Cracking The Enigma, version 1.

# **Description**

Version 1 of this project allows a user to set-up and operate on a working Enigma machine (for a detailed explanation on the machine, please refer to the following link: <https://www.youtube.com/watch?v=ybkkiGtJmkM>), as a console application.

A user must load a scheme-based XML file describing an existing machine and all of its parts (there are several working XML files included in the project). The user is then allowed to set up the machine in any way that he wishes, either randomly or manually and continue to operate on the machine: encrypt and decrypt messages, reset the machine to its original state, view encryption history and save and load a set-up machine along with its history.

This version of the project is written in Java only and relies on Java 8 in Intelij idea.

# **How to run ‘Cracking the Enigma Ver 1’**

To run the application there are two options:

1. Folder ‘Executable’ includes a .BAT file with the relevant .jar files to the project. To run the application, just double-click the RunCTE.bat file.
2. Open the CTE-Code folder using the Intelij idea. Build the project and run it from class ‘ProgramMain’ located in module CTEConsoleUI in the package ‘ui’.

# **How to use ‘Cracking the Enigma Ver 1’**

All the actions a user can perform are visible to him but are not necessarily operational. In order to operate on a machine follow these steps:

1. Load a new machine. You can either load a machine from an XML file – action 1 (which only loads the machine but does not make it operable yet), or load an existing machine (which loads it set-up for use) with action 9.

* If you’ve loaded a machine with action 9, you can instantly being to use it and skip step 2.

1. Set the machine for use. You can do that either manually, choosing the rotors, reflectors, etc. Using action 3, or let the program randomly set it up for you using action 4.

After setting up the machine, you are able to encrypt, decrypt messages, show encryption history and reset the machine to original state.