**SCHOOL OF COMPUTING (SOC)**

|  |  |
| --- | --- |
| **Date of Submission:** | 18th August 2019 |

|  |  |
| --- | --- |
| **Prepared for:** | Ms Dora chua |

|  |  |
| --- | --- |
| **Class:** | DISM 3A/02 |

|  |  |
| --- | --- |
| **Submitted by:** |  |

|  |  |
| --- | --- |
| **Student ID** | **Name** |
| 1637876 | Augustus Tan |
| 1748831 | Leong Shi Xian |

**IOT CA2**

**Step-by-step Tutorial**

**DIPLOMA IN BUSINESS INFORMATION TECHNOLOGY**

**DIPLOMA IN INFORMATION TECHNOLOGY**

**DIPLOMA IN INFOCOMM SECURITY MANAGEMENT**

**ST0324 Internet of Things (IOT)**

**2017/2018 Semester 1**

**Table of Contents**

[Section 1 Overview of project 2](#_Toc16877547)

[A. Where we have uploaded our tutorial 2](#_Toc16877548)

[B. What is the application about? 2](#_Toc16877549)

[C. How does the final RPI set-up looks like? 2](#_Toc16877550)

[D. How does the web or mobile application look like? 2](#_Toc16877551)

[E. Evidence that we have met basic requirements 3](#_Toc16877552)

[F. Bonus features on top of basic requirements 4](#_Toc16877553)

[B. Quick-start guide (Readme first) 4](#_Toc16877554)

[Section 2 Hardware requirements 5](#_Toc16877555)

[Hardware checklist 5](#_Toc16877556)

# Section 1 Overview of project

* 1. Where we have uploaded our tutorial

A video created to show how our application works

Youtube Demo : https://youtu.be/nbB8fvW2Kz4

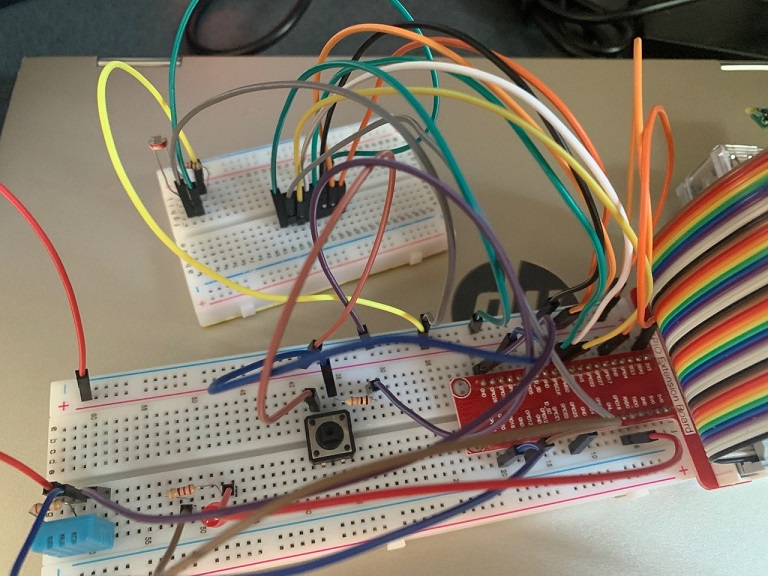
All instructions and documentation is on

Github : https://github.com/palwolus/HeyPi

* 1. What is the application about?

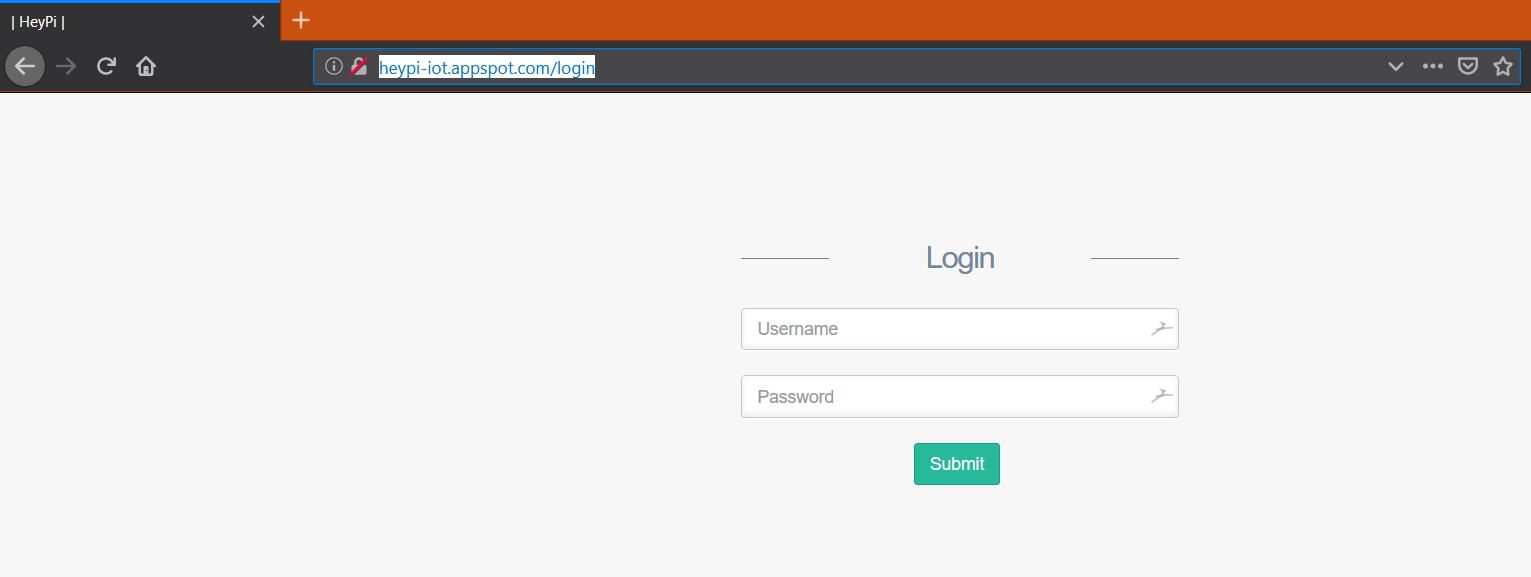
Heypi is a project done as an experiment which has a voice chat system that utilizes cloud services to carry out its functions. With the implementation of IOT products such as DHT and light sensors Heypi is able to offer more than just a typical voice chat application. Real time reports of the temperature and light can be seen, in future implementation of heypi we can make both end users of the application understand the other client's environment more.

* 1. How does the final RPI set-up looks like?

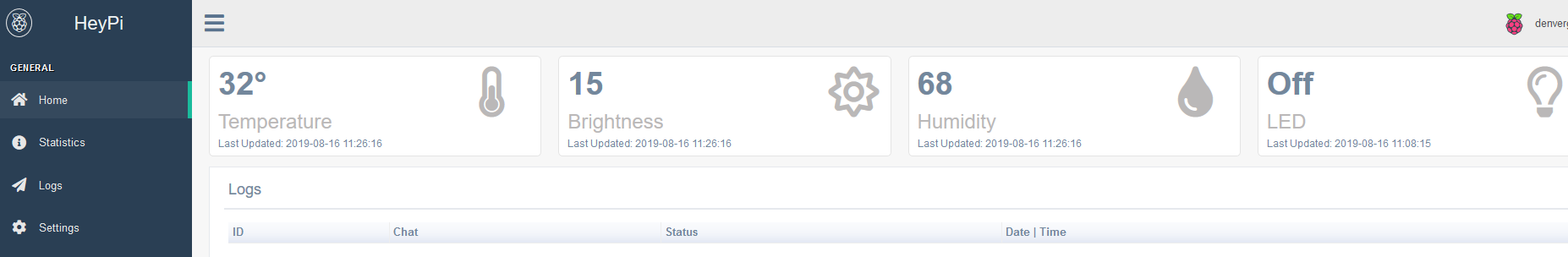


* 1. How does the web or mobile application look like?

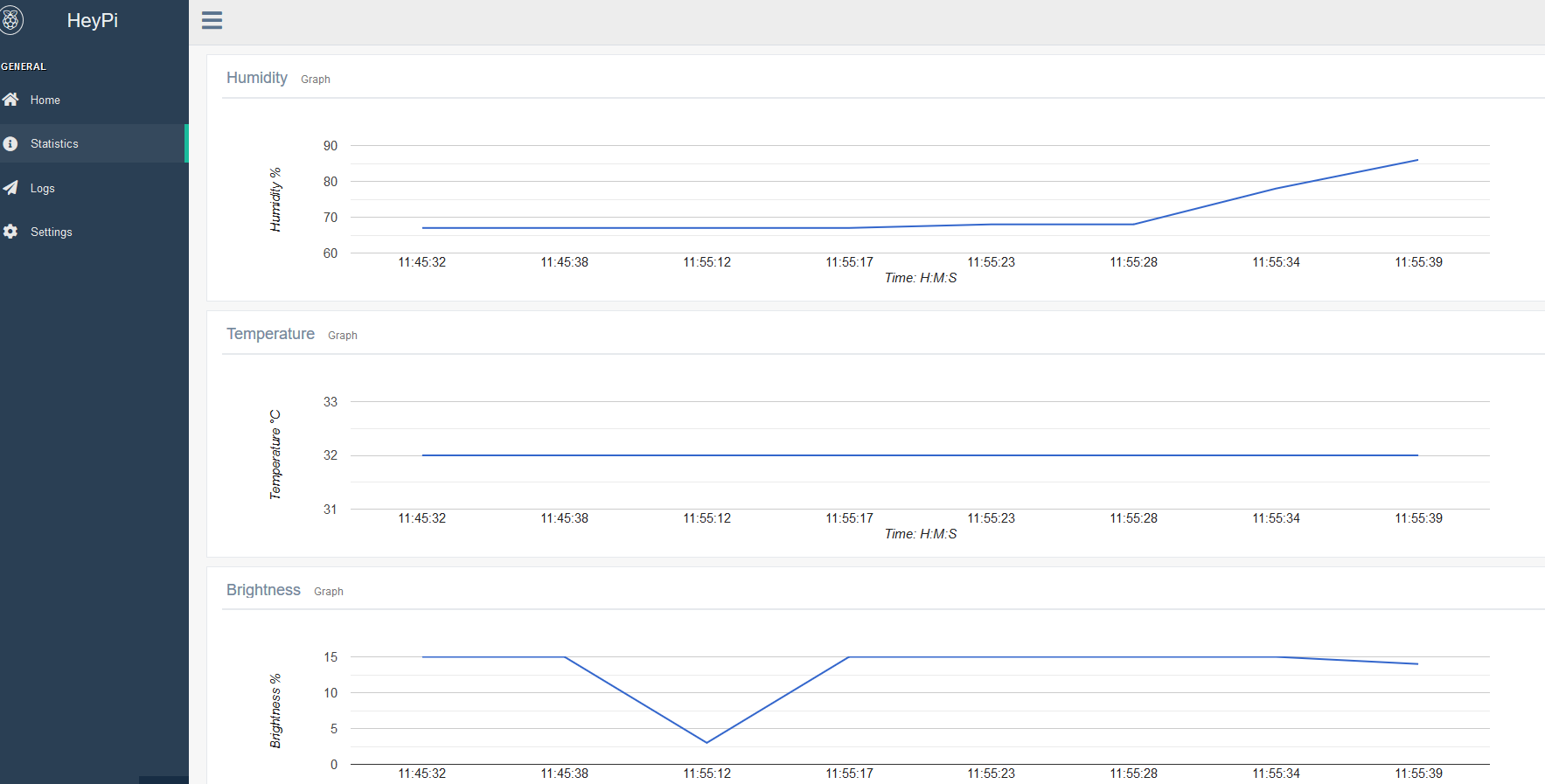
Web main login page:



Web main Dashboard:



Real time graphs on web page:



* 1. Evidence that we have met basic requirements

Provide bullet list to describe how your group has met basic requirements

|  |  |
| --- | --- |
| Requirement | Evidence |
| Used three sensors | Used DHT temeprature and humidity sensor, Light intensity sensor. Also used LED as an actuator. |
| Used MQTT | Our raspi sends MQTT data to the cloud which is then processed and shown on the dashboard, further more the web is able to publish as well turning on the LED on the raspi |
| Stored data in cloud | Stored light, temperature and humidity into a database in google cloud |
| Used cloud service | Serverless Function on Google Cloud  Web Interface is hosted on Google Cloud  Uploading of Files & Downloading from Google Cloud Storage Bucket |
| Provide real-time sensor value / status | Show the real-time value of Temperature, light and Humidity sensor |
| Provide historical sensor value/ status | Show historical vlaue of light, temperature and humidity sensor |
| Control actuator | Placed button on webpage to control a led |

* 1. Bonus features on top of basic requirements

Provide bullet list of the bonus features you have added on top of basic requirements

- All Cloud services is hosted on google cloud which is not taught in class

- One Click Run Script

- Serverless Function on Google Cloud

- Cloud Functions

- Google PubSub

- Web Interface is hosted on Google Cloud

- App Engine

- Uploading of Files & Downloading from Google Cloud

- Google Cloud Storage Bucket

- Audio Input & Output to/from Raspberry Pi

- Asynchronous voice commands to control LED, get sensors value and take picture

- Pictures are saved at (script location)/picture

- Audio Input & Output is highly customisable. GPIO Pins are customisable too

- Password is hashed for security

- A login system

* 1. Quick-start guide (Readme first)

Give a few lines of basic instructions on how I need to run your app, e.g

1. First connect hardware as shown in fritzing diagram
2. Install and upgrade required modules in the raspi as listed in github
3. Run pi.py with arguments -u <username> -p <password>
4. Log into website at <http://heypi-iot.appspot.com/login> to access the dashboard

# Section 2 Hardware requirements

Hardware checklist

The following hardware is required for the program to work flawlessly

- 1 x DHT 11 Sensor

- 1 x LED

- 1 x MCP3008 ADC

- 1 x Light sensitive resistor

- 1 x button

- 2 x Resistor 10k ohms

- 2 x Resistor 330 ohms

- 22 x wires

For the record and listen function the following hardware must be present

- 1 x USB mic (Recording)

- 1x 3.5mm earphones (Listening)

**-- End of CA2 Step-by-step tutorial --**