

**COMP07030 Professional Practice in IT Project**

**Louis Garon**

G00346194

Submitted to Galway-Mayo Institute of Technology (GMIT) in part

fulfilment for the award of Bachelor of Science (Honours) in Computing in Software Development

**Galway-Mayo Institute of Technology**

April 2021

Supervisor: Dominic Carr

**Introduction**

This project is made to show on what I as student have learned within the duration of my course as: Software Development. I have decided to do a 2D platformer game made in Unity. This will allow me to demonstrate the skills and knowledge I have acquired doing Mobile Applications Development. I used Unity Editor in order for me to edit the scenes and used C# for me to code the functions on each element of the game

**Objectives**

My objectives while working on this project/module are:

* To design an application that would require for me to show the skills and knowledge I have acquired in C# and Unity
* To gain more knowledge and experience of Applications Design Systems Development and Game Designs and Systems.
* To perform a demonstration of an application using Unity for the front end. Also using C# for the back end of the app which is the command center of the game

**Unity 2D Platformer Game**

**Instructions how to use:**

1. As soon as the user has hit the play button the user will be taken into the Introduction Scene where the user will be expected to click on the “Start Game” button.
2. The user will be then taken into the next scene which is the “Difficulty Level”, this scene is set up so that the user will be able to choose on the difficulty level. The higher the level the harder the level.
3. The main objective of the game is to collect the coins throughout the game and should be able to do it while having life, hence the life count at the top right of the level scenes
4. I’ve also added a Pause Menu for the user to either quit the game or adjust the volume of the sound FX of the game. This menu will show while in-game if the user has hit the “ESC” button.

**Systems Used:**

**Unity Games**

* Unity is the world’s leading platform for creating and operating interactive, real-time 3D content, providing the tools to make amazing [games](https://unity.com/madewith) and publish them to a wide range of [devices](https://unity.com/features/multiplatform). The Unity core platform enables entire creative teams to be more productive together.
* Thanks to the integrated two-way communication, you can switch into and out of Play mode, and pause and step a single frame without ever leaving Rider! The toolbar includes game view buttons Play, Pause, and Step, which correspond to the same buttons in Unity Editor and control the same behaviour as Unity Editor does. A small Unity icon in the status bar will indicate whether Unity Editor is connected, refreshing, or in Play mode. Any changes applied in Rider in Edit mode will be instantly delivered to Unity Editor.

**C#**

* C# programming Unity virtual reality C-sharp (C#) is a popular programming language developed by Microsoft in 2002. It has also been a main language for Unity game engine since 2005.
* C# is especially useful in XR development because this media is still in its infancy. Out of the box features are still under development, as people haven’t found the best solution for XR problems, nor the best implementations of the solutions

**Job Opportunity**

Within doing this project I have also done some research on what I could proceed as for my career path. I have done a little bit of research about a C# Developer with Game Engine Experience.

Requirements:

* 3+ years’ experience with C# and .NET
* Game Engine Experience
* Familiarity with Unity
* Experience with a Backend Platform or Product like GameSpark, Beamable or Playfab.
* Strong communication skills ESSENTIAL as you will be engaging with customers 50% of the time.
* Strong passion for game development.
* Experience with machine-learning and NLP will stand out.
* Degree in computer science is nice to have, but not required.

Responsibilities include:

* Contribute to the architecture and help shape the technical direction of the platform.
* Developing backend Game Development.
* Developing and testing mobile applications to technical specification.
* Ensuring cross-platform optimization for mobile.
* Work directly with product team to develop new features.
* Work alongside support team to resolve technical issues.
* Deliver data-driven impactful results.

**Problems encountered**

* Enemy Animations: Put an animation on the enemy it’s firing at the player, but the player won’t die and won’t lose health.
* Obstacle Collision Detection: The obstacles are spawning ,even if the player has shot the obstacles. When the player shoots/hits the obstacle, it would add points to the player but when it gets hit, it has the sound that it got hit but it would just stay at the spawning area.
* Volume Slider: Created an audio slider using the standard method, and reduce the slider to about halfway, only to find out that the audio isn’t half as loud (as you might expect), in fact it’s virtually silent.

**Conclusion**

* On Creating Animations, I searched through forums and found out that my Game Object must have an Animator Component and this must have an Animator controller Asset assigned and should have on or more Animation Clips Assigned for the Animation to work perfectly
* Obstacle Collision Detection: It turns out that the problem was, there was a wrong collider component added to the child instead of the parent.
* Volume Slider: I used a slider value range of 0.0001 – 1 instead of -80db – 0db (the 0.0001 is important, and stops the slider breaking at zero) and When passing the slider value to the SetFloat function, convert it using Mathf.Log10(value) \* 20;

Github Link:

https://github.com/louisgaron/G00346194\_PPIT\_Project.git