DESCECI2

FYP Proposal

VR Game Development using Unity and Mobile Sensing Device

by

WONG Ka Chun, MOK Yat Shing, PAN Hsuan En and LUI Ka Ming

DESCECI2

Advised by

Dr. TSOI Yau Chat Desmond & Dr. CHAN Ki Cecia

Submitted in partial fulfillment of the requirements for CPEG 4901

in the

Department of Computer Science and Computer Engineering

The Hong Kong University of Science and Technology

2022-2023

Date of submission: September 16, 2022

Table of Contents

Introduction *5*

1.1 Overview **5**

1.2 Objectives **6**

Objective 1: Build a network game for dual-player cooperating via VR and mobile devices 6

Objective 2: Maximize the immersive experience for the VR player 6

Objective 3: Maximize the gaming experience for the mobile player 6

Objective 4: Provide diverse gameplay modes 7

1.3 Literature Survey **7**

DAVIGO 8

Acron: Attack of the Squirrels 8

InVokeR 11

Literature Survey Summary 11

Methodology *13*

2.1 Game Design **13**

2.1.1 Storyline 13

2.1.2 Character Control 14

2.1.2 Game Flow Overview 15

2.1.3 Game Attribute 16

2.1.4 Mobile player 20

2.1.5 Game Field & Boss 22

2.2 Implementation **26**

2.2.1 Game Engine 26

2.2.2 VR Headset and Mobile Sensing Device 27

2.2.3 Control Interface 30

2.3 Testing **33**

2.4 Evaluation **36**

Project Planning *37*

3.1 Division of Work **37**

3.2 GANTT Chart **38**

Hardware and Software Requirements *39*

4.1 Hardware Requirements **39**

4.2 Software Requirements **39**

References *40*

Appendix *41*

A: Meeting Minutes **41**

Introduction

1.1 Overview

In 1950 the first video game called "Spacewar" was born, and a new era for the gaming industry began [1]. From 1-dimensional to 3-dimensional virtual spaces, computers to handheld devices, people are committed to taking video games to an unprecedented level. In 2022, with the advancement of game engines and motion-sensing devices, a new storm called “Virtual Reality” (VR) raids the industry once again. VR appears to be a secret key founded for opening a gate to a new dimensional world, and players can interact with video games in a more realistic and immersive way, elevating gameplay to a higher level and popularizing VR games within the industry [2].

According to statistics [3], the size of VR gaming revenues in the United States reached US$400 million in 2017. Still, the value grew exponentially to US$1.1 billion within three years, and it is expected to reach US$2.4 billion by the end of 2024. While the global VR gaming market is much larger and is expected to grow even faster, analysts have estimated that the valuation of US$11.56 billion in 2019 will reach US$92.3 billion by 2027 [3]. This infers a compound annual growth rate of 30.2%. Therefore, VR gaming is arguably the future of gaming.

Although VR games are becoming more popular, the growth rate tends to be lower than that of the computer or mobile games [4]. It turns out that most of the VR games on the market are not cross-platform. People are limited to using VR headsets as the only equipment for playing the game. Meanwhile, by indicating the previous final year projects and literature review, the few cross-platform VR games available on the market are mostly player versus player (PVP), so even fewer are player versus environment (PVE). Therefore, there is a niche for cross-platform cooperative VR games.

In this project, we decided to make a cross-platform collaborative role-play game (RPG) can be enjoyed through VR devices and mobile phones. The gameplay consists of two players with two different devices, carrying distinct responsibilities when trying to defeat the boss. In addition, instead of using VR controllers for tracking the movement of the VR player, a more novel method called "gesture recognition" will be utilized. As the hand is a derivative of the human body, interacting with virtual objects with the player's real hands is predicted as the most effective way for an immersive gaming experience [5]. With our unique design of controls and collaborative game flow, hopefully, we can transcend the gaming experience to another level.

1.2 Objectives

The project aims to create a 2-player cross-platform RPG that allows players to interact and cooperate to complete a shared task. In light of this, the main objectives are listed below:

Objective 1: Build a network game for dual-player cooperating via VR and mobile devices

The first objective is to create a two-player cooperative game where both players are connected via the internet. One player will be wearing a VR headset, while the other will be connected using a mobile phone. Throughout the game, these two players will have to work together to defeat monsters and survive. During the process, both players will grow to understand each other better and finally defeat the big boss as a collaborative effort.

Objective 2: Maximize the immersive experience for the VR player

Unlike traditional VR games, which use controllers to control the character's movements and actions, we use gesture recognition instead. This enables the VR player to directly use “body language” to interact with the game. In summary, it provides a more immersive gaming experience for players to view and immerse in the virtual world as if it is reality.

Objective 3: Maximize the gaming experience for the mobile player

In the game, players will be provided with a virtual joystick, so that players can control the movement of their main character. Mobile players can also adjust the character viewpoints by touching the mobile phone screen. One thing to note is that the perspective is different between the mobile player and the VR player, so the mobile player can freely adjust their perspective without being affected by the VR player. Mobile players can also assist VR players by picking up different items. In addition, mobile players can restore terrain destroyed by the monster.

Objective 4: Provide diverse gameplay modes

In the game, options for different attributes of magic and a defense system will also be available. Therefore, VR players can choose between different magic attacks or setup a defense system to defend himself/herself from the monster’s attacks. On the other hand, the mobile player will be able to maneuver around to pick up items such as magic potions or vials to restore the character’s HP, while dodging attacks at the same time. By providing different roles and responsibilities within the game, we believe that players will be more hooked to the game and willing to try out all available possibilities.

1.3 Literature Survey

Based on our game idea, several important elements and concepts have been extracted as the keywords for searching the literature review. Similar games having the feature of cross-platform or gesture recognition will be discussed. The benefits and drawbacks of each game will also be compared.

DAVIGO

DAVIGO [4] is a combat-based game that can support both VR and computer platforms. In this game, the computer player plays the role of a warrior fighting with the giant personated by the VR player. As shown in *Figure 1*, the face with two hands floating in the sky represents the VR player. During the game, the VR players play against one to four PC players. The giant can use both hands to tap things, slap the ground, throw rocks at the computer player and even grab the computer player. On the other hand, computer players can move very fast to sidestep the attack, place ground traps or shoot rockets to deal with the giant (*Figure 2*).

Figure 1: The VR player and computer player can also play in the game.

Figure 2: computer player can attack the VR player

Figure 3: Terrain was destroyed

The destroyable terrain makes the VR player feel like becoming a gigantic boss (*Figure 3*). Such a dynamic and flexible map would increase the diversity and playability of the game, which can increase players' excitement and make the game more realistic.

However, there are some issues in the DAVIGO. The single nature of the map is one of the restrictions. Lack of displaying Health Points information is another problem that may lead to confusion. In addition, the scene will become chaotic as the round lasts longer since the destroyed terrain cannot be restored until the round ends.

Acron: Attack of the Squirrels

Acron: Attack of the Squirrels [6] is one of the cross-platform games that can bring VR and mobile players together. One VR player takes the role of an ancient tree as the sole protector of the golden acorns. Meanwhile, one to eight mobile players incarnate as rebel squirrels for stealing the golden acorns by using an arsenal of unique abilities.

Figure 4: The mobile player can use a virtual joystick to control the movement of the character

Referring to *Figure 4*, mobile players control the movement of the squirrels through the virtual joystick, which is a way to engage mobile players in the world of gaming. The control strategy for this game is very similar to this project since both games are expected to use a virtual joystick for mobile control and a cross-platform for VR and mobile devices.

Figure 5: The mobile player can build up some terrain

One feature that is being eyes captured is that the mobile player can build the terrain freely, which creates a space for the player to establish their winning strategy and make the gameplay more diverse.

InVokeR

InVokeR [7] is a one-on-one action game on the VR platform in which players can cast a variety of combinations of offensive and defensive spells via the hand gesture. It provides both PVP and PVE modes. As shown in *Figure 6* and *Figure 7* respectively. Their objective is to defeat the enemy with different attributes of spells.

Figure 6: Casting spell by using hand gesture

Figure 7: The non-player character was attacked by the players

The game has four types of attributes: Cyclone, Earth, Ice and Fire. The player with better observation would have the advantage to break the enemy's spell with the counter attribute of the spell. And this mechanism has added a growing element to the game, which means the player gains more experience in each combat of the game, this is the reason that attracts people to keep playing this game.

In addition, different physical effects add to different magic rise up the excitement of the game. For instance, the magic of the "fireball" shown in *Figure 8* acts like an elastic ball that keeps bouncing between the wall. This element's presence adds to the player's excitment as the path of magic is unpredictable.

Figure 8: The magic of the "fireball"

Yet, the limitation of the character's movement is one of the problems that need to be adjusted. It is clumsy that the player is restricted to moving left or right when resisting or dodging the attacks. Besides, hard to aim at the target as the size is too thin.

Literature Survey Summary

	Pros	Cons
DAVIGO	● Cross-platform (VR and computer player) ● The terrain can be destroyed	● Only have PVP mode ● There is no indication of HP, etc.
Acron: Attack of the Squirrels	● There are clear rules of play and winning conditions ● Cross-platform(VR and mobile) ● Support for multiple connections (2 or more)	● Only have PVP mode
InVokeR	● Using the technique gesture recognition ● Have both PVE and PVP mode ● There is stat specific gameplay, so there are different combinations of spells	● Long distances between players ● Limitation of the movement of the player

Table 1: A summary of the pros and cons of related work

To summarize, different cross-platform games does provide a brand new gameplay experience to the user. However, most of them in the commercial market are mainly PVP. There is a gap in developing a PVE game where players can cooperate with each other. Taking the references above, those strengths will be considered into this project. Such as dynamic terrain, attributes for magic and diverse physical effects.

Meanwhile, this project is going to break the rule that VR gesture games must be stationary, a natural free movement will be introduced to the player. We believe that this can add extra dynamics to the gameplay, and offer a richer gaming experience to the players

Methodology

2.1 Game Design

2.1.1 Storyline

In 1986, the No. 4 reactor in the Chernobyl Nuclear Power Plant exploded. The large energy from the explosion opened a gateway to another dimension called the upside down. This crack causes 2 dimensions to collapse, and an ancient monster, Doomwraith, from upside down invades the world.

The main character is an ordinary kid from a carpenter family. One day he/she accidentally fell into a temple and freed the soul of Hopperfield, the greatest magician of all time, and discovered the threat coming to the world.

Hopperfield had fought with monsters before and banished them upside down. He is the only one who can defeat Doomwraith. However, he is already dead physically. Only a trace of his soul is trapped in the temple. Despite Hopperfied own vast knowledge of magic spells that can defeat the Doomwraith and its subordinates, the loss of the physical body causes him to no longer be able to form the mana for the spell casting and move away from the temple.

The appearance of an ordinary kid gives Hopperfield a glimmer of hope for saving the world that is being destroyed. So the main character decided to let Hopperfield possess his/her body. And together, they are going to save the world.

2.1.2 Character Control

Generally, there are two main players in the game: the mobile player and the VR player. As described by the storyline, the mobile player will act as an ordinary kid that is being possessed by the soul of Hopperfield, while the VR player will impersonate Hopperfield. *Figure 9 i*s the idea diagram for reference.

Figure 9: Draft for the character design

Since only a trace of the soul exists in the world, Hopperfield is relies on the ordinary kid as a medium to live in the world. But this also means the ordinary kid restricted the movement of the Hopperfield. Thus, when the VR player is acting as Hopperfield, the movement of the VR player would depend on the movement of the mobile player. In other word, two players are stuck together, sharing the same location in the game. Yet, the viewing angles of both players are not necessarily identical, and the VR player can control the viewing angle by head movement while the mobile player can drag the mobile screen to change the vision.

To cope with the great mission of defending the world from being destroyed, the cooperation of players is an essential prerequisite. As mentioned in the storyline, although Hopperfield knows magic, he struggles to form power and movement. Thus, the mobile player is needed to help the Hopperfield for collecting the mana for activating the magic and moving for Hopperfield to attack or dodge the enemy. At the same time, the VR player will be asked to pose different hand gestures to implement the spells for repulsing the enemy. Therefore, both players are interdependent.

2.1.2 Game Flow Overview

The game is mainly divided into three parts. The first part is the tutorial, the second part is the mini level type and the last part is the high level where there is a final boss.

Firstly, two tutorials will be prepared for different device players. One of them will be used to teach the mobile player to get familiar with the character's movement. Another one will be operated by the VR player, where the VR player is guided through different gestures to cast spells or skills.

In addition, the second part of the mini-levels will focus on the mobile player, with the VR player as the secondary player. At this level, the mobile player must do parkour or bypass obstacles to get through the trap. At the same time, it is the responsibility of the VR player to shoot down small monsters to avoid the monster's nuisanceto the mobile player.

Once the player has cleared the mini level and got into the dungeon's depths, the ultimate mission will be triggered, repelling the final boss. To survive under the raid of the boss, cooperation between the VR player and the mobile player is necessary. For example, the mobile player has to dodge the attack and the VR player has to knock back the monster. Both of them are essential for repulsing the enemy.

2.1.3 Game Attribute

The fowing paragraph will introduce and define several terminologies in the game.

Health Points (HP)

Health Points (HP) are used to determine the maximum damage the character can take before dying. This can be reduced by the attack from the enemy or falling into a trap. But it can also be replenished by using the collectible items called “Life Potion”.

To synchronize with the storyline that the survival of Hopperfield depends on the ordinary kid, both players are designed to share the same amount of HP, so whenever the character is being attacked, the HP for both players would be reduced, and vice versa.

Mana Points (MP)

Mana is a necessity for the VR player to activate the magic. While Mana Points (MP) is an attribute for specifying the remained mana that is consumable, which will restrict the number of spells that the VR player can be casting. But, this also can be recovered by picking up the corresponding supplies called “Mana Potion” through the mobile player.

For the same reason mentioned above, the MP showing on both players will be the same.

Attribute

The feature of type matchup is expected to be implemented in this project, meaning there exist modifiers that are applied to various attributes of magic when attacking monsters of each type.

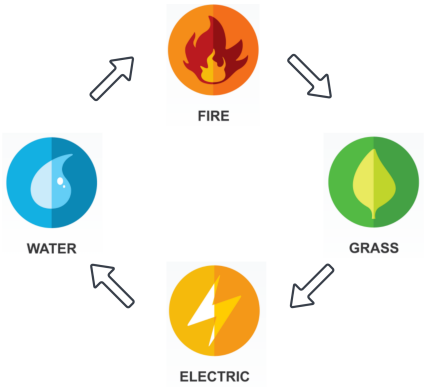


Figure 10: Type chart

As shown in *Figure 10*, four kinds of attributes are designed in the game, they are "Fire", "Water", "Grass" and "Electric".

	Defender's Attribute				
Attacker's Attribute					
		x1.0	x1.0	**x1.5**	x1.0
		**x1.5**	x1.0	x1.0	x1.0
		x1.0	x1.0	x1.0	**x1.5**
		x1.0	**x1.5**	x1.0	x1.0

Table 2: Type matchup table

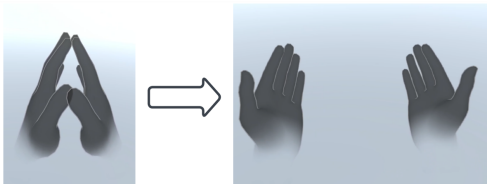
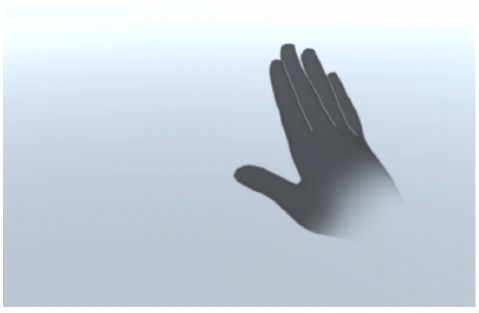
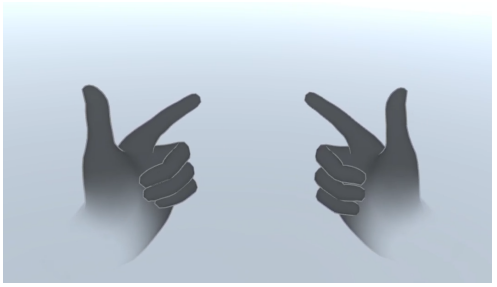
The value of the modifiers can be found in *Table 2*. There will be an advancement of damage by 1.5 times when the player uses the spell with a beneficial attribute. However, there is no damage reduction on the inferior attribute attacking the superiority attribute. Therefore, consider using the attributes against the opponent one as it can be a heavy opportunity for the player to deal with great damage.

2.1.4 VR player

The VR player acts as Hopperfield, who is allowed to cast the spell for attacking and defending the monster. With the function of hand pose detection provided by the VR headset, the player can pose several gestures with their hands to activate the magic [8]. For a better immersive experience in casting the spell, the first-person perspective is preferred for the VR player.

Hand Pose

The following table has several gestures and movements designed for the VR player to implement the spell.

Action	Description
 **Initialization** Motion description: Put one's fingers together, then separate hands slowly in a horizontal direction	• Summon the magic book in front of the VR player, representing the initialization of the cast spell • The player is able to choose the attribute of the magic by switching the page of the magic book • It is required to summon the magic book before casting any other spell • Player is allowed to switch to others attributes more than once during the mission
 **Switch**	• Switch the attribute of the magic by posing the gesture on the right side and moving horizontally to left or right • The color on the book will indicate the attribute that the player is selected, red represent fire, green represent glass, blue represent water and yellow represent thunder
 **Attack**	• Launch the ranged attribute attack • The magic will project to the places pointed by the index finge • The path of the magic attack would be in a linear or parabolic way, based on the attribute chosen by the player • It would cost 5 mana when activating the magic • Each offensive action would be separate by 0.5 second cold down

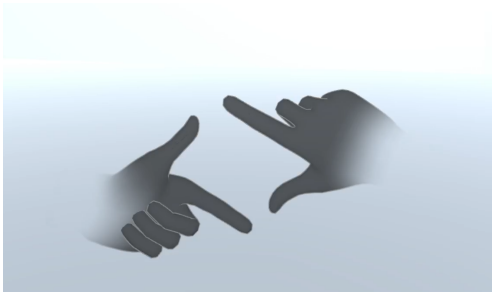
 **Defense**	● Open an instantaneous barrier that covers players for protecting the attack from the enemy in any direction ● 50 mana would be consumed for launching the barrier ● The cold down would be 20 seconds

Table 3: Hand poses for the VR player

2.1.5 Mobile player

Taking advantage of modern touch control on mobile devices, the mobile player can effectively manipulate the character, which is best for controlling the main character's movements. The detailed controlling method will be discussed in session 2.2.3 of implementation.

The main character, acting as a support to the Hopperfield. His duties are to take the VR player to a designated position via actions (*Table 4*) and gather the collectible items (Table 5) randomly spread on the map.

Actions	Description
 **Movement**	• Position the character into a favorable location, allowing the VR player to attack the weakness of the monster • Go around the game field to gather some collectible items • Dodge the monster's attacks
 **Jump**	• Dodge the monster‘s ground attacks, which can only be dodged by jumping • Reach a higher ground with the parkour skill
 **Build**	• Repair the connection (ladder, bridge) between two islands, which is damaged by the monster • Build a new connection between two islands • Build action costs some woods, the quantity depends on the importance of the connection

Table 4: Actions table

Collectible items	Description
 **Wood**	• The material for repairing/building the connection between islands • Carrying capacity: 10 • Each build action costs 2-3 woods
 **Life Potion**	• Restore the health of the character • Carrying capacity : 3 • Regeneration potion: restore 10% health by every 2 seconds, last for 10 seconds • Healing potion: instantly restore 25% health
 **Mana Potion**	• Restore the mana of Hopperfield • Carrying capacity: 5 • Mana potion: instantly restore 40% mana • Great mana potion: instantly restore 60% mana
 **Spell Book**	• 4 kinds of spell book with respect to 4 attributes (fire, water, grass and electric) • The spell book can level up the spell, the upgraded spell can deal more damage and cost less mana

Table 5: Collectible items table

2.1.6 Game Field & Boss

In coherence with the storyline, our game will take place in a desolate area with a post-apocalyptic atmosphere. Below are preliminary visual demos we have created: sporadic vegetation, massive mutated monster Doomwraith (white clay giant as placeholder), and our hero Hopperfield.

Figure 11: Capture for the boss

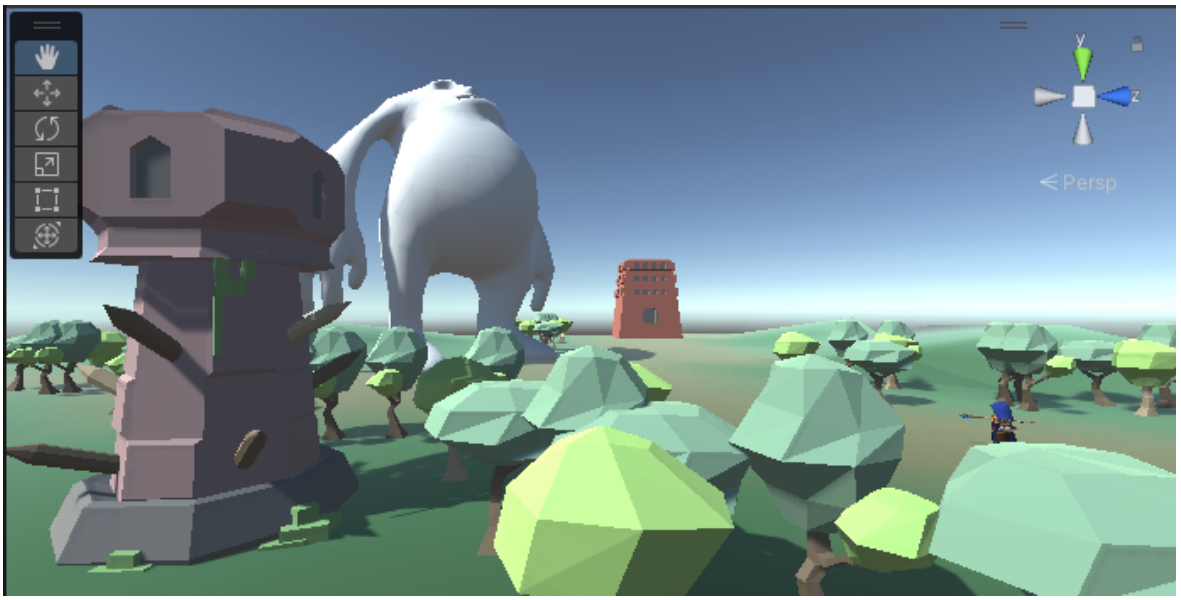


Figure 12: Capture for the boss

The evil boss, Doomwraith, will be a ginormous mutated monster that spits fireballs and attacks with its whip-like tentacles. We will implement a rule-based system to decide the actions for Doomwraith. The mechanism will be similar to the following chart:

	Long-range attack (fireballs)	**Short-range attack (tentacle whips)**
HP >= 70%	Speed: 1S, Damage: 1D	Speed: 1S*, Damage: 1D*
30% <= HP < 70%	Speed: 1.2S, Damage: 1.2D	Speed: 0.8S*, Damage: 1.2D*
HP < 30%	Speed: 1.5S, Damage: 1.5D	Speed: 0.6S*, Damage: 1.5D*

Table 6: Doomwraith's attack mechanisms

Note: S, D, S*, D* in the chart above are variables that can be set to certain values to initialize the speed and damage incurred by attacks from Doomwraith. They may also differ according to the selected difficulty of the mission.

2.1.7 Graphical Style

There are mainly three graphics styles in the market, Cube/Voxel design, low-poly design, and photorealistic design, as shown in *Figure 13, Figure 14,* and *Figure 15* respectively. After researching the pros and cons of these three graphics styles and considering our use case, we have decided to choose a low-poly design for our project.

The performance and the development cost are the most important factors of our project. The photorealistic design has the most realistic style overall, but it came with a longer development time on 3D models and textures. And the pre-modal assert in the market is too costly. Considering our budget and time, a photorealistic design might not be suitable for our project. Furthermore, our objective is a real-time multiplayer game. The photorealistic design has a long rendering time. One of the players is running on a mobile device with limited hardware, which might be an issue when rendering complicated models and high-resolution textures.

That leaves us no choice but to choose either Cube/Voxel design or a Low-poly design. We choose the Low-poly design over the Cube/Voxel design because it can provide a much better immersive experience to the player. Although low-poly models are made of polygons, it provides enough details to distinguish different objects. So, we can create a relatively realistic and detailed game field and boss, and let the player immerse in the exciting fight.

Figure 13: Cube / Voxel design

Figure 14: Low-poly design

Figure 15: Photorealistic design

2.2 Implementation

2.2.1 Game Engine

A game engine is a framework and development software that includes relevant libraries while offering tools [9]. The goal of a game engine is to provide an integrated environment, which reduces the costs and complexities of constructing a video game. For our project, we expect that using a suitable game engine will allow us to build a high-quality product without incurring immense monetary and time costs.

This section will briefly compare the 2 most popular game engines for VR development, Unity and Unreal Engine [10]. When choosing which game engine we should use, we consider the following characteristics of both game engines respectively:

a. proprietary or open source
b. documentation and community
c. supported programming languages
d. support on hardware

	Unity	**Unreal Engine**
Invention	2005	1998
Developer	Unity Technologies	Epic Games
Open Source	No	Yes
Documentation (Official tutorials)	800	115
Community (Subreddit members)	200k+	~100k
Programming language	C#, Prefab	C++, Blueprint
Target audience	Indies, coders	AAA-game studios

Table 7: Comparison table between Unity and Unreal Engine [11]

We aim to build a cross-platform game with a development timeframe of 7 months. Due to the abundance of pre-built gaming assets, available plugins, and documentation, we chose Unity as the game engine with which we will be building our game.

2.2.2 VR Headset and Mobile Sensing Device

Sensing device is crucial element would directly affect the interaction between games and players. Having a thoughtful strategy in the selection of it is vital. As the prototype of the game was introduced, two players were required to cooperate to control a single character in the game. The sensing device for VR and mobile players are needed to be considered.

VR Headset

The choice of VR headset models is an essential starting point for this project. Related work is necessary to select the most appropriate VR glasses for our project usage. As mentioned, hand pose detection is one of the functions that will be implemented. The VR glasses we purchase must be able to support this function. The following *Table 8* is a comparison table for the most common VR glasses that fulfill our exclusive requirements.

Model	Pros	Cons
Oculus Quest 2	● LCD 1832 x1920 display ● Refresh rate up to 120 ● Standalone VR headset ● Full software library ● Affordable price	● Short battery life
Oculus Rift S	● LCD 1440 x 1280 display ● Full software library ● Affordable price	● Refresh rate up to 80 ● Require cables connect to PC
Sony PlayStation VR	● Refresh rate up to 120 ● Support non-VR games ● Affordable price	● LCD 1080 x 960 display ● Require additional camera be installed for detection ● Motion-tracking is affected by the brightness of the environment ● Require cables connect to PC
Valve Index VR Kit	● LCD 1600 x 1440 display ● Refresh rate up to 120 ● High accuracy in motion-tracking ● Tracking individual finger movements	● Expensive ● Base stations were required to set up for motion-tracking ● External device need to buy for wireless connection between VR headset and PC

Table 8: A summary of the pros and cons of potential VR glasses [12] - [14]

Comparing the benefits and drawbacks of the popular VR headset models in the commercial market, it was found that "Oculus Quest 2" would be the best choice for this project development. There are two primary reasons leading us to make such a decision. The first one is due to numerous resources that exist on the internet that can be found for supporting the Oculus VR headsets. For example, the software development kit developed by the Oculus company called "Oculus Integration" is one of the tools many Unity VR developers have widely used [15]. But most importantly, Oculus Quest 2 is a standalone VR headset that has a processor inside the glasses that can be used to execute the application without any connection to an external mainframe [12]. This is suitable for the public who are used to running a minigame like the one this project will develop.

Gesture Recognition

To maximize the immersive experience of the gameplay, it is decided to utilize hand pose detection instead of employing a VR controller for sensing the player's hand movement. This allows the player to interact with the virtual object naturally and directly.

Figure 16: Oculus quest 2 (Arrow pointing to inside-out cameras)

To implement an objective of hand pose detection, the inside-out cameras on Oculus Quest 2 would be utilized. As shown in *Figure 16*, there exist 4 cameras around the front of the VR headset for detecting the environment data [12]. Based on collected information combined with image processing, location of knuckles and fingertips can be estimated in real-time [16]. Thus, the position and orientation of the player's hands and fingers can be implemented.

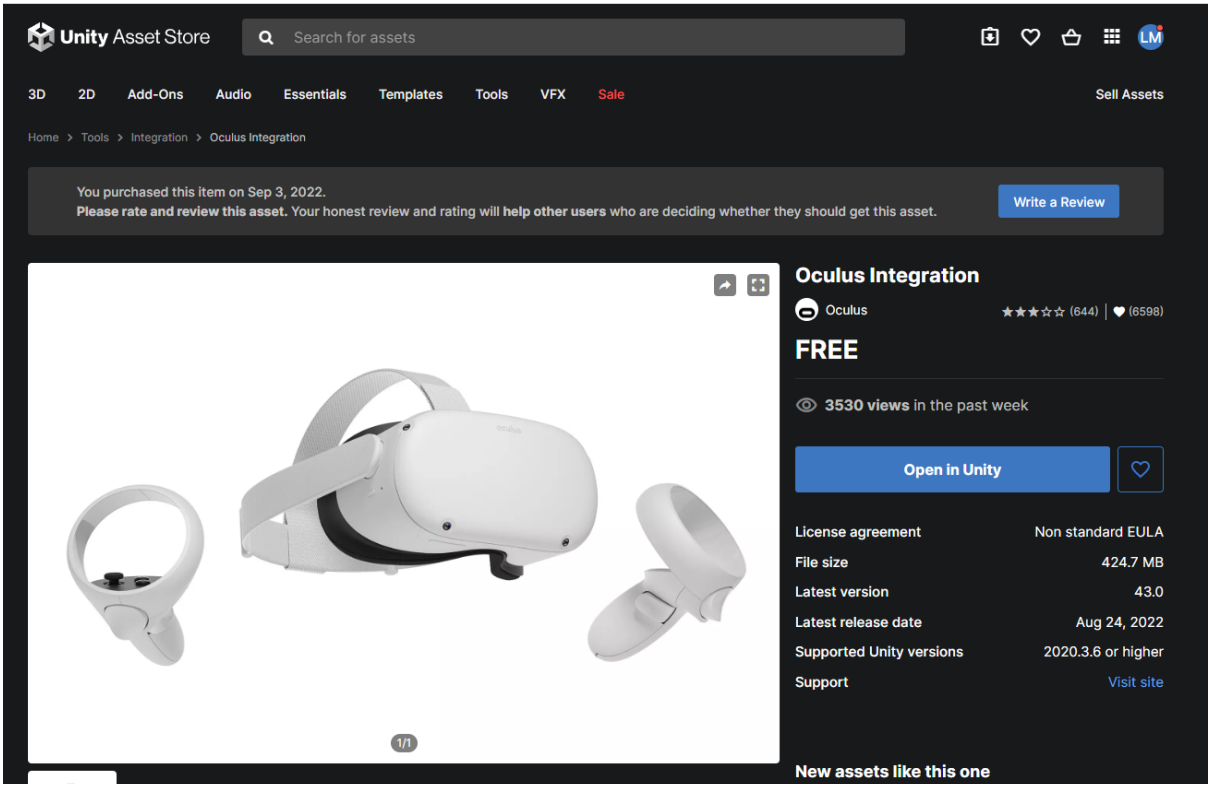


Figure 17: Screenshot on Unity Asset Store [15]

With powerful software development tools provided by Oculus, the unity asset tools as shown in *Figure 17* called "Oculus Integration" would be applied in this project [15]. This allows us to employ a trained model for gesture detection.

Mobile Phone

For the mobile player, considering the convenience and compatibility, there is no external sensing device required for capturing the player action. The mobile player would primary utilize the touch screen as the medium for interacting with the game's character. For the control detail, please refer to session 2.2.3 in implementation.

2.2.3 Control Interface

Because of our game's uniqueness, a couple of players would need to use diverse devices to cooperate to act as a single character in the game. Hence, disparate game control strategies and interfaces would be offered to VR players and mobile players.

	VR Player	Mobile Player
Game Interface	3-Dimensional	3-Dimensional
Viewpoint	First-person perspective	First-person perspective
Viewing Angle Control	Head movement	Drag the mobile screen
Movement Control	/	Press the joystick on-screen bottom left side
Attack & Defense Control	Posing hand gestures	/

Table 8: A summary of the control interface

Both players will act in a 3-Dimensional world of the player's character and view the environment form the first person perspective. The mobile player will rely on triggering the bottom on the mobile screen for controlling the movement as well as recovering the game character, while the VR player will be based on the gesture that is posed for attacking or defending.

Mobile Player Interface

The mobile player controls the main character via the on-screen interface. As shown in *Figure 18*, the player can either tap the icons to use the items, or drag the joystick/screen to move the character/ character's view.

Figure 18: mobile interface designed by ourselves

1. **Minimap:** for players to orient themselves in the game field
2. **Status Bar:** display the health and mana of character
3. **Move Controller:** control the movement of the character
4. **Items:** display the carrying items, tap to use an item
5. **Build Button:** lights up when there is a buildable connection around the character. Tap to build the connection
6. **Jump Button:** perform jump action
7. **View Controller:** half of the right screen is used to control the view of the character, eg. dragging from left to right to turn the character's view to the right

VR Player Interface

The VR player performs all the actions via hand gesture, and some sets of designed hand gestures to cast the a spell and change the attribute are discussed in session 2.1.3 of Game Design. And the status of the currently selected attribute and some other game information are displayed in the VR player interface, as shown in *Figure 19*.

Figure 19: VR interface designed by ourselves

1. **Gesture modal:** display the state of the player's gesture
2. **Spellbook:** showcase the current selected attribute
3. **Weakness:** indicate the critical spot of the monster
4. **Projectile Preview:** indicate the projectile of the spell. Every spell has their unique projectile

2.3 Testing

Hand Pose Detection

One of this project's main features and the objective is to use hand gesture detection for spell casting. Thereby testing the function of hand pose recognition is necessary.

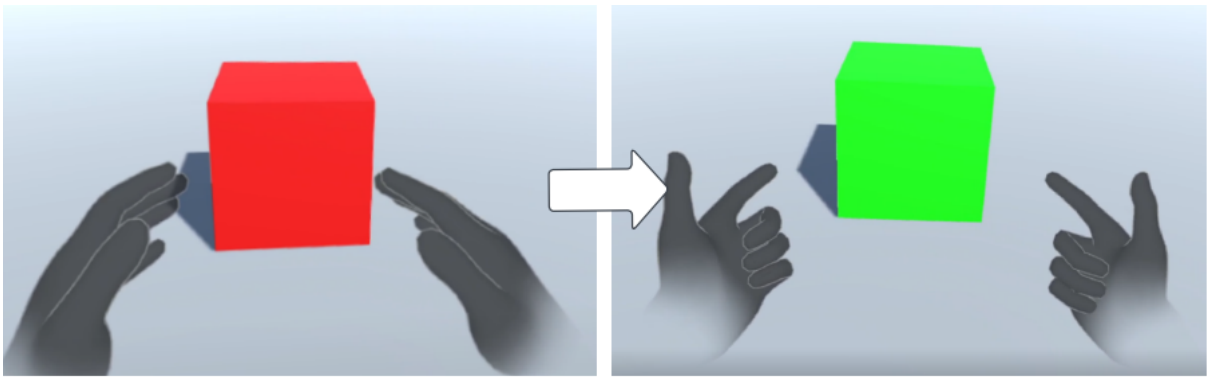


Figure 20: Utilizing the hand gesture for changing the box color

For testing the feasibility, an experiment has been made as shown in *Figure 20*. The color of the box appearing in the center will be changed from red to green when the "attack" gesture is detected. This demonstrates the idea of a hand pose for emitting the spell is practicable. However, several limitations of gesture design were found during the examination.

Firstly, the detection may be an error or inaccurate when both hands are merged, for instance, in the pose of Interlace one's fingers.

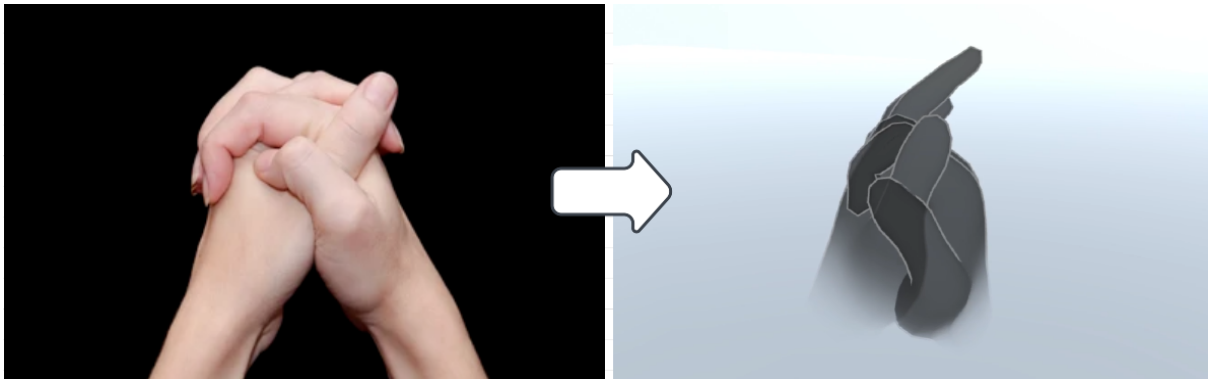


Figure 21: The detection of VR when posing the gesture of Interlace one's fingers

Observing the gesture detection function mistakenly treats the left hand as opening as shown in *Figure 21.*

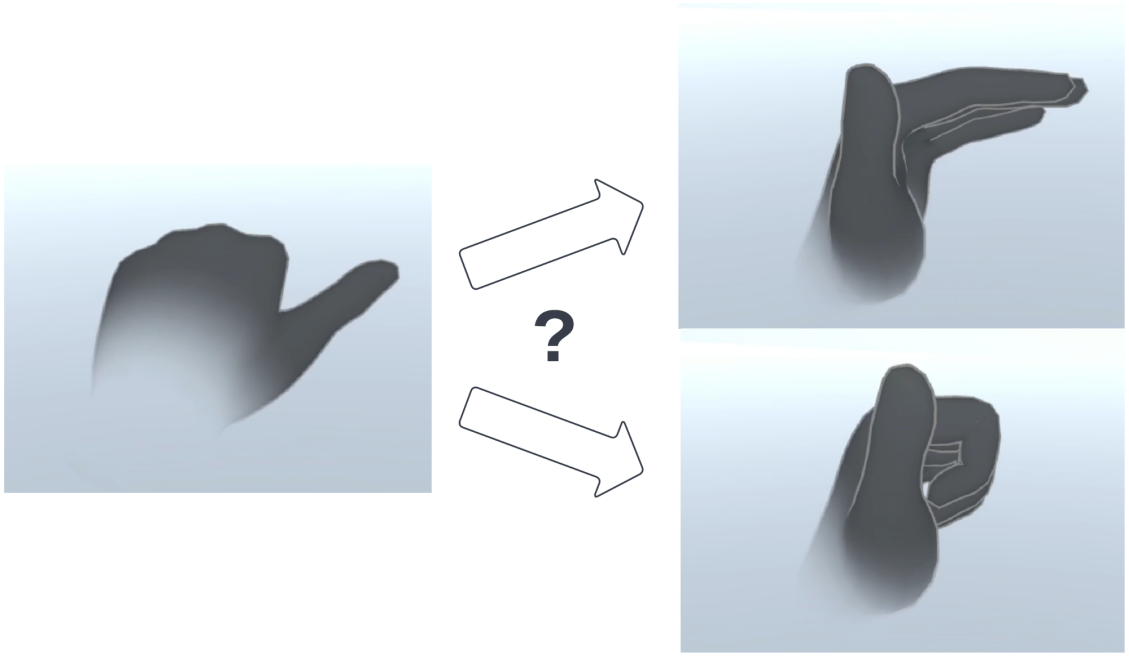


Figure 22: Indeterminate recognition happens when the inside-out camera was blocked

Moreover, there exist some cases that may lead the detection ambiguous. For example, when posing some gestures that are palm-outward. Due to the fingers being obstructed by the back of the hand, it is hard to recognize whether the player has extended their fingers. Thus, it is better to avoid using gestures that may potentially cause this situation.

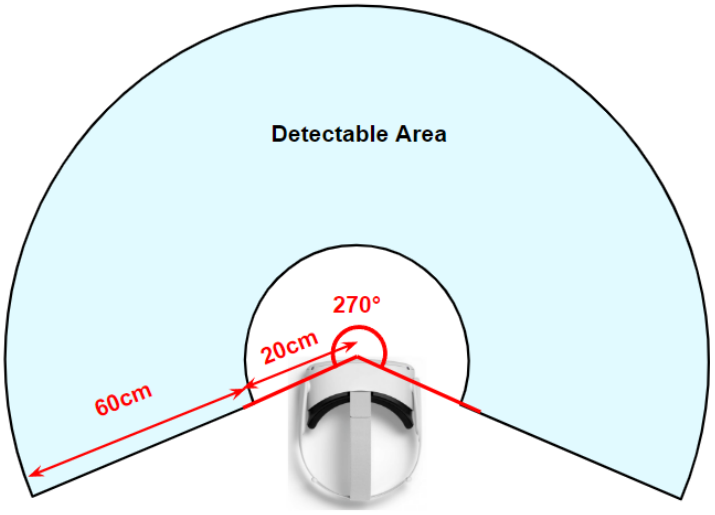


Figure 23: Detectable area for gesture recognition (Oculus Quest 2)

Besides, the range for hand gesture detection has also been measured. It is indicated that the detectable angle for the inside-out cameras is around 270 degrees, the closest distance that can be detected is approximately 20 cm and the farthest distance is about 80 cm *(Figure 23)*. Although there may exist a blind side that the hand pose detection will not function with Oculus quest 2, generally the area displayed by the VR headset will be functioning.

Network Stability

Testing the stability of the network between the VR player and the mobile player is significant. The game reaction time should be reasonably short for a cooperative and battle game. Otherwise, serious lagging would negatively affect the interaction between players and the gaming experience.

Parameter Adjustment

Adjustment to different parameters is needed to control the game's difficulty, time and balance. Including attack, defense, HP, MP, cool down, speed, jump distance, amount of items the player can hold and threshold for gesture detection.

2.4 Evaluation

The last step is to let the user evaluate our game, as the idea of sharing the same position while separating the view is very new to most people. It is important to get feedback from the user to improve our game.

We will release a beta version to our friends, and ask them the following questions to evaluate our game

1. **Control and Comfortability**: Does the VR player feels comfortable letting the mobile player control the movement? Is the procedure of casting spells too simple/ complicated? Whether the mobile player feels comfortable with the UI. Is the movement smooth enough?
2. **Game Flow**: Whether the player can learn the controls via the tutorial. Can the player notify it is a tutorial? Is the player get prepared enough before they fight with the boss?
3. **Game Play**: How is the storyline? Is it interesting?
4. **Multiplying experience**: Is there any notified latency? Is it difficult to cooperate?

Project Planning

3.1 Division of Work

T: Together L: Leader A: Assistant

		Wong Ka Chun	Mok Yat Shing	Pan Hsuan En	Lui Ka Ming
Analysis	Literature review	T	T	T	T
	Software Review	T	T	T	T
	Equipment	T	T	T	T
Design	Story and Mission	A	L	A	A
	AI Mechanism	L	A	A	A
	Hand Gesture	A	A	A	L
	Graphic Field	A	A	L	A
Impementation	3D Modeling	A	L	A	A
	VR Control	A	A	A	L
	Mobile Control	A	A	L	A
	Enemy AI	L	A	A	A
	Networking	A	L	A	A
Others	Poster Design	L	A	A	A
	Video Making	A	A	A	L
	Conduct Survey	A	A	L	A

3.2 GANTT Chart

Documentation	Design	Software	Testing

	2022						2023			
Task	**Jul**	**Aug**	**Sep**	**Oct**	**Nov**	**Dec**	**Jan**	**Feb**	**Mar**	**Apr**
Requirements Analysis										
Game Theme										
Literature Review										
Game Design										
Proposal Report										
Equipment Testing										
Software Study										
VR Control										
Mobile Control										
3D Modeling										
AI of Enemies										
Scene										
User Interface										
Progress Report										
Tuning and Debug										
Integration Testing										
Final Report										
Poster										
Presentation & Demo										
User Experience Test										

Hardware and Software Requirements

4.1 Hardware Requirements

Hardware	Specification	Use of Hardware
Oculus Quest 2	RGB LCD 1832 x 1920 Refresh rate 72 - 120 Hz	VR player testing
Tablet with Android System	Android 12 or above	Mobile player testing
Personal Computer	Windows: Windows 10 or above, 64-bit version	Game development and game execution

4.2 Software Requirements

Software	Specification	Use of Software
Unity	Version 2021.3.9f1	Game development
Blender	Version 3.2.2	Making 3D models
Oculus App	Version 43.0	Game development and game execution

References

[1] Monnens, D., & Goldberg, M. (2015). Space Odyssey: The long journey of Spacewar! from MIT to computer labs around the world. *Kinephanos: Journal of Media Studies and Popular Culture*, 124-147.

[2] Holy City Sinner, "Reasons why VR is getting more popular," *Holy City Sinner*, 18-Apr-2022. [Online]. Available: https://holycitysinner.com/2022/04/18/reasons-why-vr-is-getting-more-popular/. [Accessed: 10-Sep-2022].

[3] G. Tsagkarakis George Tsagkarakis, "Current growth and the future of VR gaming," *eGamers.io - P2E NFT Games Portal*, 01-Jun-2022. [Online]. Available: https://egamers.io/current-growth-and-the-future-of-vr-gaming/. [Accessed: 10-Sep-2022].

[4] "Davigo," *DAVIGO*. [Online]. Available: https://www.davigogame.com/. [Accessed: 10-Sep-2022].

[5] Mobility Insider April 30, "What is gesture recognition?," *Aptiv*, 30-Apr-2021. [Online]. Available: https://www.aptiv.com/en/insights/article/what-is-gesture-recognition. [Accessed: 10-Sep-2022].

[6] "Acron: Attack of the squirrels! VR game for Oculus Quest & Steam," *Resolution Games*. [Online]. Available: https://www.resolutiongames.com/acron. [Accessed: 10-Sep-2022].

[7] "Invoker," *InVokeR*. [Online]. Available: https://invokervr.com/. [Accessed: 10-Sep-2022].

[8] Huesser, C., Schubiger, S., & Çöltekin, A. (2021). Gesture Interaction in Virtual Reality: A Low-Cost Machine Learning System and a Qualitative Assessment of Effectiveness of Selected Gestures vs. Gaze and Controller Interaction. Human-Computer Interaction – INTERACT 2021, 151–160. https://doi.org/10.1007/978-3-030-85613-7_11

[9] Valencia-Garcia, Rafael; et al. (2016). *Technologies and Innovation: Second International Conference, CITI 2016, Guayaquil, Ecuador, November 23-25, 2016, Proceedings*. ISBN 9783319480244. Retrieved 2021-07-22.

[10] "The Two Engines Driving the $120B Gaming Industry Forward". *CB Insights Research*. 2018-09-20. Retrieved 2019-05-15.

[11] Evelyn Trainor-Fogleman. (2021). Unity vs Unreal Engine: *Game Engine comparison guide for 2021.* [Online]. Available: https://www.evercast.us/blog/unity-vs-unreal-engine. [Accessed: 12-Sep-2022].

[12] “Oculus device specifications: Oculus developers,” *Developer Center*. [Online]. Available: https://developer.oculus.com/resources/oculus-device-specs/. [Accessed: 8-Sep-2022].

[13] “Tech specs,” *PlayStation*. [Online]. Available: https://www.playstation.com/en-hk/ps-vr/tech-specs/. [Accessed: 9-Sep-2022].

[14] “Headset,” *Valve Index® - Upgrade your experience - Valve Corporation*. [Online]. Available: https://www.valvesoftware.com/en/index/headset. [Accessed: 9-Sep-2022].

[15] “Oculus integration: Integration,” *Unity Asset Store*. [Online]. Available: https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022. [Accessed: 9-Sep-2022].

[16] “Set up hand tracking: Oculus developers,” *Developer Center*. [Online]. Available: https://developer.oculus.com/documentation/unity/unity-handtracking/. [Accessed: 10-Sep-2022].

Appendix

Appendix A: Meeting Minutes

6.1 Minutes of the 1st Project Meeting

Date: 21 June 2022
Time: 20:30 - 21:30

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing & Pan Hsuan En

Absent: Lui Ka Ming (Due to the blackout in New Territories West after the electric cable caught fire, he is not allowed to connect the internet for the online meeting)

Items Discussed:
A brief introduction about the FYP from Desmond, including the schedule, grading scheme as well as the detail for the documentation requirement. The team members presented several ideas about the game design and received suggestions from professors.

Follow Up:

1. Explore different kinds of VR game that exists in the market
2. Find the possible development tools and software for game development

6.2 Minutes of the 2nd Project Meeting

Date: 7 July 2022
Time: 20:30 - 21:30

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Discussion on different types of gameplay strategies that exist in the market. Spotting the potential difficulty and challenge of making different types of games. For example, puzzle games, adventure games, player versus player and role-playing games, etc. Moreover, referring to previous groups for brainstorming the idea of this project.

Follow Up:

1. Do the literature review on different kinds of the game
2. Identify the niche that exists in the market
3. Re-examine our project idea

6.3 Minutes of the 3rd Project Meeting

Date: 14 July 2022
Time: 21:00 - 22:15

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Giving a presentation on several new project ideas to professors asking for suggestions. Start to define the objective and characteristics of the game. Having the discussion on the details of implementation, for instance, what kind of game engine is most suitable for developing our project? Also, conceiving the best way for controlling the game character so that the player has an immersive gameplay experience.

Follow Up:
1. Research the pros and cons of different game engines
2. Discuss more detail on the gameplay overview, objective and control method

6.4 Minutes of the 4th Project Meeting

Date: 21 July 2022
Time: 21:00 - 22:20

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Discussion on the range of the game that the team is expected to do for this project, such as the completeness of the game, how many missions, bosses, or dungeons should be done and how many functions of the game are expected to be completed. In order to complete the project timely and plainly, a prioritized list for all features of our game is made.

Follow Up:
1. Plan the timetable for the project
2. Analysis of different kinds of mobile sensing devices that exist in the market
3. Justify the graphics and game theme of this game

6.5 Minutes of the 5th Project Meeting

Date: 28 July 2022
Time: 21:00 - 22:30

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Finalizing the draft of the project idea, clarifying the traits and highlights of our game. Making a judgment on the selection of the different development tools. Deciding what kinds of mobile sensing device is suitable for the target of this project. Planning several experiments for testing the feasibility of the game.

Follow Up:

1. Purchase the VR headset as well as necessary accessories
2. Do the testing on the main objective of the game, including gesture recognition, network, and cross-platform

6.6 Minutes of the 6th Project Meeting

Date: 4 August 2022
Time: 21:00 - 22:20

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Reporting the procedure of our testing result. Besides, the team has asked different questions regarding the requirement of the proposal report. Take the reference from the previous proposal report, a detailed explanation of the structure of the proposal report given by Desmond.

Follow Up:
1. Distribute the work and set up the internal deadline for the proposal report
2. Implement the testing on character control and game field

6.7 Minutes of the 7th Project Meeting

Date: 13 August 2022
Time: 21:00 - 22:20

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
The team member found a serious problem in the implementation on player control. It turns out the software development tools that we are applying is not allow the user to control the game by using both hand gestures detection and a VR controller*. Instead, it restricted the player to use either hand gesture recognition or VR controllers for controlling the game. The team was seeking suggestions and methodologies from the professor for solving this problem.

Follow Up:
1. Search other possible software development tools that may avoid this issue
2. Apply different approaches that are mentioned during the meeting

****Remark:***
During that period, we have not come up with the idea of a cooperative game. Instead, we are expected to use the strategy of a single player that can control the character by using a single VR controller and the function of gesture recognition.

The player's left hand will hold a VR controller, which is used to stimulate the player was holding a magic wand and the joystick on it can be used to control the movement of the character. While the player can cast the spell by using the right hand to pose different hand gestures.

6.8 Minutes of the 8th Project Meeting

Date: 18 August 2022
Time: 21:00 - 22:20

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Unfortunately, we are not allowed to find a solution for solving the problem of integrating gesture recognition and VR controllers. We made a hard decision to abandon the original idea. Instead, we proposed the other two approaches as a backup plan to the professor and asked for their opinion. Finally, we decided to make a cooperative game.

Follow Up:

1. redesign the storyline, hand gestures, and battle system
2. Add more detail on the user interface, game field and game flow, etc.

6.9 Minutes of the 9th Project Meeting

Date: 25 August 2022
Time: 21:00 - 22:20

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Finalized the game direction and design, including gameplay strategy, user interface, and character control, etc.

Follow Up:

1. Finish the draft of the project proposal

6.10 Minutes of the 10th Project Meeting

Date: 11 September 2022
Time: 10:30 - 12:30

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
The team finished the first draft of the proposal report and explained the details of the proposal and projects. Desmond and Cecia gave suggestions on the context, structure, format and grammar.

Follow Up:
1. Modify the project proposal base on the suggestion that is given
2. Do the final team inner proofread on the proposal report
3. Sent to Desmond and Cecia on 14 Sep for the final re-examine

6.11 Minutes of the 11th Project Meeting

Date: 28 September 2022
Time: 16:30 - 17:40

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Mainly discussing the potential problems of VR players. Firstly, Is it difficult for the VR player to move their visual angle by turning their body? After a discussion, it seems reasonable to have little movement, but we have to consider the safety of the gameplay, thus the guideline in the player's physical setting is needed. Also, we may design a gesture to reset the VR visual angle for convenience purposes. Second, virtual reality sickness can be one of the challenges we may face. Slowing the speed of changing scenes or adding some transformation between different scenes can be the solution.

Follow Up:
1. Establish the co-working environment for the project development in GitHub
2. Set up the individual milestone to every teammate

6.12 Minutes of the 12th Project Meeting

Date: 12 October 2022
Time: 16:30 - 17:20

Present: Dr. Desmond Yau-chat Tsoi (supervisor), Dr. Cecia Ki Chan (co-supervisor), Wong Ka Chun, Mok Yat Shing, Pan Hsuan En & Lui Ka Ming

Absent: None

Items Discussed:
Reporting the current progress and asking the suggestion on the direction. One teammate is working on developing the networking by doing the synchronization test between pc and phone. Another teammate is working on an AI boss by establishing the boss model and adjusting the animation. Asking the question regarding the limitation of Oculus SDK, It is advised by the professor for searching more libraries in gesture recognition for continuous movement detection.

Follow Up:
1. Find the library for continuous movement detection
2. Test on mobile and pc synchronization