Table of contents

[Workload matrix 2](#_Toc437446893)

[I. Acknowledgement 3](#_Toc437446894)

[II. Introduction / Objective 4](#_Toc437446895)

[III. Target users 5](#_Toc437446896)

[IV. Description (task or technical details) 5](#_Toc437446897)

[V. Storyboard and Textures. 11](#_Toc437446898)

[Storyboard 11](#_Toc437446899)

[Textures 11](#_Toc437446900)

[VI. Print screen of your application with some description 13](#_Toc437446901)

[VII. Conclusion 15](#_Toc437446902)

[VIII. References 15](#_Toc437446903)

# Workload matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Task Name | Derick (TP032163) | Navresh Murugiah (TP024880) | Leysham Dripesh Doobree (TP031484) |
| Design and modelling objects. |  | √ |  |
| Applying textures. |  | √ | √ |
| Code Implementation | √ |  | √ |
| Documentation | √ |  |  |

# Acknowledgement

The aim for our project is to create a 3D environment by using the Open Graphics Library. Thank you for the guidance from Mr. Ewe Yee Seong, the lecturer of 3D Computer Graphics or 3DCG, we are able to achieve the aim of this project. Thank you for the guidance in learning the module and helping us in all of the problems we faced for this module.

# Introduction / Objective

OpenGL or Open Graphics Library is the standard API or application program interface for both 2D and 3D graphic images in computer industries (WhatIs.com, 2015). The aim of this project is to create a 3D in environment with C++ and by importing the Open Graphics Library. For this project, the aim is to create a 3D environment which is a small building in the middle of a forest with a classroom in it. The classroom has objects such as chairs and tables. The textures chose for the objects in the 3D environment can give a sinister feeling to the users. This 3D environment can be used for horror games.

For this project, these are the objectives that need to be done to achieve the aim of the project,

1. Creating objects in 3D max.
2. Adding textures to the objects.
3. Creating a camera that is able to navigate around the 3D environment.
4. Importing the objects to the 3D environment
5. Position the objects to create the described environment.

# Target users

Target users for this project is computer game students. Computer game students will have to create many games in their studies. Therefore, they may want to create a horror game. This project can be used for their game developments as the environment asset.

# Description (task or technical details)

The game world is made of the following,

* The ground
* The objects
* The camera
* Camera movement
* Light

Player Movement defines how the player will be moving the game camera around the scene.

The main keys for the movement would be:

* Up- to tilt the camera up
* Down- to tilt the camera down
* Left- to tilt left
* Right- to tilt right
* W- to move forward
* A- to move left
* S- to move backwards
* D- to move right
* Left CTRL- to crouch

The scene has been created using exported objects from other software such as 3DS MAX. Since the size that is needed in the scene cannot be determined in the software, the objects will have to be re scaled while implementing them. The placement of the objects will have to be calculated and implemented individually as even though there are duplicate of an object, their position in the scene cannot overlap others.

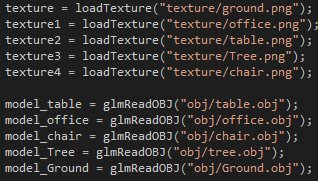
**Code Implementations**

1. Screen display



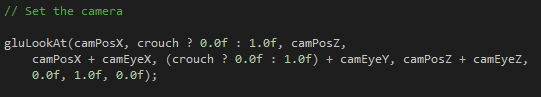
* WindowSize sets the Height and width of the produced window (resolution).
* WindowPosition is the position of the windows that has been created on the desktop.
* GLUT\_RGB sets the window to accept colors.
* GLUT\_DOUBLE- Double buffered window.
* GLUT\_DEPTH- Window with a depth buffer.

1. Importing Models and textures



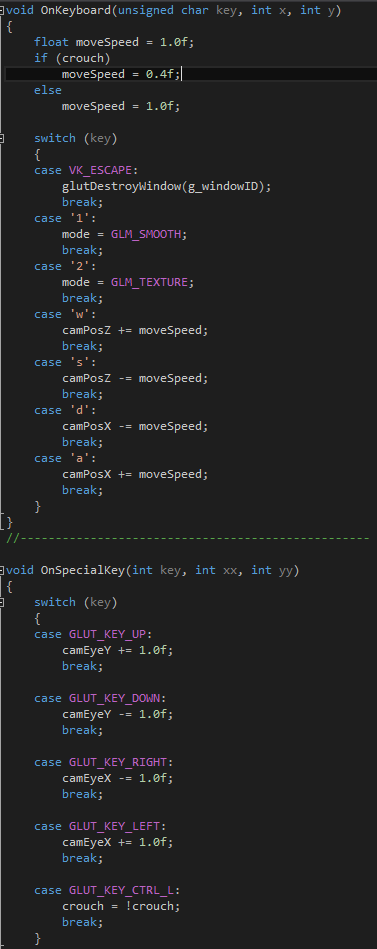
The above codes show the implementation and loading of the different textures and objects in the scene. These are coded in the in the init() function the initialises all the variables at the beginning of the program. The textures are loaded and the program looks for table.obj in the obj folder which is found in the project folder. The same applies to the textures. Once found, they are loaded into the program.

1. Camera movement and position



In the coding phase, the eye is the camera position with centre the camera viewpoint. The crouch is the position of the eye, whereby when activated, the eye Y position changes from the regular Y position to a lower one.

1. Player movement



Variable movementSpeed indicates point or vale to change when player is moving and the crouch is a state of player avatar in the world which activate with CTRL button.

The other two variables declared in this function are the camPosZ and camPosX which are explained below:

* camPosZ is increased when player press ‘w’ indication camera to move forward to the depth.
* camPosZ is decreased when player press ‘s’ indication camera to move back from the depth.
* camPosX is increased when player press ‘a’ indication camera to move right.
* camPosX is decreased when player press ‘d’ indication camera to move left.

This variable camEye main function in the program is for player to look around in the world. It relates to the tilting of the camera in the game world.

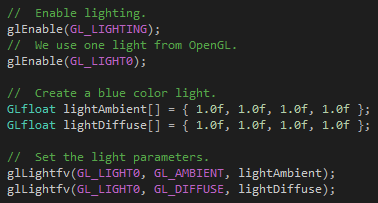
GLUT\_KEY\_UP| keyboard button up, rotate viewpoint up in Y.

GLUT\_KEY\_DOWN| keyboard button down, rotate viewpoint down in Y.

GLUT\_KEY\_RIGHT| keyboard button right, rotate viewpoint right in X.

GLUT\_KEY\_LEFT| keyboard button left, rotate viewpoint left in X.

1. Light



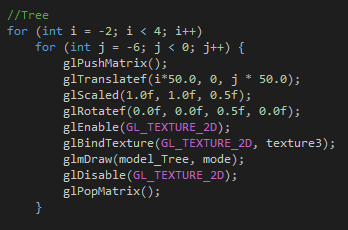
First piece of code enables the light for the game world. The second declares the amount of light sources in the game world, which in this case is only one.

1. Ground



For the drawing of the objects on the scene, matrix has been used. The vertex of the object is pushed onto the matrix. Then goes through translation, scaling and if needed rotation as well. Then the textures are applied to the object and finally the whole object with its texture is drawn in the scene. The same applies to table, chair and office.

1. Tree



For the implementation of the tree, nested for loops have been used. The reason behind it is the amount of trees that have to be placed in the game world. Thus using the for loops and setting the position of the X and Z axis where the objects are meant to be drawn. And using the condition to set the amount of trees to be drawn has provided an effective way of implementing them.

# Storyboard and Textures.

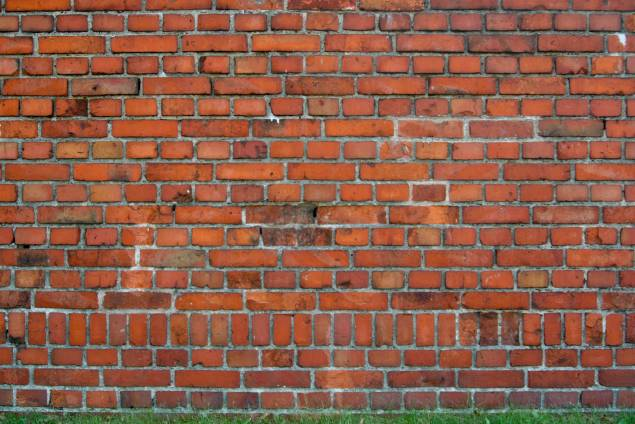
## Storyboard

The world is a horror environment in which there is a building in the middle of a forest. Inside the building, it is a class room with chairs and tables. The selected textures for all objects may enhance the creepiness of the environment. The environment will give sinister feeling for the users who is controlling the camera to look around and explore the 3D environment.

## Textures



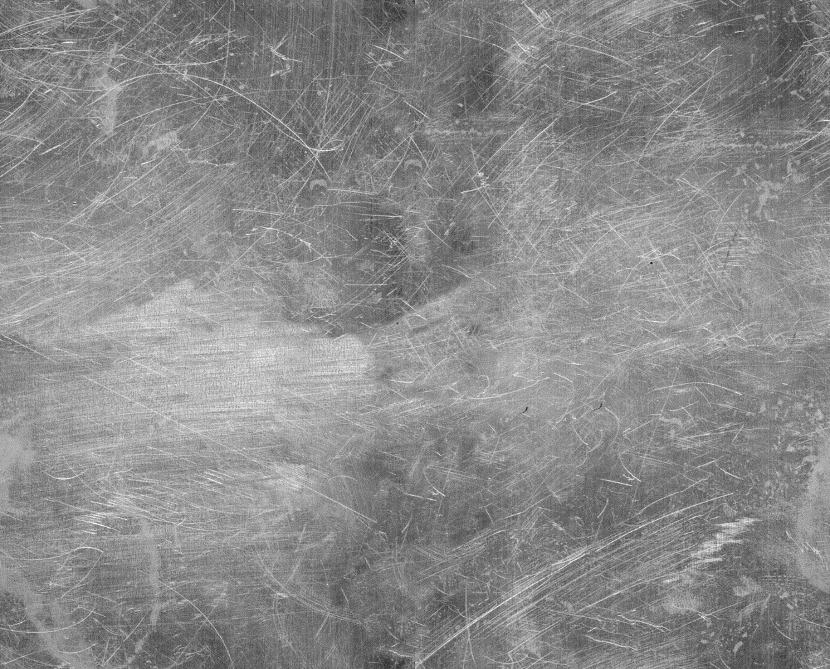
Table Texture.



Building Texture.



Ground texture.



Chair texture.

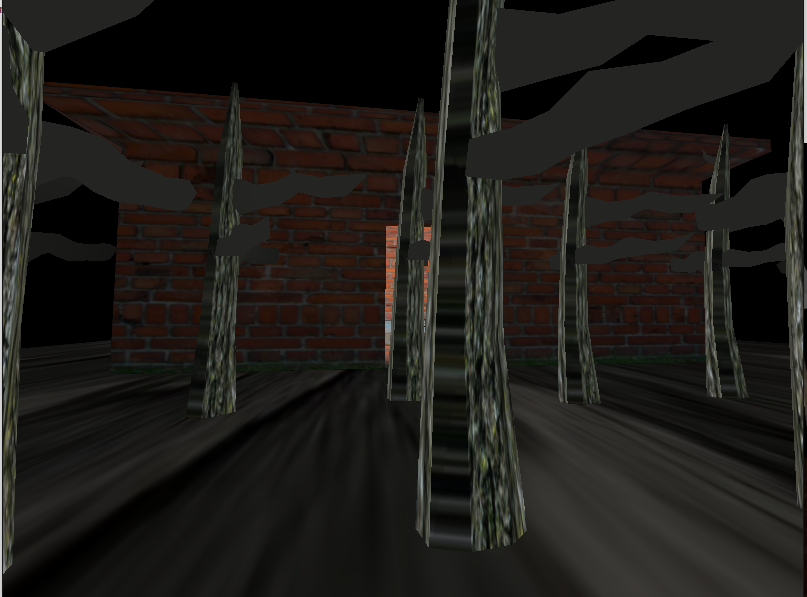


Tree texture.

# Print screen of your application with some description



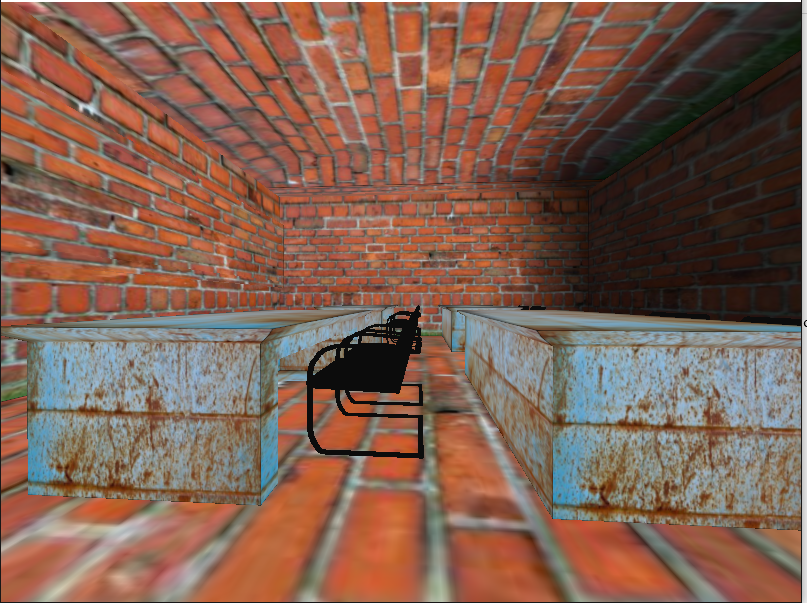
This is the 3D environment outside the building where there are many trees with dark textures and the sky is dark. Therefore, it gives a scary and sinister feeling.



Closer look to the building with brick textures



Back view of the class room. The classroom is filled with objects which are chairs and tables.



Side view of the class room. The chairs are black and the tables have the old wood textures.

# Conclusion

The 3D environment is created successfully. By completing all of the objectives that are mentioned earlier, the aim of our project has been achieved. The 3D environment is able to give sinister feeling to users. We hope that this project can help other computer game students as an asset in their game developments. Overall, we believe this project as a successful project by achieving the aim of the project.

# References

WhatIs.com, (2015). *What is OpenGL (Open Graphics Library)? - Definition from WhatIs.com*. [online] Available at: http://whatis.techtarget.com/definition/OpenGL-Open-Graphics-Library [Accessed 9 Dec. 2015].