Player 2wo: A Secondary Journey

By

Lewanika Pittman and Amro Tayeb

Senior Project Submitted to the Faculty of the Department of Computer Science

in partial fulfillment of the requirements for the degree of

BACHELOR OF SCIENCE

in Computer Science

October 09, 2017

Nashville, Tennessee

Approved:

|  |  |  |
| --- | --- | --- |
| Technical Advisor: | Dr. Erdem Erdemir |  |
| Course Instructor: | Dr. Ali Sekmen |  |
| Department Chair: | Dr. Ali Sekmen |  |

# CHAPTER 1

INTRODUCTION

In the current video game market, video game development has become more streamlined and accessible than ever, with free or very affordable open-source, multi-platform game engines such as Unity, Godot, and CryEngine paving the way for aspiring developers, with helpful tools that are useful to those with skills ranging from entry-level to AAA level. This project will involve a video game that will be developed in a second-person perspective, viewed in standard 3D, and ported to VR utilizing one of these open-source engines, Unity. Unity is one of the most popular development engines that has played host to a plethora of AAA games from the Angry Birds series, Hearthstone: Heroes of Warcraft, to highly rated indie games such as the physics-based simulator Kerbal Space Program. Our aspirations are to design our own game that could eventually be placed onto a digital mantle alongside such classics.

# Problem Statement

The main challenge will be designing a game where the A.I. controlled character will have to be programmed with a role that would generally be assigned to the user. This will be a particularly tricky challenge because the main A.I. will need to be programmed in a manner that mimics a typical human user’s behavioral pattern and thought processes. Another challenge will be integrating the entire experience into the virtual reality spectrum. Careful attention to detail must be incorporated when developing VR such as making sure that every object is properly and fully rendered in 3D to prevent the player from losing any sense of immersion. It’s also vital to carefully develop a type of field-view camera motion to prevent any depth-of-field issues or motion sickness with sensitive users.

# Literature Survey

This section will give an example of game designs similar to the project that we will undergo, as well as give the reader some definitions of some of the devices and platforms in the aforementioned pages to give an idea of what will be used and how the development process will be done.

* + 1. **Age of Empires**

Age of Empires is a series of personal computer games originally developed by Ensemble Studios and published by Microsoft Studios. The titles are historical real-time strategy games where players control a variety of civilian and military units to develop a civilization from a handful of hunter-gatherers to an expansive Iron Age empire. The first title of the series was Age of Empires, released in 1997. The artificial intelligence (AI) used in the Age of Empires series has been developed and improved regularly by designers. The AI in the game relies on tactics and strategies to win, instead of "cheating" by giving bonus resources to itself, or tweaking its units to be stronger than normal.



Figure 1.1: Age of Empires.

**1.2.2 THE SIMS SERIES**

The Sims is a life simulation video game series, developed by EA Maxis and published by Electronic Arts. The games in The Sims series are largely sandbox games, meaning that they lack any defined goals (except for some later expansion packs and console versions which introduced this gameplay style). The player creates virtual people called "Sims" and places them in houses and guides them through simulated lifestyles and that help direct their moods and satisfy their desires.



Figure 1.2: The Sims.

**1.2.3 SECOND PERSON SHOOTER ZATO**

By far the most obscure and strange game on the list, Second Person Shooter Zato, released by Japanese indie developer Himo in 2011, is a unique action game that puts you in control of a gun wielding hero that has to progress through a series of enemies. However, instead of the traditional first-person perspective, the player can only see himself through the eyes of the enemy, having the player play as if looking through a mirror.



Figure 1.3: Second Person Shooter Zato.

**1.2.4 VIRTUAL REALITY**

Virtual reality is the term used to describe a three-dimensional, computer generated environment which can be explored and interacted with by a person. That person becomes part of the virtual world or is immersed within this environment and whilst there, is able to manipulate objects or perform a series of actions.

**1.2.5 OCULUS RIFT**

Invented by a VR enthusiast named Palmer Luckey, the Oculus Rift is a set of virtual-reality goggles that will work with your gaming desktop or laptop. The Rift accomplishes virtual reality by using a pair of screens that displays two images side by side, one for each eye. A set of lenses is placed on top of the panels, focusing and reshaping the picture for each eye, and creating a stereoscopic 3D image. The goggles have embedded sensors that monitor the wearer's head motions and adjust the image accordingly.



Figure 1.4: Oculus Rift Consumer Kit.

# Research Goal and Objectives

## Goal

Our goal is to create a functional, 3D role playing game that is compatible with both a standard television and an Oculus Rift. Our objectives will include the creation of fully rendered character models, from the main character to enemy character archetypes, all equipped with functional AI behavior that’s individually appropriate to each character’s given characteristic. We will also be creating a fully functional world with properly placed interactive objects and ambience to create a realistic atmosphere. The finished product will then be ported and properly integrated to fully functionally work with the Oculus Rift.

## Associated Objectives

The specific objectives are listed as:

1. Familiarize ourselves with unity and its applications.

2. Get familiar with downloading objects from 3d warehouse website and insert them into unity.

3. Get familiar with the way of coding in unity.

4. Get familiar with ray casting in unity.

5. Find software to do 3d objects in unity.

6. Create/find the main character 3d model and the enemy models.

7. Create two different levels of the game.

8. Implement the movement of the main character and the enemies.

9. Find a soundtrack for our game.

10. Implement the audio for the game (movement, attack, enemies sound).

11. Learn unity GUI system.

12. Create two different terrains for the game.

13. Make two functional versions of the game, one on pc and the other for Oculus Rift.

14. Check for other similar games and learn how they operate.

15. Learn how to make graphics as realistic as we can.

16. Test and debug the game.

# Project Organization

The rest of the report will contain the following contents: Chapter II will contain detailed requirements analysis with both functional and non-functional requirements. Chapter III will describe architectural and detailed designs. Chapter IV will go into the design’s implementation. Chapter V will document the results. Chapter VI will finalize the report with the conclusion and discuss conceivable expansions of the project.

# CHAPTER 2

REQUIREMENTS ANALYSIS

This chapter will discuss functional and non-functional requirements regarding the game’s creation and implementation. The functional requirements will describe the system’s necessary components needed for the project’s completion. The non-functional requirements will cover all challenges that will be faced in regards to making the game experience as smooth and as presentable as possible.

# Functional Requirements

1. The game will be an rpg

1.1 The main character will level up.

1.1.1 Leveling up will increase the main characters various stats like attack and defense.

1.1.2 The main character will level up by killing enemies.

1.2 The main character will be able to discover and use equipment.

1.2.1 This equipement will be things like weapons armor or shields.

1.2.2 This equipment will have different stats or abilities.

1.3 The enemies will be set at a level that they are strong enough to challenge the main character.

1.3.1 The enemies will be a higher level the further into the game you get.

1.3.2 The enemies will also have state like attack and defense and different equipment like different kinds of weapons.

2. Every level will contain a goal or exit and obstacles and enemies.

2.1 Enemies will stand or patrol in certain areas.

2.2.1 Some enemies will hide in chests or ambush switches.

2.2.2 Enemies will attack the main character on sight.

2.2 Each level will have chests.

2.2.1 The chest will contain equipment.

2.2.2 The main character will sometimes open the chests if he finds them or the player points them out.

2.3 The obstacles will be closed doors, gates, bridges, etc.

2.3.1 These obstacles can be removes or bypassed by activating switches.

2.3.2 These switches must be activated by the main character.

3. The main character will navigate levels with the aid of the player controlling a secondary character.

3.1 The main character will use navigation to head toward the goal or exit of a level.

3.1.1 If the main character comes across a door or some other obstacle he will start wanderng around the area at random.

3.1.2 Once the obstacle has been dealt with the main character will continue heading towards the goal.

3.2 The main character will have a cone of vision.

3.2.1 The main character will only react to objects within his visual cone.

3.2.2 The main character will react to enemies if they are close enough whether they are in the cone or not.

3.3 The main character will sometimes find enemies.

3.3.1 The main character will stop what he’s doing to fight them.

3.3.2 The main character will attack constantly but will sometimes block an enemies attack using a shield or weapon.

4. The player will use a secondary character.

4.1 The secondary character will be able to fly freely around levels.

4.1.1 The secondary character will use flight to reach objects and gain a better view of the environments.

4.1.2 The secondary character will not be able to leave the main character.

4.2 The secondary character will be able to shine a light.

4.2.1 This light will alert the main character to objects or enemies.

4.2.2 This light can also be used to support the main character in combat.

4.3 The player will have to keep track of the main character and some stats.

4.3.1 The main character will have a trust meter that determines how much it listens to the player.

4.3.1 The main character will fall in pits or activate trap if he’s not warned by the player.

5. A vr headset will be usable with the game.

5.1 The vr camera will be in first person.

5.1.1 Vr controllers will be used to control the hands of different characters.

5.1.2 The vr camera will not be necessary to play the game.

5.2 The vr headset will change certain gameplay elements.

5.2.1 If the vr headset is attached the the player will have the option to help in combat by turning enemies against each other.

5.2.2 When turning enemies against each other the camera will become the first person view of the enemy and you will control their motions.

5.3 The vr headset will also allow you to take control of the main character briefly.

5.3.1 This option will only be available when the main character is fighting enemies and the enemies are low on health.

5.3.2 This simulates automatic kill moves in other games but it allows the player to control it using the controllers.

# 2.2 Non-Functional Requirements

1. The game will be developed using the Unity Engine.
2. The game will be programmed and coded with the C# programming language.
3. The game will be played using keyboard and mouse controls or Oculus touch controls.
4. The game will be programmed to be compatible with the windows 10 operating system.
5. The system will allow the user to play the game using the Oculus rift headset.
6. Frame Rate: The minimum frame rate in the game must be a minimum of twenty frames per second. The average frame rate must be greater than 30. The frame rate can also be monitored directly from the Unity graphics engine.
7. Platform: The game must run in Windows 10. Installing the game in a Windows 10 environment and run simple tests to verify if the game properly works can test this requirement.
8. Response Time: The average response time between click and reaction needs be less than 5 milliseconds. The maximum response time between click and reaction must be 2 seconds. Adding some simple classes and methods that can process and display the time needed to process operations can be used to test this requirement.
9. Required Resources: The game should be able to run with minimum of 2 GB of RAM. The game must use less than 2 GB of hard disk space. Checking the total size of the folder in which the game was installed, for the hard disk space can test this requirement. To monitor the amount of RAM being used when playing the game, the physical memory in the Windows Task Manager can be used.
10. User Interface: User interface should be clean, presentable, and compatible with both a PC screen and Oculus screen.