KV6003: Individual Project
Project Terms Of Reference

Cuthfbert Mutumbwa

Design and Implement a Rendering Engine

7th November 2018

1 Background

Creating A rendering engine is a task that requires one to understand multiple different topics and subjects, for example not only will i need to know how lighting works, but how it interacts with other objects, and how to represent it on screen. I will also need to learn how textures work, how to wrap them around an object and how the light will interact with the texture wrapped around the object.

In college i used two piece of software that are called Unreal Engine and Blender. Unreal Engine was a game engine, and Blender was a 3D Creation suite that can be used as a rendering engine. My first question when using these softwares where "How does one Create 3D graphics software", then i realised i also was unsure on how someone designs these software.

I want to create a rendering engine that uses core rendering techniuse and features, and Software engineering practices to design the engine. for design I have opted to use different types of litrature such as Designing a Modern Rendering engine [Bauchinger]. I will also litrature to choose which type of technique I will use to implement my features, this is because there are multiple different ways to implemnt features such as shadows, lighting, and even the way the engine will render a scene.

Rendering engines are still or more specifically rendering is a very interesting field of study. New rendering features are being made to improve quality in 3D animated films, and recently Nvidia released a new Graphics Card to allow Real-Time raytracing in Video Games[1]. Such Acheivments is one that is truly impressive as raytracing is known for being computationaly expensive.

2 Proposed Work

I will be using C++ as my programming language of choice,and OpenGL which is a graphics API that will allow me to create graphics progra The goal for creating a rendeirng engine is to manage graphical effects, such as lighting, shadows, texture mapping, and to apply them to a 3D scene and have them interact with other

[1] https://www.pcgamer.com/uk/what-is-ray-tracing/

3D objects. A rendering engine should not be confused with a game engine which would have me implement a physics system, and an animation system aswell.

I wish to build an engine with a 3D scene that will showcase various graphics effects. I want to be able to move around the 3D space using a camera, so i can be able to look at the scene at different perspective.I want to be able to turn toggle of, and on certain features to be able to show what the scene will look like without them, for example toggling off shadows to show how they may impact the scene visually.

3 Aims and Objectives

Aims

- To Build a Rendering Engine
- To design a maintanable piece of software using modern design principles

Objectives

1. Use software engineering design principles to implement the renderer.
2. Learn the required theory that will be needed to further understand my project.
3. Implement lighting in scene.
4. Implement Shadows in scene.
5. Implement Shaders in scene.
6. Implement Skybox in scene.
7. Implement Reflections in scene.
8. Implement Texturing in scene.
9. Implement a 3D scene enviroment to showcase the implemented features.
10. Implement Model Loading into scene.
11. Test and debug the system using tools specific for graphics programming.

4 Skills

1. Programming in C/C++ (from module KF4006)
2. Software Engineering (from module KF5012)
3. Algorithms (from module KF5008)
4. OpenGL API (Learnt enough to build a 3D scene)
5. Computer Graphics Theory (Currently Learning in Module KF6018)

5 Resources

Hardware

- Computer

Software

- Visual Studio
- GPU debugger

6 Structure and Contents of Project Report

Planned Chapters

1. Designing the Renderer
2. Rendering Engine Theory
3. Implementing the Rendering Engine
4. Creating the Scene enviroment
5. Testing and Debugging using bespoke Software
6. Evaluating the Rendering Engine

7 Marking Scheme

Project Type Software Engineering projects

Project Report Analysis

1. Designing the Renderer
2. Analysing Rendering Techniques

Synthesis

1. Implementing the Rendering Engine
2. Creating the Scene enviroment
3. Testing and Debugging using bespoke Software

Evaluation

1. Evaluating the Rendering Engine

Product Deliverables list

1. Class Diagram
2. Use Case Diagram
3. Rendering Engine Source Code
4. Test Plans

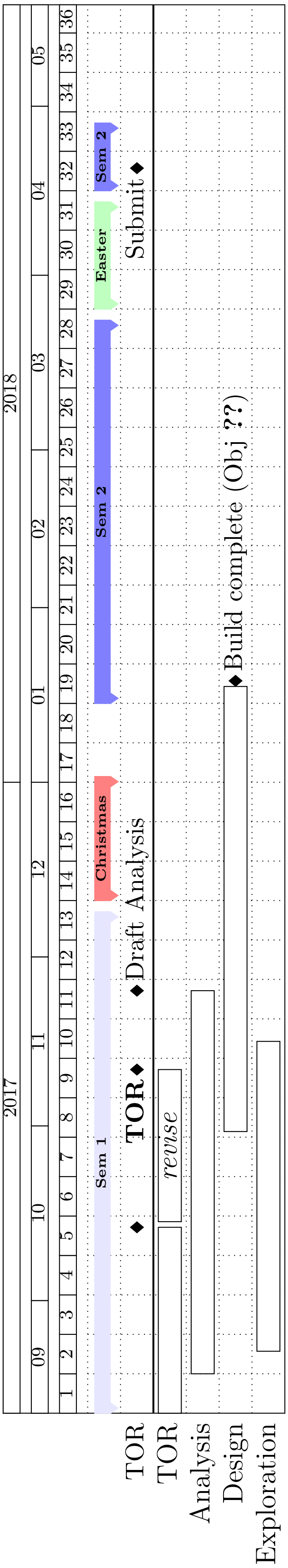
8 List of Appendices

Will Add soon

9 Project Type

This will be a Software engineering project

10 Project Plan



11 References

Matthias Bauchinger. ausgeführt am institut für computergraphik und algorithmen der technischen universität wien.