Sprint Delivery - 4

SmartFarmer - IoT Enabled Smart Farming Application

Team ID: PNT2022TMID30663 **Date:17/11/2022**

Receiving commands from IBM cloud using Python program

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": authMethod, "auth-token": authToken}

deviceCli = ibmiotf.device.Client(deviceOptions)

        #.....

except Exception as e:
```

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

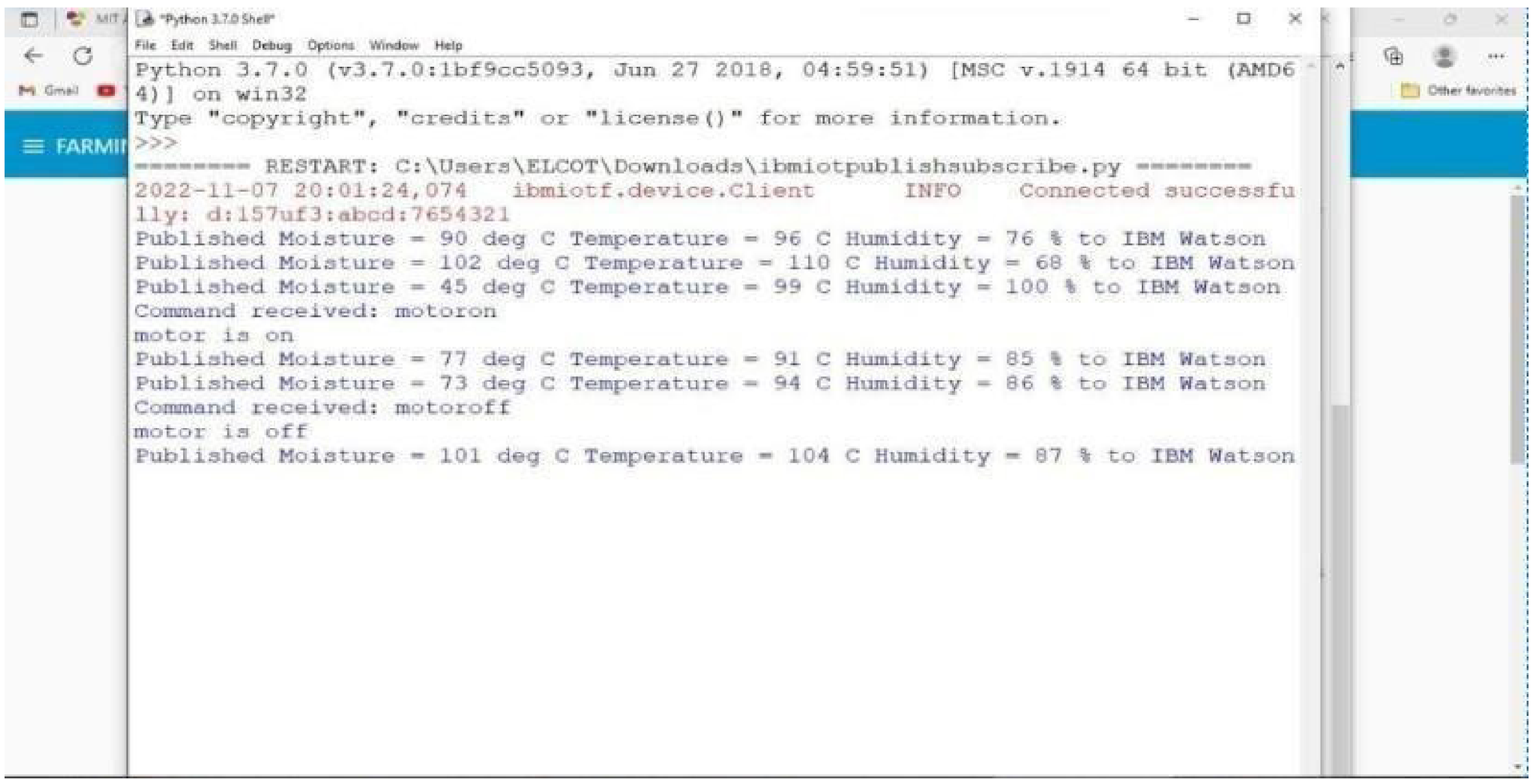
```
 sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an
eventof type "greeting" 10 times deviceCli.connect()


while True:

        #Get Sensor Data from DHT11

temp=random.randint(90,110)

Humid=random.randint(60,100)

Mois=random. Randint(20,120)  data

=

{ 'temp' : temp, 'Humid': Humid , '   Mois'   :

Mois}

        #print data

                              de

fmyOnPublishCallback():

            print ("Published Temperature = %s C" % temp, "Humidity = %s
%%" % Humid, " Moisture =%s deg c"  % Mois " 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 the device and application from the cloud

deviceCli.disconnect()
```

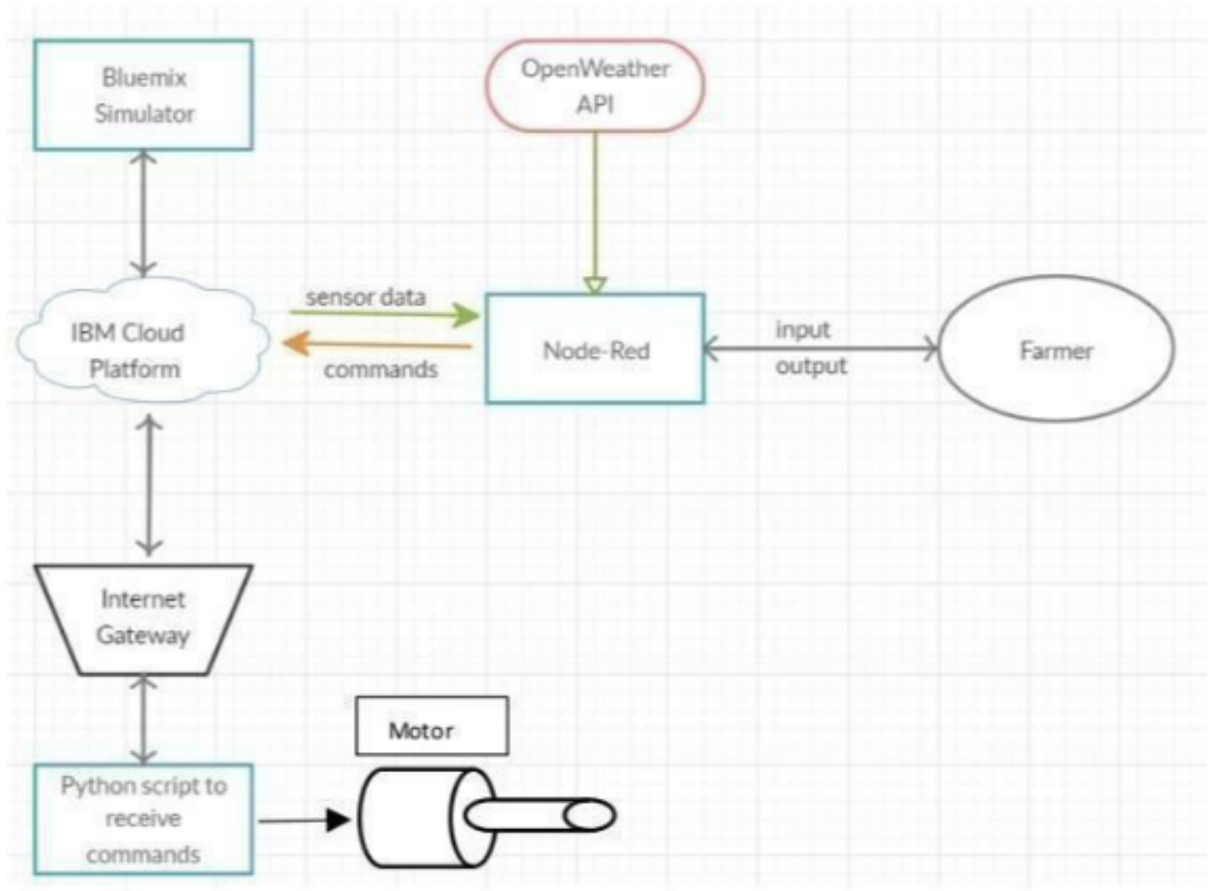
ibmiotpublishsubscribe.py - C:\Users\ELCOT\Downloads\ibmiotpublishsubscribe.py (3.7.0)

File Edit Format Run Options Window Help

```
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)
    #.....................................................
```

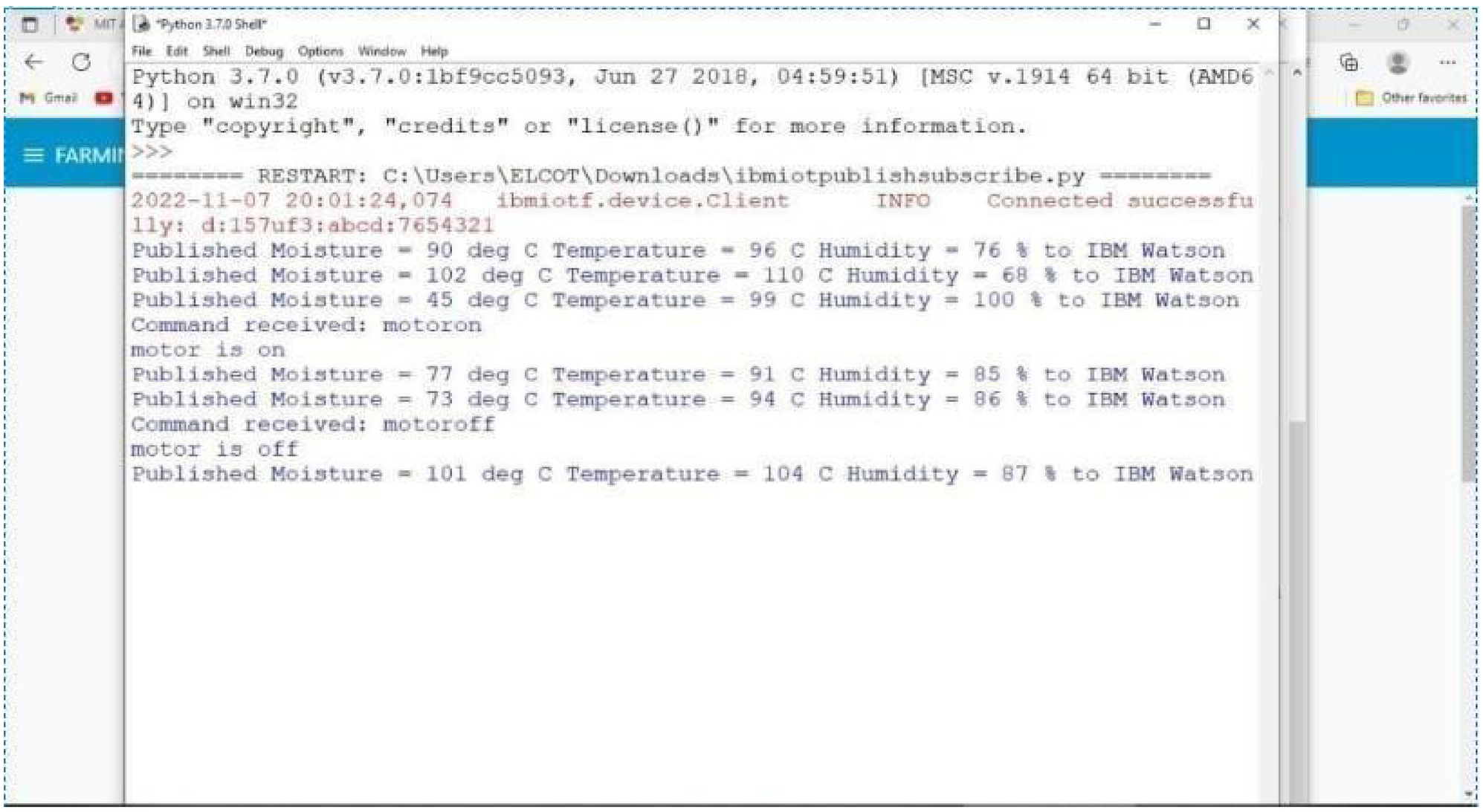
Ln: 22 Col: 21

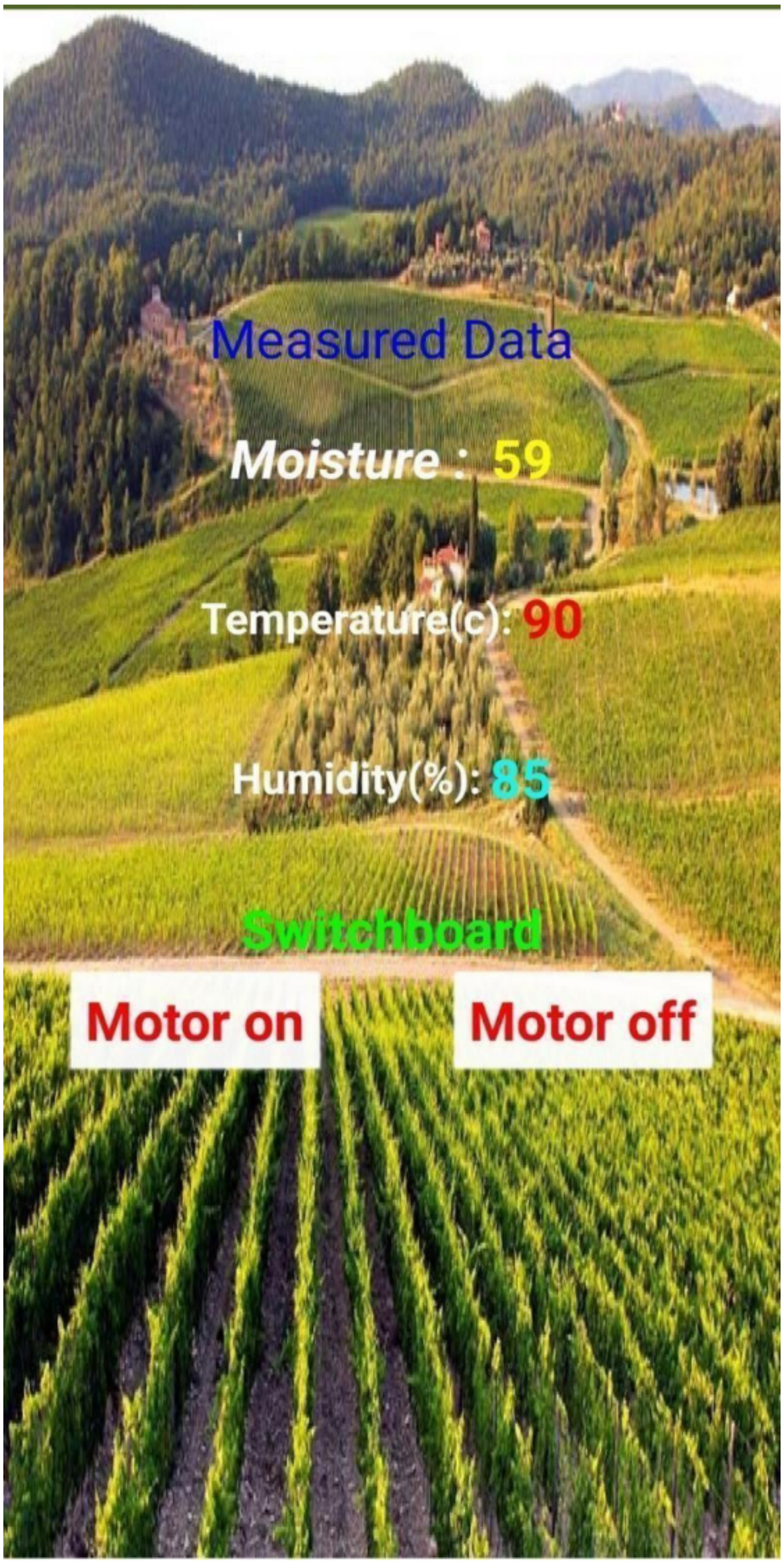


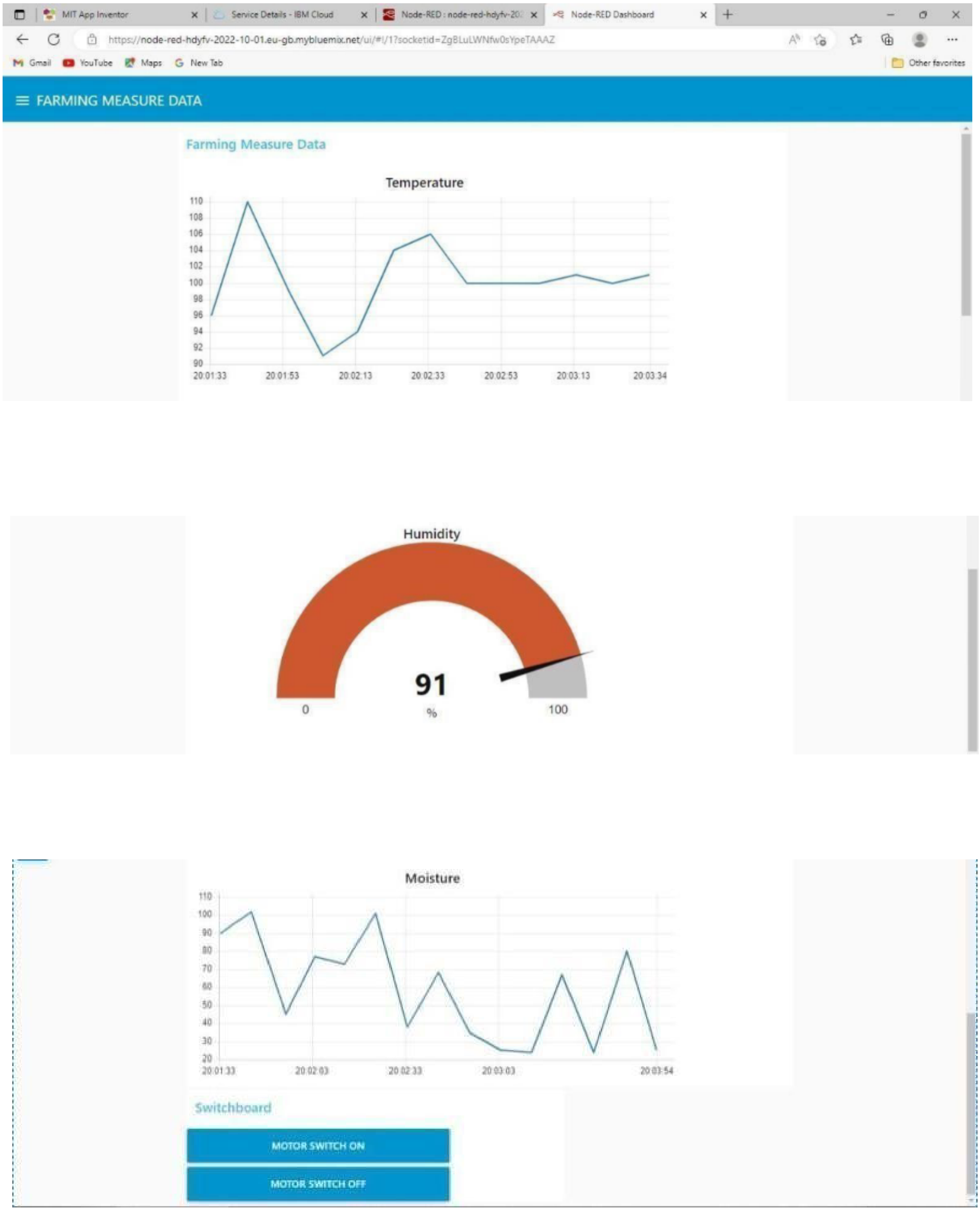


6. Flow Chart

7. Observations & Results




Measured Data
Moisture : 59
Temperature(c): 90
Humidity(%): 85
Switchboard
Motor on
Motor off



8. Advantages & Disadvantages 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.

9. 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.