SPRINT – 4

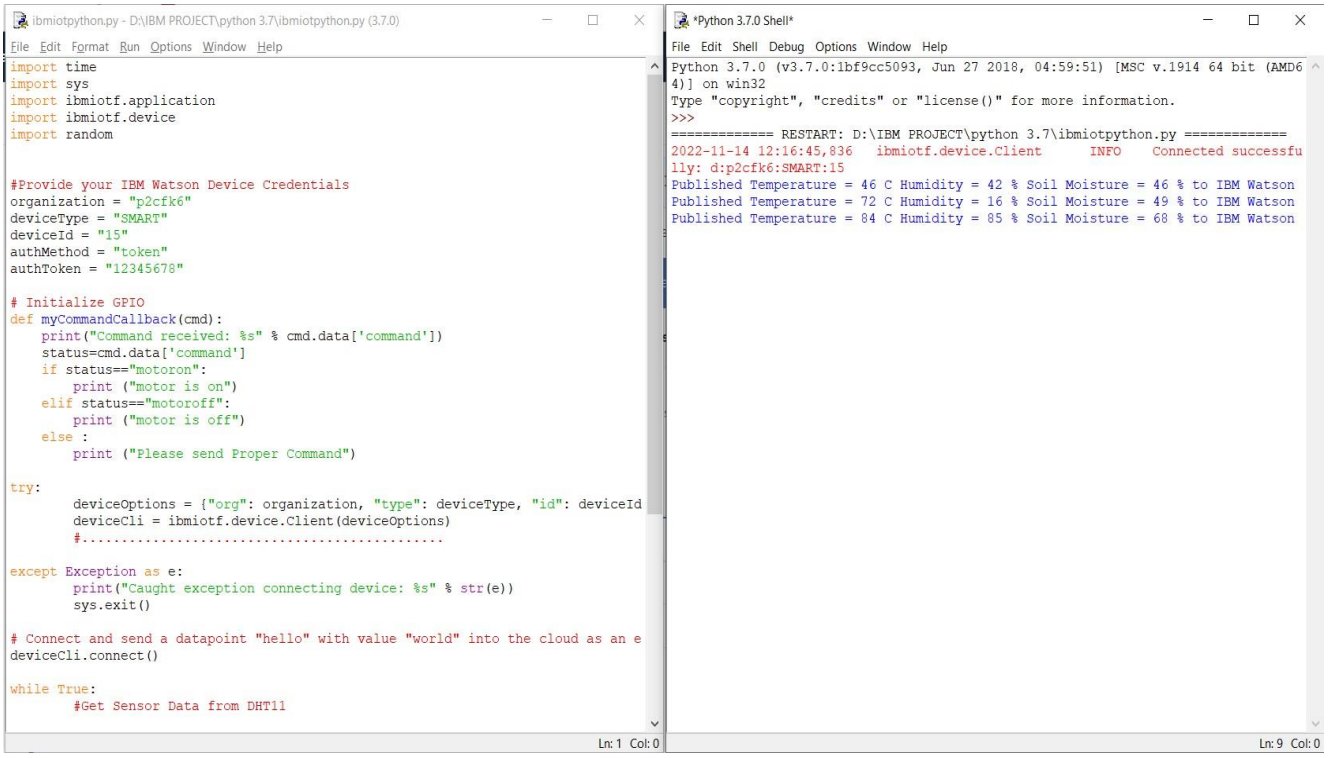
Team ID	PNT2022TMID00072
Project Name	Smart Farmer-IoT Enabled Smart Farming Application

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 = "p2cfk6"
deviceType = "SMART"
deviceId = "15"
authMethod = "token"
authToken = "12345678"

# 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
deviceCli.connect()

while True:
      temp=random.randint(0,100) # Temperature value
      Humid=random.randint(0,100) # Humidity value
      moisture = random.randint(0,100) # Soil moisture value

      data = { 'temp' : temp, 'Humid': Humid, 'Moisture' : moisture }
      #print data
      def myOnPublishCallback():
         print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid,    "Soil
Moisture = %s %%" % moisture, "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()
```

- DATA SEND FROM PYTHON PROGRAM :



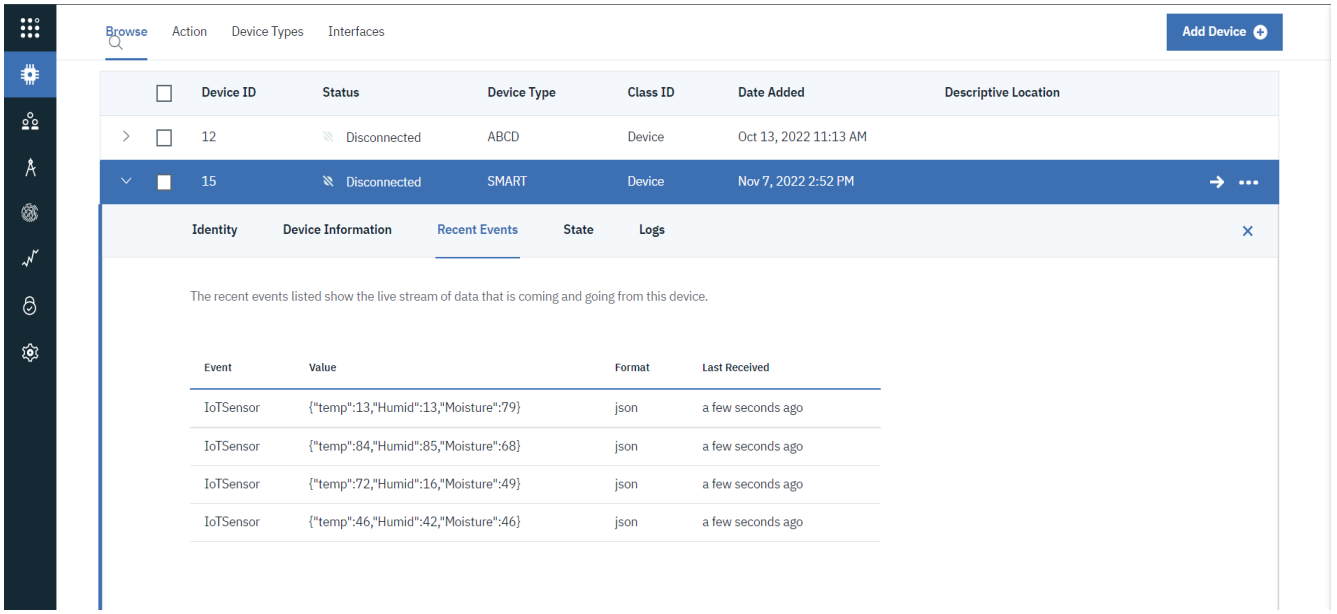
```
ibmiotpython.py - D:\IBM PROJECT\python 3.7\ibmiotpython.py (3.7.0)
File Edit Format Run Options Window Help
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "p2cfk6"
deviceType = "SMART"
deviceId = "15"
authMethod = "token"
authToken = "12345678"

# 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
        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 e
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11
```

Ln: 1 Col: 0

```
*Python 3.7.0 Shell*
File Edit Shell Debug Options Window Help
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: D:\IBM PROJECT\python 3.7\ibmiotpython.py ==========
2022-11-14 12:16:45,836   ibmiotf.device.Client      INFO    Connected successfu
lly: d:p2cfk6:SMART:15
Published Temperature = 46 C Humidity = 42 % Soil Moisture = 46 % to IBM Watson
Published Temperature = 72 C Humidity = 16 % Soil Moisture = 49 % to IBM Watson
Published Temperature = 84 C Humidity = 85 % Soil Moisture = 68 % to IBM Watson
```

Ln: 9 Col: 0

- DATA RECEIVED IN IBM CLOUD :



Browse | Action | Device Types | Interfaces

Add Device

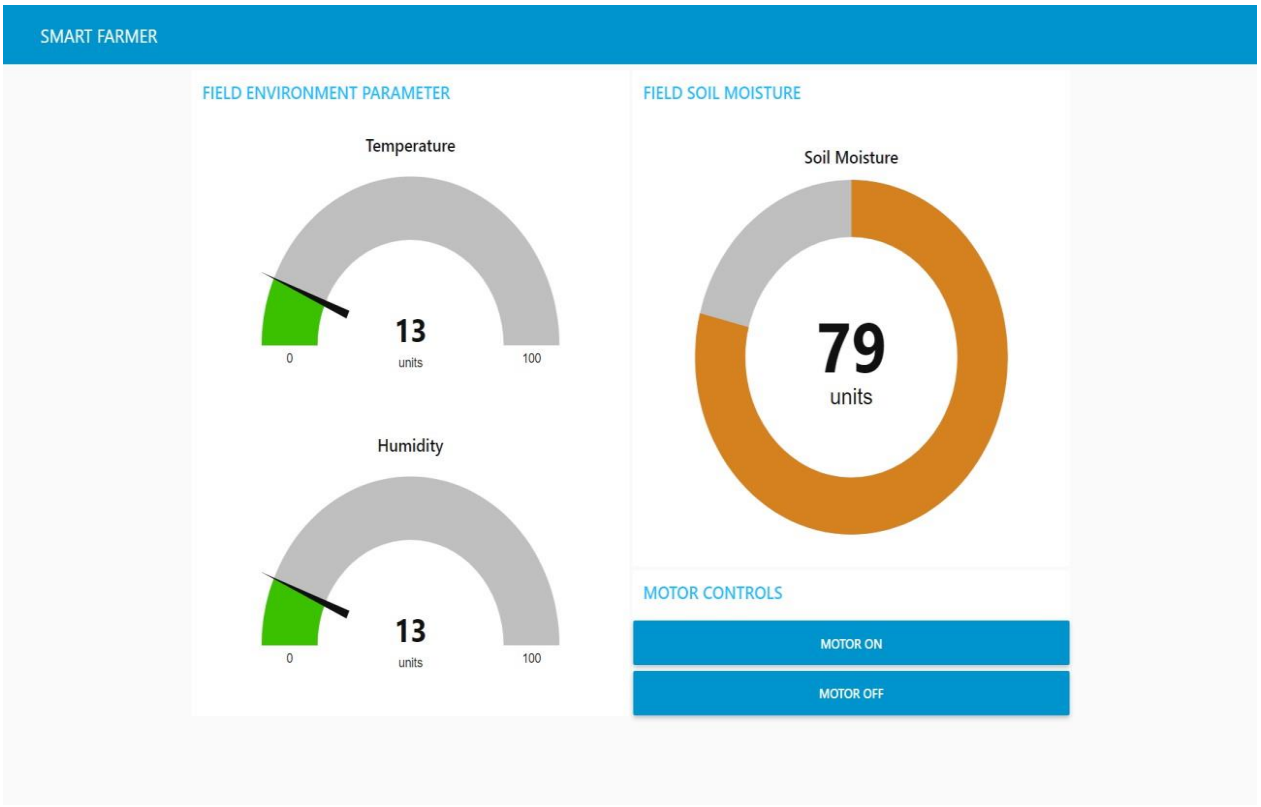
Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
12	Disconnected	ABCD	Device	Oct 13, 2022 11:13 AM	
15	Disconnected	SMART	Device	Nov 7, 2022 2:52 PM	

Identity | Device Information | Recent Events | State | Logs

The recent events listed show the live stream of data that is coming and going from this device.

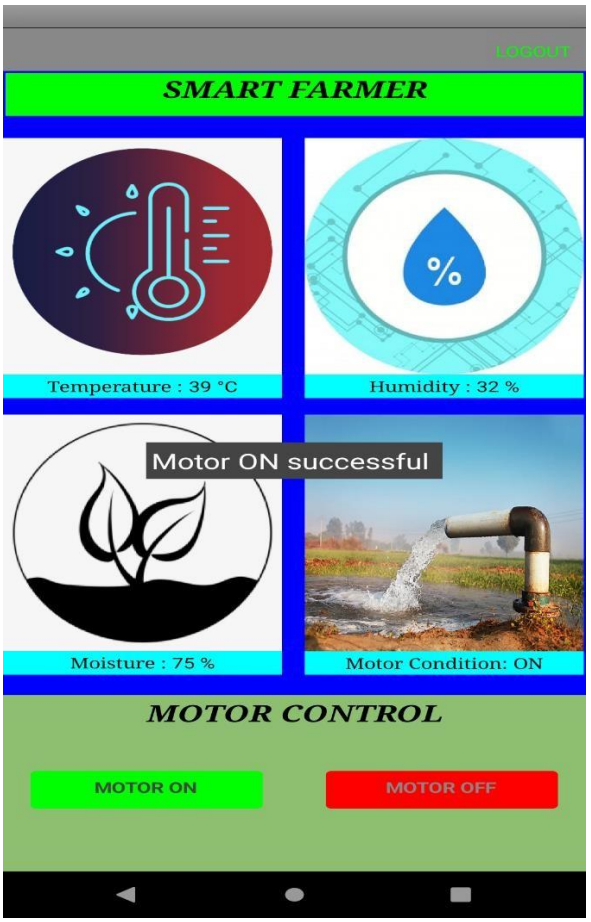
Event	Value	Format	Last Received
IoTSensor	{"temp":13,"Humid":13,"Moisture":79}	json	a few seconds ago
IoTSensor	{"temp":84,"Humid":85,"Moisture":68}	json	a few seconds ago
IoTSensor	{"temp":72,"Humid":16,"Moisture":49}	json	a few seconds ago
IoTSensor	{"temp":46,"Humid":42,"Moisture":46}	json	a few seconds ago

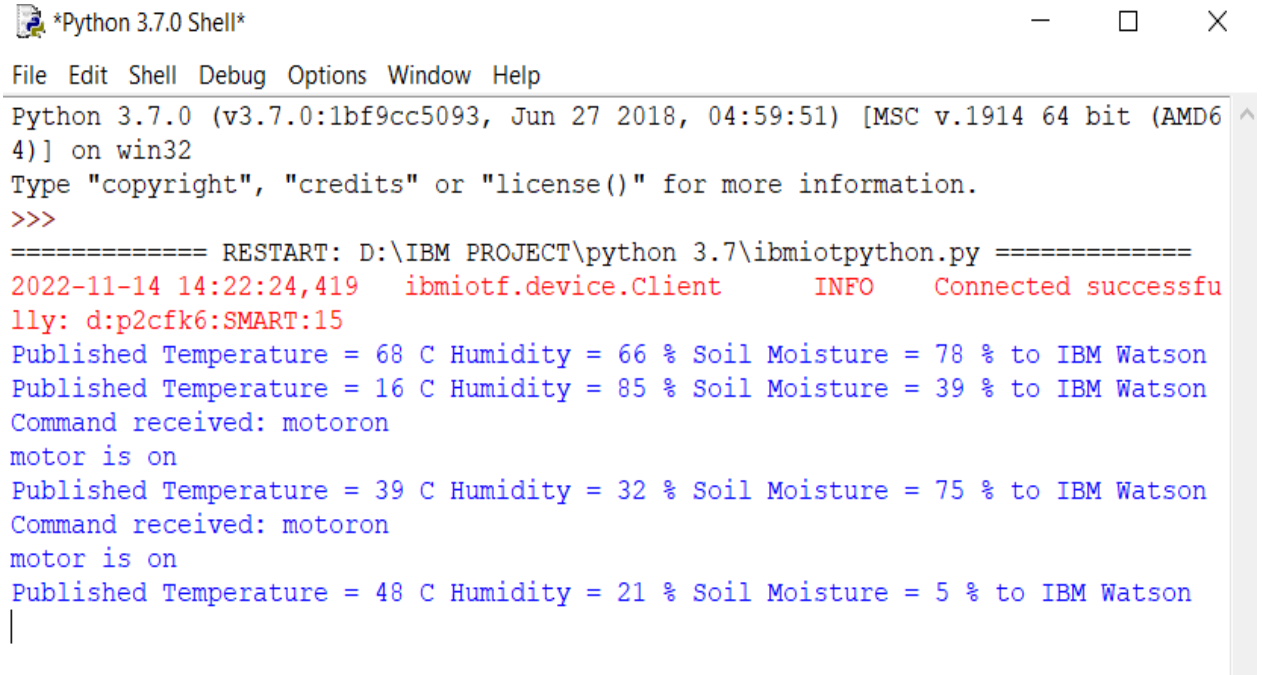
- DATA RECEIVED IN NODE – RED DASHBOARD (WEB UI)



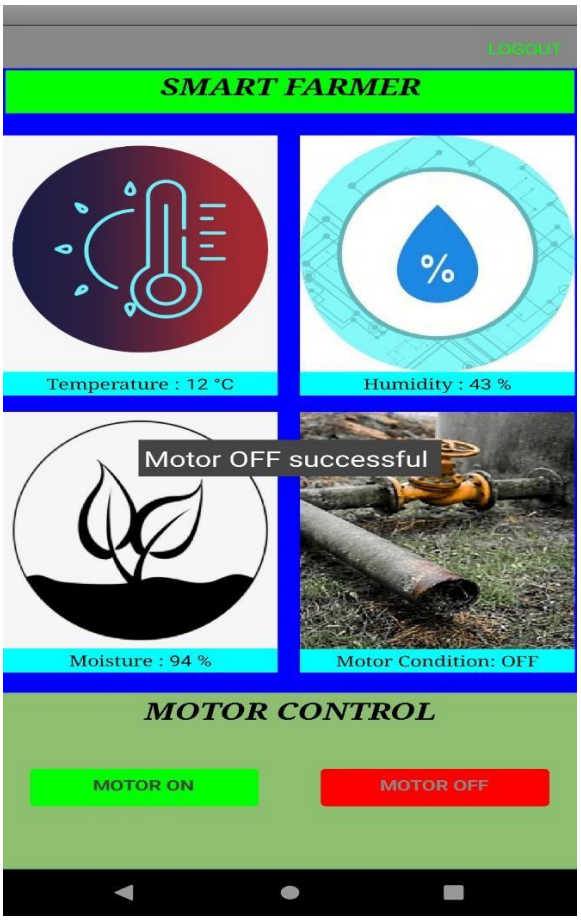
- DATA RECEIVED IN MOBILE APP

- COMMAND RECEIVED FROM WEB UI AND MOBILE APP
    - MOTOR ON COMMAND





- MOTOR OFF COMMAND



Python 3.7.0 Shell

File Edit Shell Debug Options Window Help

```
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: D:\IBM PROJECT\python 3.7\ibmiotpython.py =============
2022-11-14 14:22:24,419   ibmiotf.device.Client      INFO    Connected successfu
lly: d:p2cfk6:SMART:15
Published Temperature = 68 C Humidity = 66 % Soil Moisture = 78 % to IBM Watson
Published Temperature = 16 C Humidity = 85 % Soil Moisture = 39 % to IBM Watson
Command received: motoron
motor is on
Published Temperature = 39 C Humidity = 32 % Soil Moisture = 75 % to IBM Watson
Command received: motoron
motor is on
Published Temperature = 48 C Humidity = 21 % Soil Moisture = 5 % to IBM Watson
Published Temperature = 9 C Humidity = 29 % Soil Moisture = 44 % to IBM Watson
Published Temperature = 85 C Humidity = 64 % Soil Moisture = 17 % to IBM Watson
Command received: motoroff
motor is off
Published Temperature = 12 C Humidity = 43 % Soil Moisture = 94 % to IBM Watson
Command received: motoroff
motor is off
Published Temperature = 72 C Humidity = 86 % Soil Moisture = 0 % to IBM Watson
Published Temperature = 100 C Humidity = 95 % Soil Moisture = 90 % to IBM Watson
```