

Project Report

1. INTRODUCTION

1.1 Overview:

Internship Title: Smart Agriculture System based on IOT - SB42054

Project ID:SPS_PRO_101

Project Title:Smart Agriculture system based on IoT

Project Mentor: Durgaprasad Sir from SmartBridge

OBJECTIVES:

- To make a smart APP
- App will give the data from IOT sensor
- App will give the data from open weather api
- These information's will help to make smart agriculture concept possible.

1.2 Purpose:

There is no smart agricultural process available in current market. So, to make the smart agricultural concept possible in a smart way, we have already built a device i.e. an APP which will give the weather data from IOT sensor as well as from open weather API from a particular region. In that region farmer will understand what is the soil temperature, what is the temperature, humidity etc. based on that farmer will understand how much water is required, and then they will press the button motor on/off. Then motor will be on/off manually but it will be on/off by pressing a single button on APP. And also farmer will get the sun rise, sun set, coord, sea level, ground level, wind speed, wind degree etc.

- To make a smart APP
- App will give the data from IOT sensor
- App will give the data from open weather api
- These information's will help to make smart agriculture concept possible.

2. LITERATURE SURVEY

2.1 Existing Problem:

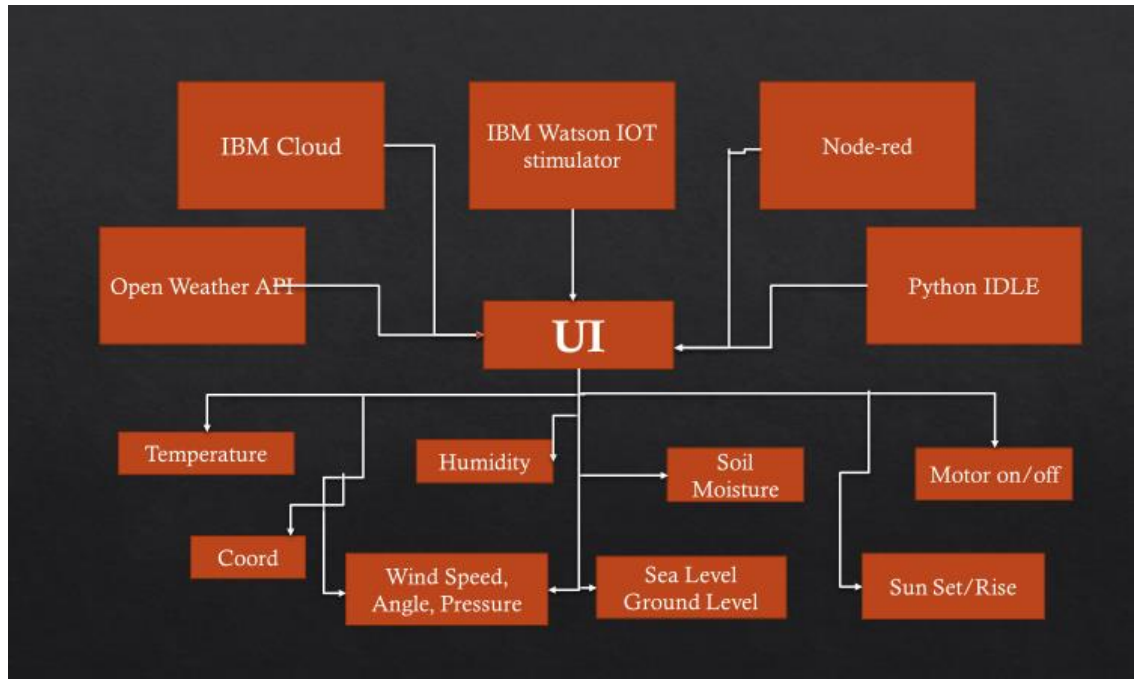
There is no smart agricultural solution available in the current market. Only a smart irrigation system available in the market. But, there is not a smart solution like digital solutions. So, the problem is that farmer is not getting the data of temperature, soil moisture or humidity. Also, farmer was unavailable to get the weather forecast like clouds, broken clouds etc, these data will help them to take necessary safety measures. So, these are the problems. And also, in a single app, farmer is getting all the data of weather of IOT sensor and also from open weather API. Farmer is getting the data like sun rise, sun set, coord value from a single app, these type of benefits were not available in any app. So, these are the existing problems.

2.2 Proposed Solution:

We have built an app. This App will give us the temperature, humidity and soil moisture data from IOT sensor and weather data from open weather API and motor on/off button. By using motor on/off button and checking the data from open weather API and IOT sensor, Farmer will understand how much water is required, then they will button of motor on/off. Then, They can give the sufficient water to their fields. Also, in a single app, farmer is getting all the data of weather of IOT sensor and also from open weather API. Farmer is getting the data like sun rise, sun set, coord value from a single app. So, We are doing our smart concept in agricultural field in a smart way.

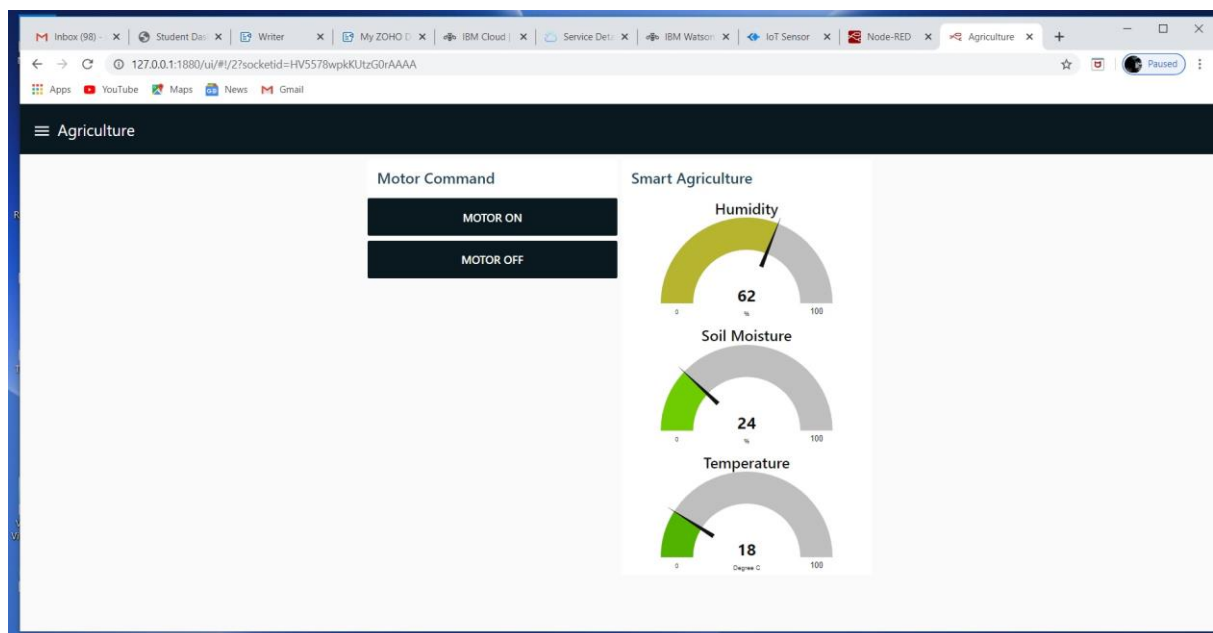
3. THEORITICAL ANALYSIS

3.1 Block Diagram:

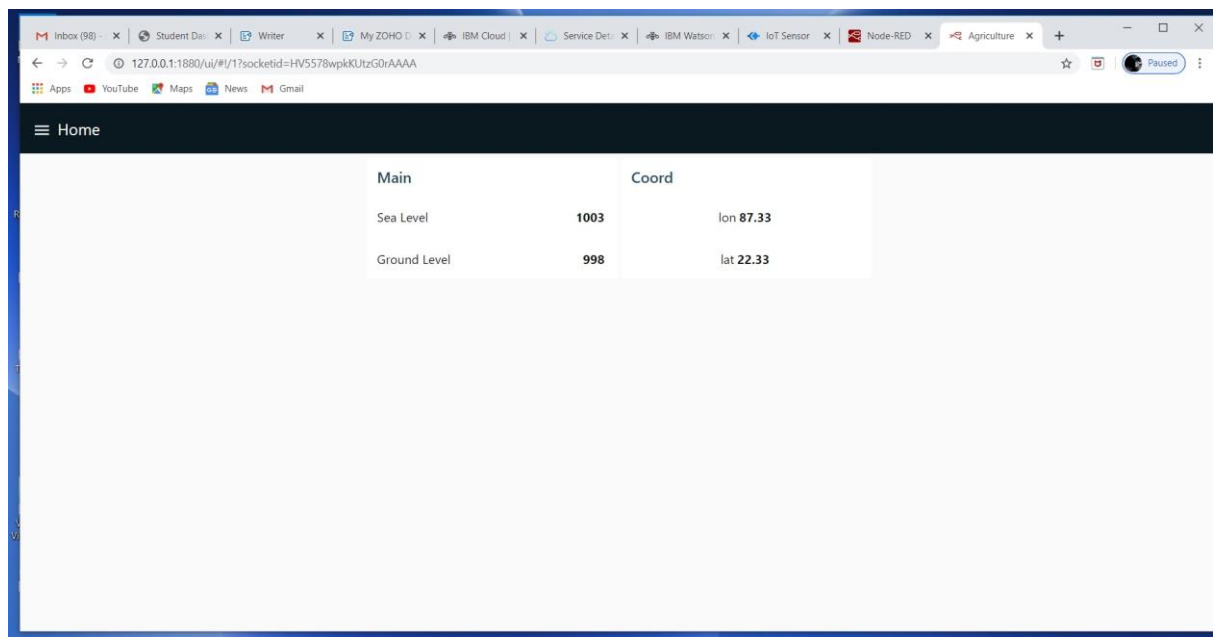


3.2 Software designing:

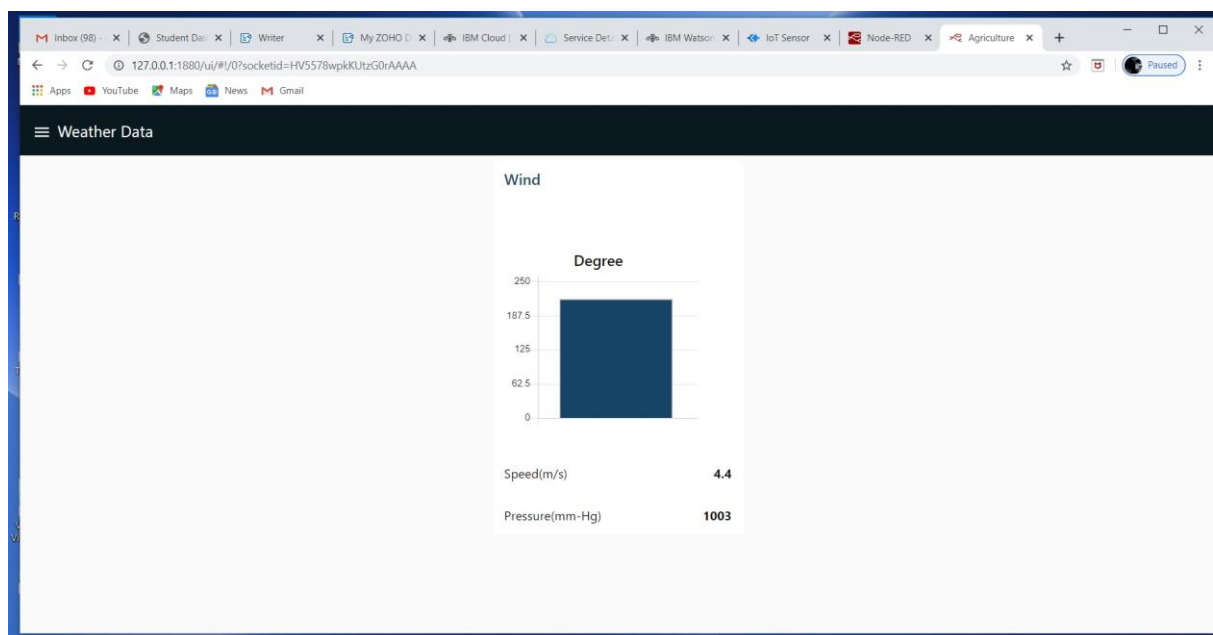
I have built an APP. I am now showing the UI. My app has 3 pages. One is for agricultural data, the data which is coming from IOT sensor and there is motor on/off button. Second page is home page for getting the value for coord, sea level , ground level. Third page is for weather data, the data is coming from open weather API.



Agricultural Page



Home Page



Weather Page

4. EXPERIMENTAL INVESTIGATIONS

I have worked with IBM IOT Cloud then, IBM IOT Watson module. Then worked with IOT sensor stimulator. Then worked with python IDLE. When worked with python idle, at the first time I have faced some problems,

then solved it and made the final solution. Then worked with open weather API.

The screenshot shows the IBM Watson IoT Platform interface. The main page is titled 'Browse Devices' and contains a table of devices. A modal window is open for creating a new event type for a device of type 'IOTDevice'.

Device Type: IOTDevice

Events 1 **New event type**

Event type name: event_1 **Send**

Schedule

20 Every Minute

Payload

Specify the event payload in the editor window or by uploading a CSV file.

```
0 {  
1   "Temperature": random(0, 100)  
2   "humidity": random(0, 100)  
3   "distance": random(0, 100)  
4 }  
5
```

Upload a CSV file **Cancel** **Save**

Device ID	Status	Device Type	Class ID	Date
Arduino	Disconnected	IOTDevice	Device	May

Items per page 50 | 1-1 of 1 item

IBM cloud

The screenshot shows the IBM Watson IoT Platform interface for a project named 'IOTProject'. The dashboard displays two charts: a donut chart showing 'Total 98 °C' and a line chart showing 'humidity' over time. A status bar at the bottom indicates '1 Simulation running'.

IOTProject **Add New Card** **Settings**

Donut chart

Total 98 °C

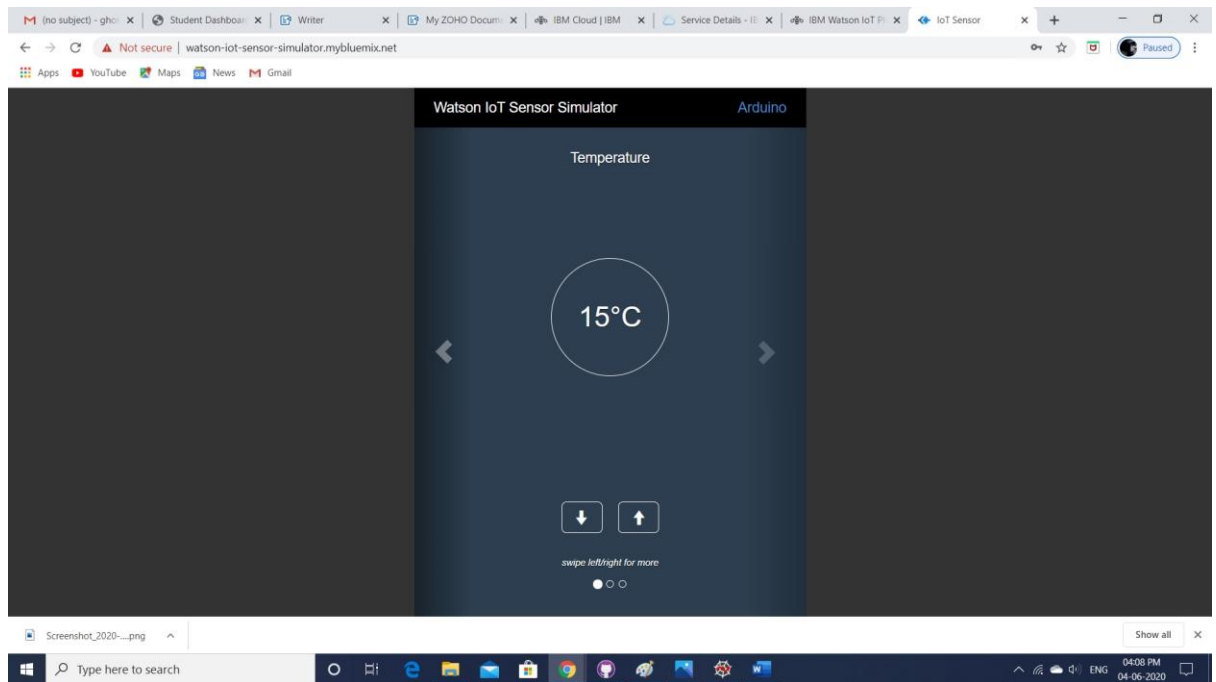
Line chart

humidity

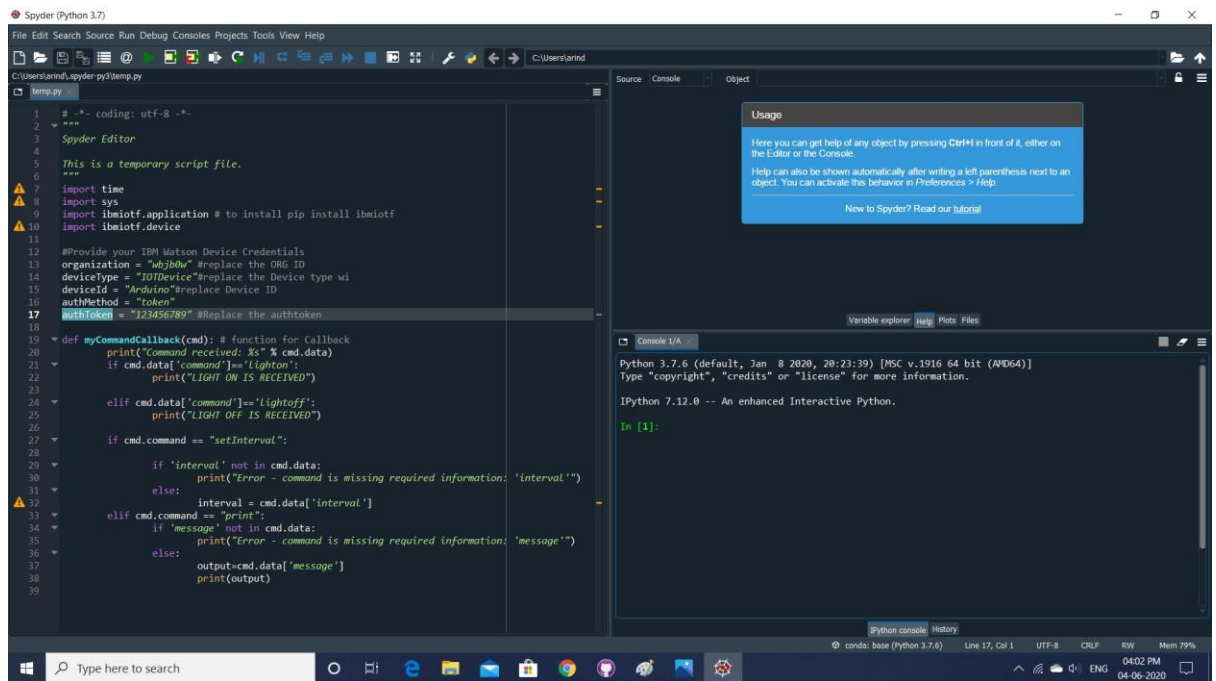
5 minutes now

1 Simulation running

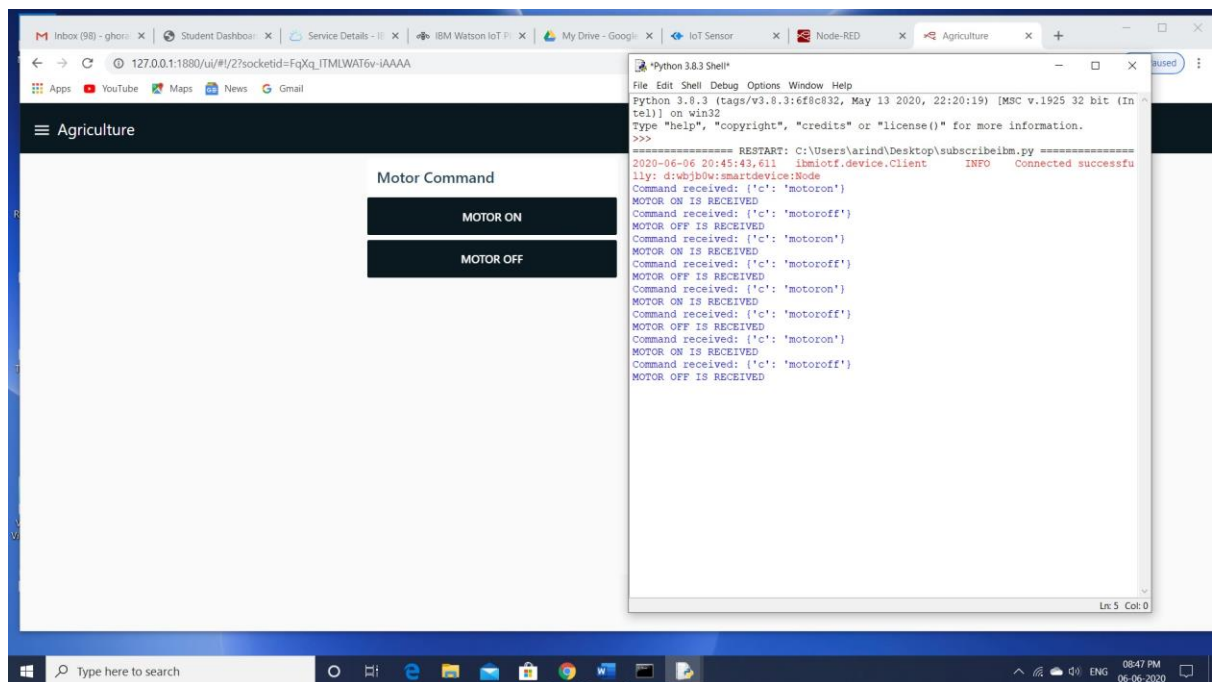
IBM IOT Watson Platform



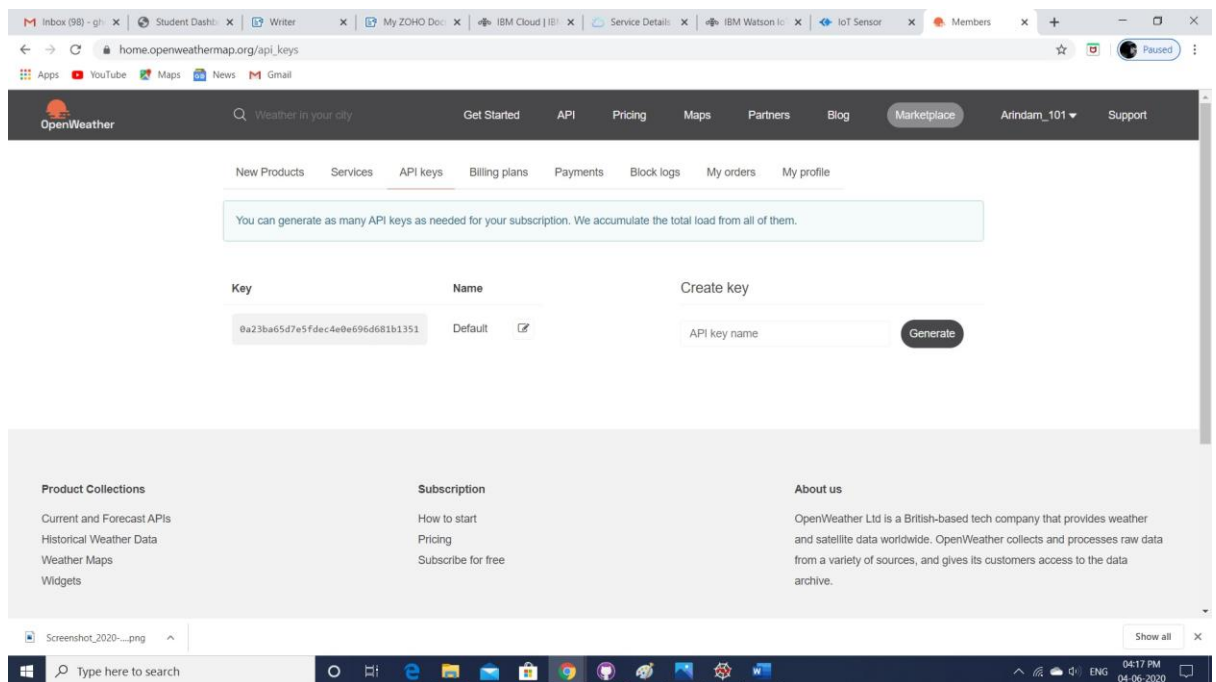
IOT watson sensor stimulator



First time, got some problems, then solved it.



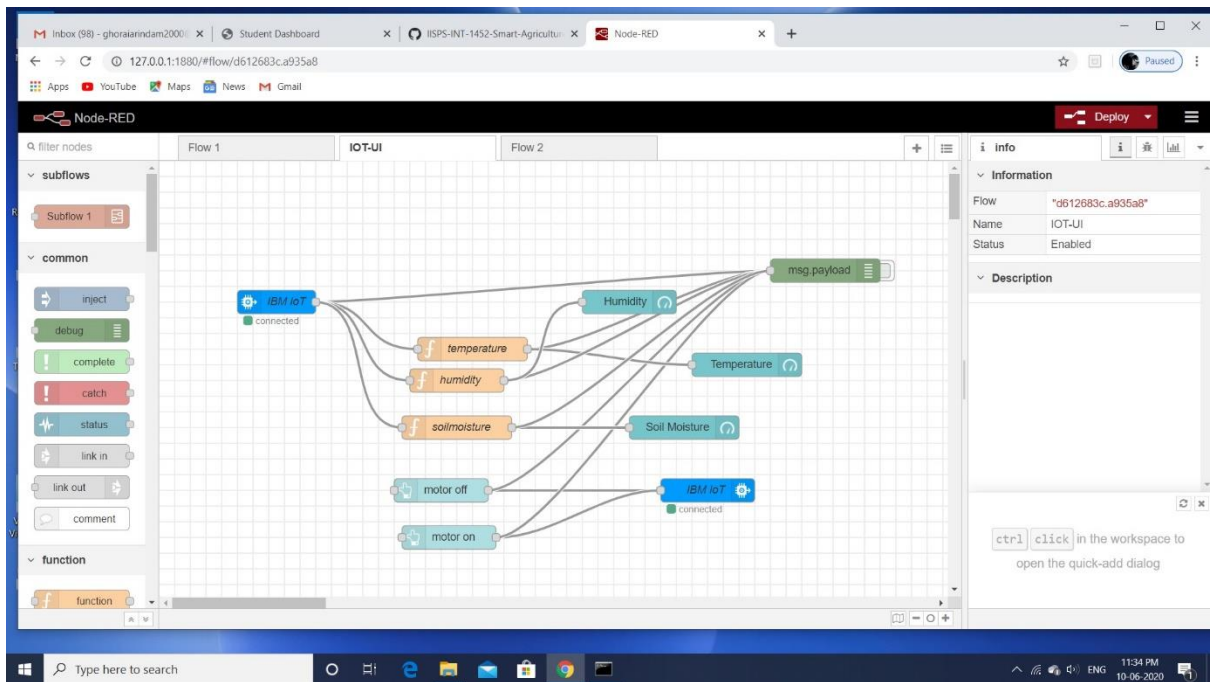
Worked with python idle. Final solution of python code.



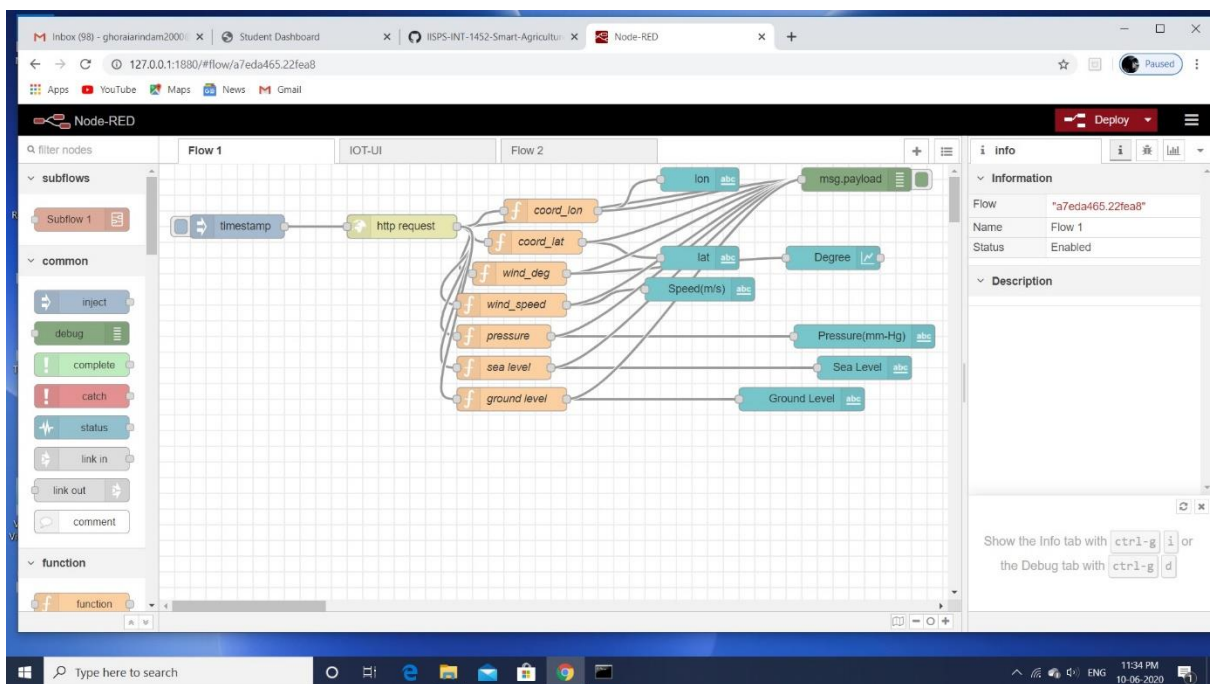
Worked with Open weather API for getting the weather data.

5. FLOWCHART

I have built 2 different flow. One is for agricultural page and another one is for weather data and home page.



Flow for agricultural page



Flow for Home page and weather page

6. RESULT

Finally, I have successfully built an APP. My app has 3 pages.

1. Agricultural Page:

In this page, we are getting the data of temperature, humidity, soil moisture from IOT sensor and Motor on/off button for manually operating purpose.

2. Home Page:

In this page, we are getting the data of coord and sea level and ground level from open weather API.

3. Weather Page:

In this page, we are getting the data of wind speed, wind angle etc. from open weather API.

7. ADVANTAGES & DISADVANTAGES

Advantages: 1. Motor on/off from a single press of a button via APP.

2. Getting the temperature, humidity, soil moisture data from IOT sensor on APP.

3. All weather-related data in a single APP.

4. It's a one stop solution.

Disadvantages: 1. All regions data will not be shown.

2. All farmers are unable to handle android phone.

8. APPLICATIONS

1. All types of farming field.

2. Agricultural Industry.

3. All android phone to get the weather data.

4. Industry.

9. CONCLUSION

Our target is to make smart agricultural concept possible. And I have successfully done this. We have already built a device i.e. an APP which will give the weather data from IOT sensor as well as from open weather API from a particular region. In that region farmer will understand what is the soil temperature, what is the temperature, humidity etc. based on that farmer will understand how much water is required, and then they will press the button motor on/off. Then motor will be on/off manually but it will be on/off by pressing a single button on APP. And also farmer will get the sun rise, sun set, coord, sea level, ground level, wind speed, wind degree etc. It's a one stop solution to make smart farming concept possible.

10. FUTURE SCOPE

All farming industry, agricultural industry will be benefited by using this APP. In future, we will improve our app to not only a particular region, it will be accessible for all regions and give the data of all regions by selecting that regions. All mobile company should use this app as inbuilt feature because people do not need to google again and again, by using a single APP they will get all the data. So, There are a lot of future scopes which will give people a lot of benefits.

11. BIBLIOGRAPHY

Websites referred:

- <https://openweathermap.org/>
- <https://smartinternz.com/assets/docs/Smart%20Home%20Automation%20using%20IBM%20cloud%20Services%20>
- <https://github.com/rachuriharish23/ibmsubscribe>
- <https://smartinternz.com/assets/docs/Sending%20HttpRequest%20to%20Open%20weather%20map%20website%20to%20get%20the%20weather%20forecast>
- <https://nodered.org/docs/getting-started/windows>
- <https://www.python.org/downloads/release/python-383/>

APPENDIX

A. Source Code:

```
import time
import sys
import ibmiotf.application # pip install ibmiotf
import ibmiotf.device

#Provide your IBM Watson Device Credentials
organization = "wbjb0w" #replace the ORG ID
deviceType = "smartdevice"#replace the Device type wi
deviceId = "Node"#replace Device ID
authMethod = "token"
authToken = "12345678" #Replace the authtoken

def myCommandCallback(cmd): # function for Callback
    print("Command received: %s" % cmd.data)
    if cmd.data['c']=='motoron':
        print("MOTOR ON IS RECEIVED")

    elif cmd.data['c']=='motoroff':
        print("MOTOR OFF IS RECEIVED")
```

```

if cmd.command == "setInterval":

    if 'interval' not in cmd.data:
        print("Error - command is missing required information:
'interval'")
    else:
        interval = cmd.data['interval']
elif cmd.command == "print":
    if 'message' not in cmd.data:
        print("Error - command is missing required information:
'message'")
    else:
        output=cmd.data['message']
        print(output)

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,
"auth-method": authMethod, "auth-token": authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)
    #.....

except Exception as e:
    print("Caught exception connecting device: %s" % str(e))
    sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an
event of type "greeting" 10 times
deviceCli.connect()

while True:

    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()

```

-----Thank YOU-----