

GNG 1103

**Design Project User and Product Manual**

**Assembly of a 3d Virtual Reality Game in Unity**

Submitted by:

Group 13 – Lucky 13

Carter Yue, 300297537

Sam Meyer, 300380141

John Riccio, 300367557

Ryan Athauda, 300374212

December 13, 2023

University of Ottawa

# Table of Contents

---

Table of Contents .....	ii
List of Figures .....	iv
List of Tables .....	v
List of Acronyms and Glossary .....	vi
Introduction.....	7
Overview.....	8
Cautions & Warnings .....	9
Getting started .....	9
Configuration Considerations .....	10
Accessing/setting up the System.....	10
System Organization & Navigation .....	11
Physical Components .....	11
Exiting the System .....	11
Using the System.....	12
Troubleshooting & Support .....	12
Error Messages or Behaviors .....	12
Special Considerations .....	12
Support .....	12

Product Documentation.....	13
BOM (Bill of Materials) .....	13
Equipment list.....	15
Testing & Validation.....	16
Conclusions and Recommendations for Future Work .....	18
Bibliography .....	20
APPENDICES .....	21
APPENDIX I: Design Files .....	21

# List of Figures

---

Figure 1. Final Prototype .....	8
Figure 2. Oculus Quest 2 .....	9

## List of Tables

---

Table 1. Acronyms .....	vi
Table 2. Glossary .....	vi
Table 3. Bill of Materials .....	13
Table 4. List of Equipment .....	15
Table 5. Prototype 1 Test Plan .....	16
Table 6. Prototype 2 Test Plan .....	17
Table 7. Prototype 3 Test Plan .....	18
Table 8. Referenced Documents .....	21

# List of Acronyms and Glossary

---

**Table 1. Acronyms**

<b>Acronym</b>	<b>Definition</b>
BOM	A list of every asset used in building our game
VR	An artificial environment generated by a computer

**Table 2. Glossary**

<b>Term</b>	<b>Acronym</b>	<b>Definition</b>
Bill of material	BOM	A list of every asset used in building our game
Virtual Reality	VR	An artificial environment generated by a computer

## Introduction

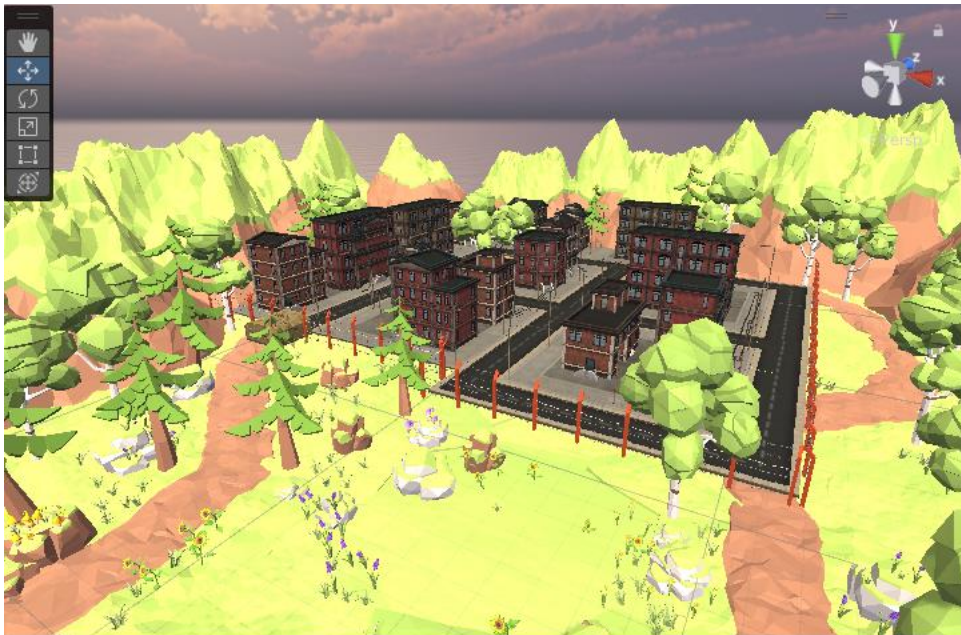
This manual is for anyone who wants to recreate our product, which is a VR game that shows how bad A.I.-controlled weapons can be. The manual will be going over all five steps of the design process, and any troubles we had along the way. Our product is not finished but was liked by clients we spoke to.

## Overview

Giving weapons autonomy is a very big problem in terms of ethics and morals, and all decision-makers should be made aware of this. It is very important that politicians, high-ranking military personnel, and the members of the United Nations (UN) Security Council be made aware of these issues before large-scale testing starts and it is too late to keep these weapons out of major conflicts, hence why our class is making these VR games.

Our target audience is the decision-makers we stated previously, and we are assuming that this audience does not have a lot of experience using VR games. The issue our group wanted to reinforce was the lack of control humans would have over who got killed by autonomous weapons.

All prototypes were built in Unity, using version 2022.3.10f1, and most assets were from the Unity Asset store, with the rest coming from Mixamo, a company specializing in 3d character modeling and animation



**Figure 1. Final Prototype**



## Cautions & Warnings

People new to using a VR headset can experience motion sickness and are advised to take the headset off if they feel any form of discomfort, and to immediately find a stable position, preferably sitting down, and to drink water. People with a history of seizures, extreme dizziness, or any condition that causes blackouts or loss of consciousness should not put on a headset unless absolutely certain the game is devoid of any triggers to the previously mentioned conditions. Take frequent breaks, stay hydrated, and remember to blink so don't end up with dry eyes. Also, when starting the game, be aware that the tank cannon will fire, as the fire1 command is linked to the left mouse button.

## Getting started



**Figure 2. Oculus Quest 2**

Figure 2 shows an Oculus Quest 2, the VR headset that was used throughout the development of this project.

**Step 1:**

Install the Oculus app on your computer

**Step 2:**

Either plug the Quest 2 into your computer with the provided cable or set up an air link through the Oculus app, and test your connection to ensure the connection is good.

**Step 3:**

Fasten the Quest 2 headset over your head and set up either a sitting or standing boundary by following the directions on the headset.

### Configuration Considerations

The only things applicable here are that the training dummies have exaggerated death scenes to show they have been knocked down.

### Accessing/setting up the System

First, you'll need to download the ZIP file labelled The-Project, which has our prototype 3 in it. Extract it, then put the folder somewhere you can remember easily. Go into Unity Hub, click Add, find the folder you just extracted, and click open. Once the file has finished processing, open it in Unity. Go through the previous steps for setting up the Oculus headset, then press the play button in the top middle of the screen.

## System Organization & Navigation

Currently, there is only one scene, with not much to change.

### **Physical Components**

A VR headset made by Oculus is required to run this game, preferably a Quest 2, along with a controller for each hand. The joystick on the controller in your left hand controls the movement and the A button on the right-hand controller fires the main cannon of the tank.

### **Movement**

If you find that the movement of the tank is too slow, find the XR interaction setup in the Hierarchy, go into the XR Origin, then the locomotion system, and click on the object labelled “Move”. A part of the Dynamic Move Provider script has a value labelled move speed, and increase or decrease that value to your liking

## Exiting the System

Press the play button in Unity again, this will stop the game, then put the headset away in the same fashion you found it. If Unity is crashing, simply press Alt-F4 or open the Task Manager by pressing the CTRL, ALT and DELETE buttons all at the same time and pressing the “Task Manager” option.

## Using the System

The game currently only has one scene, where the player is acting as the A.I. in an autonomous weapon. The weapon is moved via the left stick and the main canon is shot with the A button and the player must go around shooting test dummies.

## Troubleshooting & Support

### Error Messages or Behaviors

A behavior that was mentioned previously is that the cannon fires upon starting the game, since the Fire1 command is tied to the left mouse button and was not changed. If the user experiences the error in Unity “Unable to start Oculus XR plugin”, check the connection to your headset first, then click the edit button in the top left of the Unity window, select “Project Settings” near the bottom, go to “XR Plug-in Management”, the “OpenXR” and ensure the option for “Play Mode OpenXR Runtime” is set to Oculus OpenXR. You may need to do this every time you load up the game.

### Special Considerations

There are a lot of assets in this project, and a lot of the scripts are taken from those assets, so please be mindful of deleting things.

### Support

For support on this prototype, please ask a faculty member to contact Carter Yue through email, label the email Game Prototype, and he should respond within 1 or 2 days.

## Product Documentation

As mentioned previously, all prototypes were made using Unity3d version 2022.3.10f1. Assets were imported from either the Unity Asset Store or the Mixamo, a website with 3d character models and animations.

The game was supposed to have started with a voiceover explaining that the player is the A.I. controlling the weapon for this game, and which test dummies you're supposed to hit and which ones not to. The player would go about knocking down dummies and getting points for correct hits and losing them for incorrect ones. After a little bit, the player would then hear a small voice line from off to the side of some soldiers daring a new recruit to go onto the test track, where the player would lose control of the tank and it would fire on the soldier by itself. The screen would go black with the words "This is how in control you are with A.I. Join #TeamHuman". The hashtag is one that the client uses, so we thought it was appropriate and the first can be changed.

## BOM (Bill of Materials)

**Table 3. Bill of Materials**

<b>Item Name + description</b>	<b>Quantity</b>	<b>Cost (CAD + Tax)</b>
Flooded grounds(prototype 1 environment)	1	\$0

Tank 3D Model (prototype 1 model)	1	\$0
Free HDR sky	1	\$0
Sci-fi tank (final model used)	1	\$11.68
Modular City kit (final environment)	1	\$7.19
Camo Shader Pack	1	\$0
Low Poly Barriers	1	\$0
Free Low Poly Nature Forrest	1	\$0

## Equipment list

All equipment was used in all 3 prototypes

**Table 4. List of Equipment**

Item name	Description	Source
Oculus Quest 2	VR Headset used to test program	MakerRepo
XR Interaction Toolkit	Toolkit used to allow gameplay in VR	Unity built-in assets

## Instructions

First, you'll want to install the latest version of Unity, and immediately add the XR interaction toolkit. Start looking for assets as soon as possible, and you should try to get input from people outside of your group on how these assets look and if they line up with your future ideas. Our BOM has all the assets we used, and we separated our project into 3 subsystems:

### Subsystem 1. The storyline

The storyline is the most important part of your project since that is what is going to get your message across, and the other two subsystems are only there to support it. The story of your game should be short, to the point and obvious. Avoid using any real-world events, and excessive violence should be avoided.

## Subsystem 2. The A.I. driven Weapon

This can be the main focus of your storyline, or you could focus on the aftermath of these weapons going through the world. We chose to have the weapon be our main focus in the story to reinforce the ethical concerns of these kinds of weapons.

## Subsystem 3. The Environment

Like your storyline, avoid showing real world locations, as this can cause some people discomfort but still try to make your landscape recognizable. Avoid complex settings and large scale city's as a location for this game, as it will take too long to construct.

## Testing & Validation

**Table 5. Prototype 1 Test Plan**

Test #	Test Objective <b>Why</b>	Description of Prototype used and of Basic Test Method <b>What</b>	Description of Results to be Recorded and how these results will be used <b>How</b>	Estimated Test duration and planned start date <b>When</b>
1	See what game physics we want	Looking at free environment that came with some wind and water physics built in	See what's hard to code and/or makes my computer lag	1 hour
2	Assess Tank Model	See what we don't want out of our tank and then find what we do want	Feedback from group members, other students, and friends	1.5 hours



**Table 6. Prototype 2 Test Plan**

Test #	Test Objective <b>Why</b>	Description of Prototype used and of Basic Test Method <b>What</b>	Description of Results to be Recorded and how these results will be used <b>How</b>	Estimated Test duration and planned start date <b>When</b>
1	Make sure paid models look good	Verification with other people and our group		Starts with prototype 2, until client sees it
2	Ensure Code works properly	Make sure all moving parts go where they are supposed too	Feedback from group members and maybe other users if necessary And will be used to improve code	Starts with prototype 2, until client sees it
3	Message is clear	Seeing if there are ways our video could be misinterpreted	Asking people to watch our rough video and using feedback to change story.	Starts with test 2 being done

**Table 7. Prototype 3 Test Plan**

Test #	Test Objective <b>Why</b>	Description of Prototype used and of Basic Test Method <b>What</b>	Description of Results to be Recorded and how these results will be used <b>How</b>	Estimated Test duration and planned start date <b>When</b>
<b>1</b>	Background looks good	Prototype 3	Yes	0.5 hours
<b>2</b>	Animations of dummies and soldiers work	Prototype 3	Yes	2 hours
<b>3</b>	Audio clips work	Prototype 3	No	1 Hour

## Conclusions and Recommendations for Future Work

Developing this VR game has given us an appreciation for the design thinking process, and this experience has given us a lot of experience with each step. The most important lessons we learned was to start coming up with ideas as early as possible and to set up a way to share our work equally fast. Starting early can only give you more time in the long run, which you'll appreciate if you have other hard courses to take at the same time. Setting up a way to share work proved to be quite challenging, as none of us had ever used a website like GitHub before, and we did not have time to set it all up correctly.

If another group wanted to continue our work, I would suggest fleshing out voice lines for explanation purposes, so you don't have to do a lot of talking while the game is running on Design Day. Implementing a Start menu to really make it feel like a game would be another good idea.

If our group had more time, I think we would just go about implementing all of the things we said we didn't have time to do, and make it a really smooth experience.

If more time was put into the game, we think it could have made a significant impact on decision makers in the UN, but the game doesn't look too bad as it is.

## Bibliography

Knight, T. (2020, December 14). Oculus quest 2 / meta quest 2 - hardware review: A not too technical VR review. NookGaming. <https://www.nookgaming.com/oculus-quest-2-hardware-review/><sup>1</sup>

# APPENDICES

## APPENDIX I: Design Files

All files will be in MakerRepo

**Table 8. Referenced Documents**

Document Name	Document Location and/or URL	Issuance Date
Maker Repo	<a href="https://makerepo.com/Carrunner/1807.group-13-lucky-13">https://makerepo.com/Carrunner/1807.group-13-lucky-13</a>	