# Abstract

Contents

[Abstract 1](#_Toc97201468)

[1.0 Introduction 3](#_Toc97201469)

[2.0 Literature Review 3](#_Toc97201470)

[2.1 Mesh Generation and Rendering 3](#_Toc97201471)

[2.2 Procedural generation techniques 3](#_Toc97201472)

[2.2.1 Heightmap 3](#_Toc97201473)

[2.3 Atmospheric Rendering 3](#_Toc97201474)

[3.0 Design and Implementation 3](#_Toc97201475)

[4.0 Testing 4](#_Toc97201476)

[5.0 Evaluation 4](#_Toc97201477)

[6.0 Conclusion 4](#_Toc97201478)

# 1.0 Introduction

# 2.0 Literature Review

## 2.1 Mesh Generation and Rendering

The most basic component of the system that will be implemented as part of this dissertation is the creation and generation of a sphere mesh. This is crucial as in reality, all planets typically follow a spherical shape, due to gravitational forces pulling material to the centre of the planet (Sears 2022). For this element there exists a wide variety of techniques and algorithms to make this initial sphere. Such techniques include: UV spheres, normalized cubes, spherified cubes and icosahedron are some of the more popular methods (Cajaraville 2019). These procedures can have their effectiveness evaluated based on their: computational efficiency, distribution of vertices and how close the generated points are to the unit sphere. One additional method that fulfils all of these criteria is the Fibonacci sphere (Patel 2022). This algorithm allows for the most evenly distributed vertices compared to the previously described methods, and as remarked by Keinert et al, is a “well-known approach to generate a very uniform sampling of the sphere” (2015, 7). Unfortunately, due to the non-linear generation of the vertices, triangulating these points would prove computationally difficult (*Coding Adventure: Procedural Moons and Planets* 2020). Another downside of this approach would be the inability to allow for easy implementation of level of detail systems, due to the nature of how the points are laid out, and the computational complexity of triangulating them (Source needed). Due to these issues with triangulation, other methods were explored. One very promising technique is called the marching cubes algorithm. The method uses voxels, which is defined as “a value on a regular grid in three-dimensional space”(Anon. 2019). The algorithm works using a set of 8 voxels to form a cube, then generating a triangle based off these 8 values (Sin and Ng 2018). This technique is typically used on flat terrain, however the paper written by Sin and Ng demonstrate a method to transform the voxels into the unit sphere, allowing for the creation of spherical objects (2018).

## 2.2 Procedural generation techniques

In Computer graphics, there are many methods for procedural content generation. One of the more popular techniques within this field is the use of noise functions (Reference needed). Noise is defined as, “the random number generator of computer graphics” (Lagae et al. 2010). Of theses noise functions (such as Perlin, simplex and anisotropic) each function has their own characteristics, such as coherency.

## 2.3 Atmospheric Rendering

2.4

# 3.0 Design and Implementation

# 4.0 Testing

# 5.0 Evaluation

# 6.0 Conclusion