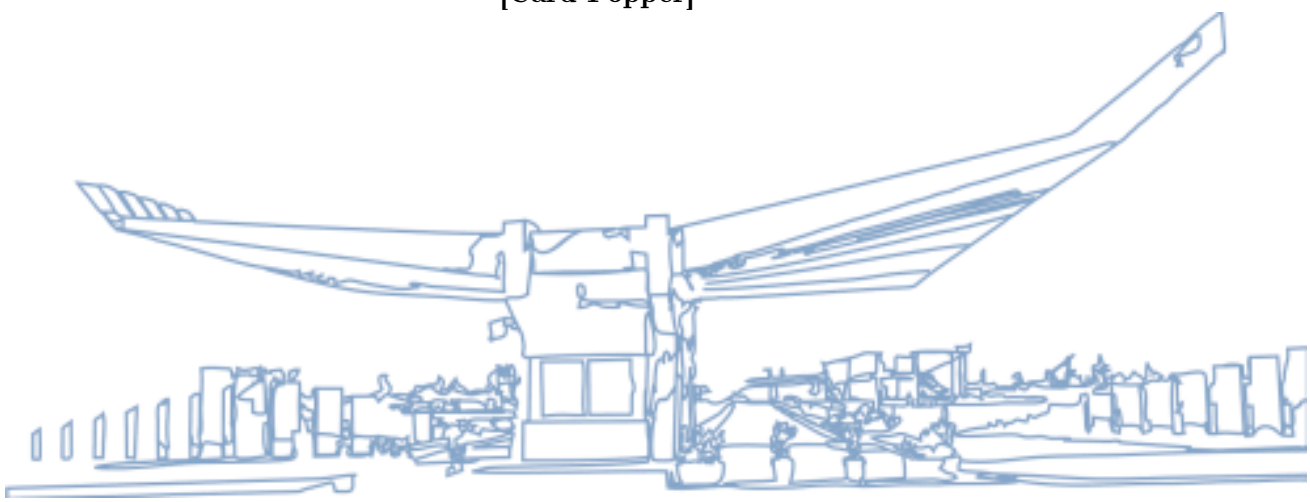
INTRODUCTION TO SOFTWARE ENGINEERING

[Card-Popper]



Team Members:

Lumturim Nela

Klevi Xhani

Ralf Poni

Armando Loka

Albion Biba

Table of Contents

1. Executive Summary

1.1 Project Overview

This project is a simple game involving matching identical looking cards in order to accumulate points in order to advance in the game. This project will offer a fun and interactive experience for kindergarteners. The scope of this project includes developing the memory and critical thinking skills of the kindergarteners.

1.2 Purpose and Scope of this Specification

The aim is to evolve kids' concentration and memory while at the same time making an entertaining game for them to experience. The difficulty will increase while the levels advance, posing a constant challenge for them.

In scope

The game works by letting the user turn over 1 card at a time with a maximum of 2 cards being facing up at the same time, if the cards match then points are awarded, if the cards don't match they're turned face down. The game also encourages the kindergarteners by adding a competitive component to the game using leaderboards.

Out of scope

The game is not intended to be helpful for children above the age of 7 due to it not providing enough of a challenge for those age groups.

2. Product/Service Description

This game requires a simple computer with minimal specifications and does not require any pre-preparation from the user and it's simple to start playing just click and play. Maintenance is needed on the back-end to keep the leaderboard up to date and functional.

2.1 Product Context

This game is a completely independent python game. The game itself is self-contained and independent; although the leaderboard system, which is not necessary to fully experience the game, operates on an online system. Scores achieved by the users are saved and uploaded to an online server which then displays in an ordered fashion the highest scores achieved by the users in the leaderboard screen. In order to achieve a functional leaderboard system.

2.2 User Characteristics

The target demographic for the game is kindergarten students, who may not be capable of performing complex logical tasks and who do not understand how to operate a device at an intermediate level. The users are expected to be able to make simple decisions incentivised by a graphical interface.

2.3 Assumptions

The program can run on any version of Windows, iOS and Linux. For the program to work Python and the Pygame must be properly installed. It is assumed that a teacher or a person with minimal programming experience should be available to set up the program.

2.4 Constraints

The project has no major constraints, the main constraints are Pygame limitations regarding performance and its appropriateness for designing this style of software. This isn't a major issue since the program isn't large enough to require memory or performance optimization

2.5 Dependencies

•Dependencies for this software include only the pygame library, and possibly a internet connection.
•The leaderboard needs a dedicated server for scores to be uploaded and stored.
•The leaderboard system requires taking data from a server and selecting the appropriate values to be displayed.

3. Requirements

The program must immediately show a main menu screen as soon as the code is run with the following options/buttons: "PLAY" button, "LEVELS" button, "LEADERBOARD" button and "EXIT GAME" button. The "PLAY" button must start the game at the first available level, The "LEVELS" button must show a screen that has all the available levels to be played with the ones unavailable to play grayed out and unable to be clicked, the "LEADERBOARD" button should show a screen with a leaderboard with player names ranked by points earned (points will be gauged depending on the time it took each person to complete a level) as well as the position of player playing right now on the board. Lastly, the "EXIT GAME" button must immediately terminate the program. It is to be noted that none of the progress will be saved and the game will start from the beginning each time the code is run, the only thing saved will be the leaderboard which will be uploaded to a server using only the points earned and the name that the player has chosen to identify his points with.

The gameplay

The template of the game will be a background graphics of which are likely to change from level to level with a pause button on the top left corner, and a number of cards appearing on top of the background face down. The amount of cards MUST always be even, however the exact amount of cards can vary from level to level as difficulty changes, it is certain however that the minimal amount of cards for each level will be at least 8. The player can play the game by clicking any of the cards one at a time, after every click the selected card will appear face-up with the user being able to see them, after a maximum of 2 cards are selected and are shown face-up, the game will immediately check if those cards are matching with each other and will proceed on one of two ways: in case they

are not matched the algorithm will provide a specific time period after which both of the cards will be returned to the face-down state and in case the cards match the algorithm will almost immediately remove those cards from the current game while checking if the those two cards are the last ones in the field at the moment, if so the level will end awarding the gathered points to the player and show a splash screen with the options to restart the level, choose level, and next level, otherwise the game will continue as normal repeating the same loop that was just mentioned.

End of level splash screen

As soon as a level ends, three buttons will appear as mentioned above in the following order side-by-side: “RESTART” button, “LEVELS” button, and “NEXT” button which have the same function as their names suggest.

3.1 Functional Requirements

REQ♯	REQUIREMENT	COMMENTS	PRIO RITY	DATE RVWD	
DR 01	The program must open by presenting the user with a list the choices: "Play", "Levels" "Leaderboard", "Exit Game"	Design Process = "Main Menu"	1	5.31.2022	
DR 02	The program must display a screen with the options "Restart", "Levels" and "Next" upon a level being cleared	Design Process = "post-level congratulatory screen"	1	5.31.2022	
DR 03	The program should feature a leaderboard which operates on an online server and display the top scores set by users	Design Process = "Leaderboard system"	2	5.31.2022	
DR 04	The program should feature colorful slides and animations for the cards and for when together	Design Process = "Sprites and Animations"	2	5.31.2022	

3.2 Non-functional requirements

The app must be designed to keep the children engaged and curious. The design has to be easy and understandable. With obvious images, shapes and instructions that don't confuse the children. Simple and attractive. We can use figures, animals, toys, cartoon characters, to make the design as familiar as possible.

In order to peak interest vibrant color will be used, within a similar palette. Colors like red and orange to capture the attention of the kids, green to provide a slight relaxation for the eyes and blue and purple to provide a relaxing yet intriguing feeling within the kids.

Smooth animations with simple transitions will be used to provide a more playful feeling.

Another thing will be a timer to always give the feeling of uneasiness and push the kids into playing in a more competitive way.

The user will not need to use functional keys, except for specific sections of the game like the end-level splash screen.

3.2.1 User-interface requirements

In addition to functions required, describe the characteristics of each interface between the product and its users (e.g., required screen formats/organization, report layouts, menu structures, error and other messages, or function keys).

The window will be a size of 1000 x 800 pixels and every action will be done through the mouse.

Home screen

As soon as the game is initiated the player should be greeted with 4 buttons laid in parallel to each other in a vertical order. Each button will have a yellow background (FFE63A hex value), with Arial text format and black text color and a size of 24. The 4 buttons will have written on them in the following order from top to bottom: "PLAY", "CHOOSE LEVEL", "LEADERBOARD", "EXIT GAME". The background will be changeable during the rounds top provide some passive stimuli to the player

In game

Every level will have a set of face down cards laid on the screen, visible to the player. A pause button in the top left corner of the screen which the user can click to pause the game, exit the level, check leaderboard and go to the levels screen button.

The cards will have a white border with a purple background on each of them on their face down state and for the face up state, each pair of cards will use types of fruits.

The pause button will have the same format as the button that appears on the home screen: FFE63A background color and Black Arial size 24 text.

Levels screen

The levels screen will show all the possible levels available for the player to play which will again appear the same way as the button mentioned until now in the game, FFE63A background color and Black Arial size 24 text. The progress in the game will be visualized on this screen by having all the

levels that the player has yet to clear appear in 949494 background color, and black 24 Arial text. There will also be a button on the top left corner which will make it possible to go back to the last screen. Also under each level button a time will be shown and a number under it which will show the time it took to complete each level and the points awarded by the game to the player for each level, represented in a white(949494) arial size 14 text.

Leaderboard screen

At the leaderboard screen there will be a box with 400 x 600 dimensions, which will show a screen with a leaderboard with player names ranked by points earned (points will be gauged depending on the time it took each person to complete a level) as well as the position of player playing right now on the board. In case the leaderboard is not connected to the database a error message will appear letting the player/user know. The leaderboard will only show the top 100 players as to not affect performance too much which will be shown in a list way with a scrollable interface, with the rank of the current player be shown at the top left corner of the screen making it possible for the player to go back to the last screen

3.2.2 Usability

Ease of use:

- The game should be easy to understand for anyone, including children aged 4 and above.
- The program mechanics should be as simple as they can without taking away any core requirements of the game.
- The end objective should be very clear and reachable.

3.2.3 Performance

- The program itself must be able to maintain itself for an indefinite period of time.
- The leaderboard system must be able to process 40 scores being uploaded at the same time.
- The leaderboard system must be able to hold 100 scores at all times.

3.2.4 Manageability/Maintainability

3.2.1.1 Monitoring

An assigned backend developer will work towards setting up and maintaining the leaderboard system, if the system detects any errors they are uploaded to a log which the developer will routinely check.

3.2.1.2 Maintenance

The game is simple enough to not need any complicated maintenance design. The game itself should contain no bugs or errors and if any are found they are to be reported to the development team which will then fix them.

3.2.1.3 Operations

In case of bugs, errors and any kind of vulnerability and/or problem within the software, a link will be provided in the Github depository in order to create support tickets and inform the developer team.

3.2.5 Security

3.2.5.1 Protection

User data which may be collected by the server required for the leaderboard system is encrypted and safely stored. In the case when the leaderboard no longer requires the user data which is stored in it, the data is promptly deleted. Otherwise no other type of information is gathered nor saved by the software.

3.2.5.2 Authorization and Authentication

There is no logging in in this software, the only authentication is for the Leaderboard section where the user need to input a username in order to save the highscore.

3.2.6 Standards Compliance

Changes made to the program must be added to a changelog, which is hosted on Trello. After major changes a discussion is to be held regarding the changes and the functionality of the program. Ideas and potential additions to the program are noted down in a to-do list and given a priority based on how important the development team and the intended audience deem it to be.

4. Design thinking methodologies

4.1 Negotiation:

The project moves forward from discussing ideas regarding the mechanics and functions of the program. Meetings are regularly held between members in order to decide how to move forward with the project.

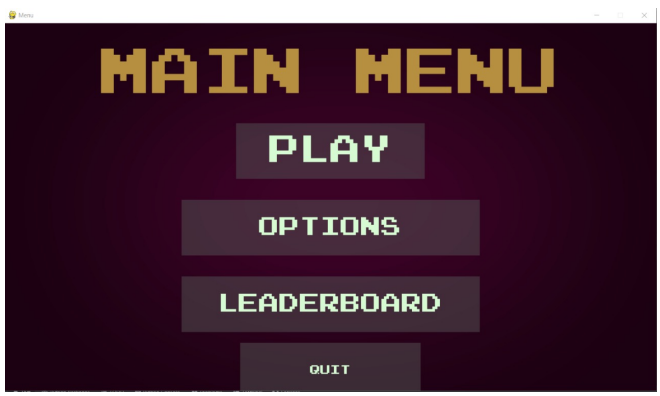
4.2 Critique:

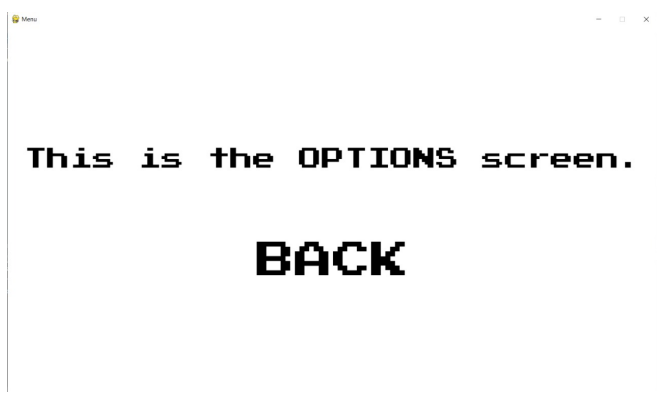
Ideas and implementations are discussed and their pros and cons are highlighted. The most beneficial ideas are put in a to-implement list and the members work to integrate these ideas into the program itself. Changes are made based on project members' liking of the idea. Each and every idea is discussed and considered.

4.3 Noticing

Members are tasked with contributing to the project by making changes or editing existing functions in the program based on their observations and how these observations relate to the project. On top of this members are to converse with and analyze the behavior of the target audience so as to decide what works and what needs to be changed.

4.4 GUI





5. Software design

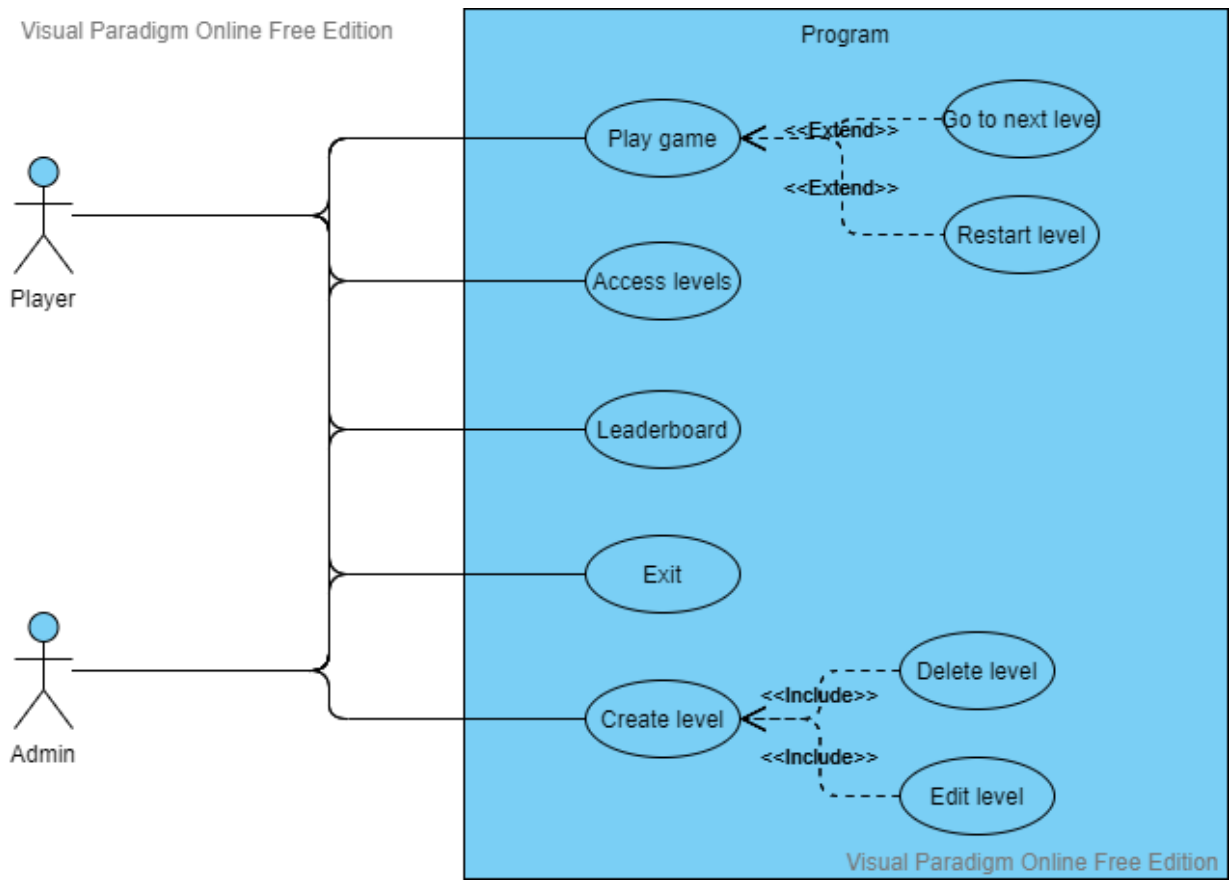
In order to design the UML diagram we used Visual Paradigm Online, an easy to use website that enables users to create different types of charts, graphs, etc. easily.

5.1 Use Case

The user case diagram will explain the interaction between the actor(kids) and the game which are explained in the diagram below.

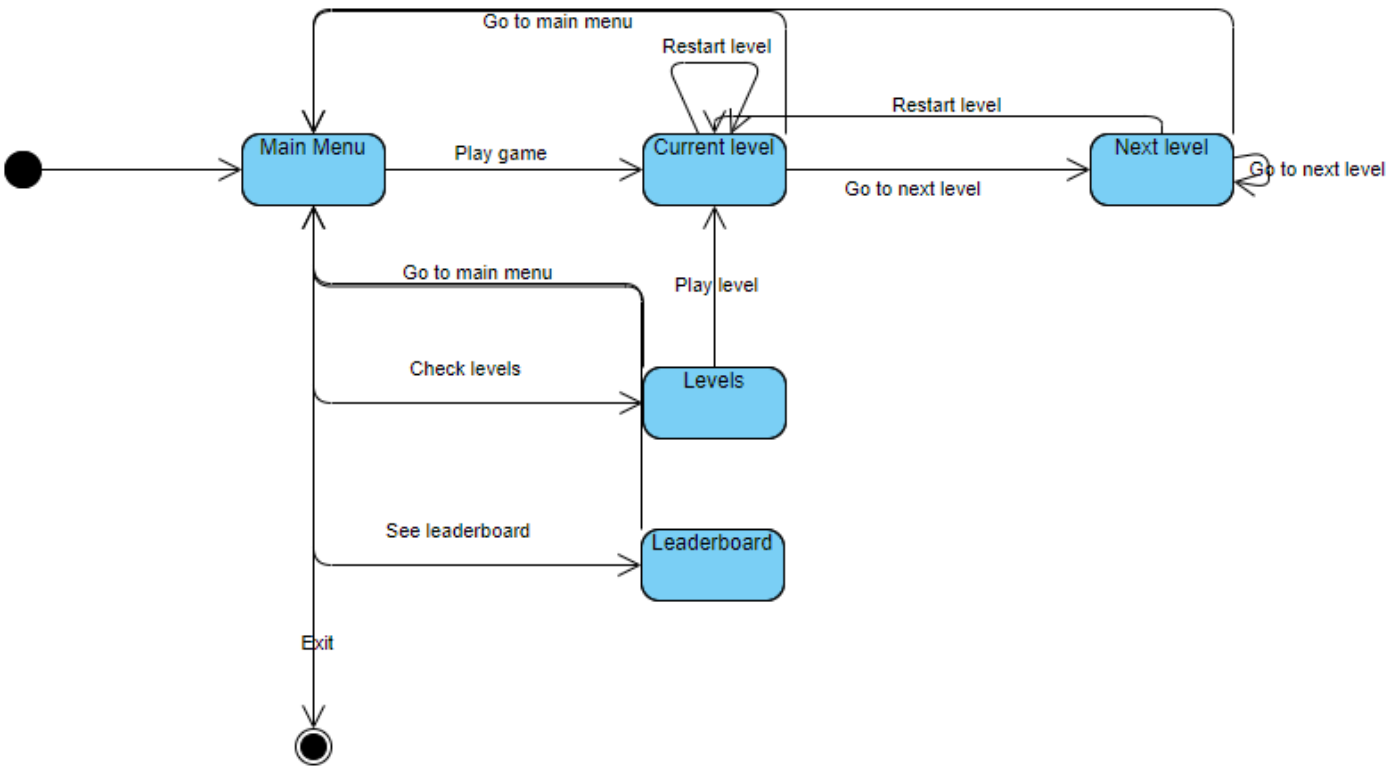
The explanation of the diagram:

- After the user has clicked on the PLAY button the game will immediately start either at the first level or the last played depending if the user had been playing before, however not in the case the game is started anew, in that case the game will immediately direct the user to the first level.
- Clicking the LEVELS button will send the user to a page showing all the available levels to play and making it possible to start at a level of his choice but only the ones previously finished.
- The LEADERBOARD will show the user a list of players with the biggest highscore in the game.
- And lastly the EXIT GAME button will terminate the program.



5.2 State diagram

The state diagram shows the path the user's interactions follow as he passes through the stages. Almost every stage is unlocked for the user with him being able to access any part of the program, the only restrictions include next level where unless the user has already won that specific level he won't be able to proceed any further in the game.



5.3 Class diagram