3D Model Viewer – Requirements

CSC 478 - Spring 2014 - Group 1 Choppers

Andrew Baker

Adam Le

Todd Previte

# Introduction

## Scope of Product

Design and build an interactive 3D model viewer, capable of displaying and manipulating geometry in 3D space.

The application will be built with objective-C in Xcode for the Mac OS X platform. If necessary, C or C++ interfaces or bindings may be used if the need arises.

OpenGL 4.0 is the minimum target API level. Pre-built 3D rendering modules and engines are not considered as part of the scope of this project. All elements of the OpenGL 4.0 or higher shall be available for use within the project.

This project is targeted at the Mac OS X platform. Other platforms are not considered as part of the scope of this project.

The application will run as a standalone application. The application will not be interacting with any other applications nor require any other applications. The interface of the application will be graphical based.

## Definitions

OpenGL – OpenGL is an application programming interface for rendering 3D graphics.

GUI – Graphical user interface

U/V – denotes the axes of the 2D texture.

API – Application programming interface

## References

NEHE OpenGL tutorials - <http://nehe.gamedev.net/>

OpenGL - http://www.opengl.org/documentation/

Parser Validation Tool – Unity3D

OBJ file format – <http://en.wikipedia.org/wiki/Wavefront_.obj_file>

# General Description

## Product Perspective

The 3D model object viewer application is being built to display and manipulate 3D models. The product was selected based on the area of expertise of the members of the project. The team members have a considerable amount of experience in the realm of digital graphics processing. This project will allow the team to showcase their previously acquired skills and build a functioning 3D model viewer.

## Product Functions

The application will be separated into three main modules, the graphical user interface, the 3D model file parser/loader and the renderer. The interface will facilitate the selection of 3D object files and GUI objects that will manipulate the 3D object. The file parser/loader will take in file information from the GUI, parse the data from the file and load it into memory. Control will then be passed to the renderer to manipulate and display the 3D model in the content window.

## User Characteristics

The target user of this application is a digital graphics enthusiast. The application will display 3D models stored in files that they either created or found on the Internet and view them in a Mac OS X application. The application will then allow for the manipulation of the 3D object. The user can edit their files in a separate editor and view the results of their modifications in the 3D model viewer.

## General Constraints

The application will only support one 3D object file format. This is due to the time constraint and limited man power available to this project. It will also not be able to perform animations. It will not modify the geometry, texture coordinates, textures. It will not export the 3D model to alternate file formats or save back to its original format. The 3D model viewer is just that, a 3D model viewer.

## Assumptions and Dependencies

The application will requires that the target machine have OS X Maverick installed.

# Specific Requirements

## Black-box Requirements

### Select and load 3D models saved in the OBJ file format

### Disable and enable lighting

### Display flat shading or smooth shading

### Select wireframe rendering mode or solid mode (triangles mode)

### Show surface normal or disable them

### Using a mouse to rotate the camera on the 3D model

### Zoom in and out on the 3D model

### Provide sample 3D models

## White-box requirements

### Load the information from OBJ files into memory

### Parse the information from OBJ files into objective-C structures