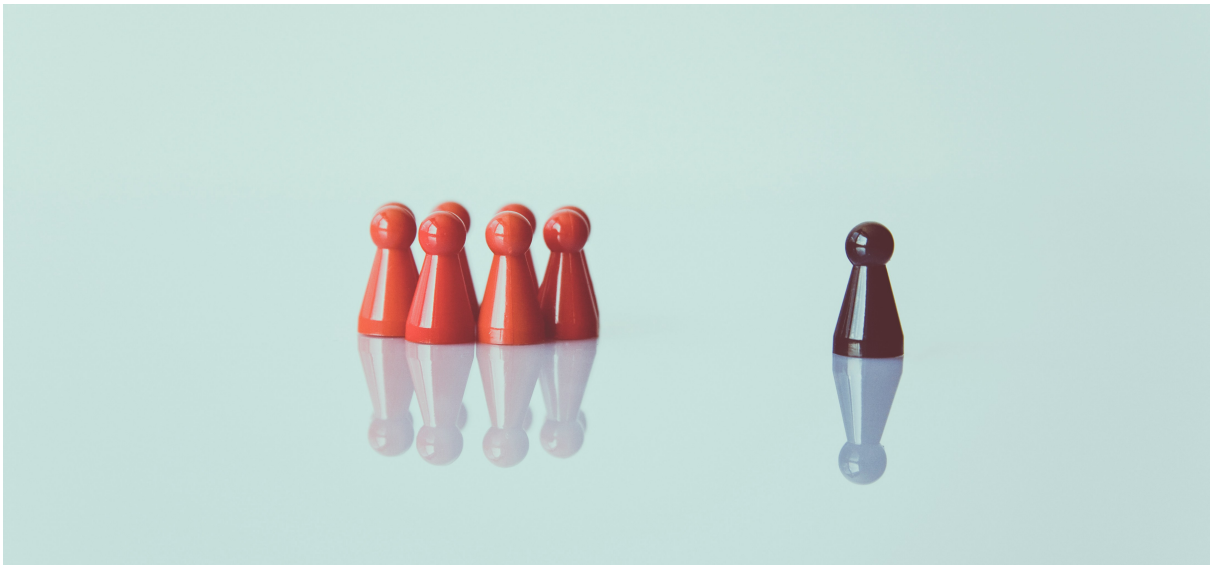

# Task 2 - Game Analysis and Prototype

## Course

Year 10 Information and Software Technology

## Submission Date

Tuesday, 15 October before 8:30 am via Reshet

## Outcomes

The following reporting outcomes are being assessed:

- Applies computational, design and systems thinking to the development of computing solutions
- Applies iterative processes to define problems and plan, design, develop and evaluate computing solutions
- Communicates ideas, processes and solutions using appropriate media

## Instructions

In this task, you will access some functional source code for a card game. Using that code, you choose which cumulative criteria you wish to complete. You will be awarded three marks and a Grade E for the classwork you have

completed on these topics. From there, you choose which inclusive level you will complete and submit.

**Grade E**

All students are awarded this result based on your participation in class activities.

**Grade D**

You must analyse the code, use appropriate research methods to explain each section/line with inline comments and compile a set of rules for the game to be added to the code in an appropriate place.

**Grade C**

You will design a fully functional prototype for the game that accurately reflects the source code using a suitable application (e.g. Figma or Axure) that uses the principles of good UI/X design.

**Grade B**

You will use a Python GUI library to convert the source code to code that displays a user interface reflecting the UI/X design in your prototype.

**Grade A**

You will add a new feature to the gameplay that is not included in the initial source code. The new feature must be reflected in the explanation of the gameplay (Grade D), the prototype (Grade C), and the GUI source code (Grade B).

## Source Code

You can access the source code at the GitHub repository below:

https://github.com/Emanuel-School/2024-10ist-task2/tree/18672cda47a6e8f40cd540396b08b2dcddb31c34

## Submission

You will submit your project as a link to an appropriately shared GitHub repository. Your repository should contain three (3) files:

- Text or markdown file with rules for gameplay

- A link (URL) to your functioning prototype
- Your final Python source code

## Marking Criteria

| Grade | Result | Performance descriptor |
|---|---|---|
| A3 | 3 | An additional feature is added to the game that is seamlessly reflected in the revised game rules prototype, GUI interface and source code |
| A2 | 2 | An additional feature is added to the game that is documented and substantially reflected in the prototype, GUI interface and source code |
| A1 | 1 | An additional feature is added to the game that is either documented OR reflected to some extent in the prototype, GUI interface and source code |
| B3 | 3 | Source code rewritten to include a GUI that is a faithful reproduction of the prototype |
| B2 | 2 | Source code rewritten to include a GUI that is substantially correct version of the prototype |
| B1 | 1 | Source code revised with. GUI to include a partial reproduction of the prototype |
| C3 | 3 | Fully functional prototype with interactivity, consistent UI design, and correct game rules |
| C2 | 2 | Partially functional prototype with interactivity, mostly consistent UI design and game rules |
| C1 | 1 | Functional prototype with aspects of UI design needing improvement and game rules inconsistently applied |
| D3 | 3 | Detailed interpretation and explanation of the source code |
| D2 | 2 | Substantially correct interpretation and explanation of the source code |
| D1 | 1 | Satisfactory interpretation and explanation of the source code |
| E | 3 | Based on classwork |