****

**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING**

**FINAL REPORT**

**TITLE:**

Smart Kitchen Automation Using IoT

**TEAM MEMBERS:**

|  |  |
| --- | --- |
| **Register Numbers** | **Names** |
| 16mis1045 | Keerthna Kayatri.N |
| 16mis1060 | Madhumitha.VJ |
| 16mis1100 | Harini.G |

# Objective:

The main objective of kitchen automation and security is to help handicapped, aged people and working women’s that will enable them to control kitchen wares and alert them in critical situations and will help them in doing things easily and efficiently.

The overall purpose of the project will encompass

* Saving time
* Accuracy in tracking the stock
* Personalised kitchenware

# Scope:

# Our project is an initiative of kitchen automation. Here we will notify the owner with the automatic grocery list generated by comparing the history of grocery and doing analytics in the THINKSPEAK cloud.

# 

# The communication between these physical objects are done using a NFC tag. This tag needs to be scanned by the owner in order to give him/her the grocery information. Analytics is done in the cloud to create grocery list periodically or even daily.

**ROLE PLAYED BY EACH STUDENT:**

|  |  |  |
| --- | --- | --- |
| **Contributing member** | **Role played** | **Topic** |
| Keerthna Kayatri.N | Installation of Microcontroller IDE, IFTTT coding. | Arduino installation, IFTTT |
| Madhumitha.VJ | Hardware integration to cloud | ThinkSpeak and IoT integration |
| Harini.G | IoT Hardware and cloud coding | Hardware assembling, ThinkSpeak integration |

**DETAILS OF THE TIME SPENT ON EACH MODULE:**

|  |  |
| --- | --- |
| **Modules** | **Time spent** |
| IoT hardware | 2 weeks |
| ThingSpeak cloud service | 10 days |
| IFTTT application | 5 days |

**STEP-BY-STEP PROCEDURE:**

**HARDWARE REQUIREMETS:**

1. NodeMCU ESP8266

The easiest way to describe the NodeMCU ESP8266 is that it is an incredibly cheap "Arduino-like" development board which has WiFi built in.

1. Load Cell

This is simply a bar of aluminium with two holes drilled at the centre - when supported at one end and with a load applied at the other an analogue reading is produced.

1. HX711 Board

This is an amplifier which converts the analogue reading produced by the load cell into a digital one readable by the NodeMCU ESP8266.

**HARDWARE SETUP:**

Connect an external Load Cell amplifier to the load cells available in the scale using the HX711 load cell amplifier module.

The load cells the scale came with have three terminals and are connected in a bridge. The HX711 load cell amplifier has two input channels, and the bridge can be straightaway connected to the S+, S-, E+, and E- pins of the module by using only one channel.

The scale Image of Loadcell Bridge has a PCB that has marked the above-said pins making the connections easy.

To interface with microcontroller HX711 comes with a serial interface having Data and Clock pins. It comes with 24-bit resolution. To interact with the amplifier using NodeMCU, we used the HX711 library by bogde with a NodeMCU. NodeMCU is used to calculate the calibration factor to fetch weight from the amplifier.

**SOFTWARE REQUIREMETS:**

**INSTALLING THE ARDUINO IDE**

1. Visit <http://www.arduino.cc/en/main/software> to download the latest Arduino IDE version for your computer’s operating system. There are versions for Windows, Mac, and Linux systems. At the download page, click on the “Windows Installer” option for the easiest installation.

2. Save the .exe file to your hard drive.

3. Open the .exe file.

4. Click the button to agree to the licensing agreement

5. Decide which components to install, then click “Next”

6. Select which folder to install the program to, then click “Install”

7. Wait for the program to finish installing, then click “Close”

8. Now find the Arduino shortcut on your Desktop and click on it. The IDE will open up and you’ll see the code editor

**CONFIGURING THE ARDUINO IDE**

The next thing to do is to make sure the software is set up for your particular Arduino board. Go to the “Tools” drop-down menu, and find “Board”. Another menu will appear, where you can select from a list of Arduino models.

**THINKSPEAK CONFIGURATION**

Once the hardware is set up, we can go ahead and create our ThingSpeak Channel.

Step 1: Go to https://thingspeak.com/ and create your ThingSpeak Account if you don’t have. Login to Your Account.

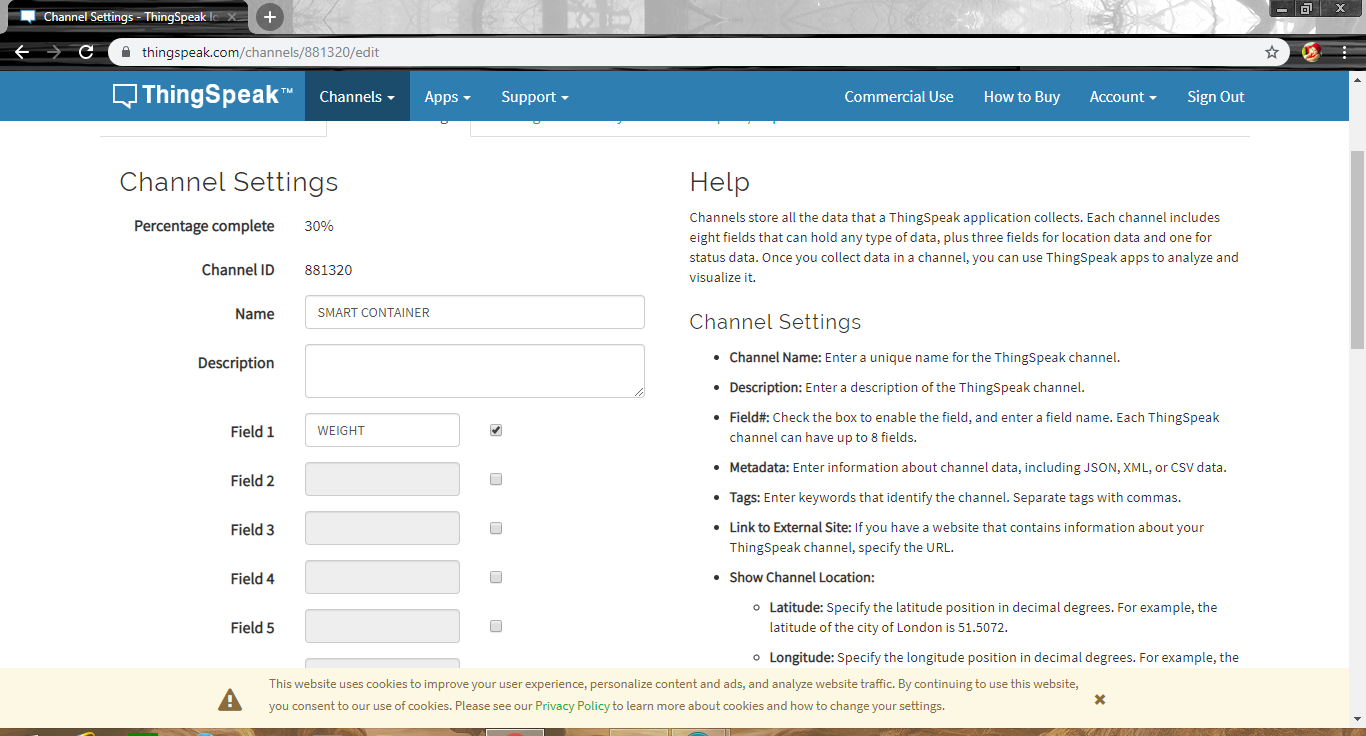
Step 2: Create a Channel by clicking ’New Channel’.

Step 3: Enter the channel details.

Name: Any Name

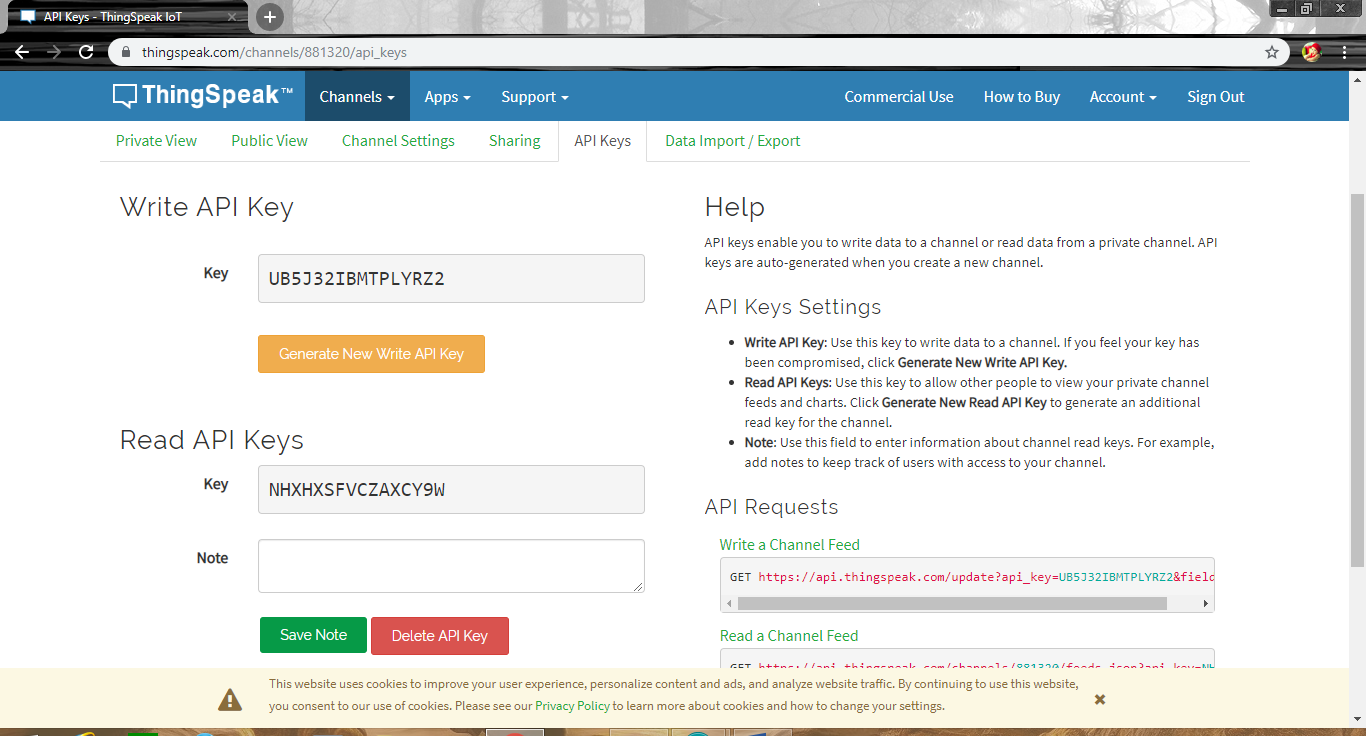
Description: Optional

Field 1: WEIGHT – This will be displayed on the analytics graph.



Save this setting.

Step 4: Now you can see the channels. Click on the ‘API Keys’ tab. Here you will get the Channel ID and API Keys. Note this down.



Step 5: Open Arduino IDE and Install the ThingSpeak Library. To do this go to Sketch>Include Library>Manage Libraries. Search for ThingSpeak and install the library.

Step 6: Now we need to modify the program with your credentials.

In the below code you need to change your Network SSID, Password and your ThingSpeak Channel and API Keys.

Replace the following content in the code,

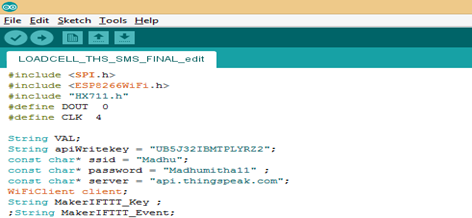
‘Your SSID Here’ – Your Wi-Fi Name

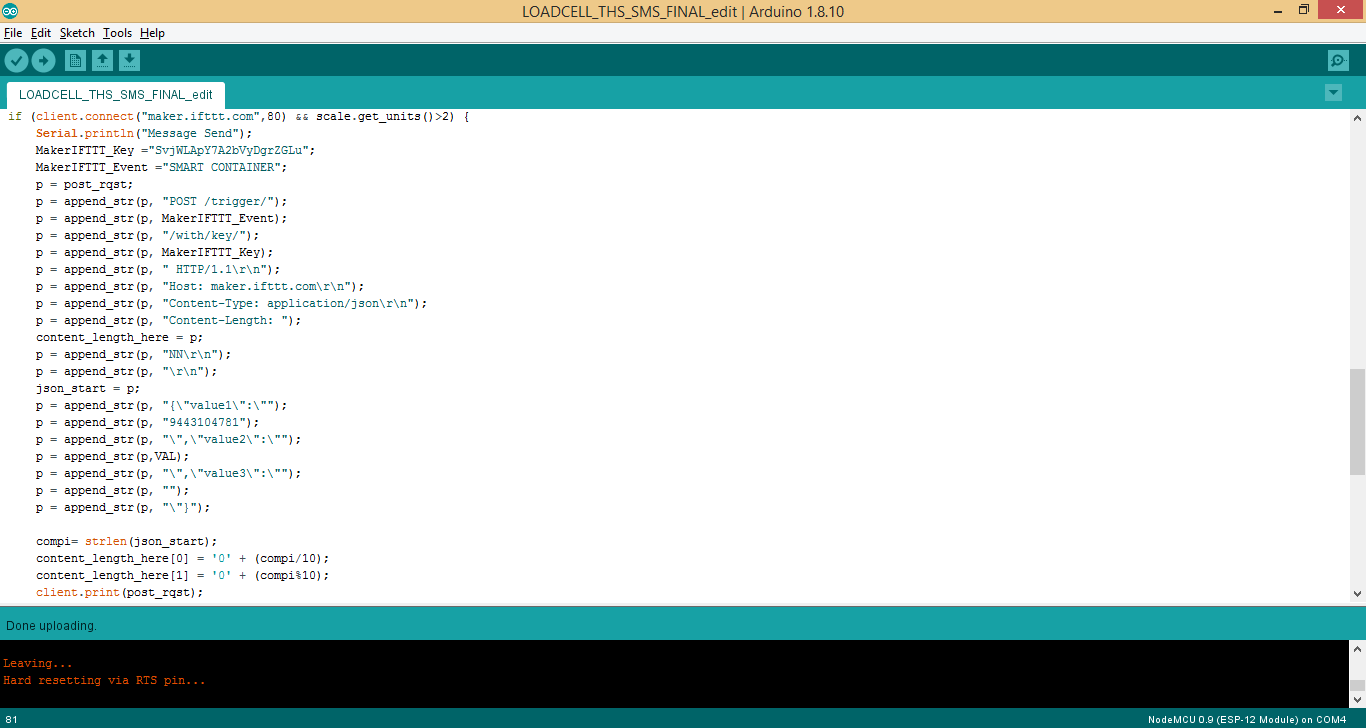
‘Your Password Here’ – Your Wi-Fi Password

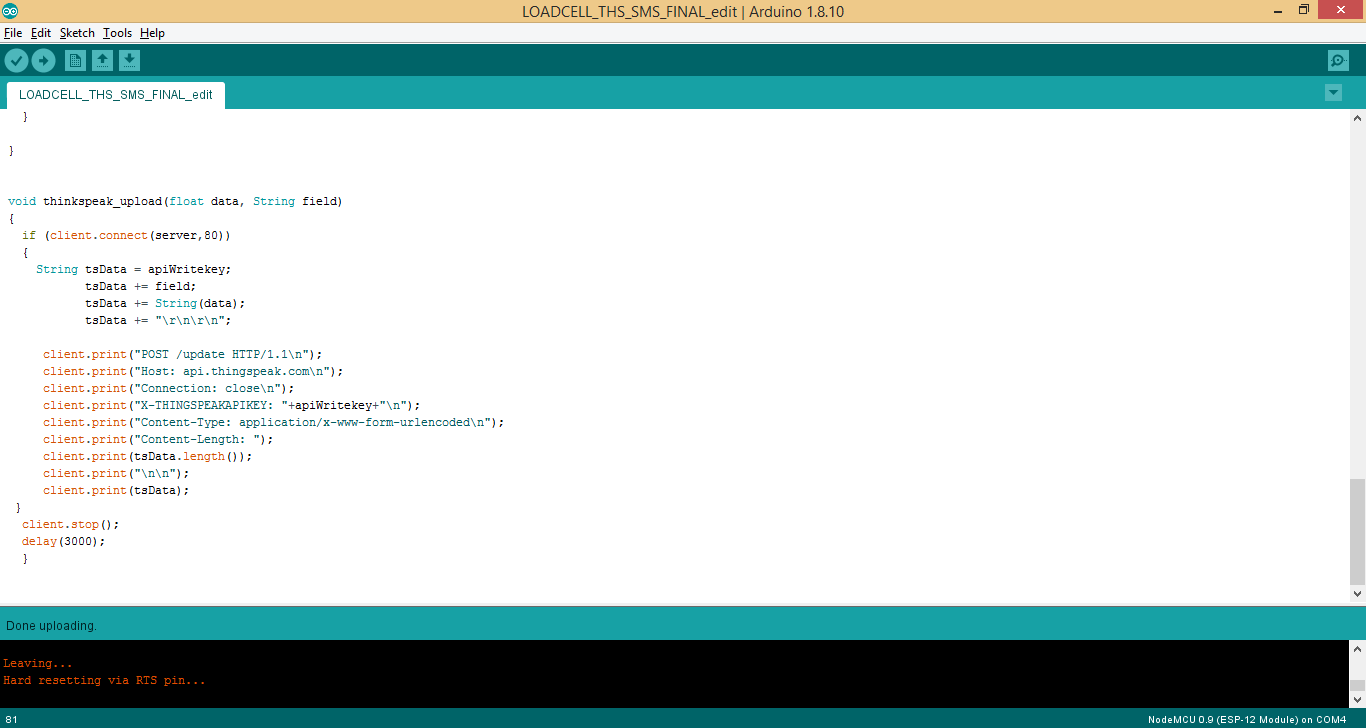
‘YYYYYY’ – Your ThingSpeak Channel Number (without Quotes)

‘XXXXXXXXXXX’ – Your Thing Speak API Key.

Upload the code. Once it is connected to Wi-Fi the data will start uploading to the ThingSpeak Channel. You can now open your Channel and see the data changes plotted on the ThingSpeak.

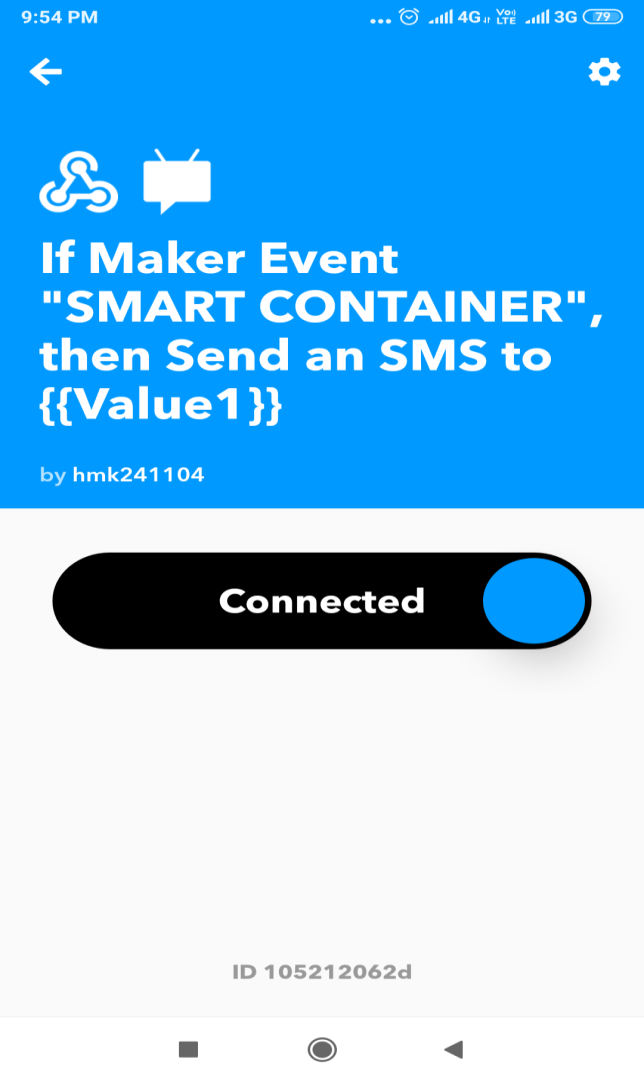




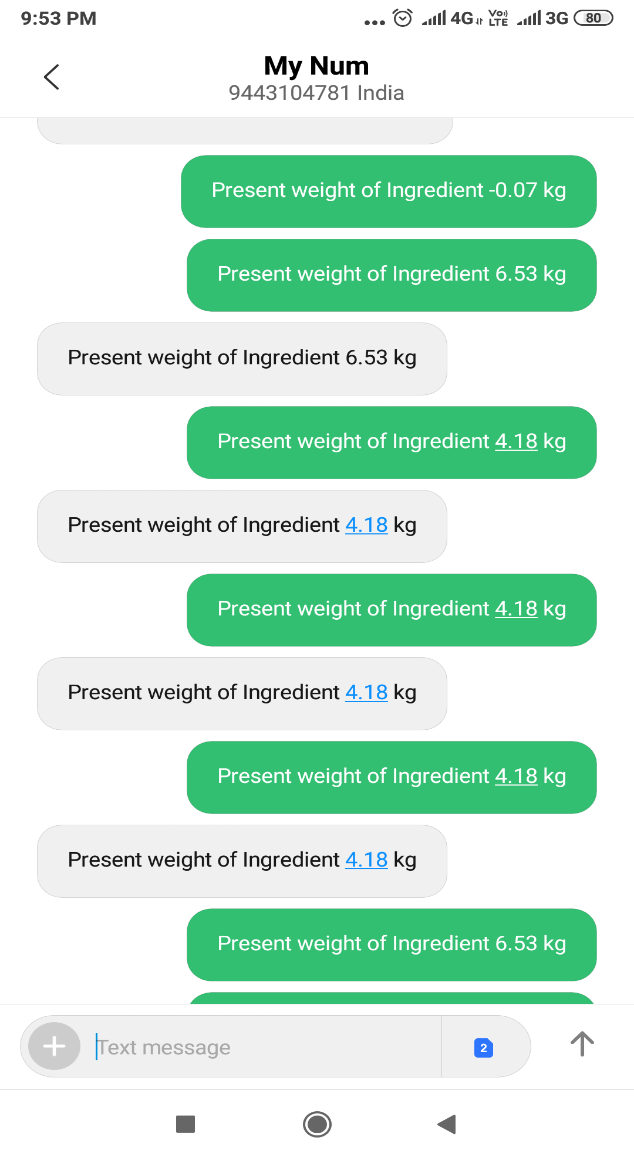


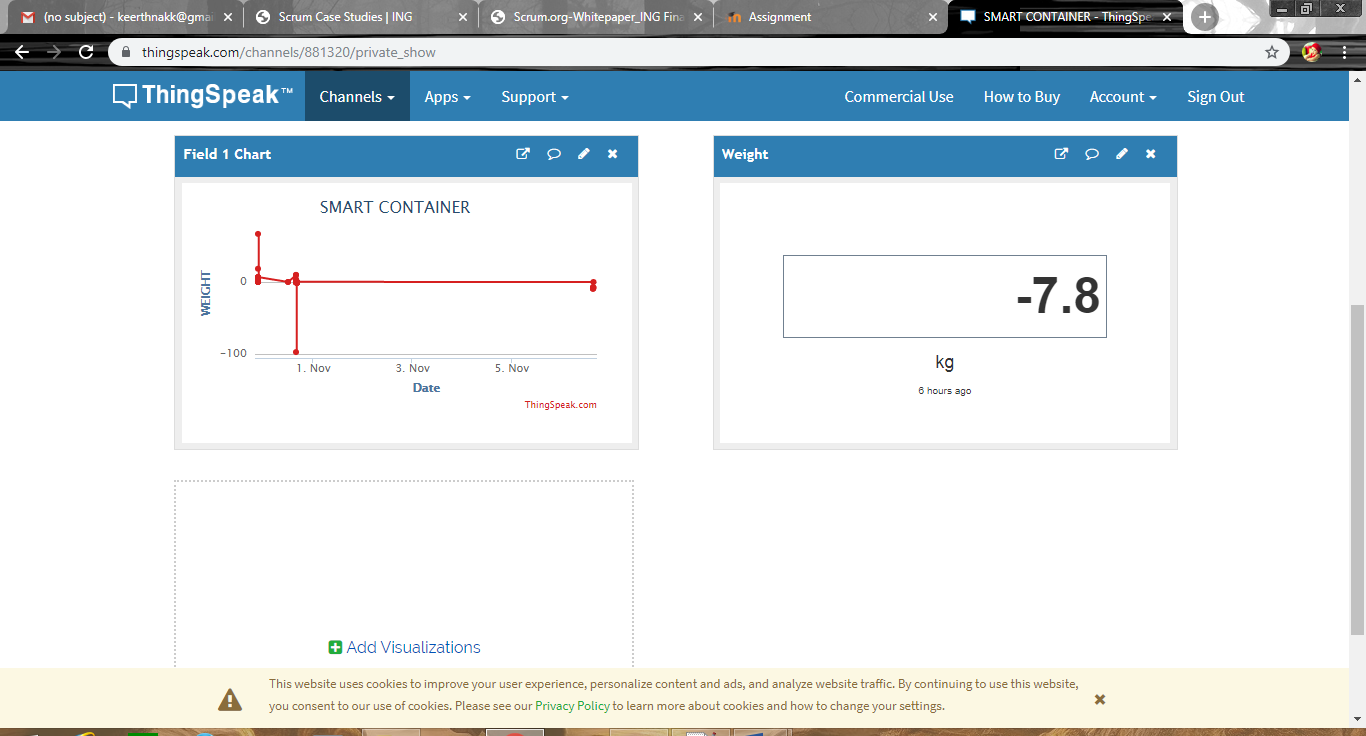
**IFTTT SETUP**

**Application Development**

****

**OUTPUT SCREENSHOTS:**

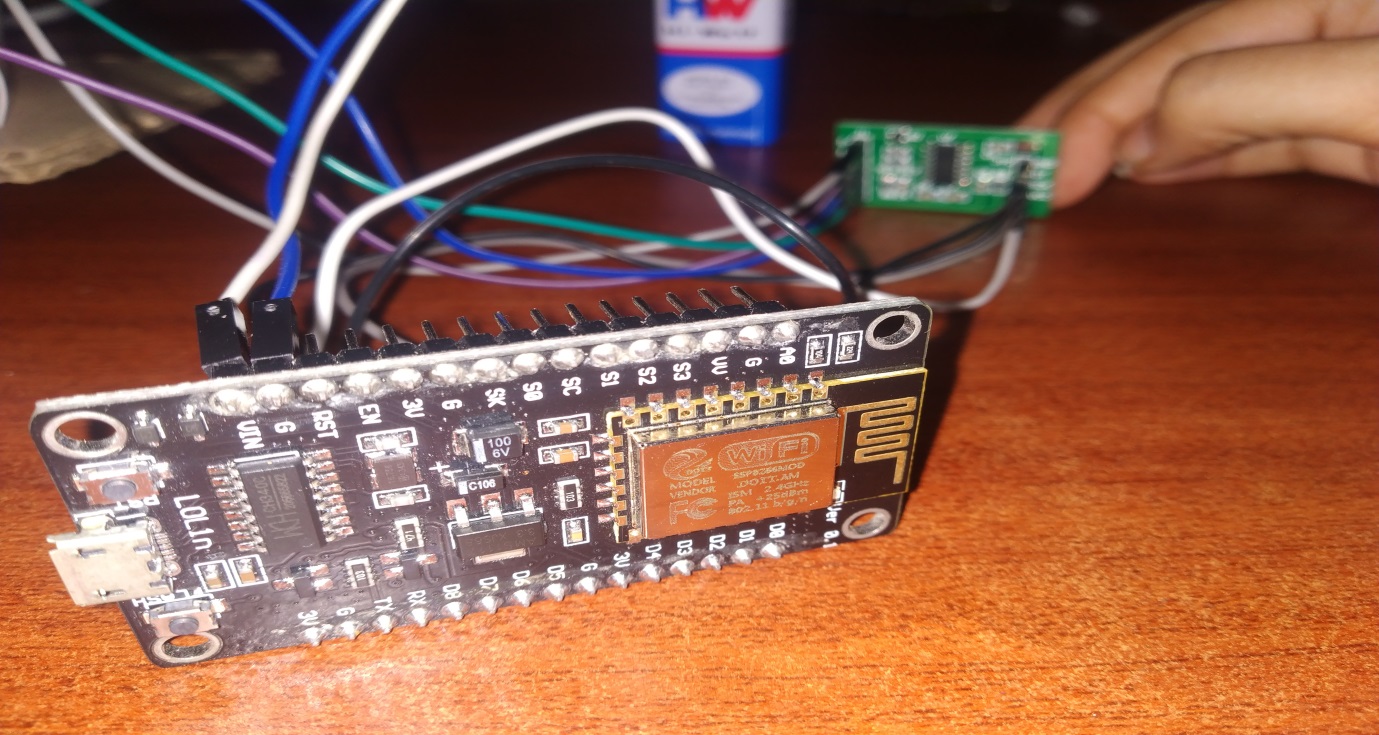


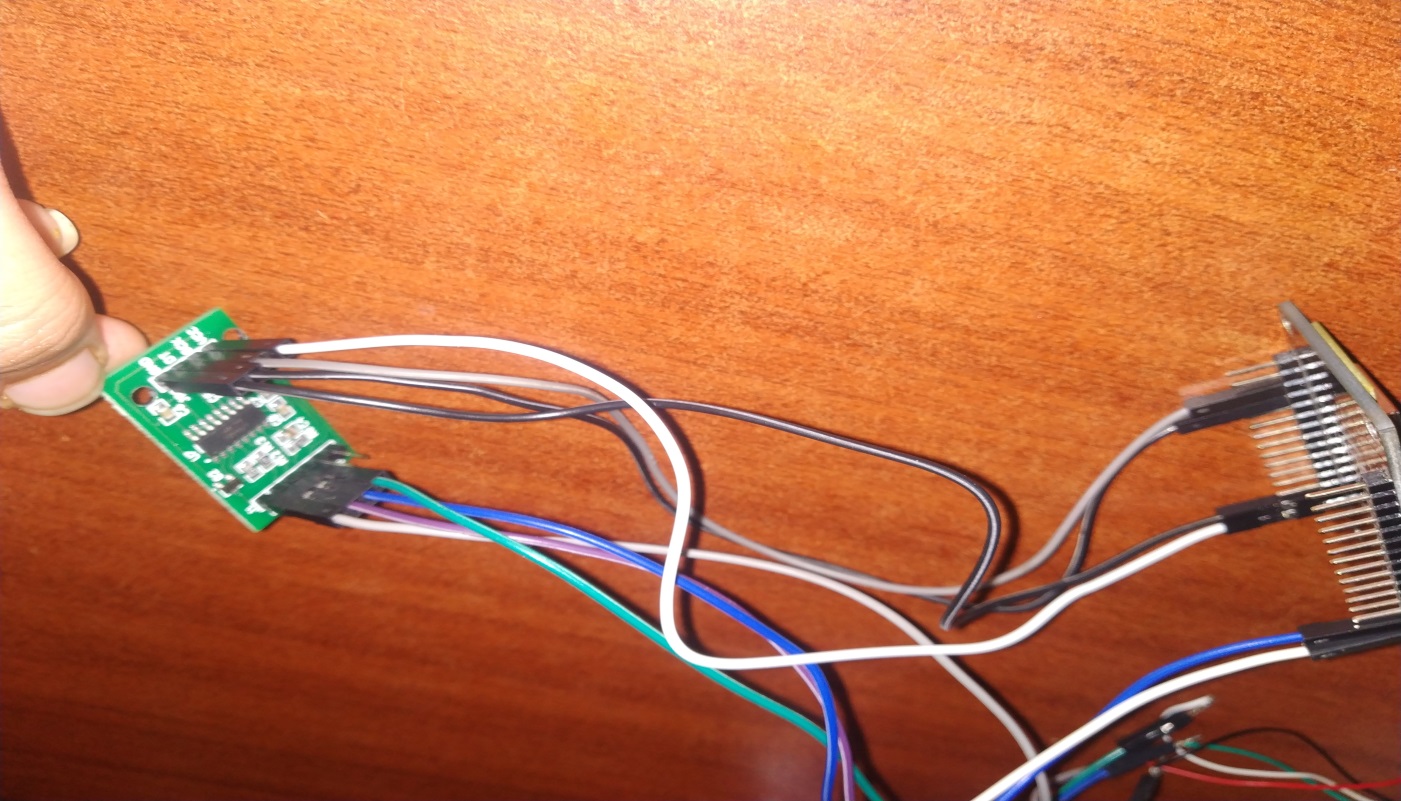


**OVERALL PROJECT COMPONENTS**

****

**NODEMCU, SENSOR AND ACTUATOR INTEGRATION.**

****

****

**RESULTS AND SIGNIFICANCE:**

Desired output of our “Smart container” system is to notify the user via text message with the weight and name of the particular ingredient if its weight goes beyond a given threshold value (often interpreted from the history saved) and also update the same to the ThingSpeak cloud.

Significance:

With a smart kitchen that can automate repetitive tasks such as monitoring appliances and tracking inventory, you can accomplish more in short span of time

Smart appliances that keep track of kitchen performance reduce human error and increase preparation consistency.

With remote monitoring through the Internet of Things, you can keep track of your kitchen no matter where you are.

**CONCLUSION AND FUTURE WORKS:**

Smart container is a simple and much needed automation which helps people in kitchen by tracking their inventory on month basis and notifying them with weight and name of the item if it levels below the threshold value and so they can refill the items before they ran out of stock.