

# Smart Farmer-IOT Enabled Smart Farming Application

## SPRINT DELIVERY – 4

<b>TITLE</b>	<b>Smart Farmer-IOT Enabled Smart Farming Application</b>
<b>DOMAIN NAME</b>	INTERNET OF THINGS
<b>TEAM ID</b>	PNT2022TMID22828
<b>LEADER NAME</b>	KOWSALYA D
<b>TEAM MEMBER NAME</b>	KAMALAKANNAN R KARTHICK S NITHEEN V P

## 5.5 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 event  
of 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    def
```

```
myOnPublishCallback():
```

```
    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 IoT")    time.sleep(10)
```

```
    deviceCli.commandCallback = myCommandCallback #
```

```
Disconnect the device and application from the cloud
```

```
deviceCli.disconnect()
```

```
ibm code.py - C:\Users\rkama\AppData\Local\Programs\Python\Python39\ibm code.py (3.9.6)
File Edit Format Run Options Window Help

import wiotp.sdk.device
import time
import os
import datetime
import random

myConfig={
    "identity":{
        "orgId":"tdo49a",
        "typeId":"NodeMCU",
        "deviceId":"12345"
    },
    "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=="motoroft"):
        print("Motor is switched OFF")
    print(" ")

while True:
    h=random.randint(0,100)
    temp=random.randint(-20,125)
    p=random.randint(0,100)
    myData={"humidity":h, 'temperature':temp,
    'ph':p}
    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()
```

```
ibm code.py - C:\Users\rkama\AppData\Local\Programs\Python\Python39\ibm code.py (3.9.6)
File Edit Format Run Options Window Help

import wiotp.sdk.device
import time
import os
import datetime
import random

myConfig={
    "identity":{
        "orgId":"tdo49a",
        "typeId":"NodeMCU",
        "deviceId":"12345"
    },
    "auth":{
        "token":"12345678"
    }
}

client=wiotp.sdk.device.DeviceClient
client.connect()

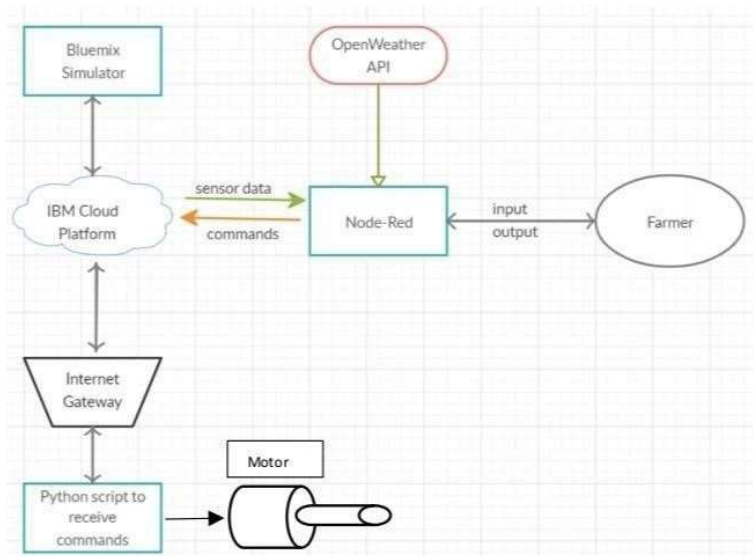
def myCommandCallback(cmd):
    print("Message received from IBM
m=cmd.data['command']
if(m=="motoron"):
    print("Motor is switched on")
elif(m=="motoroft"):
    print("Motor is switched OFF")
    print(" ")
while True:
    soil=random.randint(0,100)
    temp=random.randint(-20,125)
    hum=random.randint(0,100)
    myData={"soil_moisture":soil, 'tem
"humidity":hum}
    client.publishEvent(eventId="statu
msgFormat="json", data=myData,
qos=0,onPublish=None)
    print("Published data Successfully
time.sleep(2)
    client.commandCallback=myCommandC
    client.disconnect()
```

```
"IDE Shell 3.9.6"
File Edit Shell Debug Options Window Help

2022-11-16 22:03:29,455 wiotp.sdk.device.client.DeviceClient INFO Connecte
d successfully: d:tdo49a:NodeMCU:12345
Published data Successfully: %s {'soil_moisture': 19, 'temperature': 72, 'humidi
ty': 67}
Published data Successfully: %s {'soil_moisture': 51, 'temperature': 48, 'humidi
ty': 24}
Published data Successfully: %s {'soil_moisture': 85, 'temperature': -1, 'humidi
ty': 53}
Published data Successfully: %s {'soil_moisture': 79, 'temperature': 37, 'humidi
ty': 71}
Published data Successfully: %s {'soil_moisture': 37, 'temperature': 53, 'humidi
ty': 66}
Published data Successfully: %s {'soil_moisture': 1, 'temperature': 75, 'humidit
y': 34}
Published data Successfully: %s {'soil_moisture': 32, 'temperature': 89, 'humidi
ty': 73}
Published data Successfully: %s {'soil_moisture': 23, 'temperature': 61, 'humidi
ty': 99}
Published data Successfully: %s {'soil_moisture': 88, 'temperature': 30, 'humidi
ty': 89}
Published data Successfully: %s {'soil_moisture': 12, 'temperature': 55, 'humidi
ty': 53}
Published data Successfully: %s {'soil_moisture': 56, 'temperature': 2, 'humidit
y': 63}
Published data Successfully: %s {'soil_moisture': 65, 'temperature': -8, 'humidi
ty': 42}
Published data Successfully: %s {'soil_moisture': 47, 'temperature': 44, 'humidi
ty': 97}
Published data Successfully: %s {'soil_moisture': 59, 'temperature': 90, 'humidi
ty': 36}
Published data Successfully: %s {'soil_moisture': 68, 'temperature': 95, 'humidi
ty': 76}

-- RESTART: C:\Users\rkama\AppData\Local\Programs\Python\Python39\ibm code.py --
2022-11-16 22:04:02,843 wiotp.sdk.device.client.DeviceClient INFO Connecte
d successfully: d:tdo49a:NodeMCU:12345
Published data Successfully: %s {'soil_moisture': 81, 'temperature': 87, 'humidi
ty': 50}
```

## 6.Flow Chart



## 7.Observations & Results

```

ibm code.py - C:/Users/rkama/AppData/Local/Programs/Python/Python39/ibm code.py (3.9.6)
File Edit Format Run Options Window Help

import wiottp.sdk.device
import time
import os
import datetime
import random
myConfig={
  "identity":{
    "orgId":"td049a",
    "typeId":"NodeMCU",
    "deviceId":"12345"
  },
  "auth":{
    "token":"12345678"
  }
}
client=wiottp.sdk.device.DeviceClient()
client.connect()
def myCommandCallback(cmd):
  print("Message received from IBM")
  m=cmd.data['command']
  if(m=="motoron"):
    print("Motor is switched on")
  elif(m=="motroff"):
    print("Motor is switched OFF")
  print(" ")
  while True:
    soil=random.randint(0,100)
    temp=random.randint(-20,125)
    hum=random.randint(0,100)
    myData={'soil_moisture':soil, 'temperature':temp, 'humidity':hum}
    client.publishEvent(eventId="status", data=myData, msgFormat="json", qos=0, onPublish=None)
    print("Published data Successfully")
    time.sleep(2)
    client.commandCallback=myCommandCallback
    client.disconnect()

2022-11-16 22:03:29,455 wiottp.sdk.device.client.DeviceClient INFO Connect
d successfully: d:td049a:NodeMCU:12345
Published data Successfully: %s {'soil_moisture': 19, 'temperature': 72, 'humidi
ty': 67}
Published data Successfully: %s {'soil_moisture': 51, 'temperature': 48, 'humidi
ty': 24}
Published data Successfully: %s {'soil_moisture': 85, 'temperature': -1, 'humidi
ty': 53}
Published data Successfully: %s {'soil_moisture': 79, 'temperature': 37, 'humidi
ty': 71}
Published data Successfully: %s {'soil_moisture': 37, 'temperature': 53, 'humidi
ty': 66}
Published data Successfully: %s {'soil_moisture': 1, 'temperature': 75, 'humidit
y': 34}
Published data Successfully: %s {'soil_moisture': 32, 'temperature': 89, 'humidi
ty': 73}
Published data Successfully: %s {'soil_moisture': 23, 'temperature': 61, 'humidi
ty': 99}
Published data Successfully: %s {'soil_moisture': 88, 'temperature': 30, 'humidi
ty': 89}
Published data Successfully: %s {'soil_moisture': 12, 'temperature': 55, 'humidi
ty': 53}
Published data Successfully: %s {'soil_moisture': 56, 'temperature': 2, 'humidit
y': 63}
Published data Successfully: %s {'soil_moisture': 65, 'temperature': -8, 'humidi
ty': 42}
Published data Successfully: %s {'soil_moisture': 47, 'temperature': 44, 'humidi
ty': 97}
Published data Successfully: %s {'soil_moisture': 59, 'temperature': 90, 'humidi
ty': 36}
Published data Successfully: %s {'soil_moisture': 68, 'temperature': 95, 'humidi
ty': 76}

== RESTART: C:/Users/rkama/AppData/Local/Programs/Python/Python39/ibm code.py ==
2022-11-16 22:04:02,843 wiottp.sdk.device.client.DeviceClient INFO Connect
d successfully: d:td049a:NodeMCU:12345
Published data Successfully: %s {'soil_moisture': 81, 'temperature': 87, 'humidi
ty': 50}
Ln:14 Col:0

```

## Measured Data

Moisture : 59

Temperature(c): 90

Humidity(%): 85

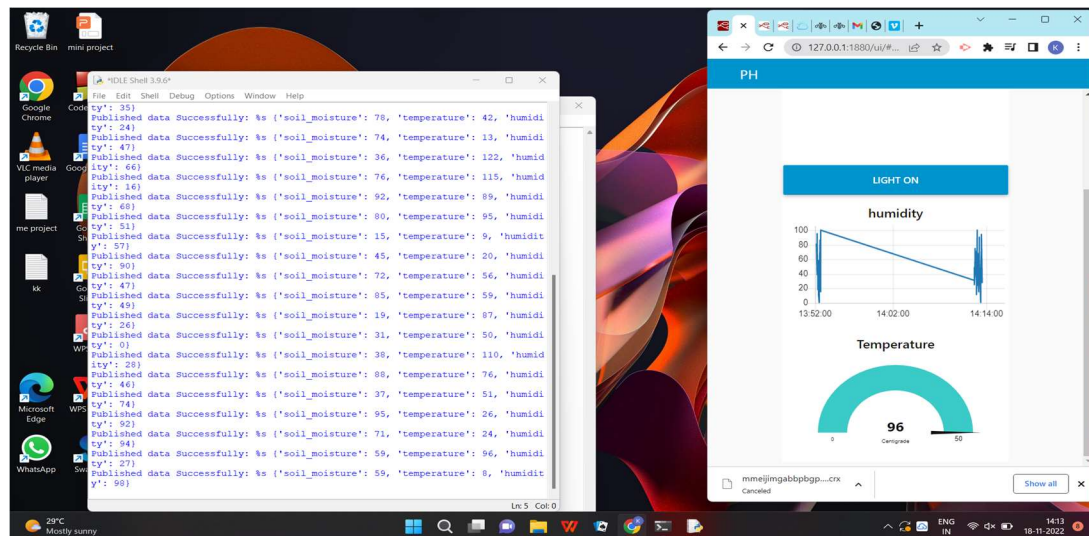
## Switchboard

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.

## **10.Bibliography**

IBM cloud reference: <https://cloud.ibm.com/>

IoT simulator : <https://watson-iot-sensor-simulator.mybluemix.net/>

OpenWeather : <https://openweathermap.org/>