Project:

Automated Mushroom Cultivation System With Raspberry Pi

Advanced Operating System[A]

M. ARIFUR RAHMAN

Team Member

DESH, SALMAN RAHMAN
ID: 13-23239-1

ALAM MD.RIFAT
ID: 13-24523-2

INDEX

INTRODUCTION

Problem domain & Solution

This project is raspberry pi based mushroom cultivation automated system. For mushroom cultivation temperature and humidity are sensitive factors, so these need to be monitored frequently to maintain proper balance of humidity and temperature. Which is very tough for human but need to resolve the issue by any how. So we decided to build an automated system to maintain this balancing of humidity and temperature based on some logical conditions and fires some actions. Like when the humidity is increased and cross its limit then ventilators are open and water spray is open to spray on mushroom. And when the temperature is decreased and it feels cold the ventilators are close and heater starts.

To complete this project we have used the following equipment:

1. Raspberry pi model 2 B
2. SD memory card
3. Ethernet Cable
4. USB Cable to power Pi
5. DHT22 sensor
6. 4.7 K Ohm resistor
7. 4 - switch Relay module
8. Some male-male, male-female and female-female Wire
9. PC/Laptop to view and control Raspberry Pi

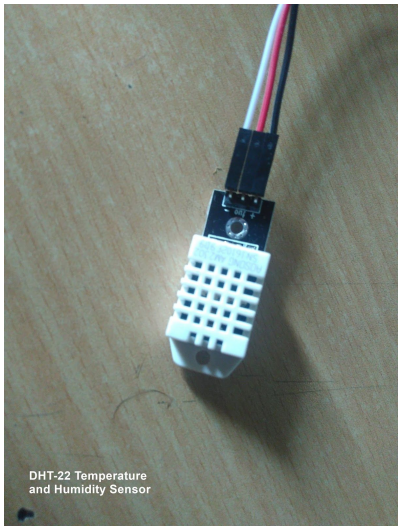

Fig: DHT-22 sensor

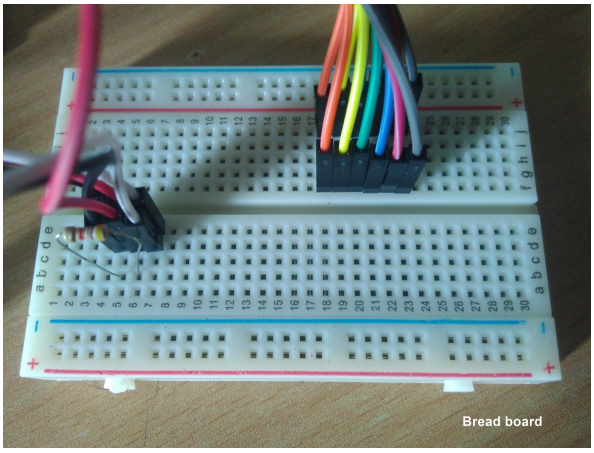

Fig: Breadboard

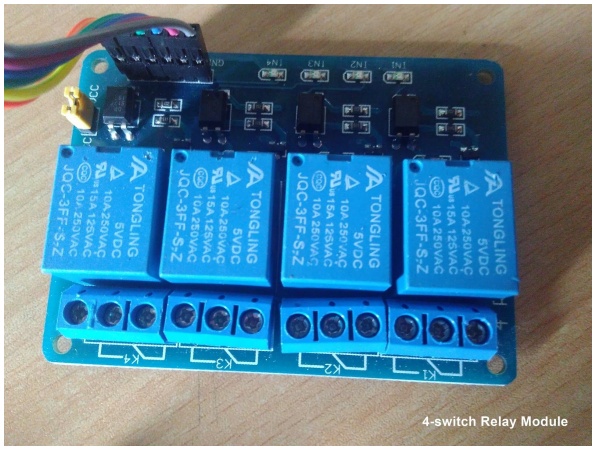

Fig: 4-switch Relay module

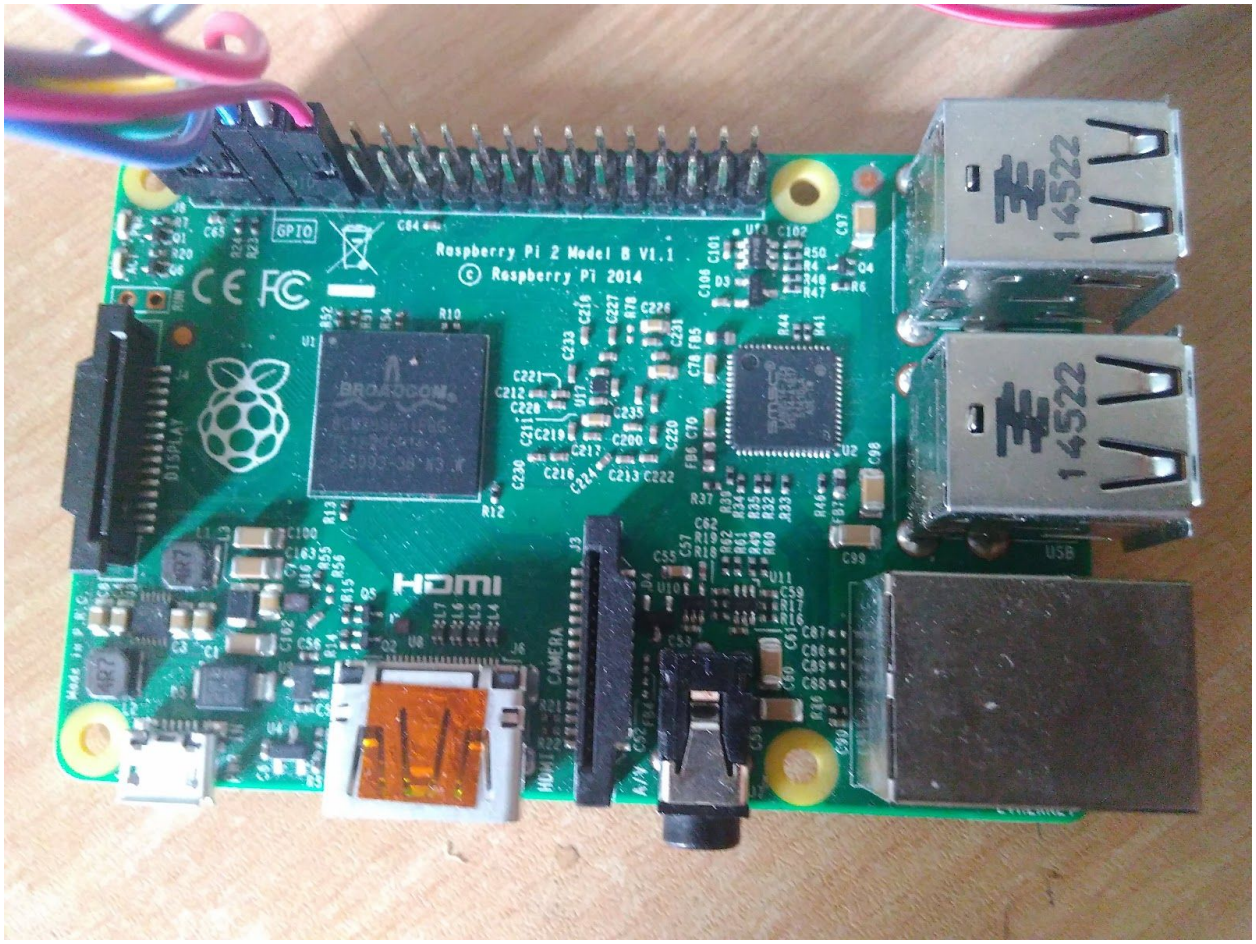

Fig: Raspberry Pi Model - 2 B

DETAILS

Step 1: OS for the Raspberry Pi

We have installed **Raspbian OS** to run on Raspberry Pi . First extract the OS files into pendrive by **win32diskimager** and Set SD to Raspberry Pi and powered it up. Raspberry Pi started to boot up. We created a file named **"ssh"** without any extension into bootable memory card to enable ssh into Raspberry pi.

Step 2: Connect Raspbery Pi to PC

Raspberry Pi is now configured and booted up and we should be able to connect it via mobaXterm software from our PC now. Before that we need to make our internet connect to allow sharing.

For windows click: Network and sharing center > Network connections > Wifi/Ethernet > Properties > Sharing > Allow both option and click OK.

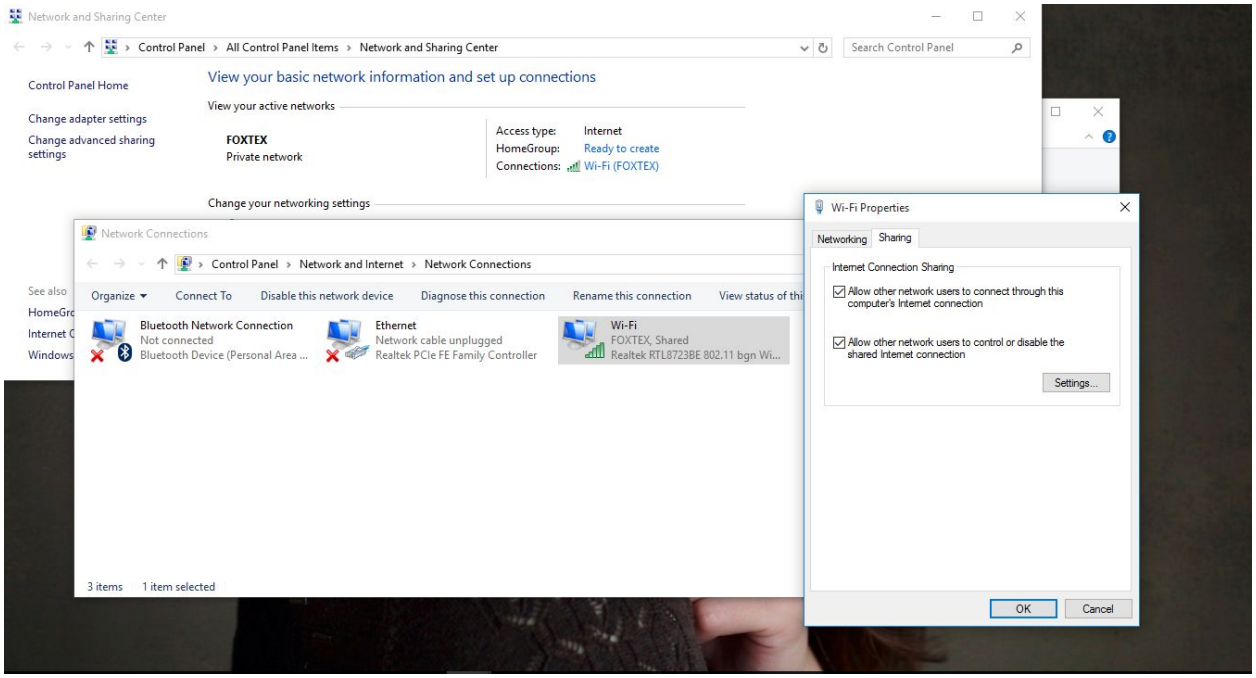

Fig: Internet connection sharing

To execute a connection in mobaXterm, hostname or IP is required in order to connect to Raspberry PI. To obtain the Raspberry Pi's IP address we used ipscan24 software. Then need to write on mobaXterm commandline **"startlxde"** to display raspberry pi.

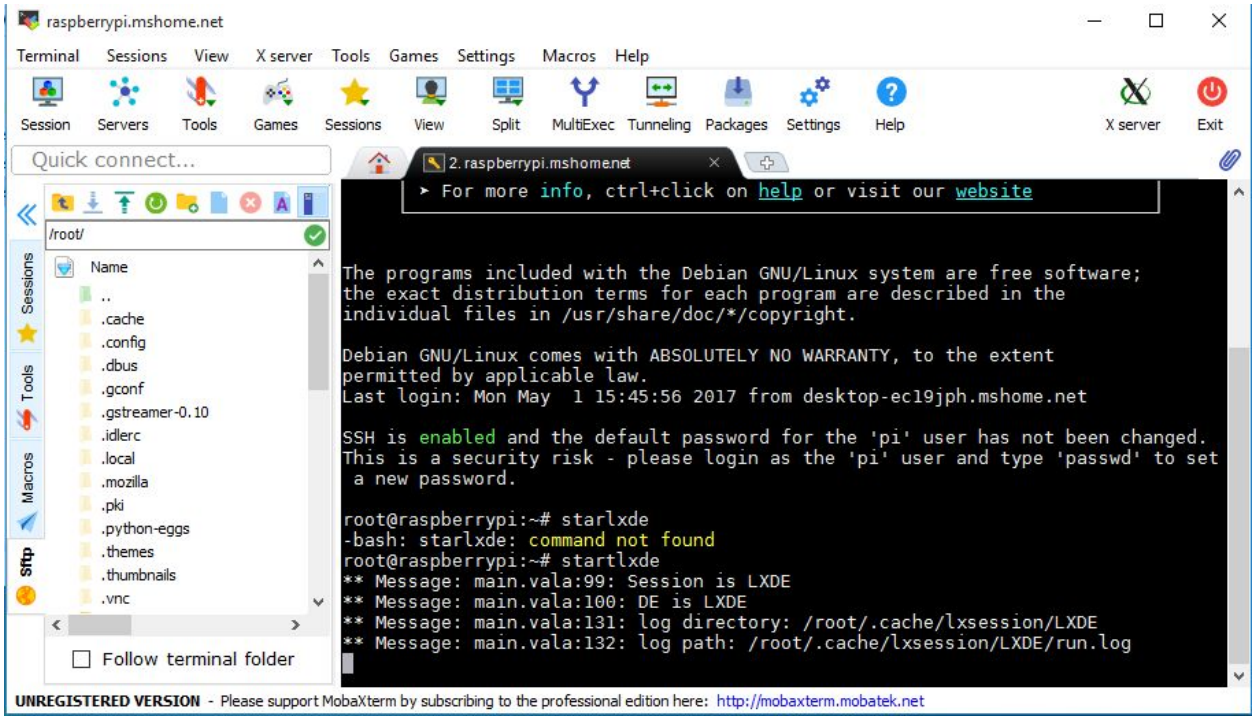

Fig: mobaXterm

Step 3: Connecting DHT-22

1. Make sure Raspberry Pi power is switched off.
2. Breadboard is used for the following connection.
3. "+" (VCC) pin to Raspberry Pi GPIO +3.3 V pin.
4. "middle" DATA pin to the GPIO pin.
5. 4.7 K-Ohm resistor is set up to "+" power pin and DATA pin
6. "-" (GND) pin to GPIO Ground pin.

Step 4: Connecting Relay Module

1. Make sure Raspberry Pi power is switched off.
2. Breadboard is used for the following connection.
3. VCC pin to Raspberry Pi GPIO 5V pin.
4. Rest of the four pins to GPIO pin.
5. GND pin to GPIO GND pin
6. Power on the Raspberry Pi

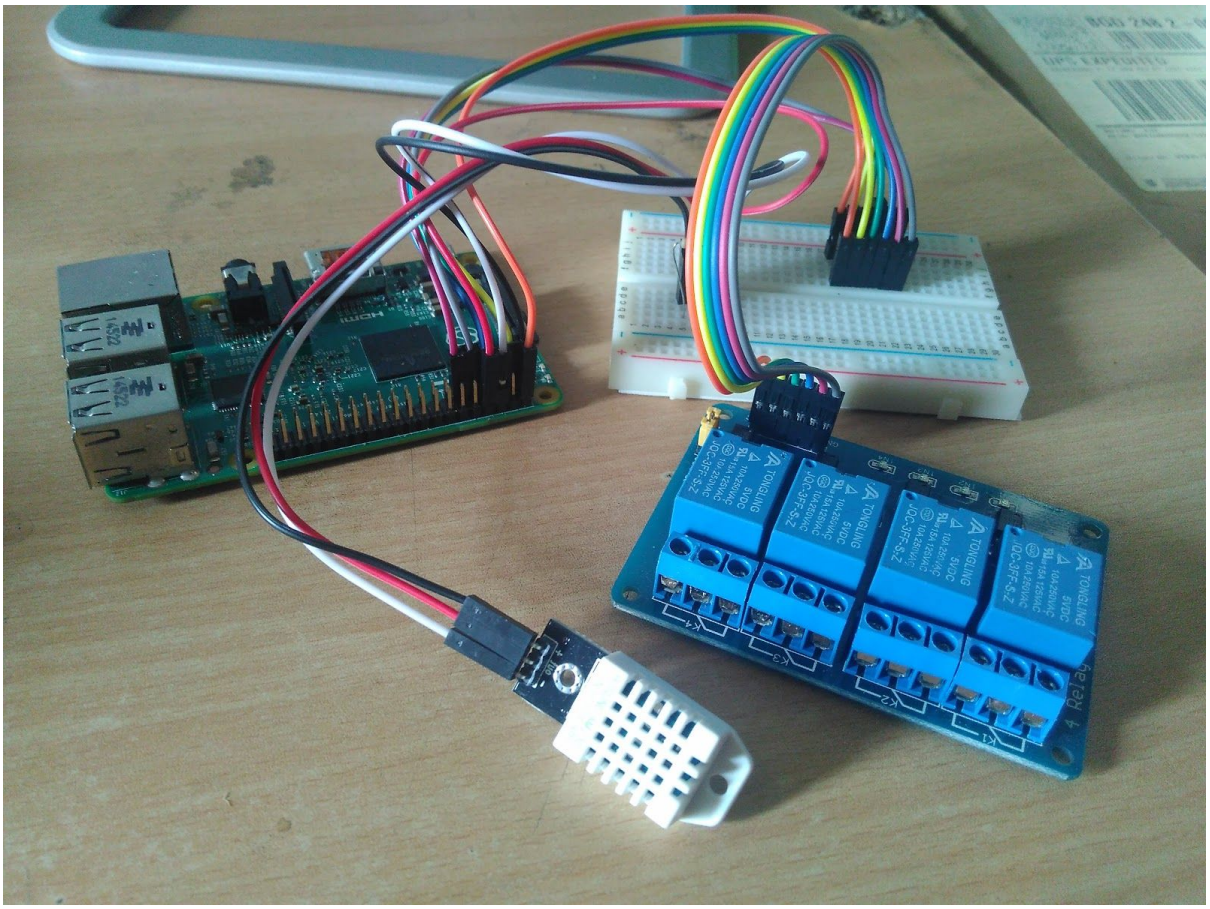

Fig: Module pin connections with Raspberry Pi GPIO pin

Step 4: Install Libraries and Packages

PIGPIO library install

sudo apt-get update

sudo apt-get upgrade

rm pigpio.zip

sudo rm -rf PIGPIO

wget abyz.co.uk/rpi/pigpio/pigpio.zip

unzip pigpio.zip

cd PIGPIO

make

sudo make install

sudo pigpiod

Downloaded "DHT22 AM2302 Sensor" from

http://abyz.co.uk/rpi/pigpio/examples.html#Python%20code

Unzip and pasted into the folder where main script will be written.

PHP5 install

sudo apt-get insall php5

MySQL install

sudo apt-get install mysql-server

Note: **set password** for mysql

PyMySQL install

sudo apt-get install pymysql

Phpmyadmin install

sudo apt-get install phpmyadmin

Note: **set password** for phpmyadmin and choose webserver **"apache"**

Apache2 server install

sudo apt-get install apache2

sudo nano /etc/apache2/apache2.conf

At the bottom write “Include /etc/phpmyadmin/apache.conf”. Save and Exit by CTRL+X

sudo /etc/init.d/apache2 restart

hostname -I (to get apache server IP)

Step 5: Write the python file

We create a new python script file and write all custom code on it. Here we check all the logic that if heat increased then the ventilators open and start spray water. When decreased then ventilators close and heater started. The timer function is here to count the time for all actions.
Create a new database and table in phpmyadmin from **“localhost/phpmyadmin”** or “raspberrypi/phpmyadmin” or “apache server IP/phpmyadmin”. Getting data from DHT22 sensor which are humidity and temperature in float value and inserted all those data to phpmyadmin database through **PyMySQL** from python script. We also **imported timer** library to get the current time and date for each action being executed and also inserted time and date value with action status to phpmyadmin database table respectively. **DHT22.py** file that we have downloaded from “http://abyz.co.uk/rpi/pigpio/examples.html#Python%20code” are need to be in the same folder to get data from DHT22 sensor. Now moved the entire folder to **“/var/www/html”** to host our project and to work with PHP.

Step 6: Web Look

We have created a php file to fetch all data by **MySQL**, that we inserted to phpmyadmin database. To view humidity and temperature data real time on web, we used **Jquery AJAX**. Also created another php file to show the history of all data that already stored to database according to time, data, humidity, temperature and action status. Overall **web design** and **responsiveness** has been done with **Bootstrap-3**.

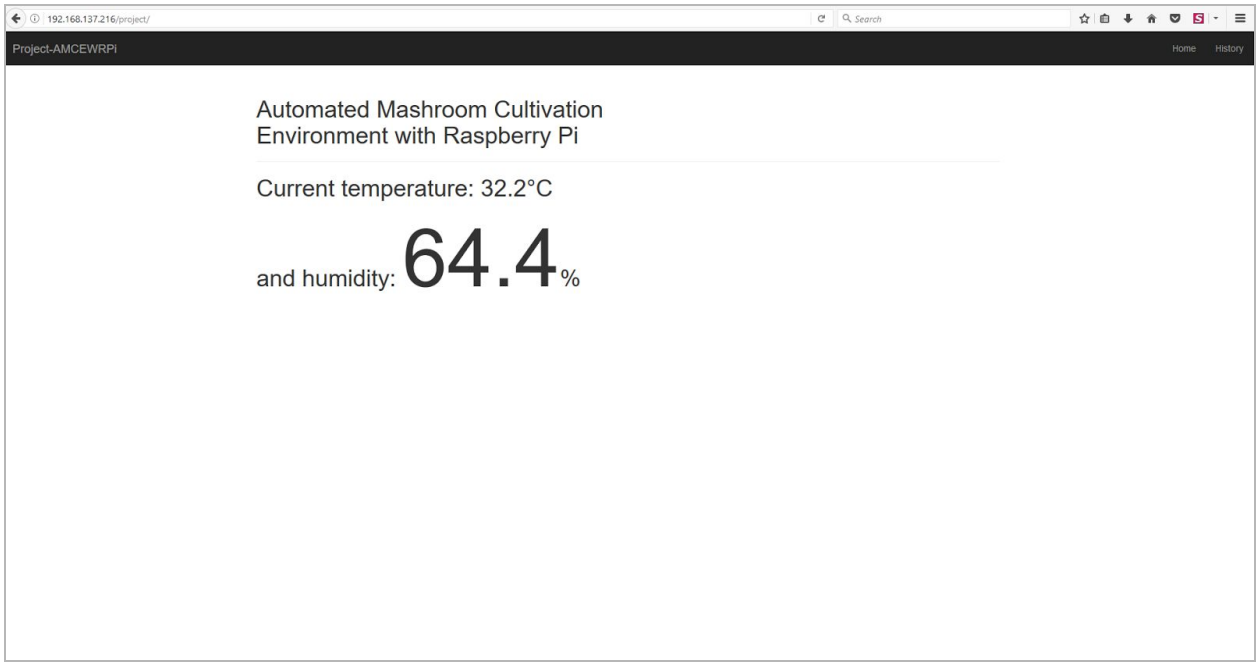

Fig: webview

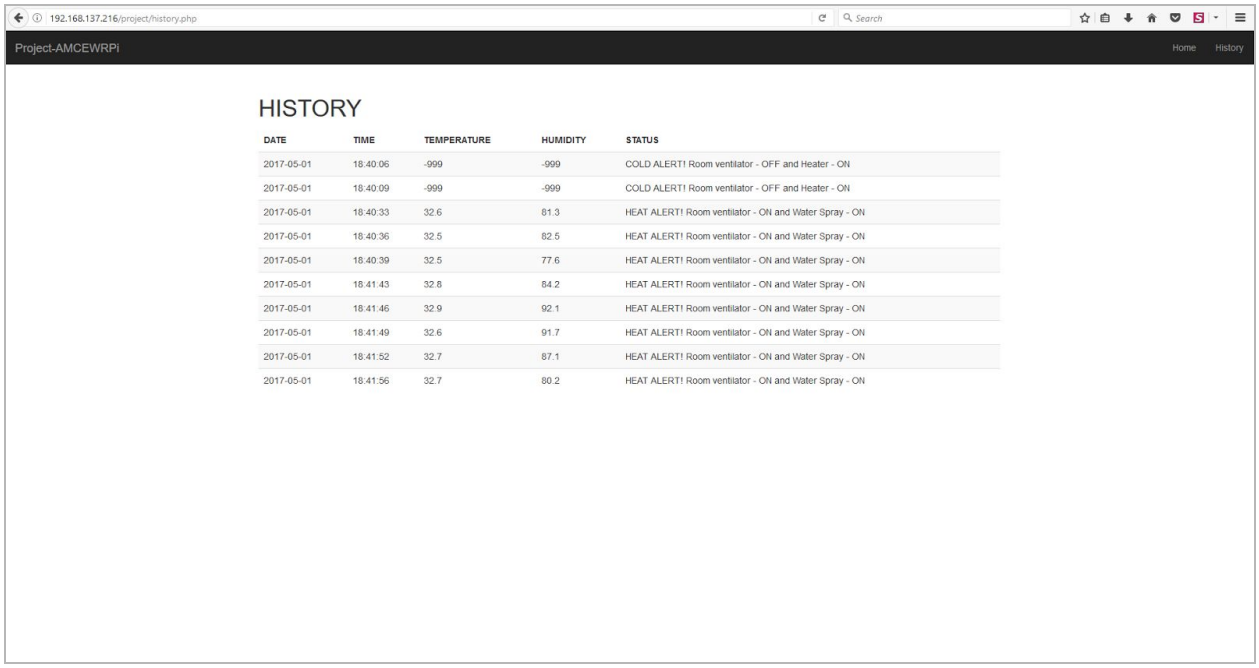

Fig: webview-2