



---

# Dissertation

---

**Blaine Burke**

B.Sc.(Hons) in Software Development

MAY 3, 2021

## **Final Year Project**

Advised by: Mr Gerard Harrison

Department of Computer Science and Applied Physics  
Galway-Mayo Institute of Technology (GMIT)

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
<b>2</b>	<b>Context</b>	<b>7</b>
2.1	Objectives . . . . .	7
2.1.1	Game Name . . . . .	7
2.1.2	Game Logo . . . . .	8
2.1.3	Art Styles . . . . .	8
2.1.4	Screen distortion effect. . . . .	9
2.2	Target Audience . . . . .	9
2.2.1	Player Movement . . . . .	9
2.2.2	Player Interactions . . . . .	9
2.2.3	Enemy AI . . . . .	9
2.2.4	Player Weapon . . . . .	9
2.2.5	Menu Area . . . . .	10
2.2.6	Day and night cycle with fog effects . . . . .	10
<b>3</b>	<b>Development Diary</b>	<b>11</b>
3.1	Virtual Reality Rig . . . . .	11
3.2	M1911 pistol . . . . .	11
3.3	Enemies . . . . .	12
3.4	Terrains . . . . .	12
<b>4</b>	<b>Methodology</b>	<b>13</b>
4.1	Project approach . . . . .	13
4.2	Testing . . . . .	14
4.3	Version Control . . . . .	14
4.4	Choosing technologies . . . . .	14
<b>5</b>	<b>Technology Review</b>	<b>16</b>
5.1	XR Interaction Toolkit . . . . .	16
5.2	Unity 3D . . . . .	16

<i>CONTENTS</i>	3
5.3 Trello Board . . . . .	17
<b>6 System Design</b>	<b>18</b>
6.1 Virtual Reality Controls . . . . .	18
6.1.1 XRRig.cs . . . . .	18
6.2 Enemy Path finding . . . . .	18
<b>7 System Evaluation</b>	<b>20</b>
7.1 robustness . . . . .	20
7.2 performance . . . . .	20
7.3 Limitations . . . . .	20
<b>8 Conclusion</b>	<b>21</b>

# About this project

**Abstract** VR or Virtual reality is a relatively new, upcoming technology that had a market size of 10.3 billion dollars in 2019[1]. It has a vast amount of uses throughout the world in many industries for research, development and training in many areas such as in medicine or in rehabilitation. It also has many obscured uses such as helping people overcome physiological phobias or social marketing which involves using commercial marketing principles and techniques to enhance the lives and welfare of people.

Monkey Business is a virtual reality game built using Unity, a free 2D/3D engine and framework. It consists of multiple mini games such as rock climbing and zombie survival. In rock climbing you must climb as fast as you can to the end of the level in order to achieve the fastest time possible. If you successfully beat the current high score your score will display in the main menu area for that level! In zombie survival, the goal is to survive while also getting as many kills as possible! Upon dying the zombies drop additional magazines for the player to pick up and use.

**Authors** My name is Blaine Burke. I am in final year of college doing a level 8 honors degree in software development in GMIT, Galway, Ireland. Since an early age I have always had an interest in video games. I watched hours upon hours of interviews with the creative minds behind my favourite games and watching videos of people playing them. Eventually I learned that many games can be modded meaning that the game play, physics or mechanics of the game could be altered in small or big ways. I saw the amazing things that people had managed to imagine and create and this inspired me more to want to create my own game. It wasn't until I began college that I really began making games. I had a module called 'Mobile Application Development' with a lecturer named Damien Costello. He taught me all the fundamentals of Unity and C#. It was then that I finally began my passion of making games.

# Chapter 1

## Introduction

The introduction should be about three to five pages long.

Monkey Business is a family friendly climbing game created using Unity 3D. The aim of the game is to climb as fast as possible to end the end of the level. At the end of the level there is a red button which the player must hit in order to end the timer. If they beat the current high score for the level the timers text will appear green and if they do not it will appear red. All of the high scores for the levels will be displayed in the main menu area on an interface.

VR or virtual reality is a virtual experience that can be designed to be similar or different to the real world. It is a relatively new, upcoming technology with a huge future potential and in 2019, had a large market size of 10.3 billion dollars[1]. It has vast amounts of uses throughout many industries for research, development and training in many areas such as in medicine or in rehabilitation. It has many obscured uses such as helping people overcome a physiological phobia or social marketing which involves using commercial marketing principles and techniques to enhance the lives and welfare of people.

When I initially began this project I had no virtual reality hardware. I spent a week researching different options and eventually narrowed my options down to two, the Oculus quest 2 or the Valve Index. They are both excellent headsets but there were several differences between them. The Oculus Quest 2 can be used in both a wireless and wired mode, it also has hand tracking and a higher resolution display. The Valve Index on the other hand also has many benefits such as individual finger tracking, a wider field of

view and a higher refresh rate. However there are significant differences in the prices. The valve index costs over one thousand euros whereas the Oculus quest 2 is priced at 349 euro for the 64 gigabyte version and 450 euro for the 256 gigabyte version. Due to being a student and having not too much disposable income, I chose the latter.

# Chapter 2

## Context

My reason for creating a virtual reality game is to demonstrate the uses of virtual reality technology whilst also creating an enjoyable family friendly game. I enjoy virtual reality development and would like to eventually try use it to aid people through things such as training or for rehabilitation.

### 2.1 Objectives

Throughout the course of this project, many of my objectives changed. Some were for technical reasons and others were due to my imagining of the end result changing. My original objectives were as follows...

#### 2.1.1 Game Name

Originally, I planned on calling my virtual reality game 'Xylophobia', meaning a fear of woods especially in the dark. This name was fitting as originally the game was an open world game that takes place in a forest where the player could freely roam around the area. There is a day and night cycle in the game, and upon the game becoming dark the player must stay in range of a light source in order to avoid their screen becoming distorted with a visual effect.

Eventually due to the game being too intensive, I needed to change my art styles to a low poly one. I decided to change the game's name to 'Monkey Business'. This was due to the bright, family friendly type art styles and game modes in the game.

### 2.1.2 Game Logo

When creating the logo I wanted it to stand out. I wanted to do this by using contrasting colours. I researched logos for companies such as Fanta and IKEA which both have their logo or company name in a contrasting colours, for example IKEA's yellow on blue. This method makes the logo stand out and eye catching. I decided I would make a few different concepts for the logo.

My first concept was to put white bold text on a brick background. I put a black shadow effect of the writing to make it stand out even more. My second concept was to keep a simple design with solid colours that, again contrast each other. I chose to use a yellow background and black bold text with a white shadow effect. I chose black and yellow as they are colourful and it gives the best visual contrasts.



### 2.1.3 Art Styles

Upon beginning the project I initially used realistic detailed art styles. This was to give a grim, realism to the game to emphasise the dark nature of a post-apocalyptic world.

Unfortunately, I encountered many issues with my original ideas. The realistic art styles proved to be extremely CPU and GPU intensive. I attempted to fix these issues by using LOD groups, occlusion culling and other rendering techniques. Sadly, they weren't enough to fix the frame rate issues. As a result I began to re-imagine the game and its art style. After researching many virtual reality game art styles for many virtual reality games such as Phasmophobia and Superhot VR. Both these games have simple graphics and low poly graphics and as a result they are easy to run for lower end systems. I found more assets on the asset store that I could that were low poly yet detailed created by Broken Vector.[2]



### **2.1.4 Screen distortion effect.**

There is a day and night cycle in the game, and upon the game becoming dark the player must stay in range of a light source in order to avoid their screen becoming distorted with a visual effect.

## **2.2 Target Audience**

The target audience of the game was originally meant for horror fans and an older age group. This is due to the game featuring zombies, weaponry and blood effects. Upon changing the game to monkey business however, I removed the blood particle effects due to the game being more family friendly.

### **2.2.1 Player Movement**

Implement a continuous movement system so the player can freely move around the play area with the left analog stick. This also includes snap turning so the player can rotate 30 degrees to the left or right using the right analog stick.

### **2.2.2 Player Interactions**

Allow the player to grab and interact with specified objects. This includes the pistol and climbing walls.

### **2.2.3 Enemy AI**

Originally I planned to have the zombies search for the player and if the player became visible the zombies would run at them and attack. I removed the searching aspect after I changed the game from Xylophobia to Monkey Business and changed it to just path finding.

### **2.2.4 Player Weapon**

Add a working pistol that can be loaded and cocked for the player to use in order to defend themselves from zombies.

### **2.2.5 Menu Area**

Create a menu area that has a main menu, high score interface and some things for the player to play around with. Originally I had created a prison area that had a climbing wall and a pistol for the player to shoot some cans.

### **2.2.6 Day and night cycle with fog effects**

I added a day and night cycle to the game as it makes the game world look more beautiful. I added fog as it increases fps in a game. This is as it lessens the amount of objects and textures to be rendered on screen so it is less taxing on the players hardware and makes the game easier to run.

# Chapter 3

## Development Diary

Throughout the course of the project I recorded my changes and additions to the project. I recorded all main updates and changes in the GameUpdates.md file.

### 3.1 Virtual Reality Rig

When I began the project, the first thing I created was the virtual reality rig. Using YouTube tutorials by Valem[3] and guides on stack overflow, I eventually created the head of the virtual reality rig. I then began to create the virtual reality hands for the player. I began by finding a hand model with the fingers being all child objects. This allowed me to add animations to the hands such as a fist and pinching hand gesture. I also created it so it takes into account how much the buttons have been pressed by getting the input devices values. The hand animations will work in sync with the amount the button has been pressed. For example by pressing down the grip button to the half way position, the hand gesture will not be a fully formed fist but will be a half formed fist gesture.

### 3.2 M1911 pistol

When creating the pistol, I had many issues. This is due to it having many moving components such as the pistol slider and the magazine. I began by finding a pistol asset that was split into several parts. This allowed me to add joint components onto the individual game objects to allow the player to have more control over the gun. I began first by attaching the XR Grab Intractable script to allow the player to be able to grab and hold the gun. I then ran into an issue where the pistol was not being held correctly in the

players hand so I created a hold transform for the pistol. I then added a sphere attach point for the pistols magazine that allowed the player to insert a magazine into the pistol. Once again I had to create an attach transform so the magazine would sit in correctly.

### 3.3 Enemies

The enemies were originally meant to patrol an area and until they spotted the player within a radius of them. This method was functional to a degree, however there was no path finding on the enemy so it would just walk in a straight line toward the player. At this stage of development, the enemies were just capsules with the scripts attached. I searched on the unity asset store and I found many different assets that I could use for my game. I also used other sites such as Turbo Squid to search for assets [4].

I then set out to work on adding path finding for my enemies.

### 3.4 Terrains

# Chapter 4

## Methodology

### 4.1 Project approach

Due to my projects large size, I took an iterative approach when doing this project meaning I created all the required game objects and scripts and then came back to them one by one, fixing and improving them. I enjoyed this approach as I was doing different things every two to three days, which kept it interesting for me and allowed me to further enjoy the development process.

Initially I began by creating the basic game objects for the game such as...

- A VR rig with simple movement scripts for the body, hands and head.
- A basic test level for development of other future game objects.
- Allowed the player to grab and hold specified objects.
- A basic enemy AI that simply walked directly at the player when in its line of sight.

They were all functional but needed improvements for example the enemy would always face the player but sometimes would get stuck and would not be able to move toward the player. I began dealing with any issues with my basic requirements by either fixing or removing things from it for example I eventually removed my first enemy as it was quite buggy and not very intelligent as it would get stuck on objects due to having no path finding algorithm implemented. I continued this cycle throughout the entirety of my project as I proved successful and efficient to me being a solo developer.

## 4.2 Testing

I took a trial and error approach to testing. I would slightly tweak things to see how well they would work and what affect it would have on its functionality and surroundings. For example when creating grab positions for the player, I would have to tweak the transform of the held item so it is positioned correctly in the players hand. I found this time consuming as I would have to stand away from my desk, put on the virtual reality headset and remotes and test a feature for sometimes only a few seconds.

I was also affected by being a solo developer as I would sometimes not notice a small mistake either in Unity or in the scripts that someone else might notice. These small errors in my code initially were frequent however over the course of the project, due to improving my knowledge of Unity and C#, I would rarely make simple mistakes.

## 4.3 Version Control

I used GitHub to upload my files but quickly ran into an issue due to the sheer size of my project. The overall size was almost ten gigabytes due to assets and other large files. To combat this issue, I only uploaded scripts, audio files and other small files to my repository. In the README.md file, I provided links to my google drive containing the current build of the game. I found this way of updating suitable as I could control the version of the game that was available to be played ensuring there are no bugs, issues or unfinished things left in the game.

## 4.4 Choosing technologies

Upon beginning the project, I researched a variety of games to see what engine and language they were created in and found a lot of high quality virtual reality games were created using Unity and C#. I also searched for tutorials and guides I could follow. I quickly discovered some small YouTube channels that provided me with some great tutorials on the basics of virtual reality. Channels such as 'Valem'[3] and 'Dave / Game Development'[5]. There were also many guides on sites such as Stack Overflow for bug fixes.

I then researched between different ways of developing the virtual reality game. I looked at the oculus virtual reality development kit and the XR Interaction Toolkits and decided to use the latter due to being well documented and having many guides.

# Chapter 5

## Technology Review

There were many technologies used throughout this project.

### 5.1 XR Interaction Toolkit

### 5.2 Unity 3D

I used Unity 3D for my game engine and this was for many reasons. I have been learning Unity in college with my lecturer, Damien Costello. He taught us the basics of Unity 2D at first with simple movement, player interacting with other game objects through collisions and key presses. He then taught us Unity 3D at the start of my final college year.

Due to being relatively new to game development, Unity is a good option. It is used in many reputable games like virtual reality games. It is also used by a vast amount of indie developers and mobile game developers. It's free of charge making it even more widely available for students and young people alike. It also has several tutorials on how to make several different types of games and an official website for tutorials too [6].

Unity also has a vast range of tutorials all over YouTube that have helped me with many projects including this one. I also used many different reputable sources for game tutorials such as Valem[3] and Brackeys[7].

I also chose to use Unity 3D as I would like to become a game developer and many companies are seeking Unity game developers.



### 5.3 Trello Board

During the project I used a trello board to keep track of what I had done and what I needed to do [8]. I created several different boards for different aspects of the game. For example I had a board for the climbing game and one for the zombie game also. I initially put all jobs in the to a to do list. I then made another list for jobs I was in the process of completing and another for finished jobs. Finally I had a list for jobs that I was having issues with so I could come back to them at a later date.

I found the trello boards useful as it kept me on track throughout the course of the project.

# Chapter 6

## System Design

### 6.1 Virtual Reality Controls

The player is able to freely move about the scene and interact with things due to several scripts.

#### 6.1.1 XRRig.cs

I got this script from the XR Interaction Toolkit. It sets up the VR Rig for the player. It allows you to set the camera height for a stationary tracking space. You can set its tracking mode origin which in this case is the floor so it sets the transform to 0,0,0. It also sets the virtual reality camera which is also the players head.

These values that are set by it are used in the continuous movement script to control the player further. ContinuousMovement.cs

### 6.2 Enemy Path finding

In Zombie survival, once spawned the zombies will move toward the players position. This is done using a nav mesh agent component. This component is attached to a mobile character in the game to allow it to navigate the Scene using the NavMesh specified. I also use it to set a variety of variables such as the enemies height, their turn speed and acceleration. I made sure to set the enemies radius just right as the smaller the radius the closer the enemies can go to walls and objects. If their radius was too high they would, in some cases not be able to reach the player in specific spots.

Originally the zombies would patrol the area and search for the player but

upon changing from the horror themed game, I changed it so the zombies just go straight to the player regardless of their position or line of sight.

As many pages as needed.

- Architecture, UML etc. An overview of the different components of the system. Diagrams etc. . . Screen shots etc.

# Chapter 7

## System Evaluation

### 7.1 robustness

### 7.2 performance

### 7.3 Limitations

As many pages as needed.

- Prove that your software is robust. How? Testing etc.
- Use performance benchmarks (space and time) if algorithmic.
- Measure the outcomes / outputs of your system / software against the objectives from the Introduction.
- Highlight any limitations or opportunities in your approach or technologies used.

## Chapter 8

### Conclusion

To conclude this document, I thoroughly enjoyed creating a virtual reality game using Unity. However there are still some aspects that I would change if I were to redo the project now.

I faced many challenges throughout this project. Due to being a solo developer, there was a large work load that had to be completed. There were levels to be designed, colour themes to be selected, art styles, game play mechanics to be created and more. I would change this by getting another person to work on the project with me. This way we can cover more quicker and it also prevents less details to be missed. Another downfall of being a solo developer is time pressure. I didn't have enough time to perfect many aspects of the game such as the pistols bullets not shooting fast enough and the weapons magazine not staying still inside the gun.

About three pages.

# Bibliography

- [1] H. Tankovska, “Forecast augmented (ar) and virtual reality (vr) market size worldwide from 2016 to 2020,” 2020.
- [2] B. Vector, “Low poly brick houses.”
- [3] Valem, “Vr game development youtube channel.”
- [4] TurboSquid, “Turbosquid.”
- [5] Dave/GameDevelopment, “Game development youtube channel.”
- [6] Unity, “Unity learn.”
- [7] Brackeys, “Game development youtube channel.”
- [8] Trello, “Trello board.”