

COSC2659 IOS Development

Assignment 2 Report

IOS Game: PikachuGame

Lecturer: Tom Huynh
Student: Nguyen The Bao Ngoc
Student ID: s3924436

Table of Contents

A. Introduction	3
B. Motivation	3
C. Game rules.....	3
D. Required Features	5
Main Requirement.....	5
1. Menu View (Welcome View):	5
2. Leaderboard View:.....	6
3. Game View:.....	7
4. Game Settings View:.....	9
5. How To Play View:.....	10
6. Sound and UI Requirements: (The following features will be demonstrate in the YouTube video).....	10
7. Device Compatibility:.....	11
Advanced Requirement	12
1. Game Progression and Levels:	12
2. Multiple Language Support:	12
3. Toggle Theme Setting:.....	13
E. Appendix.....	14
F. Link	26

A. Introduction

In this project, I am developing a simple 2D game which is called PikachuGame. This was a famous game for Vietnamese people in 2010s. The reason of its fame was that the game is really simple and relaxing. It also contains several of cute Pokemons in the Pokemon anime series which is a well-known series all around the world. The goal of this project is to demonstrate my programming skill in Swift and SwiftUI. Moreover, I also further developing my coding thinking.

B. Motivation

The reason why I chose PikachuGame is that I am also a big fan of Pokemon series especially Pikachu. When I younger, this is my favourite video game. Therefore, I want to recreate the game and introduce it to everyone. On the other hand, PikachuGame is also a suitable project for the scope of this Assignment. It is a simple 2 D game but the complexity of game logic is enough for developing my coding skill. Therefore, I decided to choose PikachuGame as my project for this Assignment.

C. Game rules

The game rule is really simple and easy to understand. When you start the game, a matrix of Pokemon which has flexible size base on stage will be display.



Figure 1:A 9x16 Pokemon Matrix

Your mission is to clear the matrix as fast as possible by choosing 2 same Pokemon Block. If the connection between the chosen blocks is valid, those blocks will be removed from the matrix. A valid connection is the connection that has no more than 3 edges from the start to the end block.



Figure 2:Valid Connection(1 edge).



Figure 3:Valid Connection(3 edges).



Figure 4:Invalid Connection (4 edges).

After finishing clearing the matrix, the remaining time that you are having will be converted to bonus score and add to your result. Your result will be recorded and display in the leaderboard where you compete with other users to earn the first position.



Ranking	Player	Scores
1.	Ngoc	825
2.	Bin	822
3.	Nick	819
4.	Willia m	810

Figure 5:Leaderboard.

If the player cannot complete the matrix before the timer run out, it a game over.

D. Required Features

Main Requirement

1. Menu View (Welcome View):

When user enter the game, a splash art view will be displayed first:

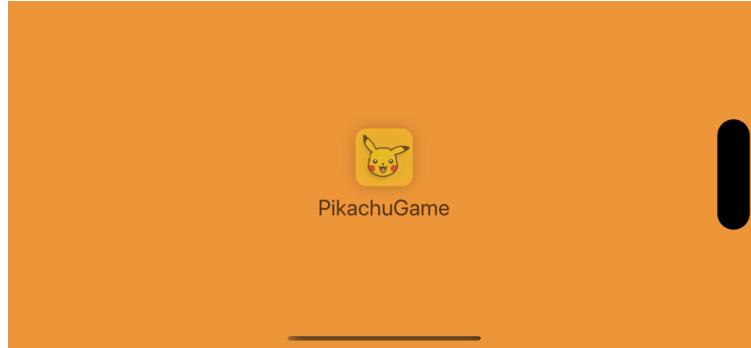


Figure 6: Splash art screen.

The Splash Screen View is a SwiftUI View which is displayed and dismissed using a @State variable naming isSplash. After being display for 2 seconds, it will disappear. Then the game will display the menu view, which contains Buttons to change the value of a @State variable to navigate to Leaderboard View, How To Play View, Setting View and Game View.

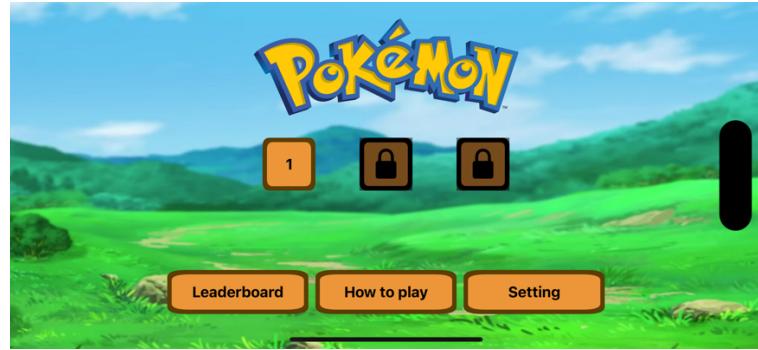


Figure 7:Menu View.

2. Leaderboard View:

The leaderboard view will display all the result score and username of each player on that device.

Ranking	Player	Scores
1.	Ngoc	825
2.	Bin	822
3.	Nick	819
4.	Willia m	810

Figure 8:LeaderBoard View

The list of results is implemented using a List of SwiftUI which get the Result data from Core Data.

By pressing on the Profile Button on the top right corner of the view, A profile sheet will be displayed which contains the detailed information of the current player.



Figure 9:Profile button popup sheet.

Inside the Profile sheet, the game will display player username, matches that they played, the player's winrate, the highest score of that player and the achievement slider. The current achievement contains 3 achievements: Complete stage 1, complete stage 2 and complete stage 3. These achievements will be achieved when player first completing the corresponding stage.

If the users want to get back to the MenuView, they can click on the arrow Button at the top left corner.

3. Game View:

Game View is one of the most important views since it contains the game and the game logic.

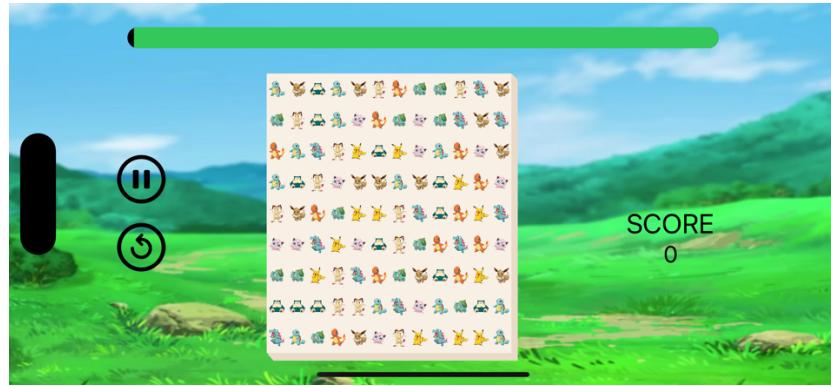


Figure 10: Game View.

This game view contains:

- Timer bar: display the remaining time of the player have. Every time the timer counts down, the timer bar will adjust the remaining time base on the difference between the current and the start time.
- Pause button: pause the game, stop the timer and display a pausing view



Figure 11: Pausing menu.

The pausing view contains button to continue the game or navigate back to the Menu View.

- Shuffle button: button to shuffle the matrix if player cannot find any valid pair.
- The Pokemon matrix: which is a SwiftUI view contains a set of pokemon and generates PokemonBlocks in a Grid base on that set.
- Score: The current score of user.

When the player finished playing, the result of that game is displayed and a button to navigate back to MenuView.



Figure 12: Game Over and Victory Menu.

The user can register their username at the first time they got into the app or they can change it in the Setting View. After player's first register, a variable will be saved into UserDefaults to store the state of the log in. If user want to change player, they can get in to Setting View and change the username.

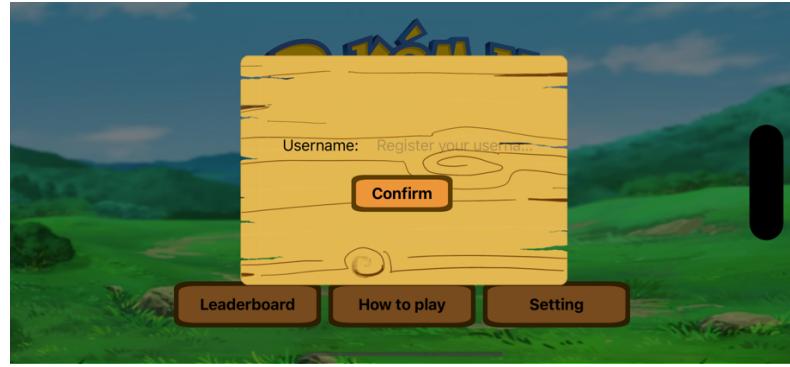


Figure 13: New user register.

4. Game Settings View:

Inside the Setting View, player can adjust the difficulty of the game, change their username, change the language and toggle the theme setting.



Figure 14: Setting View.

Whenever user want to change username, the game will get the input username and check with the PlayerData store by CoreData. If there is a match in username, the current player will be set to that Player data. Else the game will create a new Player and new Player Data.

The difficulty of the game is stored using UserDefaults by a variable naming mode and will be set when user enter a view.

5. How To Play View:

This view will display the detailed rule set of the PikachuGame. This is a SwiftUI View which use a ScrollView to display the rule.



Figure 15: How to play View.

6. Sound and UI Requirements: (The following features will be demonstrate in the YouTube video).

Background Music:

When player navigate to MenuView, LeaderBoard View, HowToPlay View and Game View, different background music will be played. The background music is implemented by using AVAudioPlayer. The player will get background music from mp3 file inside the Sounds folder.

Sound Effects:

The following action of user will trigger a sound effect:

- Pressing a button.
- Choosing a block.
- Successfully matching a pair.
- Fail to match a pair.
- Win a game.
- Loose a game.

These sound effects use another AVAudioPlayer to trigger in order to run at the same time with the background music without getting crash.

7. Device Compatibility:

For the device compatibility, the Game will fit the screen size of following device:

- iPhone 14, 14 Pro, 14 Pro Max

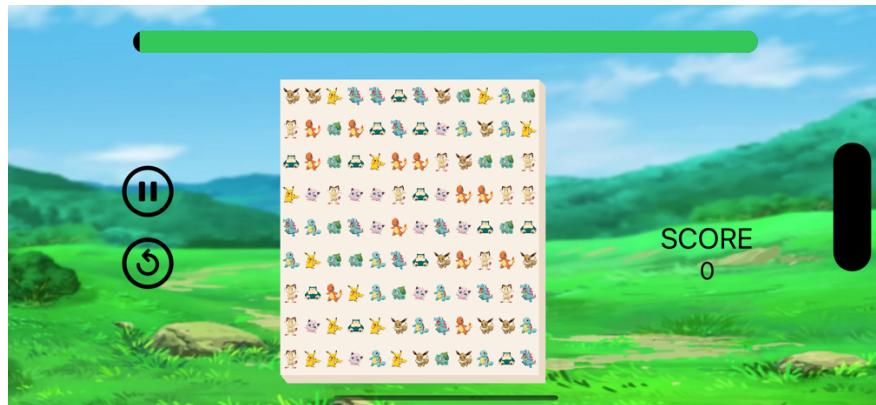


Figure 16:Game View on an Iphone 14 Pro emulator.

- iPad Pro (11 - 12.9 inch)

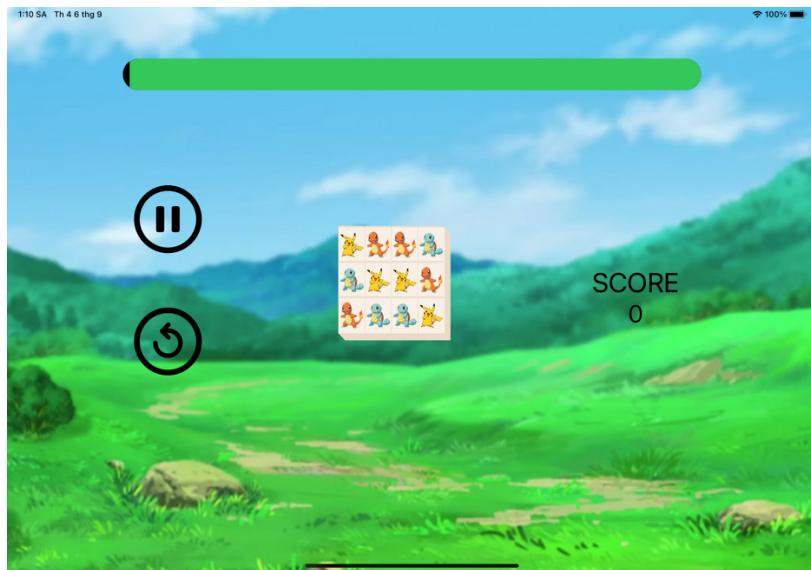


Figure 17:Game View on an IPad Pro emulator.

The theme color will not be changed by the IOS theme, this will be changed by the toggle button in Setting View.

Advanced Requirement

1. Game Progression and Levels:

The game is divided into 3 stages: 1, 2 and 3. As the stage increase, the Pokemon matrix will be increase and new Pokemon will be added into the matrix. Therefore, if you play the highest stage, there will be more cute pokemon and you can earn more point to climb the leaderboard. To implement this feature, the stage number act as a multiplier which will multiply with the rows and the columns of the PokemonGrid.



Figure 18: 3 stages of the game.

2. Multiple Language Support:

The Game support 2 languages: English and Vietnamese. The language of the game will be changed inside Setting View. The language is implemented by using UserDefault to store a variable of Language. Whenever a view is display, a variable of language will be initialized from UserDefault data and define the text that view. Another view with language will be shown in the appendix. ☺☺☺

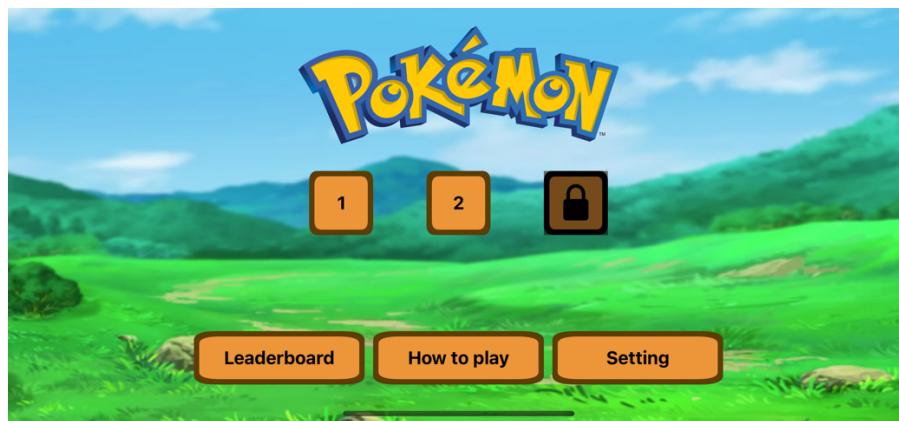


Figure 19:English Menu.

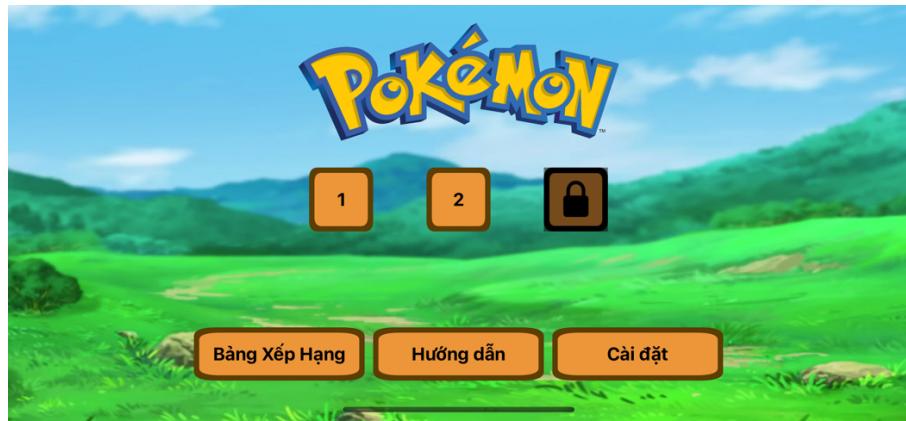


Figure 20: Vietnamese Menu.

3. Toggle Theme Setting:

In side the Setting View, there is a button can be used to toggle between the light mode and dark mode of the app. The UI of the app will vary corresponding to the theme mode. Similar to the language setting, the theme of my game is store in UserDefault and will be extracted to define the theme of every view. Example of darkmode will be found in appendix []

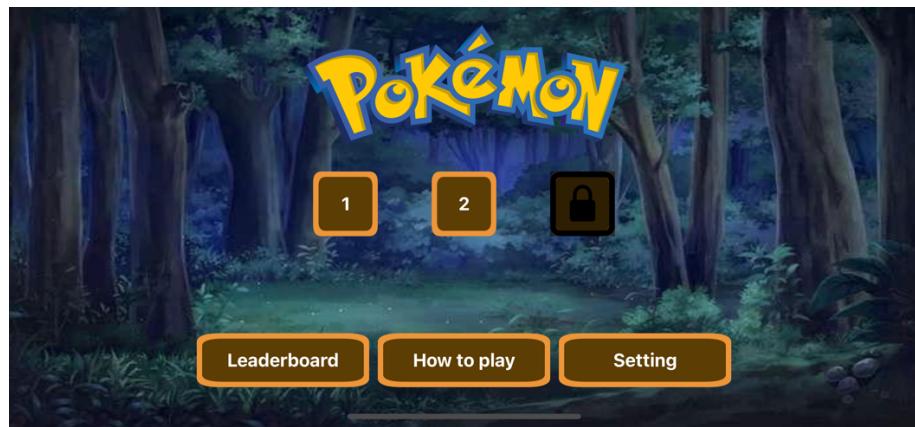
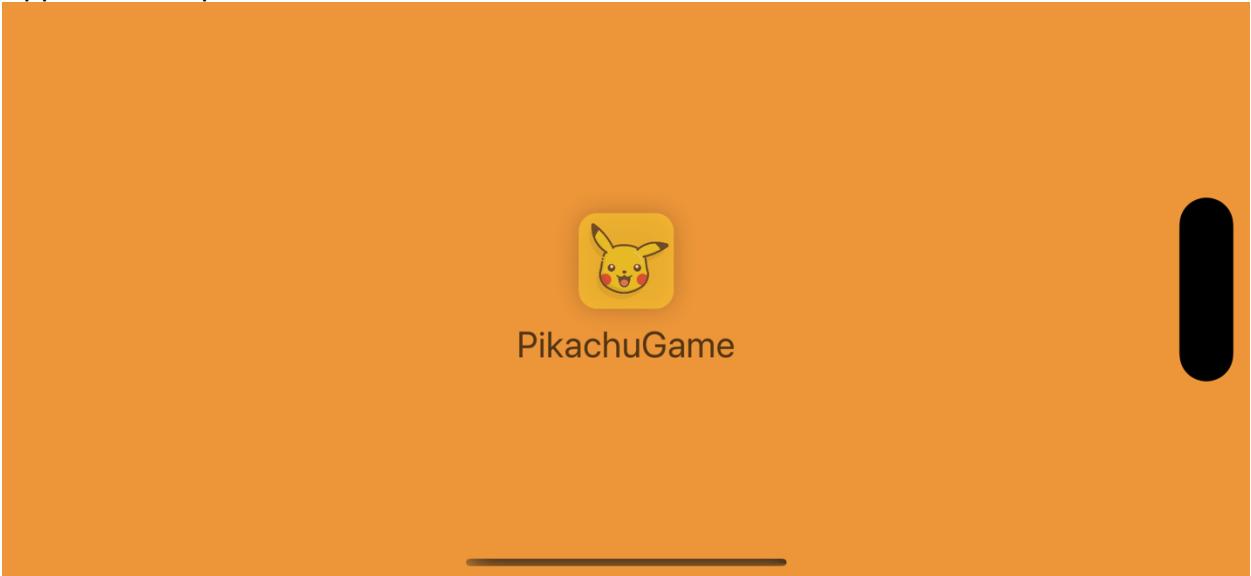


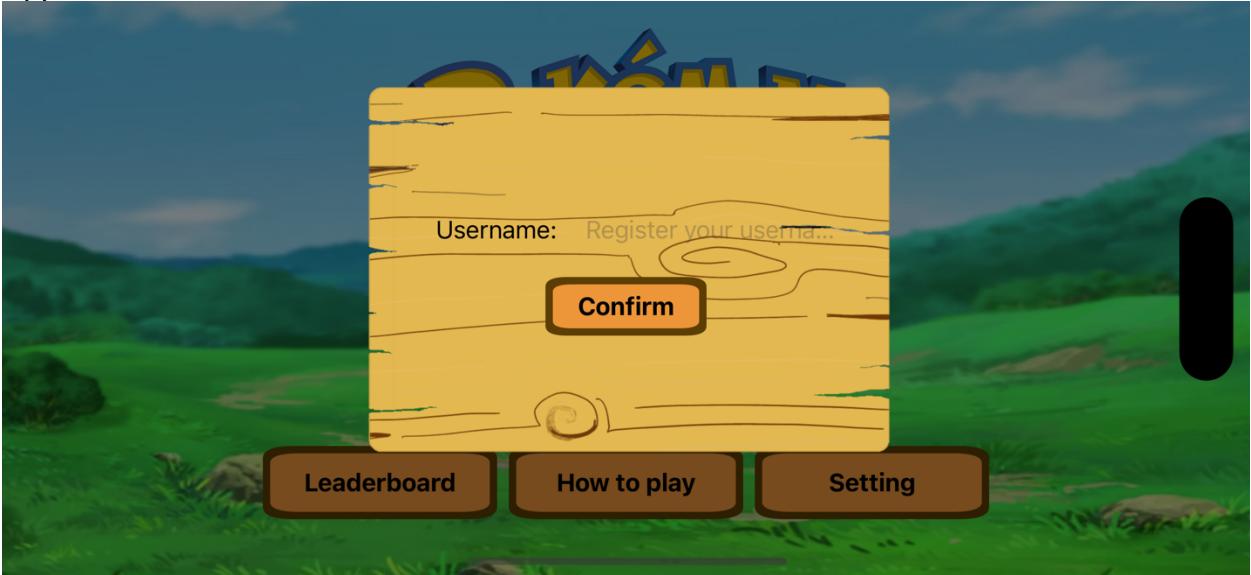
Figure 21: Menu in darkmode.

E. Appendix

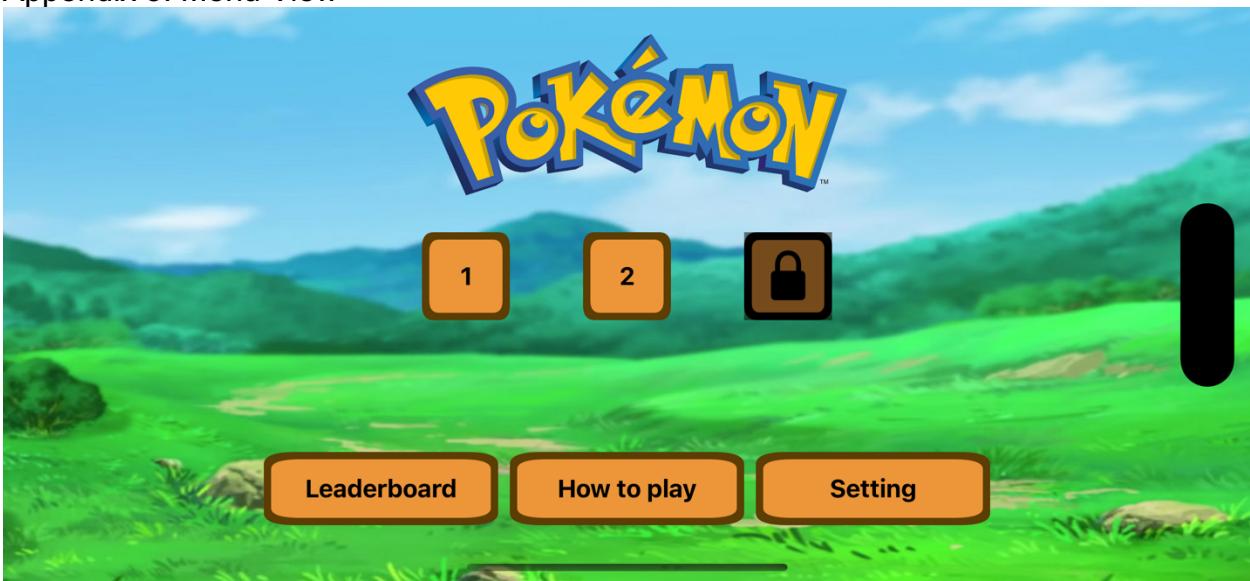
Appendix 1: Splash art



Appendix 2: First load



Appendix 3: Menu View

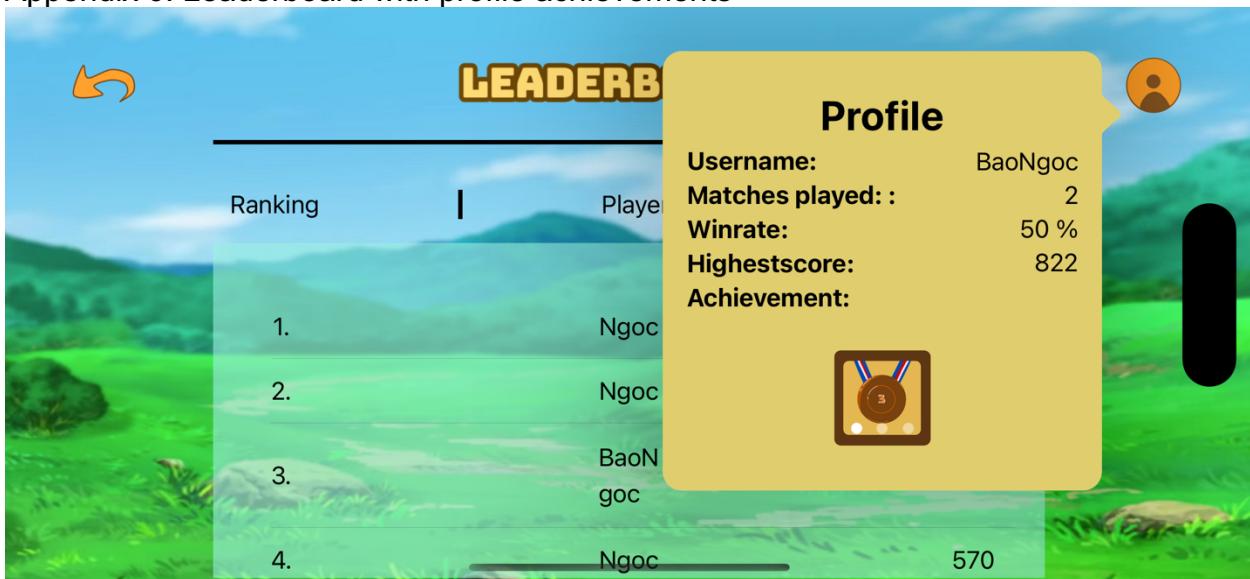


Appendix 4: Leaderboard View

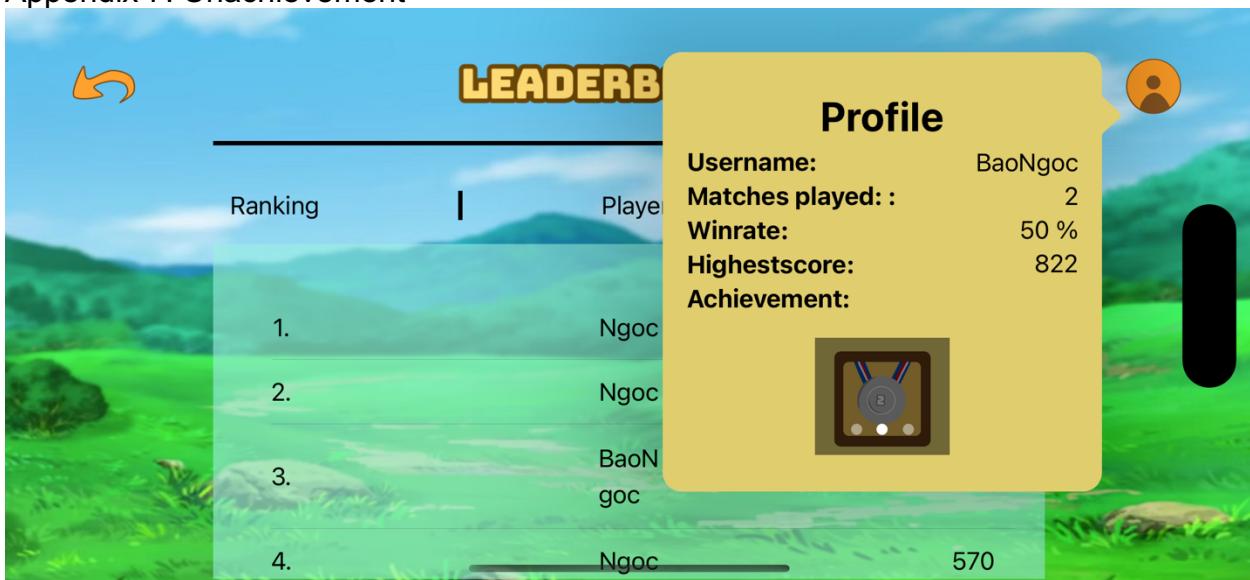
A screenshot of the game's leaderboard screen. The title "LEADERBOARD" is centered at the top. On the left is a back arrow icon, and on the right is a user profile icon. The table has three columns: "Ranking", "Player", and "Scores".

Ranking	Player	Scores
1.	Ngoc	1,434
2.	Ngoc	825
3.	BaoN goc	822
4.	Ngoc	570

Appendix 6: Leaderboard with profile achievements



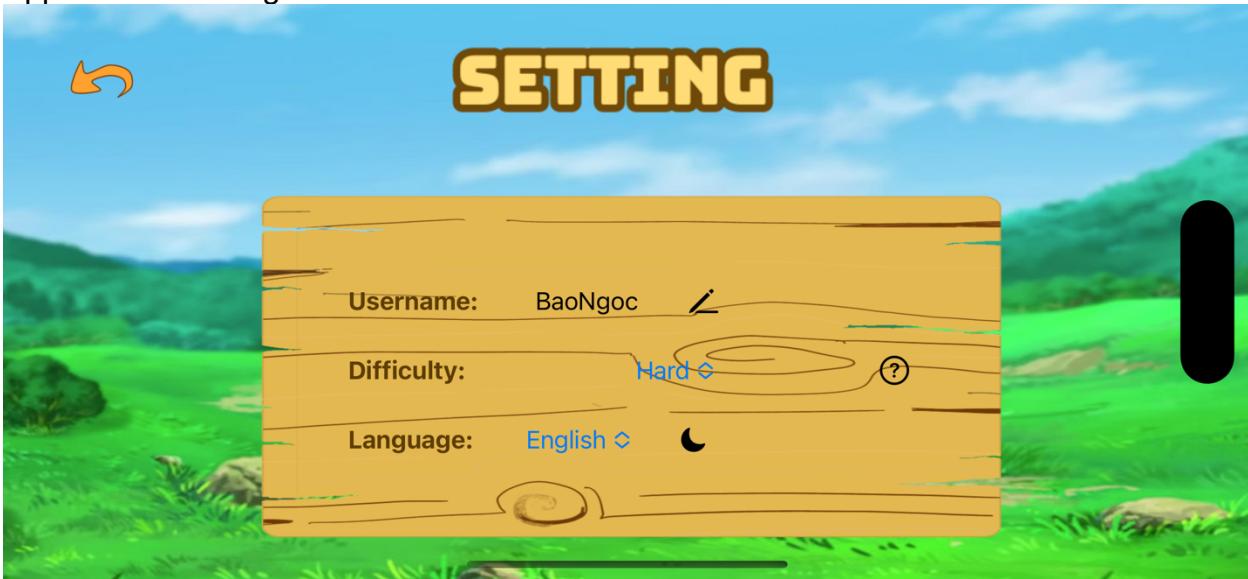
Appendix 7: Unachievement



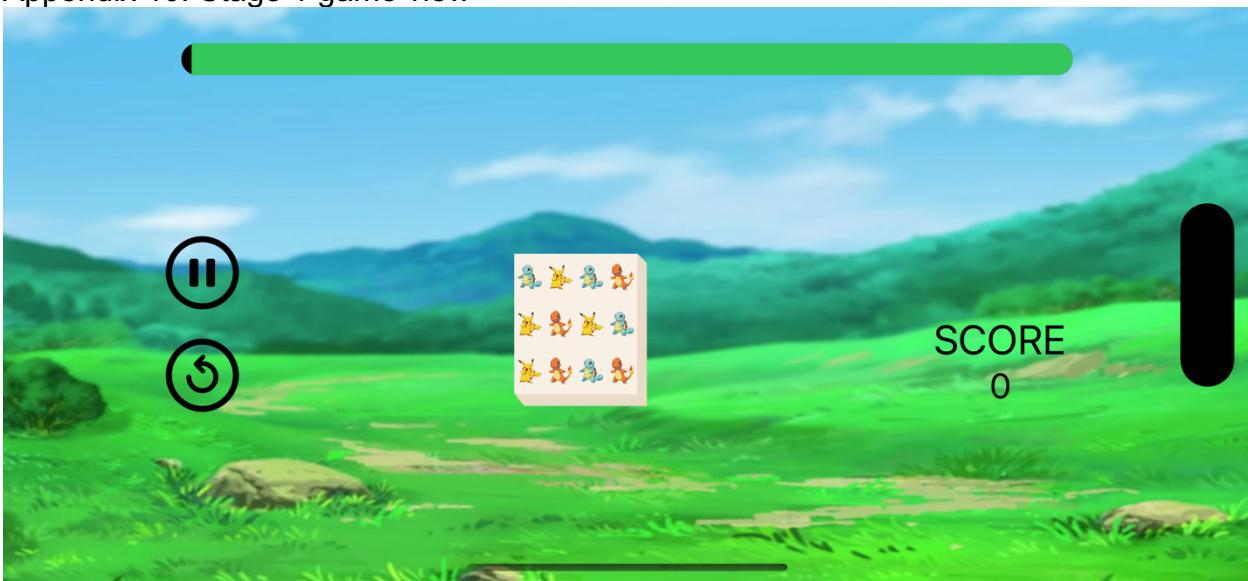
Appendix 8: How to play view.



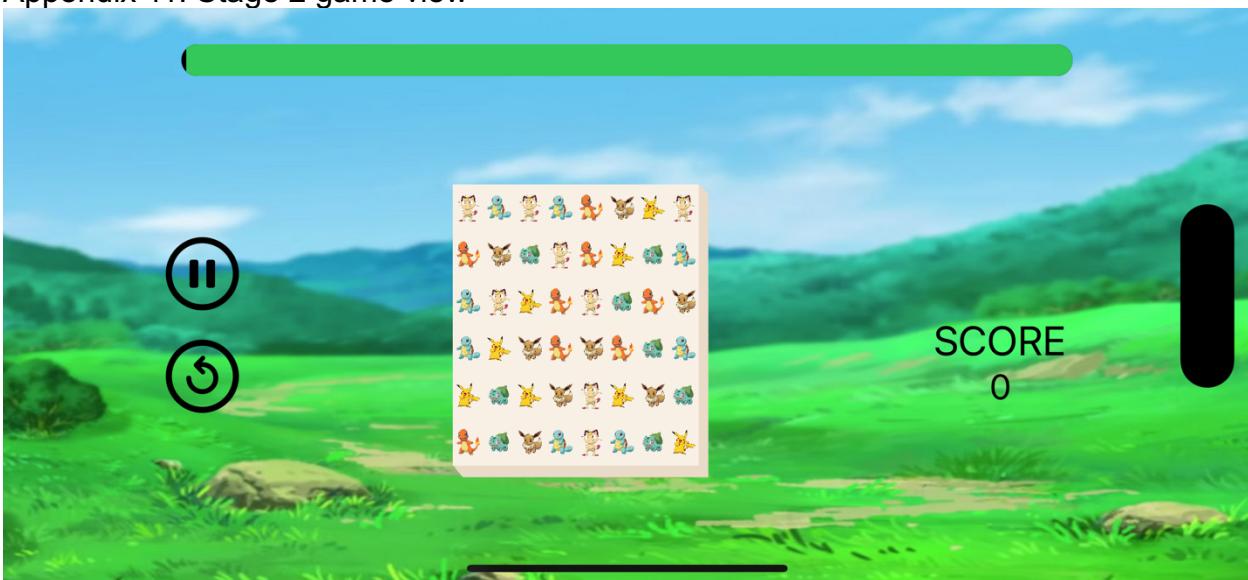
Appendix 9: Setting View



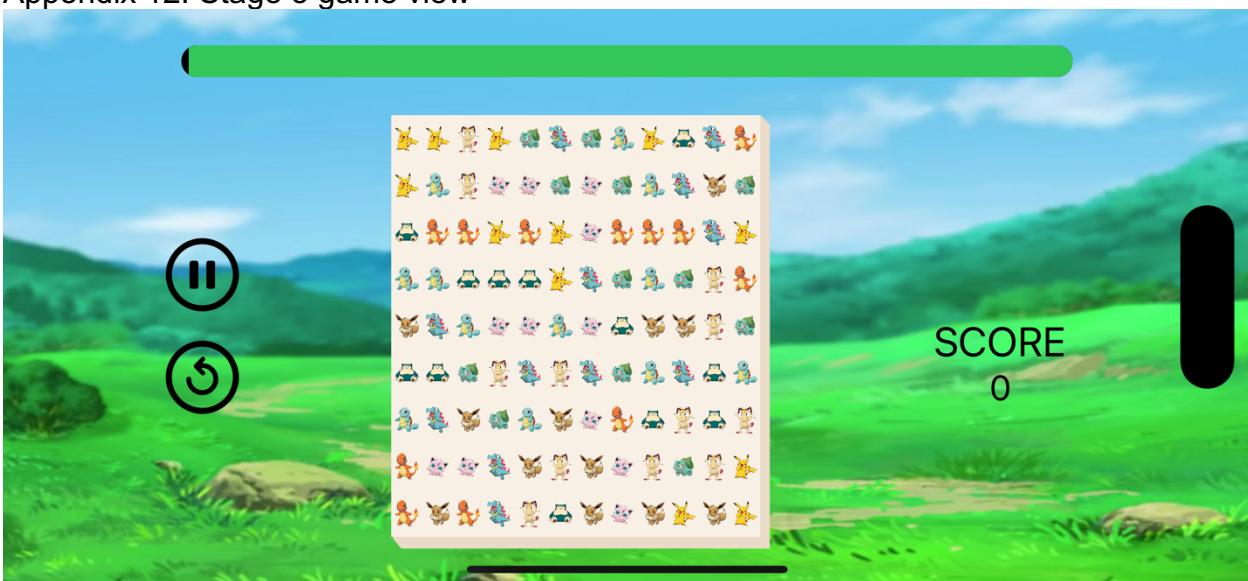
Appendix 10: Stage 1 game view



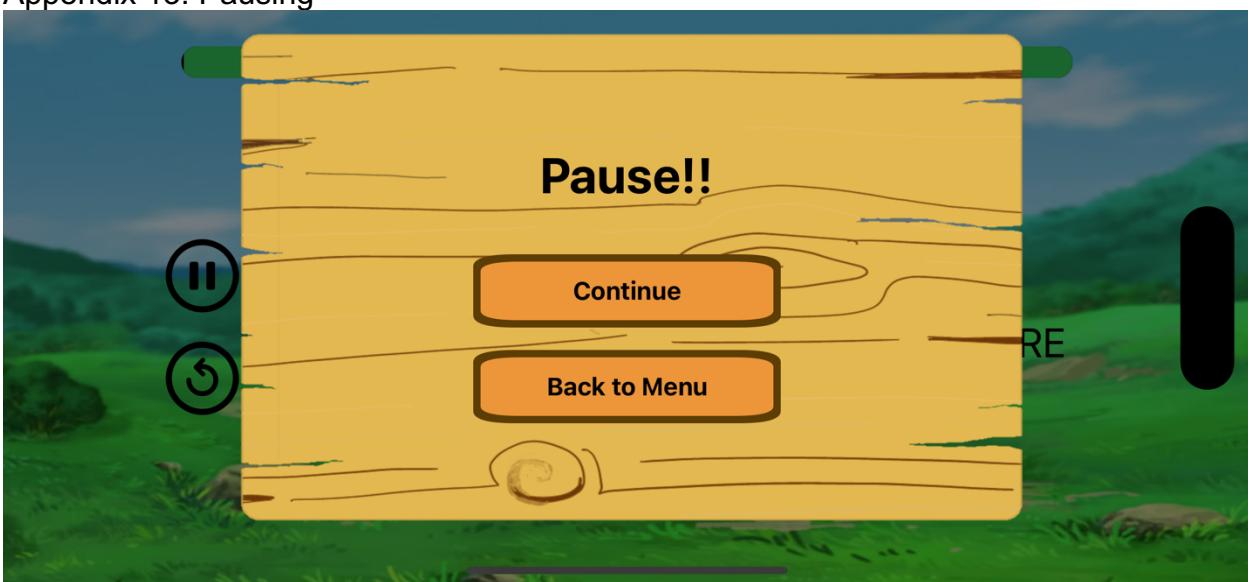
Appendix 11: Stage 2 game view



Appendix 12: Stage 3 game view



Appendix 13: Pausing



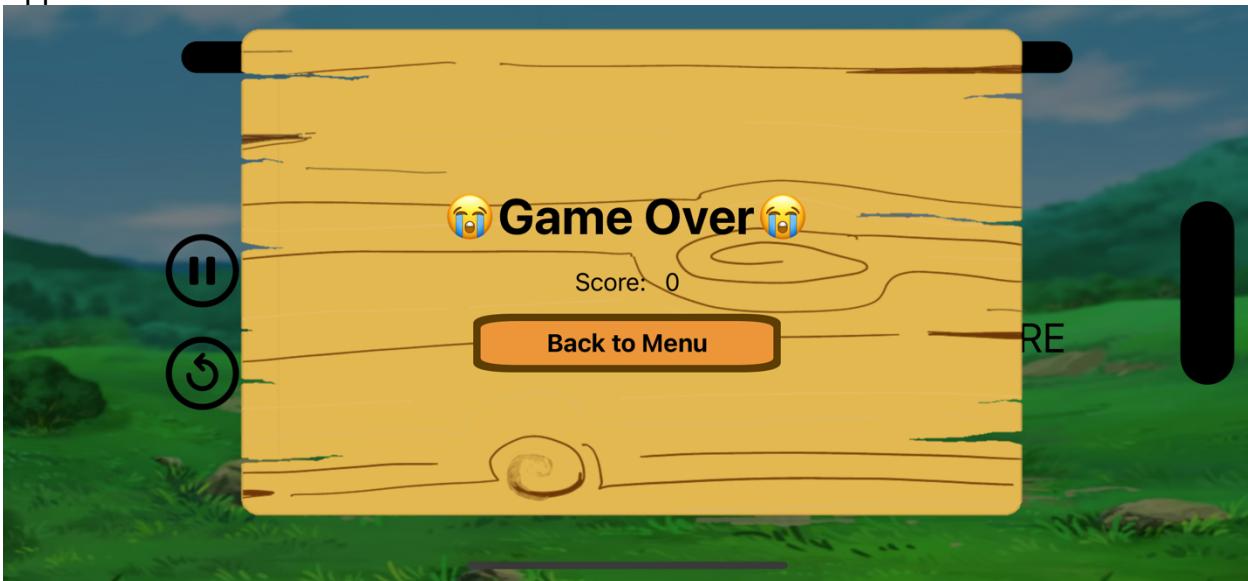
Appendix 14: Victory with new highscore



Appendix 15: Victory without highscore



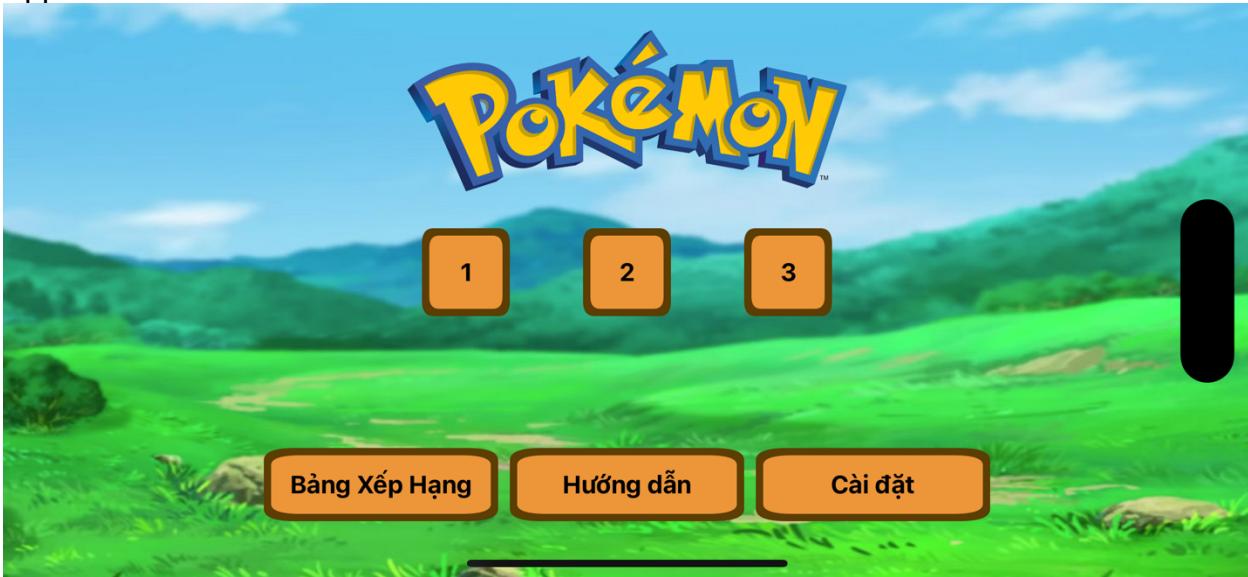
Appendix 16: Game over



Appendix 17: New achievement



Appendix 18: Vietnamese Menu



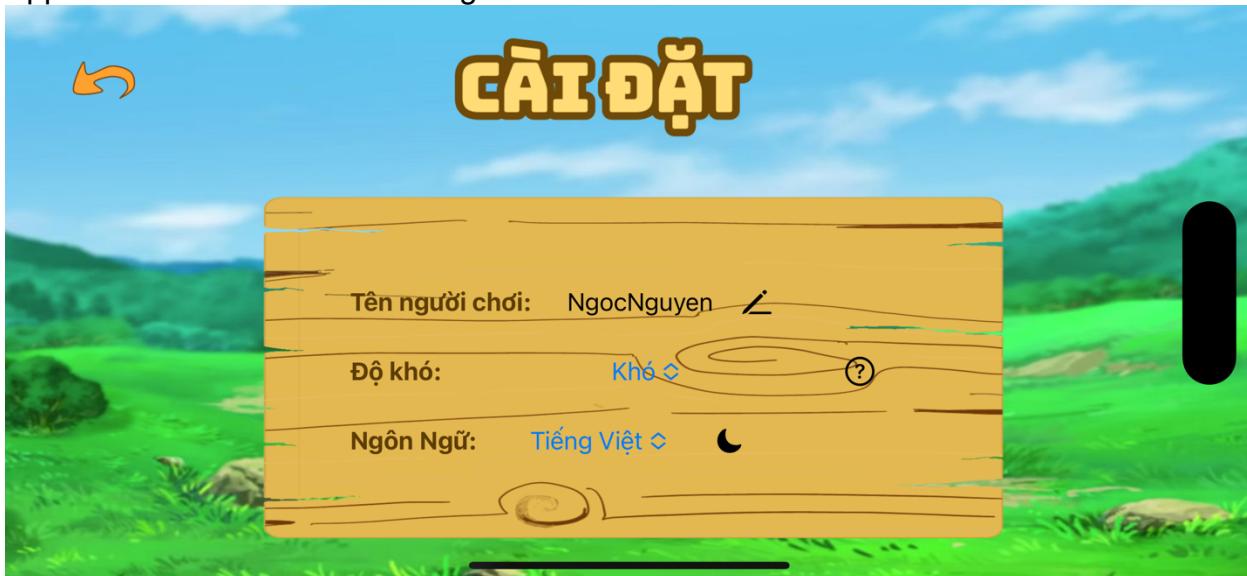
Appendix 19: Vietnamese Leaderboard

Xếp Hạng	Người Chơi	Điểm
1.	Ngoc Nguyen	1,641
2.	Ngoc	1,434
3.	Ngoc	825

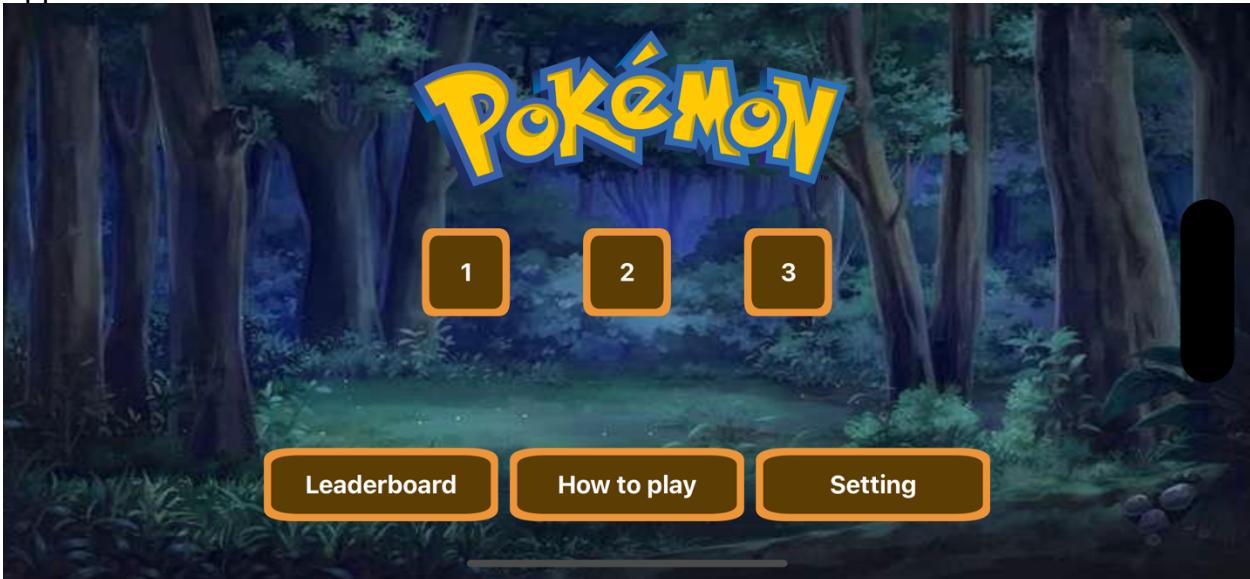
Appendix 20: Vietnamese How to play



Appendix 21: Vietnamese Setting



Appendix 22: Dark mode Menu



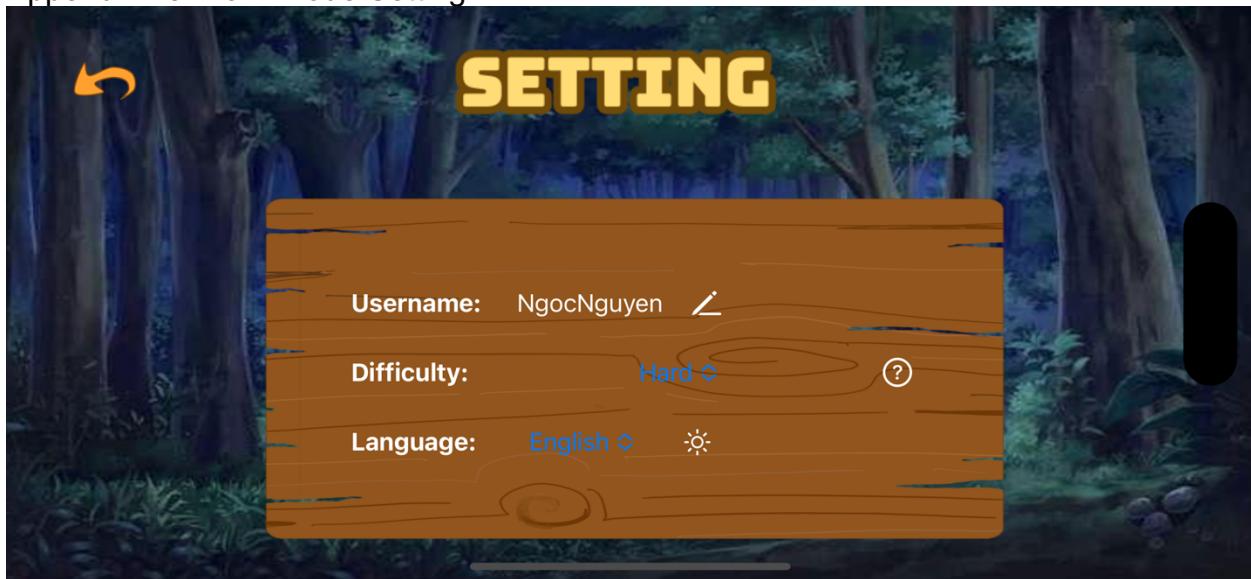
Appendix 23: Dark mode How to play



Appendix 24: Dark mode leaderboard



Appendix 25: Dark mode Setting



F. Link

Youtube video demonstration: <https://youtu.be/uuwnnieq04s>

Github repo: <https://github.com/Ngoc0123/PikachuGame.git>