Eric Arndt  
Michael Yahner  
Matthew Farkas  
  
Computer Graphics – Final Project – Due Wednesday December 5th, 2012

**Intro**

Our final project is a 3D game engine implemented using C# .Framework version x.x and Slim Dx.

**Functionality**

Shape Manipulations: The three main shape manipulations we learned in class were implemented including Translation, Rotation, and Scaling. Users can define two main types of shapes: Triangles and Squares which can be added, removed, and modified within the scene. Users also have the ability to turn objects into other shapes by using meshes (see Meshes below)

Meshes: – Meshes can be applied to objects (shapes) that will change their appearance for a user-loaded “.x” mesh file.

Lighting: Two different types of lights were implemented that we learned in class including Point and Directional light. Lights can be added, removed, and modified within the scene.

Physics: Life-like gravity was implemented using object translations and positioning of the terrain. As objects move farther away from its starting location, they appear to be under a realistic gravity force by accelerating.

Wireframe: The ability to convert a scene of objects into wireframe was implemented.

**What we each did**

**Eric Arndt:  
  
Michael Yahner:**

**Matthew Farkas:** My primary responsibility was to create most of the user-interface (GUI). The main layout was derived from existing game engines like Unity 3D, to include a main display panel shown in the middle of the screen surrounded with controls on the left and the right and a menu items bar. Descriptions of main GUI controls are as follows:

Mouse events controls for camera movement: Buttons were created to translate or rotate the camera along the x, y and z axis. In addition, mouse events were created to pan the scene by clicking and dragging within the panel and zooming in and out with the mouse wheel. When the camera is moved, labels are updated to indicate its current position.

Environment Checkboxes: A series of checkboxes were created and wired events to turn gravity on/off, turn wireframe on/off, turn global lights on/off, and to clear the current scene.

Object Input: An area for objects was created to keep track of all the user-defined objects. Menu bar items were created to easily add Squares and Triangles. For each shape that is added, a new entry would be created in the Object list drop down list. A textbox and button were created to rename objects so that they can easily be identified.

Object translations, rotation, and scaling input controls: Three areas were created to perform the basic shape translations, rotation, and scaling along the x, y, and z axis. Upon key-pressing the enter key, the shape will be positioned as indicated. Validation was added throughout to ensure a number was passed into each textbox. In addition, a color picker was created to color objects.

Light controls: An area was created, similar to the object area, to keep track of all user-defined lights. Menu bar items were created to easily add Point and Directional lights. For each light that is added, a new entry would be created in the Light list drop down list. In addition, text boxes were created with validation to manipulate the location and direction of the lights.

Function to select and load files for meshes: An open file dialog was created, that specifically loads .x files to be used as meshes for objects.