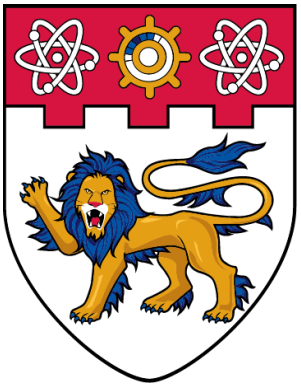


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

CZ4001 - Virtual and Augmented Reality

Virtual Reality Assignment Report

Project Name: Save Gotham!

Group Name: Team One

Date of Submission: 20/02/2022

Group Member	Matriculation Number
Koh Luo Hao	U1822537H
Lee Yi Hui Rachel	U1922672E
Ernest Ang Cheng Han	U1921310H
Mohamed Fazli Bin Abd Razak	U1821890C
He Wanru	U1820328A

Table of Contents

1. INTRODUCTION

Climate change has become one of the top concerns of people all around the world. Our earth's temperature has risen by 0.14° F (0.08° C) per decade since 1880, and the rate of warming over the past 40 years is more than twice that: 0.32° F (0.18° C) per decade since 1981. 2020 was the second-warmest year on record based on National Oceanic and Atmospheric Administration (NOAA)'s temperature data, and land areas were record warm. According to 2020's national recycling data from National Environment Agency (NEA), the domestic recycling rate stands at 13 per cent - a 10-year low (NEA, 2021).

Despite the Singapore government's efforts to encourage recycling, only 3 in 5 Singaporean households recycle (NEA, 2021). Additionally, 40% of all items placed in the recycling bins cannot be repurposed due to food and waste leakage or are simply non-recyclable. The Singapore government has been trying to raise awareness of the importance of recycling and aims to achieve a 70% overall recycling rate by 2030. However, Singapore's vision of becoming a "zero waste nation" will take the efforts of the collective efforts of the whole country. Therefore, our team created a virtual reality game that is aligned with Singapore's vision to increase recycling rates.

"Save Gotham!" is a virtual reality game played on an Oculus Rift which focuses on Climate Change and Sustainability. The objective of the game is to create awareness on the severity of climate change and urge people to do their part for the environment through proper recycling habits. We aim to cultivate good recycling habits and educate users on different recycling procedures through play. We hope that the gaming experience would provide users with an immersive experience as it allows them to manipulate and interact with their environment.

The main gameplay involves players navigating through the city and picking up trash that can be recycled and throwing them into the correct recycling bins - paper, plastic and metal. Examples of such recyclables are metal food tins and books. There will be a point system where players get more points for throwing the recyclables into the correct bin and lose points for classifying wrongly. Players are also able to beautify the city by planting flowers and creating greenery. We also implemented a leaderboard system where players can compete with their friends to accumulate as many points as possible to make the game more exciting.

To simulate and create awareness of the importance of recycling and how doing so can impact and save the environment, players will “die” from the lack of action or from wrong actions made during the game (such as depositing the wrong trash in the wrong recycling bin). Players will need to earn a minimum of 200 points within a stipulated time of 5 minutes in order to “survive” and have a chance to enter the leaderboard. This adds realism and gamification to “Save Gotham!”.

2. TECHNOLOGY REQUIREMENTS

2.1 Game Design - Unity

“Save Gotham!” was developed using Unity 3D Game Engine, a popular game development software which can also be used to create applications based on extended reality and other related-immersive technologies. The default display language will be English and in the future, we will improve our game and add more languages as we want our game to cater and serve a larger spectrum of users to maximise the educative potential of our game. Majority of the scripting and in-game logic was written in C# - the native programming language of Unity.

2.2 Backend System - Firebase

“Save Gotham!” also has a leaderboard system and high score function in order to create a friendly yet competitive spirit amongst players, motivating them to continue playing. This system was done with Google’s Firebase Realtime Database, a popular Backend-as-a-Service which did the majority of the backend logic, saving us many hours of development time. We could call the Firebase API to perform the complicated procedure involved when saving and updating players’ high scores, as well as rendering the top N players on the leaderboard.

2.3 Hardware - Oculus Rift

We implemented our game and deployed it on an Oculus Rift, which required the Oculus Rift application as well as the Unity’s XR Interaction Toolkit plugin to set up the movement and player controls to our game. This is the basis of the basic functionalities of our game, such as picking up and releasing objects which, for example, used to allow players to pick up the virtual rubbish and deposit it into the appropriate and corresponding virtual recycling bin.

3. KEY DEVELOPMENT FEATURES

3.1 Development without VR Headset

Given that our game is to be deployed and used with a Virtual Reality (VR) Headset, Oculus Rift in this context, coupled with the fact that all of us in development team do not own or have an Oculus Rift or any VR related hardware, we created a VR device simulator to simulate a connected VR Headset during development using Unity's Mock HMD XR Plugin. This made it more convenient for us, reducing development time and the number of bugs encountered when testing and deploying our program on the actual VR Headset when we had the opportunity to.

3.2 Player Interaction

Player interaction in our game is implemented using the XR Ray Interactor, which allows interaction with objects at long range. The XR Ray Interactor is customised to interact only with objects set as the same layer defined by its Raycast Mask field and displays a straight and dotted white line when aimed at a valid interactable object.

3.3 Player Movement

Our game makes use of both controller-based and teleportation-based locomotion techniques for player movement. Movement of the player is implemented using a Continuous Move Provider where players are able to move forward and backward as well as strafe left and right by interacting with the joystick on the left-hand controller. A Character Controller component is also added to the XR Rig to handle collisions with objects while moving. Turning, on the other hand, is implemented using a Continuous Turn Provider where players are able to continuously turn left and right by interacting with the joystick on the right-hand controller.

As for teleportation, Teleportation Provider is used to allow players to teleport to a target location. The XR Interaction Toolkit includes 2 components for defining where the player can teleport to. The Teleportation Area will allow the player to choose any point within an object to teleport to, while the Teleportation Anchor will always teleport the player to the same position and orientation. In our game, we made use of Teleportation Area to define the location the player can teleport to. In our game, the Teleportation Area component is attached to every walkway and road in the city.

3.4 Hover Object System

When a player hovers over an interactable object with the XR Ray Interactor, a tooltip UI displaying the object's name and related information is displayed in the left or right side of the player's visual field depending on which VR controller is used - Left-Hand or Right-Hand. This is implemented by creating an XR Grab Extension script which extends XR Grab Interactable and extending the method OnHoverEntered() to capture which XR Ray Interactor is used - Left-Hand or Right-Hand, when an interactable object is being hovered.

3.5 Interactable Objects

Interactable objects are objects that can be interacted using the XR Ray Interactor such as recyclables, flowers and the UI canvas. The recyclables in particular are implemented with different Physically-Based Rendering (PBR) materials to differentiate between the different types of recyclables. For example, in our game, plastic objects generally use a specular workflow with smoothness set to 0, paper objects generally use a metallic workflow with metallic and smoothness property set to 0 and metal objects generally use a metallic workflow with metallic and smoothness property set to 0.5.

3.6 Points System

All players start off with 0 points. With every rubbish they successfully pick up and deposit into the correct recycling bin (e.g. aluminium can thrown into the yellow recycling bin), players will gain 10 points. On the other hand, players will lose 10 points for every rubbish they pick up but unfortunately throw into the wrong recycling bin (e.g. aluminium can thrown into the red recycling bin). This is done by creating categories of game tags (e.g. Plastic, Metal, Paper, General Waste) and tagging the rubbish prefabs with the appropriate tags. We also created the corresponding colliders (e.g. Plastic Collider, Metal Collider, Paper Collider, General Waste Collider) and attached them to the appropriate rubbish bins. These colliders wait for a collision with a tagged prefab, which occurs when a player releases a rubbish prefab into the recycling bin. On collision, it checks the prefab for its tag and the rubbish's tag before appropriately updating the in-game points manager. Additionally, players will gain 20 points for grabbing flowers scattered around the city and releasing them into an empty pot. This is done by attaching an XR Socket Interactor onto the flower pot game object that updates the in-game points manager once a flower is brought to the pot. Players are given a limit of five minutes, and the game ends when the timer is up. The player wins if he manages to accumulate 200 points before the time runs out.

3.7 Pollution System

All players start with a pollution index of 0. The pollution index is controlled by the pollution manager and signifies the amount of “pollution” that exists in the environment. The pollution manager is connected to the points manager. The higher the number of points earned by the player, the lower the pollution index, while the lower the amount of points earned by the player, the higher the pollution index. Based on the pollution index, the pollution manager also controls the “state” of the environment in terms of the amount of smoke and rain that the player sees. A higher pollution index results in a worse and more polluted environment shown by more smoke and heavier rain, whereas a lower pollution index results in a better and less polluted environment shown by less smoke, lesser rain and clearer skies. The implementation of rain, smoke and lighting to reflect the environment change is explained briefly in the next section.

3.8 Rain, Smoke & Lighting Implementation

Rain in our game is implemented using two particle systems, rainfall particle system and rain mist particle system. The rainfall particle system is designed to follow the VR camera and has its render mode set to stretched billboard to obtain long cylindrical particles similar to rainfall. Additionally, the collision parameter of the particle system is set to collide with every object in the game to bring about a more realistic effect. The rain mist particle system, on the other hand, is designed very similarly to the rainfall particle system, with the exception that the render mode is set to billboard to ensure the particles are much smaller and circular to simulate the mist effect. The pollution manager controls the intensity of the rain by simply decreasing or increasing the emission rate of particles over time, depending on the pollution index.

Smoke in our game is implemented using a particle system that emits particles from a hemisphere to simulate an outward emission. The smoke effect is simulated by setting the colour over lifetime parameter such that the particles start off grey and fade in opacity over time. The lighting that is used to reflect environment change is implemented by setting a grey UI canvas in front of the VR camera, where the lighting intensity is controlled by modifying the alpha values of the UI canvas. The higher the pollution index, the higher the alpha values of the UI canvas.

3.9 Environment Prefabs

The majority of the assets and prefabs we used, such as the City Environment, Rainmaker feature, and various “trash” that the player has to pick up, were imported from Unity Asset

Store. At times, there was a need to change the shaders and materials used by the assets due to their inability to be rendered via the High Definition Render Pipeline (HDRP) or for its materials to be upgraded to High Definition Materials. HDRP is the main rendering pipeline used in most VR projects.

4. USER INTERFACES

4.1 Main Menu UI

There are 4 options that players can choose from in the main menu - Play, How to Play, About and Quit. Players can use the controllers to select “Play” to launch the game. The “How to Play” option is for new players to get to know the game instructions easily. We added pictures to make it very easy for players to understand the game play. “About” gives players an overview and objectives of the game. Lastly, the “Quit” button can be selected to exit the game. *[Refer to Appendix: Figure 1-3]*

4.2 Leaderboard UI

The top 3 players will be displayed on the leaderboard with their respective number of points earned. Persistent data storage of players’ highest scoring attempts and the querying of these data is done via Firebase. *[Refer to Appendix: Figure 4]*

4.3 Tutorial UI

The city environment is the main game play area that players will be at when they are using our application. Using the specified navigation controls, players will be able to move around the environment, jump, and crouch to pick up or release rubbish should they wish to. Before players embark on their adventure, they are presented with a tutorial in order to orientate them on what needs to be done. *[Refer to Appendix: Figure 5-6]*

4.4 Tooltip UI

While navigating the city, tooltips will pop up whenever users point to a rubbish object or a plant, educating users about that particular item and how it affects climate change. The data is based on official statistics from NEA. *[Refer to Appendix: Figure 7]*

4.5 Game-end UI

At the end of the 5 minutes from the start of the gameplay, should the player earn at least 200 points, he/she will proceed to a successful game completion scene which commends the player for his/her efforts on recycling and saving the environment. If the player has

earned more points than his/her highest scoring attempt, the system will update his/her highest scoring attempt in the Firebase database. On the other hand, if the player fails to earn the minimum number of points, the player will proceed to an unsuccessful game completion scene which prompts the user to try again and try harder to save Gotham. *[Refer to Appendix: Figure 8-9]*

5. AREAS OF IMPROVEMENT

5.1 Game Feature #1 - Vehicles and Transport

One feature that can be expanded and integrated into our game, is the use of different forms of transportation to educate players on the impact of carbon emissions generated by the various means of transport. By giving players the option to navigate the play area through toggling the various means of transport (e.g. foot, bus, car, train), we would allow the player to move at varying speeds but creating pollution at different rates as well. For instance, using the car would allow the player to move around at the highest speed but create tons of pollution, while travelling by foot (the default means of transport) will allow the player to move around at the slowest speed but without creating pollution. This feature will hence educate users about the impact that their daily travelling decisions have on the environment and adds some decision-making during the gameplay. Players will need to take this into consideration while collecting and depositing the most amount of rubbish correctly to keep pollution levels low.

5.2 Game Feature #2 - Quiz Feature

To make the game more educational, we can also create interactable quiz stations around the play area. Players will be quizzed on whatever they have learnt from interacting with the various trash and information boards in our game. For every question they answer correctly, it would earn them points and reduce the pollution index of the environment. This increases the information retention and allows them to apply what they have learnt through our game. Therefore, increasing the probability that they learn the importance of protecting our environment through simple everyday actions such as cultivating good recycling habits.

5.3 Game Feature #3 - In-Game Chat Rooms

Another feature that could be integrated into our game is the use of real-time chat rooms with other players in the game. This will add on to the collaborative and immersive experience of the game, where players can chat with one another, save the city together, as

well as discuss their insights on what they have learnt from playing our game. This creates a wholesome and fun experience for all users while learning.

5.4 Game Feature #4 - Non-Player Characters (NPCs)

Another enhancement that we could add to our game is the use of interactable Non-Player Characters. NPCs are a great way of adding realism by allowing players to interact with them. Information and details can be disseminated by these NPCs, and we could even use them as specific checkpoints that players have to finish in order to complete the game.

6. CONCLUSION & REFLECTION

According to a study by Meta, the owner of the very VR Headset we used during this project - the Oculus Rift, it is estimated that the virtual reality software market has an estimated value of over 2.2 billion USD by 2022 (Metinko, 2022). Developing and creating “Save Gotham!” was an awesome and fun experience that pushed us out of our comfort zone. From working on a new type of hardware - the Oculus Rift, to ideation and creating a game in 3 - 4 weeks, along with overcoming many obstacles, we believe that “Save Gotham!” is a successful mini-game which educates users on the importance of environmental sustainability through recycling and has great potential in improving further. Many skills were learnt as most of the developers in Team One do not have any development or design experience with C# and Unity, let alone complex immersive technologies such as XR Interaction Toolkit and XR Mock HMD Plugin. The use of extended reality and virtual reality, as well as other technical innovations, has allowed us to appreciate and create an immersive and intuitive gaming experience, while applying what we learnt during lectures.

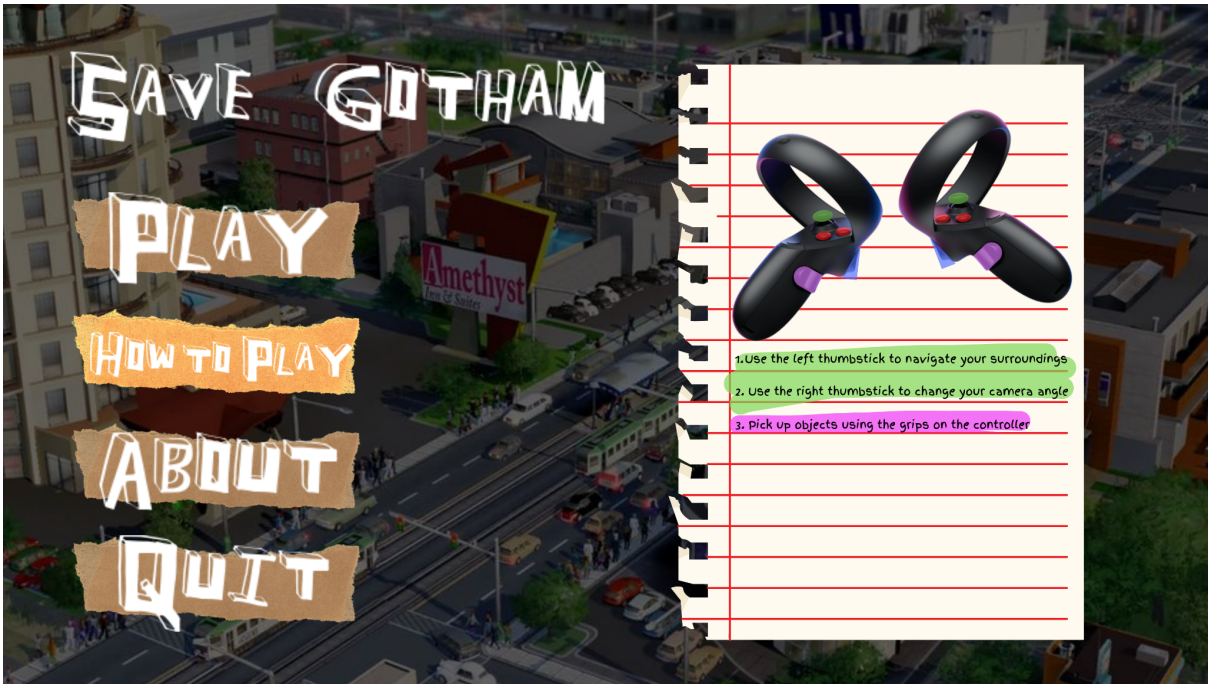
7. REFERENCES

- Andres, G. (2021, April 23). Less waste generated in Singapore in 2020, recycling rates drop due to impact of covid-19. CNA. Retrieved February 22, 2022, from https://www.channelnewsasia.com/singapore/less-waste-generated-singapore-recycling-rates-drop-239601
- Lindsey, R. (2021, March 15). Climate change: Global temperature. Climate Change: Global Temperature | NOAA Climate.gov. Retrieved February 22, 2022, from https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature
- Metinko, C. (2022, January 21). VR/AR Investments Increase Just As Metaverse Talk Heats Up—But That May Not Be The Only Reason. Retrieved 2022, February 15, from https://news.crunchbase.com/news/metaverse-augmented-reality-virtual-reality-investment/

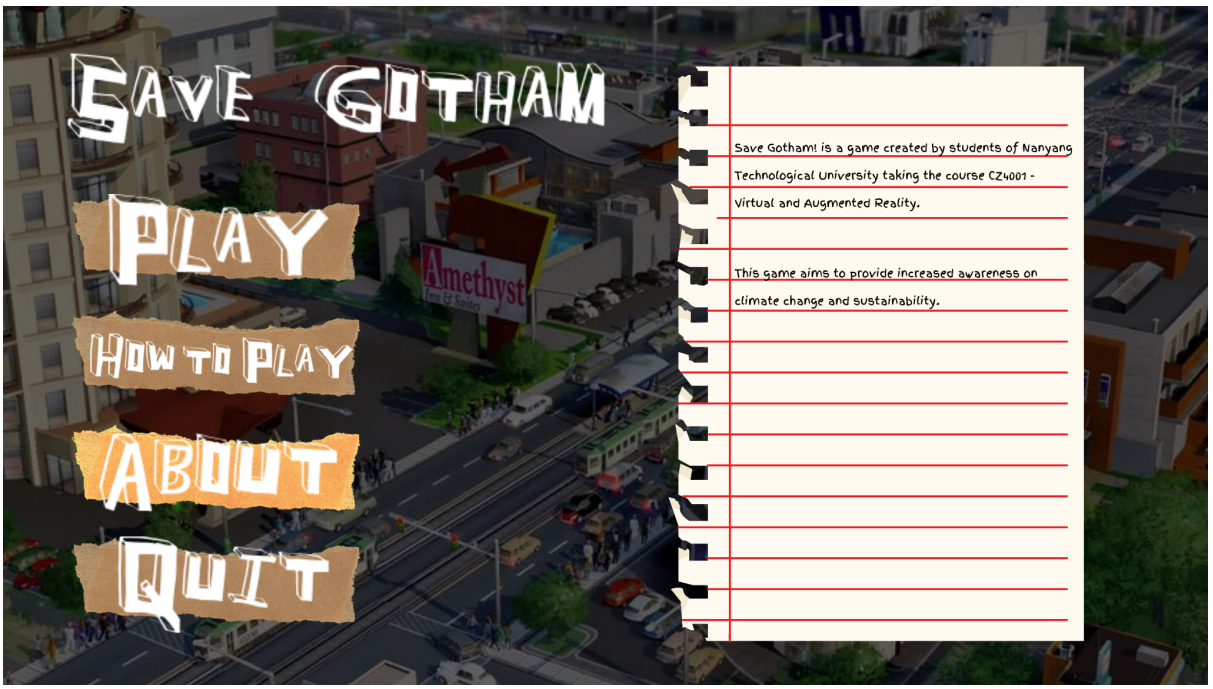
8. APPENDIX

1) MainMenuUI

[Figure 1: Main Menu]



[Figure 2: How to Play]



[Figure 3: About]

2) Leaderboard UI

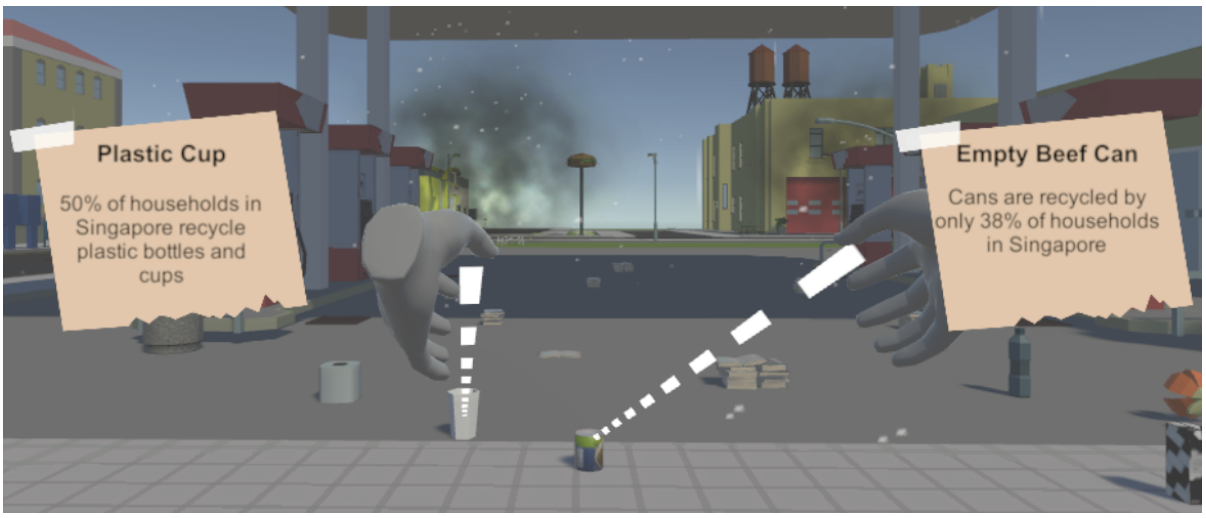
[Figure 4: Leaderboard]

3) Tutorial UI

[Figure 5: Planting]

[Figure 6: Gameplay]

4) Tooltip UI



[Figure 7: Tooltips]

5) Game End UI

[Figure 8: Game Over]

[Figure 9: Victory]