Project Development Phase

SPRINT DELIVERY – 4

Team ID	PNT2022TMID38090
Project Name	Smart Farmer - IoT Enabled Smart Farming Application.

Receiving commands from IBM cloud using Python program

```python
import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

#IBM

organization = "janesh"

deviceType = "raspberrypi"

deviceId = "12345"

authMethod = "use-token-auth"

authToken = "12345678"

#Gpio

def mycommandCallback(cmd):

print("Command Received: %s" %cmd.data['command'])

status = cmd.data['command']

if status=="lighton":

print("LED is ON")

elif status=="lightoff":

print("LED is OFF")

else:
```

```python
print("please send proper command")

try:

deviceOptions =

{"org":organization,"type":deviceType,"id":deviceId,"auth-method":authMethod,

"auth-token":authToken}

deviceCli = ibmiotf.device.Client(deviceOptions)

except Exception as e:

print("Caught exception connecting device: %s" %str(e))

sys.exit()

#CONNECT

deviceCli.connect()

while True:

temp=random.randint(0,100)

hum=random.randint(0,100)data={'temp':temp,'hum':hum}

def myOnPublishCallback():

print("Published Temperature = %s C"%temp,"Humidity = %s %%" %hum, "to

IBM Watson")

success = deviceCli.publishEvent("IoTSensor","json",data,qos=0,

on_publish=myOnPublishCallback)

if not success:

print("Not connected to IoTF")

time.sleep(10)
```
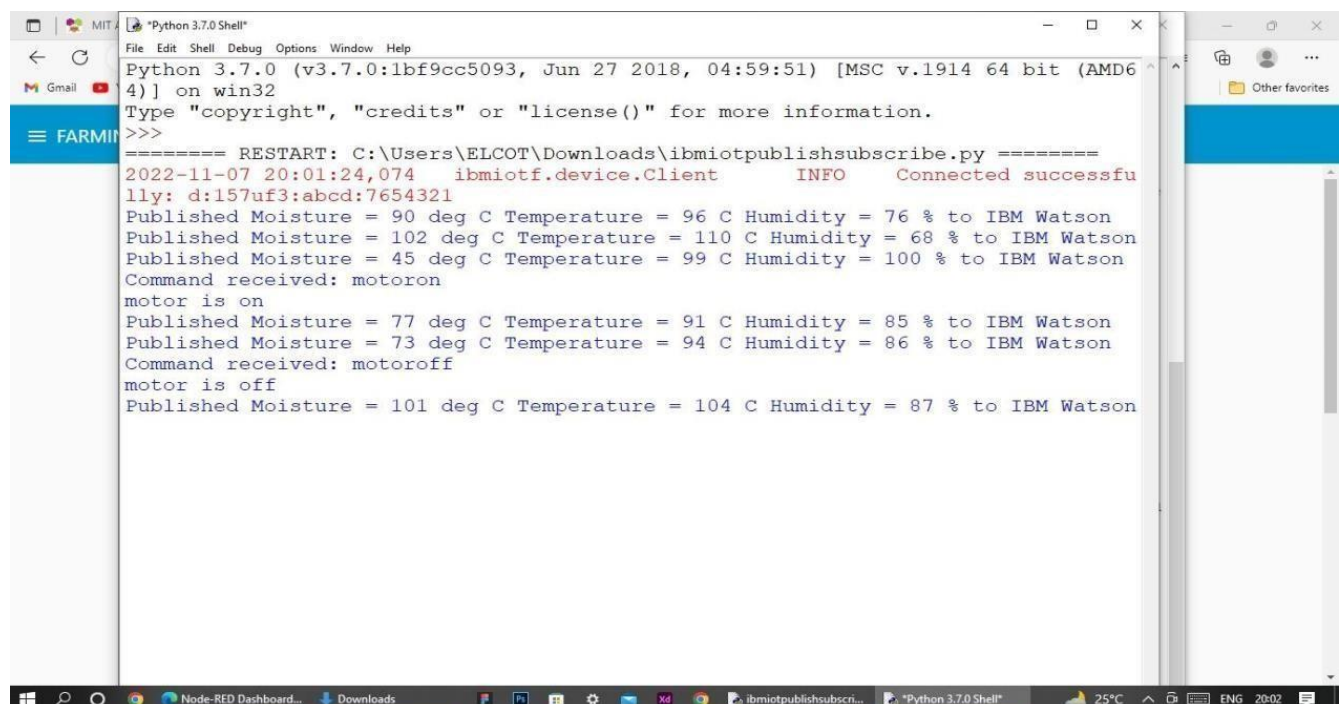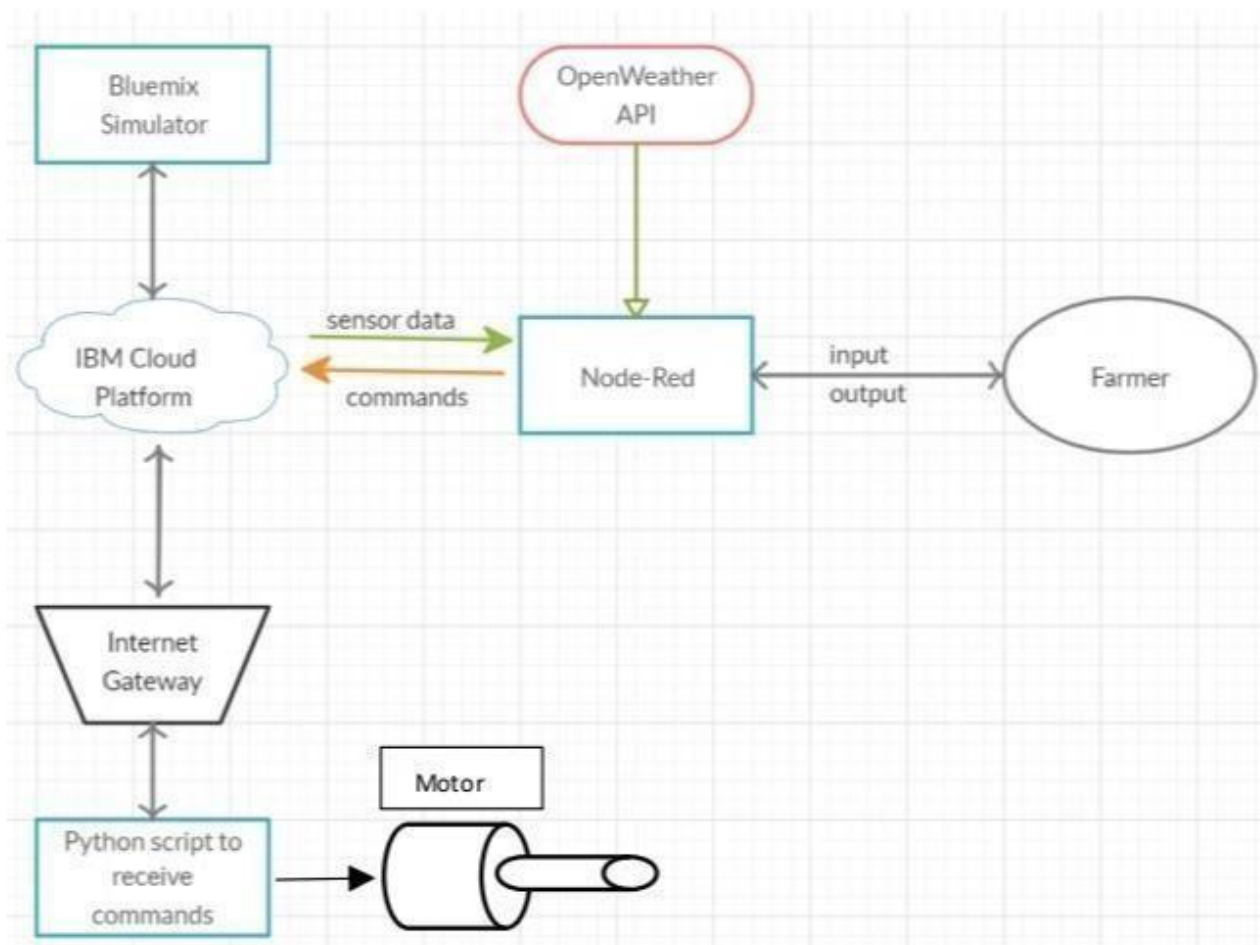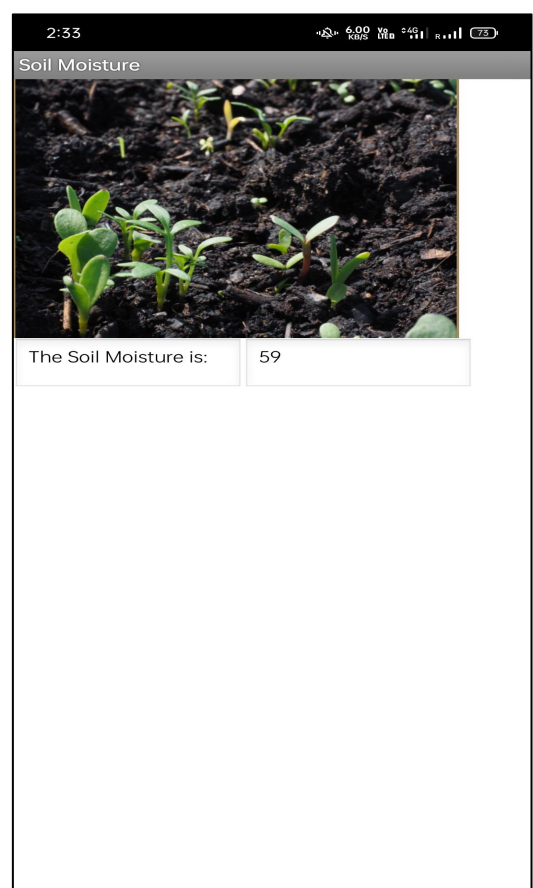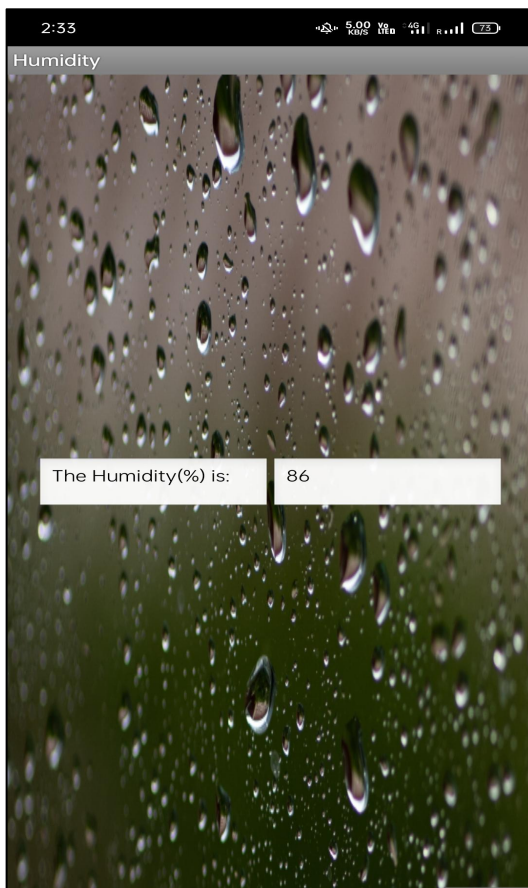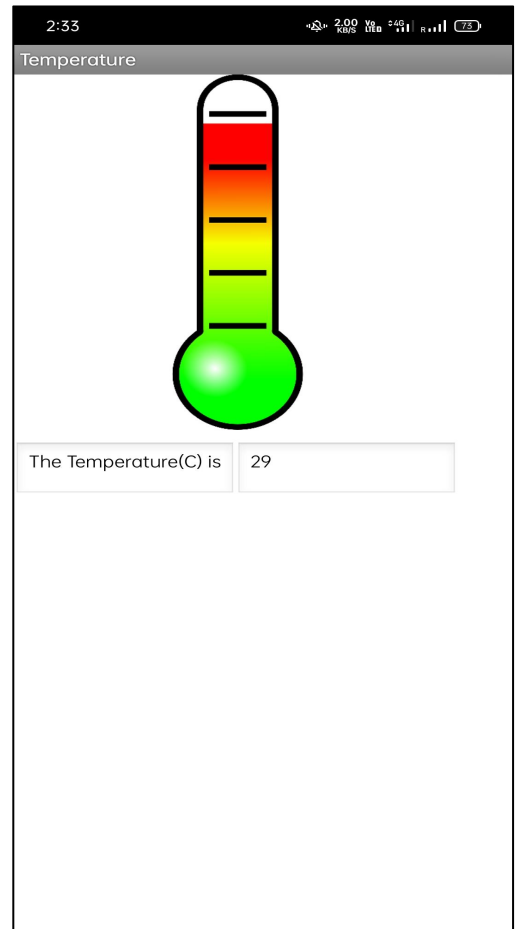
deviceCli.commandCallback = mycommandCallback

#DISCONNECT

deviceCli.disconnect()

SCREENSHOTS:


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


#Provide your IBM Watson Device Credentials
organization = "157uf3"
deviceType = "abcd"
deviceId = "7654321"
authMethod = "token"
authToken = "87654321"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print ("motor is on")
    elif status == "motoroff":
        print ("motor is off")
    else :
        print ("please send proper command")

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



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
======== RESTART: C:\Users\ELCOT\Downloads\ibmiotpublishsubscribe.py ========
2022-11-07 20:01:24,074   ibmiotf.device.Client      INFO     Connected successfu
lly: d:157uf3:abcd:7654321
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 = 87 % to IBM Watson
```

FLOW CHART:

MOBILE APPLICATION:

Temperature

Humidity

Soil_Moisture

Motor

The Temperature(C) is | 29

The Humidity(%) is: | 86

The Soil Moisture is: | 59
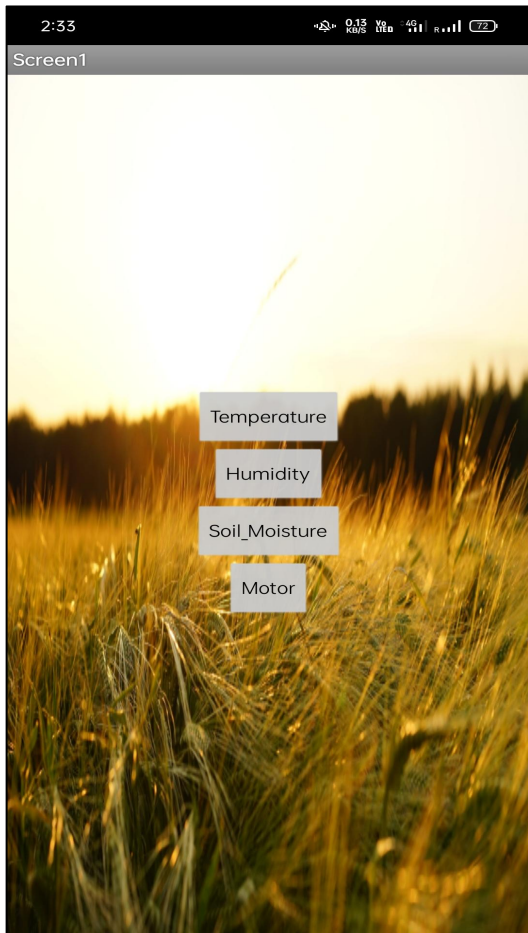
Motor ON/OFF

OBSERVATION & RESULTS:
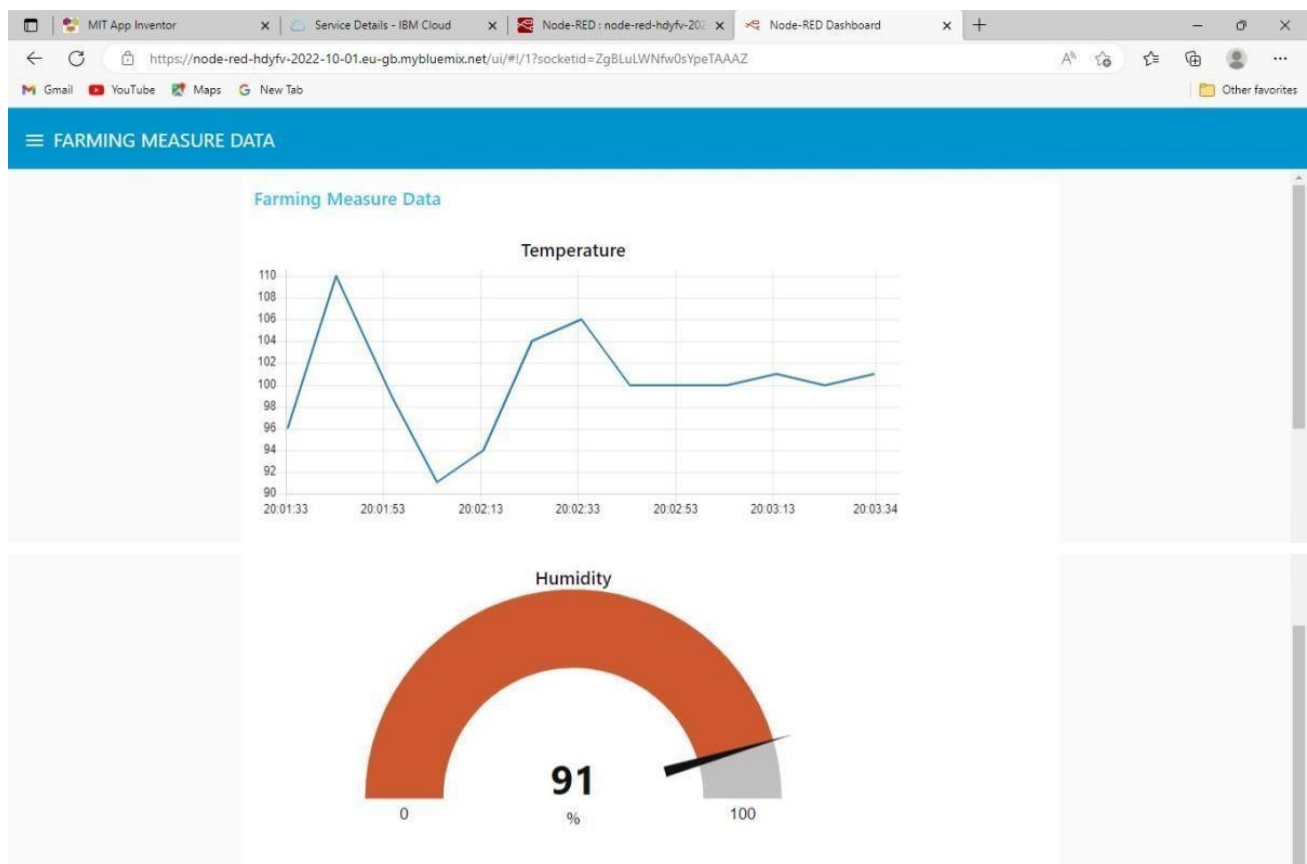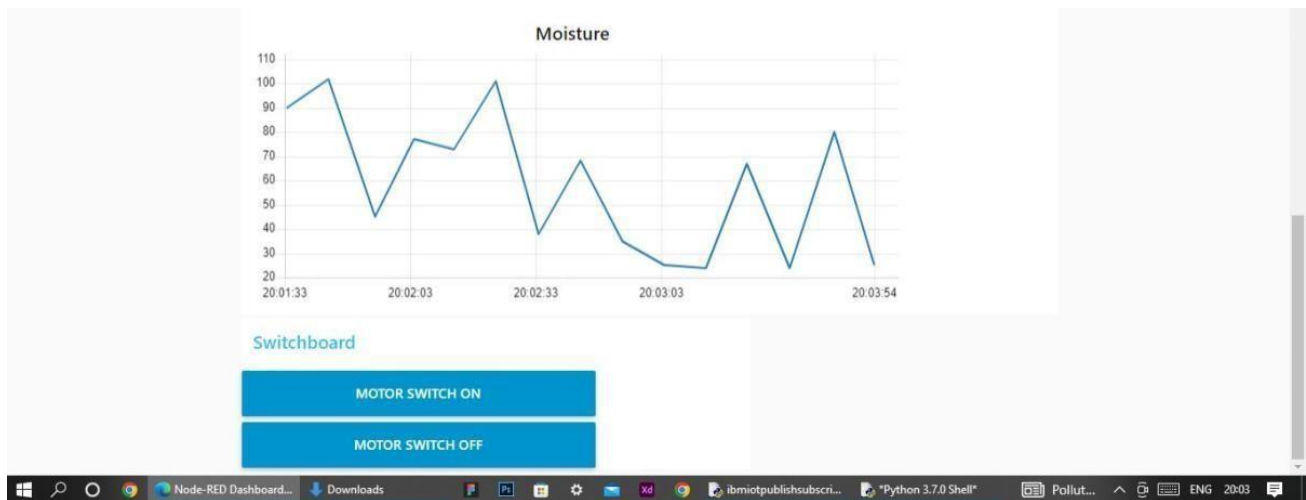
Advantages:

• Farms can be monitored and controlled remotely.

• Increase in convenience to farmers.

• Less labor cost.

• Better standards of living.

Disadvantages:

• Lack of internet/connectivity issues.

• Added cost of internet and internet gateway infrastructure.

• Farmers wanted to adapt the use of Mobile App.

CONCLUSION:

Thus the objective of the project to implement an IoT system in order to help farmers to control and monitor their farms has been implemented successfully.