrams. You can be creative and only an indicative GUI is required. Developers are free to use this design as is or to do things differently. Note: You can use the NetBeans Forms/or JavaFX to develop a GUI design.
8. **Requirements Matrix**
9.**Template File**: Use the 'Assignment1Template.dot' file available from the Unit website under the Assessment folder for creating your report document file. Download the template and save it in your directory. Locate the file in your directory using File explorer and double click on it. This will create a document file named 'document1' by default. You can save it assigning the name you prefer and continue writing the report.
10.**Report Submission**: You should submit only the word document using the Assignment 1 submission link available from the Unit website.

**Referencing Style**

- Harvard (author-date)

**Submission**
Online

**Learning Outcomes Assessed**

- Prepare software requirements specification following requirements engineering
- Apply modelling techniques to document architectural and system models as per the requirements specification

**Graduate Attributes**

- Communication
- Problem Solving
- Critical Thinking
- Information Literacy
- Information Technology Competence

## 2 Assignment 2

**Assessment Type**
Practical and Written Assessment

**Task Description**
**Assignment 2**
Due date: Friday of Week 10 (19th May 2023) 11.45 pm AEST
Weighting: 35%
Length: NA
**Objectives**
The purpose of this assessment item is to assess your skills attributable to the following learning outcomes.

- Design and implement a multi-tiered software application consisting of presentation, application and data persistence tiers.
- Conduct test-driven development, validation and verification testing.
- Discuss quality management in a range of different design and development scenarios.

**Assessment Due Date**
Week 10 Friday (19 May 2023) 11:45 pm AEST

**Return Date to Students**
Week 12 Friday (2 June 2023)

**Weighting**
35%

**Assessment Criteria**
Your tasks for this assessment.
• Provide the complete implementation and testing for the given case study project designed in the first assignment. The implementation should adhere to MVP (model view presenter) or MVC (model view controller) pattern with data

persisted to a relational database using MySQL Server.
• Write a paragraph on version control as discussed on Unit Moodle site.
**ASSESSMENT COMPLETION GUIDELINES**
1. **Implementation** Provide the entire NetBeans project containing the source code.
2. **Documentation** Provide a word document with all the test screens with annotation to demonstrate the working prototype. Also address the essay relating to version control.
3. **Test Plans** Provide test use cases for search operations
4. **Acceptance tests** This will be carried out by the marker to test if your application meets all the functional requirements.
5. **Report Submission** You should submit a zipped file of the entire project with source code and a word document addressing the assessment criteria.

### Referencing Style

- Harvard (author-date)

### Submission

Online

### Learning Outcomes Assessed

- Design and implement a multi-tiered software application consisting of presentation, application and data persistence tiers.

### Graduate Attributes

- Problem Solving
- Critical Thinking
- Information Technology Competence

## 3 Project

### Assessment Type

Project (applied)

### Task Description

**Assignment 3 - Project**
Due date: Friday of Review/Exam Week (9th June 2023) 11.45 pm AEST
Weighting: 40%
Length: NA
**Objectives**
The purpose of this assessment item is to assess your skills attributable to the following learning outcomes.

- Prepare software requirements specification following requirements engineering.
- Apply modelling techniques to document architectural and system models as per the requirements specification.
- Design and implement a multi-tiered software application consisting of presentation, application and data persistence tiers.
- Conduct test-driven development, validation and verification testing.
- Discuss quality management in a range of different design and development scenarios.
- Work independently and collaboratively in small teams.

### Assessment Due Date

Review/Exam Week Friday (9 June 2023) 11:45 pm AEST

### Return Date to Students

On the day of grades published

### Weighting

40%

### Assessment Criteria

**Assessment Task - Project**
In this project, your task is to design and implement a Java desktop application software as specified on Unit Moodle site with a team consisting of maximum **three** students. The assessment task includes design, implementation and testing the application software.
More details on the task requirements are described as follows.
(1) **Document Design and Test** Provide a comprehensive design & test document that addresses the following

requirements.
(i) **Requirements specification** Analyze the given case-study and application scenario, specify the functional and non— functional requirements.
(ii) **The architectural design** Provide the system architectural design.
(iii) **UML class diagram** (MVC or MVP pattern) Identify the classes and develop a UML class diagram with all necessary classes and interfaces to be implemented in your development.
(iv) **User interface design** Describe your design on user interface. Particularly explain button events and the functionalities.
(v) **Database schemas and design** Include an E-R diagram, tables & attributes, primary keys and foreign keys, labelling the relationships. Provide sample data in three tables in this document (these data are also required to populate into your database), and SQL scripts to create database tables and insert sample data.
(vi) Testing After completing your implementation, thoroughly test your programs to show the programs do what they should do as specified. Report any anomalies, errors or some functionalities that don't work properly. Discuss possible reasons for the functionalities that may not work. Provide a test plan with sample data used, expected results and actual results. Include evidence of testing by providing screen shots. You should include test data for correct and incorrect input.
(2) **Implementation** Provide the complete implementation and testing for the software system as required. The implementation should adhere to MVC or MVP design pattern and data persisted to a relational database using MySQL Server. Provide the entire NetBeans project (Zipped NetBeans Ant Project file) containing the source code.

**Submission**

This assessment includes both group and individual submissions. Read the instructions carefully.

**Teamwork Report (Individual)**

A student, who completes the whole project as a member of a productive team, should include your report answers addressing the following questions.

**Template A**

**1. Team Details**

a. Team Name (optional):
b. Your Name and Student ID
c. Team Leader (Name and Student ID)
d. Other Team Members (Name and Student ID)

**2. Teamwork Report**

a. Describe team forming, task assigning and your role and responsibility, and the project timeline (tasks allocated and completed dates)
b. Describe how the teamwork is conducted for the project
c. Describe a collaborative problem in the teamwork and how the problem is negotiated and solved by the team.

If you separate or quit a team in the middle of project, you use the following format and address the following questions.

**Template B**

**1. Team Details**

a. Team Name (optional):
b. Your Name and Student ID:
c. Team Leader (Name and Student ID):
d. Team Members (Name and Student ID):

**2. Teamwork Report**

a. Describe team forming, task assigning and your role and responsibility, and the project timeline
b. Describe why the teamwork has failed
c. Describe and justify the impact on you and other team members caused by the failure of teamwork

**Note**: this part is to address a team member's own individual experience with teamwork. The document is not shared between the team members; each team member needs to submit his/her own document individually. The same/copying contents of this part by two or more team members will be treated as plagiarism.

**Submission for Teamwork report**

Every team member is to prepare the report in a Microsoft Word document. This part will be marked for each team member.

You must submit your assignment via the online submission system from the unit Moodle site. Any hardcopy or email submission will not be accepted. After the marked assignments are returned, any late submissions will not be accepted.

**Note**: Please use Apache NetBeans Ant to implement your project. **Don't use NetBeans Maven.**

**What to submit for this assignment**

The team leader for each team should submit a zipped file of the entire project (NetBeans project) with source code, a word document about the design and test.

All students should submit the Teamwork report as specified above. Do not include the word documents in the zip file, you must submit it as separate files.

**Referencing Style**

- Harvard (author-date)

**Submission**
Online Group

**Learning Outcomes Assessed**

- Prepare software requirements specification following requirements engineering
- Apply modelling techniques to document architectural and system models as per the requirements specification
- Design and implement a multi-tiered software application consisting of presentation, application and data persistence tiers.
- Conduct test-driven development, validation and verification testing
- Discuss quality management in a range of different design and development scenarios
- Work independently and collaboratively in small teams.

**Graduate Attributes**

- Communication
- Problem Solving
- Critical Thinking
- Information Literacy
- Team Work
- Information Technology Competence
- Cross Cultural Competence
- Ethical practice

## 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.

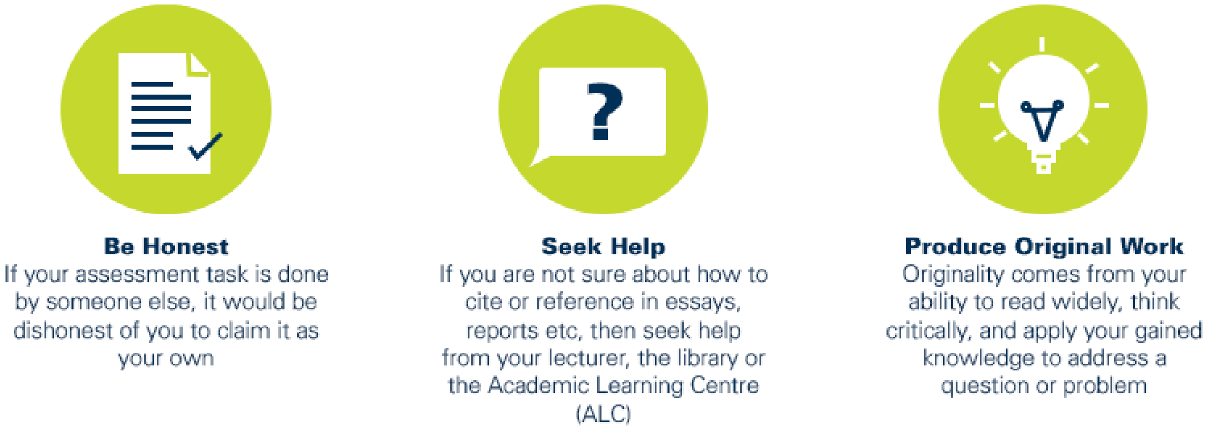
As a student, you are responsible for reading and following CQUniversity's policies, including the **Student Academic Integrity Policy and Procedure**. This policy sets out CQUniversity's expectations of you to act with integrity, examples of academic integrity breaches to avoid, the processes used to 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