Virtual Reality

2018/2019 - Fall Semestre
MEIC-A / MEIC-T

Project 1 - Simple VR scene

Group #	8
Student 1	Francisco Dourado, 69962
Student 2	Luís Fonseca, 79770
Student 3	José Gomes, 89378

Link to project repository (private share): *https://github.com/nekrotzar/rv-1819/invitations*/

Unity 3D: ***2017.1.1f1***
GVR SDK for Unity: ***GoogleVRForUnity_1.170.0.unitypackage***
JDK: ***1.8.0_111***
Target API level: ***Level 26***

The main goal of this project is to design and setup an environment that includes a camera controllable by the user. The environment consists of a maze surrounded by mountains with a 360º video skybox. The maze was built on top of a randomly generated 2D maze that textured the initial plane.

The first task was actually the last completed. It was easier to setup the camera after all of the environment was built. At that stage, the camera initially had a *FlyCam* script that would allow us to move the camera with mouse and keyboard. In the end, we added the Gvr prefabs.

First we created a Unity terrain (*Terrain*). Using the **Raise/Lower Terrain** sculpt tools, with different brushes, we created the mountains. We used **Smooth Height** to remove abrupt changes and edges and the **Paint Texture** tool using brushes with two different textures, snow (*snow.jpeg*) on the mountain tops and the rocky terrain (*mountain.jpg*).

Using a generated maze image as reference, we created a material (*maze.mat*) with that image as texture. The material was applied to a plane (*Maze Floor*) from which we built the maze, adding scaled and rotated cubes. Finally, we grouped the maze walls into an empty game object, *Walls*. This last object and *Maze Floor* were moved into an empty game object, *Maze*. Finally we place in the scene *WarFireDragon*, a 3D textured dragon mesh containing

2392 triangles, animated.The controller was edited so it would loop in idle. We centered the maze in the terrain and the creature outside the maze's exit.

We imported a Unity asset to add flames to the scene. We placed in the scene several flame prefabs named *Fire_Parent*, which contained 4 different particles systems with a point light added for illumination. We moved the flame game objects to the *Maze* object in the scene hierarchy.

We used a spherical skybox to play a 360º video. We added a sphere (*Skybox*) and applied a scale transformation by a factor of 220 so we can cover the user environment. We defined. We inverted the normals of the sphere's texture by adding the *Flip* shader to the sphere material. That way, if we're inside the sphere, the light can interact with the material. Finally, we added the video to the *Skybox* which textured it with it. This video loops indefinitely.

As for the lights, we used a directional light to illuminate our scene. However, this caused the video on the skybox not to show up, because it also needed light projected on it. Because of that we ended up adding a second directional light pointing up, towards the top of the sphere.