

IOT Based Smart Crop Protection System for Agriculture Final Code

Team ID – PNT2022TMID52016

Motor.py

```
import time

import sys

import ibmiotf.application # to install pip install ibmiotf import ibmiotf.device

# Provide your IBM Watson Device Credentials organization = "8gyz7t" # replace the ORG
ID

deviceType = "weather_monitor" # replace the Device type deviceId = "b827ebd607b5" #
replace Device ID authMethod = "token"

authToken = "LWVpQPpVQ166HWN48f" # Replace the authtoken

def myCommandCallback(cmd): # function for Callback if cmd.data['command'] ==
'motoron':

print("MOTOR ON IS RECEIVED")

elif cmd.data['command'] == 'motoroff': print("MOTOR OFF IS RECEIVED")

if cmd.command == "setInterval": if 'interval' not in cmd.data:

print("Error - command is missing required information: 'interval'")

else:

interval = cmd.data['interval'] elif cmd.command == "print":

if 'message' not in cmd.data:

print("Error - command is missing required information: 'message'")

else:

output = cmd.data['message'] print(output)

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:

deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud deviceCli.disconnect()

```

Sensor.py

```

import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

# Provide your IBM Watson Device Credentials organization = "8gyz7t" # replace the ORG
ID

deviceType = "weather_monitor" # replace the Device type deviceId = "b827ebd607b5" #
replace Device ID authMethod = "token"

authToken = "LWVpQPpVQ166HWN48f" # Replace the authToken

def myCommandCallback(cmd):

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

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:

temp=random.randint(0,100) pulse=random.randint(0,100) soil=random.randint(0,100)

data = { 'temp' : temp, 'pulse': pulse , 'soil':soil}

#print data

def myOnPublishCallback():

print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % pulse,"Soil
Moisture = %s %%" % soil,"to IBM Watson")

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

if not success:

print("Not connected to IoT") time.sleep(1)

deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud deviceCli.disconnect()

```

Node-RED Flow :

```

[
  { "id": "625574ead9839b34",
    "type": "ibmiotout",
    "z": "630c8601c5ac3295",
    "authentication": "apiKey",
    "apiKey": "ef745d48e395ccc0",
    "outputType": "cmd",
    "deviceId": "b827ebd607b5",
    "deviceType": "weather_monitor",
    "eventCommandType": "data",

```

```
"format":"json",
"data":"data", "qos":0,
"name":"IBM IoT",
"service":"registered",
"x":680,
"y":220,
"wires":[
},
{
"id":"4cff18c3274cccc4",
"type":"ui_button",
"z":"630c8601c5ac3295",
"name":"","group":"716e956.00eed6c",
"order":2,
"width":"0",
"height":"0",
"passthru":false,
"label":"MotorON",
"tooltip":"",
"color":"",
"bgcolor":"",
"className":"",
"icon":"","payload":{"command\\":"motoron\\"}, "payloadType":"str",
"topic":"motoron",
"topicType":"str", "x":360,
"y":160, "wires":[["625574ead9839b34"]]},
{
```

```
"id":"659589baceb4e0b0", "type":"ui_button", "z":"630c8601c5ac3295",
"name": "",
"group":"716e956.00eed6c", "order":3,
"width":0,
"height":0, "passthru":true, "label":"MotorOFF",
"tooltip": "",
"color": "",
"bgcolor": "",
"className": "",
"icon": "", "payload": {"command": "motoroff"}, "payloadType": "str",
"topic": "motoroff",
"topicType": "str", "x":350,
"y":220, "wires": [{"625574ead9839b34"}]},
{"id":"ef745d48e395ccc0", "type":"ibmiot", "name":"weather_monitor", "keepalive":60,
"serverName": "", "cleansession":true, "appId": "",
"shared":false},
{"id":"716e956.00eed6c",
"type":"ui_group",
"name":"Form", "tab":"7e62365e.b7e6b8", "order":1,
"disp":true, "width":6, "collapse":false},
{"id":"7e62365e.b7e6b8",
"type":"ui_tab",
"name":"contorl",
"icon":"dashboard", "order":1, "disabled":false, "hidden":false}
]
[
{
```

```
"id":"b42b5519fee73ee2", "type":"ibmiotin", "z":"03acb6ae05a0c712",
"authentication":"apiKey", "apiKey":"ef745d48e395ccc0", "inputType":"evt",
"logicalInterface":""," "ruleId":""," "deviceId":"b827ebd607b5", "applicationId":"","
"deviceType":"weather_monitor", "eventType":"+",
"commandType":"","
"format":"json",
"name":"IBMIoT", "service":"registered", "allDevices":""," "allApplications":"","
"allDeviceTypes":""," "allLogicalInterfaces":""," "allEvents":true, "allCommands":"","
"allFormats":""," "qos":0,
"x":270,
"y":180,
"wires":[["50b13e02170d73fc","d7da6c2f5302ffaf","a949797028158f3f","a71f164bc3
78bcf1"]]
},
{ "id":"50b13e02170d73fc",
"type":"function", "z":"03acb6ae05a0c712", "name":"Soil Moisture",
"func":"msg.payload = msg.payload.soil;\nglobal.set('s',msg.payload);\nreturn msg;",
"outputs":1,
"noerr":0, "initialize":"","
"finalize":"","
"libs":[], "x":490,
"y":120,
"wires":[["a949797028158f3f","ba98e701f55f04fe"]]
},
{
"id":"d7da6c2f5302ffaf", "type":"function", "z":"03acb6ae05a0c712", "name":"Humidity",
"func":"msg.payload = msg.payload.pulse;\nglobal.set('p',msg.payload)\nreturn msg;",
"outputs":1,
"noerr":0, "initialize":"","
```

```
"finalize":"","  
"libs":[], "x":480,  
"y":260, "wires":[["a949797028158f3f", "70a5b076eeb80b70"]]  
},  
{ "id":"a949797028158f3f",  
"type":"debug", "z":"03acb6ae05a0c712", "name":"IBMo/p", "active":true, "tosidebar":true,  
"console":false, "tostatus":false, "complete":"payload", "targetType":"msg",  
"statusVal":""," "statusType":"auto", "x":780,  
"y":180,  
"wires":[]  
},  
{  
"id":"70a5b076eeb80b70", "type":"ui_gauge", "z":"03acb6ae05a0c712", "name":"","  
"group":"f4cb8513b95c98a4", "order":6,  
"width":"0",  
"height":"0",  
"gtype":"gage",  
"title":"Humidity",  
"label":"Percentage(%)",  
"format":"{{ value }}", "min":0, "max":"100",  
"colors":["#00b500", "#e6e600", "#ca3838"],  
"seg1":"","  
"seg2":"","  
"className":""," "x":860,  
"y":260,  
"wires":[]  
},  
{
```

```
"id":"a71f164bc378bcf1", "type":"function", "z":"03acb6ae05a0c712",
"name":"Temperature",

"func":"msg.payload=msg.payload.temp;\nglobal.set('t',msg.payload);\nreturn msg;",
"outputs":1,

"noerr":0, "initialize": "",

"finalize": "",

"libs":[], "x":490,

"y":360,

"wires":[["8e8b63b110c5ec2d","a949797028158f3f"]]

},

{

"id":"8e8b63b110c5ec2d", "type":"ui_gauge", "z":"03acb6ae05a0c712", "name": "",
"group":"f4cb8513b95c98a4", "order":11,

"width":"0",

"height":"0",

"ctype":"gauge", "title":"Temperature", "label":"DegreeCelcius",

"format":"{{ value }}", "min":0, "max":"100",

"colors":["#00b500", "#e6e600", "#ca3838"], "seg1": "",

"seg2": "",

"className": "", "x":790,

"y":360,

"wires":[]

},

{

"id":"ba98e701f55f04fe", "type":"ui_gauge", "z":"03acb6ae05a0c712", "name": "",
"group":"f4cb8513b95c98a4", "order":1,

"width":"0",

"height":"0",
```



```
"gtype":"gage", "title":"Soil Moisture", "label":"Percentage(%)",  
"format":"{{value}}", "min":0, "max":"100",  
"colors":["#00b500", "#e6e600", "#ca3838"], "seg1": "",  
"seg2": "",  
"className": "", "x": 790,  
"y": 120,  
"wires": []  
},  
{  
"id": "a259673baf5f0f98", "type": "httpin", "z": "03acb6ae05a0c712", "name": "",  
"url": "/sensor",  
"method": "get", "upload": false, "swaggerDoc": "", "x": 370,  
"y": 500,  
"wires": [["18a8cdbf7943d27a"]]  
},  
{  
"id": "18a8cdbf7943d27a", "type": "function", "z": "03acb6ae05a0c712",  
"name": "httpfunction",  
"func": "msg.payload{\"pulse\":global.get('p'),\"temp\":global.get('t'),\"soil\":global.get('s')};\nreturn msg;",  
"outputs": 1,  
"noerr": 0, "initialize": "",  
"finalize": "",  
"libs": [], "x": 630,  
"y": 500, "wires": [["5c7996d53a445412"]]  
},  
{ "id": "5c7996d53a445412",  
"type": "httpresponse", "z": "03acb6ae05a0c712", "name": "",
```

```
"statusCode":"","  
"headers":{ }, "x":870,  
"y":500,  
"wires":[]  
},  
{  
"id":"ef745d48e395ccc0", "type":"ibmiot", "name":"weather_monitor", "keepalive":"60",  
"serverName":""," "cleansession":true, "appId":"","  
"shared":false},  
{  
"id":"f4cb8513b95c98a4", "type":"ui_group",  
"name":"monitor", "tab":"1f4cb829.2fdee8", "order":2,  
"disp":true, "width":"6", "collapse":false, "className":""  
},  
{  
"id":"1f4cb829.2fdee8",  
"type":"ui_tab",  
"name":"Home",  
"icon":"dashboard", "order":3, "disabled":false, "hidden":false }
```