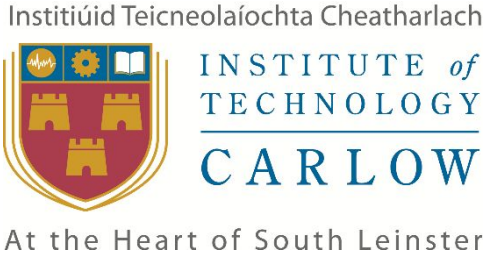

Computer Games Development
CW208

Technical Design Document

Year IV

Brian O'Neill

C00213756

03/05/2020

Contents

Game Architecture

List of Scripts and their usage:

GrassStats - Used to store and apply the grass stats.

TreeStats - Used to store and apply the grass stats.

Lsystem - Used to create the Trees and the Grass in the scene.

Manger - Used for updating/toggle the Ui and also for the deletion of the objects selected.

TiledRoadCreator - Used to create the roads depending on the angle of the directional tool which is also handled inside this script. Used to create the area and also to reset the area.

TEst - Used to create the buildings as a whole. (Floors,walls,roof and also all the components needed including colliders and also the rigidbody.)

ObjectCollisions - placed on the trees,grass,buildings,etc. To tell them what to do once they have collided with certain objects.

TileCollisions - Used on the road objects to decide what object to destroy if overlap.

GrassCreator - Set the number of grass objects that can be created depending on the box size.Also decides the position of those grass objects. Also adds the object created to the parent object.

RockCreator - Set the number of rock objects that can be created depending on the box size.Also decides the position of those rock objects. Also adds the object created to the parent object.

TreeCreator - Positions the trees based on the poisson disk sampling in this script and then based on how many points are returned,create that number of trees.

RockGenerator - takes a basic sphere mesh and then applies perlin noise to each individual vertice of the mesh to make the object look more like a rock.

Wave - takes the mesh from the object that this script is on and then makes the mesh wave to simulate waves in the scene.

LightEnable - Used to enable and disable the lights in the scene.

MouseEvents - Changes the color of an object that the mouse is over and then reverts it after the mouse is not over it.

SavePrefab - Used to save the whole object in the scene(Commented out since there was problems towards the end of the project.

SelectBox - Used to update and create the rectangle used to select the area that the other objects are spawned inside while also updating all variables.

Utility - A few functions that had been used in multiple other scripts so I just put them all in here and use them from here. Also this is where the combine mesh method is for combining all the gameobjects meshes.

charCont - Used to control the position of the moveable camera.

RotateToMouse - Rotates the camera with mouse movement.

RoadCreator - Old Method Used for an old script that is no longer in the project but kept incase I needed it again.

RoadGenerator - Old method for making roads, Currently not in use.

TestGen - Old method for making roads, Currently not in use.

Features

Feature: Road Creation

Tasks:

1. Create a script named Tiled Road Creator, This script will perform the creation of roads. This script will hold the code for the directional tool as well as the conditions necessary for the creation and alteration of roads.
2. Create an empty gameobject.
3. Apply the Road Creator script to the object.
4. Populate the public fields on the script with the correct objects.

Feature: Building Generation

Tasks:

1. Create a Builder script, that will allow the generation of buildings along the road as the road is being created.
2. First I began creating the floor of each room.
3. After the floor was done, I began to create walls that would surround the floor and encase it.
4. I then looked into repeating this pattern on each floor of the building by making a new room on each floor.
5. After this and I had a multi tier building I added a method to create the roof on top of the building.
6. Attach this to the same empty object as the road creator script.

Feature: Vegetation

Tasks:

1. Create an L-System Script that will handle the generation of the trees and also the grass in the game world.
2. Inside this script add any components the object might need after creation like a collider,rigidbody.
3. Fill in the necessary parameters on the script in the gameobject and then save it.
4. Place this script onto an empty Gameobject and add this object as a prefab to the game.
5. From here I made another script called Tree Creator and inside this script is where the rectangle box creator will create these objects inside that rectangle.
6. Place this script on an empty gameobject and this is called whenever the correct option has been selected and the box is being used.
7. This works both ways for trees and the grass.

Feature: Ui Controls

Tasks:

1. Added an event system to my scene.
2. From here I started by adding a road type selector which allows me to change the type of road that my road creator will use when making the roads.
3. Next I added a reset button that allowed me to reset the scene so if you make a mistake you can reset the editor.
4. I then added a switch camera button that allows the player to change perspective and move in and around the scene.
5. Once I had the trees,rocks and grass to a standard I was happy with I added creation buttons on the side that allows me to create these objects but only if they are ticked.
6. The player will use a rectangle select tool to choose the area they wish to make these objects.
7. I added a save button and track name to the ui to allow the user to name the area they create and then save it.

Feature: Combining Meshes

Tasks:

1. Added a function to my utility class that gets all the mesh filters of all the child objects.
2. Using those filters it adds all the meshes to the correct lists and also saves the materials to their own list.
3. Once it has checked all the filters on the parent and all its children,.
4. Then I combine all these mesh variables into one and apply it to the original mesh of the object I passed it.
5. After this is done I delete the old objects that I have taken the meshes off.
6. This helps alleviate some performance issues.
7. I then call this function on any object that I might require.

Feature: Directional Tool

Tasks:

1. The Directional tool is created using a line renderer that is placed on the road creator object.
2. The line renderers have two points, one point is set to the previously created object and the other is placed on your cursor so when you move around you mouse the new object is created depending on the angle of the line.
3. Once you are ready release the mouse button and then the algorithm will take in the angle and then create the correct road.

CRC Cards

Class Name : TiledRoadCreator	
Subclasses :	
Superclasses :	
Responsibilities **<u>Variables</u>** ● Main Camera ● Player Controlled Camera ● Straight road ● Corner road ● Intersection ● T-Junction ● Grass Tile ● Ramp ● Bridge Tile ● Corner Bridge Tile ● Road Lamp ● Street sign ● Traffic Light ● Area Size ● Spawn Chance for the building ● Overall Area Parent Object ● Container for the roads ● Current object ● List of ground tiles. ● List of roads ● Line renderer Object ● Is the mouse moving ● Rotation of object ● Position of the current Tile ● Start point (position of most recent created tile)	Collaborators ● Test Script(Building Creator Script) ● Utility script ● Drop down list(part of the UI) ● Button for resetting. ● Line renderer component

• Selector angle(angle of line renderer) • Previous Direction • Collider position(for checking for water) • firstHitWater(checks when you hit the water) • originalY (original height before moving up) • Loading in the objects into the script **Methods** • Create a Road Object. • Alter road given correct circumstances. • Spawn building when a road is being created. • Creates roadside objects. • Updates and creates the line renderer to show the direction of the next road being created. • Resets the scene. • Changes the tile type depending on the players selection	

Class Name : TEst	
Subclasses :	
Superclasses :	
Responsibilities **<u>Variables</u>** ● Building Parent Object ● List of floor tiles ● List of wall tiles ● List of corners ● List of roof Tiles ● Wall Object ● Floor object ● Window object ● Window Variants ● Roof ● Door ● Fence ● Mailbox ● PathTile ● IndoorLight ● Number of floors of the building ● Size of the building floor ● Position of the building ● Original Color **<u>Methods</u>** ● Create the Overall Building ● Create floor ● Create Walls ● Create Mailbox ● Create Garden ● Create Path ● Create House Light for indoor illumination ● Create Roof	Collaborators ● Utility script.

• Create Basic Roof • Create Rounded Roof • Create Slanted Roof • Create Pyramid Roof • Create Flat Roof • Create TwoTiered Roof • Create Barn Roof • Create Chimney for the roof • Create Dormer Window for the roof • Add Necessary components to the building. • Add necessary garden components • Clear the list • Select the correct roads.	

Class Name : Lsystem	
Subclasses :	
Superclasses :	
Responsibilities **<u>Variables</u>** ● Object you are using ● The type you want to use ● Axiom ● Number of iterations ● Scaling applied to objects ● Angle of rotation ● Dictionary of rules to be followed by the system ● currentString to be iterated over ● Stack of transforms ● Is it generating ● Empty parent object for the object ● Starting angle of the object ● Density of the grass ● Spacing between the grass **<u>Methods</u>** ● Generate the object through the rules. ● GenerateLSystem(Iterations) ● Generate Single Iteration ● Reset Start Up rules	Collaborators ● Utility script.

Class Name : SelectBox	
Subclasses :	
Superclasses :	
Responsibilities **<u>Variables</u>** ● Current camera ● Image of the Square ● Start Point ● End Point ● Centre of the box ● Size of the area of the box ● Is it being moved **<u>Methods</u>** ● Gets the mouse position in the world ● Gets the magnitude of the line ● Update the size of the box with a mouse drag event and scale the image as well as updating all the values necessary.	Collaborators ● Camera in the scene ● Square Image

Faculty of Science

Open-Book and Remote Assessment Cover Page

Student Name: Brian O'Neill

Student Number: C00213756

Lecturer Name: Philip Bourke

Module: Project II

Stage/Year: 4th Year

Date: 03/05/2020

Declaration

This examination/assessment will be submitted using Turnitin as the online submission tool. By submitting my examination/assessment to Turnitin, I am declaring that this examination/assessment is my own work. I understand that I may be required to orally defend any of my answers, to the lecturer, at a given time after the examination/assessment has been completed, as outlined in the student regulations.