SMART FARMER-IOT ENABLED SMART FARMING APPLICATION

PROJECT REPORT

TEAM ID:PNT2022TMID31297

Submitted by

ANUKEERTHANA S	**710719104012**
ANANDHAN V	**710719104011**
DHANUSRIYA A S	**710719104028**
MANI BHARATHI M	**710719104058**

CONTENT

1. **INTRODUCTION**
   - **1.1 Project Overview**
   - **1.2 Purpose**
2. **LITERATURE SURVEY**
   - **2.1 Existing problem**
   - **2.2 Problem Statement Definition**
3. **IDEATION &PROPOSED SOLUTION**
   - **3.1 Empathy Map Canvas**
   - **3.2 Ideation & Brainstorming**
   - **3.3 Proposed Solution**
   - **3.4 Problem Solution fit**
4. **REQUIREMENT ANALYSIS**
   - **4.1 Functional Requirement**
   - **4.2 Non-Functional Requirement**
5. **PROJECT DESIGN**
   - **5.1 Data Flow Diagram**
   - **5.2 Solution & Technical Architecture**
6. **PROJECT PLANNING & SCHEDULING**
   - **6.1 Sprint Planning & Estimation**
   - **6.2 Sprint Delivery Schedule**
   - **6.3 Reports from JIRA**
7. **CODING AND SOLUTION**
   - **7.1 Feature 1**
   - **7.2 Feature 2**
   - **7.3 Database Schema (if Applicable)**
8. **TESTING**
   - **8.1 Test Cases**
   - **8.2 User Acceptance Testing**
9. **RESULT**
   - **9.1 Performance Metrics**
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**

    Source Code, GitHub Link

1.INTRODUCTION

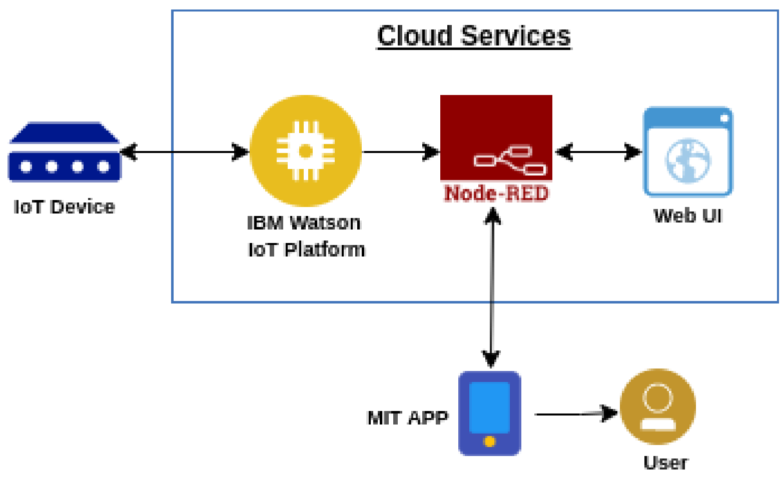
1.1 PROJECT OVERVIEW

IoT- based agriculture system helps the farmer in monitoring different parameters of his field like soil moisture, Temperature, humidity using some sensors. Farmers can monitorall the sensor parameters by using a web or mobile application even if the farmer is not nearhis field. Watering the 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.

IoT is network that connects physical objects or things embedded with electronics, software and sensors through network connectivity that collects and transfers data using cloud for communication. Data is transferred through internet without human to human or human to computer interaction. In this project we have not used any hardware. Instead of real soil and temperature conditions, sensors IBM IoT Simulator is used which can transmit soil moisture temperature as required.

Project requirements: Node-RED, IBM Cloud, IBM Watson IoT, Node.js, IBM Device, IBM IoT Simulator, Python 3.7, Open Weather API platform.

Project Deliverables: Application for IoT based Smart Agriculture System



1.2 PURPOSE

IoT based farming improves the entire agriculture system by monitoring the field in real-time. With the help of IoT in agriculture not only saves the time but also reduces the extravagant use of resources such as water and electricity. Sometimes due to over or less supply of water in the agricultural field crops may not grow proper. Using IoT supply of water and growth of plants can be satisfied to a greater extent. The flow of water can be controlled from the application.

Smart agriculture is a farming system which uses IoT technology. This emerging system increases the quantity and quality of agricultural products. IoT devices provide information about nature of farming fields and then take action depending on the farmer input.

The main goal of my project is to use IoT in the agriculture field in order to collect data instantly (soil Moister, temperature, humidity…), which will help one to monitor some environment conditions remotely, effectively and enhance tremendously the production and therefore the income of farmers. The present prototype is developed using Arduino technology, which comprise specific sensors, and a WIFI module that helps to collect instant data online. Worth mentioning the testing of this prototype generated, highly accurate data because while we were collecting them remotely any environmental changes were detected instantly and taking in consideration to make decisions.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

Watering the field is a difficult process, Farmers have to wait in the field until the water covers the whole farm field. Power Supply is also one of the problems. In Village Side, the power supply may vary. The Biggest Challenges Faced by IoT in the Agricultural Sector are Lack of Information, High Adoption, Cost and Security Concerns, etc. The farmers do not have that much knowledge on the internet of things and good internet connection is required. So farmers don’t know how to use the web application and to make a connection if any component get failed.

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

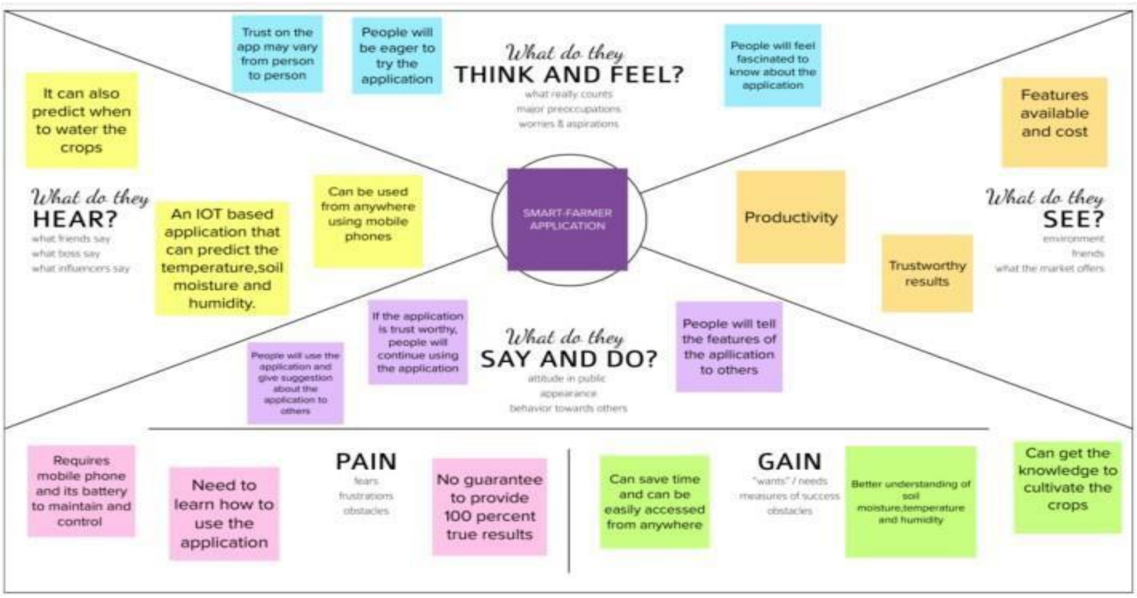
2.2 PROBLEM STATEMENT DEFINITION

The Biggest Challenges Faced by IoT in the Agricultural Sector are Lack of Information, High Adoption, Cost and Security. The farmers do not have that much knowledge on the internet of things and good internet connection is required. Power Supply is also one of the problems In Village Side, the power supply may vary. So farmers don’t know how to use the web application and to make a connection if any component get failed.

3.IDEATION & PROPOSED SOLUTION

3.1EMPATHY MAP CANVAS

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

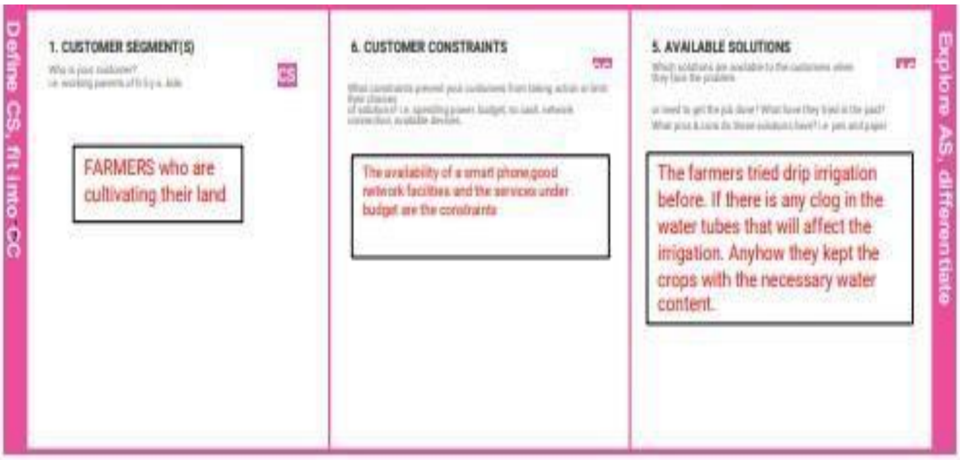


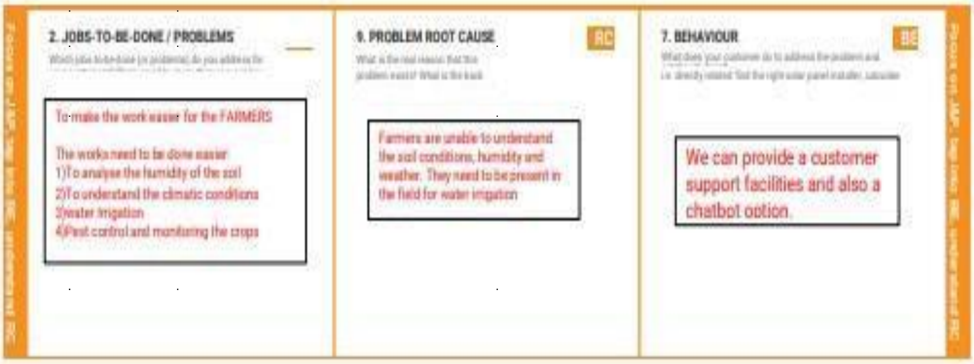
3.2 IDEATION AND BRAINSTORMING

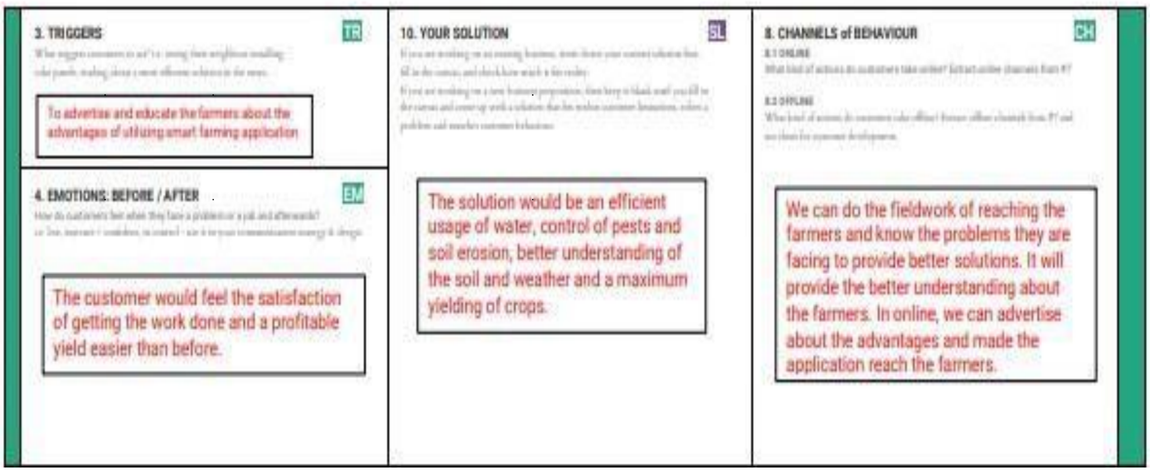
3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement {Problem to be solved)	Thebetterunderstanding of the soil, weather. humidity andwater irrigatiOn should be made easier. Theautomation would help farmer'S can do theil' farming from anywhel'e. Thewater level andpest contl'OI monitoring under application should be done.
2.	Idea/ Solution description	TheappUcation is a embedded network of different devices which make a self configuring network. The new de lopments of Smart Farming with use of IoT, by day turning the face of con ntionalag,iculture methods by not only making it optimal but also making it cost efficient for farmel's andre:dudng crop wastage.
3.	Novelty/ Uniqueness	Smart Farming appUcation impl'oves the entil'e Agriculture system by monitoring the field in l'eal limeuSingIOT devices. With the help of sensors andinterconnectivity, the Internet of Things in Agriculture hasnot only saved the timeof the farmel's but hasalso !'educed the vast useof resources such asWater andElectricity.
4.	social Impact/ Customer satisfaction	Smart farming, the dependency on manual labour has!'educed signific.antly. TheproceS:Ses like pe-s:t control, fertilizing, and irrigation al'e incl'easingly becoming automated. andfarmer'S can control them remotefy. The use of sma.rt IoT sensol's canmaintain these pl'ocesses.inc:reasing cropproduction.
S.	BusineS:S Model (Revenue Model)	It istrying to execute this technique aswe need to introduce an Al'duino sensol'swhich was modified with anArduino that takes l'eceived signals from sensol'S. Easy operability and maintenance. Requil'ed low time for maintain. Cost is l'easonable.
6.	Sc.alability of the Solution	Theadaptability of a system to incl'ease the capacity of smart farming. The sensorsand act.uator'Susedhereenables the efficiency of the system. The system isscalable for the usage and themonitoring of the crops.

3.4 PROBLEM SOLUTION FIT







4 REQUIREMENT ANALYSIS

4.2FUNCTIONAL REQUIREMENT

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	Enter e-mail address and password E-mail: Enter email address Password: Enter password
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Log In	It will serve authentication for Logging into the system
FR-4	Manage modules	It will manage system admins, roles of user and user permission
FR-5	Check whether condition	It will perform Temperature monitoring status and humidity monitoring status
FR-6	Log Out	Exit from the system

4.3 NON-FUNCTIONAL REQUIREMENT

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Better understanding of the application, learning ability, usage efficiency, technology advancement and time saving.
NFR-2	**Security**	Their preferences and decisions are kept secret. Their data would help them to achieve greater yield by analysing the records kept secretly.
NFR-3	**Reliability**	The shared protection achieves a better trade-off between costs and reliability. The model uses dedicated and shared protection methodology to avoid farm service outages.
NFR-4	**Performance**	The process of implementing integrated sensors with sensing soil and environmental parameters in farming will be more efficient. Since performance is a major concern for customers, it would attract them.
NFR-5	**Availability**	Automatic adjustment of farming equipment made possible by linking information like crops or 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, real time decision making is feasible in an environment.

5 PROJECT DESIGN

5.2DATAFLOW DIAGRAM

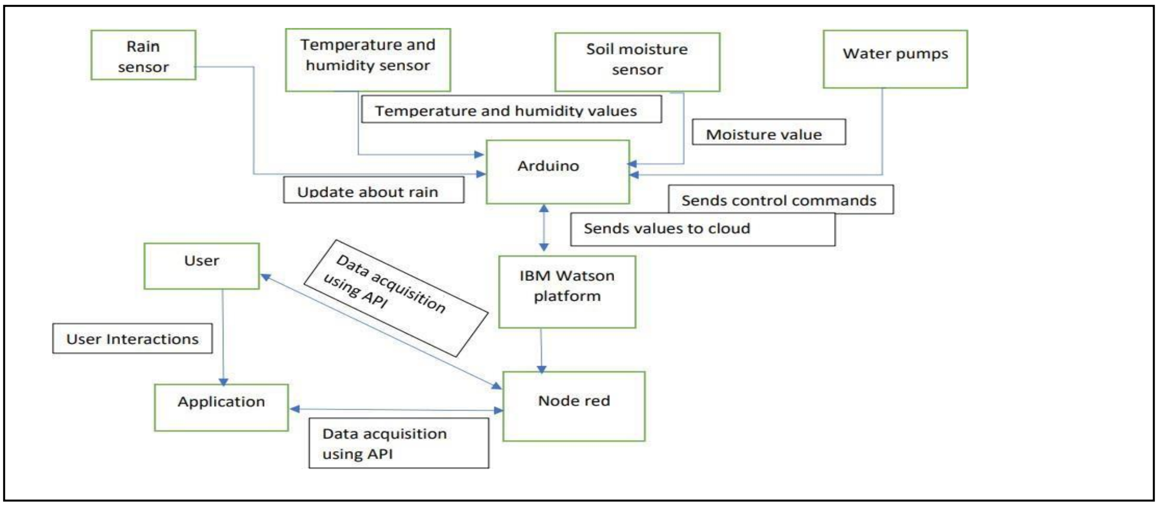
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

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 sensorsand whether data from the weather API.

NODE-RED is used as a programming tool to write the hardware, software, and APIs. TheMQTT 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.



5.3 SOLUTION & TECHNICAL ARCHITECTURE

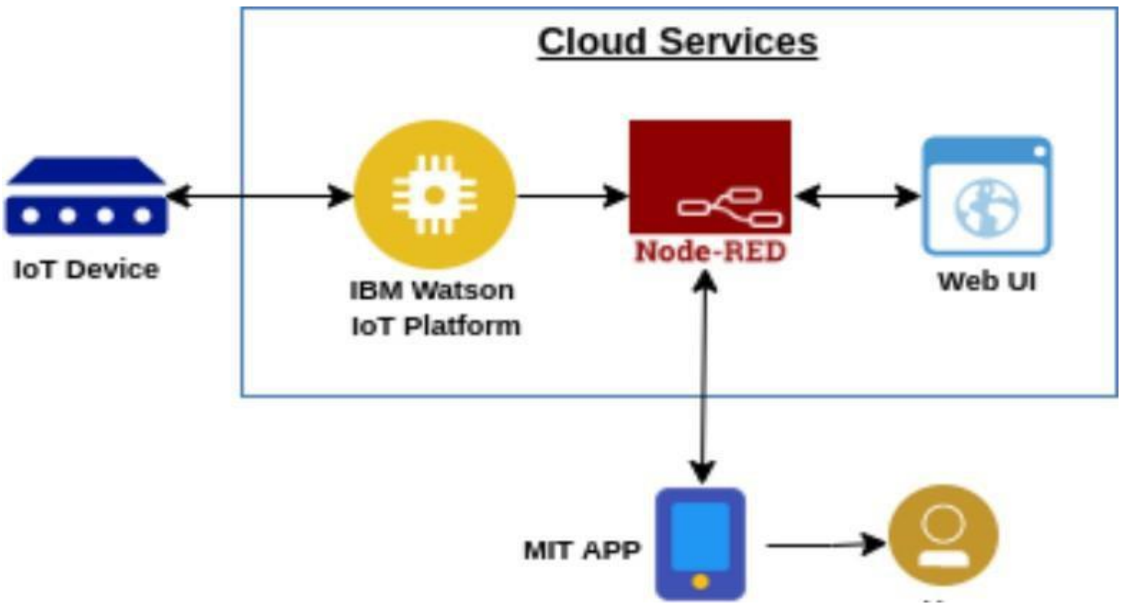
The different soil parameters temperature, soil moistures and then humidity are sensed usingdifferent sensors and obtained value is stored in IBM cloud.

Arduino Uno is used as a processing unit that process the data obtained from the sensors andweather data from the weather API

NODE-RED is used as a programming tool to write the hardware, software and APIs. TheMQTT 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, whether to water the crop or not depending upon the sensor values, By using the app,they can remotelyoperate the motor switch(0n/off).



6 PROJECT PLANNING & SCHEDULING

Sprint	User Story Number	User Story/ Task	Story Points	Priority	Team Members
Sprint 1	USN-1	Sensors and wi-fi module with python code	2	High	ANUKEERTHANA S ANANDHAN V DHANUSRIYA A S MANI BHARATHI M
Sprint 2	USN-2	IBM Watson IoT Platform, workflows for IoT Scenarios using Node-Red	2	High	ANUKEERTHANA S ANANDHAN V DHANUSRIYA A S MANI BHARATHI M
Sprint 3	USN-3	To Develop an Mobile application using MIT	2	High	ANUKEERTHANA S ANANDHAN V DHANUSRIYA A S MANI BHARATHI M
Sprint 4	USN-4	To make the user to interact with Software	2	High	ANUKEERTHANA S ANANDHAN V DHANUSRIYA A S MANI BHARATHI M

7 CODING AND SOLUTION

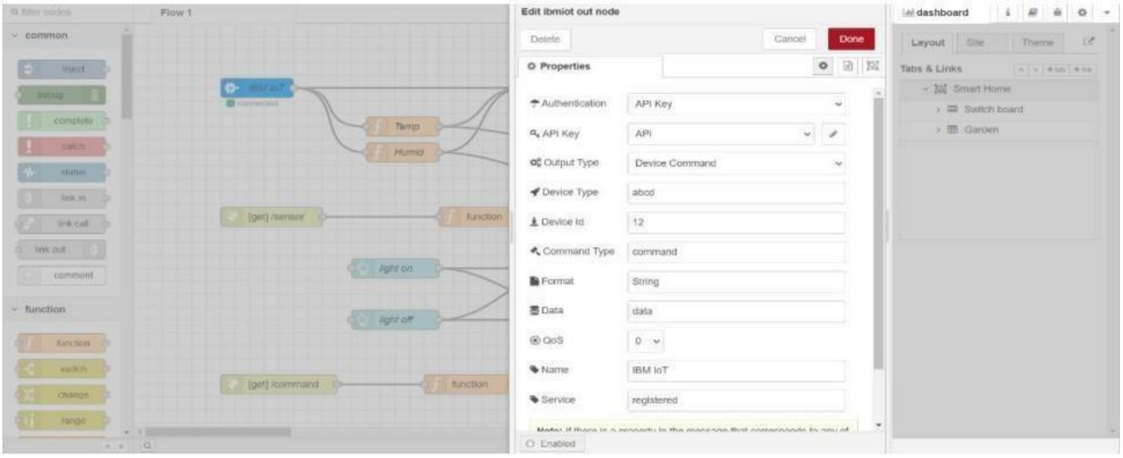
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials

organization = "1xIO8d"
deviceType = "abcd"
deviceId = 12
token = "12345678"
authMethod = "token"

# Initialize GPIO

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

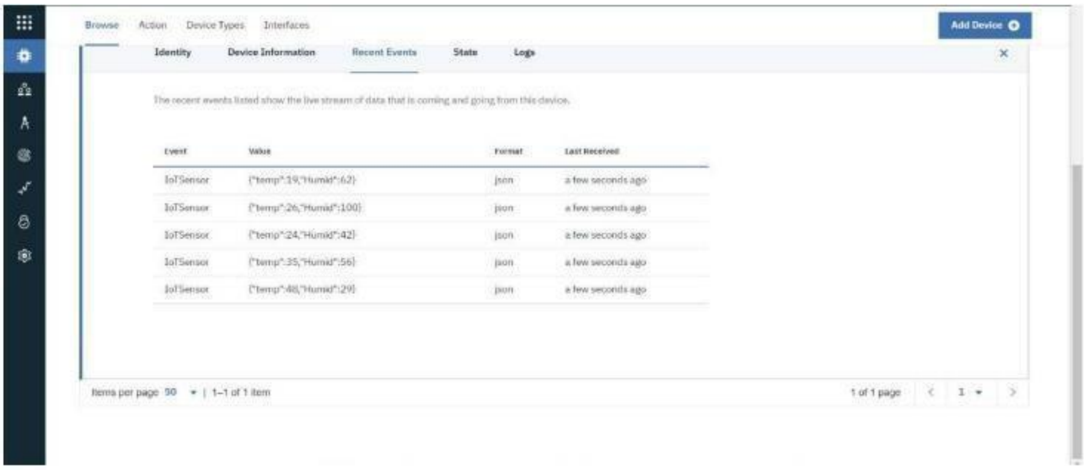
```
#print(cmd)

try:

deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth- authMethod,
"auth-token": Token}
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 type
"greeting" 10 times
deviceCli.connect()

while True:
#Get Sensor Data from DHT11
temp=random.randint(90,110)
hum=random.randint(60,100)
soil=random.randint(20,100)
data = { 'temp' : temp, 'humid': humid , 'Mois': Mois}
#print data def myOnPublishCallback( ):
print (f"Published temp = {temp} C , humid = {humid} , Mois = {Mois} deg c to IBM
success = deviceCli.publishEvent("IoTSensor", "json", data,
qos=0,on_publish=myOnPublishCallback) if not success:
print("Not connected to IoTF") time.sleep(10)
deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud deviceCli.disconnect
```

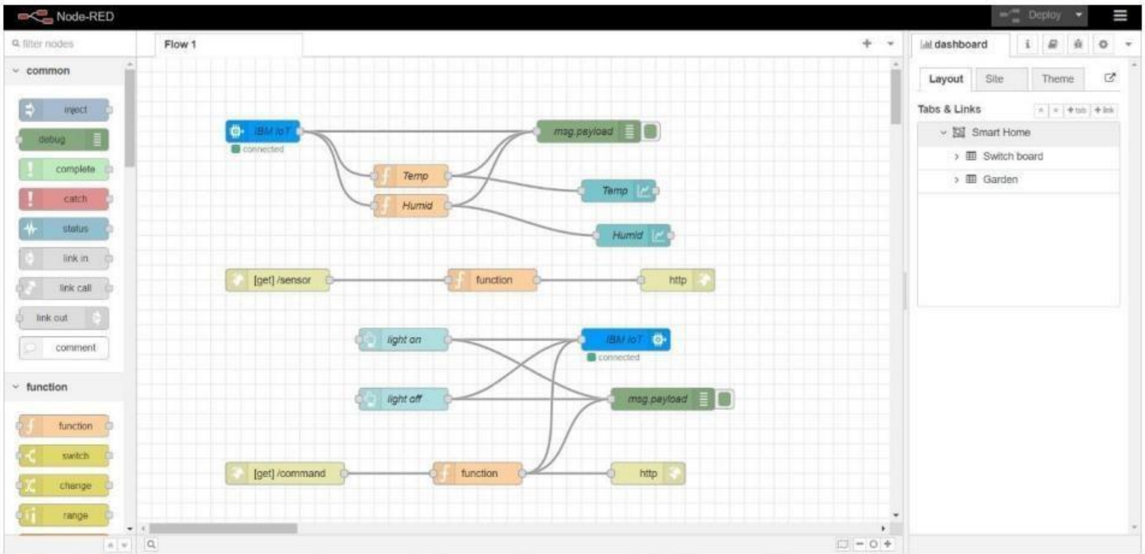
8 TESTING

8.2 TESTCASES

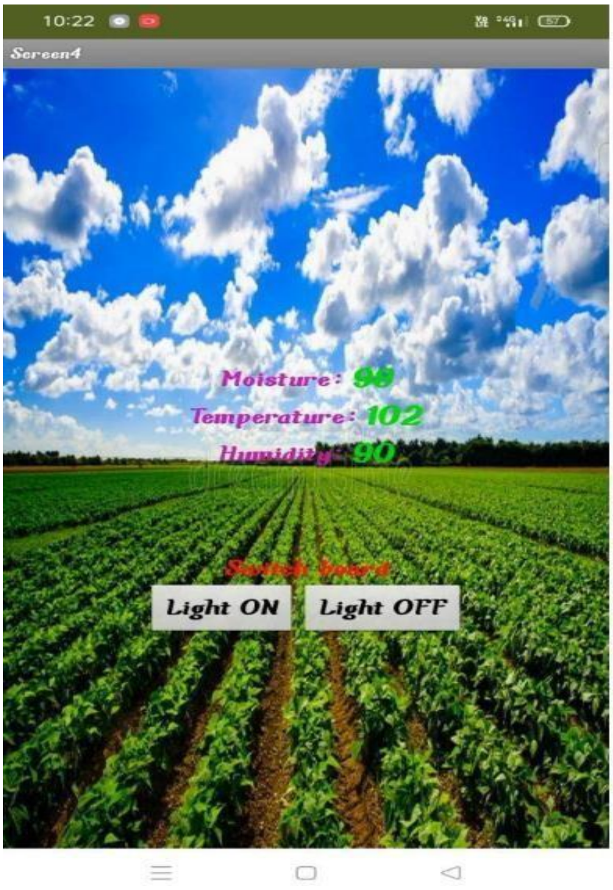
Web application using node red.







8.3 USER ACCEPTANCE TESTING



9 RESULTS

9.2PERFORMANCE MATRICS

So finally when we run the python code it is going to connect the IBM Watson platform and connecting to the node-red after that is going to connect the mobile application.so we can see output in the fourth window.

10 ADVANTAGES & 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 conceivableconstantly. 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 anddifferent 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. This technology 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 farmerto 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 farmer’s 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 updatefew 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 fromelectric 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 this project.

We can use GSM technology to this project so that the farmers can get the information directly tohis 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.

13 APPENDIX

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

#Provide your IBM Watson Device
organization = "1x108d"
deviceType = "abcd" deviceId = “12” authMethod = “token”
authToken = “12345678”
# Initialize GPIO

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

deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth- authMethod,
"auth-token": Token}
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 type
"greeting" 10
deviceCli.connect()

while True:
#Get Sensor Data from DHT11 temp=random.randint(90,110)
hum=random.randint(60,100 soil=random.randint(20,100)
data = { 'temp' : temp, 'hum': hum , 'soil': soil} #print data def myOnPublishCallback( ):
print (f"Published temp = {temp} C , humid = {humid} , Mois = {Mois} deg c to IBM
success = deviceCli.publishEvent("IoTSensor", "json", data)
qos=0,on_publish=myOnPublishCallback)
if not success:
print("Not connected to IoTF")
time.sleep(10)
deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud deviceCli.disconnect
```

Github Link : https://github.com/IBM-EPBL/IBM-Project-38542-1660382299