Mid-Project Progress Report

Project Title: Graphical Enigma Simulator.
Student: Majed Monem
Supervisor: Prof. John Arnott

At this stage of the academic year I feel more confident of completing this project than I perhaps did at the start. This is mainly due to undertaking the Graphics module during first semester which has given me an insight on how to produce a graphical representation of the Enigma machine.

I feel that the project is progressing reasonably well, considering that I had no previous experience of programming a 3D graphics simulation, or even working with 3D graphics, until the Graphics module was well underway. The realistic mark which I think I am heading towards would be a 2:1 should I continue to work at the standard which I have been working at.

The purpose of this project is to develop a Graphical Enigma simulator which demonstrates how the enigma machine was able to encrypt plain text into cipher text. It shall demonstrate this in the form of 3D graphics which visually show the current being passed through the rotors as a character is being encrypted.

The aim for the first part of the academic year was to complete all the necessary documents and obtain ethics approval, all of which has been achieved, as well as background research. The documents are although revised on a regular basis to ensure they are up to date with the implementation being done.
Various documents have been created, these include the following;
Gantt chart (Appendix A)
Interface Design
Requirements
Specification Requirement
Use Case

The interface design includes ideas for the user interface which is to be implemented and the final design should match the implemented interface. The requirements documents contains a draft of the requirements which have then been formatted into the user case and specification requirement documents. As work is being done, it is being tracked using the Gantt chart for start and end dates. Along with these I have started the introduction and background sections of the final report, which are revised and updated on a regular basis. I have also been using Github as source control to back up my files and record progress made.

Before I could begin doing anything the first task was to familiarize myself with what the Enigma machine actually was and how it operated. Majority of this was literature reviewed online about the history of the Enigma machine and how it was used by the German army to communicate with their allies during the wars. The Enigma machine was an electromechanical device which scrambled a plain text message into ciphered text using a letter substitution system. It primarily used three rotors to scramble the letters, however different models were developed with a various number of rotors, to increase complexity of encryption. While researching, I have also came across various diagrams which demonstrated how the Enigma machine operated by passing current through the rotors once a key was pressed on the keyboard. Once the current is passed back, the encrypted letter is illuminated on the lightboard. Also various diagrams representing the components of a rotor where found, which I have adapted from to create the objects using blender.

The approach used is the traditional iterative software development approach. With this I have been able to implement parts of the program and revise the documents as required. This was a more suitable approach to development as opposed to the waterfall model or an agile approach because this project had requirements which would be revisited later on in the development cycle.

The end users are thought to be students who would be learning about encryption methods used in the past, specifically learning about the Enigma machine encryption system. It is intended to give them an insight on the inner workings of how exactly the encryption process operated. The intention is to obtain students to test the program once developed. The feedback provided by the students will hopefully provide any improvements which can be made.

Initially a simulation with one rotor will be developed, then once that is functioning, three rotors will be implemented. Essentially the one rotor simulation will be a prototype.

A Gantt chart (Appendix A) has been created to outline the tasks and the expected dates to finish. I expect to finish the coding side of the program by the end of February at the latest, leaving me with March to perform testing and complete the final report. For the remaining time I intend to prepare for the demonstration show.

In addition to researching the Enigma machine, a GUI library compatible with OpenGL 3 and GLFW was required. This was required because throughout the Graphics module this was the windowing system being used and to change from using GLFW to something like GLUT, GLEW or SDL would have meant a change in the code structure, rendering the material I have learned useless. It was also important to find a library compatible with the latest OpenGL language as previous versions use deprecated code, which again was not the way I learned. Finding a compatible library turned out to be a more strenuous task that first thought. Various libraries are available but not many that I was able to get successfully working and fully understand. The ones researched were, CEGUI, libRocket, ufoLib, AntTweakBar, IMGUI, Qt and FLTK. After testing out every library I had settled for IMGUI, Immediate Mode Graphical User Interface. The library was by far the simplest to get working and integrate with the code I already had. It provides simple widgets which can be added to create a user interface. One of the drawbacks however is that it does not allow multi line text in the text box widget. However since this is an open source library maintained on Github, they may update their library to implement this in the future.