Name: Thinh T. Diep
Class: COSC 2500 - Intro to Computer Systems and Platform Technologies
Professor: Anna-Lyza Felipe Sanchos & Long Nguyen Minh
Assignment 1: MicroBit
Date: 16/11/2019

Reflection

Before enrolling in this course and starting this project, I already have some knowledge in programming, and I decided to program in JavaScript, a language that is actually somewhat new to me, but one that I am learning at the moment in another class, Web Programming.

The first version of this project was just a simple Morse Code Translator. A user can input “dot” and “dash” in Morse with the Micro:bit buttons, shake it and a corresponding character would appear. It accomplish this by comparing (filter() function) the input string from the user with a hard-coded Morse Code dictionary array. I originally plan for this to be my complete project.

The second version of this project implements the transmission and encryption process. The project that I saw on the Micro:bit website that was the inspiration for my project actually consists of two Micro:bits communicating with each other, so I wanted to do something similar. What I have done with my own project resembles texting or a messaging app.

Encryption is the last part of this project. As we have learned in class, messages sent online are encrypted during the transmission process and only displayed at both ends, so again I wanted to do something similar. I chose to allow user to freely enter keys for the board, and messages sent between these boards would only display correctly if they have the same key. I did a little bit of research on cryptography, but I decided that having a robust and secure encryption is not the main point of this assignment, so I implemented a homebrewed algorithm. The algorithm is a form of Caesar encryption where it exchanges each character with some other character. During the process the encryption key is converted back and forth between decimal and binary numbers. The algorithm allows for non-encryption if the encryption key is set to empty.

There are many things that can be improved in this project should I choose to develop it further. Firstly, the simple hardware of a Micro:bit is not the best fit for typing characters and numbers, I would use a Raspberry Pi for further development. Secondly, this version of the program requires the two users to know the encryption key beforehand and enter it manually. It would be better that the two connected Micro:bit can establish a key between themselves so that they can automatically authenticate each other but not let other unauthorized devices see the messages (much like how real applications work). Thirdly, many functions, like the conversion functions, are not optimized and can be redesigned to be better. Fourth and lastly, it is well known that a homebrewed encryption algorithm is almost always a bad design and can be cracked by a determined hacker with resources. In a real application (messaging app), this part would be the most important aspect, and this requires the author (namely me) to be much more proficient in cryptography.