ProjectDevelopmentPhase
Sprint-3

Date	14thNovember2022
Team ID	PNT2022TMID35067
ProjectName	Project:SignswithSmartConnectivityforBetterRoadSafety.
Marks	20Marks

Sprint	FunctionalRequirement	User StoryNumber	UserStory/Task	Story Points	Priority	TeamMembers
Sprint-3		US-1	Develop a pythonscripttopublishrandomsensordatasuchastemperature,humidity,visibilitytotheIBMIoTplatform.	7	High	R.Anciya N.Ezhil mugisha P.Brindha N.Bavithra
Sprint-3		US-2	Afterdevelopingpythoncode,commandsarereceivedprintthestatementswhichrepresentthecontrolofthedevices.	5	Medium	R.Anciya N.Ezhil Mugisha P.Brindha N.Bavithra
Sprint-3		US-3	Publish Data to theIBMCloud.	8	High	R.Anciya N.Ezhil Mugisha P.Brindha N.Bavithra

US-1Developapythonscripttopublishrandomsensordatasuchastemperature, humidity and visibility to the IBM IoT Platform

```
import
timeimport
sys
importibmiotf.applicationi
mport
ibmiotf.deviceimportrand
om

#ProvideyourIBMWatsonDeviceCredentials
organization="33lnun"
deviceType="PNT2022TMID35067"deviceId
="PNT2022TMID35067"
authMethod="token"
authToken="BGM(9-Tgfy&lrHmglp"


#IntializeGPIO

defmyCommandCallback(cmd):
    print("Commandreceived:%s%cmd.data['command']")status=cmd.data['comman
    d']
    ifstatus=="lighton":print
        ("ledison")
    else:
        print("lediso ff")

    #print(cmd)

try:
    deviceOptions={"org":organization,"type":deviceType,"id":deviceId,"auth-
method":authMethod,"auth-token":authToken}
    deviceCli=ibmiotf.device.Client(deviceOptions)#.....................
    ...

exceptExceptionase:
    print("Caughtexceptionconnectingdevice:%s"%str(e))sys.exit()

    #Connectandsendadatapoint"hello"withvalue"world"intothecloudasanevent
of type "greeting" 10 times
    deviceCli.connect()

whileTrue:
    #GetSensorDatafromDHT11

    temp=random.randint(0,100)h
    umid=random.randint(0,100)
```

```
    visi=random.randint(0,100)

    data={'temperature'=temp,'humidity'=humid,'visibility'=visi}
    #printdata
    defmyOnPublishCallback():
        print("Publishedtemperature=%sC"%temp,"humidity=%s%%"
%humid,"visibility=%s%%"%visi,"toIBMWatson")

        success=deviceCli.publishEvent("IoTSensor","json",data,qos=0,on_publish=my
OnPublishCallback)
        ifnotsuccess:
            print("NotconnectedtoIoTF")time.slee
            p(1)

            deviceCli.commandCallback=myCommandCallback

            #Disconnect thedeviceandapplicationfrom thecloud
            deviceCli.disconnect(
                )
```