Mark Agluba

Jeremy Tang

Antonio Simonelli

**Final Documentation**

1.

a. The project consists of three clients getting data from physical hardware (breadboard and the wiring) and showing it to a user interface (JavaFX) as well as simultaneously sending it to a broker. The clients can send data to the broker by publishing their messages to their own topic and making sure that the message is secured(keystore). Other clients could subscribe to other's topic to receive their data, but before getting the data, there will be a verification method. The result is that all three clients will simultaneously see their data in a graphical user interface with one screen.

b.

* . Upon launching the application, it will prompt a login for the MQTT
* . After the MQTT login, it will prompt another login for the keystore
* There will be a total of 15 tiles after logging in. Five for the client that logged in, and 5 others to the 2 clients
* . It will be a couple of seconds, but the tiles for other clients will start to fill.
* Users will send out their keys and also receive from others and save.
* Data sent is signed and therefore data received needs to be verified using the corresponding key
* After data Is verified, it is used to update the dashboard
* . There is an exit button to end the application

c. We decided to prompt the user first for the login to make sure that only people that subscribed to the broker will be able to see their data as well as the data from others that subscribed.

We made tiles for each data because it is a bit boring to just look at text, so it was better to present the data in some type of way.

d. We did a couple of unit tests for the project but not all of them were done, because most of the methods were related to JavaFX and MQTT, we didn’t have any idea on how to test it. We had a problem with one of RaspberryPi’s NetBeans where it wouldn’t be able to run any of the threads that was being processed in the tiles page, so whenever it would try to load the tiles page it would not run any processes that was put in a thread, and on one of the teammates vscode would not be able to run the code at all. There are some errors when exiting the application since threads are still running, but it doesn’t hinder the application itself. Program only accepts absolute paths which is a bit annoying.