Task-1:

Milestone-1: Project Scope, Schedule, Team & Deliverable

Project Summary:

Agriculture plays a crucial role in the life of an economy. It is the backbone of our economic system, so improving the quality and way of production is crucial. Here comes the Smart Agriculture system.

Smart agriculture helps in automated farming, collection of data from the field and then analyses it so that the farmer can make accurate decision in order to grow high quality crop.

IoT based Smart Farming also improves the entire Agriculture system by monitoring the field in real-time. With the help of sensors and interconnectivity, the Internet of Things in Agriculture has not only saved the time of the farmers but has also reduced the extravagant use of resources such as Water and Electricity.

So in this project I have developed a mobile application using which a farmer can monitor the temperature, humidity and soil moisture parameters along with weather forecasting details. Based on these details he can water the crops by controlling the motors through the app.

Project Requirements:

- Github and slackAccount
- IBM Account
- Node-RED
- Python
- Open Weather API
- MIT app inventor

Functional Requirements:

Sno	Functional requirement description
1.	Farmer must be able to receive the weather forecast every hour.
2.	The mobile app must be user friendly to the farmer
3.	Farmer must be able to monitor the temperature, humidity and soil moisture parameters along with weather forecasting details.
4.	Based on the forecast parameters he must be able to control the motor if needed.

Technical Requirements:

- The farmer must have a mobile to use the app.
- He must have basic knowledge to operate the app.
- The app must be user friendly.
- The app must be reliable and efficient.

Software Requirements:

- IBM cloud Account
- Node-RED
- Watson IoT platform
- python
- IoT simulator
- Open Weather API

Project Deliverables:

An efficient and reliable app to monitor the temperature, humidity, soil moisture and control the motors to turn water on/off if needed.

Project Team:

JEEVAK RAJ S-INTERNSHIP(SB15881)

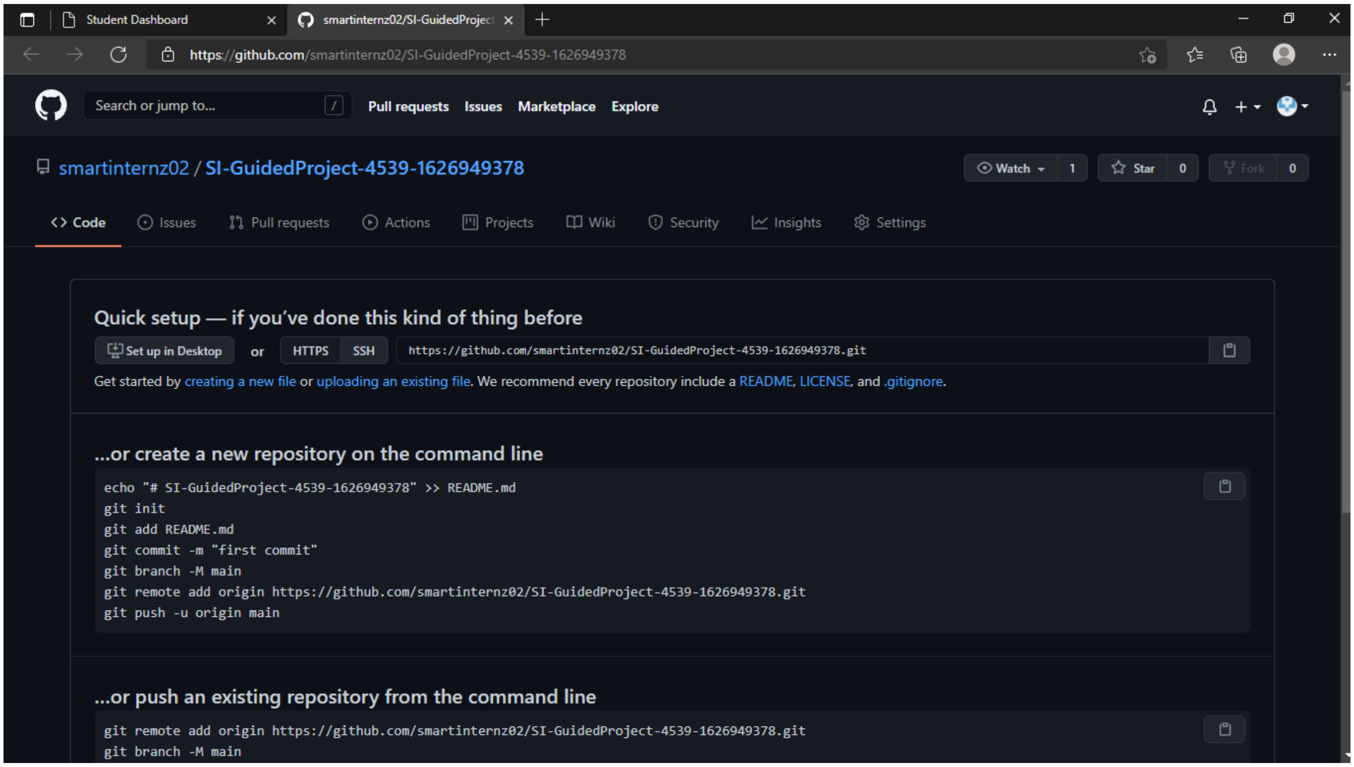
Project Schedule:

- creating all the accounts needed - July 23
- installing required software - July 24
- connecting to IoT simulator and installing required nodes- July25
- setting up Open Weather API - July 26
- Building a Web App - July 27
- Configuring device and controlling motor - July 28
- remaining work - July 30
- Report making -everday

Milestone-2:

Setup the development environment

GITHUB Account:



Quick setup — if you've done this kind of thing before

https://github.com/smartinternz02/SI-GuidedProject-4539-1626949378.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

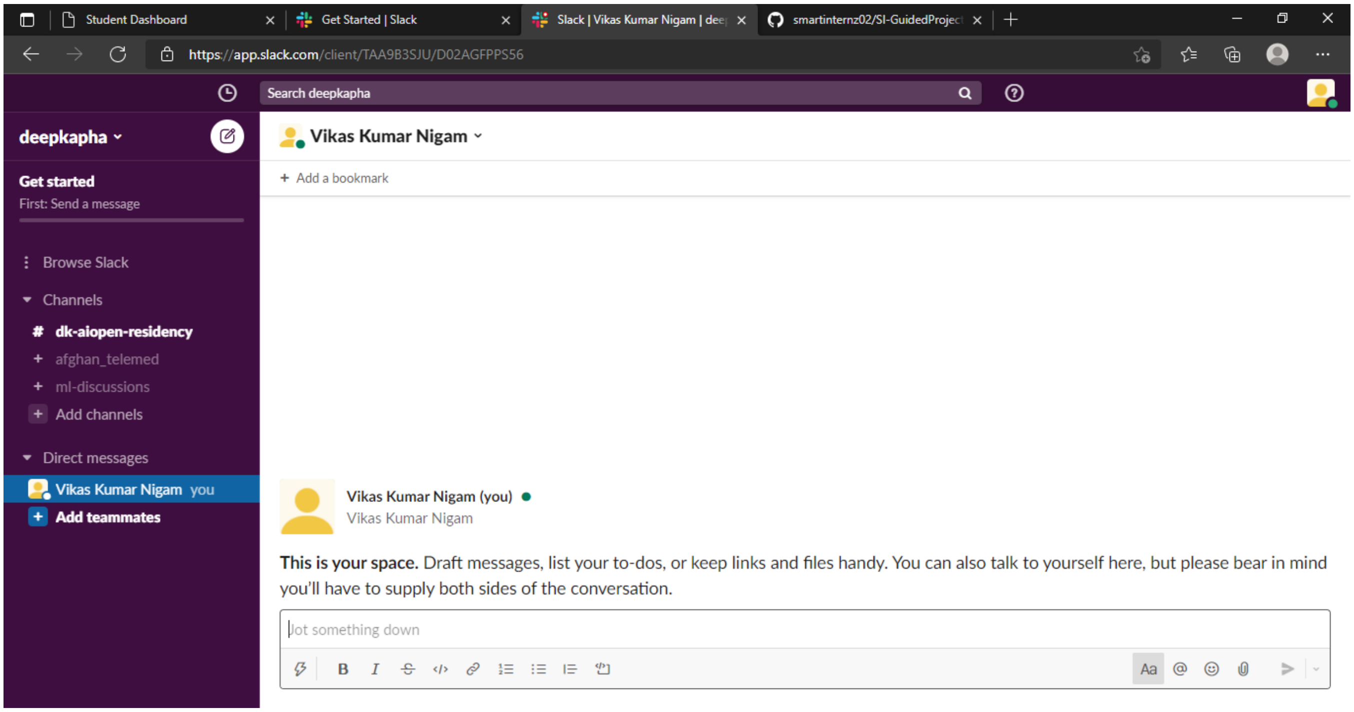
...or create a new repository on the command line

```
echo "# SI-GuidedProject-4539-1626949378" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/smartinternz02/SI-GuidedProject-4539-1626949378.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/smartinternz02/SI-GuidedProject-4539-1626949378.git
git branch -M main
```

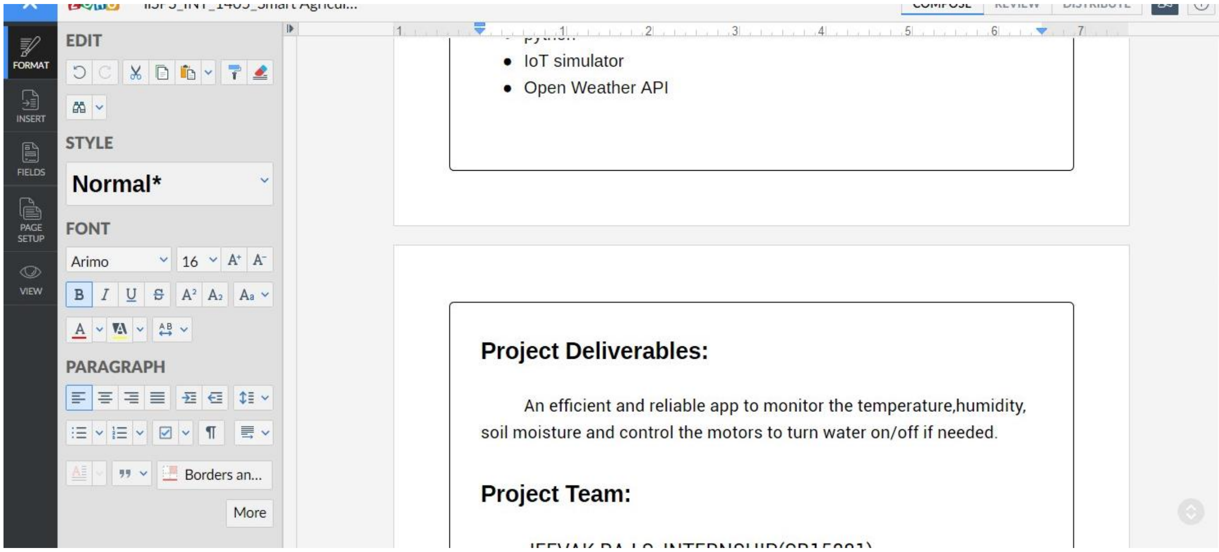
Slack Account:



Vikas Kumar Nigam (you)
Vikas Kumar Nigam

This is your space. Draft messages, list your to-dos, or keep links and files handy. You can also talk to yourself here, but please bear in mind you'll have to supply both sides of the conversation.

Document writer:



- python
- IoT simulator
- Open Weather API

Project Deliverables:

An efficient and reliable app to monitor the temperature,humidity, soil moisture and control the motors to turn water on/off if needed.

Project Team:

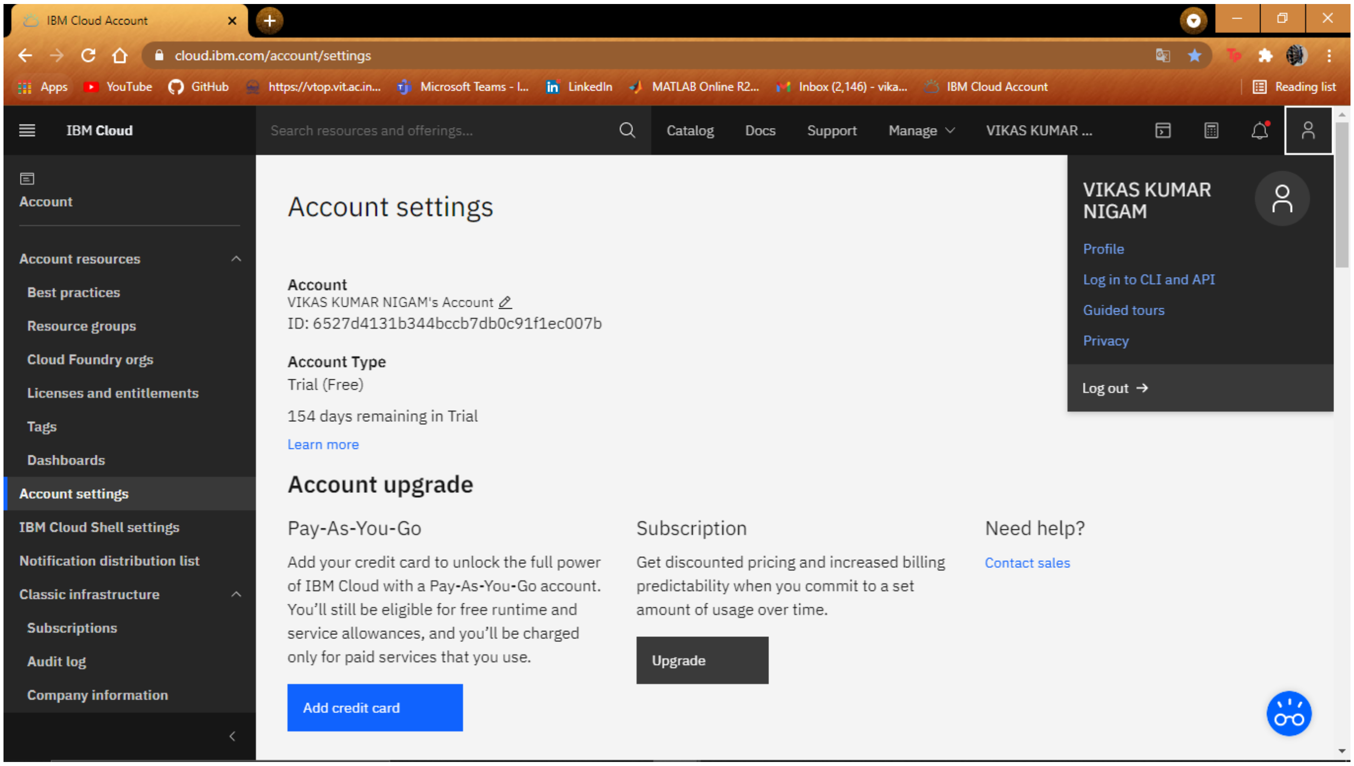
JEEVAK RA L S INTERNSHIP(SB15081)

Task-2

Explore IBM Cloud Platform

Milestone-1: Create IBM Cloud Account

IBM Academic Initiative Account :



Account settings

Account
VIKAS KUMAR NIGAM's Account
ID: 6527d4131b344bccb7db0c91f1ec007b

Account Type
Trial (Free)

154 days remaining in Trial

Learn more

Account upgrade

Pay-As-You-Go

Add your credit card to unlock the full power of IBM Cloud with a Pay-As-You-Go account. You'll still be eligible for free runtime and service allowances, and you'll be charged only for paid services that you use.

Add credit card

Subscription

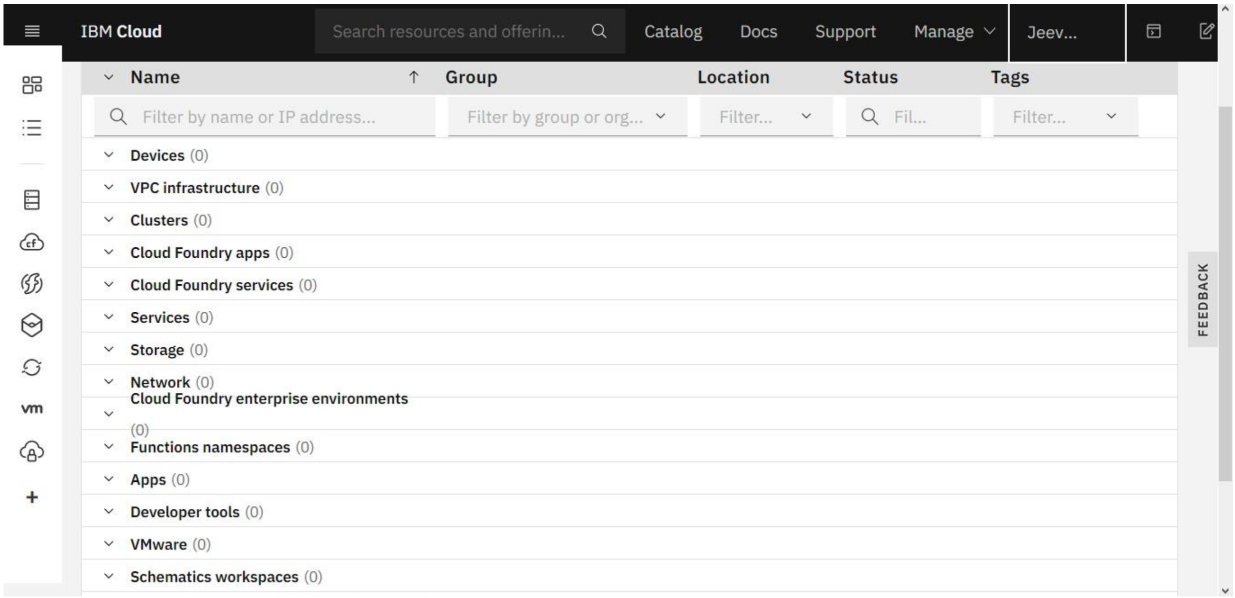
Get discounted pricing and increased billing predictability when you commit to a set amount of usage over time.

Upgrade

Need help?

Contact sales

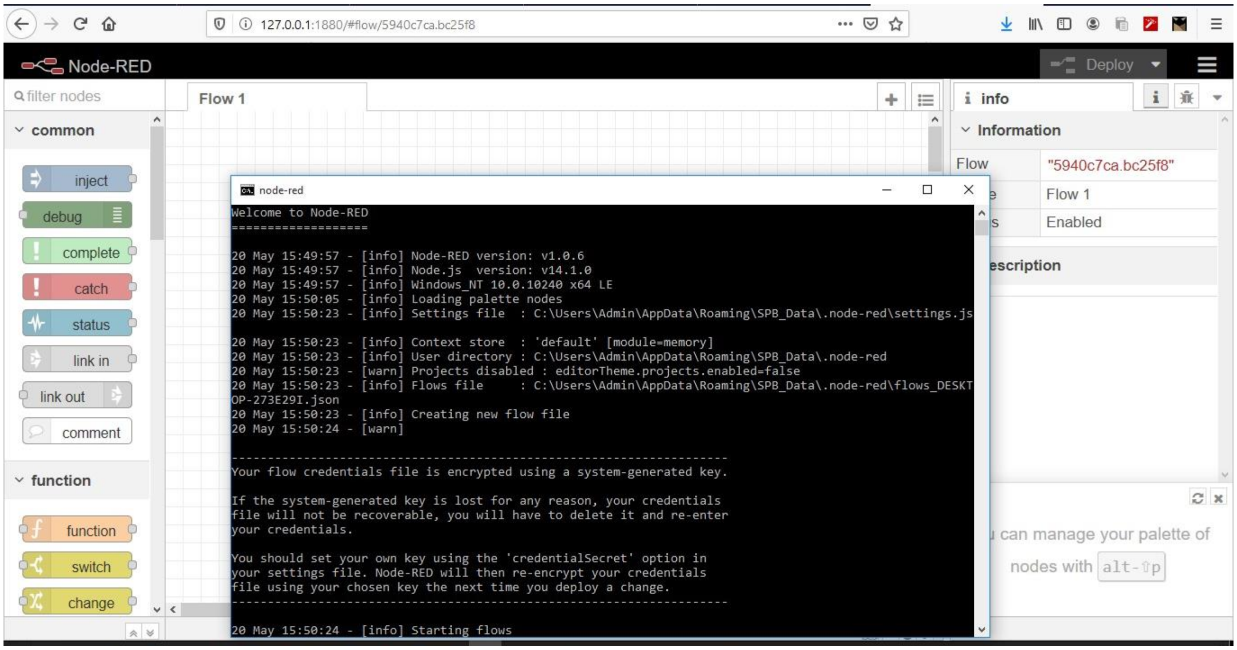
IBM cloud Account:



Devices (0)
VPC infrastructure (0)
Clusters (0)
Cloud Foundry apps (0)
Cloud Foundry services (0)
Services (0)
Storage (0)
Network (0)
Cloud Foundry enterprise environments (0)
Functions namespaces (0)
Apps (0)
Developer tools (0)
VMware (0)
Schematics workspaces (0)

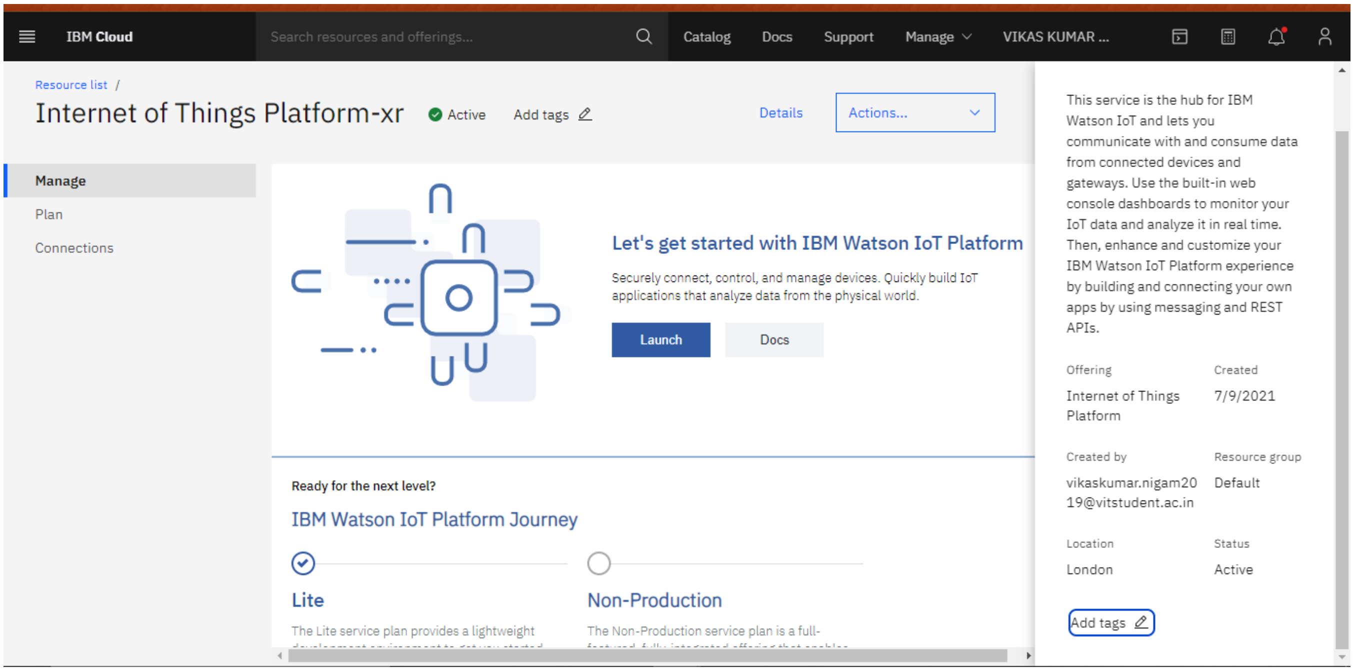
Milestone-2: Install Node-RED locally

Node-RED:



```
Welcome to Node-RED
===================

20 May 15:49:57 - [info] Node-RED version: v1.0.6
20 May 15:49:57 - [info] Node.js  version: v14.1.0
20 May 15:49:57 - [info] Windows_NT 10.0.10240 x64 LE
20 May 15:50:05 - [info] Loading palette nodes
20 May 15:50:23 - [info] Settings file  : C:\Users\Admin\AppData\Roaming\SPB_Data\.node-red\settings.js

20 May 15:50:23 - [info] Context store  : 'default' [module=memory]
20 May 15:50:23 - [info] User directory : C:\Users\Admin\AppData\Roaming\SPB_Data\.node-red
20 May 15:50:23 - [warn] Projects disabled : editorTheme.projects.enabled=false
20 May 15:50:23 - [info] Flows file     : C:\Users\Admin\AppData\Roaming\SPB_Data\.node-red\flows_DESKT
OP-273E29I.json
20 May 15:50:23 - [info] Creating new flow file
20 May 15:50:24 - [warn]

---------------------------------------------------------------------
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
---------------------------------------------------------------------

20 May 15:50:24 - [info] Starting flows
```

Milestone-3: IBM Watson IoT platform:



Internet of Things Platform-xr ● Active Add tags

Let's get started with IBM Watson IoT Platform

Securely connect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.

Ready for the next level?

IBM Watson IoT Platform Journey

✓ Lite

The Lite service plan provides a lightweight

○ Non-Production

The Non-Production service plan is a full-

This service is the hub for IBM Watson IoT and lets you communicate with and consume data from connected devices and gateways. Use the built-in web console dashboards to monitor your IoT data and analyze it in real time. Then, enhance and customize your IBM Watson IoT Platform experience by building and connecting your own apps by using messaging and REST APIs.

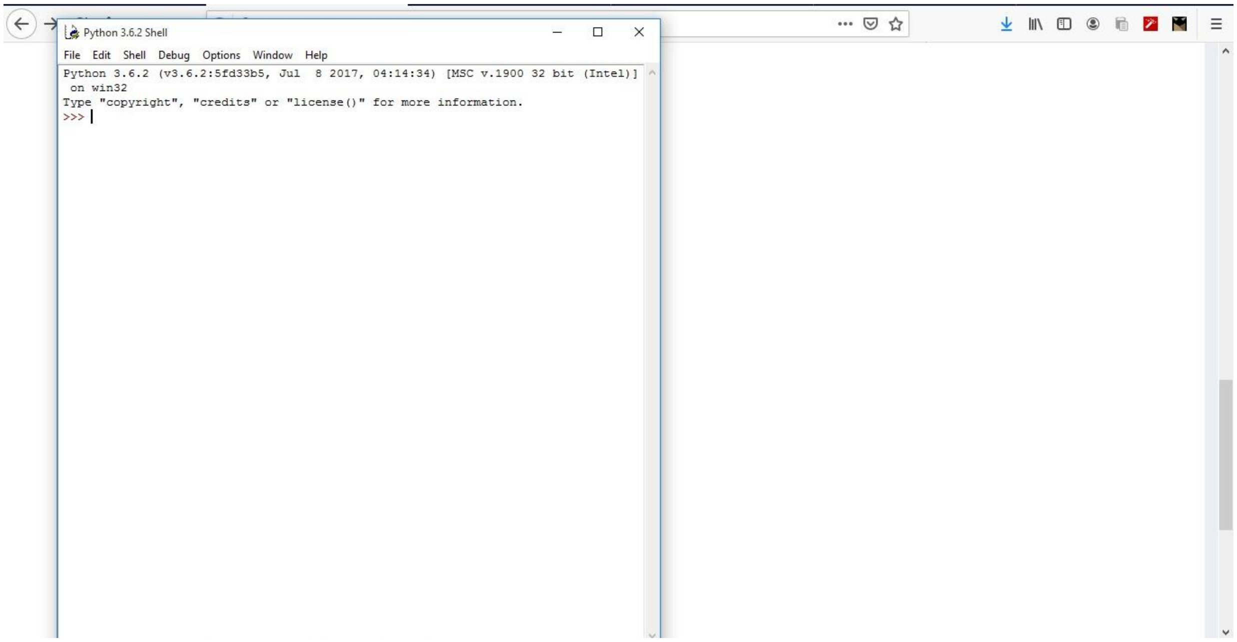
Offering	Created
Internet of Things Platform	7/9/2021

Created by	Resource group
vikaskumar.nigam2019@vitstudent.ac.in	Default

Location	Status
London	Active

Add tags

Milestone-4: Python IDE:

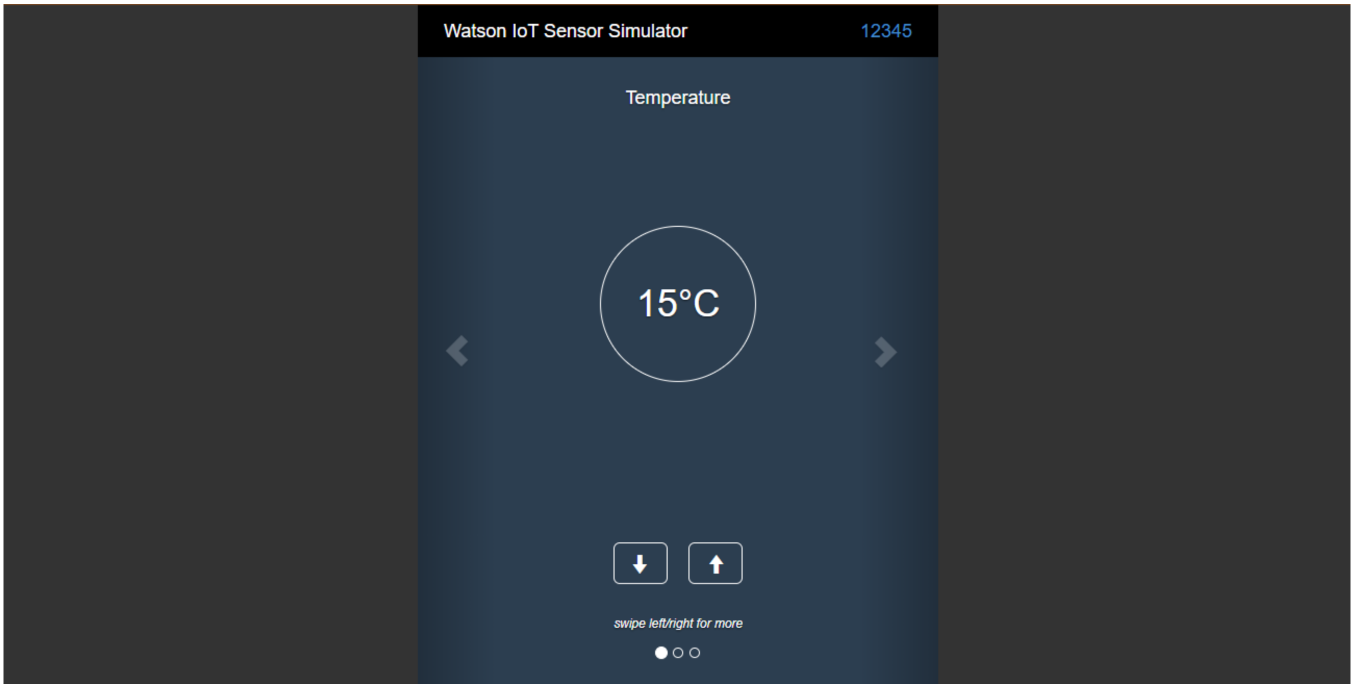


```
Python 3.6.2 (v3.6.2:5fd33b5, Jul  8 2017, 04:14:34) [MSC v.1900 32 bit (Intel)]
 on win32
Type "copyright", "credits" or "license()" for more information.
>>> 
```

Task-3

Connect the IoT simulator to Watson IoT platform

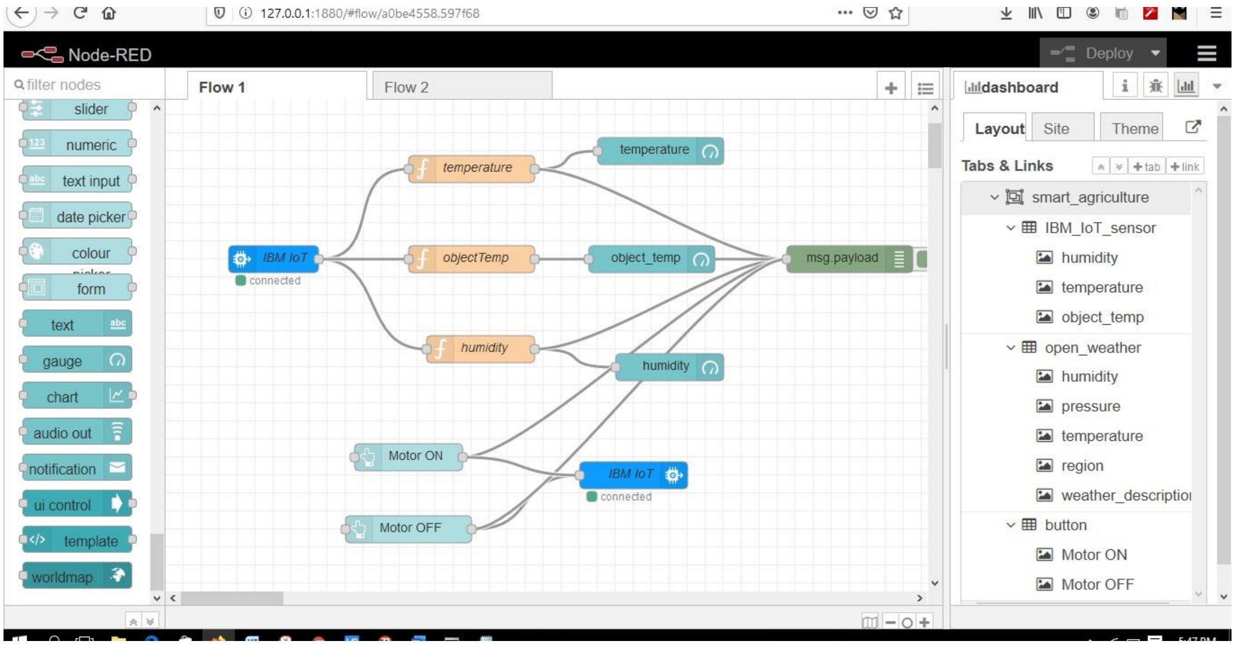
Milestone-1: screenshot of connection



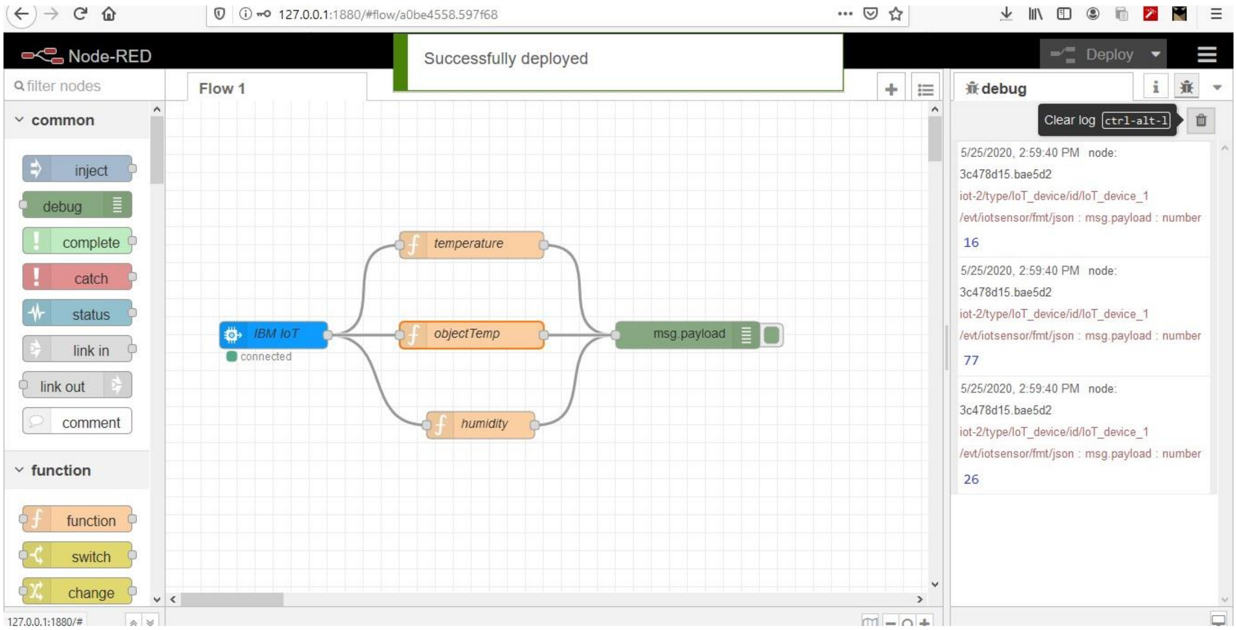
Task-4

Configure the Nodered to get the Data from IBM IOT platform and Open Weather API

Milestone-1: Installing required nodes:



Milestone-2: Connect to your IBM IOT device to get the Simulator Data

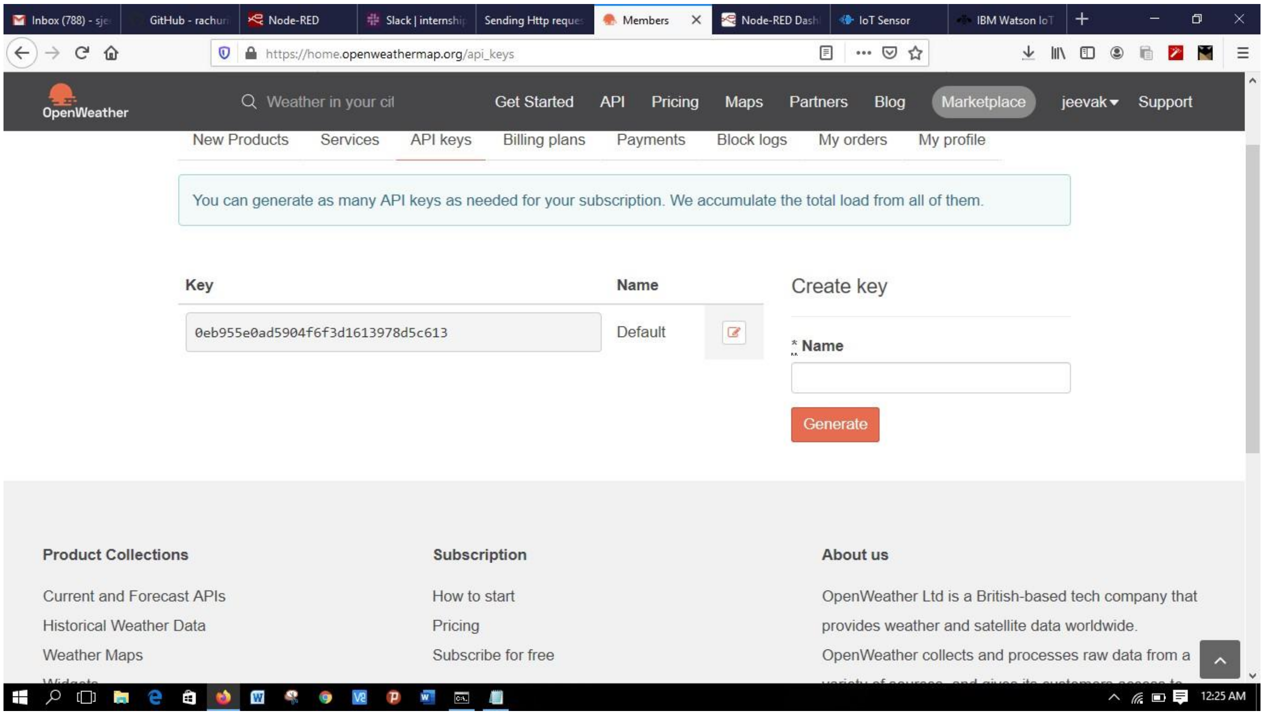


Successfully deployed

```
5/25/2020, 2:59:40 PM  node:
3c478d15.bae5d2
iot-2/type/IoT_device/id/IoT_device_1
/evt/iotsensor/fmt/json : msg.payload : number
16
5/25/2020, 2:59:40 PM  node:
3c478d15.bae5d2
iot-2/type/IoT_device/id/IoT_device_1
/evt/iotsensor/fmt/json : msg.payload : number
77
5/25/2020, 2:59:40 PM  node:
3c478d15.bae5d2
iot-2/type/IoT_device/id/IoT_device_1
/evt/iotsensor/fmt/json : msg.payload : number
26
```

Milestone-3:

Create an account in Open Weather API and Configure your Open weather API Platform



You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key	Name
0eb955e0ad5904f6f3d1613978d5c613	Default

Create key

* Name

Product Collections

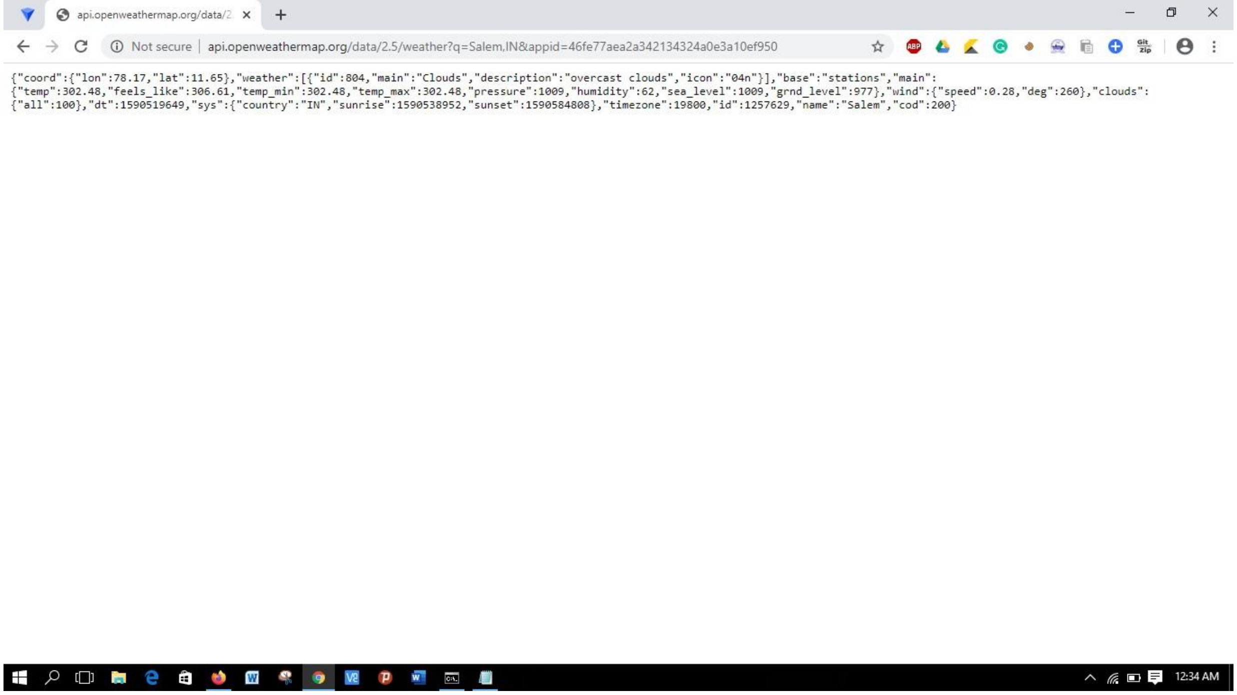
Current and Forecast APIs
Historical Weather Data
Weather Maps

Subscription

How to start
Pricing
Subscribe for free

About us

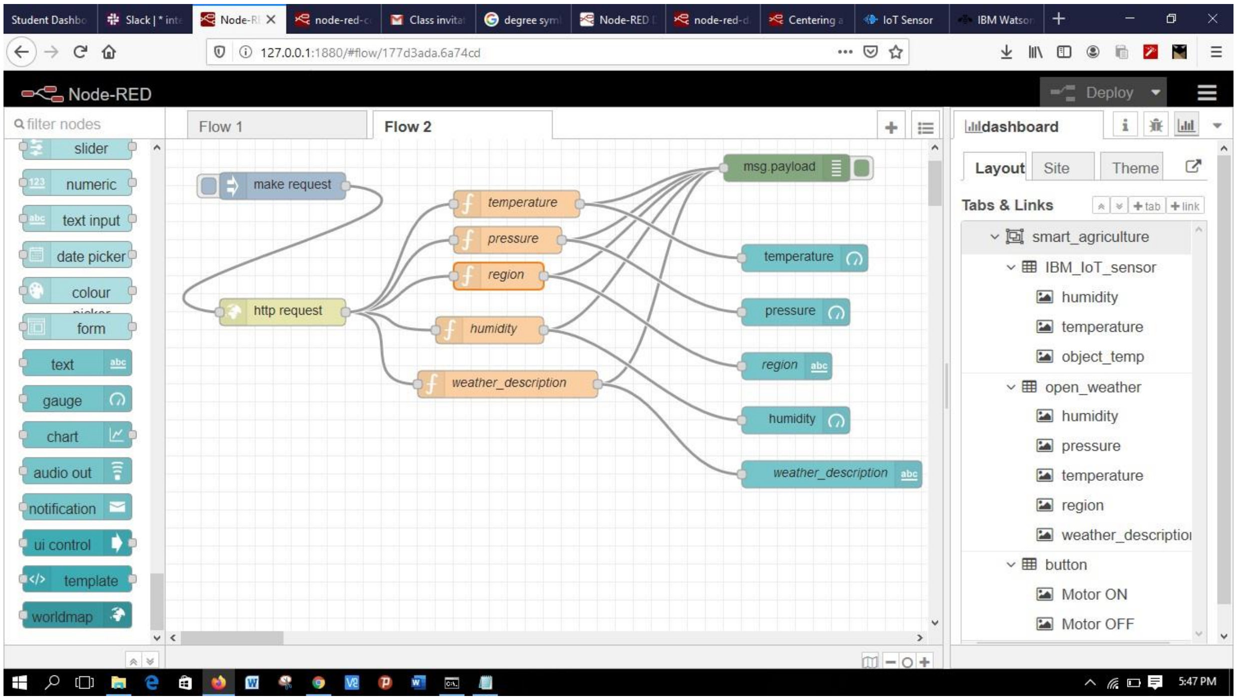
OpenWeather Ltd is a British-based tech company that provides weather and satellite data worldwide. OpenWeather collects and processes raw data from a


Not secure
api.openweathermap.org/data/2.5/weather?q=Salem,IN&appid=46fe77aea2a342134324a0e3a10ef950
12:34 AM

```
{"coord":{"lon":78.17,"lat":11.65},"weather":[{"id":804,"main":"Clouds","description":"overcast clouds","icon":"04n"}],"base":"stations","main":
{"temp":302.48,"feels_like":306.61,"temp_min":302.48,"temp_max":302.48,"pressure":1009,"humidity":62,"sea_level":1009,"grnd_level":977},"wind":{"speed":0.28,"deg":260},"clouds":
{"all":100},"dt":1590519649,"sys":{"country":"IN","sunrise":1590538952,"sunset":1590584808},"timezone":19800,"id":1257629,"name":"Salem","cod":200}
```

Milestone-4:

Configure your nodered to get the weather forecasting data using http requests

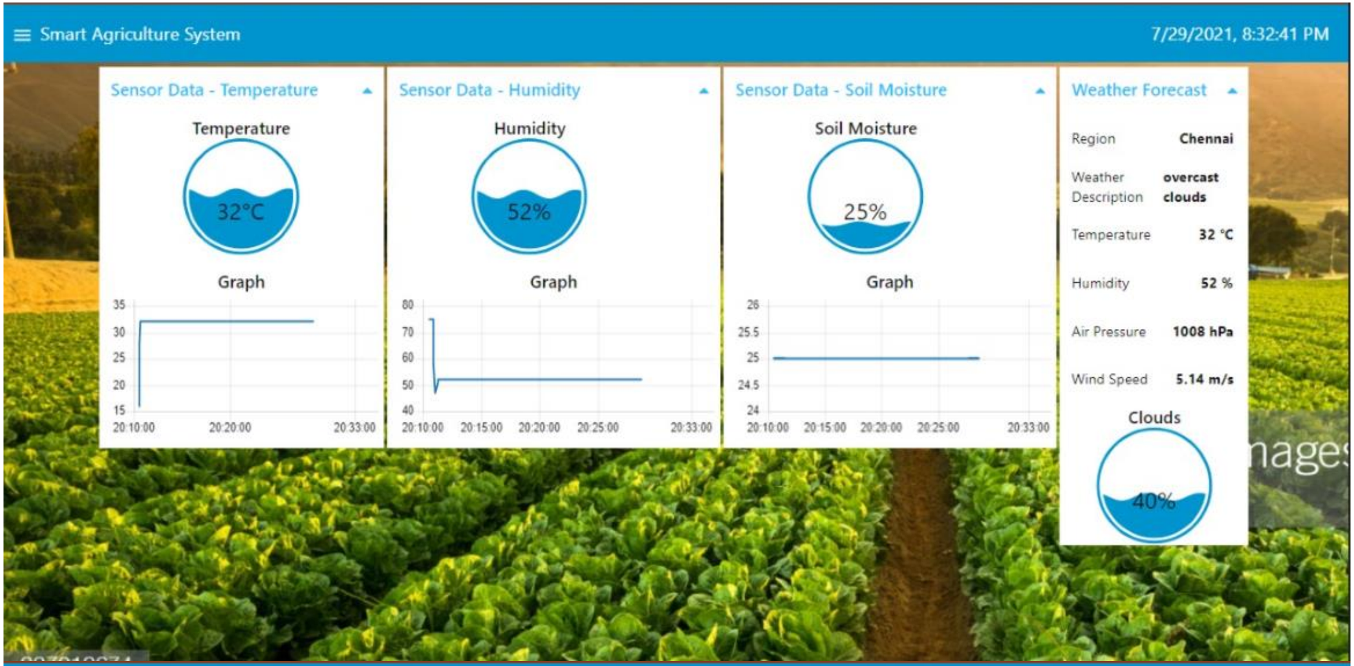


Task-5

Building a Web App

Milestone-1:

To display in UI :



Smart Agriculture System 7/29/2021, 8:32:41 PM

Sensor Data - Temperature

Temperature 32°C

Sensor Data - Humidity

Humidity 52%

Sensor Data - Soil Moisture

Soil Moisture 25%

Weather Forecast

Region Chennai

Weather Description overcast clouds

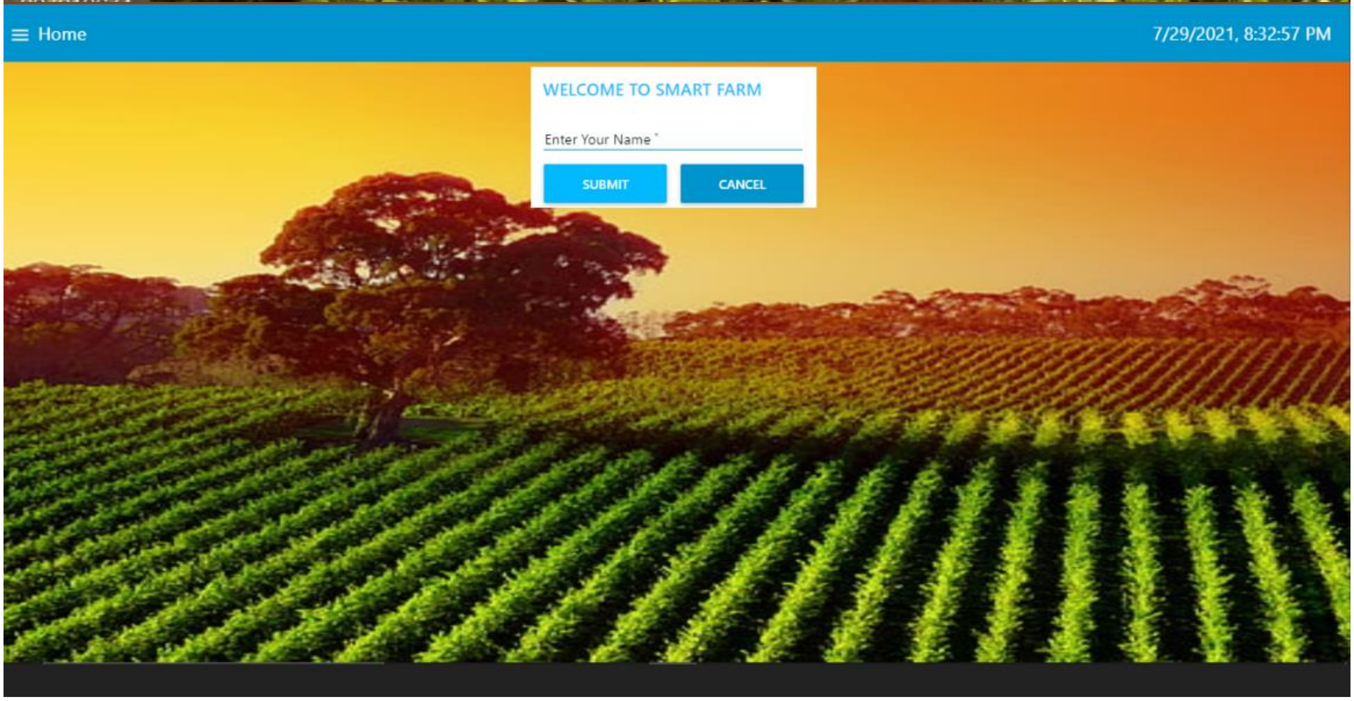
Temperature 32 °C

Humidity 52 %

Air Pressure 1008 hPa

Wind Speed 5.14 m/s

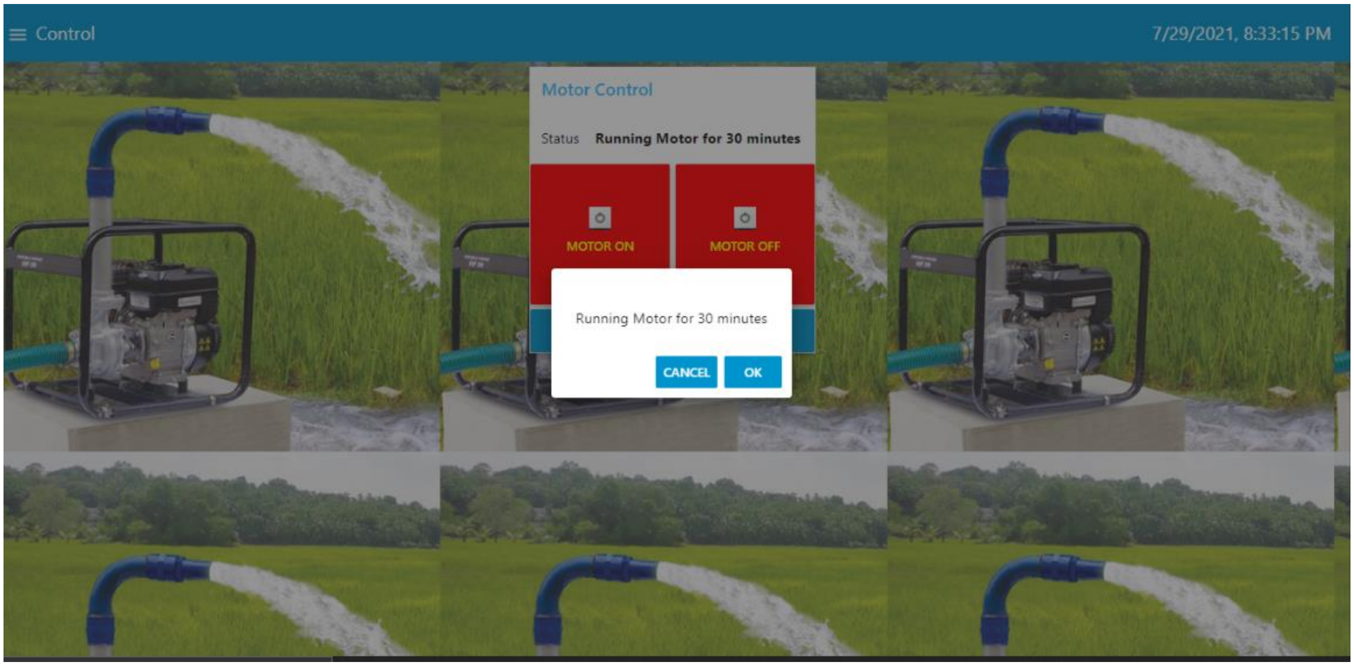
Clouds 40%



Home 7/29/2021, 8:32:57 PM

WELCOME TO SMART FARM

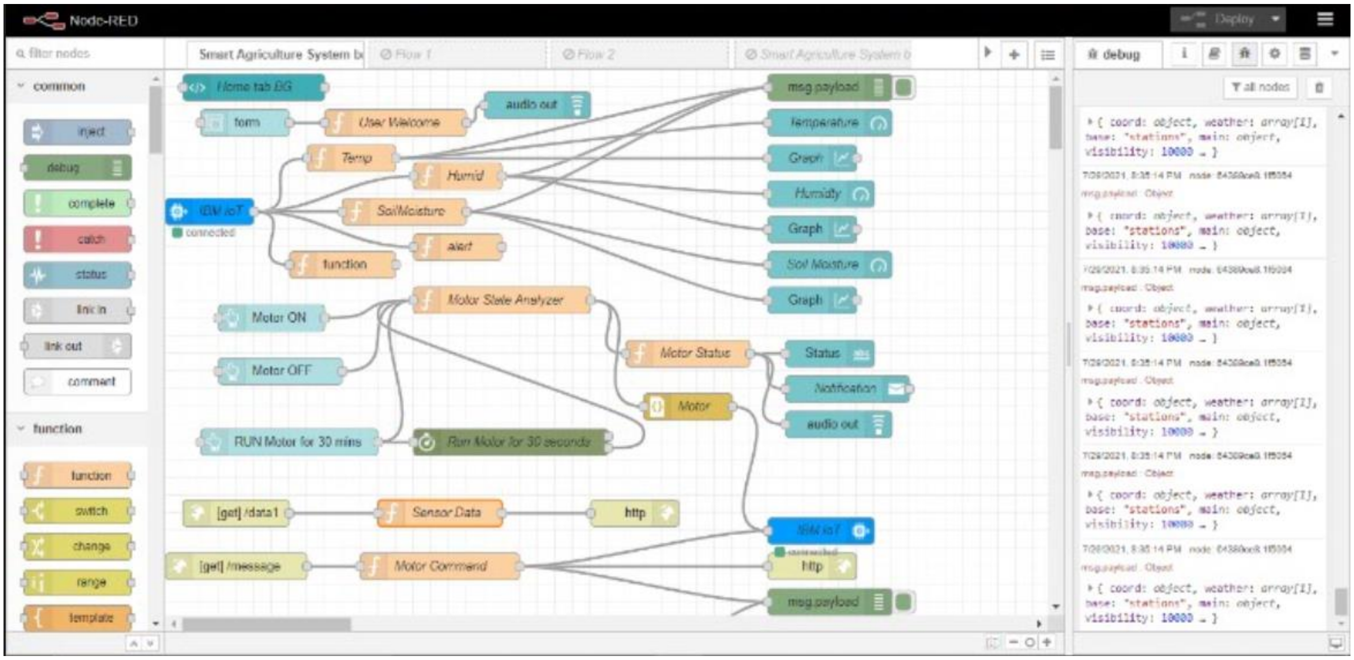
Enter Your Name *



Status **Running Motor for 30 minutes**

Running Motor for 30 minutes

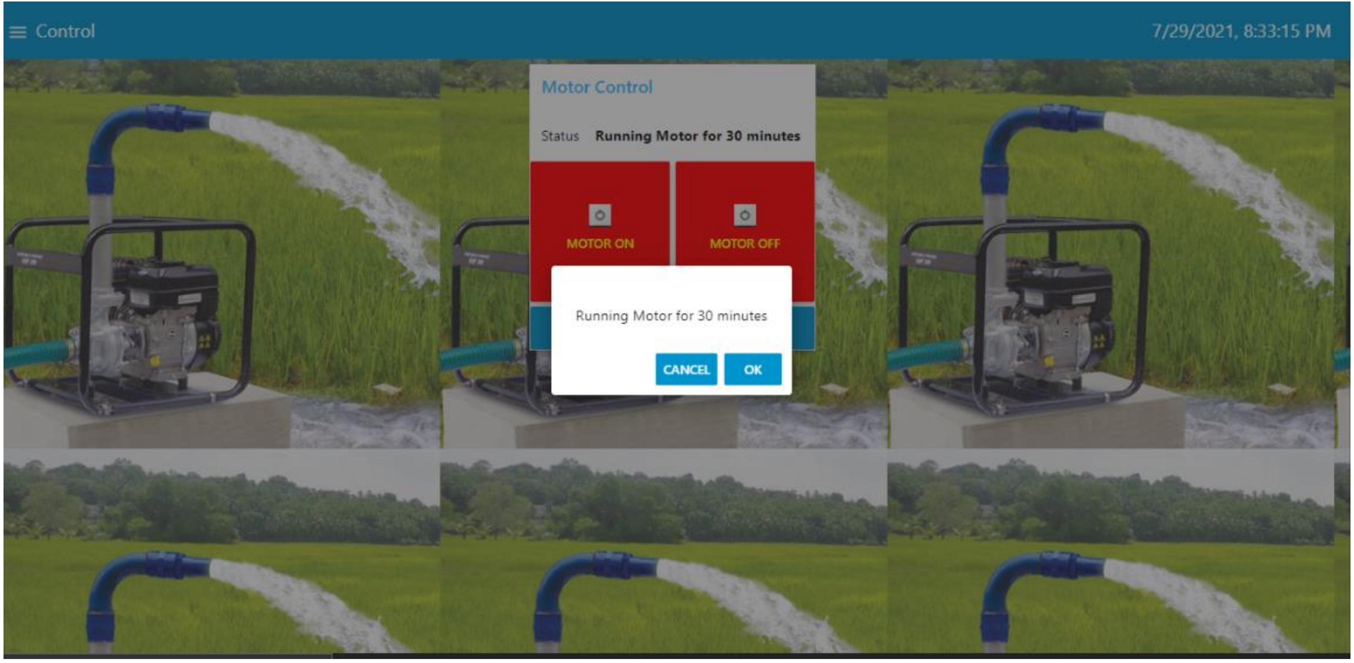
FLOW:



```
▸{ coord: object, weather: array[1],
base: "stations", main: object,
visibility: 10000 … }
7/29/2021, 8:35:14 PM  node: 64389ce8.1f5054
msg.payload : Object
▸{ coord: object, weather: array[1],
base: "stations", main: object,
visibility: 10000 … }
7/29/2021, 8:35:14 PM  node: 64389ce8.1f5054
msg.payload : Object
▸{ coord: object, weather: array[1],
base: "stations", main: object,
visibility: 10000 … }
7/29/2021, 8:35:14 PM  node: 64389ce8.1f5054
msg.payload : Object
▸{ coord: object, weather: array[1],
base: "stations", main: object,
visibility: 10000 … }
7/29/2021, 8:35:14 PM  node: 64389ce8.1f5054
msg.payload : Object
▸{ coord: object, weather: array[1],
base: "stations", main: object,
visibility: 10000 … }
7/29/2021, 8:35:14 PM  node: 64389ce8.1f5054
msg.payload : Object
▸{ coord: object, weather: array[1],
base: "stations", main: object,
visibility: 10000 … }
```

Milestone-2:

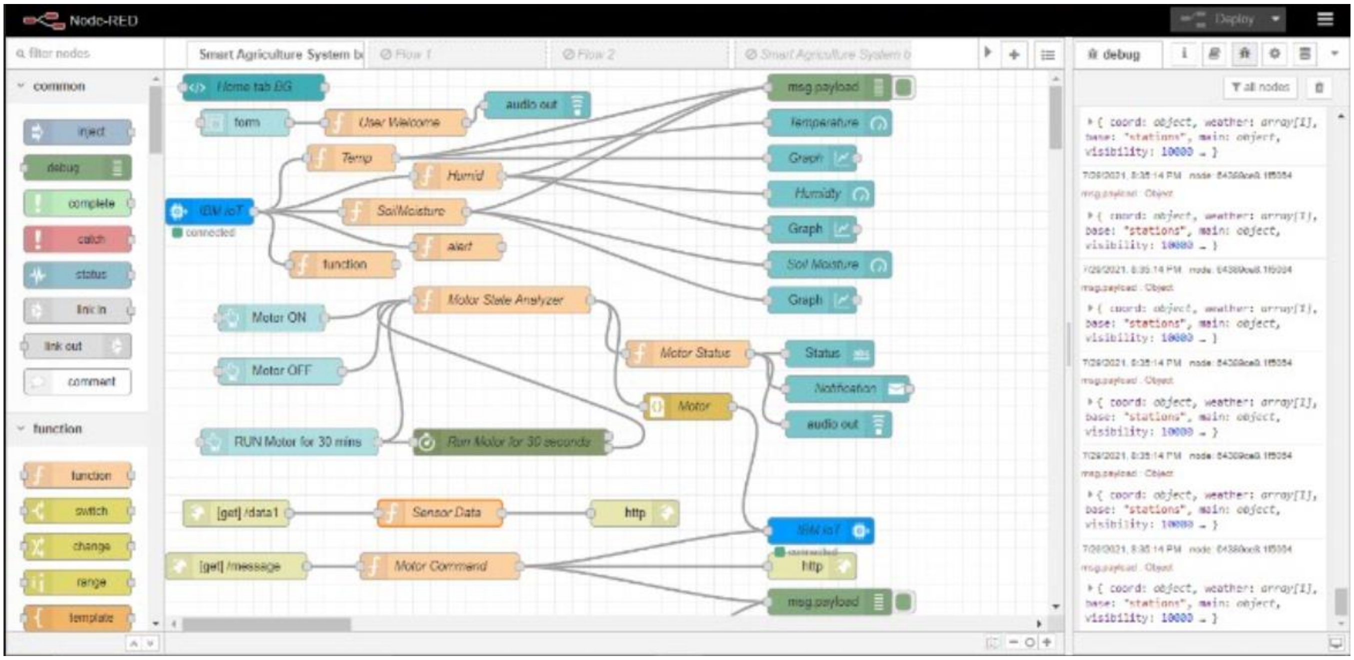
Configure the buttons:



Status Running Motor for 30 minutes

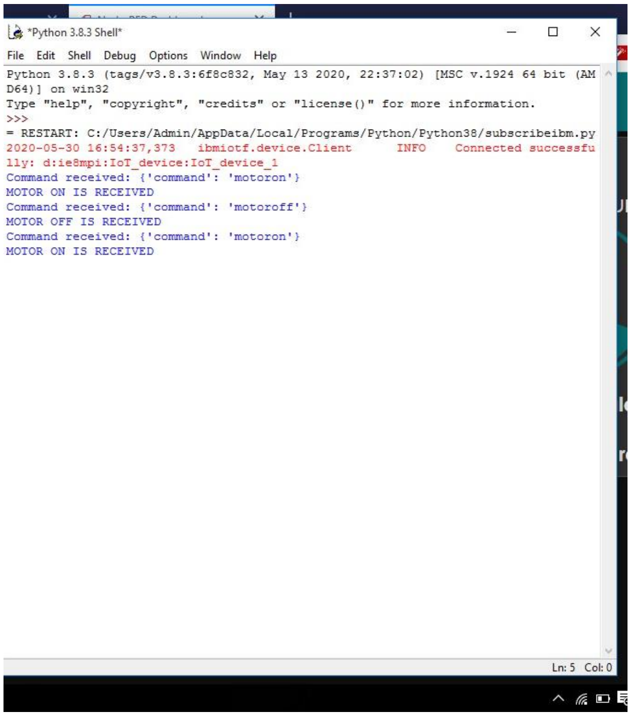
Running Motor for 30 minutes

Flow:



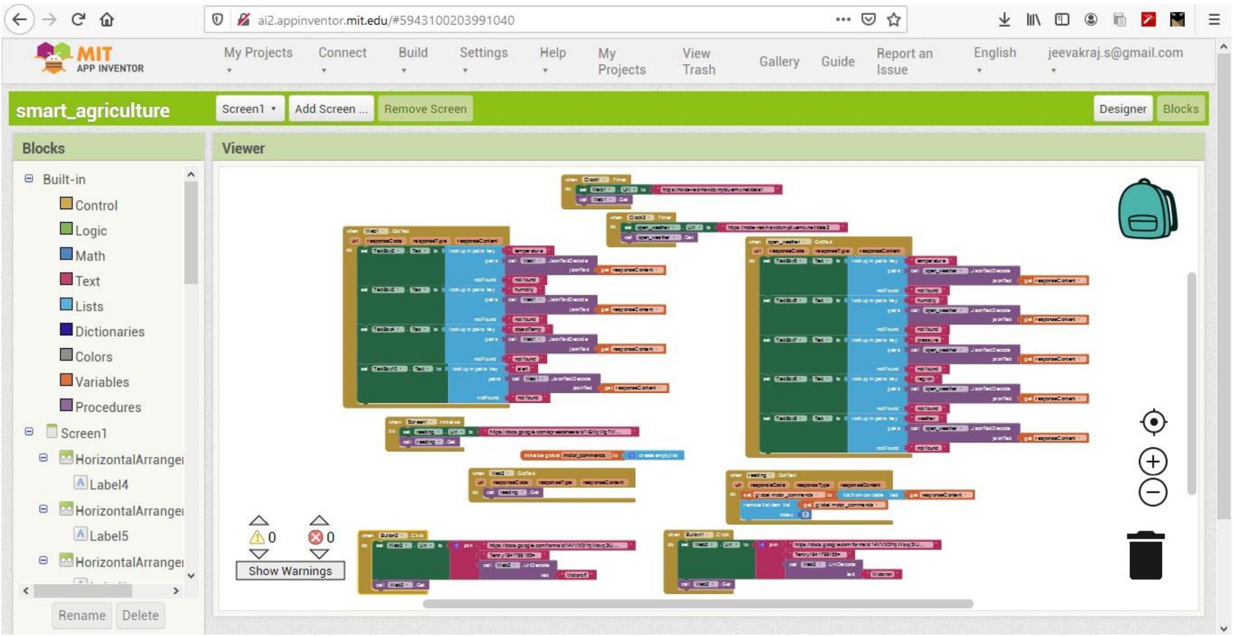
```
▸ { coord: object, weather: array[1], base: "stations", main: object, visibility: 10000 … }
7/29/2021, 8:35:14 PM node: 64309ce0.1f5054
msg.payload : Object
▸ { coord: object, weather: array[1], base: "stations", main: object, visibility: 10000 … }
7/29/2021, 8:35:14 PM node: 64309ce0.1f5054
msg.payload : Object
▸ { coord: object, weather: array[1], base: "stations", main: object, visibility: 10000 … }
7/29/2021, 8:35:14 PM node: 64309ce0.1f5054
msg.payload : Object
▸ { coord: object, weather: array[1], base: "stations", main: object, visibility: 10000 … }
7/29/2021, 8:35:14 PM node: 64309ce0.1f5054
msg.payload : Object
▸ { coord: object, weather: array[1], base: "stations", main: object, visibility: 10000 … }
7/29/2021, 8:35:14 PM node: 64309ce0.1f5054
msg.payload : Object
▸ { coord: object, weather: array[1], base: "stations", main: object, visibility: 10000 … }
```

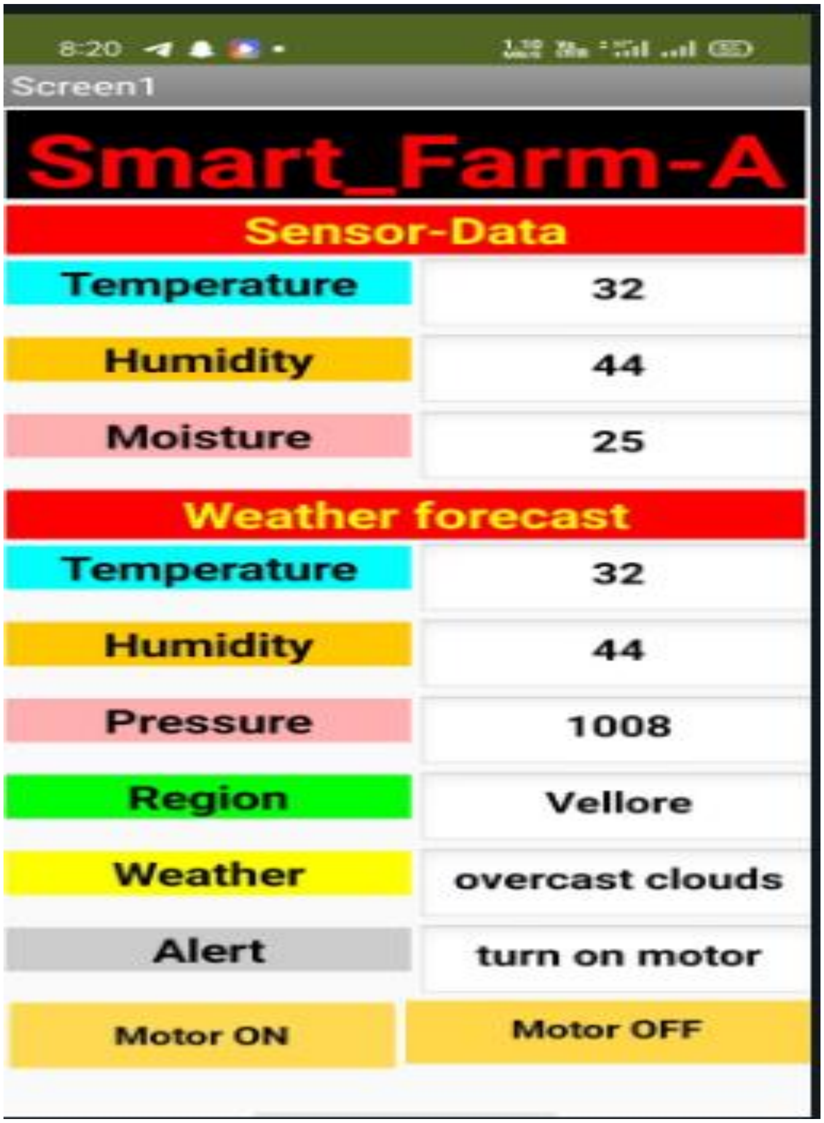
Motor ON/OFF in python:



```
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:37:02) [MSC v.1924 64 bit (AM
D64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Users/Admin/AppData/Local/Programs/Python/Python38/subscribeibm.py
2020-05-30 16:54:37,373   ibmiotf.device.Client      INFO    Connected successfu
lly: d:ie8mpi:IoT_device:IoT_device_1
Command received: {'command': 'motoron'}
MOTOR ON IS RECEIVED
Command received: {'command': 'motoroff'}
MOTOR OFF IS RECEIVED
Command received: {'command': 'motoron'}
MOTOR ON IS RECEIVED
```

MIT APP INVENTOR:





Smart_Farm-A

Sensor-Data

Temperature	32
Humidity	44
Moisture	25

Weather forecast

Temperature	32
Humidity	44
Pressure	1008
Region	Vellore
Weather	overcast clouds
Alert	turn on motor