CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 Functional Requirements

Here, we have made an attempt to design a Code using OpenGL. The software has been written using C language and data structures like simple structure type are used.

The project requires the access of OpenGL utility toolkit through the use of the header file "gl/glut.h". This header file, in addition to the usual header files is needed for the working of the project. For running the program, any basic PC running compatible version of Microsoft Visual Studio is sufficient.

3.2 Non-Functional Requirements

The software should produce the informative error messages, if any errors are found in the input program. It should use memory as less as possible, dynamic memory allocation is preferable to accomplish this task.

3.3 Hardware Requirements

The minimum/recommended hardware configuration required for developing the proposed software is given below:

- PENTIUM-2 and above compatible systems.
- 512 MB RAM.
- Approximately 170 MB free space in the hard disk.
- Hard disk access time must be less than 19 milliseconds

3.4 Software Requirements

- Open GL

- WINDOWS XP/ME/9X
- Ubuntu

3.4.1 Why C Language for this project?

C is a minimalist programming language. Among its design goals were that it could be compiled in a straight forward manner using a relatively simple compiler, provided low-level access to memory, generated only a few machine language instructions for each of its core language elements and did not require extensive run-time support. As a result, C code is suitable for many system programming applications that had traditionally been implemented in assembly language.

Despite its low-level capabilities, the language was designed to encourage machine independent programming. A standard compliant and portably written C program can be compiled for a very wide variety of computer platforms and operating systems with minimal change to its source code.

3.4.2 Graphics in C

C itself doesn't support any graphics library. In order for C to be completely portable from platform to platform it has focused on providing platform independent functions, such as file access, text string manipulation, mathematical functions, etc. So in order to get graphics in C, one needs to do one of four things:

- Write a 16 bit DOS program and use assembly language for graphics
- Make calls to the Windows API
- Make calls to the OpenGL subsystem
- Use a graphics library

3.4.3 OpenGL subsystem

OpenGL provides a powerful but primitive set of rendering commands, and all higher-level drawing must be done in terms of these commands. Also, OpenGL programs have to use the underlying mechanisms of the windowing system. A number of libraries exist to simplify the programming tasks, including the following:

- The OpenGL Utility Library (GLU) contains several routines that use lower-level OpenGL commands to perform such tasks as setting up matrices for specific viewing orientations and projections, performing polygon tessellation, and

rendering surfaces. The library is provided as part of every OpenGL implementation.

- The OpenGL Utility Toolkit (GLUT) is a window system-independent toolkit. GLUT routines use the prefix glut.