# Sprint 6 – Endless Terrain Generation 22/03/2019 – 31/03/2019

# Abstract

The goal of this sprint is to create a graphic interface were the user can change the terrain settings and the level of detail of the terrain during the execution of the program. The level of detail of the terrain is going to be used in a later sprint to create an endless terrain.

# Research

Previous knowledge in the creation of user interface for unity is going to be used. For the implementation of the LOD (Level of Detail) all the information that I found was using the pre-build functionality in unity. I tried to figure out a way how I could implement this functionality by myself (Figure 5).

# Sprint Review

During the implementation of sprint, creating the user interface was a straight forward process without any major issues (Figure 1).

The problems start rising when implementing a way to reduce the level of detail of the terrain. One of the problems that took me some time to understand how to solve it, was the fact that each time I decreased the level of detail some quads on the mesh were not rendered properly (Figure 2). To resolve this issue duplicated my function named SetMeshData with an int parameter, this allowed to update my mesh data by passing the map size / level of detail (Figure 3). Now that the level of detail of the terrain was decreased, but some of the triangles were invisible I had to debug the program and going through the code, but I couldn’t find a way to fix it. I Tried to google it and research about Unity Level of Detail but the information that I found was not helpful at all, so I decided to draw my programming logic on the notebook (Figure 5), this not only helped me to figure out the solution but also opened my possibilities. I decided to make the terrain map , but I needed to know what numbers I could use to divide 255 without giving me a float value, to solve this I found that the factors of 255 were 1,3,5,15,17,61,85,255 (Figure 6). When I found the factors of 255 Implementing the final solution was easy (Figure 7).

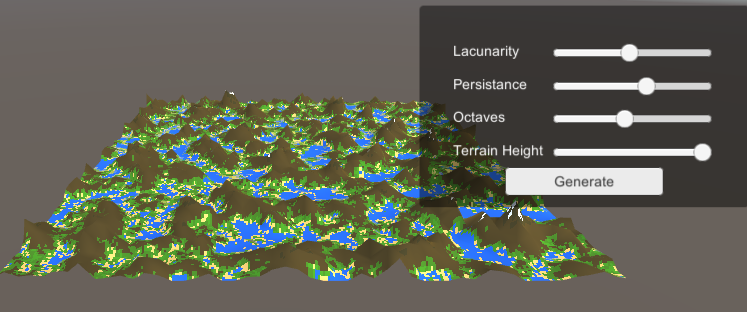


Figure 1 - User Interface

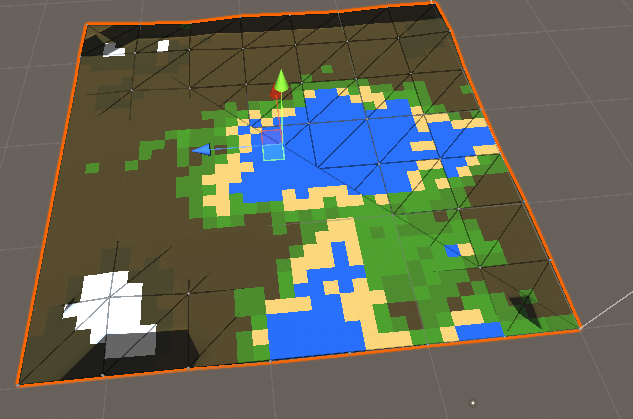


Figure 2 - Rendering Problem

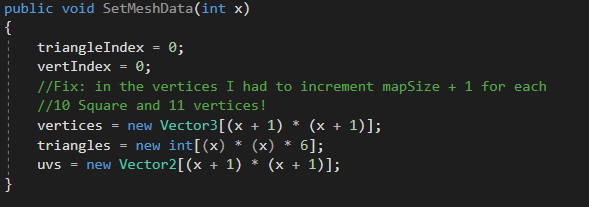


Figure 3 - Creation of the new function SetMeshData(int x)

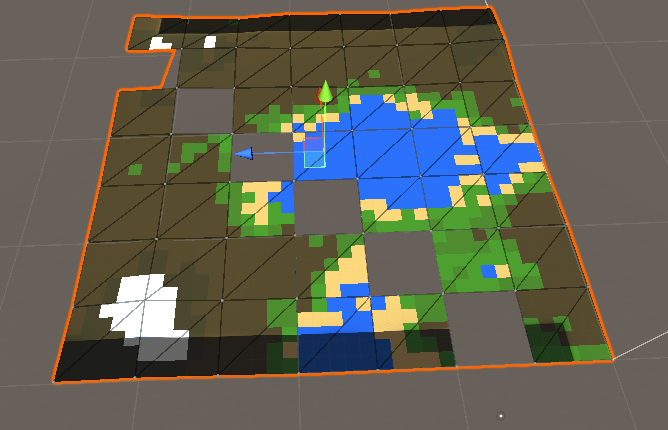


Figure 4 - Terrain Rendering, invisible triangles

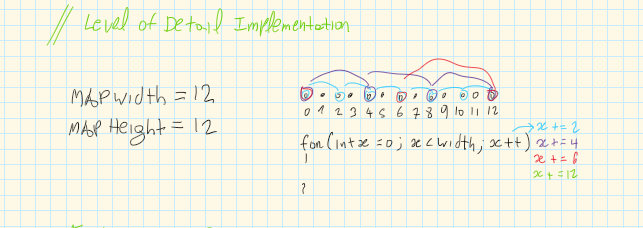


Figure 5 – LOD

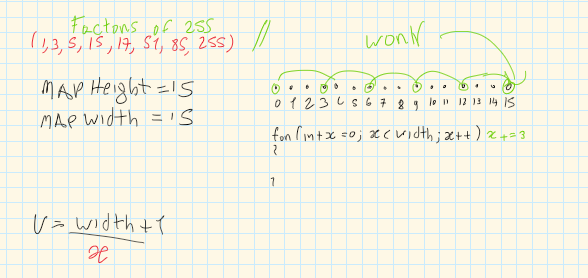


Figure 6 - Final Solution

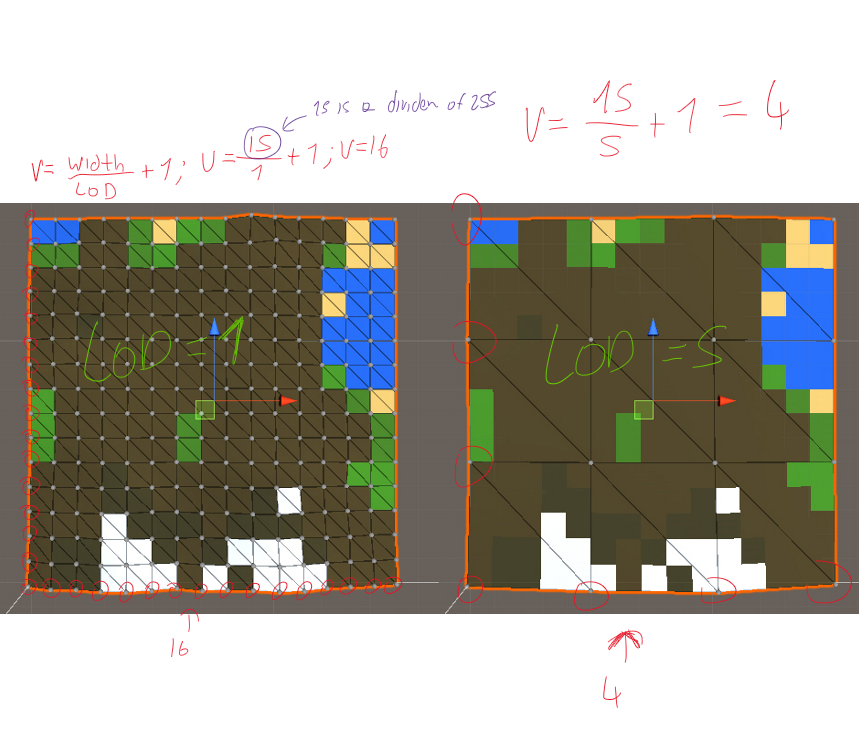


Figure 7 - Level of Detail Working

# WBS

1. Research (40%) (22 hours)
2. Implement graphical user interface (10%) (5 hours)
3. Implement the level of detail functionality (50%) (27 hours)

# Reading List

UNITY TECHNOLOGIES, 2019. *Unity User Manual (2018.3)/UI/Basic Layout*[viewed 10/03/ 2019]. Available from: <https://docs.unity3d.com/Manual/UIBasicLayout.html>

BRACKEYS, 2018. MAKE YOUR GAME RUN SMOOTH - Unity LOD Tutorial [viewed 10/03/ 2019]. Available from: <https://www.youtube.com/watch?v=ifNyVS2_6f8>

ARNDT, J., 2014. Setting up an LOD System - Unity Game Engine&nbsp;[viewed 11/03/ 2019]. Available from: <https://www.youtube.com/watch?v=IzlU_xvTK3Y>

Top of Form

Bottom of Form

# References