PROJECT REPORT

SMART FARMING – AN IOT ENABLED SMART FARMING APPLICATION

TEAMID: PNT2022TMID43402

Team members:

Team lead – SRIDHARAN S N

Member 1 – VASANTH G

Member 2 – PRAWIN BALAJI K S

Member 3 – SRIRAM NAREN S

1. INTRODUCTION

1.1 Project Overview

IoT is bringing revolution to almost every aspect of our lives by changing how we do things. The use of Smart IoT devices is on the rise with all the industries heavily investing in IoT. The main aims of investing in IoT are to improve operations efficiency, improve product quality, and reduce the costs of production.

The Agricultural industry is among the industries seeking to reap the benefits of IoT.

Purpose

The use of IoT in agriculture is commonly referred to as Smart Farming or Smart Agriculture. It uses various IoT sensors to send the farm's data, like humidity, temperature, soil moisture, etc. to the cloud which can be monitored and controlled from anywhere in the world.

2. LITERATURE SURVEY

2.1 Existing problem

India is agriculture sector, on either side, is losing ground every day, affecting the ecosystem's output capacity. In order to restore vitality and put agriculture back on a path of higher growth, there is a growing need to resolve the issue. A large-scale agricultural system necessitates a great deal of upkeep, knowledge, and oversight. The IoT is a network of interconnected devices that can transmit and receive data over the internet and carry out tasks without human involvement. Agriculture provides a wealth of data analysis parameters, resulting in increased crop yields. The use of IoT devices in smart farming aids in the modernization of information and communication. For better crop growth moisture, mineral, light and other factors can be assumed. This research looks into a few of these characteristics for data analysis with the goal of assisting users in making better agricultural decisions using IoT. The technique is intended to help farmers increase their agricultural output.

1)Accidental deforestation

2) Soil erosion

3) High water usage

4) Energy wastage

5) Carbon emissions

6) Time consuming process

7) Poor outcomes of cultivation

8) Defense priority not customized to

Prevent intruding animals

2.2 References

[1] Dahane, A., Benameur, R., Kechar, B., & Benyamina, A. (2020, October). An IoT based smart farming system using machine learning. In 2020 International Symposium on Networks, Computers and Communications (ISNCC) (pp. 1-6). IEEE.

[2] Farooq, M. S., Riaz, S., Abid, A., Abid, K., & Naeem, M. A. (2019). A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming. Ieee Access, 7, 156237-156271.

[3] Farooq, M. S., Sohail, O. O., Abid, A., & Rasheed, S. (2022). A survey on the role of iot in agriculture for the implementation of smart livestock environment. IEEE Access, 10, 9483-9505.

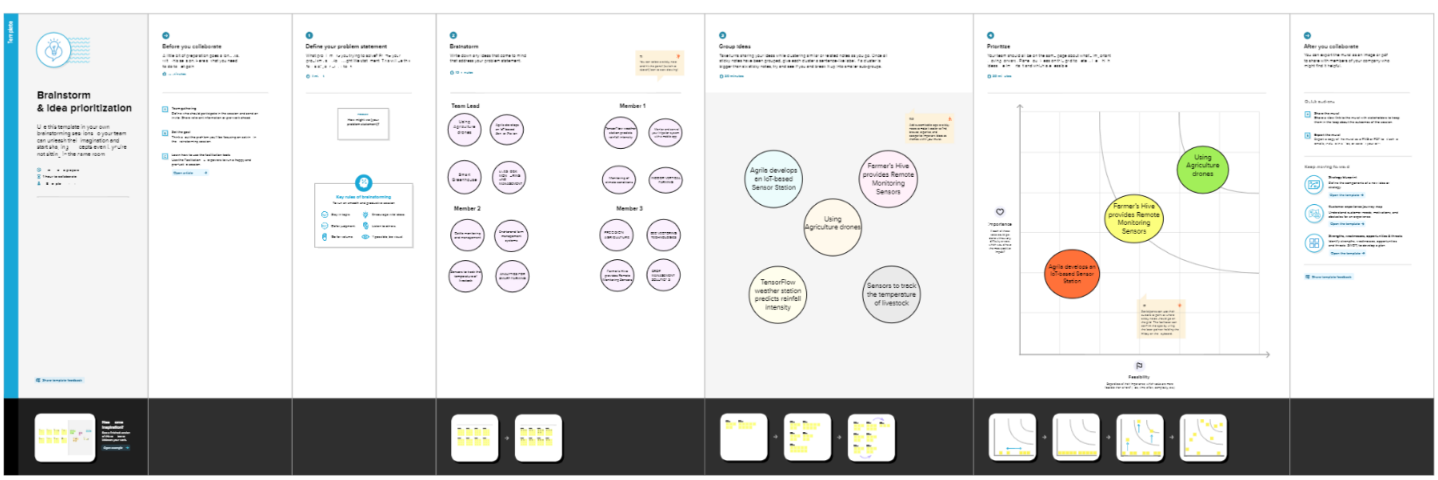
2.3 Problem Statement Definition

To provide efficient decision web using wireless sensor network which handle different activities of farm and provides useful information associated with farm. Information associated with Soil moisture, Temperature and Humidity content. Due to the atmospheric condition, water level increasing Farmers get lot of distractions which isn't good for Agriculture. Water level is managed by farmers in both Automatic/Manual using that mobile application. it'll make easier to farmers. Performing agriculture is incredibly much time consuming

3. IDEATION & PROPOSED SOLUTION

3.1 Ideation & Brainstorming

Ideation is the process where you generate ideas and solutions through sessions such as Sketching, Prototyping, Brainstorming, Brainwriting, Worst Possible Idea, and a wealth of other ideation techniques.



3.2 **Proposed Solution**

3.3 Problem Statement

To make farming simpler by identifying a few agricultural restrictions, overcoming those limits, and leveraging IOT to boost production quality and quantity.

Idea / Solution description

BY using smart agriculture techniques like monitoring farms climate, temperature, humidity, smart irrigation and soil analysis.

Novelty

Smart irrigation powered by solar allows you to use intelligent sensors to track temperature, moisture, and humidity.

Social Impact / Customer Satisfaction

By employing this technology, we can control soil erosion better than the current irrigation system that is available today. Better output yield will result.

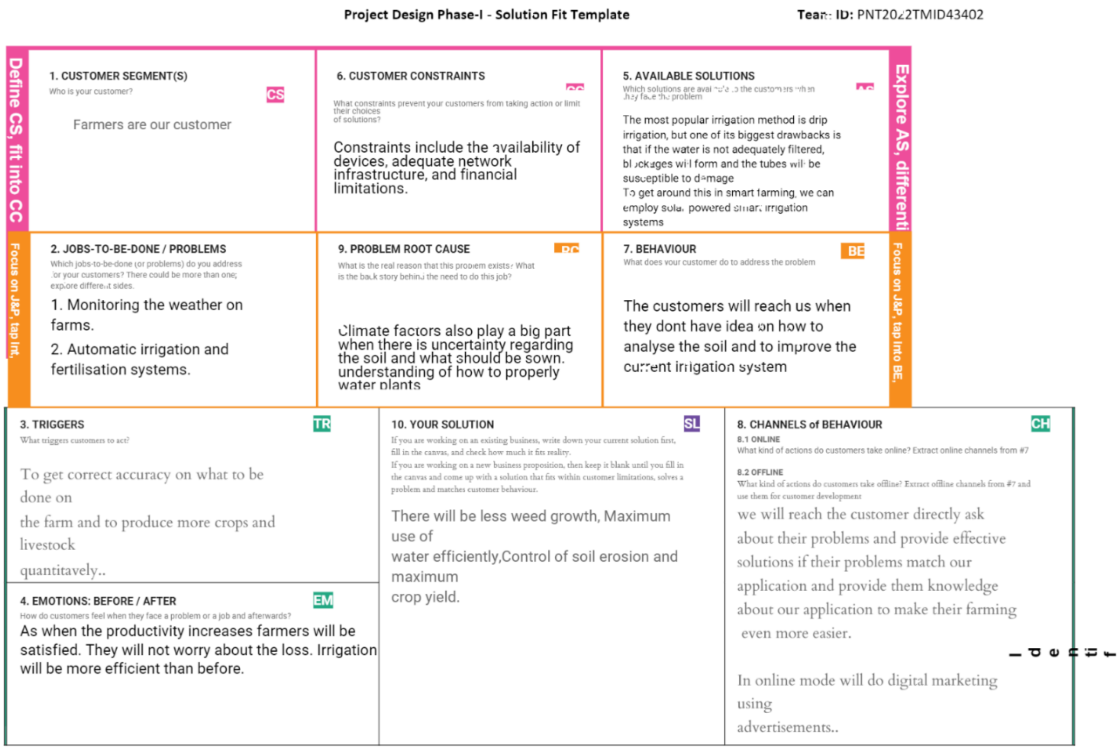
Business Model (Revenue Model)

As productivity rises, customer happiness rises as well, driving up demand for the application and, ultimately, the business’s income.

Scalability of the Solution

It is unquestionably scalable, and we may tighten the restrictions if an issue occurs.

3.4 <u>Problem Solution fit</u>



4. REQUIREMENT ANALYSIS

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Log in to system	Check Credentials Check Roles of Access.

FR-4	Manage Modules	Manage System Admins Manage Roles of User Manage User permission
FR-5	Check whether details	Temperature details Humidity details
FR-6	Log out	Exit

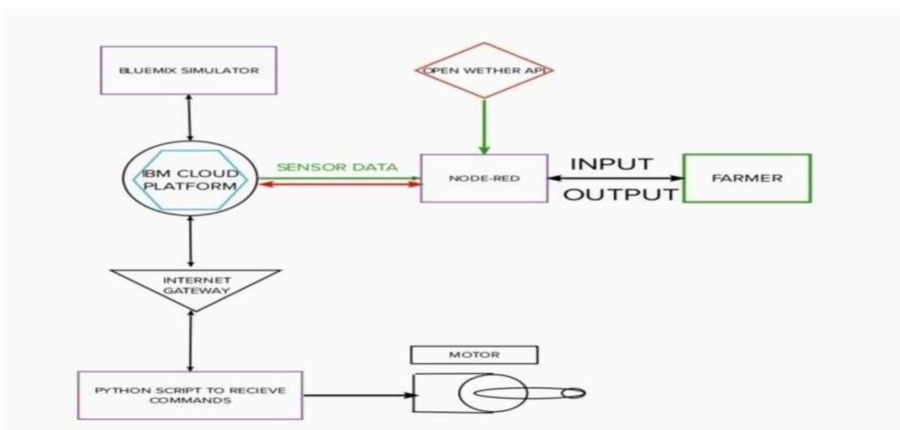
Non-functional Requirements:

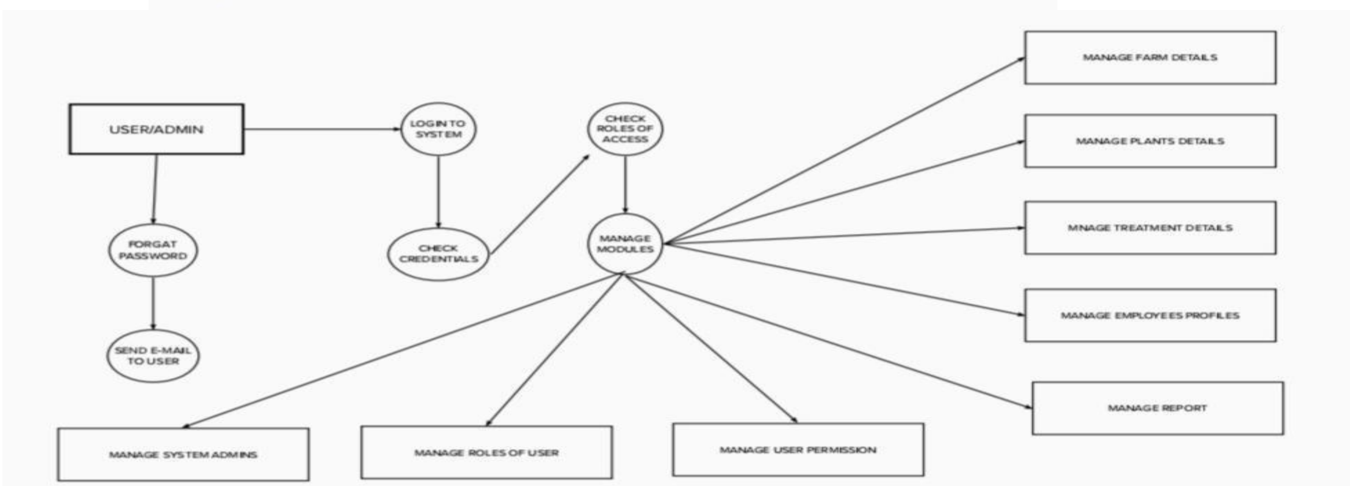
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Usability includes easy learn ability, efficiency in use, remember ability, lack of errors in operation and subjective pleasure.
NFR-2	Security	Sensitive and private data must be protected from their production until the decision-making and storage stages.
NFR-3	Reliability	The shared protection achieves a better trade-off between costs and reliability.
NFR-4	Performance	the idea of implementing integrated sensors with sensing soil and environmental or ambient parameters in farming will be more efficient for overall monitoring.
NFR-5	Availability	Automatic adjustment of farming equipment made possible by linking information like crops/weather and equipment to auto-adjust temperature, humidity, etc.
NFR-6	Scalability	Scalability is a major concern for IoT platforms. It has shown that different architectural choices of IoT platforms affect system scalability and that automatic real time decision-making is feasible in an environment composed of dozens of thousand.

5. PROJECT DESIGN

5.1 Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.





DFD 0 FLOW SMART FAMING APPLICATION USING IOT

- The different soil parameters temperature, soil moistures and then humidity are sensed using different sensors and obtained value is stored in the IBM cloud.

- Arduino UNO is used as a processing Unit that process the data obtained from the sensors and whether data from the weather API.

- NODE-RED is used as a programming tool to write the hardware, software, and APIs. The MQTT protocol is followed for the communication.

- All the collected data are provided to the user through a mobile application that was developed using the MIT app inventor. The user could plan through an app, weather to water the crop or not depending upon the sensor values. By using the app they can remotely operate to the motor switch.

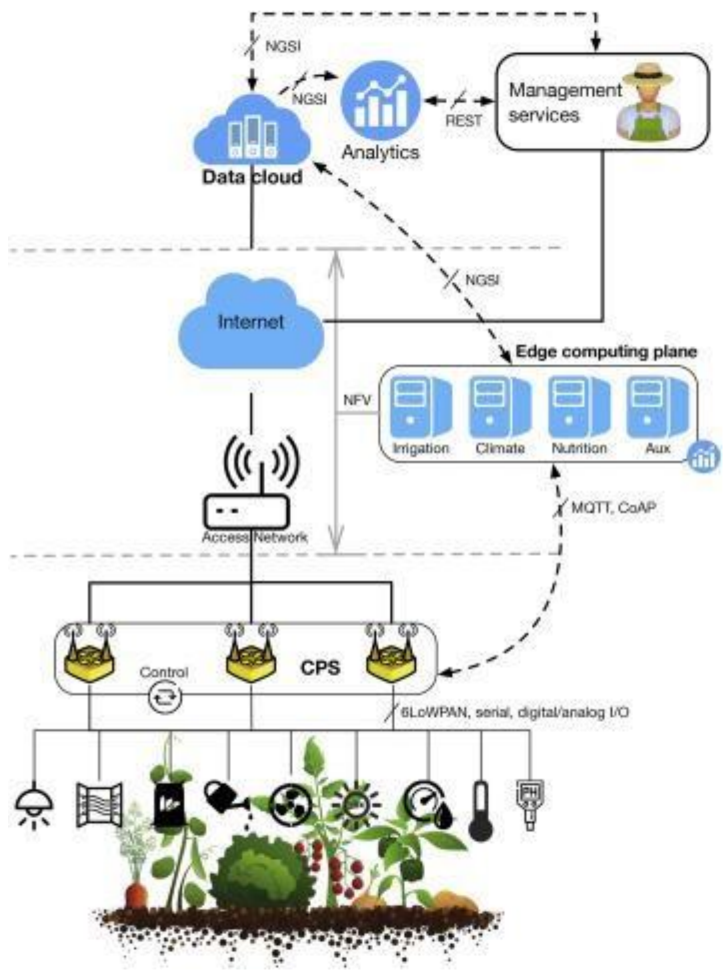
5.2 Solution & Technical Architecture:

Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Discover the finest technological solution to address current company issues..
- Describe the software's design, features, functionality, and other elements to the project's stakeholders..
- Specify the features, stages of development, and requirements for the solution.
- Offer guidelines for how the solution is created, managed, and delivered.

Example - Solution Architecture Diagram:



5. PROJECT PLANNING & SCHEDULING

5.1 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Creation of IBM cloud	US-1	Create the IBM Cloud services which are being used in this project.	6	High	Prawin Balaji K S Sridharan S N,
Sprint-1		US-2	Configure the IBM Cloud services which are being used in completing this project.	4	Medium	Sriram Naren S Vasanth G
Sprint-1		US-3	IBM Watson IoT platform acts as the mediator to connect the web application to IoT devices, so create the IBM Watson IoT platform.	5	Medium	Sridharan S N, Prawin Balaji K S
Sprint-1		US-4	In order to connect the IoT device to the IBM cloud, create a device in the IBM Watson IoT platform and get the device credentials.	5	High	Sriram Naren S Vasanth G
Sprint-2	Creation of Node-RED	US-1	Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.	10	High	Sridharan S N Prawin Balaji K S

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2		US-2	Create a Node-RED service.	10	Medium	Vasanth G Sriram Naren S
Sprint-3	Developing python script	US-1	Develop a python script to publish random sensor data such as temperature, moisture, soil and humidity to the IBM IoT platform	7	High	Sridharan S N Prawin Balaji KS
Sprint-3		US-2	After developing python code, commands are received just print the statements which represent the control of the devices.	5	Medium	Vasanth G Sriram Naren S
Sprint-3		US-3	Publish Data to The IBM Cloud	8	High	Prawin Balaji K S Vasanth G Sridharan S N Sriram Naren S
Sprint-4	Publish data to the cloud	US-1	Create Web UI in Node-Red	10	High	Prawin Balaji K S Vasanth G Sridharan S N Sriram Naren S
Sprint-4		US-2	Configure the Node-RED flow to receive data from the IBM IoT platform and also use Cloudant DB nodes to store the received sensor data in the cloudant DB	10	Medium	Prawin Balaji K S Vasanth G Sridharan S N Sriram Naren S

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV = sprint duration/Velocity = 20/6 = 3.33

6. **CODING & SOLUTIONING :**

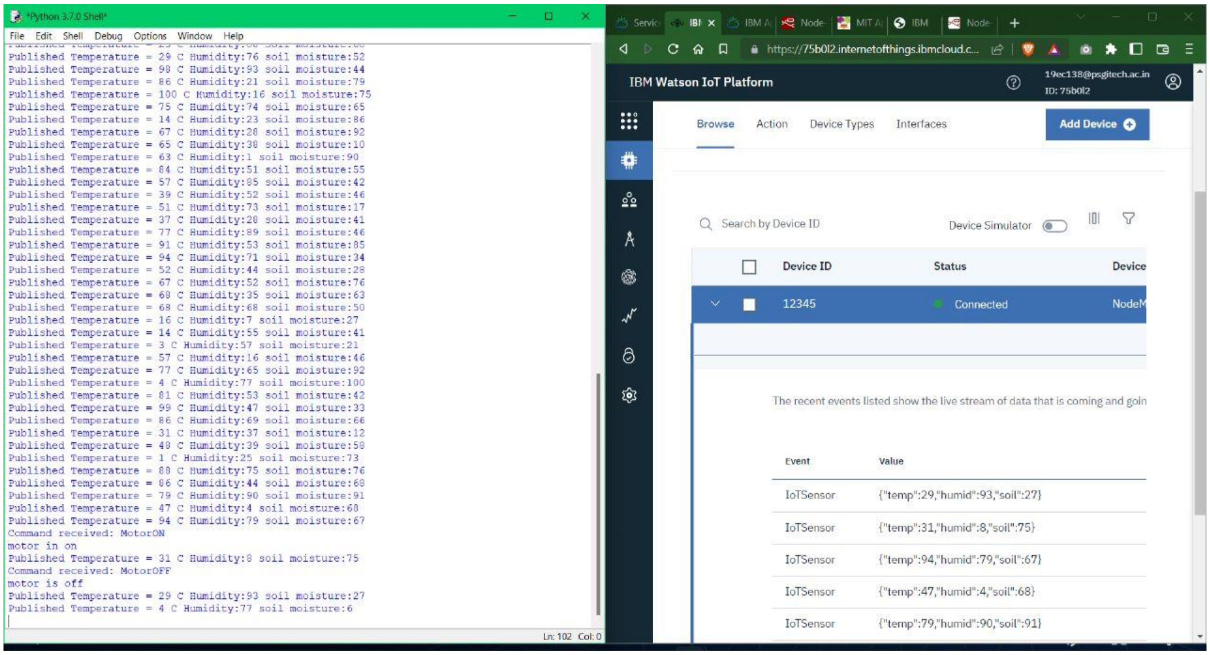
Python Code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "75b0l2"
deviceType = "NodeMCU"
deviceId = "12345"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status =="MotorON":
        print("motor in on")
    else :
        print ("motor is off")

try:
     deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
```

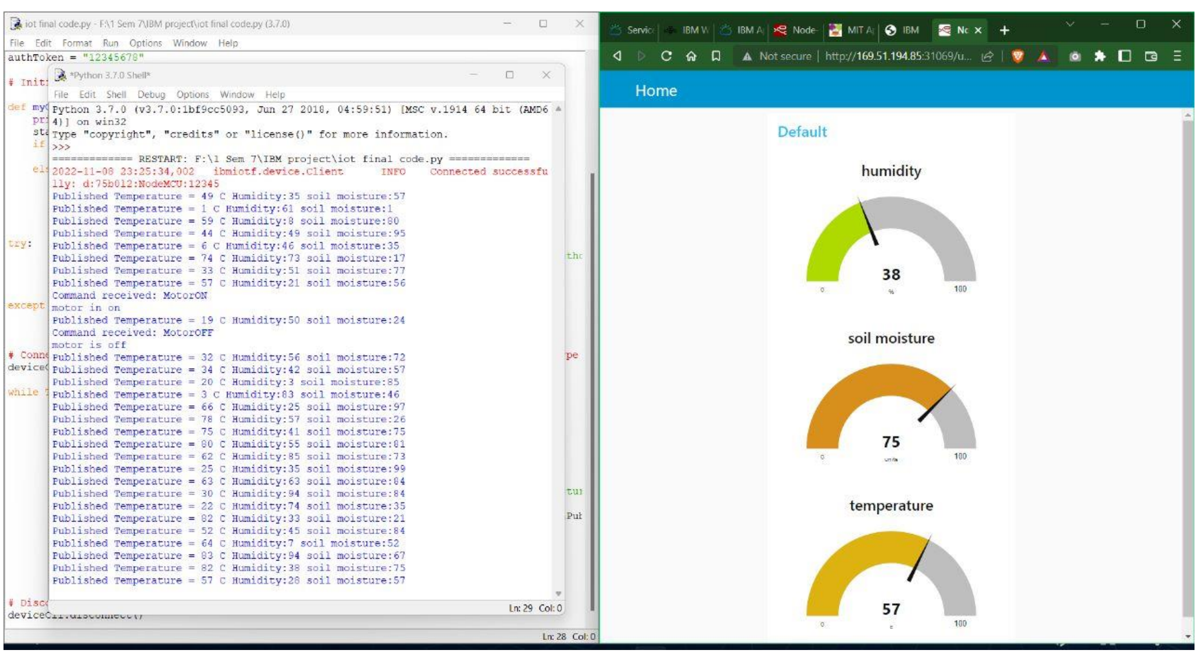
```
        deviceCli = ibmiotf.device.Client(deviceOptions)

    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:
            #Get Sensor Data from DHT11
            temp=random.randint(0,100)
            humid=random.randint(0,100)
            soil=random.randint(0,100)
            data = {'temp' : temp, 'humid' : humid, 'soil' : soil}
            #print data
            def myOnPublishCallback():
                print ("Published Temperature = %s C" % temp, "Humidity:%s" %humid, "soil
moisture:%s" %soil)
            success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
            if not success:
                print("Not connected to IoTF")
            time.sleep(1)
            deviceCli.commandCallback = myCommandCallback

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

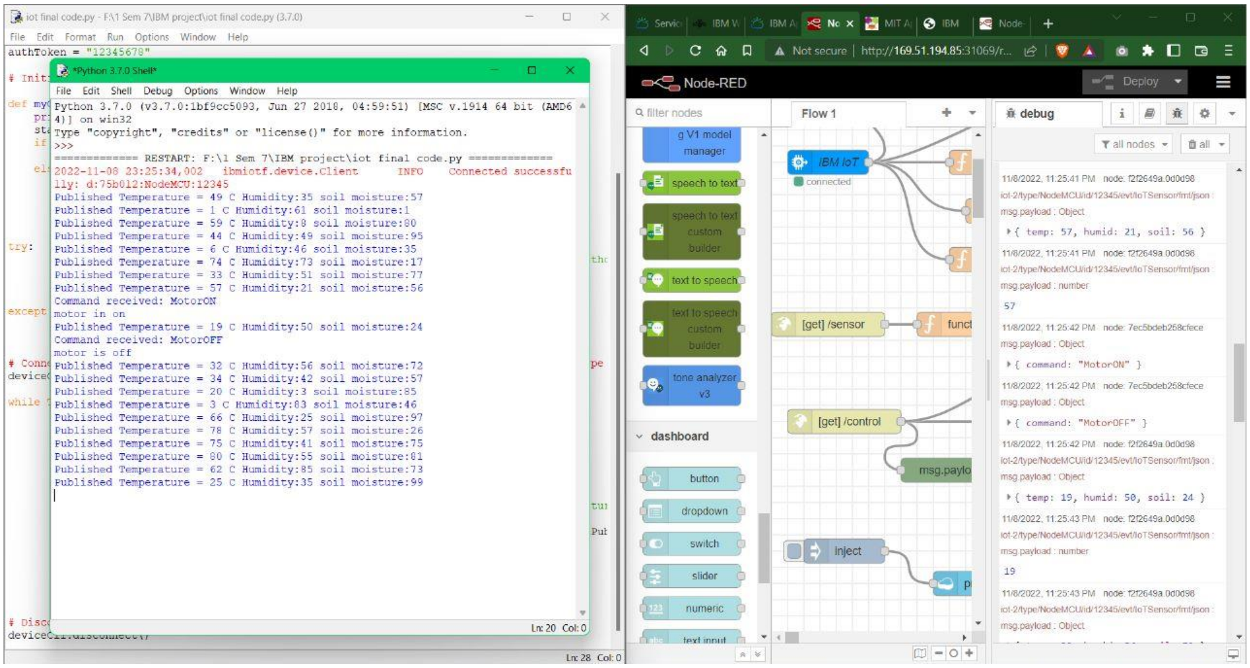
7. RESULTS

Sending the sensor values to the cloud:

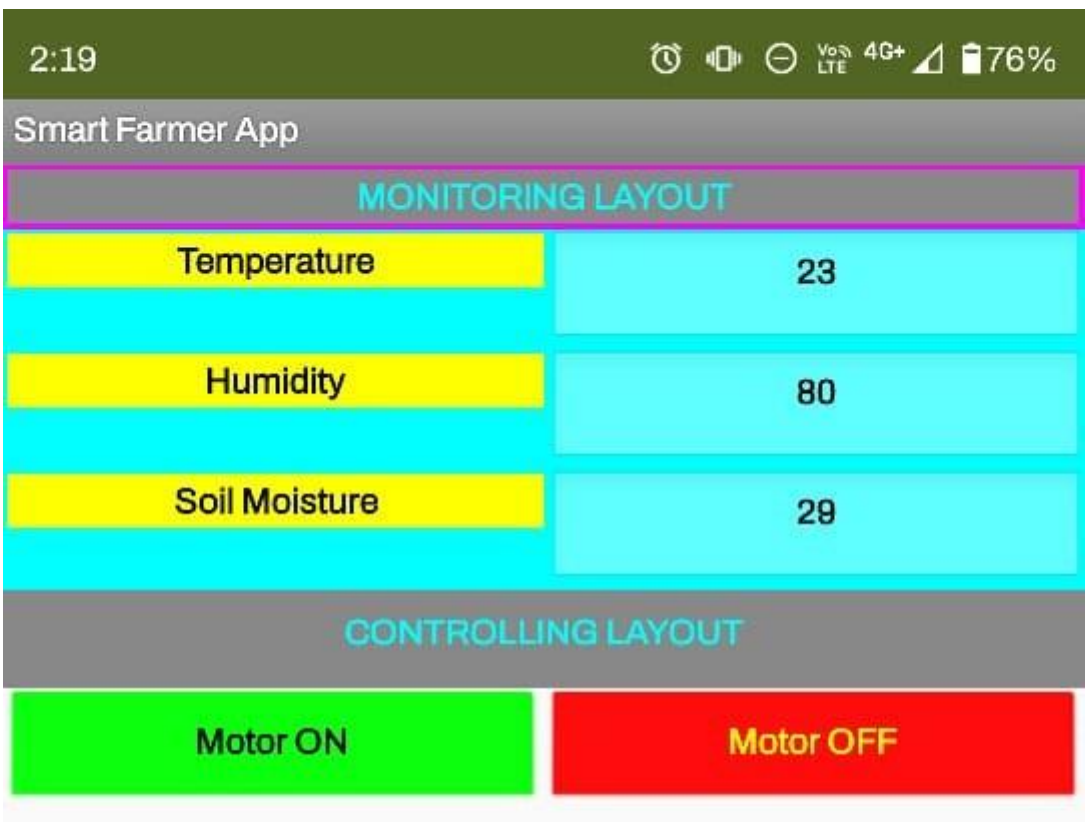


From cloud to Node-RED and UI:





From Node-RED to mobile application:



8. ADVANTAGES & DISADVANTAGES

Advantages:

• Farms can be monitored and controlled remotely.

• Increase in convenience to farmers.

• Less labor cost.

• Better standards of living.

Disadvantages:

• Lack of internet/connectivity issues.

• Added cost of internet and internet gateway infrastructure.

• Farmers wanted to adapt the use of Mobile App.

9. CONCLUSION

Thus the smart agriculture using IoT will revolutionized the world of farming and it will increase the productivity as well as improve the quality and can save lives of farmer. There is an urgent need for a system that makes the agricultural process easier and burden free from the farmer’s side. With the recent advancement of technology it has become necessary to increase the annual crop production output of our country India, an entirely agro centric economy. The ability to conserve the natural resources as well as giving a splendid boost to the production of the crops is one of the main aims of incorporating such technology into the agricultural domain of the country. To save farmer’s effort, water and time has been the most important consideration

10.FUTURE SCOPE

The prospects are high that smart farming will change agriculture in a great way. Smart farming is expected to bridge the gap between large and small-scale farmers in both developing and developed countries. Technological advancement, growth in the internet of things, and the introduction of smartphones have contributed immensely to the adoption of technology in agriculture. Different countries understand the worth of these technologies, which explains why most countries are eager to promote the implementation of precision farming techniques.

There is no doubt that most agricultural operations that were practiced traditionally have changed significantly nowadays. This can be attributed to technological advancement the adoption of smart farming techniques and methodologies such as the use of machines, devices, sensors, and information technology. Presently, farmers make use of sophisticated technologies like aerial images, moisture and temperature sensors, GPS technology and robots. Such technology makes farming not only to be a profitable venture but also an environmentally friendly, safer, and efficient.

11.APPENDIX

Source Code

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "75b0l2"
deviceType = "NodeMCU"
```

```
deviceId = "12345"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status =="MotorON":
        print("motor in on")
    else :
        print ("motor is off")

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:
```

```
        #Get Sensor Data from DHT11
        temp=random.randint(0,100)
        humid=random.randint(0,100)
        soil=random.randint(0,100)
        data = {'temp' : temp, 'humid' : humid, 'soil' : soil}
        #print data
        def myOnPublishCallback():
          print ("Published Temperature = %s C" % temp, "Humidity:%s" %humid, "soil moisture:%s" %soil)

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
        if not success:
          print("Not connected to IoTF")
        time.sleep(1)
        deviceCli.commandCallback = myCommandCallback

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

GitHub & Project Demo Link:

GitHub - https://github.com/IBM-EPBL/IBM-Project-42595-1660669060

Project demo link - https://drive.google.com/file/d/1QrpWpYW_Y7xXhnIKYn-a8_zx-IDkCset/view?usp=share_link