SPRINT-IV

PROJECT DEVELOPMENT PHASE

DATE	**14 NOVEMBER 2022**
TEAM ID	**PNT2022TMID49507**
PROJECT NAME	**PROJECT-IOT BASED SAFETY GADGETS FOR CHILD SAFETY MONITORING AND NOTIFICATION**

CODING:

```
import serial as io

gsm = io.Serial("/dev/ttyUSB0",9600, timeout=0.5)

gsm.flush()


def sendSms(msg):
   print("Sending SMS\n")
   gsm.write(b'AT+CMGF=1\r\n')
   sleep(0.5)
   gsm.write(b'AT+CMGS=\"')
   serialcmd = args["mobile"]
   gsm.write(serialcmd.encode())
   gsm.write(b'\"\r\n')
   sleep(0.5)
   data = msg
   gsm.write(data.encode())
```

```
    gsm.write(b'\x1A')
    sleep(3)


sendSms("Alert child is missing…..")
```

CODING NOTIFICATION SENDING:

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



#Provide your IBM Watson Device Credentials
organization = "8bchp4"
deviceType = "pdharani"
deviceId = "12345678"
authMethod = "token"
authToken = "12345678"
```

```
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:
    #Get Sensor Data from DHT11


    name="pooja"
    #latitude=8.9179987
    #longitude=98.0527826


    latitude=10.7905
    longitude=80.2707
```

```
        data = { 'name' : name, 'lat': latitude, 'lon':longitude }

        #print data

        def myOnPublishCallback():

            print ("Published name = %s " % name, "lat = %s " % latitude, "lon = %s " % longitude, "to IBM
Watson")



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

        if not success:

            print("Not connected to IoTF")

        time.sleep(5)



        deviceCli.commandCallback = 'myCommandCallback'




client.disconnect()
```