SMART FARMER-IOT ENABLED SMART FARMING APPLICATION

PROJECT REPORT

Submitted by

VIVEKA VARATHAN R(737819ECR218)

SHIVANAND K(737819ECR176)

SUDHAN A(737819ECL197)

SARAVANAN K(737819ECR167)

Team ID:PNT2022TMID04737

in partial fulfilment

for the award of the degree

of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

KONGU ENGINEERING COLLEGE

(AUTONOMOUS)

PERUNDURAI, ERODE-638060

PROJECT REPORT

1. INTRODUCTION.. 01

1.1 Project requirements

2. LITERATURE SURVEY….. 03

2.1 Reference

3. PROPOSED METHOD...05

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Solution fit

4. REQUIREMENT ANALYSIS... 11

4.1 Functional requirements

4.2 Non-Functional requirements

5. PROJECT DESIGN... 12

5.1 Data Flow Diagrams

5.2 Solution and Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING 18

6.1 Sprint Planning and Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7. CODING & SOLUTIONING 20

7.1 Feature 1

7.2 Feature 2

8. TESTING.. 22

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS 24

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES.. 25

10.1 Advantage
10.2 Disadvantage

11. CONCLUSION AND FUTURE SCOPE....................................... 26

11.1 Conclusion
11.2 Future scope

12. APPENDIX.. 27

12.1 Source Code and GitHub

12.2 Project Demonstration Link

CHAPTER 1

INTRODUCTION

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.

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. IOT- Enabled Smart Farming agriculture system helps the farmer in monitoring different parameters of his field like soil moisture, temperature, humidity using some sensors. Farmer can monitor all the sensor parameters by using a web or mobile application even if the farmer is not near his field. Watering the crop is one of the Important task for the farmers. They can make the decision whether to water the crop or postpone it by monitoring the sensor parameters and control the motor pumps from the mobile application itself. Data is transferred through internet without human to human or human to computer interaction. All the sensor parameters are stored in the IBM Cloudant DB.

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.

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.

1.1 PROJCT 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

CHAPTER 2

LITERATURE SURVEY

Sensor Based Smart Agriculture with IoT Technologies by M. Pyingkodi, K.Thenmozhi, K.Nanthini, M. Karthikeyan, Suresh Palarimath, V. Erajavign (2022) stated that IoT is a new technology trend used in almost every area thing, when connected to the internet and to each other, when you connect to the internet or interconnect, your entire system will be smarter. We have used IoT in all areas of our lives, including smart cities, smart homes, and smart retail. Much more. From 9.6 billion by 2050, agriculture needs to deliver even faster to meet this type of demand. This is possible with the latest technology, especially the IoT. The IoT enables labour free farms. The most significant tool for the IoT is the sensor. The important objective of sensors are used to determine the soil's physical qualities and the environment. The main applications of sensors are control and supervise, safety, alarm, diagnostics, and analytics. Sensors make innovative agriculture more effective and trouble-free. In agriculture, the sensor is mainly used for measuring, measuring NPK (Nitrogen, Phosphorus, Potassium) levels, and detecting disease and soil moisture content. The main solution to this problem is smart farming, which modernizes traditional farming practices. This paper narrates the role of IoT application in smart agriculture. Smart farming is also known as precision farming hence it uses accurate information to draw outcomes. It demonstrates the different sensors, applications, challenges, strengths and weaknesses that support the IoT and agriculture.

Smart Farm Monitoring Using LoRa Enabled IoT published by Ravi Kishore Kodali, Subbachari Yerroju, Shubhi Sahu (2019). Agricultural practices need to be transformed in order to overcome future food scarcity due to overpopulation across the globe. By employing emerging, disruptive technologies like IoT in the agricultural sector, it is possible to monitor farm fields using low-cost and low-power consuming devices, to automate irrigation systems for efficient usage of water resources. Weather forecast using IoT can help to plan farm filed activities like sowing, harrowing, harvesting, etc. This reduces negative impacts like yield losses due to uncertain weather changes. LPWAN technologies serve IoT applications in a better possible way so that these applications can overcome bandwidth, power and coverage constraints which are main drawbacks in other wireless communication technologies. In agricultural fields, LoRaWAN protocol or LoRa in LPWAN space gives additional advantages like scalability, security and robustness in designing IoT applications. In this paper, a smart farm monitoring model is proposed. This model utilizes LoRa communication mechanism to send sensor data like temperature (°C), humidity (%) and soil moisture (%) from the transmitter node to receiver node. The receiving node which is Wi-Fi enabled uses MQTT services to monitor the data in IBM Watson IoT platform and to store the same data in IBM cloud DB service.

Towards Smart Agriculture Monitoring Using Fuzzy Systems published by Noramalina Abdullah, Noor Aerina Binti Durani, King Soon Siong, Mohamad Farid Bin Shari, Vicky Kong Wei Hau, Wong Ngei Siong Ir Khairul Azman Ahmad (2020). Conventional farming is labor-consuming and the need to continuously monitor crops can be a burden for farmers. By realizing the concept of smart farming based on Internet of Things (IoT) technology, farmers can use a mobile application to observe and monitor air humidity, air temperature, and soil moisture-factors that can affect plant growth. Furthermore, the use of timers to control the pumps in conventional watering systems is not always practical in real-life cases. This paper proposes a framework that enables advanced fuzzy logic to control a pump's switching time according to user-defined variables, whereby sensors are the main aspect of and contributor to the system. Our proposed idea offers great potential for excellent performance as an interface between the sensors as the input and the IoT as the output medium. A comparison is made between the proposed system and manual handling. The results prove that the water consumption and watering time has been reduced significantly.

The world population growth is increasing the demand for food production stated in the paper named “A Systematic Review of IoT Solutions for Smart Farming” by Emerson Navarro,Nuno Cost,António Pereira(2020). Furthermore, the reduction of the workforce in rural areas and the increase in production costs are challenges for food production nowadays. Smart farming is a farm management concept that may use Internet of Things (IoT) to overcome the current challenges of food production. This work uses the preferred reporting items for systematic reviews (PRISMA) methodology to systematically review the existing literature on smart farming with IoT. The review aims to identify the main devices, platforms, network protocols, processing data technologies and the applicability of smart farming with IoT to agriculture. The review shows an evolution in the way data is processed in recent years. Traditional approaches mostly used data in a reactive manner. In more recent approaches, however, new technological developments allowed the use of data to prevent crop problems and to improve the accuracy of crop diagnosis.

2.1 REFERNCE:

[1] Divya J., Divya M.,Janani V.”IoT based Smart Soil Monitoring System for Agricultural Production” 2017.

[2] H.G.C.R.Laksiri, H.A.C.Dharmagunawardhana, J.V.Wijayakulasooriya ”Design and Optimization of loT Based Smart Irrigation System in Sri Lanka”2019 .

[3] Anushree Math, Layak Ali, Pruthviraj U” Development of Smart Drip Irriga- tion System Using IoT”2018.

[4] Dweepayan Mishra1, Arzeena Khan2 Rajeev Tiwari3, Shuchi Upadhay,” Automated Irrigation System-IoT Based Approach”,2018.

[5] R. Nageswara Rao, B.Sridhar,”IOT BASED SMART CROP-FIELD MONI- TORING AND AUTOMATION IRRIGATION SYSTEM”. 2018

[6] Shweta B. Saraf, Dhanashri H. Gawal,” IoT Based Smart Irrigation Monitoring and Controlling System”.2017

[7] Shrihari M”, A Smart Wireless System to Automate Production of Crops and Stop Intrusion Using Deep Learning” 2020.

[8] G. Sushanth1, and S. Sujatha”, IOT Based Smart Agriculture System”2018.

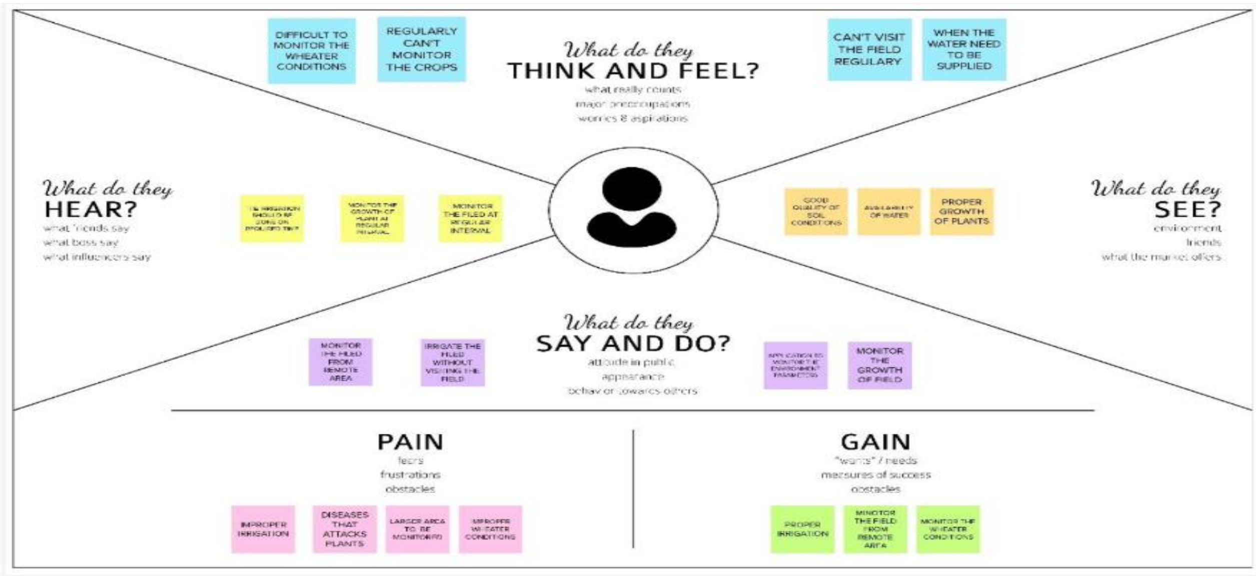
[9] Vaishali S, Suraj S, Vignesh G, Dhivya S and Udhayakumar S,” Mobile Integrated Smart Irrigation Management and Monitoring System Using IOT”,2017

CHAPTER 3
PROPOSED METHOD

3.1 EMPATHY 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

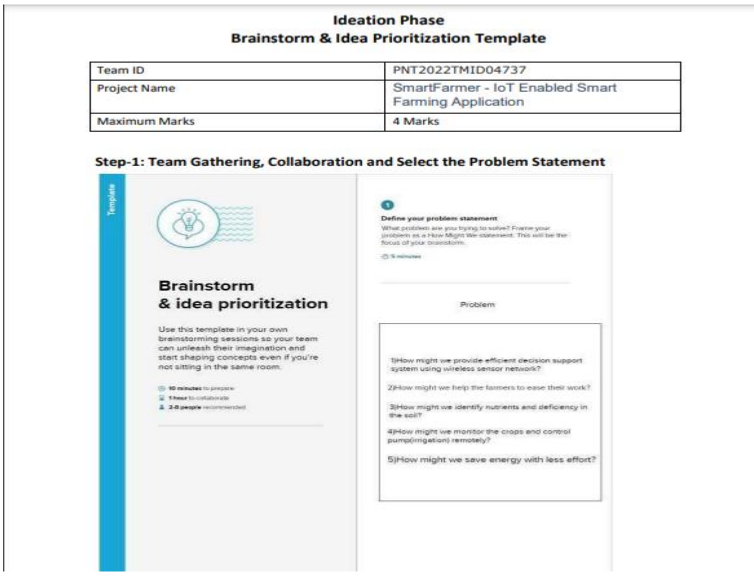
Empathy Map



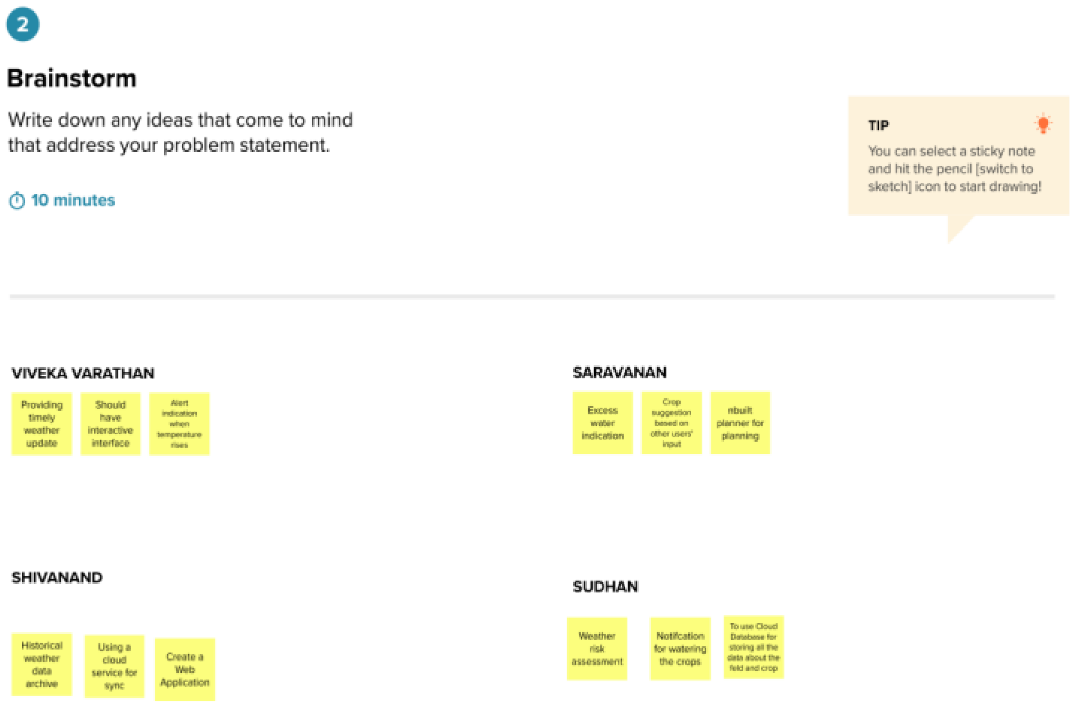
https://app.mural.co/invitation/mural/vivekavarathanr9950/1662825826196?sender=u895c87852a0cbe2d24a11594&key=cbf59e5f-763a-4859-b140-c1ca665311c4

3.2 IDEATION AND BRAINSTROMING:

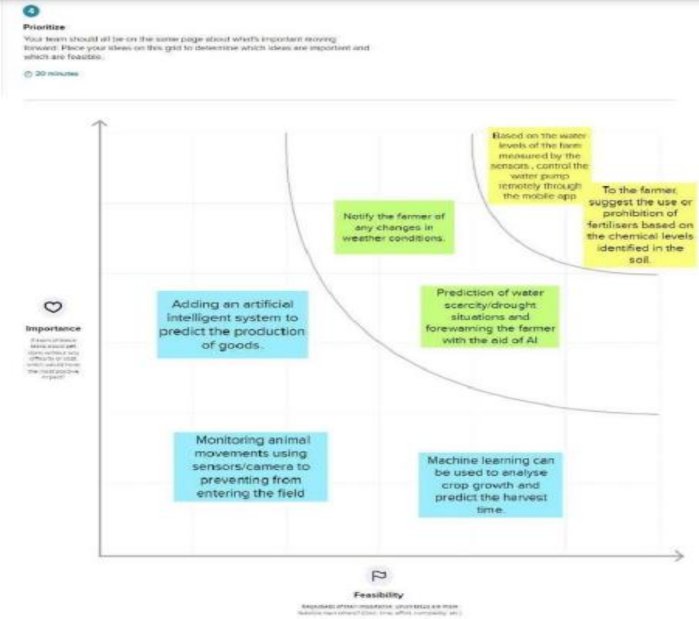
Step-1: Team Gathering, Collaboration and Select the Problem Statements:



Step-2: Brainstorm, Idea Listing and Grouping



Step-3: Idea Prioritization

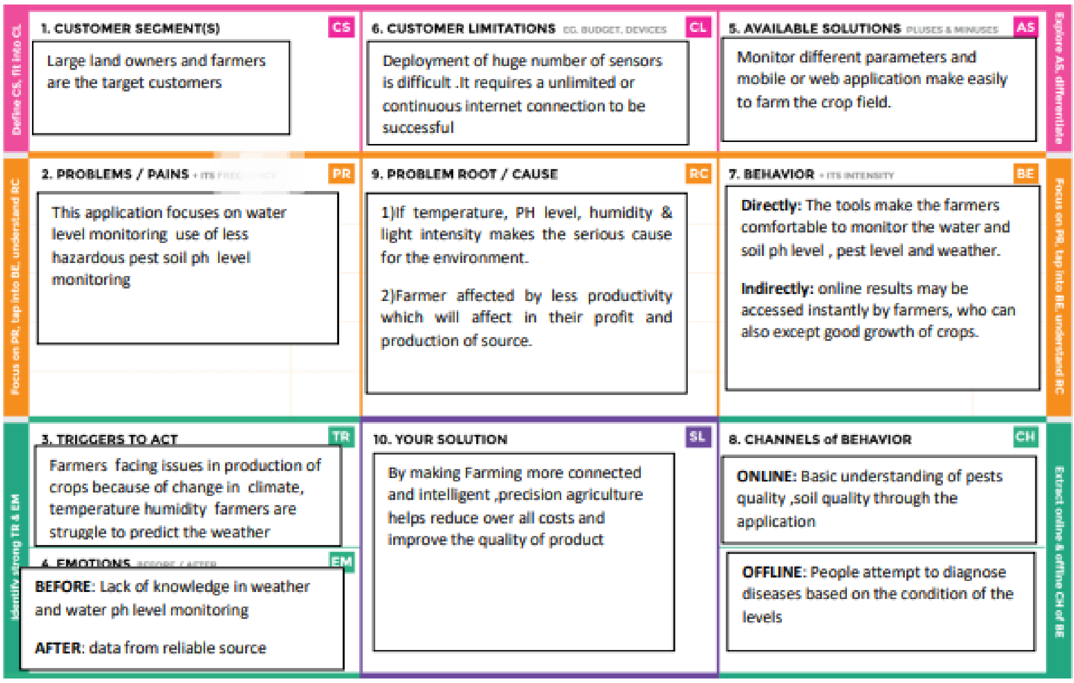


3.3 PROPOSED SOLUTION:

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	Utilizing an effective decision support system a network of wireless sensors that can manage various agriculture operations and provides vital information to temperature, humidity, and soil moisture content. Water level and weather conditions Farmers experience more distractions, which is not good favourable to agriculture.
2.	Idea / Solution description	As is the case of precision agriculture Smart farming technique enables farmers better to monitor the fields and maintain the humidity level accordingly. The Data collected by sensors, In terms of humidity, temperature, moisture, and dew detections help in determining the weather pattern in Farms. So cultivation is done for suitable crops
3.	Novelty / Uniqueness	Water being a precious resource must be utilized efficiently. Agriculture is one of those areas which consumes lot of water. Irrigation to the farm is a time consuming process and must be done on timely basis. As aimed, through this work an autoirrigation system measuring the moisture content, and the water level. Later harvesting the excess water from the cultivation field and recycled back to the tank.

4.	Social Impact / Customer Satisfaction	Smart farming, the dependency on manual labour has reduced significantly. The processes like pest control, fertilizing, and irrigation are increasingly becoming automated, and farmers can control them remotely. The use of smart IOT sensors can maintain these processes, increasing crop production.
5.	Business Model (Revenue Model)	It is trying to execute this technique as we need to introduce an arduino gadget which was modified with an Arduino that takes received signals from sensors. Easy operatability and maintenance. Required low time for maintain. Cost is reasonable.
6.	Scalability of the Solution	Scalability is another requirement that should be considered in a smart farming platform. Scalability refers to the ability to increase available resources and system capability without the need to go through a major system redesign orimplementation. We can increase the capacity for data processing by increasing the cloud resources inthe second layer and computation resources in the third layer. The challenges related to scalability in smart farming fall into two categories.They are Capacity and Performance.

3.4 SOLUTION FIT:



CHAPTER 4
REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through gmail/LinkedIn Mobile Application Via wifi
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Authentication	Using biometrics or PIN authentication to carry out some delicate app operations.
FR-4	Cloud Database	Database service on cloud
FR-5	Cloud Connectivity	Connecting Hardware to cloud

Above are the functional requirements of the proposed solution.

4.2 NON-FUNCTIONAL REQUIREMENTS:

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

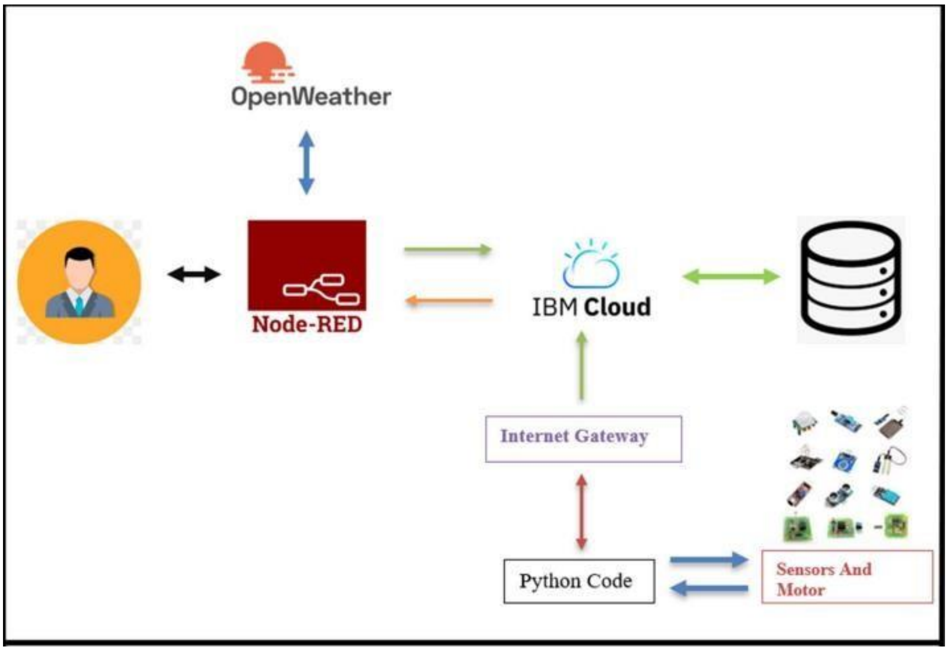
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The dashboard needs to be clear, uncluttered, and adaptable.
NFR-2	**Security**	Authentication - The user have a private dashboard for secured access.
NFR-3	**Reliability**	High-quality sensors were employed to provide long-lasting high precision and accuracy.
NFR-4	**Performance**	Performance can be enhanced by employing efficient sensors and developing efficient code.

CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW 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.



1. The different soil parameters temperature, soil moistures and then humidity are sensed using different sensors and obtained value is stored in the IBM cloud.
2. Arduino UNO is used as a processing Unit that process the data obtained from the sensors and whether data from the weather API.
3. NODE-RED is used as a programming tool to write the hardware, software and APIs. The MQTT protocol is followed for the communication.
4. All the collected data are provided to the user through a mobile application that was developed using the MIT app inventor. The user could make a decision 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 AND TECHNICAL ARCHITECTURE:

• The various factors like temperature, soil moisture level, humidity are measured using various sensor and are stored in IBM cloud.

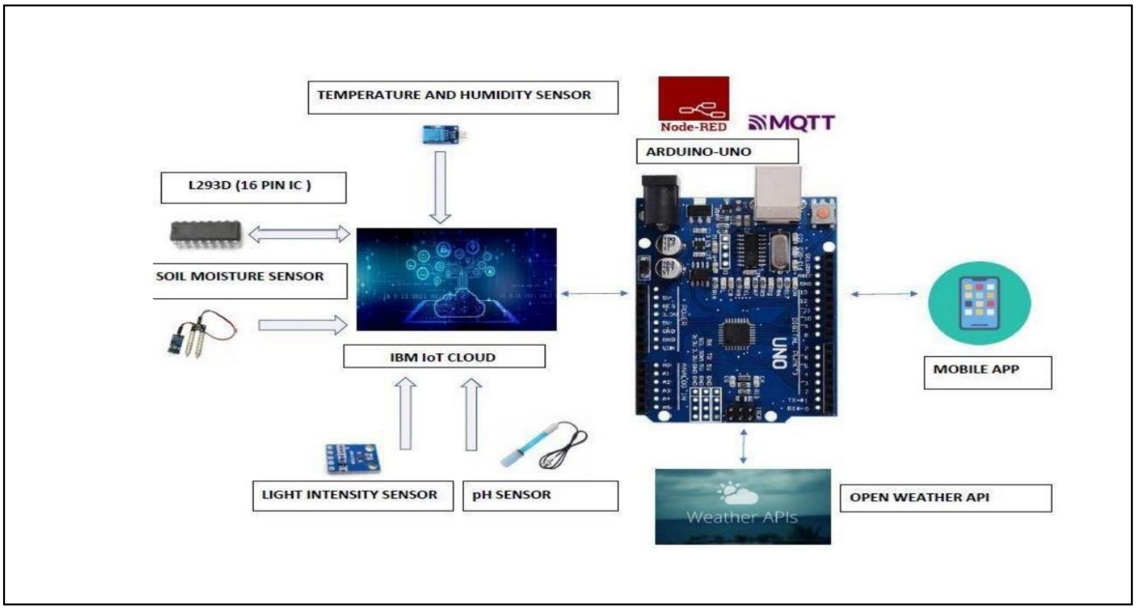
• The Arduino UNO is to process the data received from the sensor and weather from weather API.

• Node-RED is a programming tool for wiring the hardware devices, API and online services.

• MIT app Invertor is a visual programming environment that allows everyone to build fully functional apps.

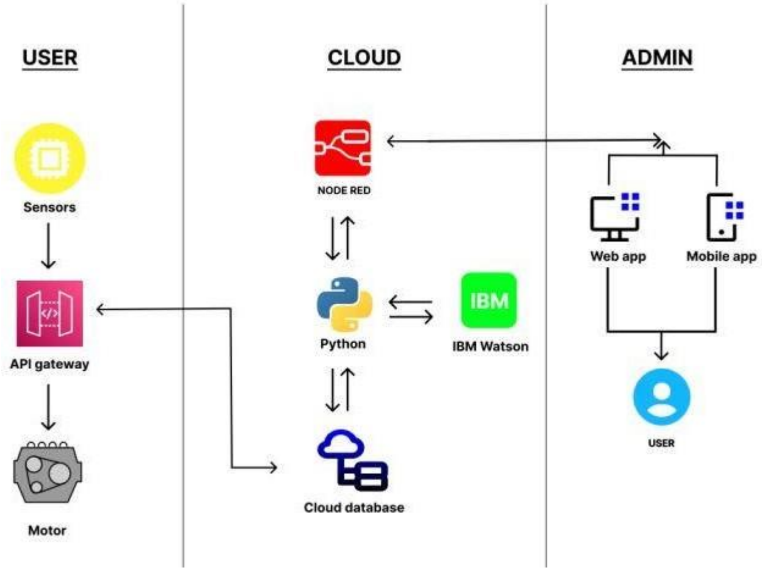
• Extra care for plants to improve growth and save water for dry conditions.

Solution Architecture Diagram:



Technical Architecture

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



Components & Technologies:

S.No	Component	Description	Technology
1	User Interface	Mobile application	MIT app inventor
2	Reading Parameters	Reading Field parameters like Soil moisture, Humidity, Temperature	Various Sensors nodes
3	Hardware device	For use of sensor and value	Microcontroller/ Microprocessor Board
4	Cloud connectivity	Connecting hardware to cloud	Wifi module or ESP32
5	Cloud Database	Database Service on Cloud	IBM Watson
6	External API-1	Purpose of External API used in the application	IBM Watson API

Application Characteristics:

S.No	Characteristics	Description	Technology
1	Open-Source Frameworks	List the open-source frameworks used	Python
2	Security Implementations	List all the security / access controls implemented, use of firewalls etc.	User login Credentials
3	Scalable Architecture	Every Cloud Services are hosted separately and make is scalable separately	IBM auto scaling
4	Availability	To make use the application and data are available 24/7	IBM cloud load balancer
5	Performance	To increase the performance the application in hosted in the high-performance instance	Can handle connected sensors data and Network connectivity simultaneously

5.3 USER STORIES:

Title	Description	Date
Brainstrom & Idea prioritization	Layout the problems that want to solve, from that shortlist the top 3 problems & getting appropriate ideas for that problems	19 September 2022
Literature Survey	Gathering information about the project topic from referring the technical papers, research publications etc…	20 September 2022
Empathy Map	Empathy Map is prepared to know about the user(agriculturist) Pain& Gain and their problems then their view about this process	20 September 2022
Proposed Solution	Proposed solution is the document, which includes the problem statement, ideas for the problems, social impact, business model, scalability of solution etc…	20 September 2022
Problem Solution Fit	In this have the customer segment, before and after the process, problem having available solutions and our solution for that problems etc…	21 September 2022
Solution Architecture	The solution will be in the form of architecture, it includes functional & nonfunctional.	21 September 2022
Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application	21 September 2022

Data Flow Diagrams	Draw the data flow Diagrams and submit for review.	03 November 2022
Technology Architecture	Architecture diagram.	03 November 2022
Sprint Delivery	Prepare the Sprint delivery on Number of Sprint planning meetings organized, Minutes of meeting recorded.	14 November 2022
Milestone & Activity List	Prepare the milestones & Activity list of the project.	14 November 2022
Project Development Delivery of Sprint- 1,2,3&4	Develop & submit the developed code by testing it.	IN PROGRESS…

CHAPTER 6

PROJECT PLANNING & SCHEDULING

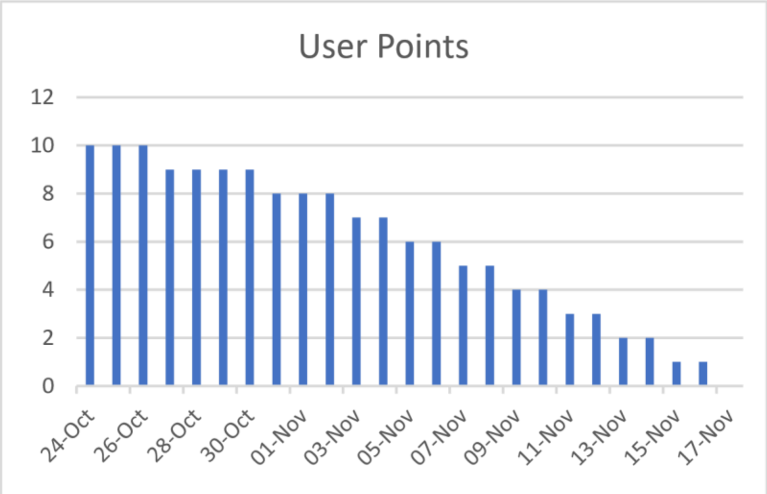
6.1 SPRINT PLANNING & ESTIMATING:

Sprint	Functional requirement	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my Gmail, email or through phone number then you can received the OTP or Verification Code.	10	High	Viveka varathan.R
Sprint-1	Confimation	USN-2	As a user, I will receive confirmation Gmail once I have registered for the application.	4	Low	Sudhan A
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	6	Medium	Shivavand K
Sprint-2	Simulation	USN-4	Connect sensors and ESP 32	4	Low	Saravanan S
Sprint -2	Software	USN-5	Develop a python code to publish random sensor data	6	Medium	Shivavand K
Sprint-2	Simulation	USN-6	Connect the data with IBM cloud	10	High	Shandheep E
Sprint-3	Simulation	USN-7	Establishing Node-RED connection	8	Medium	Sudhan A
Sprint-3	App development	USN-8	Application development using MIT app inventor	12	High	Viveka varathan. R
Sprint-4	Simulation	USN-9	Connecting the developed application with NodeRED	6	Medium	Sudhan A
Sprint-4	Testing	USN-10	Testing the application	12	High	Shivanand K

6.2 SPRINT DELIVERY SCHEDULE:

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

6.3 REPORT FROM JIRA:



CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE 1:


```
import wiotp.sdk.device
import time
import random
myConfig = {
"identity": {
"orgId": "m5ttid",
"typeId": "k3",
"deviceId": "123456"

},
"auth": {
    "token": "12345678"
}
}
client = wiotp.sdk.device.DeviceClient (config=myConfig,
logHandlers=None)
client.connect ()
def myCommandCallback (cmd) :
    print ("Message received from IBM IoT Platform: %s" %
    cmd.data['command'])
    m=cmd.data['command']
    if (m=="motoron"):
        print ("Motor is switched on")
    elif (m=="motoroff"):
        print ("Motor is switched OFF")
    print (" ")
while True:
    moist=random.randint (0,100)
    temp=random.randint (-20, 125)
    hum=random.randint (0, 100)
    myData={'soilmoisture':moist, 'temperature':temp, 'humidity':hum}
    client.publishEvent (eventId="status", msgFormat="json",
    data=myData, qos=0 , onPublish=None)
    print ("Published data Successfully: %s", myData)
    time.sleep (2)
    client.commandCallback = myCommandCallback
client.disconnect()
```

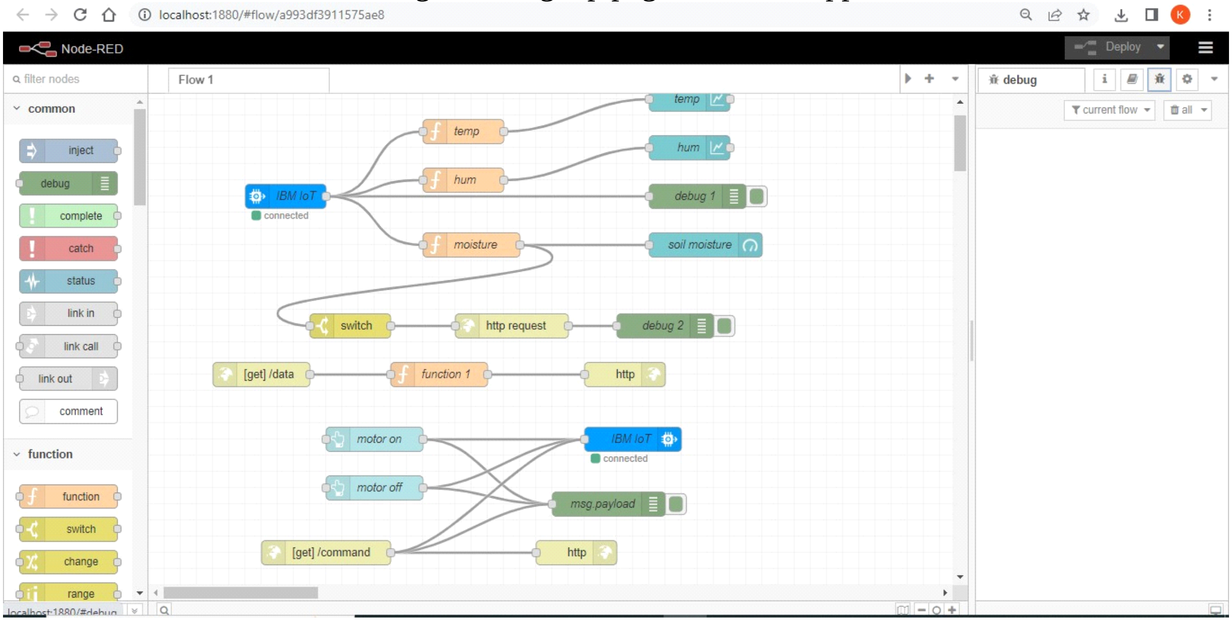
```
import wiotp.sdk.device

import time
import os
import datetime
import random
 myConfig ={
"identity": {
"orgId": "m5ttid",
"typeId": "k3", "deviceId": "123456"
},
"auth": {
"token": "12345678"
}
```

```
}
client=wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
client.connect ()
def myCommandCallback (cmd) :
print("Message received from IBM IoT Platform: %s" %cmd.data['command'])
m=cmd.data['command']
if (m=="motoron"): print("Motor is switchedon")
elif (m=="motoroff"):
print ("Motor is switchedOFF") print (" ")
while True:
moist =random.randint (0,100)
temp=random.randint (-20, 125)
hum=random.randint                    (0,                    100)
myData={'moisture':moist,'temperature':temp,'humidity':hum}
client.publishEvent (eventId="status", msgFormat="json", data=myData, qos=0
,
onPublish=None)
print ("Published data Successfully: %s",myData) time.sleep (2)
client.commandCallback =myCommandCallback
client.disconnect ()
```

7.2 FEATURE 2:

These are the blocks of the login and signup page of mobile application.

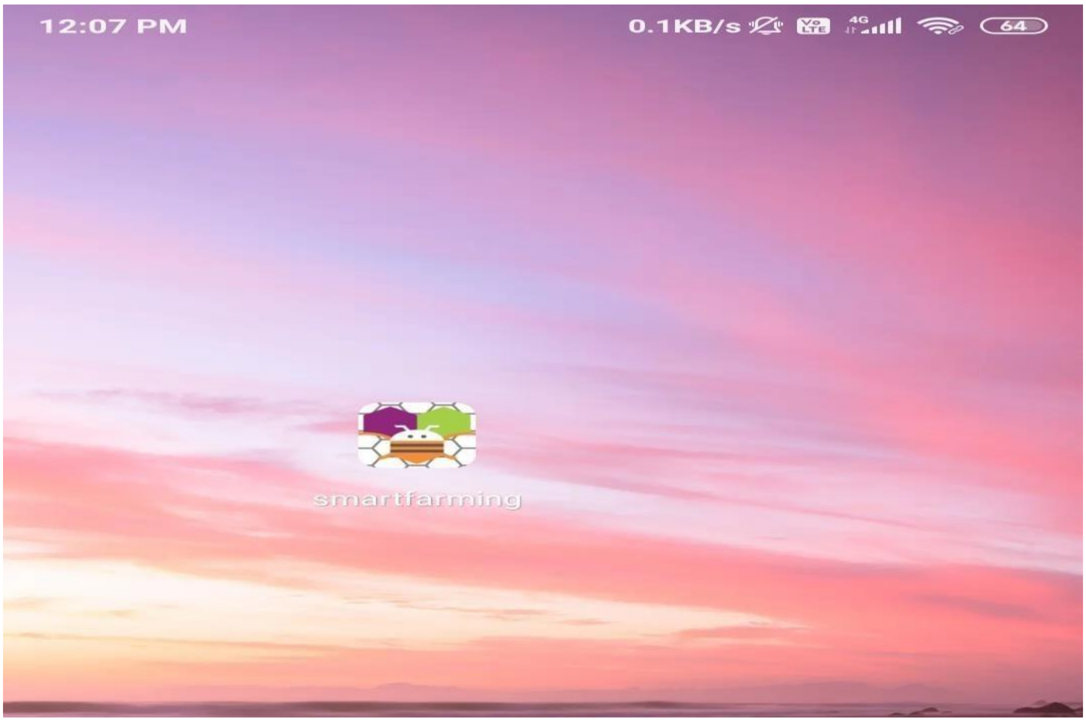


CHAPTER 8
TESTING

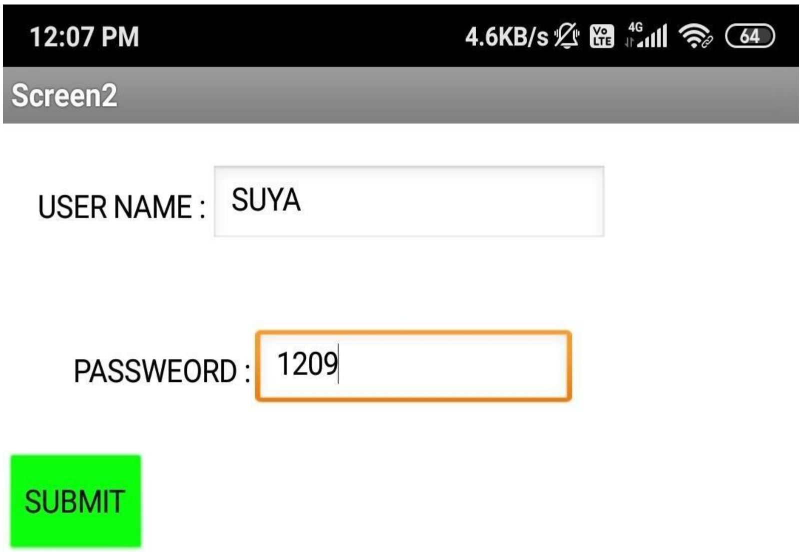
8.1 TEST CASES:

Step-1: First user need to download the android APK file from MIT app inventor where we developed our mobile application and install in their mobiles.

Step-2: After successful installation we can find app icon in our mobile as shown below.

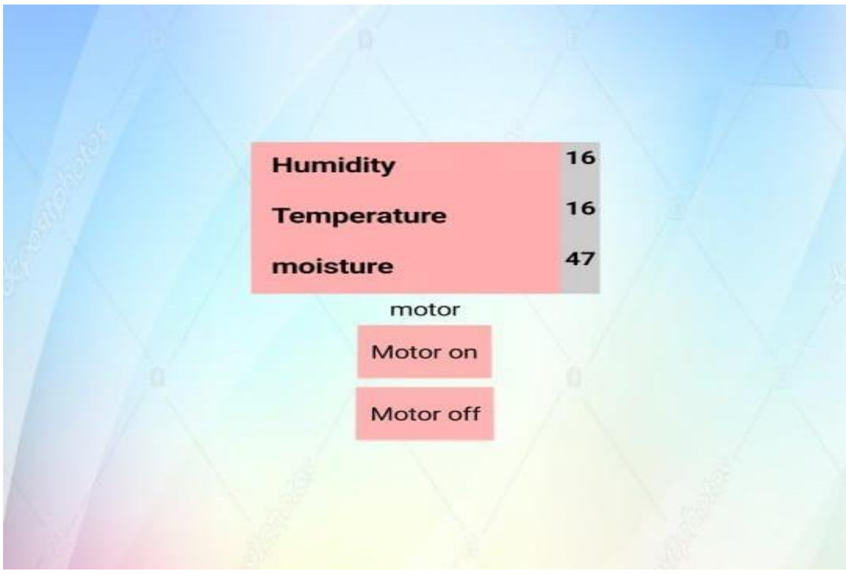


Step-3: After clicking the app icon it ask the user need to create username and password.so give username and password and click the signup button. The user can see interface like these as shown below.



8.2 USER ACCEPTANCE TESTING:

After successful login. The next page will be open. In that page we can see the real time temperature, humidity and soil moisture reading and motor ON and motor OFF control button also as shown below.

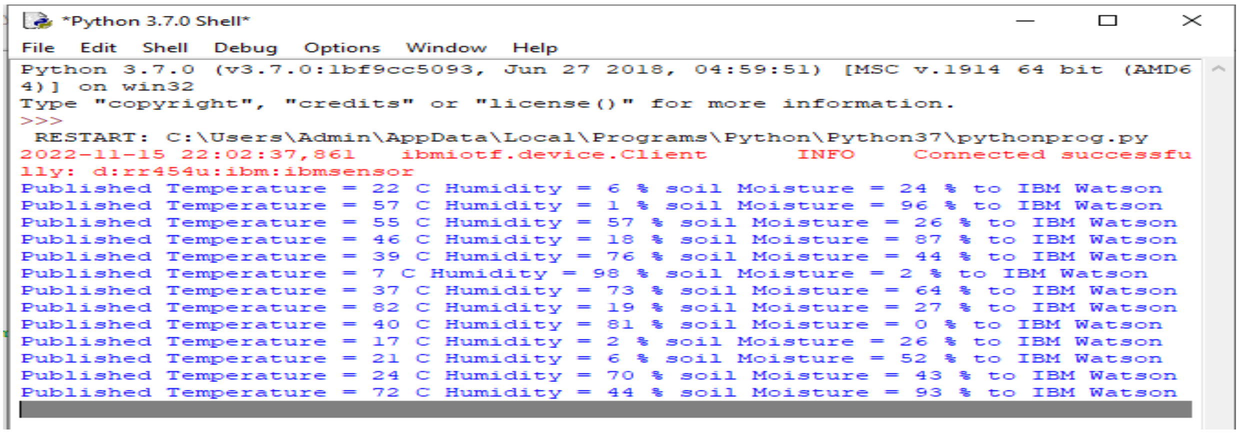


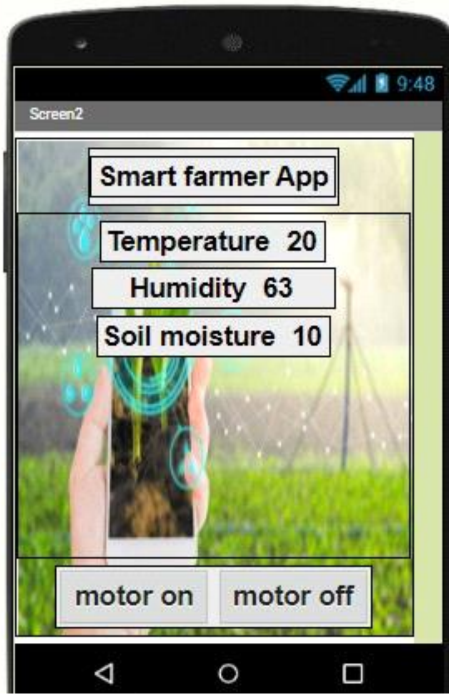
we are successfully created the IOT enabled smart farming application.

CHAPTER 9
RESULTS

9.1 PERFORMANCE METRICES:

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.





CHAPTER 10

ADVANTAGES & DISADVANTAGES

10.1 ADVANTAGES:

- All the data like climatic conditions and changes in them, soil or crop conditions everything can be easily monitored.
- Risk of crop damage can be lowered to a greater extent.
- Many difficult challenges can be avoided making the process automated and the quality of crops can be maintained.
- The process included in farming can be controlled using the web applications from anywhere, anytime.

10.2 DISADVANTAGES:

- Smart Agriculture requires internet connectivity continuously, but rural parts cannot fulfill this requirement.
- Any faults in the sensors can cause great loss in the agriculture, due to wrong records and the actions of automated processes.
- IoT devices need much money to implement.

CHAPTER 11

CONCLUSION AND FUTURE SCOPE

11.1 CONCLUSION:

So finally we build A IoT Web Application for smart agricultural system using Watson IoT platform, Watson simulator, IBM cloud and Node-RED and MIT app Inventor

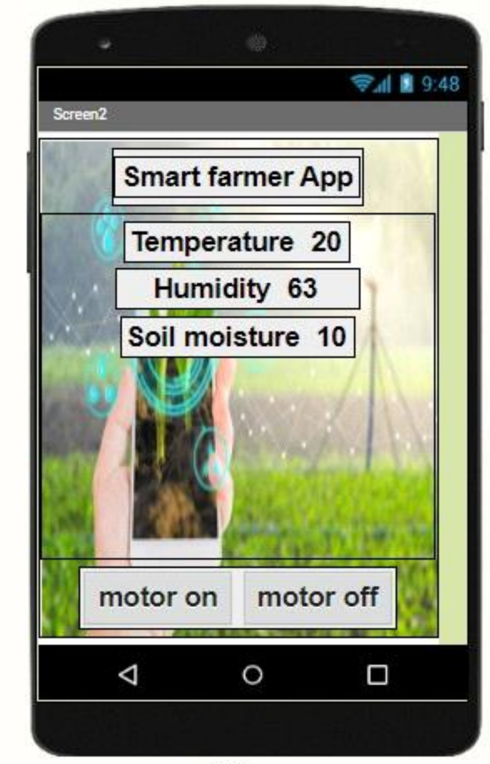
11.2 FUTURE SCOPE:

In future due to more demand of good and more farming in less time, for betterment of the crops and reducing the usage of extravagant resources like electricity and water IoT can be implemented in most of the places.

CHAPTER 12

APPENDIX

12.1 SOURCE CODE:




```
/*....................................retrieving to Cloud.............................*/

void PublishData(float temperature, float humidity) {
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data to ibm cloud
  */
  String payload = "{\"temperature\":";
  payload += temperature;
  payload += "," "\"Humidity\":";
  payload += humidity;
  payload += "}";


  Serial.print("Sending payload: ");
  Serial.println(payload);


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will print publish
ok in Serial monitor or else it will print publish failed
  } else {
```

```
    Serial.println("Publish failed");
  }

}
```

```
void mqttconnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!!!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }

     initManagedDevice();
     Serial.println();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

void initManagedDevice() {
  if (client.subscribe(subscribetopic)) {
    Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{

  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++) {
    //Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }
  Serial.println("data: "+ data3);
  if(data3=="lighton")
  {
```

```
Serial.println(data3);
digitalWrite(LED,HIGH);
  }
  else
  {
Serial.println(data3);
digitalWrite(LED,LOW);
  }
data3="";
}
```

GitHub:

Name	GitHub (User Name)
Team Leader(VIVEKA VARATHAN R)	Vivek15122001
Team Member(SHIVANAND K)	shiva2002k
Team Member(SUDHAN A)	SudhanAnand2002
Team Member(SARAVANAN S)	saravananbe

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-22117-1659805313