CSE212: SOFTWARE DEVELOPMENT METHODOLOGIES

YEDITEPE UNIVERSITY

SPRING 2024

TERM PROJECT – DUE DATE JUNE 3RD, 2024

As a term project, you are required to develop a single player Memory Cards game. The game must consist of 3 different levels, from easy to hard. Furthermore, the player should be able to choose whichever level they want to start from, however, if they start playing the game from the first level, they are required to go through all levels to reach the third level.

While developing the game you can use the following statements and figures as a guideline:

- For usability purposes, you are required to implement a graphical user interface (GUI) for your application.
- Once the game starts, a main menu should appear (see Fig 1).

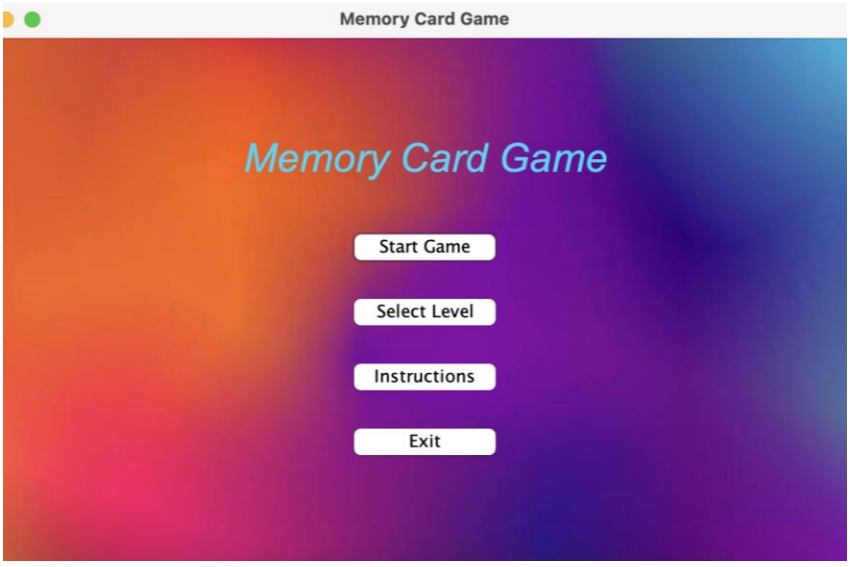


Fig 1. The main menu for the Memory Card game.

- On this menu, there must be 4 buttons in total. If the player presses the *Start Game* button, the game will start from Level 1.
- If the player presses the *Select Level* button, they should be given an option of choosing the level they want to play.

- If the player presses the *Instructions* button, details regarding the gameplay should be presented to the player (see Fig 2).

Fig 2. Instructions window for the game.

- If the user presses the *Exit* button, the application should terminate.
- Once the user presses the *Start Game* button and the game starts from Level 1, the screen below should appear to the player (see Fig 3).

Fig 3. Memory Card Level 1 screenshot.

- Notice that the player has a total of 18 tries until they lose the first level of the game. Furthermore, please observe the menu structure which consists of *Game*, *About* and *Exit* items (see Fig 4).

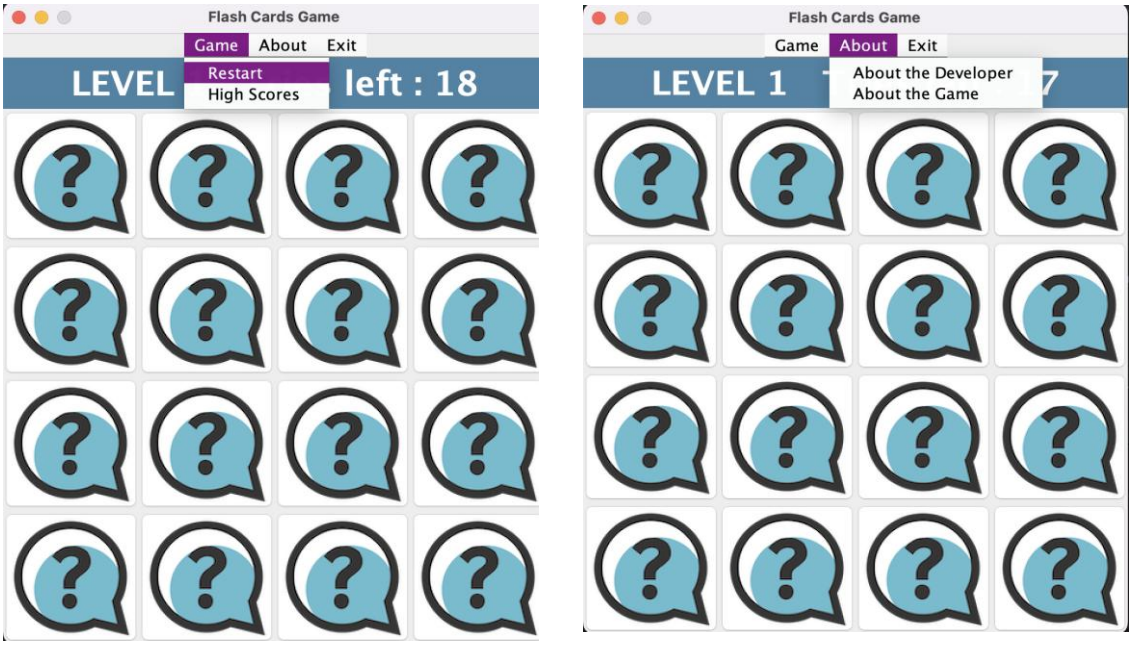


Fig 4. The menu bar structure for the game.

- The *Game* menu item should have *Restart* and *High Scores* subitems. The *Restart* subitem should allow the player to replay the level from the beginning. The *High Scores* subitem should display the last 10 scores gathered by the player. *High Scores*, which entails the name of the player and their scores, should be stored on a flat text file.

- The scores of the player should be calculated according to the below guidelines:

  - The player's initial score should be set to 0.
  - If the player makes a correct match, +5 points should be added to their overall points for Level 1, +4 points for Level 2 and +3 points should be added for Level 3.
  - If the player makes a bad match there should be a penalty. For every single bad match made by the player, -1 should be subtracted from the player's overall score for Level 1, -2 points should be subtracted for Level 2 and -3 points of penalty should be applied to the player's score.

- The *About* menu item should have *About the Game* and *About the Developer* subitems. *About the Game* subitem should contain information about the game and *About the Developer* subitem should contain information about the student -name, surname and number (see Fig 4).
- It is of utmost importance that the placement of images are randomised every single time a new game or a new level is started. (Hint: You can use Collections for this purpose.)
- **If you employ Threads for this process for every single level, you will receive a BONUS. [10 POINTS]**
- The first level's cards are themed around Internet and Social Media while the second level's cards revolve around Cyber Security. Lastly, the third level focuses on Gaming PC parts.
- Last but not least, for Level 3, you are required to implement *Threads* such that **the cards will be shuffled every single time the player makes a bad match**.

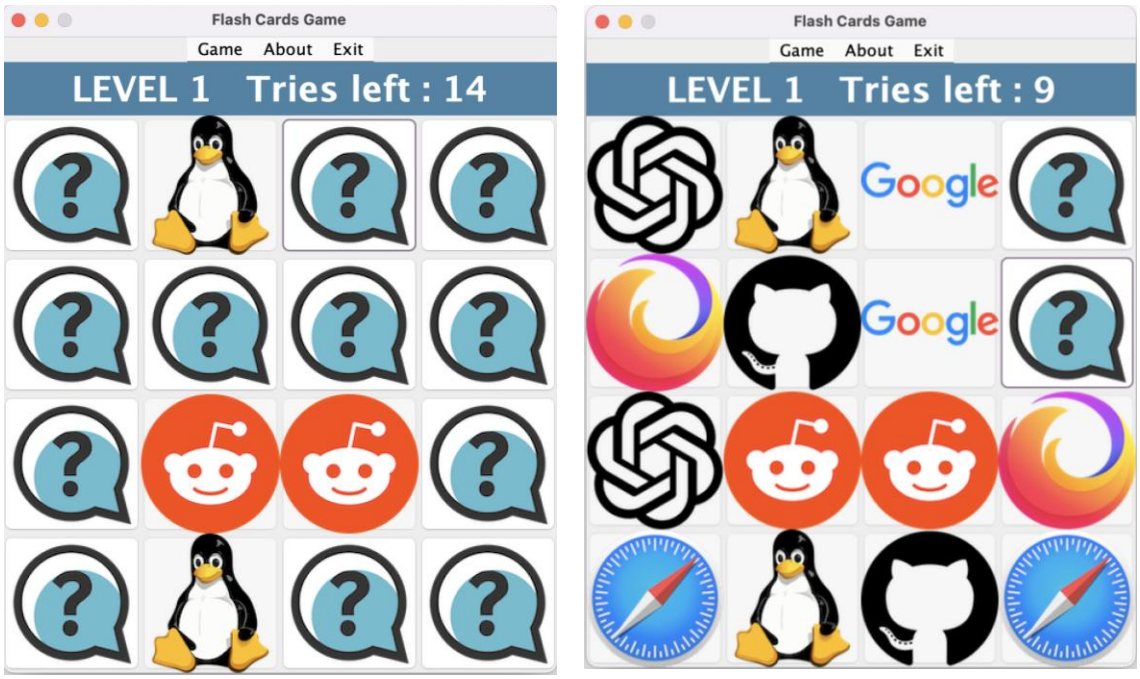
PS: Every asset used for the making of this game can be found on the course's YULEARN page. It is highly recommended that you utilise the assets provided, rather than finding your own assets. You may want to resize the assets if need be.

Keep in mind that the submissions will be checked using a plagiarism checker. Projects with high similarity scores will NOT be marked.

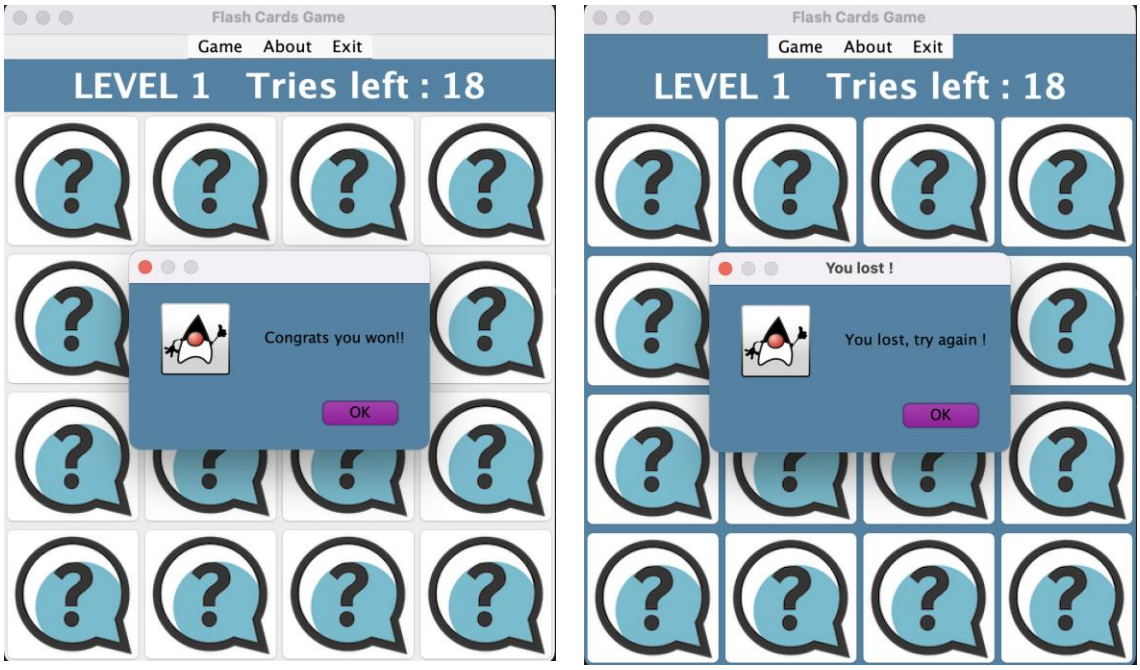
Submit your project in a zip file [class files will not be marked], which has your student number as name, through the YULEARN (https://yulearn.yeditepe.edu.tr) by the end of Monday, June 3rd, 2024.

Appendix: Application Walkthrough

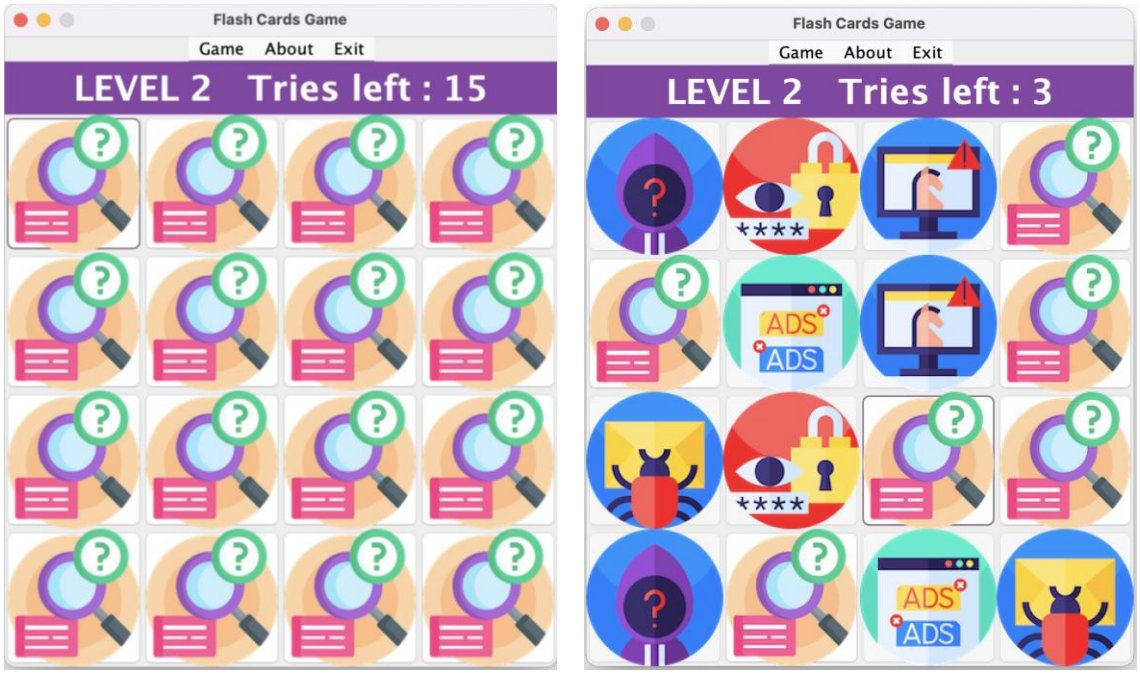
Please refer to the figures below for further information about the game and level logic. Keep your GUI as similar as possible to the images shown in these figures. Further creativity can be added, however, they will NOT be marked.



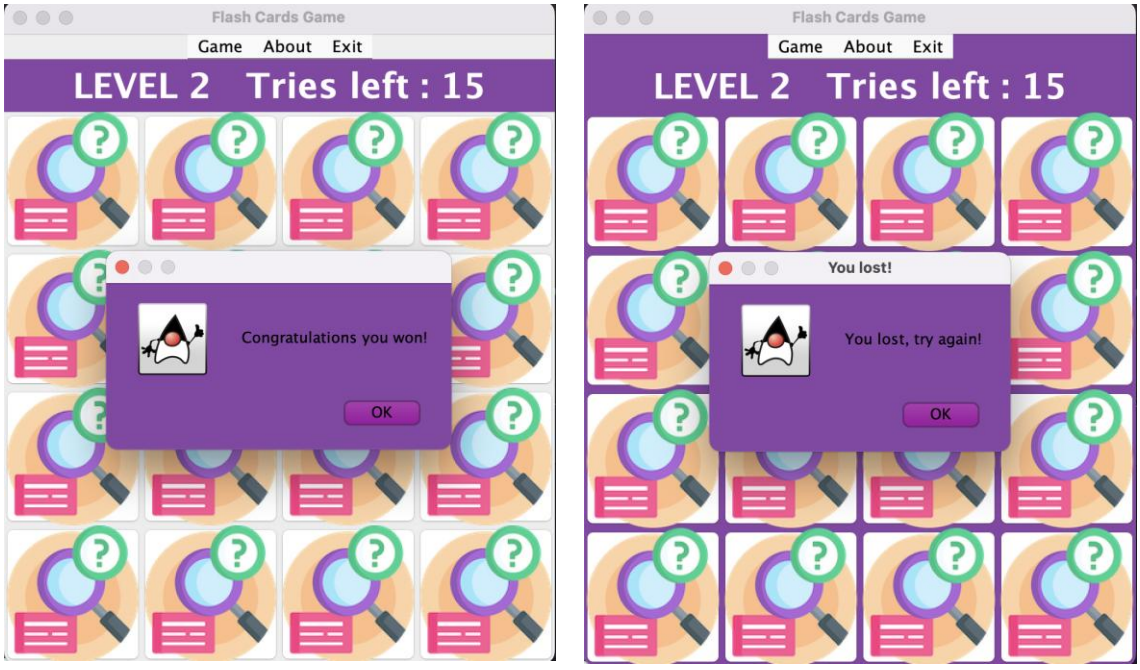
Level 1 guessing process. Notice that the Tries Left value keeps decreasing as the player makes matches. If they do a correct match, tries left will not decrease.



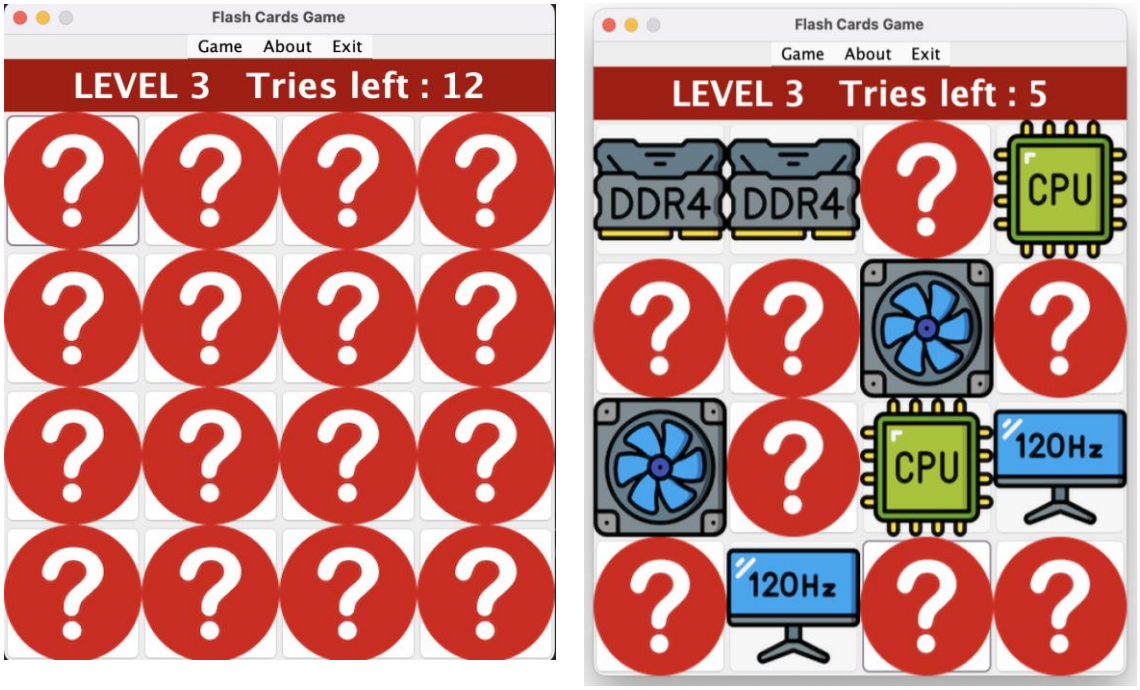
Level 1 win-lose situations. If the player wins this level, they should be forwarded to the next level. If they lose, they should be forwarded to the first level.



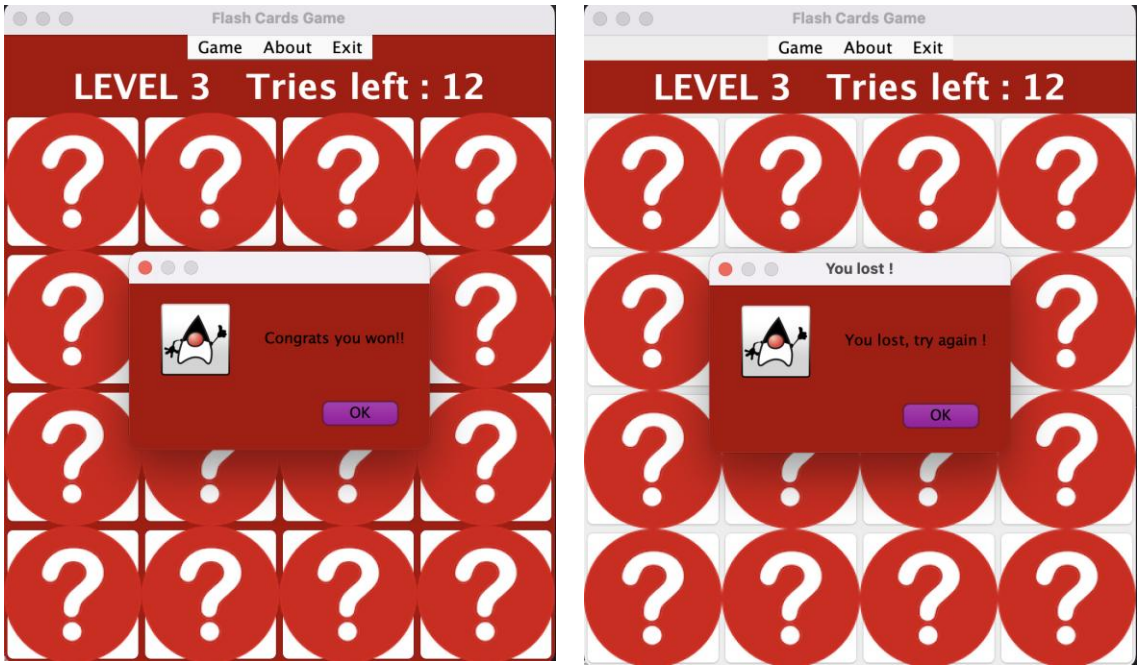
Level 2 guessing process. Tries Left value initially starts at 15 in this level.



Level 2 win-lose situations. If the player wins this level, they should be forwarded to the last level. If they lose, they should be sent back to the first level.



Level 3 guessing process. Tries Left value initially starts at 12 in this level.



Level 3 win-lose situations. If the player wins this level, a game over screen should appear, informing the player of their final score. If they lose, they should be sent back to the first level.