Project Name:smart farmer-IOT based Smart Farming Application

Team ID:PNT2022MID46860

Team leader:S.karthickraj

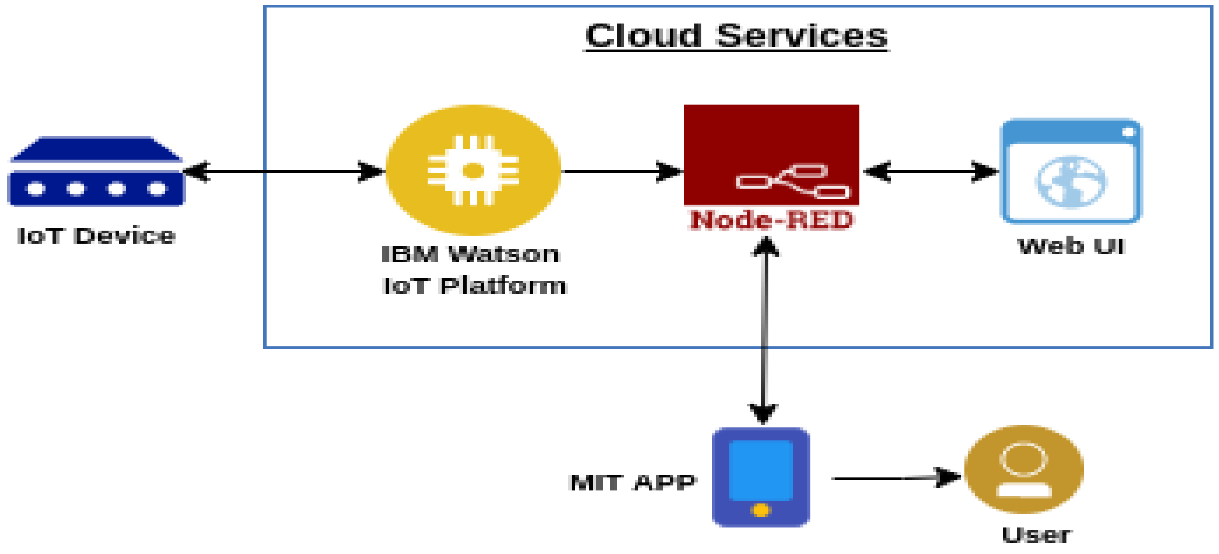
Team members:

A.Arunnathan
W.Deepak evlin
T.Dharun
G.Sivanatham

1.INTRODUCTION

1.1 Project Overview

IoT-based agriculture system helps the farmer in monitoring different parameters of his fieldlike soil moisture, Temperature, humidity using some sensors. Farmers can monitor all the sensorparameters by using a web or mobile application even if the farmer is not near his field. Wateringthe crop is one of the important tasks for the farmers. They can make the decision whether to water the crop or postpone it by monitoring the sensor parameters and controlling the motor pumps from the mobile application itself.



1.2 Purpose

The smart agriculture model main aim **to avoid water wastage in the irrigation process**. It is low cost and efficient system Is shown below. It includes NodeMCU, Arduino Nano, sensors like soil moisture and Dht11, solenoid valves, relays.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

The challenges of a smart agriculture system include the integration of these sensors and tyingthe sensor data to the analytics driving automation and response activities. When integrated,the use of data analytics can reduce the overall cost of agriculture and contribute to higher production from the same amount of area through precise control of water, fertilizer and light. Smart methods allow for farming on smaller and more distributed lands through remote monitoring, whether indoor or outdoor.

To successfully deploy a smart agriculture system, consider setting up a communications network that can integrate a limited number of sensors across a large area of farmland. This will require third-party network provisioning or setting up a private network consisting of access points and uplinks to a private backhaul network, which channels all the data traffic to centralized monitoring software or an analytics head-end system

- It is not a secure system.
- There is no motion detection for protection of agriculture field.
- Automation is not available.

2.2 REFERENCES

[1] ISSN No:-2456-2165 Volume 4, Issue 2 Feb – 2019: "Solars' Energy: - A safe and reliable, eco-friendly and sustainable Clean Energy Option for Future India: - A Review."

[2] Universal Paper of advanced science and science and exploration technology. [2] GRD Journals- Global Research and Development Journal for Engineering | Volume 4 | Issue 3 | February (2019) ISSN: 2455-5703 “Design and Implementation of an Advanced Security Systemfor Farm Protection from Wild Animals”.

[3] International Journal of Innovationsin Engineering and Science, Impact Factor Value 4.046 e-ISSN: 2456-3463 Vol.4, No. 5, 2019 “Solar Powered Smart Fencing System for Agriculture Protection using GSM & Wireless Camera”.

[4] International Journal of Management, Technology And Engineering ISSN NO : 2249-7455 Volume 8, Issue VII, JULY/2018”Protecting Crops From Birds, Using Sound Technology In Agriculture” [5] American Journal of Engineering Research (AJER)2018 eISSN: 2320-0847 p- ISSN : 2320- 0936 Volume-7, Issue-7, pp-326-330 “Moisture Sensing Automatic Plant WateringSystem Using Arduino Uno”.

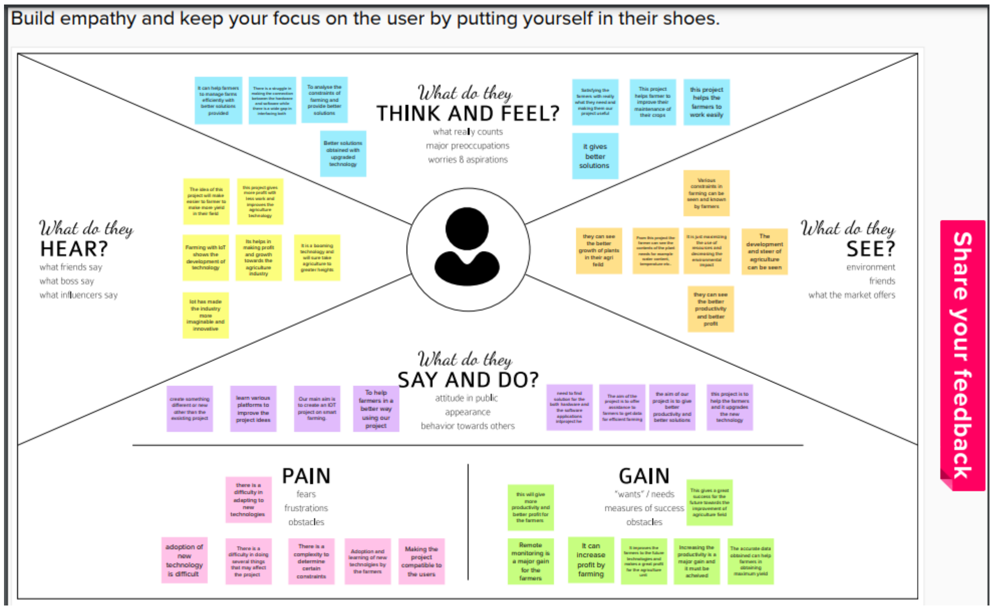
2.3 PROBLEM STATEMENT DEFINITION

The soil moisture sensor measures wetness content in the soil. The Arduino UNO microcontroller used to receive input from a various sensors and it can be controlled automatically. When soil moisture sensor goes low the water pump will be on and it exceeds defined levels of the water motor will turn off automatically. We can constantly monitor the growth of a crop using ultrasonic sensor. PIR sensor detects the motion or unusual movement in the agricultural land. This device his very helpful to the former to monitor and control environmental parameters at their field. The farmers did not go to theirfield, they can remotely monitor and control using cloud.

3.IDEATION & PROPOSED SOLUTION

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. Ideation is also the third stage in the Design Thinking process.

3.1 EMPATHY MAP CANVAS



Build empathy and keep your focus on the user by putting yourself in their shoes.

It can help farmers to manage farms efficiently with better solutions provided

There is a struggle in making the connection between the hardware and software while there is a wide gap in interfacing both

To analyse the constraints of farming and provide better solutions

Better solutions obtained with upgraded technology

Satisfying the farmers with really what they need and making them our project useful

This project helps farmer to improve their maintenance of their crops

this project helps the farmers to work easily

It gives better solutions

The idea of this project will make easier to farmer to make more yield in their field

this project gives more profit with less work and improves the agriculture technology

Farming with IoT shows the development of technology

Its helps in making profit and growth towards the agriculture industry

It is a booming technology and will sure take agriculture to greater heights

Iot has made the industry more imaginable and innovative

Various constraints in farming can be seen and known by farmers

they can see the better growth of plants in their agri feild

From this project the farmer can see the contents of the plant needs for example water content, temperature etc..

It is just maximizing the use of resources and decreasing the environmental impact

The development and steer of agriculture can be seen

they can see the better productivity and better profit

create something different or new other than the existing project

learn various platforms to improve the project ideas

Our main aim is to create an IOT project on smart farming.

To help farmers in a better way using our project

need to find solution for the both hardware and the software applications inlproject he

The aim of the project is to offer assistance to farmers to get data for efficient farming

the aim of our project is to give better productivity and better solutions

this project is to help the farmers and it upgrades the new technology

there is a difficulty in adapting to new technologies

adoption of new technology is difficult

There is a difficulty in doing several things that may affect the project

There is a complexity to determine certain constraints

Adoption and learning of new technolgies by the farmers

Making the project compatible to the users

this will give more productivity and better profit for the farmers

This gives a great success for the future towards the improvement of agriculture field

Remote monitoring is a major gain for the farmers

It can increase profit by farming

It improves the farmers to the future technologies and makes a great profit for the agriculture unit

Increasing the productivity is a major gain and it must be acheived

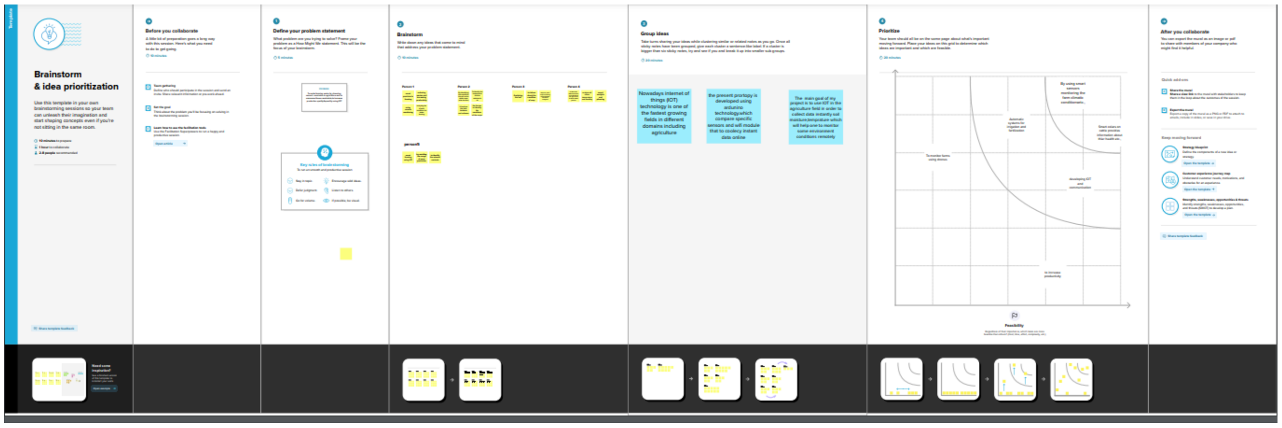
The accurate data obtained can help farmers in obtaining maximum yield

3.2 IDEATION AND BRAINSTORMING

Introduction on Internet of Things (IoT), application of IoT in agricultural field to improve the yield and quality by reducing the cost is provided. The sensors which are used in the architecture are discussed briefly and the process of transmission of data from the agriculture field to the central system is explained. The proposed system advantages are included. In addition, open research issues, challenges, and future of IoT in agricultural field are highlighted. The concept is basically developed on an idea, where there are numerous things or objects - such as Arduino, sensors, GSM models, LCD display, etc., that are connected with the Internet. Each of the objects has a different address and is able to interact with other items. The things or objects co-operate with each other to reach a common goal.

We are going to construct a smart agricultural monitoring system which can collect crucial agricultural data and send it to an IoT platform called Thingspeak in real time where the data can be logged and analyzed. The logged data on Thingspeak is in graphical format, a botanistor a reasonably knowledged farmer can analyze the data (from anywhere in the world) to make sensible changes in the supplied resources (to crops) to obtain high quality yield.Smart agriculture monitoring system or simply smart farming is an emerging technology concept where data from several agricultural fields ranging from small to large scale and its surrounding are collected using smart electronic sensors. The collected data are analyzed by experts and local farmers to draw short term and long-term conclusion on weather pattern, soil fertility, current quality of crops, amount of water that will be required for next week to amonth etc.

We can take smart farming a step further by automating several parts of farming, for example smart irrigation and water management. We can apply predictive algorithms on microcontrollers or SoC to calculate the amount of water that will be required today for a particular agriculture field. Say, if there was rain yesterday and the quantity of water requiredtoday is going to be less. Similarly, if humidity was high the evaporation of water at upper ground level is going to be less, so water required will be less than normal, thus reducing water usage.


Template
Brainstorm & idea prioritization
Before you collaborate
Define your problem statement
Key rules of brainstorming
Brainstorm
Group ideas
Prioritize
After you collaborate

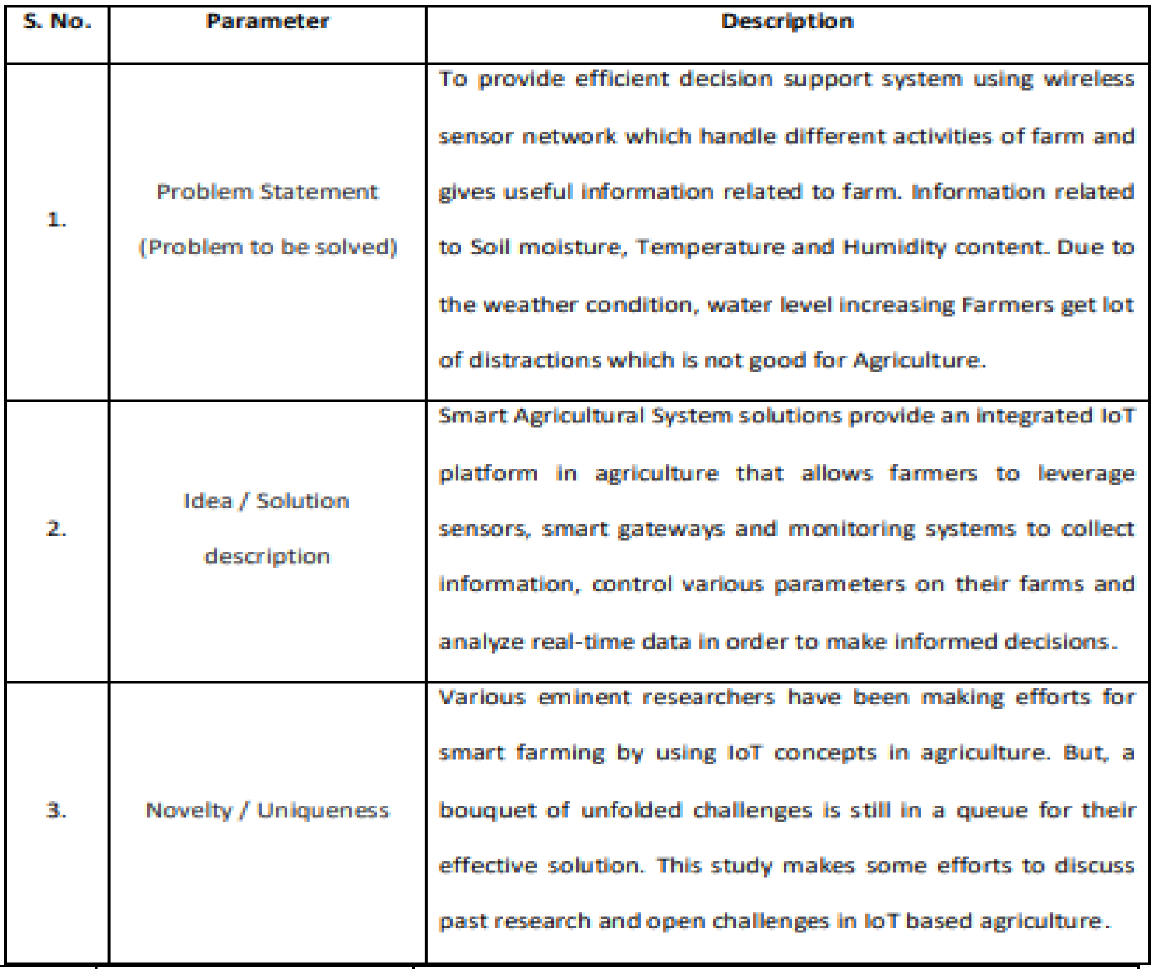
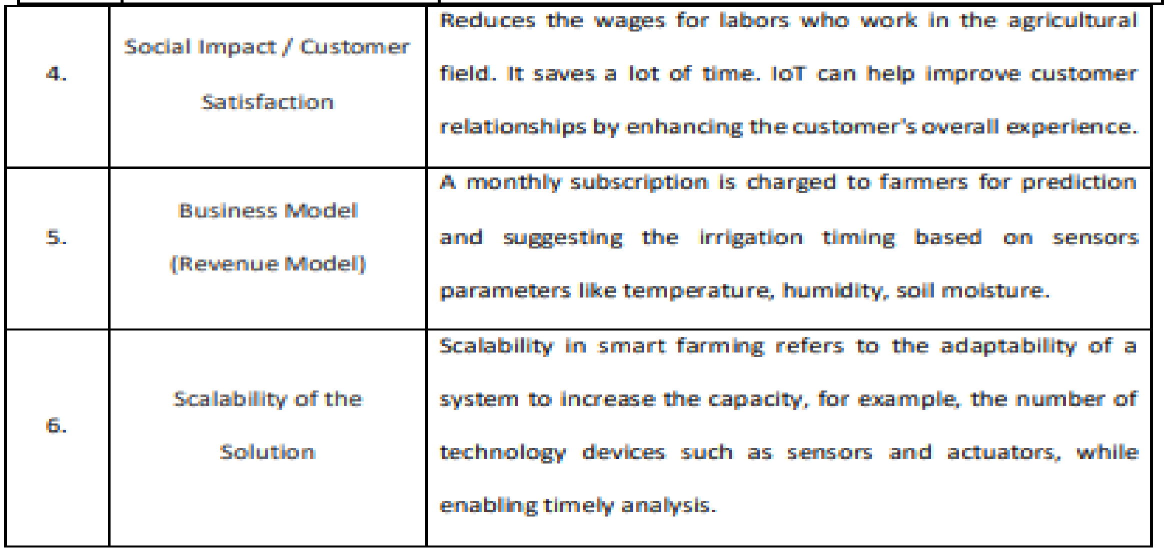
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Nowadays internet of things (IOT) technology is one of the fastest growing fields in different domains including agriculture

the present prortopy is developed using arduino technology which compare specific sensors and wifi module that to coolecy instant data online

The main goal of my project is to use IOT in the agriculture field in order to collect data instantly soil moisture,temprature which will help one to monitor some environment conditions remotely

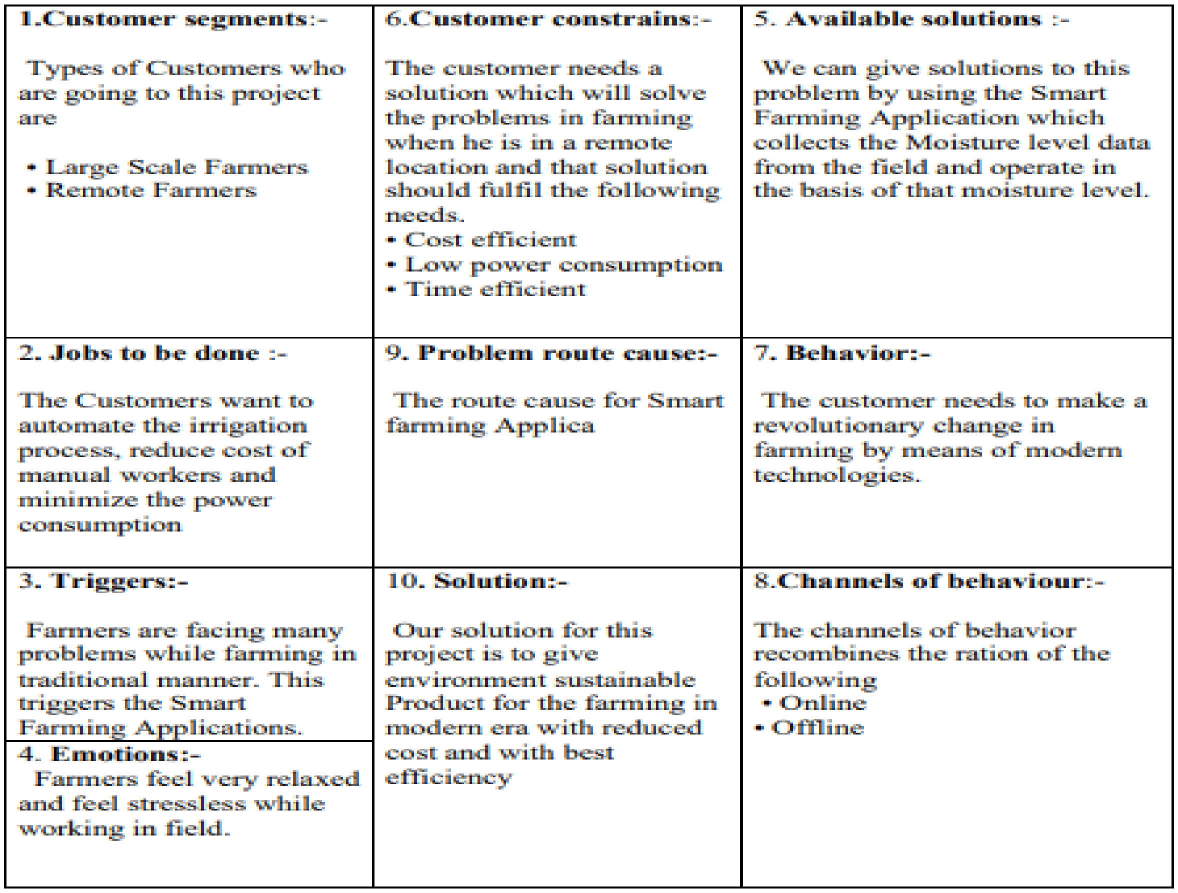
3.3 PROPOSED SOLUTION

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To provide efficient decision support system using wireless sensor network which handle different activities of farm and gives useful information related to farm. Information related to Soil moisture, Temperature and Humidity content. Due to the weather condition, water level increasing Farmers get lot of distractions which is not good for Agriculture.
2.	Idea / Solution description	Smart Agricultural System solutions provide an integrated IoT platform in agriculture that allows farmers to leverage sensors, smart gateways and monitoring systems to collect information, control various parameters on their farms and analyze real-time data in order to make informed decisions.
3.	Novelty / Uniqueness	Various eminent researchers have been making efforts for smart farming by using IoT concepts in agriculture. But, a bouquet of unfolded challenges is still in a queue for their effective solution. This study makes some efforts to discuss past research and open challenges in IoT based agriculture.
4.	Social Impact / Customer Satisfaction	Reduces the wages for labors who work in the agricultural field. It saves a lot of time. IoT can help improve customer relationships by enhancing the customer's overall experience.
5.	Business Model (Revenue Model)	A monthly subscription is charged to farmers for prediction and suggesting the irrigation timing based on sensors parameters like temperature, humidity, soil moisture.
6.	Scalability of the Solution	Scalability in smart farming refers to the adaptability of a system to increase the capacity, for example, the number of technology devices such as sensors and actuators, while enabling timely analysis.

3.4 PROBLEM SOLUTION FIT

Functional requirements are the desired operations of a program, or system as defined



1.Customer segments:- Types of Customers who are going to this project are • Large Scale Farmers • Remote Farmers	**6.Customer constrains:-** The customer needs a solution which will solve the problems in farming when he is in a remote location and that solution should fulfil the following needs. • Cost efficient • Low power consumption • Time efficient	**5. Available solutions :-** We can give solutions to this problem by using the Smart Farming Application which collects the Moisture level data from the field and operate in the basis of that moisture level.
2. Jobs to be done :- The Customers want to automate the irrigation process, reduce cost of manual workers and minimize the power consumption	**9. Problem route cause:-** The route cause for Smart farming Applica	**7. Behavior:-** The customer needs to make a revolutionary change in farming by means of modern technologies.
3. Triggers:- Farmers are facing many problems while farming in traditional manner. This triggers the Smart Farming Applications.	**10. Solution:-** Our solution for this project is to give environment sustainable Product for the farming in modern era with reduced cost and with best efficiency	**8.Channels of behaviour:-** The channels of behavior recombines the ration of the following • Online • Offline
4. Emotions:- Farmers feel very relaxed and feel stressless while working in field.		

4.REQUIREMENT ANALYSIS

Functional requirements are the desired operations of a program, or system as defined in software development and systems engineering. The systems in systems engineering can be either software electronic hardware or combination software-driven electronics.

Two types of Functional Requirements

- functional requirement
- non-functional requirements

4.1FUNCTIONAL REQUIREMENT

- IoT Platforms Require Diverse Connectivity. ...
- IoT Platforms Leverage Applications. ...
- IoT Platforms Manage a Range of Devices. ...
- IoT Platforms Generate Massive Amounts of Data. ...
- IoT Platforms Require Powerful Analytics.

4.2 NON FUNCTIONAL REQUIREMENTS

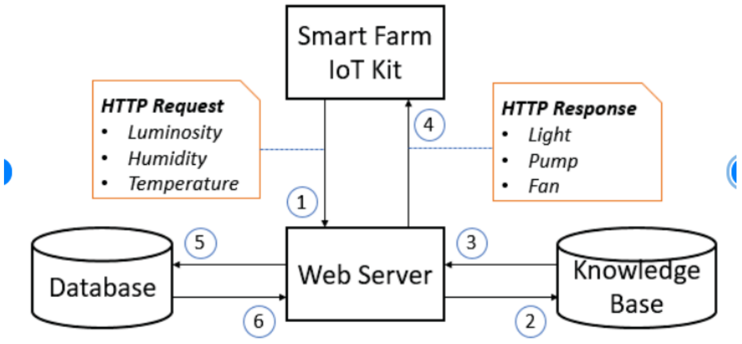
Non-functional requirements describe how a system must behave and establish constraints of its functionality. This type of requirements is also known as the system's quality attributes. Attributes such as performance, security, usability, compatibility are not the feature of the system, they are a required characteristic. They are "developing" properties that emerge from the whole arrangement and hence we can't compose a particular line of code to execute them.

5.PROGECT DESIGN

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.

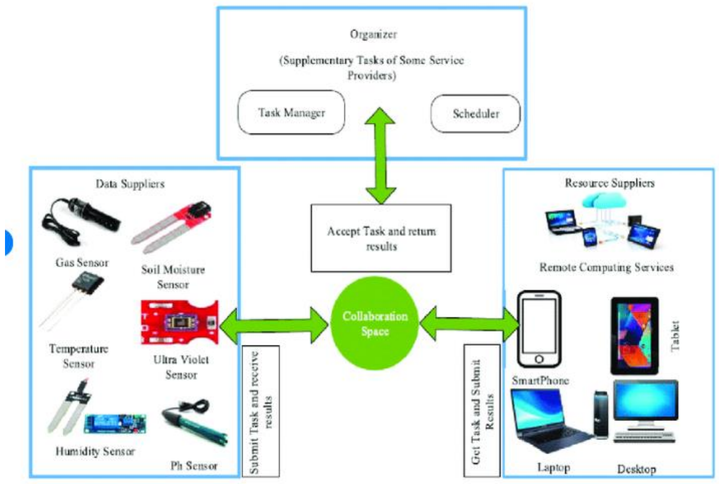
- The different soil parameters temperature, soil moistures and then humidity are sensed using different sensors and obtained value is stored in the IBM cloud.
- Arduino UNO is used as a processing Unit that process the data obtained from the sensors and whether data from the weather API.
- NODE-RED 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 operateto the motor switch.



5.2 SOLUTION AND TECHNICAL ARCHITECTURE

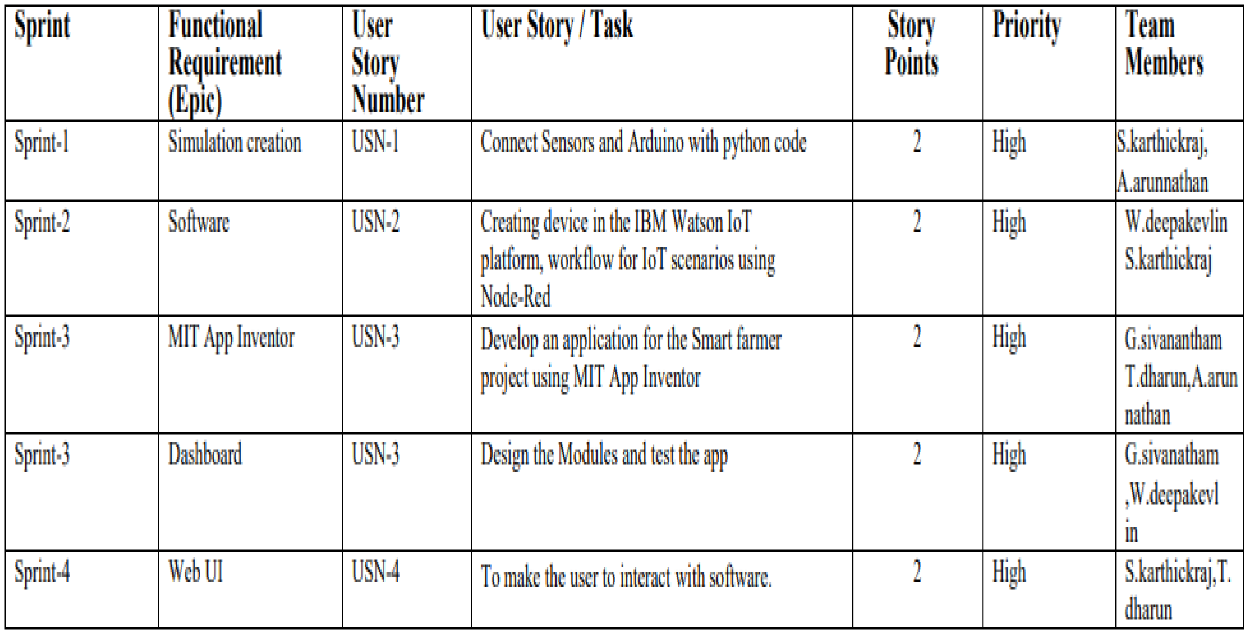
The Deliverable shall include the architectural diagram as below and the information as per the Guidelines:

- The different soil parameters temperature, soil moistures and then humidity are sensedusing 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 sensorsand 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 decide through an app, weather to water the crop or not depending upon the sensor values. By using the app, they can remotelyoperate the motor switch.



6.PROJECT PLANNING AND SCHEDULING

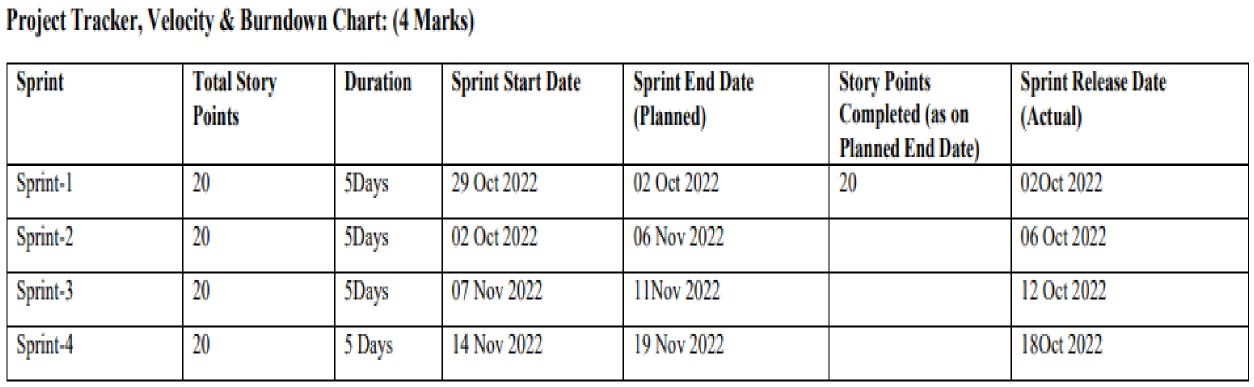
6.1 SPRINT PLANNING



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Simulation creation	USN-1	Connect Sensors and Arduino with python code	2	High	S.karthickraj, A.arunnathan
Sprint-2	Software	USN-2	Creating device in the IBM Watson IoT platform, workflow for IoT scenarios using Node-Red	2	High	W.deepakevlin S.karthickraj
Sprint-3	MIT App Inventor	USN-3	Develop an application for the Smart farmer project using MIT App Inventor	2	High	G.sivanantham T.dharun,A.arun nathan
Sprint-3	Dashboard	USN-3	Design the Modules and test the app	2	High	G.sivanatham ,W.deepakevl in
Sprint-4	Web UI	USN-4	To make the user to interact with software.	2	High	S.karthickraj,T. dharun

6.2 SPRINT DELIVERY SCHEDULE

The outcome of the sprint is a deliverable, albeit with some increments. The scrum is used for projects like Web Technology or development of a product for the new market, i.e. the product with many requirements or fast-changing requirement. When does Sprint Planning take place? Sprint planning occurs on the first day of a new sprint. The event should occur after the sprint review and retrospective from the previous sprint so that any output from those discussions can be considered when planning for the new sprint.



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	5Days	29 Oct 2022	02 Oct 2022	20	02Oct 2022
Sprint-2	20	5Days	02 Oct 2022	06 Nov 2022		06 Oct 2022
Sprint-3	20	5Days	07 Nov 2022	11Nov 2022		12 Oct 2022
Sprint-4	20	5 Days	14 Nov 2022	19 Nov 2022		18Oct 2022

7.CODING AND SOLUTIONING

7.1FEATURE 1

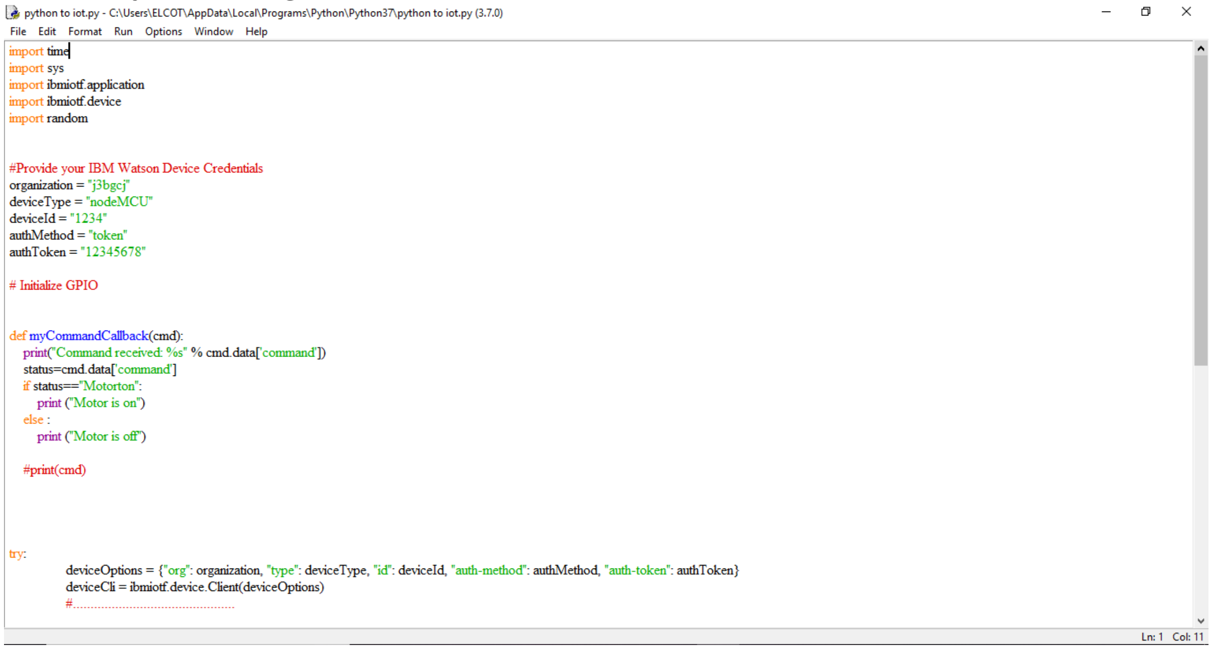
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "j3bgcj"
deviceType = "nodeMCU"
deviceId = "1234"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO


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

    #print(cmd)




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(0,30)
        Humid=random.randint(0,100)
        soil=random.randint(30,50)

        data = { 'temp' : temp, 'Humid': Humid , 'soil_moisture':soil}
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid,"soil_moisture
=%sC"% soil ,"to IBM Watson")

        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()
```

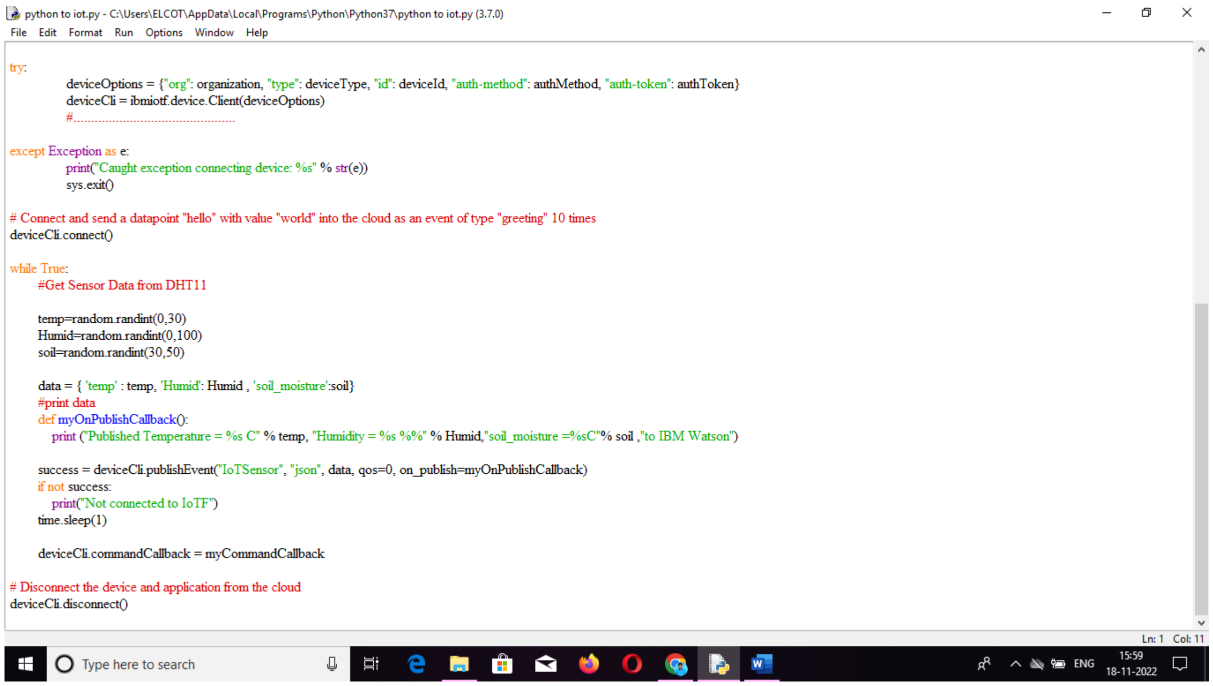
8.TESTING

8.1TEST CASES

Python coding test


python to iot.py - C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\python to iot.py (3.7.0)
File Edit Format Run Options Window Help
Ln: 1 Col: 11

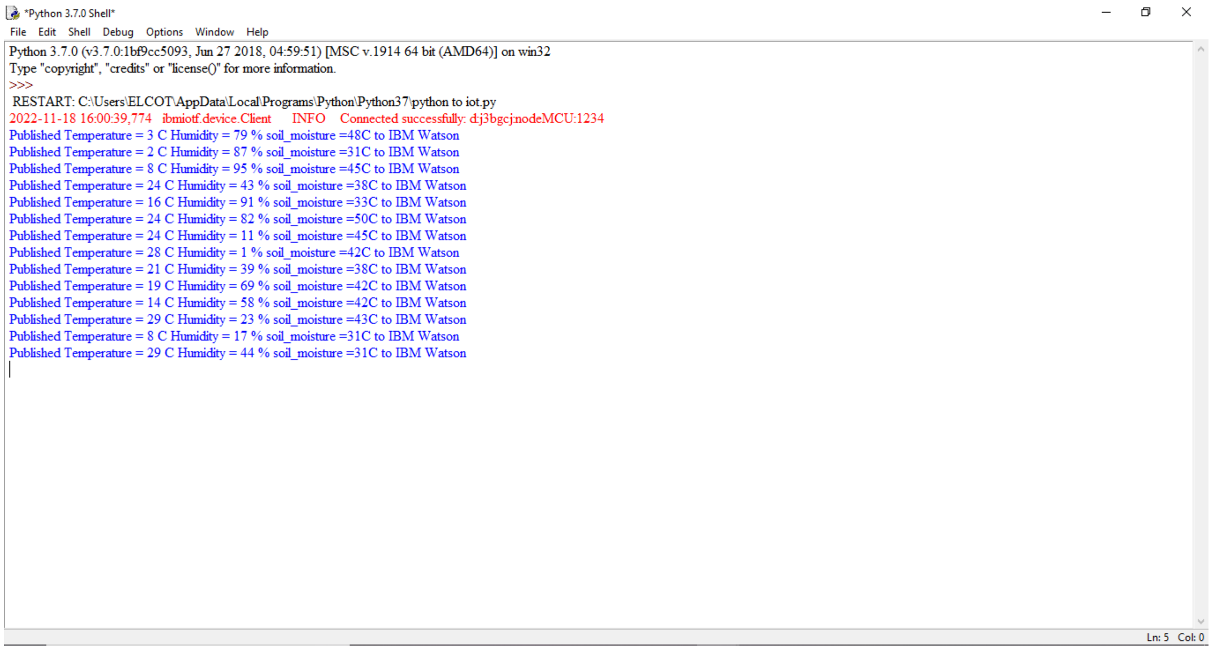
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "j3bgcj"
deviceType = "nodeMCU"
deviceId = "1234"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO


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

    #print(cmd)



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



python to iot.py - C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\python to iot.py (3.7.0)

```
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(0,30)
        Humid=random.randint(0,100)
        soil=random.randint(30,50)

        data = { 'temp' : temp, 'Humid': Humid , 'soil_moisture':soil}
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid,"soil_moisture =%sC"% soil ,"to IBM Watson")

        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()
```

Output

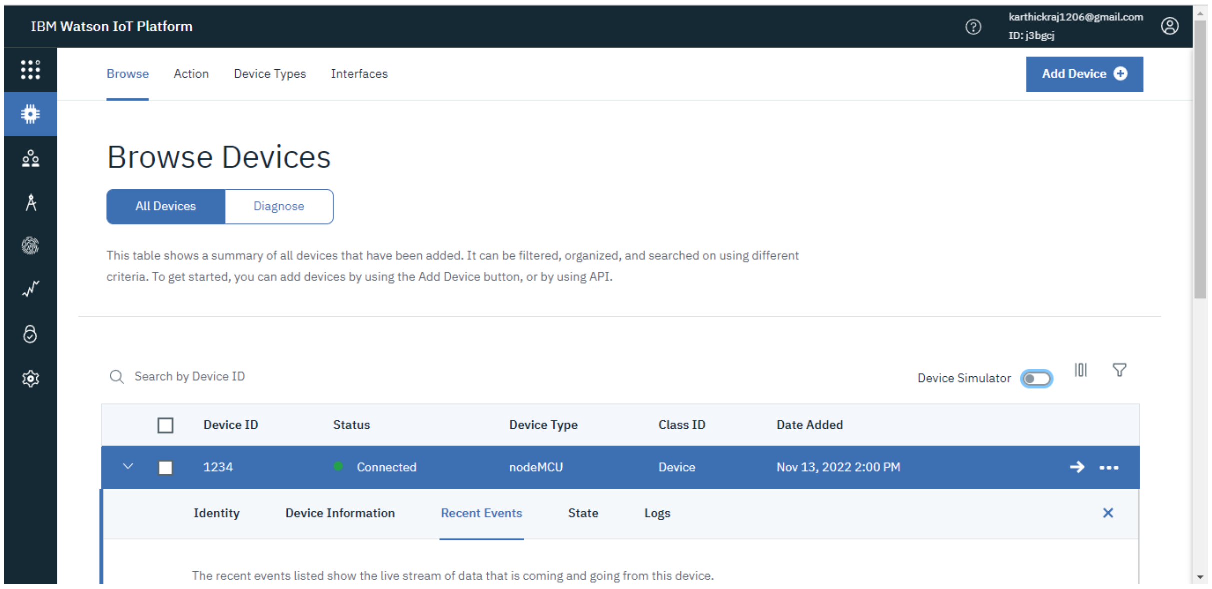


Python 3.7.0 Shell

```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\python to iot.py
2022-11-18 16:00:39,774   ibmiotf.device.Client     INFO    Connected successfully: d:j3bgcj:nodeMCU:1234
Published Temperature = 3 C Humidity = 79 % soil_moisture =48C to IBM Watson
Published Temperature = 2 C Humidity = 87 % soil_moisture =31C to IBM Watson
Published Temperature = 8 C Humidity = 95 % soil_moisture =45C to IBM Watson
Published Temperature = 24 C Humidity = 43 % soil_moisture =38C to IBM Watson
Published Temperature = 16 C Humidity = 91 % soil_moisture =33C to IBM Watson
Published Temperature = 24 C Humidity = 82 % soil_moisture =50C to IBM Watson
Published Temperature = 24 C Humidity = 11 % soil_moisture =45C to IBM Watson
Published Temperature = 28 C Humidity = 1 % soil_moisture =42C to IBM Watson
Published Temperature = 21 C Humidity = 39 % soil_moisture =38C to IBM Watson
Published Temperature = 19 C Humidity = 69 % soil_moisture =42C to IBM Watson
Published Temperature = 14 C Humidity = 58 % soil_moisture =42C to IBM Watson
Published Temperature = 29 C Humidity = 23 % soil_moisture =43C to IBM Watson
Published Temperature = 8 C Humidity = 17 % soil_moisture =31C to IBM Watson
Published Temperature = 29 C Humidity = 44 % soil_moisture =31C to IBM Watson
```

8.2USER ACCEPTANCE TESTING

DEVICE CAN BE CONNECT TO PYTHON CODE



IBM Watson IoT Platform

karthickraj1206@gmail.com
ID: j3bgcj

Browse Action Device Types Interfaces Add Device

Browse Devices

All Devices Diagnose

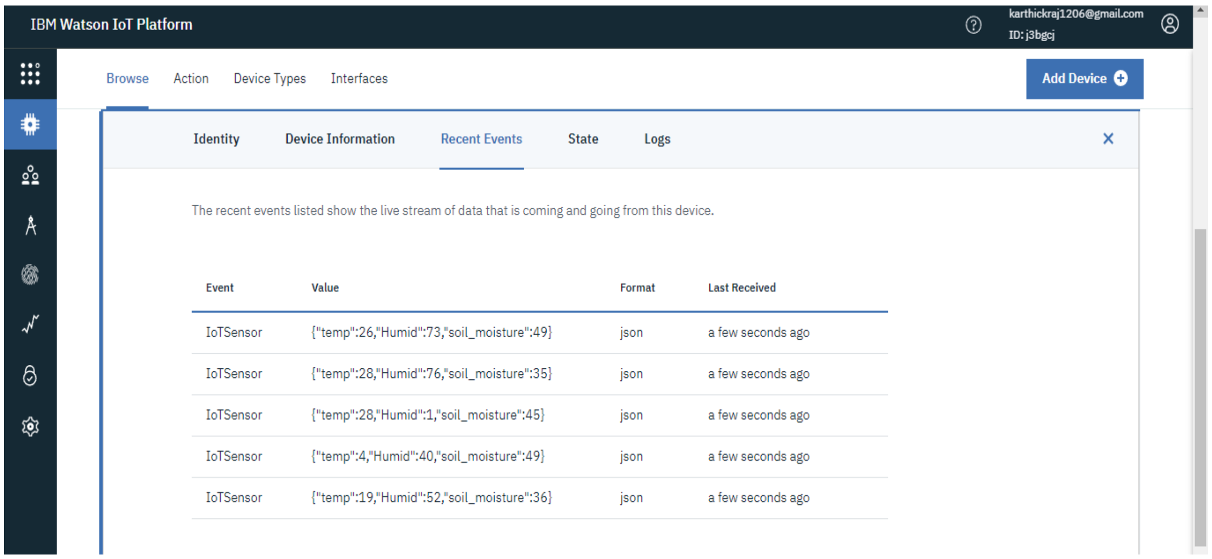
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Search by Device ID Device Simulator

Device ID	Status	Device Type	Class ID	Date Added
1234	Connected	nodeMCU	Device	Nov 13, 2022 2:00 PM

Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.



IBM Watson IoT Platform

karthickraj1206@gmail.com
ID: j3bgcj

Browse Action Device Types Interfaces Add Device

Identity Device Information Recent Events State Logs

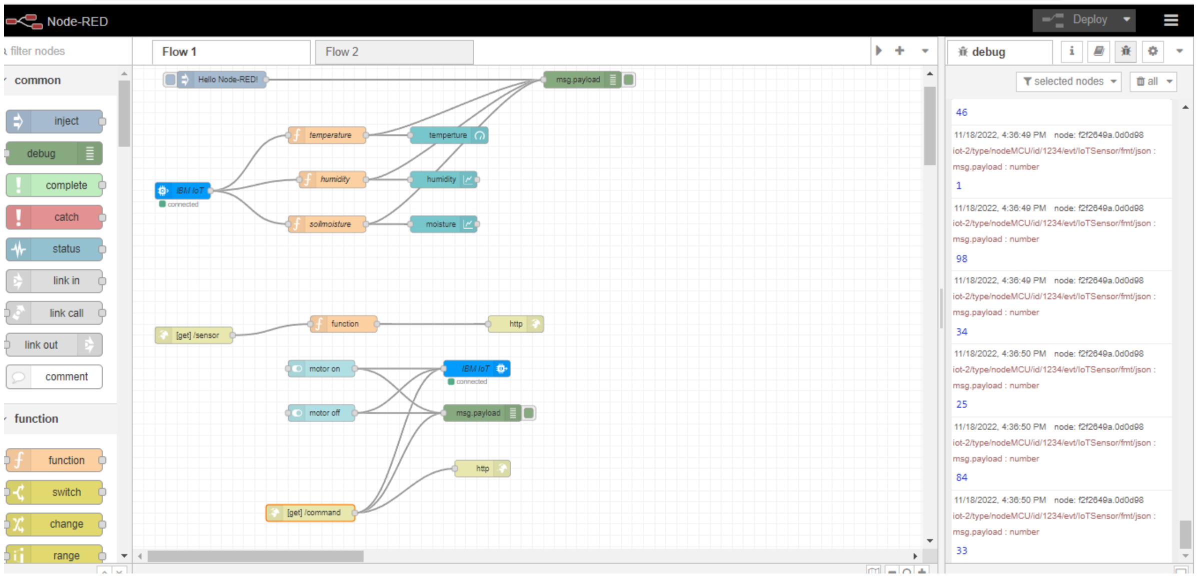
The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
IoTSensor	{"temp":26,"Humid":73,"soil_moisture":49}	json	a few seconds ago
IoTSensor	{"temp":28,"Humid":76,"soil_moisture":35}	json	a few seconds ago
IoTSensor	{"temp":28,"Humid":1,"soil_moisture":45}	json	a few seconds ago
IoTSensor	{"temp":4,"Humid":40,"soil_moisture":49}	json	a few seconds ago
IoTSensor	{"temp":19,"Humid":52,"soil_moisture":36}	json	a few seconds ago

9.RESULTS

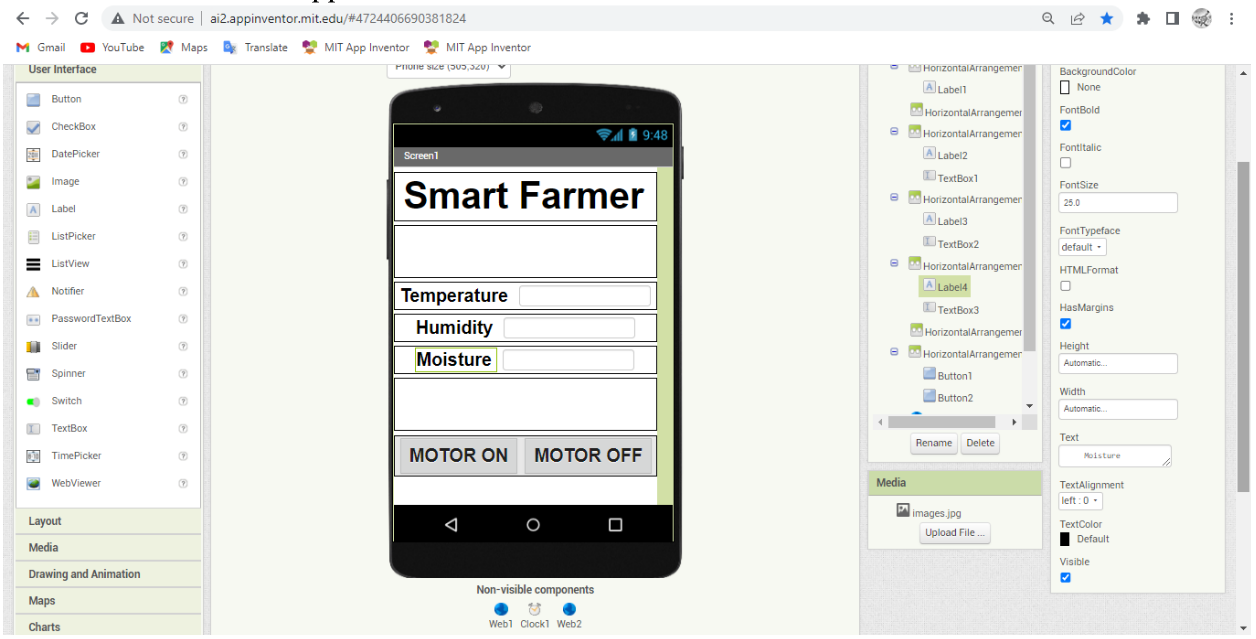
9.1PERFORMANCE METRICS

Node-red connect to the ibm iot



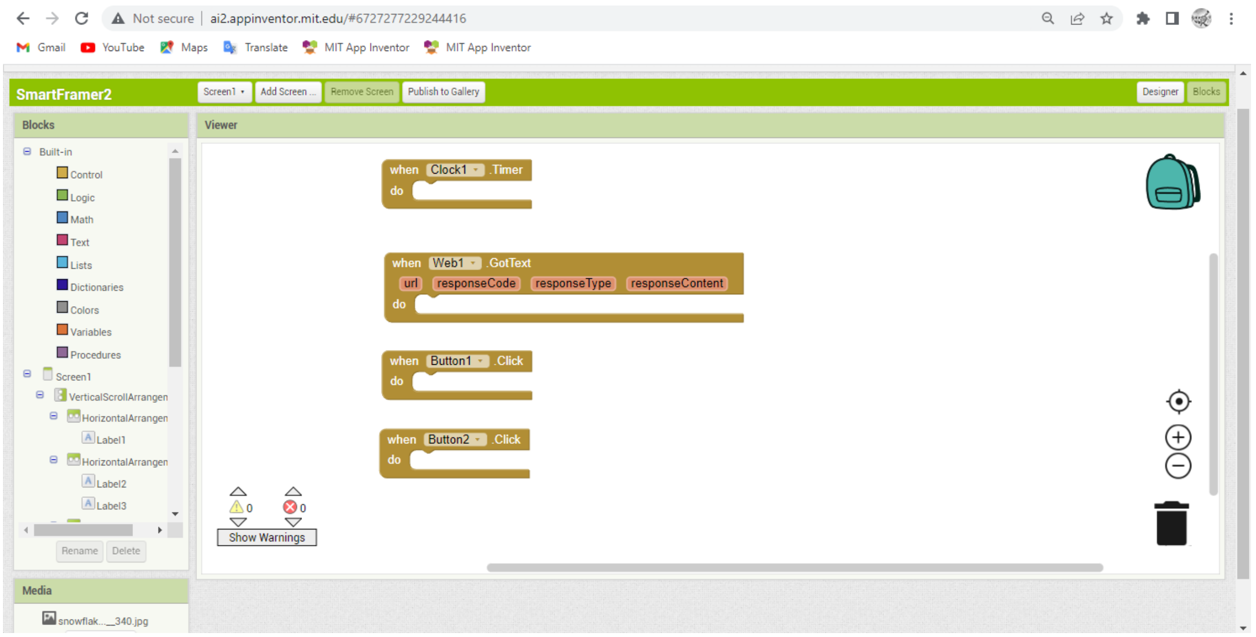
```
46
11/18/2022, 4:36:49 PM   node: f2f2649a.0d0d98
iot-2/type/nodeMCU/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
1
11/18/2022, 4:36:49 PM   node: f2f2649a.0d0d98
iot-2/type/nodeMCU/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
98
11/18/2022, 4:36:49 PM   node: f2f2649a.0d0d98
iot-2/type/nodeMCU/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
34
11/18/2022, 4:36:50 PM   node: f2f2649a.0d0d98
iot-2/type/nodeMCU/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
25
11/18/2022, 4:36:50 PM   node: f2f2649a.0d0d98
iot-2/type/nodeMCU/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
84
11/18/2022, 4:36:50 PM   node: f2f2649a.0d0d98
iot-2/type/nodeMCU/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
33
```

Node red connect web application mit

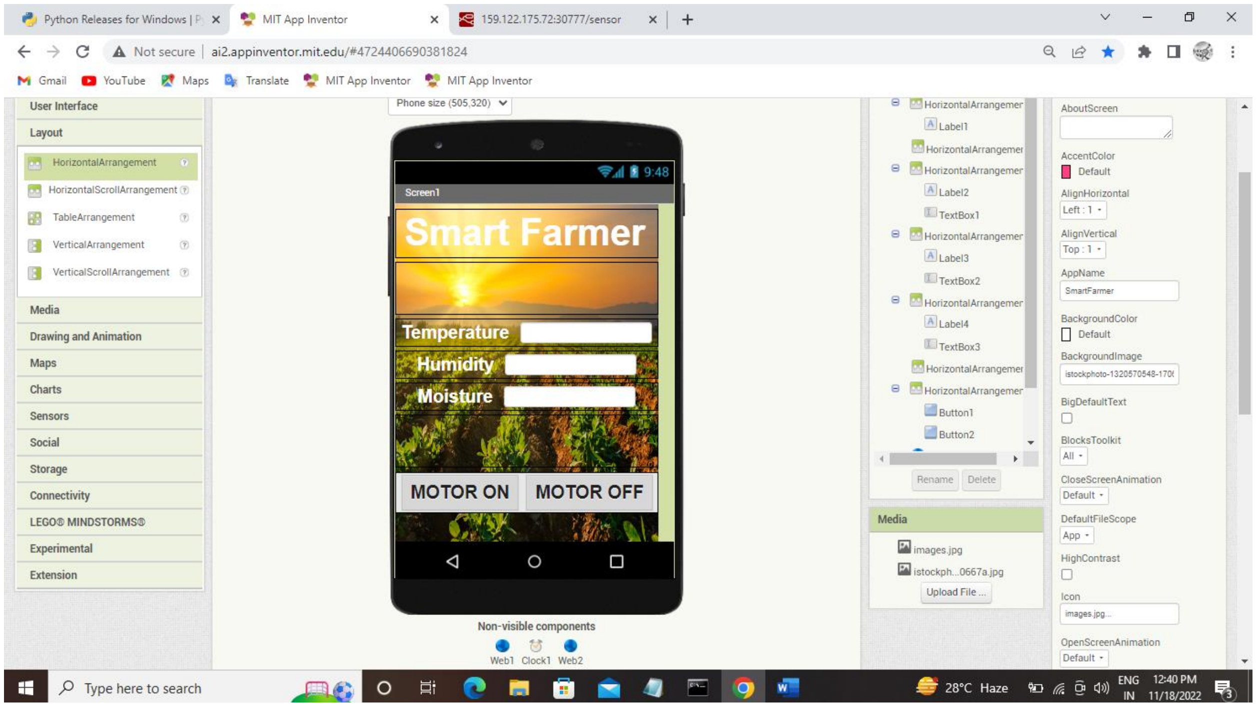


BackgroundColor: None
FontBold: ✓
FontItalic: ☐
FontSize: 25.0
FontTypeface: default
HTMLFormat: ☐
HasMargins: ✓
Height: Automatic...
Width: Automatic...
Text: Moisture
TextAlignment: left : 0
TextColor: Default
Visible: ✓

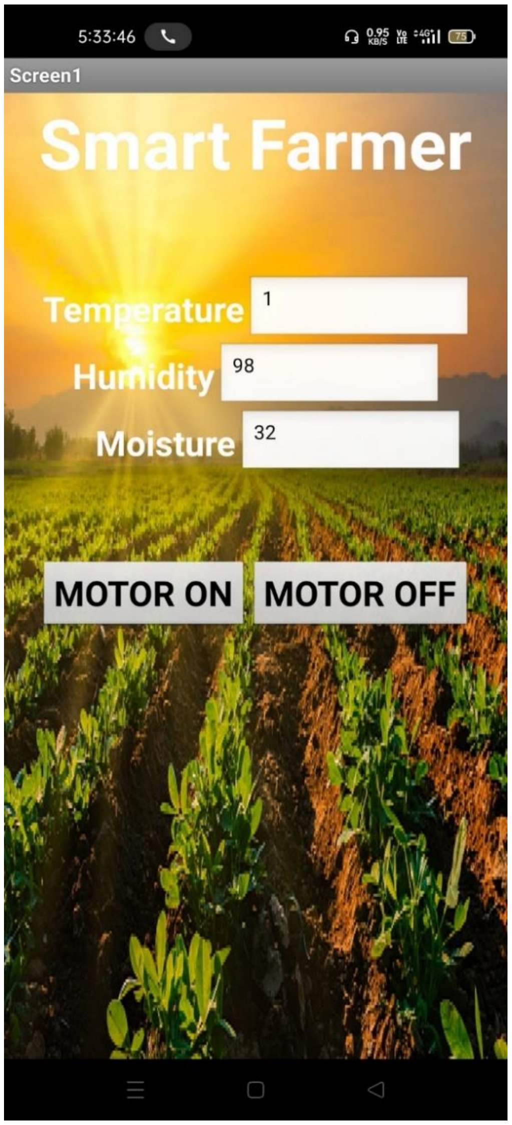
Back end flow



Frontend application



Mobile screen shot



Temperature 1

Humidity 98

Moisture 32

10.ADVANTAGES AND DISADVANTAGES

Advantages:

- A remote control system can help in working irrigation system valves dependent on schedule. Irrigating remote farm properties can be exceptionally troublesome and labor- intensive. It gets hard to comprehend when the valves were started and whether the ideal measure of water was distributed.

- For situations where a quick reaction is required, manual valve actuation may not be conceivable constantly. Thus, remote observing and control of irrigation systems, generators or wind machines or some other motor-driven hardware become the next logical step.

- Various solutions are available to monitor engine statistics and starting or stopping the engine. When the client chooses to begin or stop the motor, the program transmits a sign to the unit within seconds by means of a mobile phone system.

- Submersible weight sensors or ultrasonic sensors can screen the degree of tanks, lakes, wells and different kinds of fluid stockpiling like fuel and compost. The product figures volume dependent on the tank or lake geometry after some time. It conveys alarms dependent on various conditions.

DISADVANTAGES:

- The smart agriculture needs availability of internet continuously. Rural part of most of the developing countries do not fulfil this requirement. Moreover internet connection is slower.
- The smart farming based equipment require farmers to understand and learn the use of technology. This is major challenge in adopting smart agriculture farming at large scale across the countries.

11.CONCLUSION

Farmers can benefit greatly from an IoT-based smart agriculture system. As a result of the lack of irrigation, agriculture suffers. Climate factors such as humidity, temperature, and moisture can be adjusted dependent on the local environmental variables. Thistechnology also detects animal invasions, which are a major cause of crop loss. This technology aids in the scheduling of irrigation based on present data from the field and records from a climate source. It helps in deciding the farmer to whether to do irrigation or not to do. Continuous internet connectivity is required for continuous monitoring of data from sensors. This also can be overcome by using GSM unit as an alternative of mobile app. By GSM, SMS can be sent to farmers phone.

12.FUTURE SCOPE

In the current project we have implemented the project that can protect and maintain the the crop. In this project the farmer monitor and control the field remotely. In future we can add or update few more things to this project

- . We can create few more models of the same project ,so that the farmer can have information of a entire.
- We can update the this project by using solar power mechanism. So that the power supply from electric poles can be replaced with solar panels. It reduces the power line cost. It will be a one time investment. We can add solar fencing technology to thisproject.
- We can use GSM technology to this project so that the farmers can get the information directly to his home through SMS. This helps the farmer to get information if there is a internet issues.
- We can add camera feature so that the farmer can monitor his field in real time. This helps in avoiding thefts.

13.APPENDIX

Source Code

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


#Provide your IBM Watson Device Credentials
organization = "j3bgcj"
deviceType = "nodeMCU"
deviceId = "1234"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO


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

    #print(cmd)




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(0,30)
        Humid=random.randint(0,100)
        soil=random.randint(30,50)

        data = { 'temp' : temp, 'Humid': Humid , 'soil_moisture':soil}
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" %
Humid,"soil_moisture =%sC"% soil ,"to IBM Watson")

        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 LINK:

https://github.com/IBM-EPBL/IBM-Project-53628-1661425026

Project demo link:

https://youtu.be/LloFSxB4X8Y