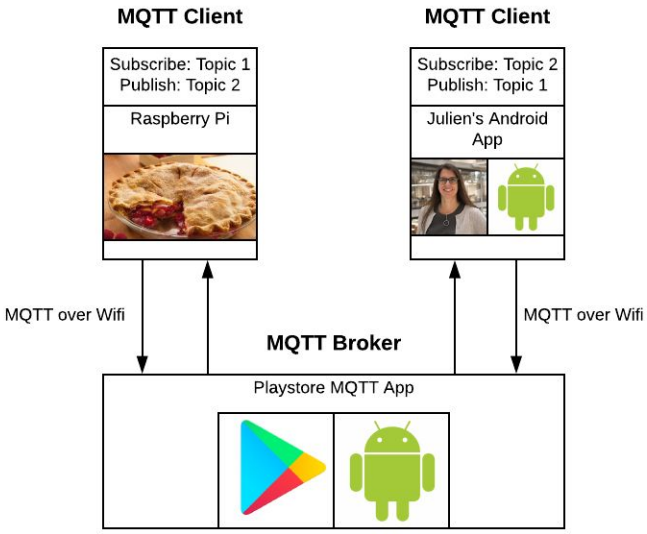
Brian Peterson
Abraham George

Milestone - 3

Step - 6

Architecture



On Julien's app we publish our message on Topic 1.

This is sent to the MQTT broker on the Playstore app.

The broker distributes the message to the clients that are subscribed to Topic 1.

The raspberry pi receives this message, and publishes a response message to Topic 2.

The MQTT broker sees the incoming message and distributes it back to the Julien's Android app because the app is subscribed to Topic 2.

Step - 8

It is better to run the MQTT broker on the Raspberry Pi rather than the mobile phone for the following reasons:

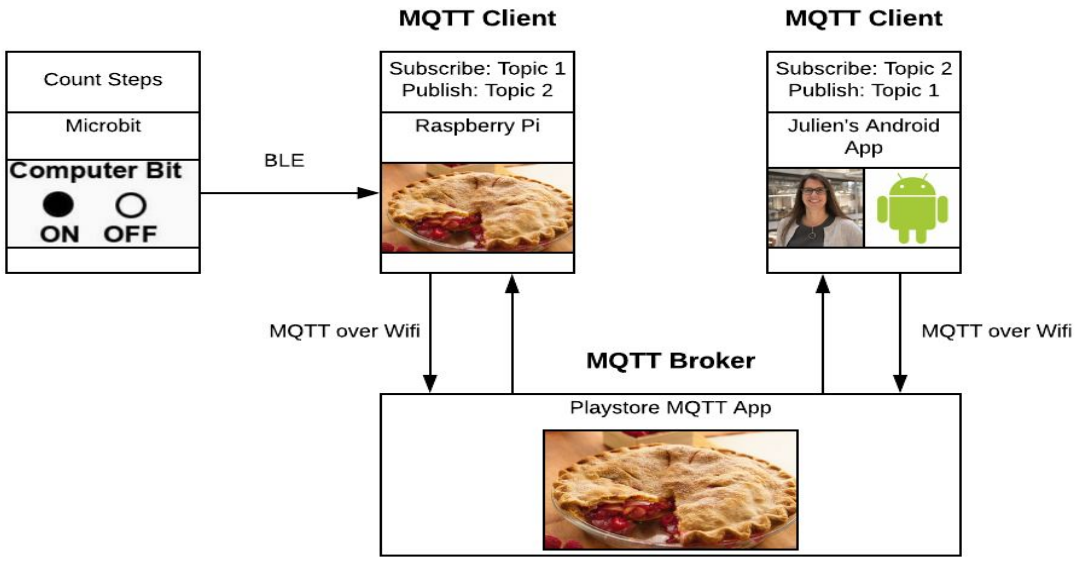
- Running the broker on the mobile is a huge battery drain
- The MQTT client needs to be on the same network as the MQTT broker. The raspberry pi can only communicate on it's local network because of the UT network restrictions so our broker must use that network. If this is the case, we couldn't use our phones for internet, which would be very inconvenient.

Step - 9

- Updated the Kotlin code to pick up the address of the MQTT broker running on the pi

- Created a python file running on the pi which would listen for the eddystone beacons sent out by the microbit via BLE. This gave us the step count. The code would then publish the step count to the MQTT broker running on the pi.
- The android app is subscribed to receive the step counts from the broker. Once the step count is received by the Android app, the app displays the count to the UI.

Updated Architecture



Sequence Diagram