PROJECT FINAL REPORT

PROJECT NAME:

Ænigma

PROJECT AUTHOR:

Dan Cassidy

PROJECT PURPOSE:

This application will be able to use the Enigma cipher, developed by the Germans in the early 1920's, to encode and decode local messages. The goal is to be able to mimic the machine settings perfectly, including different numbers & types of rotors and a different reflector.

PROJECT FUNCTIONALITY:

First off, the application is not 100% functional. Because I ran out of time, there are certain features that are not working, comments are relatively scarce in the code, and there are A LOT of todo's.

Due to issues replicating the functionality of certain types of Enigma machines, the app currently only supports the Enigma machines used by the military, namely the I, M3, and M4 models. Everything relating to those models works well.

The current version of the application does not save the user's changes to the machine settings, though it does show what the current ones are. Additionally, the "Help" and "About" menu functionalities are not enabled.

KNOWN ISSUES:

- Changes to the machine settings do not save.
- Available reflectors and rotors do not update when changing the machine type.
- Appearance of freezing when pasting a large amount of text into the input field.
- Sluggishness when transitioning to the machine settings activity.
- Code is for the most part only lightly commented.
- MANY todo's in the code.
- "Help" and "About" menu functionality is currently disabled.

MAJOR TECHNIQUES USED:

The "most major" technique I used was the MVP pattern of software architecture. This was a significant undertaking specifically because my model, the Enigma machine, is quite complex, and not giving in to the temptation to simply have the activities talk directly to the model was very difficult.

I chose not to use fragments as they would have added another layer of complexity on top of my app, but in order to shuttle data, specifically the presenter, from activity to activity I had to implement the Parcelable interface.

I also utilized the menu to handle actions such as pasting text to the input, copying text from the output, and clearing the input and output boxes.

Difficulties Encountered:

I encountered two major difficulties when creating this app. The first was, as I mentioned in the Project Functionality section, getting the program to mimic the commercial models of the Enigma machine. Ultimately I was unsuccessful at this and had to pare down the app to only support the military models, but not before sinking many, many hours into that aspect of the app, in an ultimately wasted effort.

The second major difficulty was simply the time required. I grossly underestimated the time needed to handle the UI of the app, specifically in dealing with the machine settings, and as a result ran short on time and was unable to comment things anywhere near to my satisfaction, or get the settings fully working. (At this moment, you can view all of the settings, but changing the UI elements representing them does nothing.)

Another significant obstacle in the development of this app was figuring out how to pass objects from one activity to another. I used the Parcelable interface for this, and ran into some pretty big headaches, but was ultimately able to finish it. Of the obstacles I was able to overcome, this one was probably the worst.

And the last difficulty, but definitely not least, was dealing with the user interface. While Android is relatively mature as compared to just a few years ago, it is still all over the place in some areas, and it shows. Because it is ever-changing, there are a great many things that simply do not work in a logical manner, or have crazy interactions with other settings, or simply don't work at all when certain attributes are set, or sometimes not set!

Test Run Video:

Note: I have preset the app to some random settings to demonstrate encoding, as well as the bugs of machine settings not saving and available reflectors and rotors not updating when machine type is changed.

http://youtu.be/H_pwHqI-UkU