COMP 30540 Game Development

Drone Destroyer

Eoin Goslin (18204142)

Msc. in Computer Science

UCD School of Computer Science

University College Dublin

April 2019

Table of Contents

Introduction

This short report will discuss how the drone destroyer game was created and the process of learning to use Unity and Blender. There were many obstacles faced while making this game and I found it a considerable challenge but definitely learned a lot. After creating the game, I would like to use Unity again and feel that making another game in the future would not be as daunting of a task after what was learned throughout this project.

Getting Started

1.1. Unity

Unity is such a massive software that when it is first opened it can be quite daunting to look at. I found it hard to follow tutorials to do with moving a player or setting up a scene at the beginning as I was not familiar with any of the concepts (such as a player having a Rigidbody) so I went back to the very beginning. I looked to the Unity docs to help going through beginner lessons on what the inspector was and different Unity concepts such as adding scripts to objects. I created a survival shooter [1] following a Unity tutorial which started from the very basics and this really gave me a good grounding to begin this project.

1.2. Blender

Having no experience at all in using this software or any modelling software I found this quite a challenge. I wanted to create a rope in Blender but unfortunately after following numerous tutorials none of them worked. I was able to create the rope object but it was not quite right when it was imported into Unity. I wanted something to appear in the game that I had modelled so I created a house [2] to represent the airport worker house and placed it on a Unity plane which was used as the healing area in the game. When players fly over the heart, they regain health if they have enough healing power left.

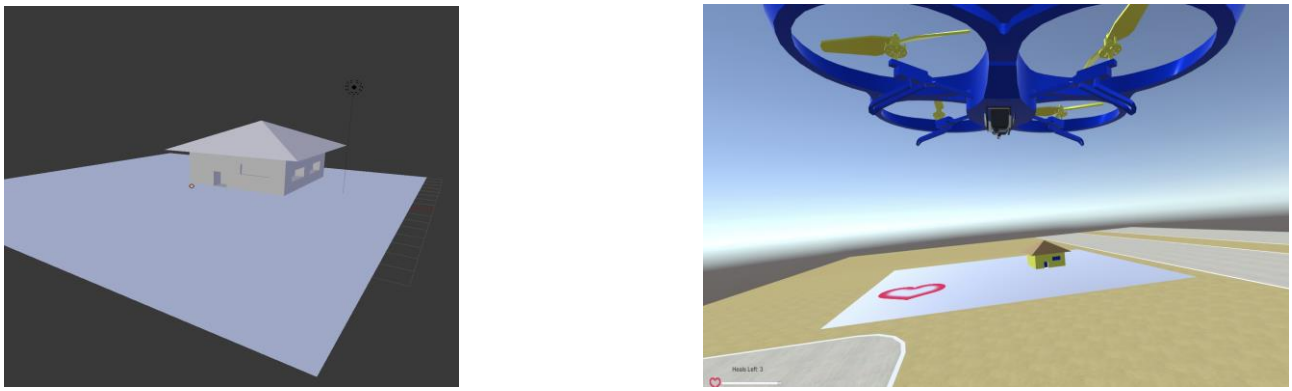

Figure 1. Modelled house in Blender and Unity

Creating Multiple Scenes

1.3. Main Menu

The main menu is where the user is first placed when the game begins. It was not until near the end of this project that the main menu and multiple scenes were created. The main game scene in which the drone was flown only existed as I was not aware of how to move between scenes and capture button presses so the core game was focused on first. I learned that Unity has quite an elegant way of creating multiple scenes. Any scene with a camera can be a menu scene all that is needed is to create a canvas and place UI elements onto this canvas.

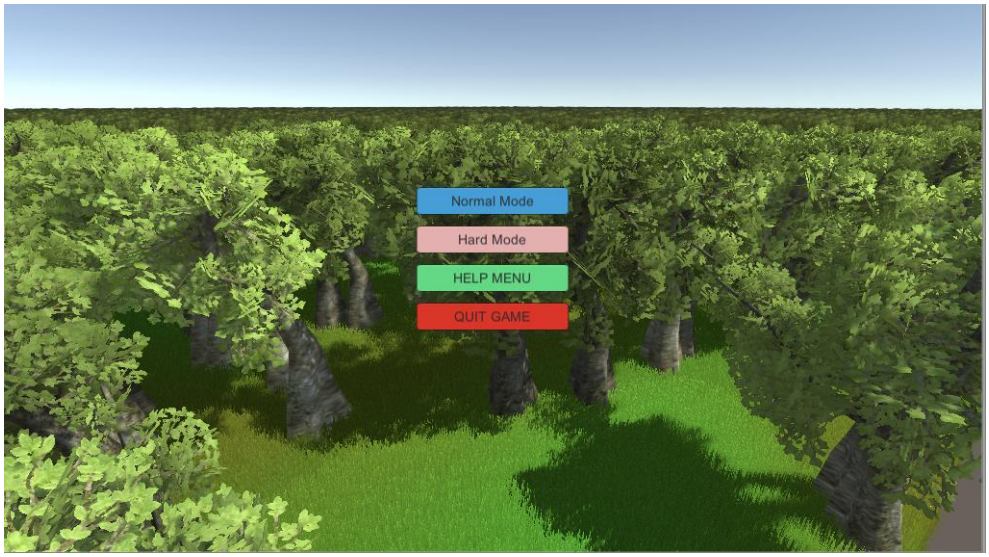

Figure 2. Main Menu with moving trees and grass

Each option is a UI Button that has the same game object attached to it. This game object has a script attached to it with a method that is called depending on what button was chosen. The background was first static but I knew that this was just a scene so anything could happen in the background so I wanted to create some moving elements. The grass and trees were placed on top of a terrain object. A Unity 3D object called a 'Wind Zone' was placed in the scene which has multiple settings and can interact with scene items such as adding wind to the grass and trees making them sway. I liked the idea of creating a peaceful start page and then when the game begins a player is thrown into the drone fight to add to the intensity of the game.

1.4. Help Menu

The help menu shows how the player can interact with the game and the same music is played from the main menu scene. It was a challenge to not make the music restart when traversing through the menu but Unity has a solution for this. The music does not start again moving between scene as a "DontDestroyOnLoad" method is used passing in the music object that is not to be removed from the game when the scene finishes.

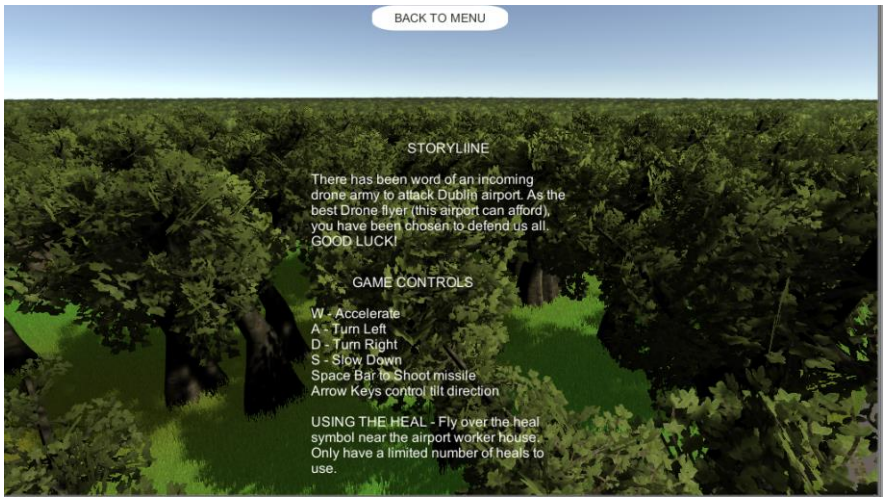

Figure 3. Help Menu showing instructions to the user

Main Scene

1.5. The Player

The main player drone was obtained from the Unity asset store for free [3] along with the beginning airport scene [4]. This drone appeared with many other files but I just took the drone, it included no functionality for movement or firing it was just the 3D model without a gun. I wanted each rotor to move and move at different speeds depending on what way the player moved as well as the sound to change when moving faster. To achieve this, each of the rotors was tagged and the main player object had a script attached which would get all of these rotors. The player had a public value for speed from a separate script that was created which could be accessed through the "GetComponent" in Unity, depending on what the value was the rotor would be rotating using the "Quaternion.Euler" method. The gun for the player and the enemy bullet was obtained from the Unity asset store as well as part of a helicopter package and it is actually a model of a chain gun [5]. The music was played with an AudioSource component and using a logarithmic rollof the sound changes.

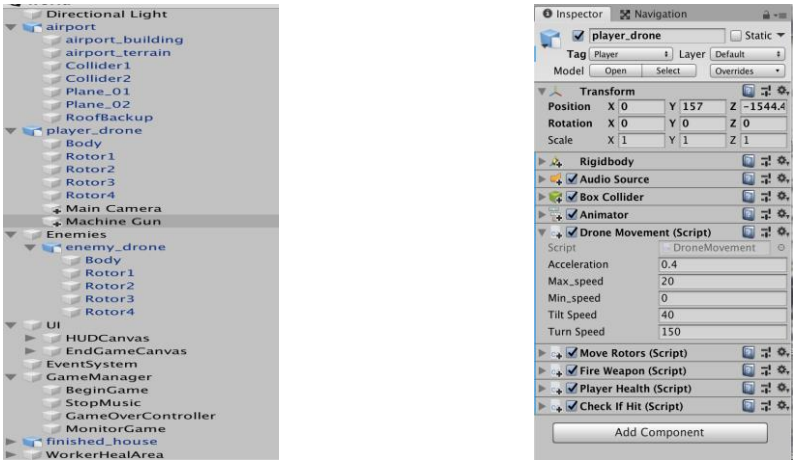

Figure 4. Scene manager and inspector showing objects and scripts on objects

1.6. Setting up the scene

The main scene has multiple objects and at first it was difficult to understand how to manipulate the scene dynamically from a script at runtime rather than just dragging in a set number of objects. I learned that the "Instantiate" method is used in Unity to spawn objects in the scene and this was used to create the enemies so they can appear in random locations and the number of enemies can be easily changed. For the enemy health, a slider value is used which is a UI element placed on a canvas. The number of heals remaining is also displayed.

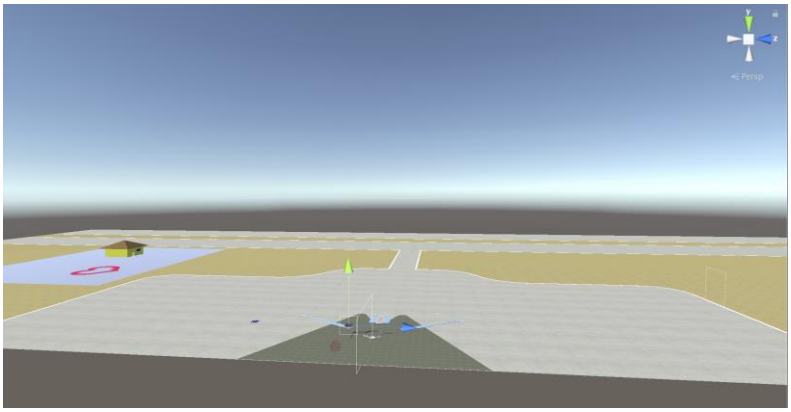

Figure 4. Help Scene with all starting objects visible as well as health canvas at the front

1.7. Enemy Drone

The enemy drone moves towards the player and fires once the player is in range. To move the player a "Nav Mesh Agent" was originally tried but this did not work as I was not able to place a mesh in the sky so the drone would only be able to follow the player if they were grounded. To get around this, I used Unity Vectors which come with "MoveTowards" method which can be called in every frame by placing them in the update method. This workaround seems effective but, in the future, I would like to learn how to use Unity's AI for flying objects.

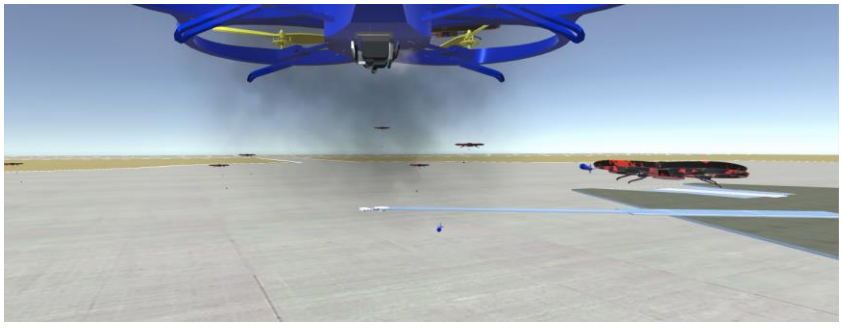

Figure 5. Drones facing player and smoke from player being hit

Conclusion

Although a lot was added to this game, I was not able to add all the features that I had planned due to time constraints. The rope bullet took two days to try and model and use in Unity but it did not work so I would like to achieve this in future versions. Two difficulties exist in the game offering extra enemy damage, more enemies and less heals as an option for players who want a challenge. An extra level I wanted to add was a timed round in which there are 50 drones and the player does not have a gun and no heals are available and the player must try to evade the enemy for as long as they can trying to beat their score. I have learned how to add/remove guns and use timers so this would not take a vast amount of time but I think it would add an extra dimension to the game. A boss drone as well to appear after all drones have been beaten which would have greater fire power and would move quicker is something I would like to implement.

From never having created a 3D game before or used the Unity engine, this game has challenged me considerably but has been very rewarding. I think the game showcases a lot of the skills I have learned and there are numerous personal embellishments to make it a bit more unique.

References

[1] Unity (2019) Survival Shooter Tutorial [online]. Available at: https://unity3d.com/learn/tutorials/s/survival-shooter-tutorial (accessed 14 April 2019).

[2] YouTube (2019) Making a simple house in Blender [online]/ Available at: https://www.youtube.com/watch?v=ITfkvQdWaRA (accessed 14 April 2019).

[3] Unity Asset Store (2019) Drone [online]. Available at: https://assetstore.unity.com/publishers/20994 (accessed 14 April 2019).

[4] Unity Asset Store (2019) Sample Racer Environment Pack [online]. Available at: https://assetstore.unity.com/packages/3d/environments/urban/sample-racer-environment-pack-63641 (accessed 14 April 2019).

[5] Unity Asset Store (2019) Military Aircraft Pack[online]. Available at: https://assetstore.unity.com/packages/3d/vehicles/air/military-aircraft-pack-115124 (accessed 14 April 2019).