IoT-ENABLED SMART FARMING APPLICATION

Team No: PNT2022TMID34723

Team Leader: Femi Sterlin Dhas A

Team Member: Abiya Brittolin M, Abitha K, Aswini D B

Project Report Format

1. **INTRODUCTION**
    a. Project Overview
    b. Purpose
2. **LITERATURE SURVEY**
    a. Existing problem
    b. References
    c. Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
    a. Empathy Map Canvas
    b. Ideation & Brainstorming
    c. Proposed Solution
    d. Problem Solution fit
4. **REQUIREMENT ANALYSIS**
    a. Functional requirement
    b. Non-Functional requirements
5. **PROJECT DESIGN**
    a. Data Flow Diagrams
    b. Solution & Technical Architecture
    c. User Stories
6. **PROJECT PLANNING & SCHEDULING**
    a. Sprint Planning & Estimation
    b. Sprint Delivery Schedule
7. **CODING & SOLUTIONING** 8. **TESTING**
    a. Test Cases
    b. User Acceptance Testing
9. **RESULTS**
    a. Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**

11.**CONCLUSION** 12. **FUTURE SCOPE**

13.**APPENDIX** Source

Code

GitHub & Project Demo Link

1.INTRODUCTION

Most important factors for the quality an productivity of plant growth are temperature, humidity and light. Continuous monitoring of these environmental variables provides valuable information to the grower to better understand, how each factor affects growth and how to maximize crop productiveness. WSN composed of hundreds of nodes which have ability of sensing, actuation and communicating, has great advantages in terms of high accuracy, fault tolerance, flexibility, cost, autonomy and robustness compared to wired ones**.**

1.1 PROJECT OVERVIEW

The objectives of this report is to proposed IoT based Smart Farming System which will enable farmers to have live data of soil moisture environment temperature at very low cost. IOT application in agriculture, benefits of 10T in agriculture and IOT and agriculture current scenario and future forecasts.

1.2 PURPOSE

Though on a superficial scale it seems unpredictable, it can be determined with certain parameters with which crop planning can be done.Maintenance of farm fields during and after cultivation are also important. These can be performed by measuring soil moisture, humidity and temperature.IoT-based agriculture system helps the farmer in monitoring different parameters of his field like soil moisture, Temperature, humidity using some sensors. Farmers can monitor all the sensor parameters by using a web or mobile application even if the farmer is not near his field.

2.LITERATURE SURVEY

Agriculture is essential to India's economy and people's survival. The purpose of this project is to create an embedded-based soil monitoring and irrigation system that will reduce manual field monitoring and provide information via a mobile app. The method is intended to help farmers increase their agricultural output. A pH sensor, a temperature sensor, and a humidity sensor are among the tools used to examine the soil. Based on the findings, farmers may plant the best crop for the land. The sensor data is sent to the field manager through Wi-Fi, and the crop advice is created with the help of the mobile app. When the soil temperature is high, an automatic watering system is used. The crop image is gathered and forwarded to the field manager for pesticide advice.Internet of Things technique in irrigation for the purpose to save water. In this paper states that Soil constitution is related with the availability of elements of nourishment plant requires as well as the presence in soil of elements and chemical composition that exist at different proportion that are best nourishment to plants and soil organisms and appropriate water to plant is most essential for all of the other nourishment to work at best.

2.1 EXISTING PROBLEM

There has been several attempts and solution to help farmers adopt technological practices. Fewsolutions restricted their performance with just suggestions and alerts. While few employed IoT independent electronics. Few of the cases of previous attempts and researches are described below.

i. “IoT based smart sensors agriculture stick for live temperature and moisture monitoring using Arduino, cloud computing & solar technology”. This work was performed using Cloud computing platform (Things Speak) for data acquisition. The circuit was designed using Arduino and DHT 11 sensors. ii. “Smart Farming using IoT, a solution for optimally monitoring conditions”. This work used ESP-32 based IoT platform and Blynk mobile application. iii. “Smart farming using IoT”. The automation and interface part made use of water pump and HTTP protocol for parameters monitoring using website.

2.2 REFERENCE

1.International Journal of Innovations in Engineering and Science, Impact Factor Value 4.046 e-ISSN: 24563463 Vol.4, No. 5, 2019 “Solar Powered Smart Fencing System for Agriculture Protection using GSM & Wireless Camera”.

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: 24555703 “Design and Implementation of an Advanced Security System for Farm Protection from Wild Animals

3. Divya J., Divya M.,JananiV.”IoT based Smart Soil Monitoring System for Agricultural Production” 2017.

4. Nikesh Gondchawar and Prof. Dr.R.S.Kawitkar, “IoT Based Smart Agriculture,” International Journal of Advanced Research in Computer and Communication Engineering, Vol. 5, Issue 6, June2016.

5. Dweepayan Mishra1 ,Arzeena Khan2 Rajeev Tiwari3 , Shuchi Upadhay,”Automated Irrigation SystemIoT Based Approach”,2018. [6] R. Nageswara Rao, B.Sridhar,”IOT BASED SMART CROP-FIELD MONITORING AND AUTOMATION IRRIGATION SYSTEM”. 2018.

2.3 PROBLEM STATEMENT DEFINITION

The problem statement in a nutshell covers all the possible technical aspects that can be included by farmer to convert farming in to smart and efficient farming. IoT enabled smart farming, on a wider perspective, concentrates on connecting all the independently operating sub-systems in farming automation into a single entity. IoT-based agriculture system helps the farmer in monitoring different parameters of his field like soil moisture, temperature, and humidity using some sensors.A strong customer problem statement should provide a detailed description of your customer’s current situation. Consider how they feel, the financial and emotional impact of their current situation, and any other important details about their thoughts or feelings.Creating a customer problem statement is easy with Miro. Using our collaborative online whiteboard, you can create an online problem statement that’s easy to follow and shareable with your team. All you have to do is sign up for free, select this template, and follow your templateThe idea of IoT is further extended with the help of mobile and web application where farmers can monitor all the sensor parameters even if the farmer is not near his 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.

3.IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviors and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem ant 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 & BRAIN STORMING

TEAM IDEAS

FEMI STERLIN DHAS A

Automate irrigation process using temperature of soil.

Automate irrigation using measurement of moisture of soil

ABIYA BRITTOLIN M

A website guide about the product and features

A website that describes the features of the application, also provides important advices in farming regarding optimum conditions, crop health , use of resources etc.

ABITHA K

We can sense and program the moisture level

We can simplify the drip irrigation into time controlled irrigation.

Automate irrigation using any Robots.

ASWINI D B

We can make good design and programming of soil moisture and temperature.

We can automate and design Audino for programming.

IDEAS FROM THE BRAIN STORMING

FEMI STERLIN DHAS A

User friendly application for farmland parameter Awareness about IoT in agricultural domain through various applications

ABIYA BROTTOLIN M

Usage of raspberry pi over other processor for IoT applications. Deciding the specification of sensors to be deployed based on the range of operation per square feet of farmlands.

ABITHA K

A website guide about the product and features .Provision through application to remotely control agricultural instruments

ASWINI D B

Various protocols used in IoT. Application with simple UI but efficient usage. Interface between website, application and sensors. Utilizing ai to improvise accuracy.

3.3 PROPOSED SOLUTION

S.No	**Parameter**	Description
1.	Problem Statement (Problem to besolved)	To incorporate the process of working andalso elevate the smart farming using IOT enabled smart Farming technique since the traditional Farming technique I very Complex one.
2.	Idea /Solution description	To automate irrigation in accordance to theamount of moisture present in soilAutomation of irrigation to amount of moisture

3.	Novelty / Uniqueness	Automation of irrigation to amount of moisture.
4.	Social Impact / Customer Satisfaction	The problems faced by the farmers in the process of irrigation gets solved and this full filsand saves their crops from over irrigation
5.	Business Model (Revenue Model)	The process of fulfilling this process brings revolution in drip irrigation systems also makesa revolutionary change in market
6.	Scalability of the Solution	The design scale of solution has been plannedin a compact manner

3.4 PROBLEM SOLUTION FIT

4. REQURIMENT ANALYSIS

4.1FUNCTIONAL REQURIMENT

FR.No	Functional Requriment (Epic)	Sub Requriment (Story/Sub Task)
FR -1	IoT Devices	Sensors and wifi module
FR-2	Software	Web UI,Node -Red, IBM-watson, MIT app
FR-3		
FR-3		

4.2 NON FUNTIONAL REQURIMENT

Following are the non-functional requirements of the proposed solution

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Time consumability is less,Increasd production.
NFR-2	Security	It has low level of security features due to integration of sensor data.
NFR-3	Reliability	Accuracy of data and hence it is Reliable.
NFR-4	Performance	Performance is high and productivity is high.
NFR-5	Availability	With permitted network connectivity the application is accessible.
NFR-6	Scalability	It is perfectly scalable with many new constraints can be added.

5-PROJECT DESIGN

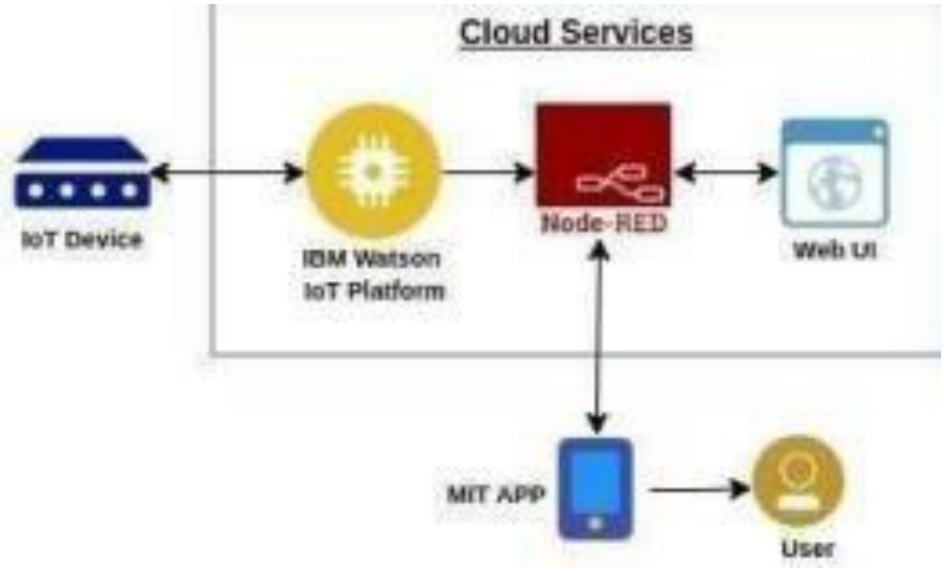
5.1 Data Flow Diagrams

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.

2. 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.

3. 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 remotely operate the motor switch.



5.2 Solution & Technical Architecture

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

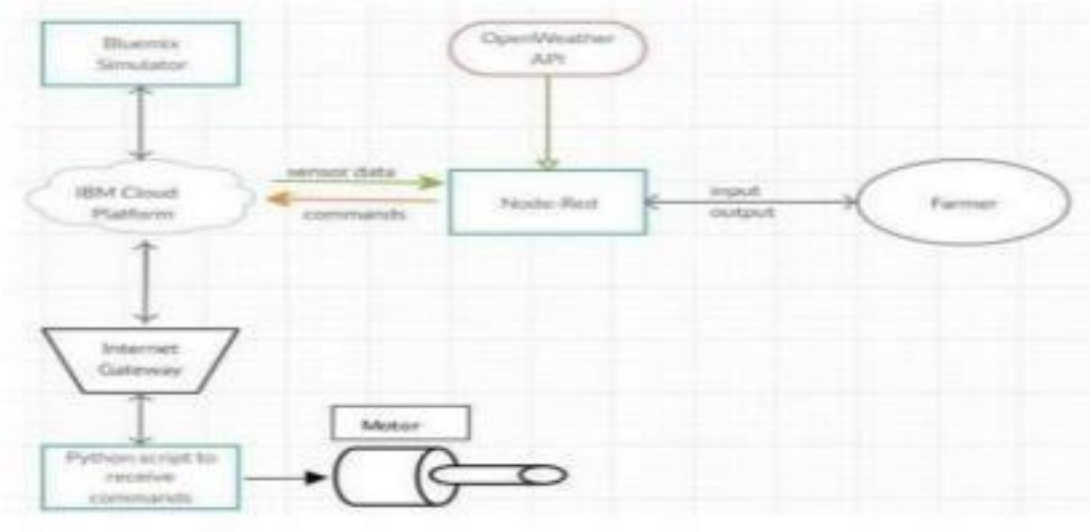
1. The different soil parameters temperature, soil moistures and then humidity are sensed using different sensors and obtained value is stored in the IBM cloud.

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 decide through an

app, weather to water the crop or not depending upon the sensor values. By using the app, they can remotely operate the motor switch.



6. PROJECT PLANNING & SCHEDULING

6.1 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	Registration (Farmer Mobile User)	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password..	3	High	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin .M,Aswini.D.B

Sprint-1	Login	USN-2	As a user, I will receive confirmation email once I have registered for the application	3	High	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin .M,Aswini.D.B
Sprint-1	Data Visualization	USN-4	As a user, I can register for the application through Gmail	2	Low	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin .M,Aswini.D.B
Sprint-1	Registration (Chemical Manufacturer - Web user)	USN-1	As a new user, I want to first register using my organization email and create a password for the account.	4	Medium	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin .M,Aswini.D.B
Sprint-2	User Interface	USN-1	As a user, I can register	3	High	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin .M,Aswini.D.B
			for the application through Face book.			
Sprint-2	Login	USN-2	As a registered user, Indeed to easily login log into my registered account via the web page in minimum time	3	High	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin .M,Aswini.D.B

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**

Sprint-3	Registration (Farmer -Web User)	USN-1	As a user, I can log in to the application by entering email and password	3	High	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin. M,Aswini.D.B
Sprint-3	Web UI	USN-3	As a user, I need to have a user friendly interface to easily view and access the resources.	3	Medium	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin. M,Aswini.D.B
Sprint-4	Web UI	USN-3	As a user, I need to have a friendly user interface to easily view and access the resources	3	Medium	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin. M,Aswini.D.B
Sprint-4	Login	USN-2	As a registered user, I need to easily log in using the registered account via the webpage.	3	High	Femi Sterlin Dhas.A,Abitha.K,Abiya Brittolin. M,Aswini.D.B

Project Tracker, Velocity & Burndown Chart:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint1	12	6 Days	29 Oct 2022	25 Oct 2022	20	19 Oct 2022
Sprint2	6	6 Days	26 Oct 2022	1 Nov 2022	20	27 Oct 2022

Sprint3	6	6 Days	02 Nov 2022	08 Nov 2022	20	04 Nov 2022
Sprint4	6	6 Days	09 Nov 2022	15 Nov 2022	20	13 Nov 2022

7. CODING & SOLUTIONING

7.1 Feature 1 import wiotp.sdk.device import time

import os import datetime import random myConfig

={

"identity": {

"orgId": "0hzydu",

"typeId": "NodeMCU",
"deviceId": "12345"

},

"auth": {

"token": "12345678"

} } client
= wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None) client.conne ct
() 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

()

7.1 Feature 2

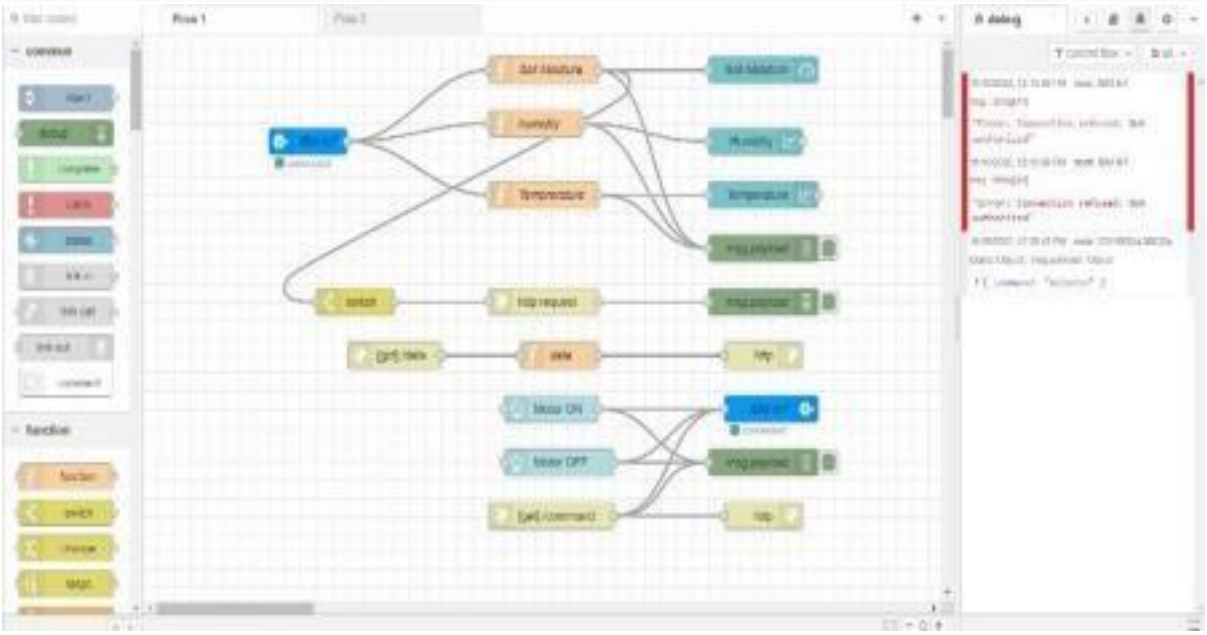
```
/*

Plant Watering System



The circuit:

- Water pump

Power supply: 4.5~12V DC Interface: Brown +; Blue -
- Temperature/moisture sensor Power supply: 3.3-5v

- Moisture sensor Power supply: 3.3-5v

*/

#include "DHT.h"

#define DHTPIN 2// what digital pin we're connected to #define DHTTYPE  DHT22 //
DHT 22 (AM2302), AM2321



DHT dht(DHTPIN, DHTTYPE);
const int SOIL_MOISTURE_SENSOR_PIN = A0; const int  WATER_PUMP_PIN = 4;



const int dry = 520; const int wet = 270;  const

int moistureLevels = (dry - wet) / 3;



 // TODO: Should we have a counter so if it waters for X times, then take a break?
```

```
// OPTIMIZE: how dry to start watering and for how long.
const int  soilMoistureSartWatering = 400;  const int
soilMoistureStopWatering = 300;

// 60 seconds  const long waterDuration

= 1000L * 60L;

// 60 seconds  const long sensorReadIntervals

= 1000L * 60L;

// 2 hr

const long waterIntervals = 1000L * 60L * 60L * 2; long lastWaterTime = -
waterIntervals - 1;  boolean isWatering = false;
void setup()

{ Serial.begin(9600); pinMode(WATER_PUMP_PIN, OUTPUT);



waterPumpOff(); dht.begin();



}



void loop()

{ mainLoop ();

}



void mainLoop() {  float temperature =

getTemperature(); float humidity = getHumidity(); long
```

```
soilMoisture =

analogRead(SOIL_MOISTURE_SENSOR_PIN);

Serial.println("Soil Moisture: " + readableSoilMoisture(soilMoisture) + ", " +
soilMoisture);

Serial.println("Temperature: " + String(temperature) + "   *F");Serial.println("Humidity:

" + String(humidity) + " %");



if (millis() - lastWaterTime > waterIntervals)

{waterPlants(soilMoisture); lastWaterTime = millis();

}

delay(sensorReadIntervals);

}
void waterPlants(int soilMoisture) {

// Should this take a moving avg of the soilMoisture?


// Can get outliers on the right after watering. if (soilMoisture
>  soilMoistureSartWatering)

{ isWatering = true
```

8. TESTING

8.1 Test Cases



8.2 User Acceptance Testing

9. RESULTS

9.1 PERFORMANCE METRICS

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 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 Farming 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 farmer to whether to do Smart farming 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 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.

- 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 an entire.

- We can update 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 this project.

- 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 an 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 wiotp.sdk.device

import time   import os

import datetime

import random myConfig ={
```

"identity": {

"orgId": "0hzydu",

"typeId": "NodeMCU",

"deviceId": "12345"

},

"auth": {

"token": "12345678"

}

} client = wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None) client.connec t () def myCommandCallback (cmd) :

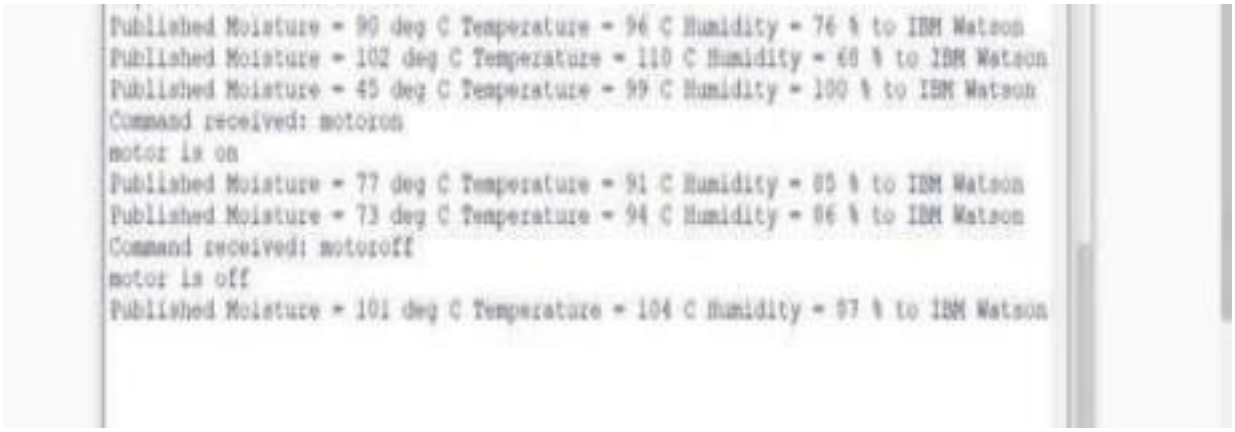
print("Message received from IBM IoT Platform: %s"

%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.publishEve nt (eventId="status", msgFormat="json", data=myData, qos=0 , onPublish=None) print ("Published data Successfully: %s",myData) time.sleep (2) client.commandCallback =myCommandCallback client.disconnect ()

OUTPUT:


Published Moisture = 90 deg C Temperature = 96 C Humidity = 76 % to IBM Watson
Published Moisture = 102 deg C Temperature = 110 C Humidity = 68 % to IBM Watson
Published Moisture = 45 deg C Temperature = 99 C Humidity = 100 % to IBM Watson
Command received: motoron
motor is on
Published Moisture = 77 deg C Temperature = 91 C Humidity = 85 % to IBM Watson
Published Moisture = 73 deg C Temperature = 94 C Humidity = 86 % to IBM Watson
Command received: motoroff
motor is off
Published Moisture = 101 deg C Temperature = 104 C Humidity = 97 % to IBM Watson