PROJECT DEVELOPMENT PHASE

PROJECT DEVELOPMENT DELIVERY OF SPRINT 3

Date	12 Nov 2022
Team ID	PNT2022TMID38592
Project Name	SmartFarmer-IOT Enabled Smart Farming Agriculture
Marks	

CODE EXPLANATION

The below described code is what we have developed for connecting with IBM IoT cloud. The code also connects with the Node- Red Service and displays the output frequently. Once the Code is simulated, the code runs with the output of temperature, humidity and pH value. This code links with the IBM IoT Platform and then to Node-Red, Finally the result is displayed in our Application.

PYTHON CODE:

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

#Narasimhan IBM
organization = "59lw3i"
deviceType = "device_1"
deviceId = "12345"
authMethod = "token"
authToken = "123456789"

#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()

#CONNECCT
deviceCli.connect()

while True:
    temperature=random.randint(0,100)
    humidity=random.randint(0,100)
    moisture=random.randint(0,100)

    data={'temperature':temperature,'humidity':humidity,'moisture':moisture}

    def myOnPublishCallback():
        print("Published Temperature = %s C"%temperature,"Humidity = %s %%" %humidity,"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

deviceCli.disconnect()
```