Requirements

This document shall outline the requirements, in an informal format however. The main purpose is to gather ideas of what should and must be included in the Graphical Enigma simulator.

Functional

- The system shall visually demonstrate the principle of poly-alphabetic substitution in operation in the scrambling unit of an enigma machine.
- The system shall scramble plain text into cipher text.
- The system shall unscramble cipher text into plain text.
- The system shall demonstrate the operation of encryption in one rotor.
- The system shall demonstrate the operation of decryption in one rotor.
- The system may demonstrate the operation of encryption in three rotors.
- The system may demonstrate the operation of decryption in three rotors.
- The simulation shall be demonstrated using animation.
- The system may include an attack method, which could become a game, where the user would guess the encrypted plain text.

Non Functional

- The interface shall be presented in a graphical format.
- The system shall be compatible on Windows OS.
- The system may be compatible with Mac OS and Linux OS.
- The system should be developing using C++ and the Visual Studio IDE.

Functional Requirements

Visual Representation

Description: The system shall visually demonstrate principle of poly-alphabetic substitution in operation in the scrambling unit of an enigma machine.
Rationale: This will give a clear understanding on the operation of poly-alphabetic substitution.
Priority: High

Encryption

Description: The system shall scramble plain text into cipher text.
Rationale: This function will encrypt the plain text provided by user input into cipher text.
Priority: High

Decryption

Description: The system shall unscramble cipher text into plain text.
Rationale: This function will decrypt the cipher text into the original plain text.
Priority: High

One Rotor – Encryption

Description: The system shall demonstrate the operation of encryption in one rotor.

Rationale: This will allow the user to see graphically the process of encryption in one rotor. This shall include the demonstration of current flowing in the rotor from the initially selected character from the keyboard to the cipher character on the switch board.
Priority: High

One Rotor - Decryption
Description: The system shall demonstrate the operation of decryption in one rotor.
Rationale: This will allow the user to see graphically the process of decryption in one rotor. This shall include the demonstration of current flowing in the rotor from the initially selected character from the switch board to the character on the keyboard.
Priority: High

Three Rotors - Encryption
Description: The system may demonstrate the operation of encryption in three rotors.
Rationale: This will allow the user to see graphically the process of encryption in three rotors. This shall include the demonstration of current flowing in the rotors from the initially selected character from the keyboard to the cipher character on the switch board.
Priority: Medium

Three Rotors - Decryption
Description: The system may demonstrate the operation of decryption in three rotors.
Rationale: This will allow the user to see graphically the process of decryption in three rotors. This shall include the demonstration of current flowing in the rotors from the initially selected character from the switch board to the character on the keyboard.
Priority: Medium

Animation
Description: The simulation shall be demonstrated using animation.
Rationale: As well as being graphically presented the simulation will also demonstrate the various operations of encryption using animations.
Priority: High

Attack Method
Description: The system may include an attack method, which could become a game, where the user would guess the encrypted plain text.
Rationale: A sort of game could almost be created using this simulator in which the user of the system could attempt to decrypt the cipher text manually.
Priority: Low

Non-Functional Requirements

Graphical User Interface

Description: The interface shall be presented in a graphical format.
Rationale: The system must be graphically presented to the user rather than text based as this is much more interactive and easier to use.
Priority: High

Operating System - Windows

Description: The system shall be compatible on Windows OS.
Rationale: Since majority of the users would be running a Windows operating System, the system must be compatible and fully function on Windows.
Priority: High

Operating System – Mac/Linux

Description: The system may be compatible on Mac/Linux OS.
Rationale: A minority of the users may use a mac or Linux operating system therefore it would be ideal to cater for their needs thus the system may be compatible with these operating systems.
Priority: Medium

Development

Description: The system should be developing using C++ and the Visual Studio IDE.
Rationale: In order to promptly start development, a familiar programming language and environment should be used.
Priority: High