NEMESIS

Introduction

Nemesis is the most significant project I've worked on as part of my Game Design and Development Course under the guidance of Prof. Timothy Scott Moyers Jr. It is a third person fantasy game implemented in Unreal Engine based on task completion and includes battles with AI characters with a unique level design and animations. This project was implemented in a team of two people over the months of January-April, 2019.

Features

Terrain

1. **Landscape Tool -** We used landscape tool to create the mountainous terrain.

2. **Snow Material** - We created our own custom material for the snow mountain with a rocky texture without shine using **Landscape Coords** and **Linear Interpolation**.

3. **Snowfall -** We used three images to form textures. We then created 3 separate materials using **Linear Interpolation.** We used these materials to create a particle system and attached it to the Third Person Character Blueprint.

4. **Fog -** We created fog to support the winter theme.

5. **Frozen Lake and Waterfall -** We created a custom material for the frozen lake and waterfall using **Panner.**

6. **Ice Material -** We created a snow material to overlap our assets.

7. **River -** We imported Water Plane from the Unreal Marketplace and customized it. We changed the opacity of the water material to **Translucent** and made use of **Custom Stencils** so that water doesn't enter the boat. We edited several other fields like **Large and small wave tiling, speed and distance**.

8. **Cave -** We created our own cave using after importing rocks from the Marketplace.

9. **Waterfall -** We used Water Materials from Unreal Marketplace and customised it to change the texture and color of the water.

10. **Platform and Battlefield -** We created our platform by assembling and customising assets.

Assets, Materials and Particle Effects-

We imported Infinity Blade Icelands, Soul Cave, WaterPlane and Water Materials and FX Variety Pack from the Unreal Marketplace and used these for majority of our assets. We also imported assets from online sources. We also used several of the particle effects of our Paragon characters.

1. We made many of these assets like pots and barrels **destructible** using Apex Destructible Mesh Plugin.

2. We created our own particle effects for the blue aura (Level 1) and the fireplace (Level 3) by customising the fire particle effect.

3. We created our own fireball (Level 3) by creating 2 particle effects and using **projectile motion** on one of them.

4. We gave power to our character by modifying his particle effects, adding projectile motion and event hits to the the same.

5. We created a translucent custom material for the bridge (Level 3).

6. We create a powerball and Unity stone by finding assets and assigning our customised materials to them.

7. We use ice crystals creating asset, particle effects and a light source to recharge mana for the character.

HUD -

We used the following in our widgets -
1. A customised font
2. Health bars for characters created via Adobe Illustrator
3. Image imported for meat edited via Illustrator

Powers of Characters -

Our Character -
1. **Throws fireballs** from chest in the presence of red aura.
2. **Throws power** from his ring if he has a power ball and enough mana.
3. Has a sword and can attack enemies with **3 different types of sword animations**.
4. Has a **massive sword attack** for more health damage.
5. When he kills a villain, he can increase his health using a **health pickup**.

Villains -
1. Can attack the character with **2 types of sword animations** if he is within a 300 meter radius.
2. Have **3 hit react animations**.

3. Can **sense** if the character is within his sphere and moves towards him.
4. **Rotates along with character**.

Character and Animations -

We imported the Paragon Gideon, Paragon Greystone and Mountain Dragon characters from the Unreal Marketplace. We modified these as follows -

1. We modified the **skeleton** of our characters to add **sockets**. The sockets are used for -
    a. Throwing fireballs from Gideon's chest.
    b. Throwing power from his ring.
    c. Placing the Unity Stone.
    d. Giving characters a sword and allowing them to attack with it.
    e. Attaching our character to the Dragon and allowing him to fly with it.

2. We created **Animation Blueprints** for major animations like recharging mana, gathering aura from treasure chest, throwing fireballs, throwing power and finishing a level.

3. We also **modified existing animations** in several places, like placing the unity stone and riding the boat. We **recorded animations**, **added** and **removed key frames**.

4. We made changes in the existing animation blueprints of our characters to make them hold a weapon and attack.

5. We used **Animation Retargeting** to target animations from different skeleton. We mapped the skeleton of one character to another and retargeted animations. So, we used many of Greystone's animations on Gideon.

6. We add **anim notifications** to add **sword trails** when our characters attack using a sword. We use **three types of attack animations** which are chosen at random when a our character attacks. We also have another animation for **massive attack** which causes more damage.

7. We also added **different hit react animations** so the characters can respond when they are attacked. The characters also react with a different animation when they are killed.

8. Character **possess** the boat and rides it with an animation.

9. The boat and barrels present in the water have **buoyancy** implemented according to the height of the water and the mass of the asset so that they can float on water.

Level Sequence -

We've implemented 4 level sequences in our game -

1. As the game begins, a dragon flies over our head with the suitable animation and drops a sword along with particle effects for thunder and aura. The camera also moves to focus on the dragon followed by the sword. We added audio tracks to this too.

2. When the player moves towards the sword, a level sequence is initiated where he holds the sword and keeps it with his armour.

3. When the platform rises, the boss villain comes towards us on a dragon. Several animations are sequentially implemented along with camera focusing on the dragon with the boss attached to it. Then he gets down from the dragon and has his own animations.

4. Our character takes off on the dragon after winning the boss fight.
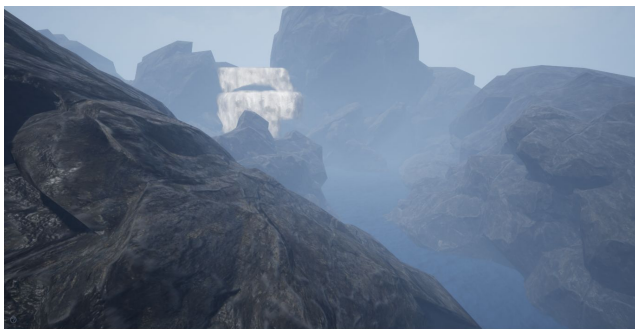
Audio -

We added background music and some basic audio and dialogues.

Coding -

We implemented Unreal blueprint coding to code the different functionalities within the game.
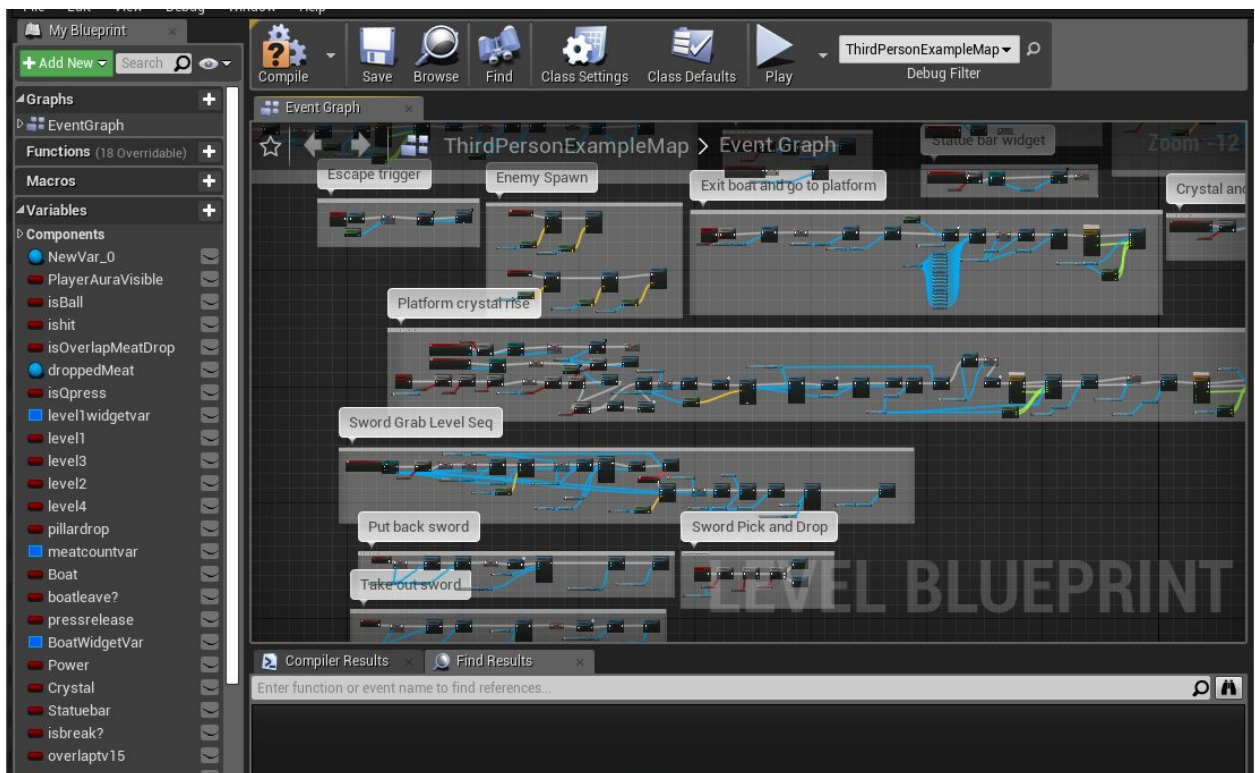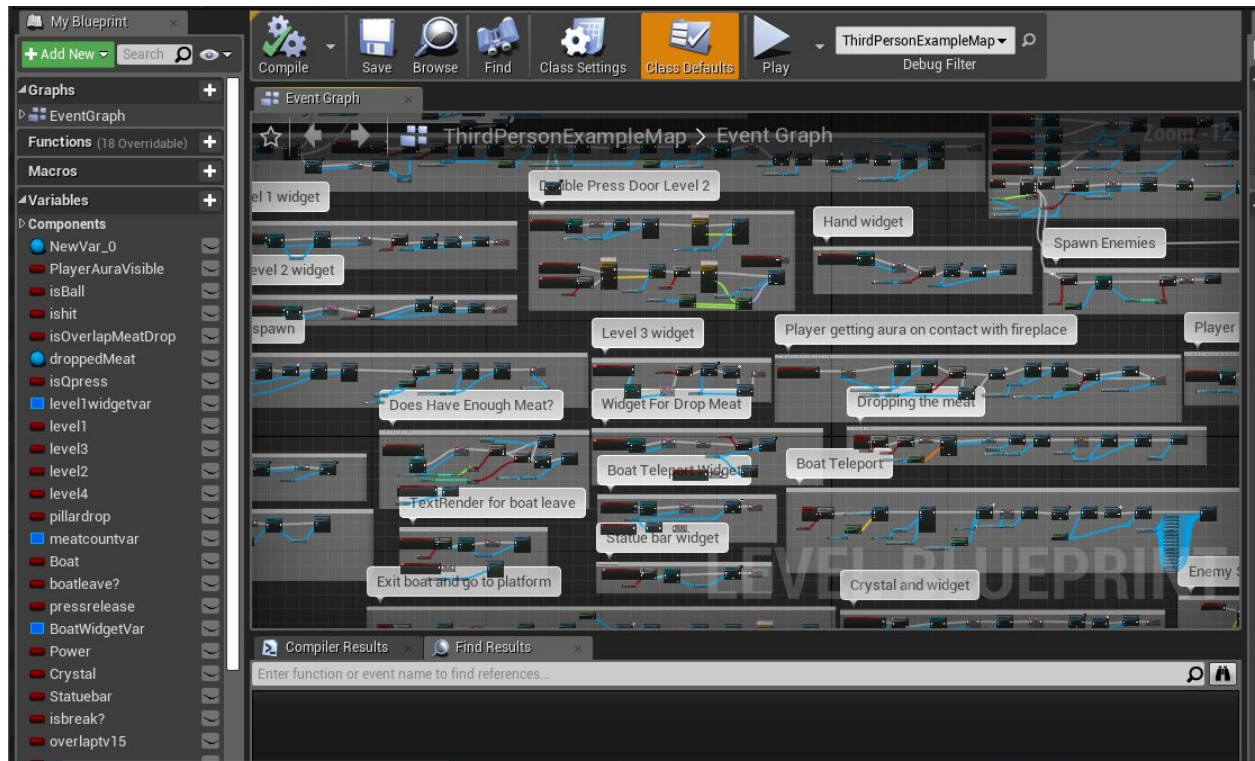
Technical Challenges

Since, this was my project in game development, I faced several challenges such as the lack of a system with a good enough graphic card leading to low fps. Since, there was no way I could arrange a different system, I decided to dedicate more time to the project and continued working on the one I had. A better system would have definitely helped me speed up the process and include more functionality in the game such as a forest scene.

Another challenge was unfamiliarity with Unreal Engine. I watched some tutorials and worked on the environment regularly to familiarise myself. I also practised blueprint coding to implement the tasks within the game.

Sometimes, I faced problems thinking of creative ideas for the storyline of the game as my mind would draw a blank. In these cases, I watched the gameplays of some famous games and drew inspiration from them but made sure all the functionalities are original.

Other challenges included not finding the right assets or particle effects or animations for the game. To resolve these, I designed and created these on my own and tried to do it within the required deadline.

The major challenges involved removing rare bugs from the game such as animation glitches and water within a boat. The bugs took 1-2 days at a stretch to be removed by going through several articles in the unreal forums and watching tutorials related to them. But, they overall helped my understanding and allowed me to learn new techniques.

Individual Contribution

Since the project involved several tiny details, all of which were significant, there was no clear distinction of roles. I worked on all aspects of the game, from designing the characters and their animations to programming all the functionalities. I also worked on setting the storyline of the game and designed the whole terrain and environment of the game including carefully editing each asset used.

Conclusion

This project was an enriching experience as it opened to me the world of Game Design and Development and the features involved in creating a game. It made me realise how much time and effort is put in by several people to build a good game.

I understood the importance of creative ideas and innovation in the success of a game. Any game is incomplete without an authentic and interesting storyline and fails to draw the attention of the player. Our game was addictive and indulges the player completely due to the realistic setting of the game and the realistic animations at each movement of the characters within the game. Thus, creativity, great ideas and perfection are important to the process of building a good game. Without perfection, the game would be full of bugs and degrade user experience.

This project also made me realise the importance of teamwork for the success of a project. Creating a flexible environment is important for the flow of ideas and constructive criticism helps bring out better results.

Thus, I realised that imagination, diligence, perseverance and amity among team members are important for the successful launch of the game.