Project Development Phase

Sprint-2

Project Title: Real-time River Water Quality Monitoring And Control System

Team ID: PMT2022TMID34542

Sprint Description:

In this sprint, lets discuss about the python code development and its simulation which is connected with the IBM IoT Cloud and NodeRed Service.

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

#Provide your IBM Watson Device Credentials
Organization = "hwaz8c"
DeviceType = "1234"
deviceID = "12345"
authMethod = "token"
authToken = "123456789"

# Initialize GPTO
def mycommandcallback(cmd):
  print("command received: %s" % cmd.data['command'])
  status=cmd.data['command']
  if status =="lighton":
    print ("led is on")
  elif status =="lightoff":
    print("led 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 oftype "greeting" 10 times
deviceCli.connect()


while True:
  #Get Sensor Data from DHT11
  temp=random.randint(90,110)
  Humid=random.randint(60,100)
  PH=int(7)

  data={ 'temp' : temp, 'Humid' : Humid , 'PH' : PH}


  #print data
  def myOnPublishCallback() :
    print ("Published Temperature = %s C" % temp, "Humidity = %s
%%" % Humid, "Ph value= 7 ,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()
```