PROJECT REPORT
ON

Smart Farmer - IoT Enabled Smart Farming Application

TEAM ID: PNT2022TMID34687

TEAM MEMBERS:

Sweetlin Derisha S - 962219104111 (Team Leader)

Roshini A R – 962219104091 (Team Member)

Sherin Davisha D – 962219104101 (Team Member)

Sowmiya A – 962219104108 (Team Member)

INDEX

1.INTRODUCTION

1.1. Project Overview

Internet of Things is a dynamic global information network, supports several applications for users such as healthcare organizations, security, smart transports, traffic management, Epayment, smart farming etc. Researchers estimate that IoT will consist of 50 billion objects by 2020. Most of the organizations can be monitored and controlled by smart IoT devices and applications. This smart agriculture using IOT system is powered by Arduino, it consists of Temperature sensor, Moisture sensor, water level sensor. When the IOT based agriculture monitoring system starts, it checks the water level, humidity and moisture level.

1.2. Purpose

IoT based Smart Farming improves the entire Agriculture system by monitoring the field in real-time. With the help of sensors and interconnectivity, the Internet of Things in Agriculture has not only saved the time of the farmers but has also reduced the extravagant use of resources such as Water and Electricity.

2. LITERATURE SURVEY

2.1. Existing Problem

The farming was held by manual method in which the farmers don’t know when the rain falls and at which temperature they need to water the plants so that there we be a gradual decrease in productivity. As we all know the recent news many farmers die due to decrease in productivity. Internet of Things has a strong backbone of various enabling technologies Wireless Sensor Networks, Cloud Computing, Big Data, Embedded Systems, Security Protocols and Architectures, Protocols enabling communication, web services, Internet and Search Engines. The enabling of Wireless Sensor

Network (WSN): It consists of various sensors/nodes which are integrated together to monitor various sorts of data. Cloud Computing: Cloud Computing also known as on-demand computing is a type of Internet based computing which provides shared processing resources and data to computers and other devices on demand. They can be in different forms like IaaS, PaaS, SaaS, DaaS etc. Big Data Analytics: Big data analytics is the process of examining large datasets containing various forms of data types—i.e. Big Data – to uncover hidden patterns, unknown correlations, market trends, customer preferences and other useful business information. Communication Protocols: They form the backbone of IoT systems to enable connectivity and coupling to applications and these protocols facilitate exchange of data over the network as these protocols enable data exchange formats, data encoding and addressing. Embedded Systems: It is a sort of computer system which consists of both hardware and software to perform specific tasks. It includes microprocessor/microcontroller, RAM/ROM, networking components, I/O units and storage devices.

2.2. References

1. https//:www.researchgate.et

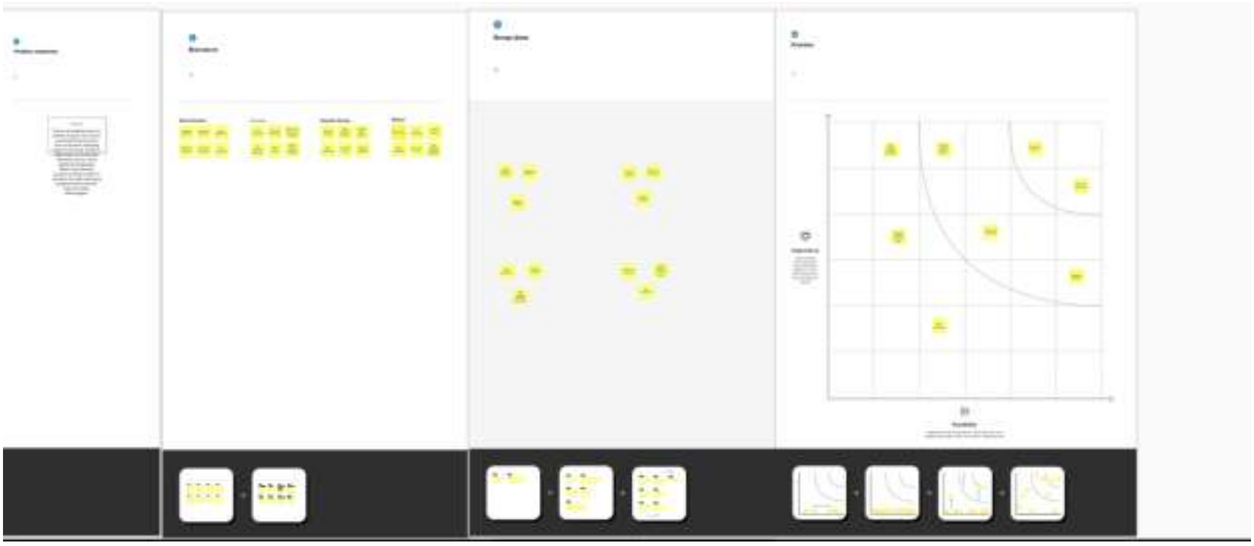
2. https//:www.wikipedia.org

2.3. Problem Statement Definition

Overuse of pesticides and fertilizer in agricultural fields leads to destruction of the crop as well as reduces the efficiency of the field increasing the soil vulnerability toward pest. IoT applications may be used to update the farmer/user about type & quantity of pesticide required by the crop. The challenges of a smart agriculture system include the integration of these sensors and tying the sensor data to the analytics driving automation and response activities.

3. IDEATION & PROPOSED SOLUTION

3.2 Ideation & Brainstorming



3.3 Proposed Solution

Project Design Phase-1
Proposed Solution

Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Farmers are under pressure to produce more food and use less energy and water in the process. A remote monitoring and control system will help farmers deal effectively with these pressures.
2.	Idea / Solution description	smart farming allows farmers to constantly monitor the field and livestock conditions by the use of IoT sensors, software, and data and enables them to take informed decisions regarding the same.
3.	Novelty / Uniqueness	IoT in agriculture uses robots, drones, remote sensors, and computer imaging combined with continuously progressing machine learning and analytical tools for monitoring crops, surveying, and mapping the fields, and providing data to farmers for rational farm management plans to save both time and money.
4.	Social Impact / Customer Satisfaction	**Increased production**: the optimisation of all the processes related to agriculture and livestock-rearing increases production rates. **Water saving:** weather forecasts and sensors that measure soil moisture mean watering only when necessary and for the right length of time.
5.	Business Model (Revenue Model)	A good business model is one that supports a viable business for customers and delivers value easily and efficiently. The IoT business model you choose or create is only restricted by your creativity and willingness to try.
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

Project Design Phase-I

PROBLEM SOLUTION FIT

	1.CUSTOMER SEGMENT	6.CUSTOMER CONSTRAINTS	5.AVALIABLE SOLUTION
	smart contracts with the integration of IoT devices in pre-harvesting and post-harvesting segments of agriculture. while IoT devices collect data from the field level, and smart contracts	purposed a proof of concept to enable lo-power, resource-constrained IoT end food retail storefronts as well as the customers can use their smart mobile phone as a portal	Internet of things (IoT) is a promising technology which provides efficient and reliable solutions towards the modernization of several domains. IoT based solutions are being developed to automatically maintain and monitor agricultural farms with minimal human involvement.

	2.Applicability of IoT in Agriculture:	9.IoT and mechanization in agriculture: problems, solutions, and prospects	7.BENEFITS
	Smart Farming is a hi-tech and effective system of doing agriculture and growing food in a sustainable way. It is an application of implementing connected devices and innovative technologies	The IoT, acronym for the Internet of Things, is a coordination of interconnected digital	IoT and ICT in agriculture, there are several benefits from the ... IoT-related threats in agriculture, which are nonetheless interwoven strongly with PA, Agriculture 4.0 and smart farming

3. TRIGGERS Whenever the temperature goes above the threshold temperature, the database will trigger an action to the decision logic which then sends a notification to the developed **4. EMOTIONS: BEFORE / AFTER** Are opposed to new ideas and they do not want to adopt IoT in the agricultural sector we get numerous benefits, but still, there are challenges faced by IoT in agricultural sectors. The biggest challenges faced by IoT in the agricultural sector are lack of information, high adoption costs, and security concerns, etc	**10. YOUR SOLUTION** The advent of technology has helped multiple sectors in attaining profitability. One such sector is agriculture. Internet of Things (IoT) implementation in this field has resulted in the term smart farming.	**8.IOT TRANSFORMING THE FUTURE OF AGRICULTURE** IoT solutions are focused on helping farmers close the supply demand gap, by ensuring high yields, profitability, and protection of the environment. The approach of using IoT technology to ensure optimum application of resources to achieve high crop yields and reduce operational costs is called precision agriculture. IoT in agriculture technologies comprise specialized equipment, wireless connectivity, software and IT services.

4. REQUIREMENT ANALYSIS

14.1 Functional requirement

Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	23 October 2022
Team ID	PNT2022TMID34687
Project Name	SmartFarmer - IoT Enabled Smart Farming Application
Maximum Marks	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form. Registration through Gmail. Registration through LinkedIN.
FR-2	User Confirmation	Confirmation via Email. Confirmation via OTP.
FR-3	Sensor Function for farming System	Measure the Temperature and Humidity Measure the Soil. Monitoring Check the crop diseases.
FR-4	Manage Modules	Manage Roles of User. Manage User permission.
FR-5	Check whether details	Temperature details. Humidity details.
FR-6	Data Management	Manage the data of weather conditions.. Manage the data of crop conditions. Manage the data of live stock conditions.

Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	➢ Farmers are very conservative in their choice of technology so that they reduce risks and tend to choose traditional techniques. ➢ User friendly guidelines for users to avail the features. ➢ Most simplistic user interface for ease of use.
NFR-2	**Security**	➢ The adoption of sensor based technologies and cloud supported smart applications in agriculture has unleashed opportunities for adversaries to orchestrate cyber attacks. ➢ All the details about the user are protected from unauthorized access. ➢ Detection and identification of any misfunctions of sensors.
NFR-3	**Reliability**	➢ Implementing Mesh IoT Networks. ➢ Building a Multi-layered defence for IoT Networks.
NFR-4	**Performance**	The use of modern technology solutions helps to achieve the maximum performances thus resulting in better quality and quantity yields.
NFR-5	**Availability**	This app is available for all platforms.
NFR-6	**Scalability**	Scalability refers to the ability to increase available resources and system capability without the need to go through a major system redesign or implementation.

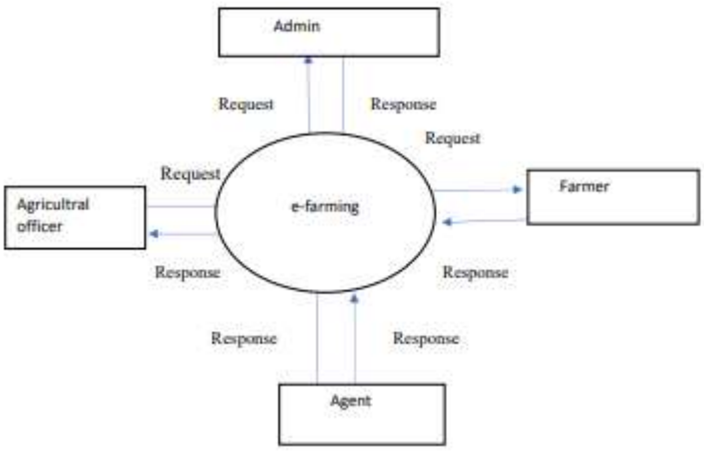
5 PROJECT DESIGN

5.1 Data Flow Diagram

Data Flow Diagrams:

In this study, we research issues emerging in developing a cost-effective smart farming system. By proposing a system architecture and relevant solutions, we successfully integrate different modules related to sensing systems, communication, and data analytics into a whole system that not only monitors the farm environment but also performs remote...

Example:



User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	Evaluating methods	Evaluating concept
Customer	Sensors: soil,water,light,humidity temperture	Session 1: Vihti 28.6.2012	As indicated above the Smart spraying concept and pilot was elaborated during the last design phase (June-October 2012) MTT Smart Spraying System developers and VTT human factors
		Session 2: Jyväskylä 25.-26.10.2012	The developed concepts and user interfaces were demonstrated to the participants of a national KoneAgria

User Type	Functional Requirement (Epic)	Evaluating methods	Evaluating concept
			fair (Agrimachine) in Jyväskylä at the 25.-26.10.2012.
		National discussion panel 24.10.2012	As an activity of the WP700 VTT and MTT organised a national discussion panel at VTT Otaniemi site, on the 24.10.2012. In this session four pilots Fruit and Vegetables, Tracking, Tracing and Awareness Meat (TTAM)
		Results	he original overview model of the smart spraying concept was improved according to the elaboration of the pilot MTT and VTT decided to accomplish.

5.2Technical Architecture

Project Design Phase-I
Solution Architecture

Date	18 October 2022
Team ID	PNT2022TMID34687
Project Name	Project – Smart Farmer - Io Enabled Smart Farming Application
Maximum Marks	4 Marks

Solution Architecture:

The working of smart devices in farming allows farmers to apply amount of resources like water, fertilizers, etc.... at the right time and right place in right time. This type of farming is a priceless tool for composing chemicals in soil. The devices in this field will be user friendly and helps the farmers to plan irrigation and more activities based on environmental and soil conditions. The parameters like temperature, humidity, pH value and soil moisture can be monitored by these device controllers and can be informed through some graphical representations.

Solution Architecture Diagram:

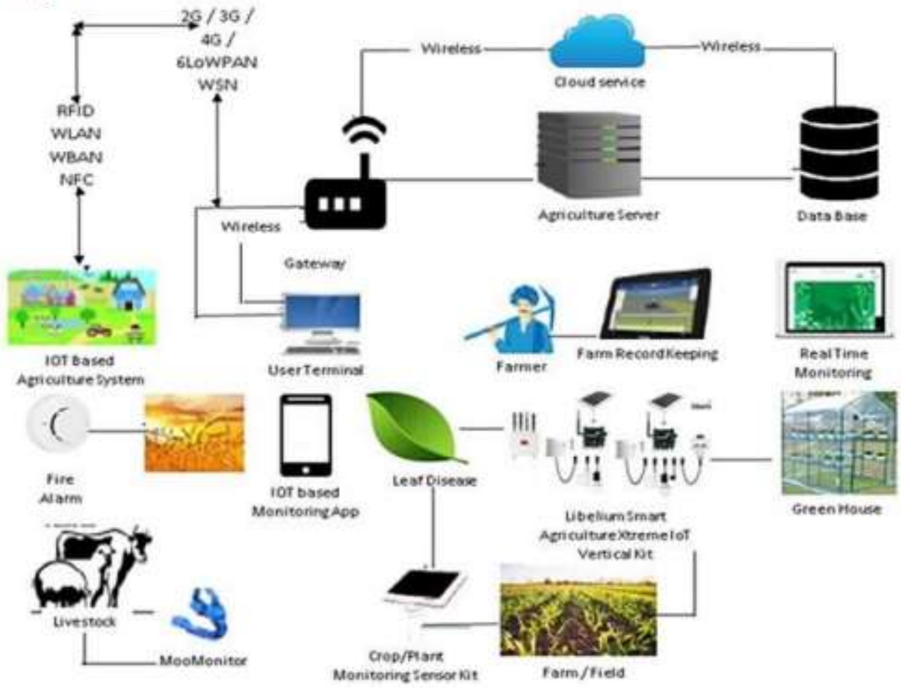


Figure 1: Architecture and data flow of Smart Farmer - Io Enabled Smart Farming Application

6 PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Project Planning Phase
Sprint Delivery Plan

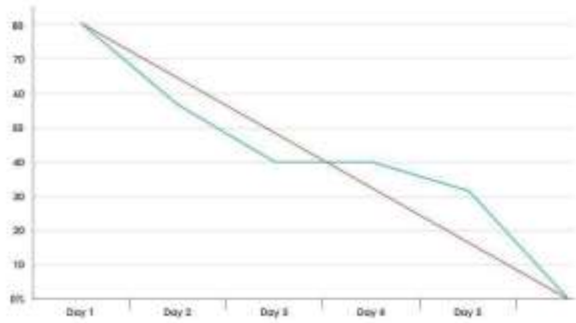
Date	28 October 2022
Project Name	Smart farmer- IoT based smart farming application
Maximum Marks	8 Marks
TEAM ID	PNT2022TMID34687

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Sprint	Functional Requirement (Epic)	User Story Number	User Story /Task	Story Points	Priority	Team Member
Sprint-1	Registration (Farmer Mobile User)	UNS-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Sweetlin Derisha S(Leader),Roshini A R, Sherin Davisha D, Sowmiya A
Sprint-1	Login	UNS-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Sweetlin Derisha S(Leader), Roshini A R, Sherin Davisha D, Sowmiya A
Sprint-1	Data Visualization	UNS-4	As a user, I can register for the application through GMAIL	2	Medium	Sweetlin Derisha S(Leader), Roshini A R, Sherin Davisha D, Sowmiya A
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.	2	High	Sweetlin Derisha S(Leader), Roshini A R, Sherin Davisha D, Sowmiya A
Sprint - 1	Registration (Chemical Manufacturer - Mobile User)	USN - 1	As a user, I want to first register using my email and create a password for the account.	3	High	Sweetlin Derisha S(Leader), Roshini A R, Sherin Davisha D, Sowmiya A

6.2 velocity

Burndown Chart:



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

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	11 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	17 Nov 2022

7.CODING & SOLUTIONING

(Explain the features added in the project along with code)

7.1 Feature 1

- We Added Weather Map Parameter like (Temperature, Pressure, Humidity) of Farmer’s Location, that is Displayed in Mobile Application & WEB UI
- **Python Code Show Below**

```
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
import requests, json

ms=0
# Enter your API key here
api_key = "[illegible]"
# base_url variable to store url
base_url = "http://api.openweathermap.org/data/2.5/weather?"
# Give city name
city_name = 'Chennai, IN'
# complete_url variable to store
# complete url address
complete_url = base_url + "appid=" + api_key + "&q=" + city_name

status='motor off'
myConfig = {
    "identity": {
        "orgId": "[illegible]",
        "typeId": "MyDeviceType",
        "deviceId":"12345"
    },
    "auth": {
        "token": "[illegible]"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
    if(m=="MOTOR ON"):#if motor is on
        print("MOTOR  IS ON")
        global status
        status="motor on"
        myData={'temperature':temp, 'humidity':hum,'soilmoisture':sm_percentage,'status':status,'api_temperature':api_temperature,'api_pressure':api_pressure
        client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
        print("Published data Successfully: %s", myData)
```

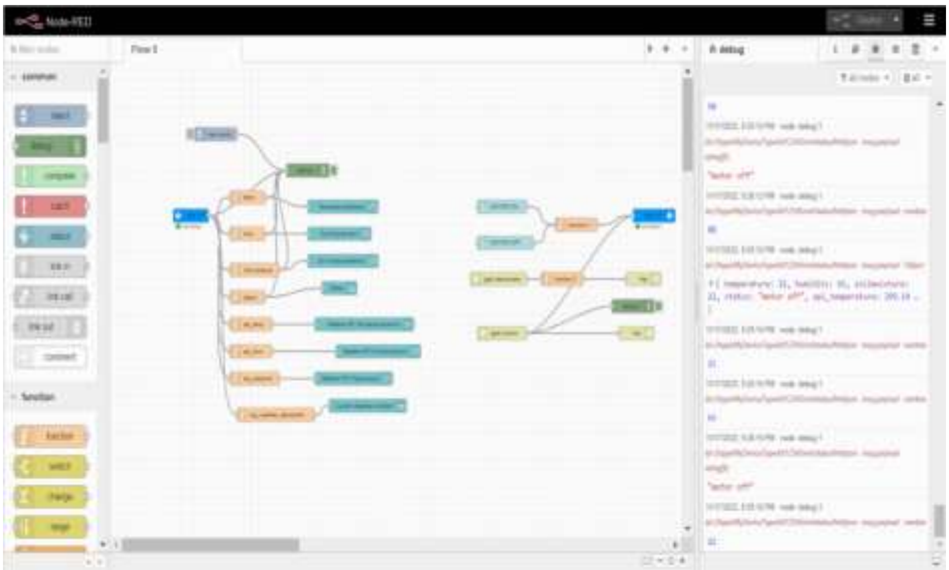
```
        time.sleep(2)


client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
    # get method of requests module
    # return response object
    response = requests.get(complete_url)
    # json method of response object
    # convert json format data into
    # python format data
    x = response.json()
    # Now x contains list of nested dictionaries
    # Check the value of "cod" key is equal to
    # "404", means city is found otherwise,
    # city is not found
    if x["cod"] != "404":



        y = x["main"]


        api_temperature = y["temp"]#getting api temperature data


        api_pressure = y["pressure"]#getting api pressure data


        api_humidity = y["humidity"] #getting api humidity data


        z = x["weather"]


        api_weather_description = z[0]["description"]#getting api weather condition data
```

```
        api_weather_description = z[0]["description"]#getting api weather condition data

    temp=random.randint(-20,125)#genrating ranom values for temperature
    hum=random.randint(0,100)#genrating ranom values for humidity
    soilmoisture=random.randint(0,1023)#analog sensor
    sm_percentage=(soilmoisture/1023)*100
    sm_percentage=int(sm_percentage)#genrating ranom values for soilmoisture
    myData={'temperature':temp, 'humidity':hum,'soilmoisture':sm_percentage,'status':status,'api_temperature':api_temperature,'api_pressure':api_pressure,'ap
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)

time.sleep(2)
client.disconnect()
```

8.TESTING

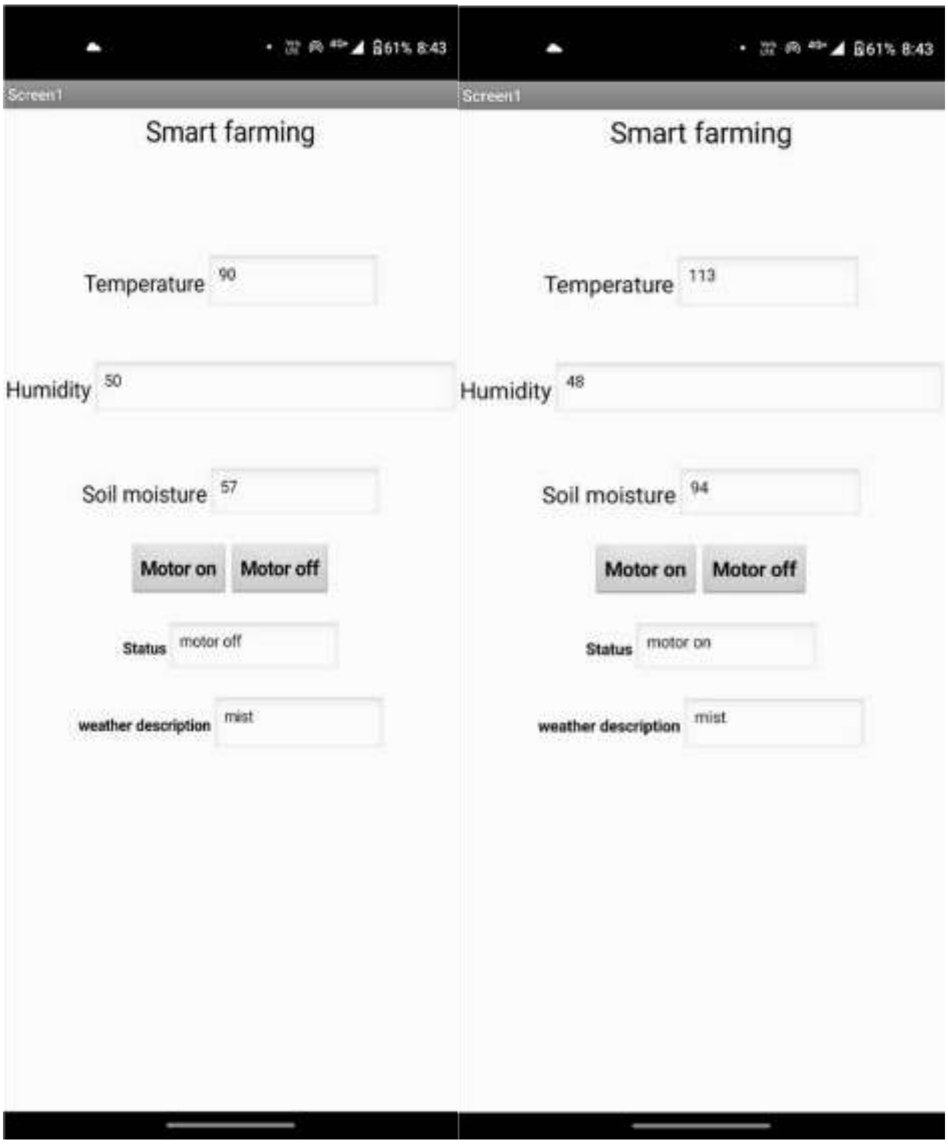
8.1 Testing Output of Python Code

Node Red Connected and Publishes the Values



8.2 User Acceptance Testing

The Output Live Data is Show In Mobile Application


61% 8:43
Screen1
Smart farming
Temperature
90
Humidity
50
Soil moisture
57
Motor on
Motor off
Status
motor off
weather description
mist
61% 8:43
Screen1
Smart farming
Temperature
113
Humidity
48
Soil moisture
94
Motor on
Motor off
Status
motor on
weather description
mist

9.RESULTS

9.1Performance Metrics

10. ADVANTAGES AND DISADVANTAGES

Advantages:

● Crop Condition Details

● Weather Forecast

● Remote Monitoring

● Easy To Use UI

● Data Collection

● Analysis of Data ●
Remote Motor Control

Disadvantages:

● Privacy Issue

● Internet Connectivity

11. CONCLUSION:

Smart Agriculture System Based On Internet Of Things candeliver the farmer all the required information like temperature,humidity, soil mositure of the crop in realtime and also theweather forecast at fingertips. Also instead of using manualbased Motor control, the farmer can do this remotely anywhere aslong as he's connected to network.To make this possible we have used IBM CloudPlatform,Watson Iot Platform, Openweather API and Node-red to gatherand show the information on Web Application. By using a PythonScript we were able to subscribe to IBM platform to send andreceive commands to motor for controlling it.Using this Smart Agriculture System the farmer can not onlymonitor all the required data in realtime but also can make smartdecisions for better yield based on the data collected.In this wayhe can produce yieldeffectively and also earn profitably morebased on accurate data received.

12. FUTURE SCOPE:

Future scope of this smart agriculture system will be to addmore sensors to the existing micro controller, to add increase thcurrent functionality or to do more automated tasks likeautomatic watering system, adding pest control information andgeotagging the farm etc. This information can be shared onconsent to Government authorities or Private companies for more suggestions of better techniques remotely. As the data stored canbe used for reefernce and analysis which can be very helpful infuture.

13.APPENDIX

Source Code

```
#IBM Watson IOT Platform

#pip install wiotp-sdk import

wiotp.sdk.device import time

import random import

requests, json



ms=0

# Enter your API key here

api_key = "a0db30a689a774b93ffcb58ef2eddfda"

# base_url variable to store url

base_url = "http://api.openweathermap.org/data/2.5/weather?"

# Give city name city_name

= 'Chennai, IN'

# complete_url variable to store #

complete url address

complete_url = base_url + "appid=" + api_key + "&q=" + city_name
```

```
status='motor off' myConfig
= {

"identity": {
"orgId": "17lsro",
"typeId": "MyDeviceType",
"deviceId":"12345"
},
"auth": {
"token": "GkatKdiUS?UVHKvnAD"
}
}


def myCommandCallback(cmd):
print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
m=cmd.data['command']          if(m=="MOTOR   ON"):#if   motor   is   on
print("MOTOR  IS ON")     global status     status='motor on'

myData={'temperature':temp,
'humidity':hum,'soilmoisture':sm_percentage,'status':status,'api_temperature':api_temperature,'api_pres
sure':api_pressure,'api_humidity':api_humidity,'api_weather_description':api_weather_description}
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
print("Published data Successfully: %s", myData)


time.sleep(2)


elif(m=="MOTOR     OFF"):#if     motor     is     off
print("MOTOR IS OFF")


status='motor off'
myData={'temperature':temp,
'humidity':hum,'soilmoisture':sm_percentage,'status':status,'api_temperature':api_temperature,'api_pres
sure':api_pressure,'api_humidity':api_humidity,'api_weather_description':api_weather_description}
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
print("Published data Successfully: %s", myData)
```

```
time.sleep(2)

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None) client.connect()

while True:
# get method of requests module     #
return response object
response = requests.get(complete_url)
# json method of response object
# convert json format data into
# python format data     x
= response.json()
# Now x contains list of nested dictionaries
# Check the value of "cod" key is equal to
# "404", means city is found otherwise,
# city is not found     if
x["cod"] != "404":

y = x["main"]

api_temperature = y["temp"]#getting api temperature data

api_pressure = y["pressure"]#getting api pressure data
```

```
api_humidity = y["humidity"] #getting api humidity data

z = x["weather"]

api_weather_description = z[0]["description"]#getting api weather condition data

temp=random.randint(-20,125)#geneating ranom values for temperature
hum=random.randint(0,100)#geneating ranom values for humidity
soilmoisture=random.randint(0,1023)#analog sensor
sm_percentage=(soilmoisture/1023)*100
sm_percentage=int(sm_percentage)#geneating ranom values for soilmoisture
myData={'temperature':temp,
'humidity':hum,'soilmoisture':sm_percentage,'status':status,'api_temperature':api_temperature,'api_pressure':api_pressure,'api_humidity':api_humidity,'api_weather_description':api_weather_description}
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
print("Published data Successfully: %s", myData) client.commandCallback = myCommandCallback
time.sleep(2)

time.sleep(2) client.disconnect()
```