Dillon Butler (dillonb@vt.edu)
Sterling Davis (sdsilver@vt.edu)
Chris Shute (cshute@vt.edu)
3/30/19
ECE 4564
Assignment 2 Report

Design:

The "Wishing Well" project functions using two Raspberry Pi 3 B+s. One Raspberry Pi is referred to as the Capture RPi and the other is referred to as the Repository RPi. The Capture RPi first receives a tweet from Twitter containing the hashtag ECE4674T10. These tweets contain a command, place, subject, and message. The command portion indicates whether the the Capture RPi should produce a new message or consume them from the database. If it is a produce command, the Capture RPi stores the place, subject, and message in a Mongo Database. It then communicates via RabbitMQ with the Repository RPi so that the Repository RPi outputs the message on the screen. If the command portion of the tweet is a consume command, then the Repository RPi will communicate with the Capture Pi so that it will receive all messages that fit the place and subject specified in the tweet and output them on the screen. Meanwhile, if the system is waiting for a command, an LED will light up white. If it has received a produce command, it will light up red. If it has received a consume command, it lights up green. The twitter stream is constantly running and receiving new tweets, so the system will work for multiple tweets.

Outcome:

The outcome of the project was as expected. The "Wishing Well" met all requirements set by the assignment document. The Capture RPi receives a tweet and parses it, stores the information in a MongoDB, lights up an LED accordingly and finally sends the information via RabbitMQ to the Repository Pi. The Repository Pi then outputs a message based on what the command was and what the specific message was. RabbitMQ proved to be the only hiccup in designing the code, but eventually we got it working. Overall the team learned about new APIs, RabbitMQ, and MongoDB.