RAMAIAH
Institute of Technology

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CSE08: COMPUTER GRAPHICS

TERM: Jan-May 2020

BLENDER PROJECT

Submitted to
Dr. S Seema
(Professor)

Submitted By
Darshan R Konnur
1MS17CS026

Rubrics for Assignment 1:

Assessment	Excellent (10)	Very Good (6-8)	Satisfactory (3-5)
Assignment 1 (Blender)	A complex scene with all graphics features rendered	A complete scene with few concepts highlighted	A simple scene with minimal effects

Signature of the Guide

ACKNOWLEDGEMENT

I, Darshan R Konnur, student of Computer Science Engineering of 3rd year in Ramaiah Institute of Technology, Bengaluru, it is my privilege to express my sincerest regards to my project guide, Dr. S Seema, for the valuable inputs, able guidance, encouragement, whole-hearted cooperation and constructive criticism throughout the duration of my project. I deeply express my sincere thanks to our Head of Department Dr. Anita Kanavalli for encouraging and allowing us to have this amazing course in our curriculum. I would like to express my deepest appreciation to all those who provided me the possibility to complete this report.

TABLE OF CONTENTS

Chapter No. **Title** **Page No.**

ACKNOWLEDGEMENT…….…i

1. INTRODUCTION

1.1 Purpose……..……...1

1.2 Scope…….……...1

1.3 Definitions and abbreviations.…….....1

1.4 Overview.……...…….……....1

2. OVERALL DESCRIPTION

2.1 Project perspective

2.1.1 Hardware Interfaces ……1

2.1.2 Software Interfaces……1

2.2 Project functions……...2

2.3 Constraints……2

3. SPECIFIC REQUIREMENTS

3.1 External interface requirements

3.1.1 User Interfaces ……2

3.1.2 Hardware Interfaces ……2

3.1.3 Software Interfaces……3

3.2 Performance Requirements...……...3

4. BASIC METHODOLOGY

4.1 3D Modelling……...3

4.2 Sculpting……...3

4.3 Texturing……..3

4.4 Lighting……4

4.5 Rendering images/animation……4

5. CONCLUSION…….....4

6. REFERENCES……...5

1. INTRODUCTION

1.1 Purpose
The purpose of the project is to develop 3D application using 3D Blender tool and to build a complex 3D scene and apply rendering techniques and demonstrate it.

1.2 Scope
The 3D application was built using Blender software. Blender is a free and open-source 3D computer graphics software toolset used for creating animated films, visual effects, art, 3D printed models, motion graphics, interactive 3D applications, and computer games. Blender's features include 3D modeling, UV unwrapping, texturing, raster graphics editing, rigging and skinning, fluid and smoke simulation, particle simulation, soft body simulation, sculpting, animating, match moving, rendering, motion graphics, video editing, and compositing.

1.3 Definitions and abbreviations
- **Mesh:** A mesh is a collection of vertices, edges, and faces that describe the shape of a 3D object.
- **Modeling:** Modeling is process of creating 3D models using different meshes given and by tweaking around those fundamental meshes.
- **Sculpting:** Sculpting is the use of software that offers tools to push, pull, smooth, grab, pinch or otherwise manipulate a digital object as if it were made of a real-life substance such as clay.
- **Rendering:** Rendering is the process of turning the digital description of a scene, at a particular moment in time, into an image. Rendering is a lot like taking a photograph of a scene that only exists in the computer. Scenes are described in simplified, human-accessible ways, and the computer then uses that description to calculate all of the implied (non-explicit) things to determine the colour of each pixel in the entire image.

1.4 Overview
I used to play lot of action shooting games, out of all I have played, the maps had the ruined areas. I was always fascinated creating one such. So, this project aims at rendering a 3D model of a 3D real world size model of a ruined building (Ruins).

2. OVERALL DESCRIPTION

2.1 Project perspective
The Project perspectives are mentioned below. Hardware interfaces and Software interfaces which are aimed to be rendered are as follows.

2.1.1 Hardware interfaces
Blender being a software platform for creating 3D applications, supports 3 render engines, namely, Cycles, Eevee, Workbench. Workbench just renders using the default 3D viewport drawing system for modeling, texturing, etc. Eevee is Blender’s very own very powerful real time rendering engine. Cycles engine is an unbiased path tracing render engine using CPU or GPU(if available).

2.1.2 Software interfaces
Blender provides many editors such as video editing, animation, scripting, motion tracking, game logic, etc. Hence, is a handy software.

2.2 Project functions
The primary aim/function of the project is to render an animation video for the theme I have opted for.

2.3 Constraints
There were plenty of constraints while doing this project. To list some, firstly, I was using a laptop with not much powerful GPU to render quickly, fortunately, it was par the optimal requirement. Secondly, the availability of textures for the models were very less on the internet, so had to rely on untrusted sources for the same.

3. SPECIFIC REQUIREMENTS

3.1 External interface requirements

3.1.1. User interfaces
Blender's user interface incorporates the following concepts:

- **Editing modes:** The two primary modes of work are Object Mode and Edit Mode, which are toggled with the Tab key. Object mode is used to manipulate individual objects as a unit, while Edit mode is used to manipulate the actual object data. For example, Object Mode can be used to move, scale, and rotate entire polygon meshes, and Edit Mode can be used to manipulate the individual vertices of a single mesh. There are also several other modes, such as Vertex Paint, Weight Paint, and Sculpt Mode.
- **Hotkey usage:** Most of the commands are accessible via hotkeys. There are also comprehensive GUI menus.
- **Numeric input:** Numeric buttons can be “dragged” to change their value directly without the need to aim at a particular widget, as well as being set using the keyboard. Both sliders and number buttons can be constrained to various step sizes with modifiers like the Ctrl and Shift keys. Python expressions can also be typed directly into number entry fields, allowing mathematical expressions to specify values.
- **Workspace management:** The Blender GUI builds its own tiled windowing system on top of one or multiple windows provided by the underlying platform. One platform window (often sized to fill the screen) is divided into sections and subsections that can be of any type of Blender's views or window-types. The user can define multiple layouts of such Blender windows, called screens, and switch quickly between them by selecting from a menu or with keyboard shortcuts. Each window-type's own GUI elements can be controlled with the same tools that manipulate 3D view. For example, one can zoom in and out of GUI-buttons using similar controls, one zooms in and out in the 3D viewport. The GUI viewport and screen layout are fully user-customizable. It is possible to set up the interface for specific tasks such as video editing or UV mapping or texturing by hiding features not used for the task.

3.1.2 Hardware interfaces
● **Minimum:**
- 64-bit dual core 2Ghz CPU with SSE2 support
- 4 GB RAM
- 1280×768 display
- Mouse, trackpad or pen + tablet
- Graphics card with 1 GB RAM, OpenGL 3.3 Less than 10-year-old

● **Recommended:**
- 64-bit quad core CPU
- 16 GB RAM
- Full HD display
- Three button mouse or pen+tablet
- Graphics card with 4 GB RAM

● **Optimal:**
- 64-bit eight core CPU
- 32 GB RAM
- Full HD displays
- Three button mouse and pen+tablet
- Graphics card with +12 GB RAM

● **Supported Graphics Cards:**
- NVIDIA: GeForce 400 and newer, Quadro Tesla GPU architecture and newer, including RTX-based cards, with NVIDIA drivers (list of all GeForce and Quadro GPUs)
- AMD: GCN 1st gen and newer (list of all AMD GPUs)
- Intel: Haswell and newer (list of all Intel GPUs)
- macOS: version 10.12 or newer with supported hardware

3.1.3 Software interfaces
Blender is available for Windows 7 and above, Mac OS X 10.6 and above, and Linux. Blender 2.76b is the last supported release for Windows XP and version 2.63 was the last supported release for PowerPC.

3.2 Performance requirements
- Modeling of 3D structures using meshes and other add-ons**.**
- Sculpting must be on point to make it realistic.
- Prior knowledge on rendering engines.
- Lighting should be proper to constitute photorealism.
- Rendering techniques are essential for smooth and time saving and high-quality rendering.

4. BASIC METHODOLOGY

4.1 3D Modelling
First the idea for a blender project has to be concreted. After that, the required models are designed from the basic mesh objects or from the add-on objects in the modeling panel.

4.2 Sculpting
When the required models are ready, the models that need sculpting are sculpted using many brushes provided by blender under Sculpting panel. We can push, pull, smooth, grab, pinch or otherwise manipulate a digital object as if it were made of a real-life substance such as clay.

4.3 Texturing
Once all the objects are completely modelled and are sculpted, it is time to add on a material and a texture to an object. Blender provides a platform where we can unwrap the faces of the objects and can texturize them or even texturize them using nodes.

4.4 Lighting

Before setting up the lights for scene, we need to know about the rendering engine that we are going to use in the project to render an image/animation. Because of the variation in casting shadows of rendering engines, it is essential to choose a rendering engine first. Once after we choose it, lighting for the scene is set up. For lighting, blender provides point lights, sun light, spotlights and also HDR lights. HDR lightings were used for the project.

4.5 Rendering images/animation

We have selected a rendering engine for our project. Generally, Eevee engines are used in gaming animations where the shadow detailing is less priority, whereas, Cycles engines are used in the real video animations that require proper detailing of every material used. Since the constraints and also much detailing wasn't required, I have used Eevee engine to render the project. We can render the project as an image or as an animation.

5. CONCLUSION

Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline—modeling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation. Advanced users employ Blender's API for Python scripting to customize the application and write specialized tools; often these are included in Blender's future releases. Blender is well suited to individuals and small studios who benefit from its unified pipeline and responsive development process. I could successfully render an animation from the scene that I modelled on blender. It was amazing learning blender and was fun too. Hopefully, I will be doing many more projects on Blender.

6. REFERENCES

- https://www.youtube.com/watch?v=MF1qEhBSfq4&list=PLa1F2ddGya_-UvuAqHAksYnB0qL9yWDO6
- https://www.youtube.com/user/AndrewPPrice
- https://www.udemy.com/course/blendertutorial/
- https://www.udemy.com/course/architectural-design-animation-in-blender-3d-graphics/
- Blender Documentation.
- Google images for references.