Memory Card Game

Team Members

• محمد اشرف فوزي علوفه	3900
• محمد اشرف محمود محمود	3901
• محمد السيد حسن احمد	3902
• محمد السيد عبد الحميد	3904
• محمد السيد عبدالمعبود ابراهيم	3906
• محمد السيد محمد السيد	3908
• محمد عادل عبدالوهاب	3942
• محمد عامر عبدالعزيز	3943
• مهند عاصم عثمان علي	4058

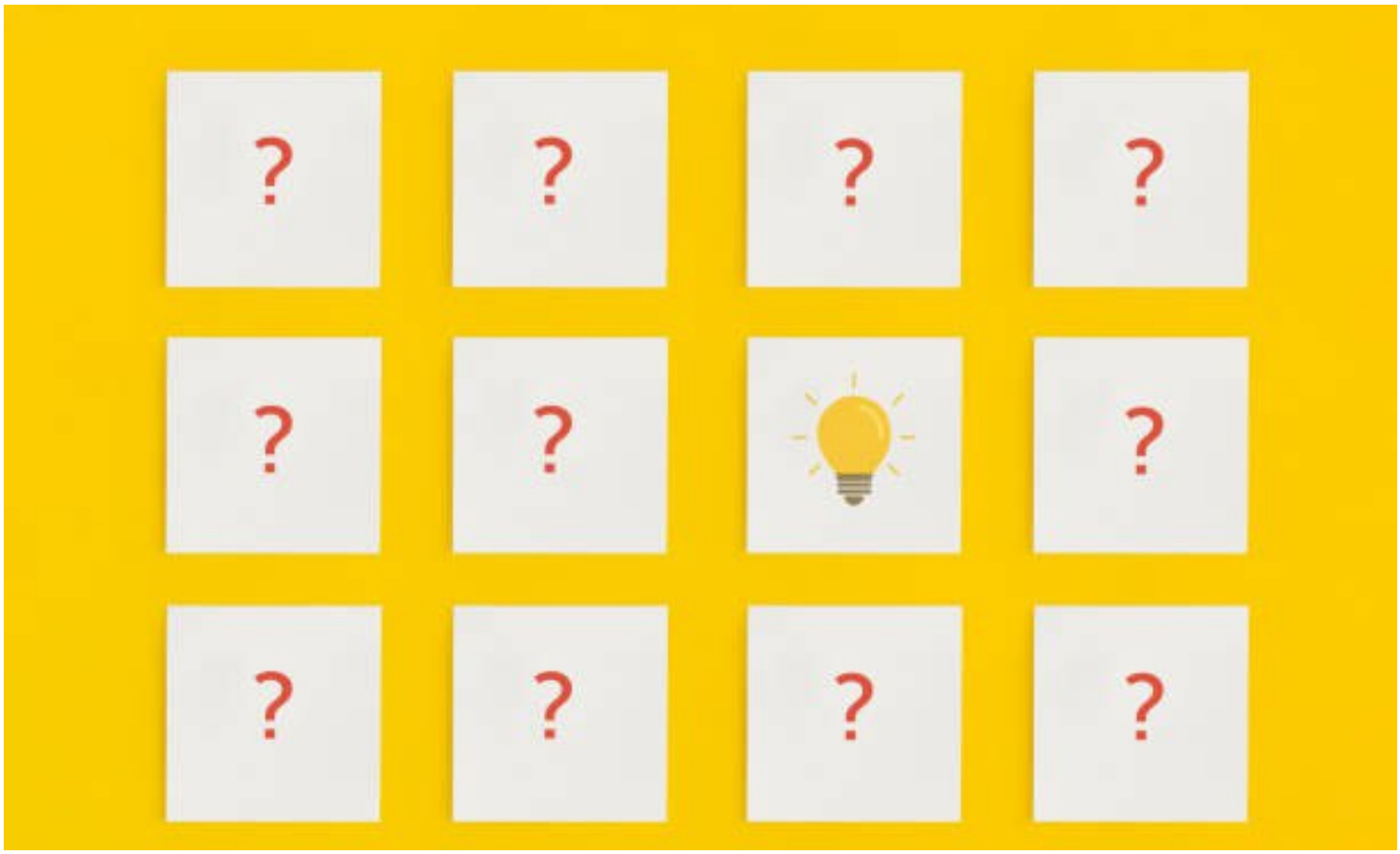
MAIN IDEA

Memory Card game is a memory game that consists of a set of cards randomly arranged on the screen. Each card contains a picture, number, or symbol.

The game starts when the player clicks on any card to reveal what is underneath it, and then clicks on another card in an attempt to find the card that has the same picture, number, or symbol. If a pair of matching cards is found, they are displayed on the screen and removed from the game. If the two cards are not matching, they return to their original state after a short period and are hidden again.

The winner of the game is determined when all the cards are removed from the game. The difficulty of the game can be increased by adjusting the number of cards in the settings.

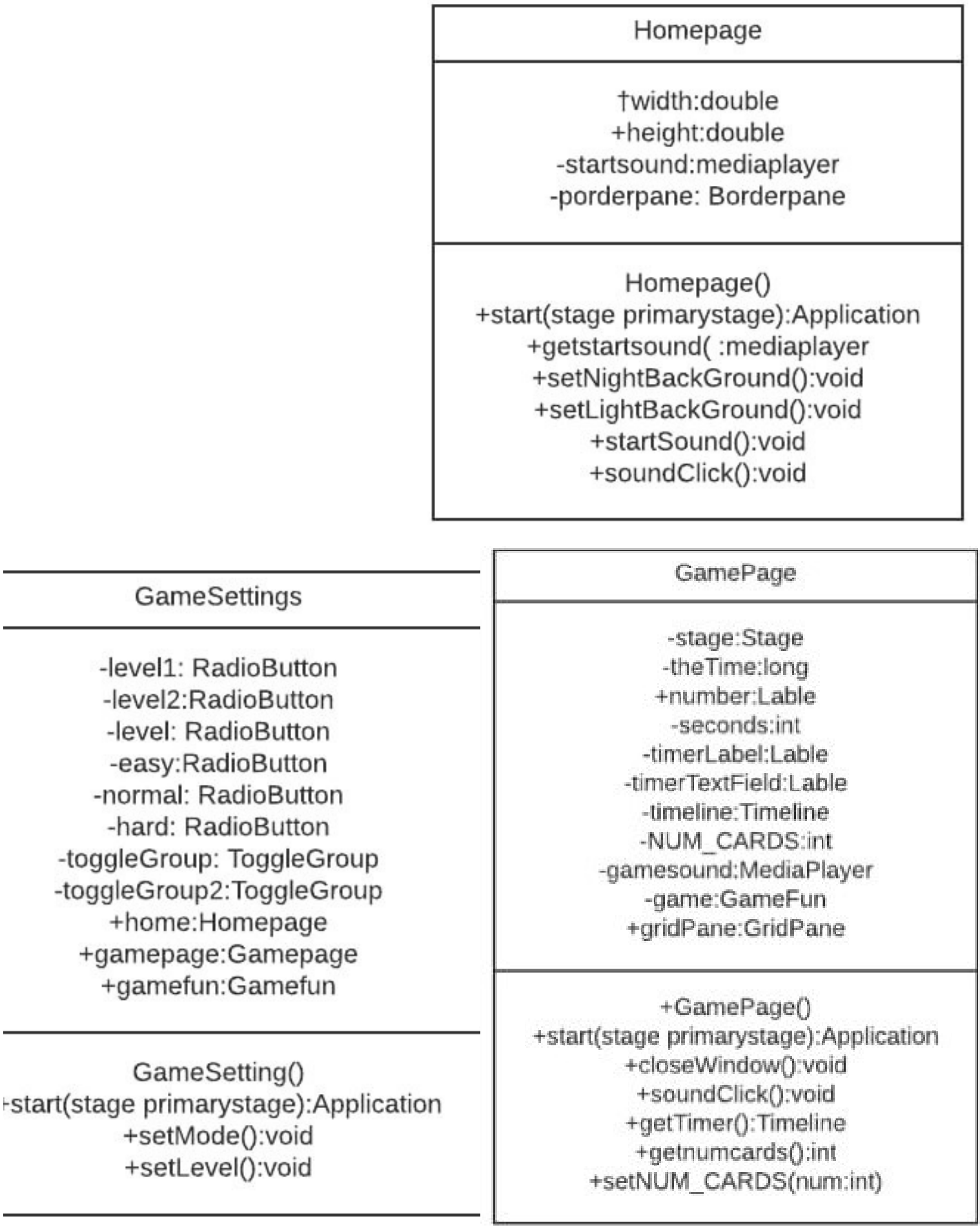
Memory Card game is an entertaining game that improves memory and focus. It can be easily played by anyone and is suitable for all age groups.

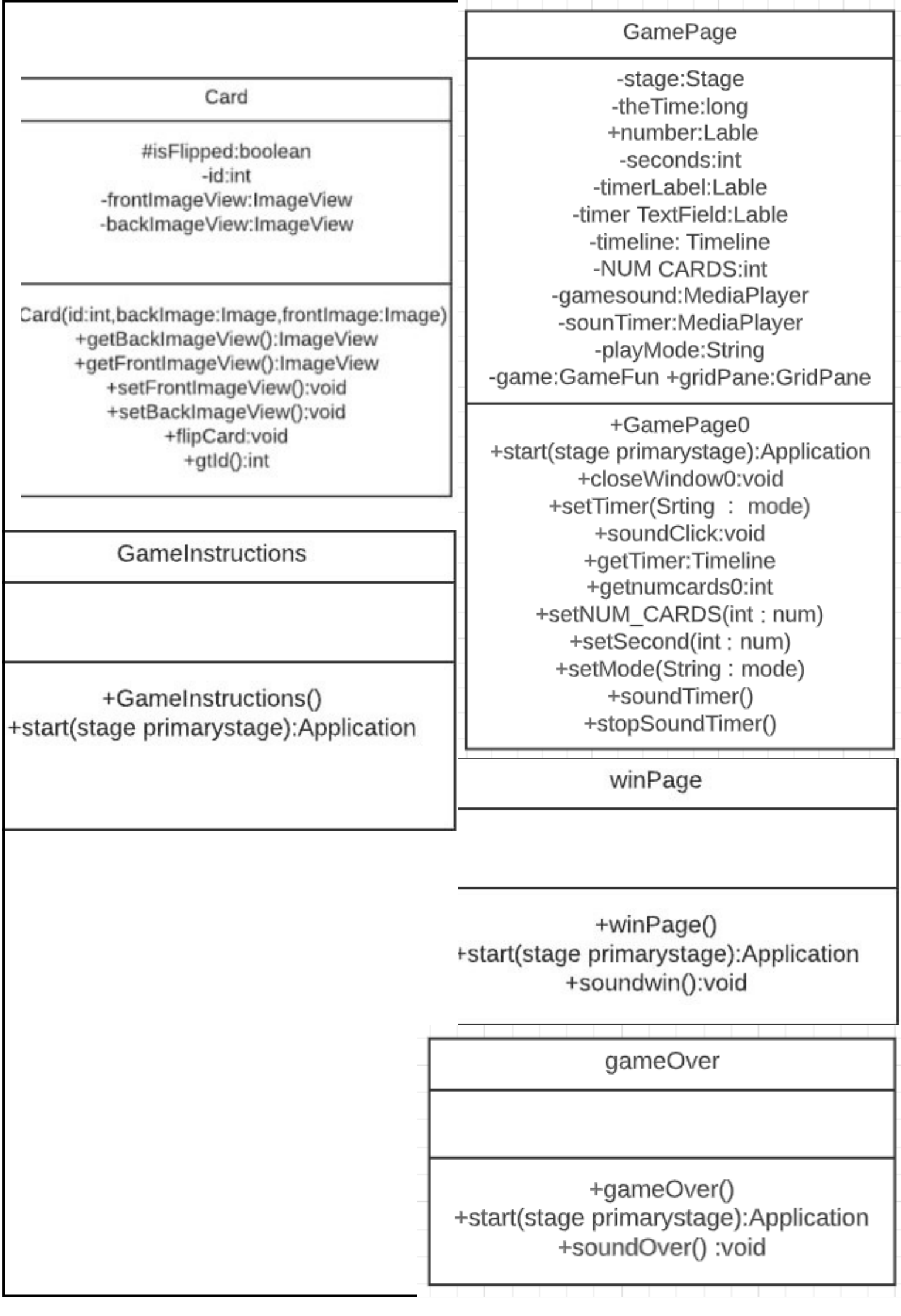


Used libraries

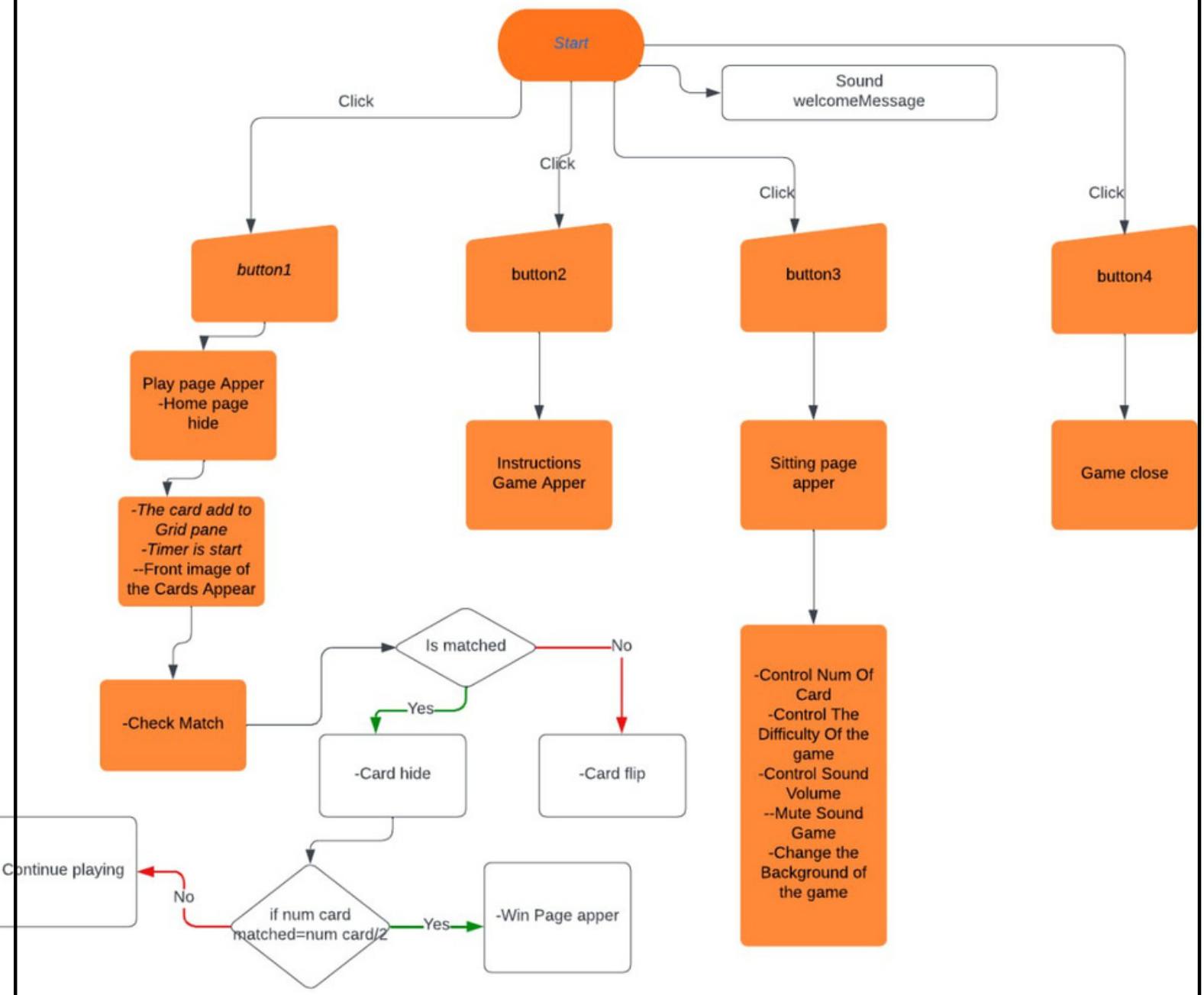
```
package Applecation;

import javafx.scene.paint.Color;
import javafx.application.Application;
import javafx.geometry.Insets;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.layout.*;
import javafx.stage.Stage;
import javafx.animation.KeyFrame;
import javafx.animation.Timeline;
import javafx.geometry.Pos;
import javafx.util.Duration;
import javafx.scene.image.Image;
import javafx.scene.text.Font;
import javafx.scene.text.Text;
import javafx.scene.media.Media;
import javafx.scene.media.MediaPlayer;
import java.io.File;
import java.util.logging.Level;
import java.util.logging.Logger;
import javafx.animation.Animation;
import javafx.animation.TranslateTransition;
import javafx.scene.effect.DropShadow;
import javafx.scene.image.ImageView;
import javafx.geometry.Orientation;
import javafx.scene.effect.BoxBlur;
import javafx.scene.shape.Rectangle;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
import java.util.Timer;
import java.util.TimerTask;
import javafx.animation.RotateTransition;
import javafx.animation.SequentialTransition;
import javafx.scene.transform.Rotate;
```

UML Diagram


Homepage
†width:double
+height:double
-startsound:mediaplayer
-porderpane: Borderpane
Homepage()
+start(stage primarystage):Application
+getstartsound(:mediaplayer
+setNightBackGround():void
+setLightBackGround():void
+startSound():void
+soundClick():void
GameSettings
-level1: RadioButton
-level2:RadioButton
-level: RadioButton
-easy:RadioButton
-normal: RadioButton
-hard: RadioButton
-toggleGroup: ToggleGroup
-toggleGroup2:ToggleGroup
+home:Homepage
+gamepage:Gamepage
+gamefun:Gamefun
GameSetting()
-start(stage primarystage):Application
+setMode():void
+setLevel():void
GamePage
-stage:Stage
-theTime:long
+number:Lable
-seconds:int
-timerLabel:Lable
-timerTextField:Lable
-timeline:Timeline
-NUM_CARDS:int
-gamesound:MediaPlayer
-game:GameFun
+gridPane:GridPane
+GamePage()
+start(stage primarystage):Application
+closeWindow():void
+soundClick():void
+getTimer():Timeline
+getnumcards():int
+setNUM_CARDS(num:int)

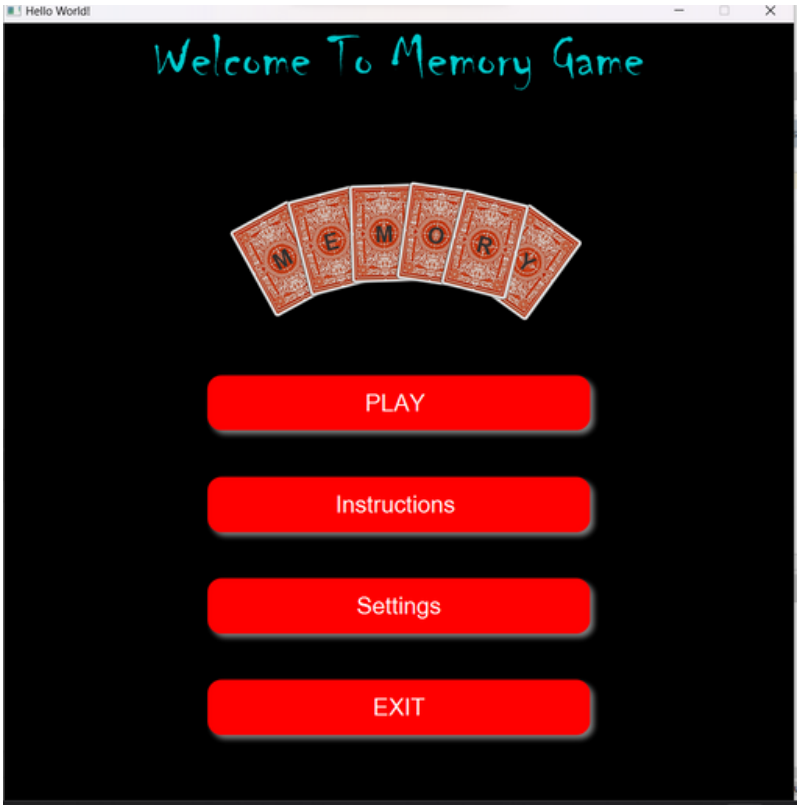

GamePage
-stage:Stage
-theTime:long
+number:Lable
-seconds:int
-timerLabel:Lable
-timer TextField:Lable
-timeline: Timeline
-NUM CARDS:int
-gamesound:MediaPlayer
-sounTimer:MediaPlayer
-playMode:String
-game:GameFun +gridPane:GridPane
+GamePage0
+start(stage primarystage):Application
+closeWindow0:void
+setTimer(Srting : mode)
+soundClick:void
+getTimer:Timeline
+getnumcards0:int
+setNUM_CARDS(int : num)
+setSecond(int : num)
+setMode(String : mode)
+soundTimer()
+stopSoundTimer()
Card
#isFlipped:boolean
-id:int
-frontImageView:ImageView
-backImageView:ImageView
Card(id:int,backImage:Image,frontImage:Image)
+getBackImageView():ImageView
+getFrontImageView():ImageView
+setFrontImageView():void
+setBackImageView():void
+flipCard:void
+gtId():int
GameInstructions
+GameInstructions()
+start(stage primarystage):Application
winPage
+winPage()
+start(stage primarystage):Application
+soundwin():void
gameOver
+gameOver()
+start(stage primarystage):Application
+soundOver() :void

Flow Chart


Start
Sound welcomeMessage
Click
Click
Click
Click
button1
button2
button3
button4
Play page Apper -Home page hide
Instructions Game Apper
Sitting page apper
Game close
-The card add to Grid pane -Timer is start --Front image of the Cards Appear
-Check Match
Is matched
No
Yes
-Card hide
-Card flip
-Control Num Of Card -Control The Difficulty Of the game -Control Sound Volume --Mute Sound Game -Change the Background of the game
Continue playing
No
if num card matched=num card/2
Yes
-Win Page apper

Buttons and its function

- When hovering over the buttons, their color changes, and when leaving them, they return to their original color.

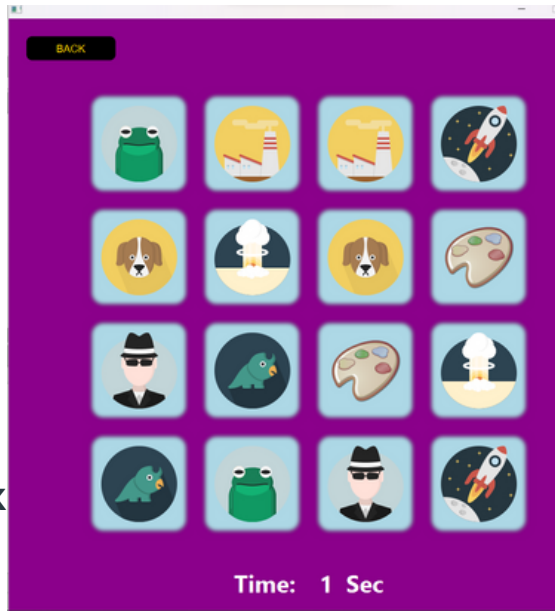


- "Play" button: It mutes the main interface sound and starts the game, opening a new page for the game.
- "Instructions" button: It displays a page that explains how to play and the rules of the game.
- "Settings" button: It displays the game settings page where the player can change some settings such as sound level and the background color of the main page, among others.
- "Exit" button: It closes the game and exits it.

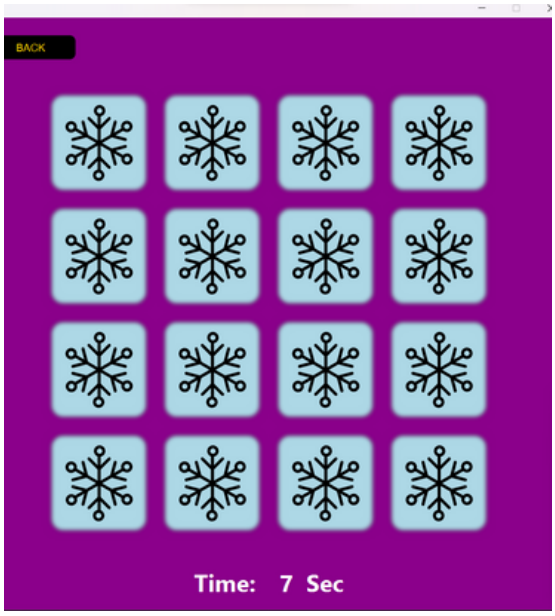
- At the beginning, the first letter appears, and then the rest of the letters appear with a certain delay, and as more letters are added, the delay for the appearance of the next letter increases.

Playing mechanism

- "Cards" are rectangles, each rectangle carries a front image, a back image, and a unique identifier. A copy of each card is created with an opposite identifier to the original image, so that we can control not flipping the same card when clicking on it multiple times.



- The Game starts with displaying the cards in their front image for a certain period of time, then all cards are flipped to their back image.



- Two cards are selected and sent to the check matched function to verify their match. If the same card is clicked twice, it will be excluded and not counted as the second card. If the two cards match, they will be hidden and removed from the group. If there is no match, the cards will return to their back image and the game continues until all matching cards are collected. Then, the winning page will appear, displaying the number of attempts used and the time taken to collect all the cards after flipping them.

Sitting page and its function

1

- The function "setLevel" changes the number of cards based on the value selected by the player through choosing the appropriate radio button.

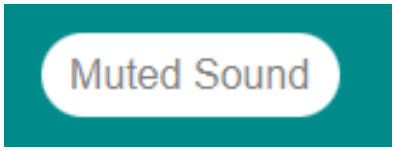
2

- The function "setMode" changes the time that the cards are displayed in their front image at the beginning of the game, before flipping them to their back image.
- If the "hard" level is selected, a countdown timer will be activated, and if it reaches zero, the player will lose.

3

- It allows volume controlling the starting sound level, which plays on the home page.

4



- It is a toggle button that mutes the starting sound on the home page, and when clicked again, it plays the sound again.

5

- It allows controlling the background color of the starting page, whether it is white or black, based on the player's choice of one of the available radio buttons.

Challenges:

- Creating cards with unique front and back images for each card.
- Implementing card matching logic and determining the appropriate comparison method (solved by creating a unique ID for each card).
- Handling the issue of clicking the same card multiple times being considered as a match (solved by creating an opposite ID for the copied card).
- Linking the settings page to the game and making it affect the game directly.

Potential Enhancements:

- Adding multiple difficulty levels to the game.
- Adding a limited number of attempts for hard mode, where the player loses if they exhaust them all.
- Adding game modes, including the ability to challenge a computer opponent by creating a bot for random selection.
- Adding online multiplayer capability to challenge other players.

Thank you