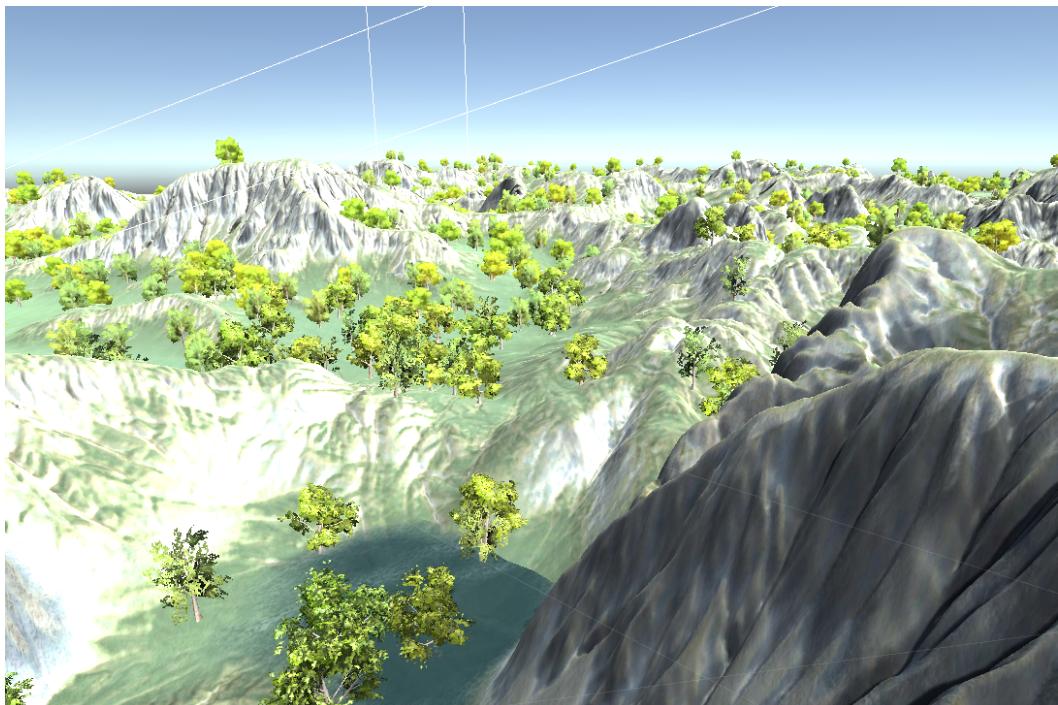


A. Overview

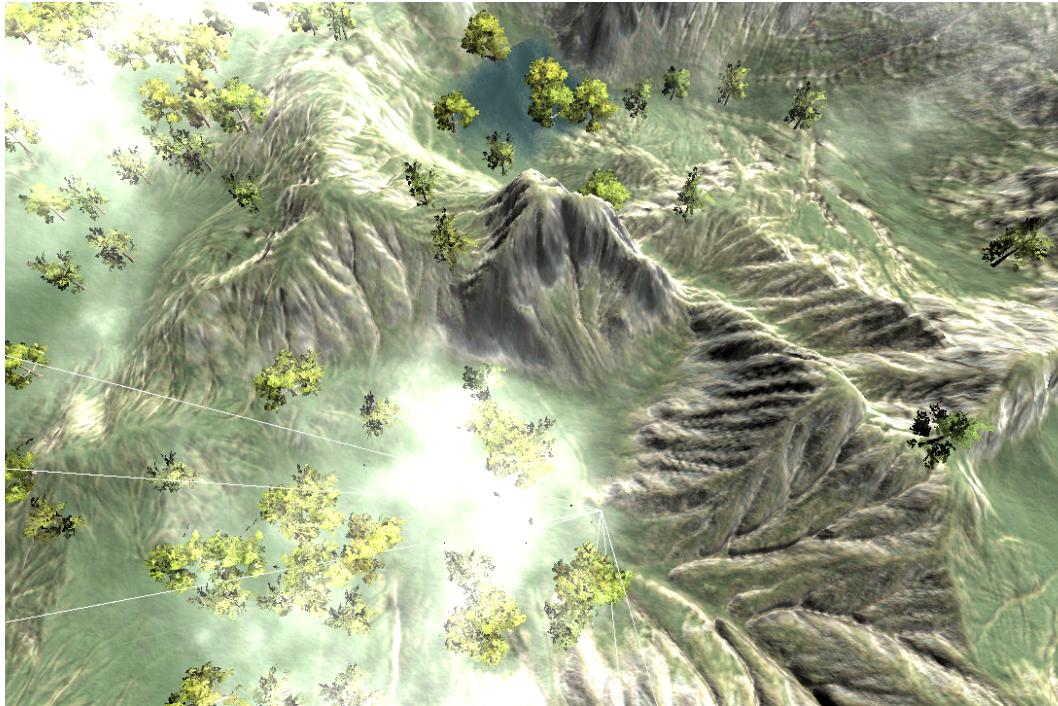
The game created is an 3D open world style role playing game made with Unity. It features a robot character that is stranded on an Earth-like nature planet and needs to collect resources to rebuild their ship while avoiding enemies. If the player collides with the enemy, then the player loses points. The enemies are able to chase the player around the field up to a certain distance. The players can use obstacles around in order to evade and hide from the enemy AI. The player can also go around collecting bolts and other various items in order to rebuild their ship. This can be seen on the top screen inventory in the main part of the game. A demo video of the player dodging enemies and exploring the terrain is submitted through Canvas.

B. Gallery



Overview of Map

Madiha Abdul Maajid
Final Project Report
Computer Graphics
4 May 2022



Top Down Detail View of Mountain Range and Clouds



Enemies

C. Code

I. Player Controls

<https://github.com/madihabdul/428FinalProject/blob/main/thirdpController.cs>

II. Player Camera/Collision/Follow

<https://github.com/madihabdul/428FinalProject/blob/main/rigidBody.cs>

III. Inventory

<https://github.com/madihabdul/428FinalProject/blob/main/PlayerInventory.cs>

IV. Collectible Items

<https://github.com/madihabdul/428FinalProject/blob/main/AssetCollection.cs>

V. Enemy Patroller

<https://github.com/madihabdul/428FinalProject/blob/main/enemyAI.cs>

VI. Music

<https://github.com/madihabdul/428FinalProject/blob/main/musicscript.cs>

D. Challenges

Creating the character camera and player controls were quite difficult as in Unity. Although Unity has built-in methods to create a camera and controls, it only works for test characters and you have to make your own system for a consistent experience.

The terrain was also difficult to balance as I wanted a realistic terrain while also not overloading it to keep performance optimal. Recognizing objects and enemies was also a challenge, and the code associated had to be tested multiple times in various scenarios to ensure that it was working as intended. Also, deciding the type of rendering pipeline, whether the 3D Core, URP, or HDRP was a difficult choice as each type of rendering had its own benefits and disappointments.

Ultimately, I chose the 3D core running on 2021.2.016b as it was more sophisticated than the last stable release and the URP rendering pipeline had some issues using older materials. A lot of time was dedicated to simply trying out which rendering pipeline would be most compatible with the intended goals.

Another challenge was implementing an enemy AI. Initially, I wanted an enemy AI that chased the player around. However, it was difficult to handle given the terrain, and there were issues with the NavMesh that would have taken too long to fix. I still wanted visible enemies, so instead I created enemies that could move with waypoints throughout the field for a more polished look. Additional challenges were the items themselves, though the game detects the collision for the items, it does not disappear.