#### Gliondar



Emma Grealy C16304876

Submitted in partial fulfillment of the requirements for the degree of B.Sc in Business Computing

Technological University of Dublin

DT 354/4

Supervised by: Neil O’Connor

May 2020

This is an original work. All References and assistance are acknowledged.

Signed: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Table of Contents

[Introduction 5](#_Toc39215281)

[Gliondar Objectives 6](#_Toc39215282)

[Registered User: 6](#_Toc39215283)

[Anonymous User: 6](#_Toc39215284)

[Business Case 7](#_Toc39215285)

[Business Rules/Assumptions 8](#_Toc39215286)

[Business Actors 9](#_Toc39215287)

[Customer Account 9](#_Toc39215288)

[Anonymous User 9](#_Toc39215289)

[Functional Requirements 10](#_Toc39215290)

[Non-Functional 12](#_Toc39215291)

[Usability: 12](#_Toc39215292)

[Security: 12](#_Toc39215293)

[Scalability: 12](#_Toc39215294)

[Performance: 12](#_Toc39215295)

[UseCase Diagram 13](#_Toc39215296)

[ER Diagram 14](#_Toc39215297)

[Technologies Used 15](#_Toc39215298)

[Unity Run Through 20](#_Toc39215299)

[UI Design/ Layout 23](#_Toc39215300)

[Opening Scene: Register/Login 23](#_Toc39215301)

[Next Scene: Start Screen 23](#_Toc39215302)

[How to Play Scene 24](#_Toc39215303)

[Game Selection Scene 24](#_Toc39215304)

[Game Scenes 25](#_Toc39215305)

[Leader Board 26](#_Toc39215306)

[Player Profile 26](#_Toc39215307)

[Database Layout and Design 27](#_Toc39215308)

[Test Plan 28](#_Toc39215309)

[Obstacles faced 30](#_Toc39215310)

[Unity 30](#_Toc39215311)

[Github 30](#_Toc39215312)

[Conclusion 31](#_Toc39215313)

[References 32](#_Toc39215314)

Acknowledgements

I would like to thank everyone who has helped me not only through my final year, but throughout the entire duration of this course. It was a long and tedious process and I am certain I wouldn’t have been able to do it without the support system I have around me.

I would firstly like to thank my family who have always been a constant source of support and encouragement. The regular checking in and reassurance is something I will always remember. I would also like to add a special thank you to my parents for all their hard work to ensure I could go to college by paying me fees. I will be forever grateful for this.

I would also like to thank my supervisor Neil for his constant guidance and support through this entire process. The regular meetings and assistance will never be forgotten. I would also like to thank my second reader Thoa who offered valuable guidance and opinions on different aspects of my project.

I owe a special thanks to my friends from outside college and boyfriend Aaron who always had my best interest at heart. They understood the sacrifices that needed to be made to ensure I did as well as I could during this process. I will never forget the late night walks and face time calls to help as best they could, even if they didn’t fully understand what I was talking about.

Thank you to my fellow classmates who were always friendly and willing to help or just chat after a though assignment or exam. I owe endless thanks to my classmate and friend Tomi. Without Tomis support and guidance through these four years I would not be where I am.

# Introduction

My idea for this project stemmed from a few different experiences when it came to learning Irish or any language really. While I was in school the focus in Irish lessons was to learn answers off by heart, reproduce them, and receive a good grade. There was no focus on the continued use of the language or ensuring that we knew what it was we were saying.   
When I tried to use Duolingo to improve my Irish, I found their approach to be restrictive. You had to learn the phrases they chose and cover the topics they have set up for you. Although the app is well thought out and structured I felt as though it was still a forced learning environment for the users. They user had to cover topics and phrases they may never use or say.  
In Ireland we also have a larger number of tourists who come to Ireland every single year (11.2million in 2019), and there is an extremely small number of those that can speak any Irish, even to just say some simple words or phrases, or to even read some of our road signs!  
When creating my app I wanted to try and create a way for a person at any level of experience to learn in a fun way. Allowing people to learn about topics or areas that they wanted to and believed they would use. I thought if I made learning fun, and allowed the user to have a say in what they learn, and allow them to learn at their own pace it would encourage them to keep learning and developing their language skills.  
My aim with this application is to promote conversational Irish. To promote the application users to speak Irish, to a standard they choose, as much as possible. By allowing the user to hear Irish more and more while playing a fun game believe it will help encourage consistent use of the app resulting in the growth of conversational Irish.

# Gliondar Objectives

The overall objective for this app is to help the user learn conversational Irish in a fun new way. They will be hearing the Irish and seeing images that relate to the word they hear to help them learn by both hearing and seeing. I chose to include images as opposed to words in this application as I did not want spelling to be the focus. They will learn to speak Irish as opposed to learning off reams of grammar. Simple conversational topics to promote the continued use of the language, not forcing anyone to learn off answer or spend time learning about topics they will never discuss on a daily basis.  
There are two types of users for this game, a registered user and an anonymous user.

## Registered User:

* Register and Login to their account.
* Play a demo version of the game to understand how the game works.
* View a selection of game topics to learn about.
* Click on a particular topic and play the game.
* Have their score for this game recorded.
* Replay the game and try improving or exiting and trying a new game.
* They can view a leader board to see how they are doing compared to other users.
* They can view their own user profile to track their progress.

## Anonymous User:

* View a selection of game topics to learn about.
* Click on a particular topic and play the game.
* Replay the game to try improving or exiting and trying a new game.
* No scores or records are ever kept for this user.

# Business Case

Gliondar is directly marketed at anyone who wants to try and learn to speak Irish. Your experience level is irrelevant. You could be a tourist coming to the country for the first time, or someone who wants to brush up on their spoken Irish skills. You can learn about any topics you choose to learn about, see your progress compared to other application users and also keep a record of your progress as you go.

The game itself is not meant to be difficult to play; the focus is on the learning aspect. I want the user to be able to replay the game and learn as they play. I did not want to create an ‘online learning’ feel, I wanted to ensure the user could play it on a bus, or sitting in a waiting room. By playing regularly the user will learn while also enjoying their time playing the game and seeing their score improve.  
My aim for the leader board is to encourage a user to keep trying to do better. In schools a grade was final and you could not improve or try again. With my game the user can take as many attempts as they need till they are satisfied they have improved. I believe the ability to learn from mistakes, improve and try again is a necessary approach for learning any language.

# Business Rules/Assumptions

* A user does not have to make an account, but they do if they want to keep a record of their scores and monitor their improvements.
* A user can choose to play a demo version of the game to learn how the game works before jumping straight into the application.
* A user does not need to play every game available to them. They can play whatever games they see as valuable to their learning.
* A user can replay a game as many times as they feel they need in order to learn the words or the phrase in that level.
* A user can see a leader board to see how fellow users are doing and to help motivate them to keep working and learning.
* A user can view their own profile to see personal information and also see their total points earned while playing these games.

Requirements

## Business Actors

There are two types of users for this account. There is a registered user and an anonymous user. If you wish to access all functionality you need to be a registered user.

## Customer Account

If a customer wants to access full functionality that this application has to offer, they must create an account. Their account must have a unique email and a password. Once the customer is logged in, they are met by the game selection screen. Here they can access the variety of topics on offer. The customer can click on a topic and play the game for that topic. When they complete the game they can chose one of two options. They can replay the game and try improving their score, or they can exit back to the game selection screen. If they go back to the game selection screen they will be given the chance to view the leader board. The leader board shows the top scoring app users. The leader board is set up to show the overall top scorers, but you can view based on level if you choose.

## Anonymous User

An anonymous user is able to access all the games available on the application. However, their scores are never recorded, so they will not be able to record their progress or compare it with any registered users.

# Functional Requirements

Customer Functional Requirements

|  |  |  |
| --- | --- | --- |
| Function | Requirement | Justification |
| Register & Login | All customers must be able to register and log into their own accounts in order to access the full functionality of the application. | Customers should have an account to log into so that they can save their scores and compare their results with fellow users |
| Start Scene | The customer is given three options. “Start” allows them to see all the games on offer. “HowTo” is a quick demo of how the games work. “Quit” closes the application. | The user should be given the option to learn how to play the game before starting, but a forced tutorial has a tendency to be skipped. By making it an option to learn how the game works, users may choose to learn how to play, or take the trial and error approach. |
| Game Picker | The user is given a variety of topics to choose from. This screen displays word games, but they can choose to go to a new scene which allows them to learn phrases/sentences. They can also opt t see the leader board | The user is given the power to choose what they learn about. They can start off easy and play word games, or learn phrases or sentences. This is allowing the user to advance their skills when they are ready |
| Games | The user sees number of images on the screen. They then head a word in Irish, and must attempt to match it to the correct picture. When they drag the player to a picture they will hear the word for that picture to know if they are right or not, and their score is recorded.  The same approach applies for the phrases games but they will have English sentence at the top of the screen and collect the Irish words in the right order. | The aim of the games is to allow the user to hear Irish and match it to what they believe is the correct answer. By using images and not words it also allows the user to not worry about spelling or grammar as the focus is learning to speak Irish, and to be able to have a conversation in Irish. |
| Leader Board | The user can come here to view all the users with the highest scores overall. | This is to help encourage the user to see other people using the app and doing well, to help encourage them to do the same. |
| User Profile | The user can come here and view their personal information. | The player profile holds information about the player such as their email and scores on levels they have played. |

# Non-Functional

Usability: The aim for the app is that it provides a fun learning experience for those who want to learn to speak Irish. The app helps the user to hear Irish and match it to images or sentences/phrases to help them to be able to speak Irish more in their daily lives. It is crucial that the app is easy to navigate through and use so that anyone can use it. The game itself is also not difficult to learn to play so that the focus can be on the learning rather than a highly skilled game play. I think that while developing this app I have achieved that, from the interface, to the buttons and the game itself.

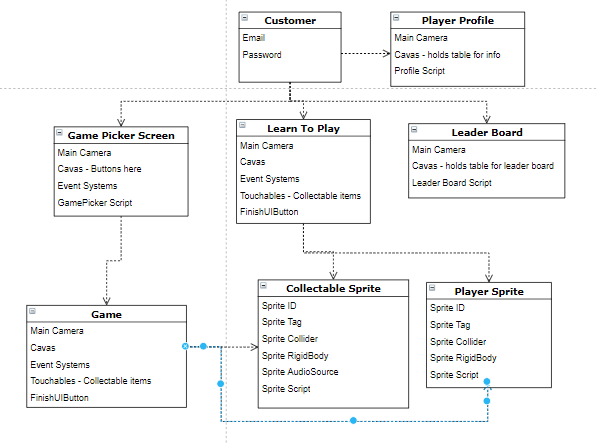
Security: It is essential for anything involving people’s information to have levels of security. Although there is not a huge amount of important user information within the app, it has been developed so only those with levels of authority can access other user’s information. The app is a safe environment where user information is cannot be compromised.

Scalability: The app’s underlying database is Firebase which allows for realtime data updates for large amounts of data. The app would need to be able to be used by large amount of people at a given time.

Performance: The leader board feature of this app requires it to be run in realtime. Scores need to be updated after a game in order to allow a user to see how well they are doing compared to other users and to see their own progress.

# UseCase Diagram

# ER Diagram



# Technologies Used

Below are the main technologies that I have used throughout the creating of my app. I have provided a brief explanation as to what they were used for and I have also provided a detailed explanation of Unity further on in this documentation.

|  |  |
| --- | --- |
| unity.png | **What is Unity?**  Unity is a cross-platform game engine developed by Unity Technologies. It is the platform that I used to create my project. It has high graphics for game development and it allows you to add the functionality to each feature of the application. |
| **How I used it:**  My project was created completely using unity as I developed it. I chose to use unity as it allowed me to create an android application while also offering me high quality graphics and game development tools. It provided me with the tools required to create and deploy a high quality gaming application.  I had to add certain extensions or configuration files for, example I had to add configuration files for Firebase in order to connect my application with my database. | |

|  |  |
| --- | --- |
| Image result for firebase | **What is Firebase?:**  Firebase is a realtime database that I thought would benefit the app and allow the users to access information easily. It gives you functionality like analytics, databases, messaging and crash reporting so you can move quickly and focus on your users. |
| **How I used it:**  The database that I used for my project is Firebase. I used firebase to authenticate the users of my app, store their information, perform security features and access information such as users’ scores.  I chose firebase as it was a real time database, so that scores could update automatically and it provided security features for user’s information. | |

|  |  |
| --- | --- |
| C Sharp (programming language) - Wikipedia | **What is C#?:**  C# is an object-oriented programming language from Microsoft that aims to combine the computing power of C++ with the programming ease of Visual Basic. |
| **How I used it:**  C# is the language used by unity to create scripts. Scripts are used to allow you to trigger game events, modify Component properties over time and respond to user input in any way you like. I created scripts for things such as the player (the movements and actions), the leader board (accessing firebase database) and each individual level, just to name a few. | |
| Visual Studio Code - Wikipedia | **What is Visual Studio Code?:**  Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. |
| **How I used it:**  Visual Studio Code is where I coded up all my scripts in C# for my project. I created all my controls, level settings and sprite controls here. VS Code has features such as syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more which I have always found very helful. | |

|  |  |
| --- | --- |
| https://upload.wikimedia.org/wikipedia/commons/thumb/4/45/The_GIMP_icon_-_gnome.svg/1200px-The_GIMP_icon_-_gnome.svg.png | **What is Gimp?:**  Gimp is an open-source raster graphics editor used for image retouching and editing, free-form drawing, converting between different image formats, and more specialized tasks. |
| **How I used it:**  I used gimp to edit images, allowing me to turn them into sprites. I needed to remove all the backgrounds on any image I chose to use and then export it as a png file so that unity accepted it as a sprite. | |

|  |  |
| --- | --- |
| Audacity (@getaudacity) | Twitter | **What is Audacity?:**  Audacity is an open-source digital audio editor and recording application software. |
| **How I used it:**  I used Audacity to record all of my audio for this application.  It allowed me to export my recordings as audio sources in order to be accepted in unity. I could edit and crop all my recordings within this application. | |

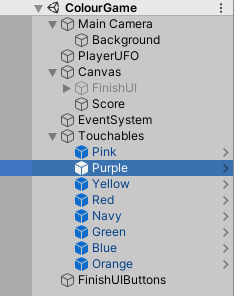
|  |  |
| --- | --- |
| Image result for github | **What is Github?:**  Github is a web-based hosting service for version control using Git. It also provides a Web-based graphical interface. It also provides access control and several collaboration features, such as a wikis and basic task management tools for every project |
| **How I used it:**  I used github to store different stages of my project as it progressed. I would commit and push updates to github in order to keep a record of my updates and changes. If it was needed I could go to a previous version of the app and work from that point. | |

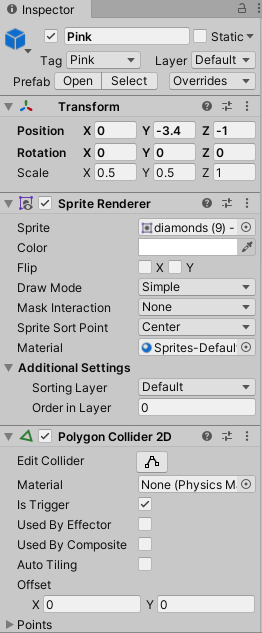
|  |  |
| --- | --- |
| macOS dock logo changed color · Issue #2913 · desktop/desktop · GitHub | **What is Github Desktop?:**  Github Desktop is designed to simplify essential steps in your GitHub workflow. |
| **How I used it:**  I used this to help me push my files to Github as a lot of my files were too large to push to Github remotely and needed to filter through all my files and push only the essential ones to Github. Github desktop makes it easy to see what files are updated and able to be committed. | |

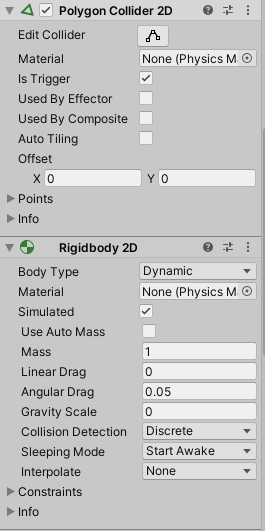
# Unity Run Through

Unity is described as the “ultimate game development platform”. It can be used to build high-quality 3D and 2D games, deploy them across mobile, desktop, VR/AR, console. It is a cross-platform game engine which is primarily used to develop video games and simulations for computers, consoles and mobile devices.

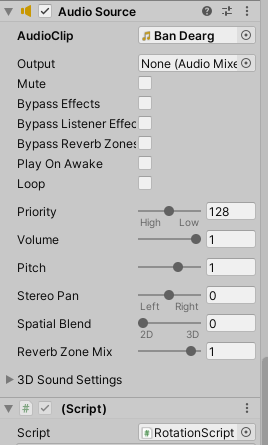
When creating a unity game there are many different steps and requirements you must follow and include in your application to ensure it works correctly. You must create scenes to hold everything you require in your game. I will explain what goes into one game scene below.

Here is what is kept inside a scene. Within my scene I have a main camera which is what will be seen by the user. This is where I place my background as I want it to cover the entire screen. Then I put my player sprite (simple 2D objects that have graphical images) onto the scene. The player UFO will have a C# script attached to it to control its movements. I then attach a canvas to my scene. A canvas is a component that controls how a group of UI elements will be rendered. Inside my canvas I place my finishing scene information, the score and the option to replay or go back to the game selector scene. I then have my event system, this is a way of sending events to objects in the application based on input, and be it keyboard, mouse, touch, or custom input.

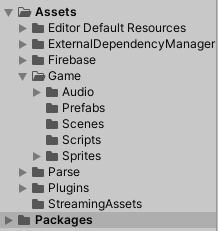
Then I have my touchables. My touchables are the sprites the user will aim to collect. To ensure my sprites are working correctly I must do a number of things with them. I must put them into a prefabs folder. Unity’s **Prefab** system allows you to create, configure, and store a GameObject complete with all its components, property values, and child **GameObjects** as a reusable Asset.

I then must ensure all the required components are added to each sprite. I place the sprite in the position I want, and ensure the Z position is -1 so that it is above the background position.

I then have the sprite render details which are created once you drag a sprite into a scene. I must create a Collider2D for my sprite so that when the player sprite collides with it, it is ‘destroyed’. It must be set to be a trigger so we can create the appropriate trigger code.

The Rigidbody2D is used to control features such as they gravity (which must be 0 so it does not fall off the scene, the sleeping mode (incase I wanted it to appear at a certain stage in the game) and the body type (designed to move under stimulation).

Audio is a key feature of my application and so I must ensure I add the right audio clip to my sprite, it must not be set to “play on awake” or else it will play as soon as they game is opened and then set the audio sound settings to be just right.

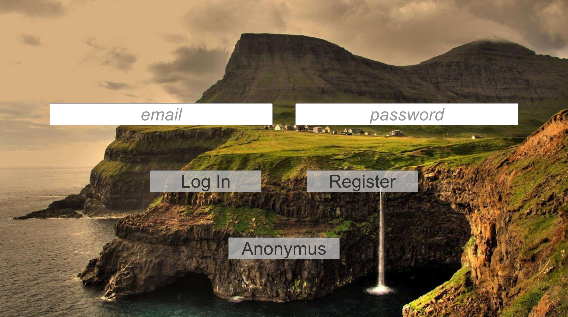
Unity’s file structure is also something new I had to come to learn about.

The Asset packages are collections of files and data from Unity Projects, or elements of Projects, which are compressed and stored in one file, similar to zip files. This is where I did all my work. I created the folders within the game folder to store my scripts, scenes sprites prefabs and audios. There are some files that I did not create, but imported. Files like the plugins and firebase file. These hold details for things like linking my app to firebase and my android plugins.

# UI Design/ Layout

(Graphics taken from Unity)

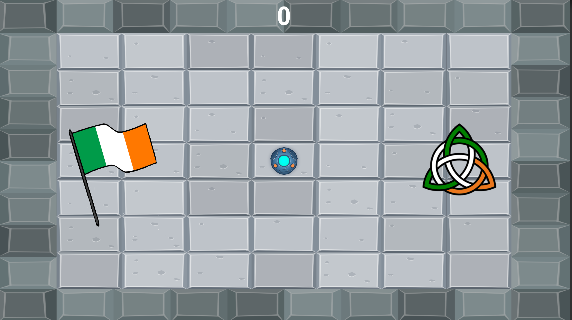
## Opening Scene: Register/Login

The first screen the user will meet is the Register / Login screen. Here they can create their account or login to their existing account. They also have the option to play the game without an account as an anonymous player.

## Next Scene: Start Screen

Once they have been logged in they will be brought to this start screen. Here they see three options available. They will have the option to start playing the games available on the application, or they can choose to play a demo game learning how the game works. They also have the option to quit the game here if they wish.

## How to Play Scene

This is the how to play scene. It is a simple demo version of the game they will meet when they chose the start option. Here they will be guided by an audio source through how the game works. It will explain what the goal of the game is and how to control their player sprite. They can replay this level as many times as they need.

## Game Selection Scene

The game selection scene is where the user will be able to view the many game topics on offer. Here we see a few different word games on offer. In the top right corner we see “Phrases Games”. This brings the user to a similar scene but instead of offering word games they are offered games to help them learn key sentences or phrase. This section would be expanding their vocabulary from simple words to usable sentences such as how to introduce themselves or order food (food would have been covered in words so they can modify the sentence to suit themselves).

## Game Scenes

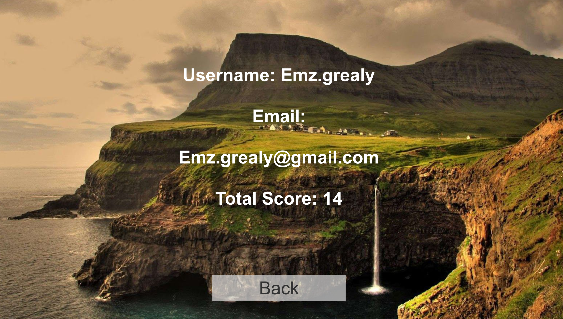


Here are three game options available to the user. They will be met by a scene with a number of images. I will use the actions game as the example level to help me explain the game. They will hear a word in Irish (EG:to sleep “ina coladh”). They will then have to attempt to drag their player controller to the correct action. The correct action in this case is the middle sprite on the bottom row of the girl sleeping. When they select an action they will hear the Irish for the spite they have collided with. The audio played at that moment will either confirm they are right, and their score increase, or let them know they are wrong and their score will not increase.

## Leader Board

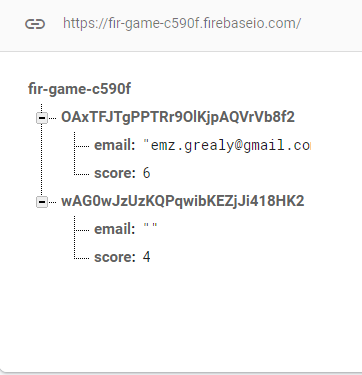
Here the user can come and view the top scoring players in the app. The idea of the leader board is to encourage the user to keep playing, trying new levels and continuing to improve as they go.

## Player Profile

The player profile is a personal page for the user to go and view their information. Here we can view things such as their email address, their username that would be used as their leader board name (email address without the @emailaddress ) and their total score earned over all the levels. This is their own personal reference to how they are doing. An anonymous user would not see any information in their profile as they have no information saved about them.

# Database Layout and Design

The database that I used for my project was Firebase. Firebase allows for realtime updates which are exactly what I wanted for my application.

Within my database I record each user. Each user will then hold their personal information (email and password) along with their highest scores recorded on each level they play. This user here has played two levels and scored a total of 6 and 4. Therefore in their player profile their overall score will be displayed as 10.

# Test Plan

User Acceptance Tests

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Title | Description | Expected Result | Result |
| 1 | Register New User | A new customer should be able to register for their own account. | Customer account is created and added to the database. | PASS |
| 2 | Empty Field Register | Attempt to create an account leaving a field empty. | User receives error message. | PASS |
| 3 | Login | All users should be able to login to their accounts. | User is successfully logged in. | PASS |
| 4 | Incorrect Login | Attempt to login to account with the wrong details. | User receives error message. | PASS |
| 5 | Empty Field Login | Attempt to login to account with leaving fields empty. | User receives error message | PASS |
| 6 | Play Demo Game | Attempt to play the demo game. | A demo game will allow the user to learn how the game works before diving into the variety of games on offer | PASS |
| 7 | Play Game | Attempt to play a certain game | User will play the game and their score is added to the DB | PASS |
| 8 | RePlay Game | Attempt to play a certain game again | User will play the game and if their score is higher, that score is added to the DB | PASS |
| 9 | View Leader Board | Attempt to view the top scoring accounts playing the game. | They will see the top 10 accounts and their total score earned through all games. | PASS |
| 10 | View Player Profile | Attempt to view a player’s personal profile containing personal info. | They will be able to see their personal information and their total score earned after combining all levels played. | PASS |

# Obstacles faced

## Unity

Although unity is a wonderful development tool to use for game creation there are many obstacles you can face because of the complexity and components required for each part of the project.  
I had designed a completely different game to the one I chose to use in the end. The game was a flappy bird style application where the user would tap the screen and the bird would jump up each time, and try to get through pipe gaps to collect the correct answer for the sound heard. However I faced many issues with this game. The size of the project itself caused it to glitch often and unfortunately this was something I had no control over. Secondly, adding the features I wanted, such as different gaps in the pipes and sprites in the gaps was very difficult to create as it was a moving screen and the pipes were being randomly created meaning it was very difficult to judge where the sprites would form. This resulted in me deciding to scrap this game and start from scratch with a new game.

## Github

Another issue I faced was pushing to github. Due to the size of my project pushing to github was very difficult. I attempted to use gits large file storage add on which was supposed to help push much larger files. Unfortunately this did not work for me. I then had to download Unity Desktop which allowed me to filter through the files and deselect auto created binary files and certain add ons like firebase functionality, which meant I could then push to github.

# Conclusion

In conclusion, I am happy with the mobile application that I have developed. I remember hearing all about the 4th year projects done before me, and reading their documentation all about their business ideas and the technologies they used. I thought that I would never be able to create something to a presentable standard in just one year, but I am very proud of myself and what I have created.  
I chose to use a brand new application development tool and language to challenge myself and broaden my skills. I did not want my application to be the same as everyone else. I enjoyed the process of game development and working with these new technologies.

I found that I learned so much during this experience. The constant feedback from supervisors along with opinions and ideas from classmates and friends really helped me with this entire process. If I was to do it all over again, I would definitely try to allocate more time to learning not just diving right into an idea. With development it really is a process of trial and error and I learned that starting smaller and making the mistakes earlier leaves more time to learn and improve. Unfortunately for me I spent a lot of time working on something that never made it to the demo stages.

# References

Unity Documentation: <https://docs.unity3d.com/Manual/index.html>

Firebase and Unity: <https://firebase.google.com/docs/unity/setup/?gclid=Cj0KCQjw7qn1BRDqARIsAKMbHDYXjo75lqjtWh5UpW4dXX23ohhcH0AryubjbKAjj-6uAK86cDA3y0EaAioWEALw_wcB>

Firebase and Unity: <https://www.youtube.com/watch?v=A6du3DUTIPI&t=7s>

Github Repo: <https://github.com/emmagrealy/Gliondar>

Unity and Github: <https://www.youtube.com/watch?v=liSi6E1HrHo&t=1s>

Unity and Github: <https://www.youtube.com/watch?v=WzrSRd1-w78&t=77s>