Name: Diro Baloska
Matrikelnummer: 566367

In this project, a star reactor is being modelled. It contains 3 generated meshes, 2 cylinder object, 2 prefabs with their own generated position, 4 plains, 5 cameras, 6 Shader graphs, 1 visual effect and 4 texture renderers. 1 cylinder and 1 camera belongs to player property for first person point of view.

The 3 generated meshes are the star which is basically a sphere, toruses which are named as ring in the hierarchy, and the holder which is a modified formula of cone. The script to generate the sphere is called BallMesh, for the toruses is TorusMesh, and for holder is HolderMesh. The first mesh generated is actually a perlin noise terain of a plain, but it wasn't useful for my project so the object is changed into Minecraft liked terrain. Only the terrain is flat with different prefabs created, brownish texture and grey texture. Creating the ball comes second after the terrain, then torus and at last the holder. They have similar structure as a script, the difference is only the order of vertices, triangles and uvs.

For the shader, I created unlit master for all of the generated mesh. For the star, I want the texture to be moving according to the time. So it has UV as extra for the texture before connecting it to the master node. The same as torus and because I want the torus not to be fully torus, I created an extra node of noise and combined it to the alpha value of the master node. Then by using alpha clip, I can define which shade should be invisible and which one should be visible. With that, I created the Atom look alike object which rotates circling the star. Then for the holder, I used the texture of the terrain multiplied the value of the colour towards 255 by using vector1 node.

For the terrain, it is created by using TerrainMesh script. In the script, it takes 2 prefabs as parameters which are the brownish terrain and the normal grey terrain. They are generated randomly, either the brownish terrain or the normal grey terrain. The brownish terrain is basically identical to the normal grey terrain. The difference is that there is another colour node multiplied to the texture resulting a brownish texture.

The other cylinder covering the whole reactor is created to be glass like. It has transparent look with a little shade of orange and fresnel node added to the material resulting into a reflection like look. And because the reactor is orange (the star), the fresnel node has orange colour.

The last are the monitors. They are created by using plains and cameras for each position. The cameras are placed from the from where the player is being spawned, bottom, left and right. The position of the monitors corresponds to the position of the camera. All of the shader is using unlit graphs.

Then the last is the player. It is a group of game objects, cylinder and camera. The cylinder is responsible for the movement and rotating the camera to the left or right, where as the camera is responsible for looking up and down. This control is based on Brackeys' tutorial on how to create an fps game. But only the essential part of controlling the body is being taken, which means, the player could fall down and so on since it is not kinematic and it falls down simply according to isGravity function. Jump function is added but the player should be really careful since the object can turn upside down and there is no way out of this position and the only option is to quit the game.

The last is the visual effect. It is using simple swarm particle system. It doesn't have any extra node and it is being done by using trial and failure until it fits to the effect the star needs. Basically, this is a model for a research about star reactor and monitoring the star is the main thing. A lot of

Brackeys' tutorial are being used as inspiration but the things being done are purely based on the things I understood from the tutorial without any intention of copying.

While generating the meshes, I tried to add component of mesh collider and the collider is not built correctly and instead it only works only when is convex is true, and it won't work even if the created mesh is assigned into sharedMesh of mesh collider. That is the only problem I have in this project.