OLA = Online Learning Adviser

QUT Online

IFQ563 Assignment Specifications

Overview

The goal of these assignments is to gain practical experience in object-oriented software development through object modelling with design diagrams and applying standard software design principles and patterns. This knowledge will be demonstrated by creating and documenting an object-oriented design and implementing it using C# for a real-world motivated problem.

While you can talk about the assignments with your OLA and peers, the submissions must be your own original work. Copying the work of others is not tolerated at QUT and submissions will be checked for code plagiarism against online repositories.

Description

You are to provide an extensible framework for many different two-player board games. To demonstrate that your framework can be easily adapted to different games, your design (including all design diagrams) must include ONE of the following games in the same software:

- Numerical Tic-Tac-Toe: A variation of the classic Tic-Tac-Toe game. Two players take turns placing the numbers 1 to 9 on a 3x3 board. The first player plays with the odd numbers, the second player plays with the even numbers. All numbers can be used only once. The player who first puts down 15 points in a line (sum of a horizontal, vertical, or diagonal row of three numbers) wins the game. Any other variant of Tic-Tac-Toe is not acceptable.
- Wild Tic-Tac-Toe: Like the classic Tic-Tac-Toe game, two players take turns placing an X or an O piece on a 3x3 board. However, in this game players can choose to place either X or O on each move. The first player who creates a line of three X 's in a row or three O 's in a row (horizontally, vertically, or diagonally) wins the game. Any other variant of Tic-Tac-Toe is not acceptable.

To demonstrate the feasibility and effectiveness of the design, you must implement the games that correspond to your design using C# on .NET 6.

Requirements

Your design should extract as many commonalities as possible from the different games so that the framework is extensible and reusable for a wide variety of board games.

Your system should cater for different modes of play, including:

- Human versus human
- Computer versus human

With human players, the system must check the validity of moves as they are entered. With computer players, the system can randomly select a valid move.

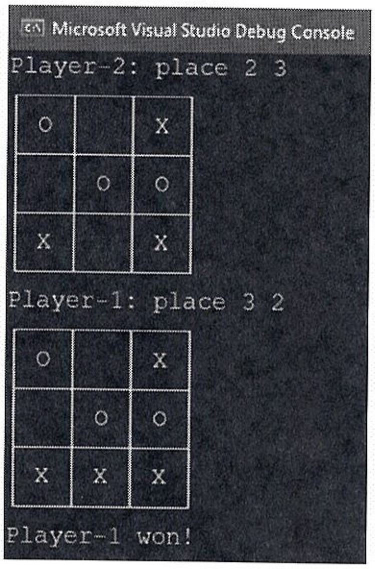
Games can be played from start to finish, and should be able to be saved and restored from any state of play (i.e. a save file). A game should be replayable from any position after being loaded from a save file.

During a game, all moves made by both players should be undoable and redoable (i.e. the full history of moves are tracked). But the history of the moves does not have to be saved into a save file. That is, immediately after loading the saved state from a file, undo and redo operations are not available (until new moves have been made).

You should provide at least a primitive online help system (i.e. accessible within the running game, rather than a separate offline help document) to assist users with the available commands (also give some examples if they are not obvious to use).

Comments on requirements

- The requirements are what you should aim to achieve in your project. Even if you do not implement all features, you should at least cover all requirements in your design.
- Some requirements may be vague and open to interpretation. As long as you clearly state any assumptions you make in your design report, we will accept reasonable interpretations.
- The important aspect is that your design for the core structure and functionality of the two-player board game meets the requirements, in a clear and easily understandable way. You should steer your design towards a general framework and consider your implementation as a proof-of-concept of your design, rather than a fully-featured, highly intelligent game-playing program.
- Your implementation must be a C# console application on .NET 6 and provide a text-based command-line interface (e.g. using either ASCII or Unicode) of the game as it is played. Some marks have been allocated for usability, so be sure that whatever interface you use, the commands are simple and clear. The following screenshot shows a simple example of playing a game of the classic Tic-Tac- Toe by two human players, displayed in Unicode. For the game of Numerical Tic-Tac-Toe, numbers 1-9 are placed on the board.



```
Player-2: place 2 3
┌───┬───┬───┐
│ O │   │ X │
├───┼───┼───┤
│   │ O │ O │
├───┼───┼───┤
│ X │   │ X │
└───┴───┴───┘
Player-1: place 3 2
┌───┬───┬───┐
│ O │   │ X │
├───┼───┼───┤
│   │ O │ O │
├───┼───┼───┤
│ X │ X │ X │
└───┴───┴───┘
Player-1 won!
```

Teaching Period 3a, 2024

IFQ563 Object Oriented Design

Assignment 1: Preliminary design

Word limit: 2 PDF pages per submission, 6 PDF pages total.

Weighting: 30%

Due date: 11:59pm AEST Sunday 28 July 2024 (Week 2)

After you have read this information, head over to the **Assignment Q&A** (https://canvas.qutonline.edu.au/courses/1658/discussion_topics/139333) discussion board to ask any questions and see what your peers are saying about this assignment.

Assignment overview

In this assignment, you will complete three design tasks, contributing to the overall design of **Assignment 2: Final design report and implementation** (https://canvas.qutonline.edu.au/courses/1658/assignments/7799). The purpose of each design task is to provide a checkpoint on your overall project progress and give you an opportunity to receive feedback from your peers and Online Learning Advisor (OLA) to improve the overall design of your project in the final design report.

Learning material from **Module 1: Object oriented (OO) mechanisms in C#** (https://canvas.qutonline.edu.au/courses/1658/modules/17077) and **Module 2: Design diagrams** (https://canvas.qutonline.edu.au/courses/1658/modules/17079) is required to complete this assignment.

(https://canvas.qutonline.edu.au/courses/1658/assignments/7795) Learning material from **Module 3: Design principles and patterns (https://canvas.qutonline.edu.au/courses/1658/modules/17080)** and **Module 4: Summary of design patterns (https://canvas.qutonline.edu.au/courses/1658/modules/17081)** will help you to refine your design for **Assignment 2: Final design report and implementation (https://canvas.qutonline.edu.au/courses/1658/assignments/7799)**.

This assignment supports **unit learning outcomes 2 and 3**.

The use of artificial intelligence tools (for example Chat GPT) is not permitted in any assessment in this unit. The use of such tools when not authorised may be treated as a breach of **MOPP C/5.3 Academic integrity (https://www.mopp.qut.edu.au/C/C_05_03.jsp)** and appropriate penalties imposed.

Assignment details

Read the **IFQ563 Assignment specifications (DOCX 129 KB) (https://canvas.qutonline.edu.au/courses/1658/files/444786?wrap=1)** **(https://canvas.qutonline.edu.au/courses/1658/files/444786/download?download_frd=1)** for a description of the assignment framework and requlrements before contlnulng.

The design tasks should record your design based on the project requirements outlined in the IFQ563 Assignment specifications document. No implementation detail should be provided. Each task should be a PDF document format, up to two pages in length.

You are required to complete the following three design tasks:

- CRC analysis
- class diagram
- object diagrams and sequence diagrams.

You will have an opportunity to share your progress with your OLA and receive feedback on your design tasks before you submit this assignment. This will occur in the following discussion board:

- **2.10 Discussion: Assignment feedback opportunity 1** (https://canvas.qutonline.edu.au/courses/1658/discussion_topics/139340).

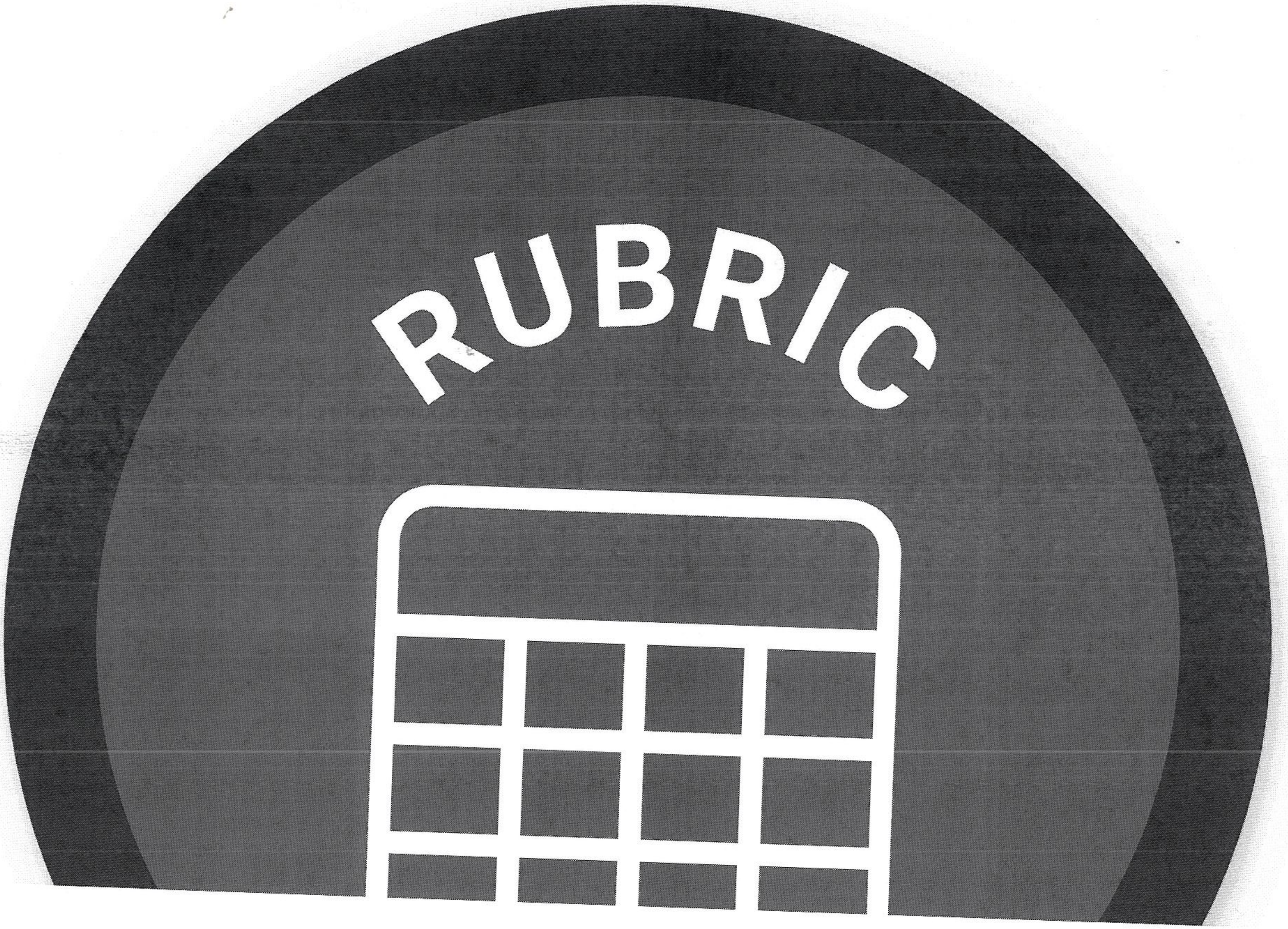
Throughout the learning materials, you will be reminded to complete each of these design tasks in the following activities:

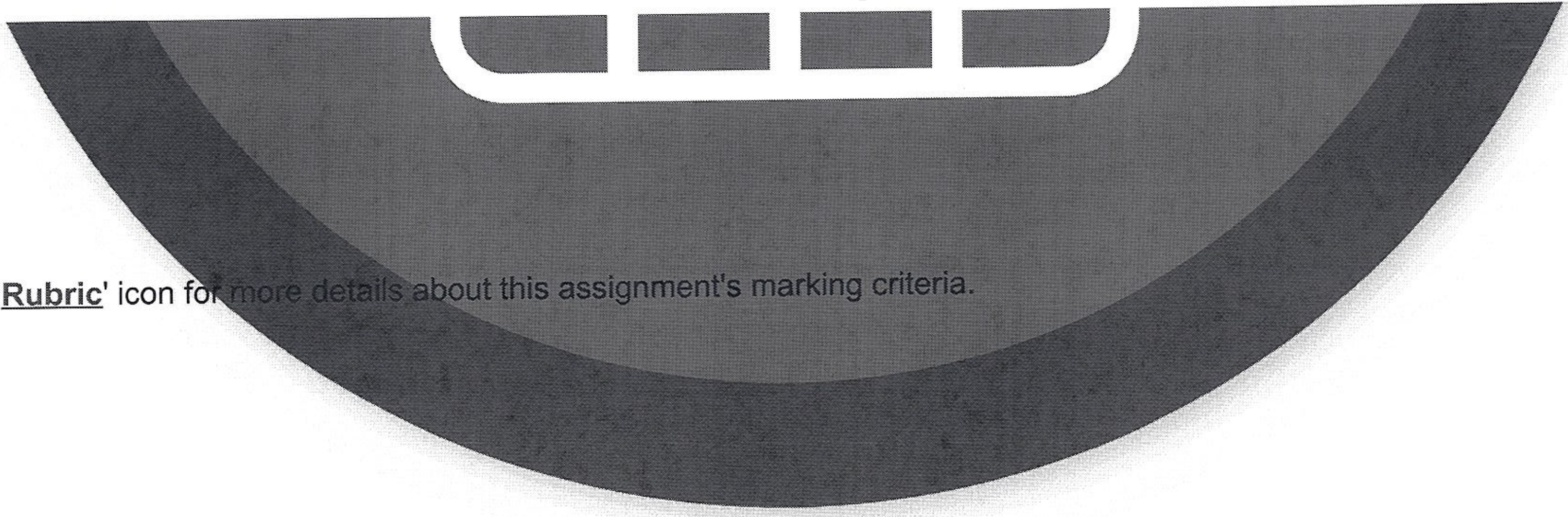
- **2.5 Activity: CRC analysis** (https://canvas.qutonline.edu.au/courses/1658/pages/2-dot-8-activity-crc-analysis)
- **2.7 Activity: Class diagrams** (https://canvas.qutonline.edu.au/courses/1658/pages/3-dot-9-activity-class-diagrams)
- **2.9 Activity: Object and sequence diagram** (https://canvas.qutonline.edu.au/courses/1658/pages/4-dot-10-activity-object-and-sequence-diagram).

Your design tasks have been split into weekly tasks to help manage your workload. You should review all of these design tasks as soon as possible to ensure you can receive targeted feedback if you require it.

Assignment criteria

1. CRC analysis.
2. Class diagram.
3. Object diagrams.
4. Sequence diagrams.


RUBRIC



Rubric

Select the '**Rubric**' icon for more details about this assignment's marking criteria.

▶ Assessment declaration

▶ Submission details overview

▶ Request Assignment Extension