
Computer Graphics Report

Assessment Task 3

Feedback

Georgia(My partner)

- Controls were difficult to pick up.
- Needs a respawn point at the first level.
- Wasn't sure where to go.
- Moving the mouse around when UI is up was annoying.
- Enemies take too long to kill.

Changes in response to feedback

- Added respawn point for the first level so the game doesn't teleport you back to the main level.
- Made floating text and particles to make it clear where the player must go.
- Need to fix UI interaction with NPC and add controls information.
- Can adjust enemy health or weapon damage easily in editor.

Evaluation of technologies, techniques and programs

Unity Shader Graph - is a node-based system that allows the user to visually author shaders and see the effects in real-time. Through class resources and tutorials online I was able to learn enough about Shader Graph to create multiple custom materials for various objects in my game. I played around with a few different styles before landing on something I liked for my game.

I decided on a nice bright glowing shader for a bunch of objects like enemy turrets, grapple objects and the swinging line renderer. With the shader I created I was able to change a few properties to give it a different look for each object. The shader has colour properties for the top and bottom to change colour through the object, the smoothness and metallic properties change the glow effect. This shader worked well for the objects in my game as the skybox was a dark galaxy themed material.

The second primary shader I created was a pixel shader that I used with the particle system to create a pixelated firework effect. This shader has a "Bits" property in this shader to adjust how pixelated the texture is. I applied this material to the renderer component of particle system and tweaked a few other settings (noise, emission rates, rotations) to get the firework looking effect. These objects were used in conjunction with a post processing volume to give them the glowing effect.

Animations and Interactable GUI - I set up a character model at the start of the game which can perform various animations at the click of a button. The animations were basic dancing animations and

mostly used to demonstrate the evidence criteria for our assessment. I packaged them with a silly narrative to give the NPC a purpose and the player an objective to complete. To do this I call "CrossFadeInFixedTime" on the animator component and pass in the name of the animation and the time it takes to transition.

The GUI consists of a few buttons and text that the user can interact with when they are close enough to the Dancing NPC. There is one button that brings up quest text to give the player some direction. All this UI disappears when the player leaves the trigger zone around the Dancing NPC through the "OnTriggerExit" function. Once the NPC dialogue button has been clicked then the player can access the first level. To do this I added a collider to the teleport point, and it only becomes enabled once the quest has been given. The particles change and some "Level 1" text appears over the teleporter to direct the player to the right location. Most of these changes are simple enabling and disabling of objects through code. This works the same way when the player completes the level and must return to the main platform.