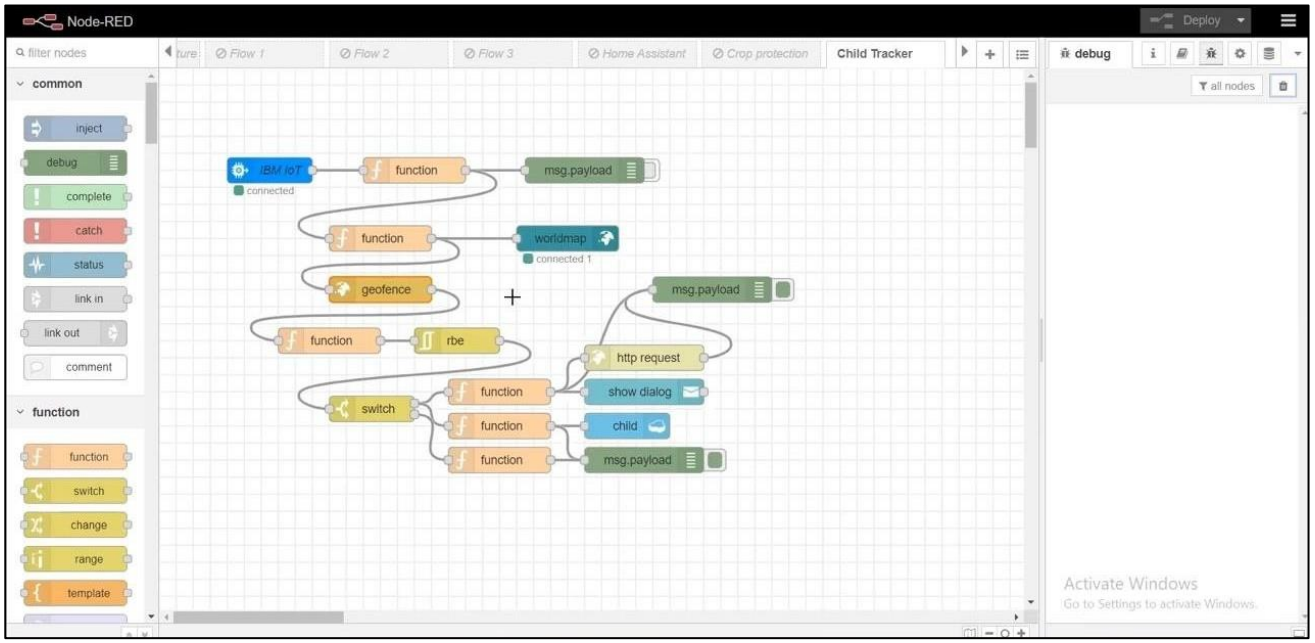
WEB – APPLICATION USING NODE-RED

AIM:

Develop the web application using Node-RED

STEPS TO BE DONE:

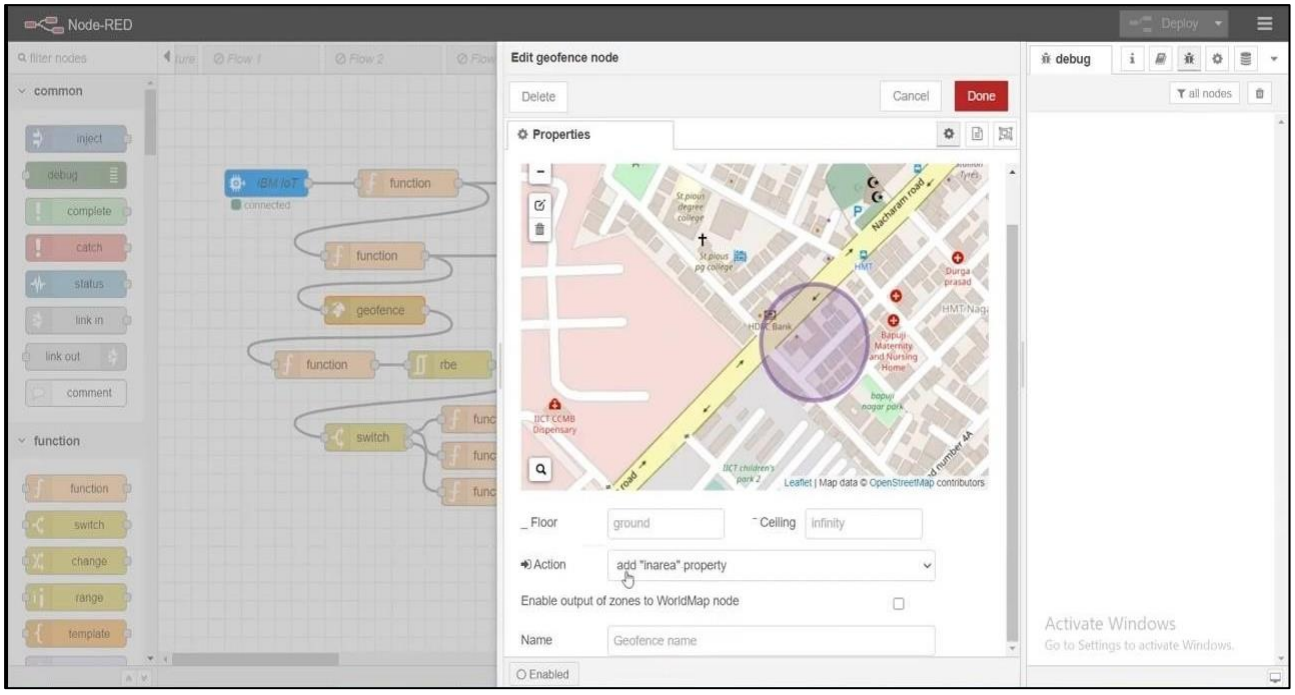
- Opened a Node-RED project



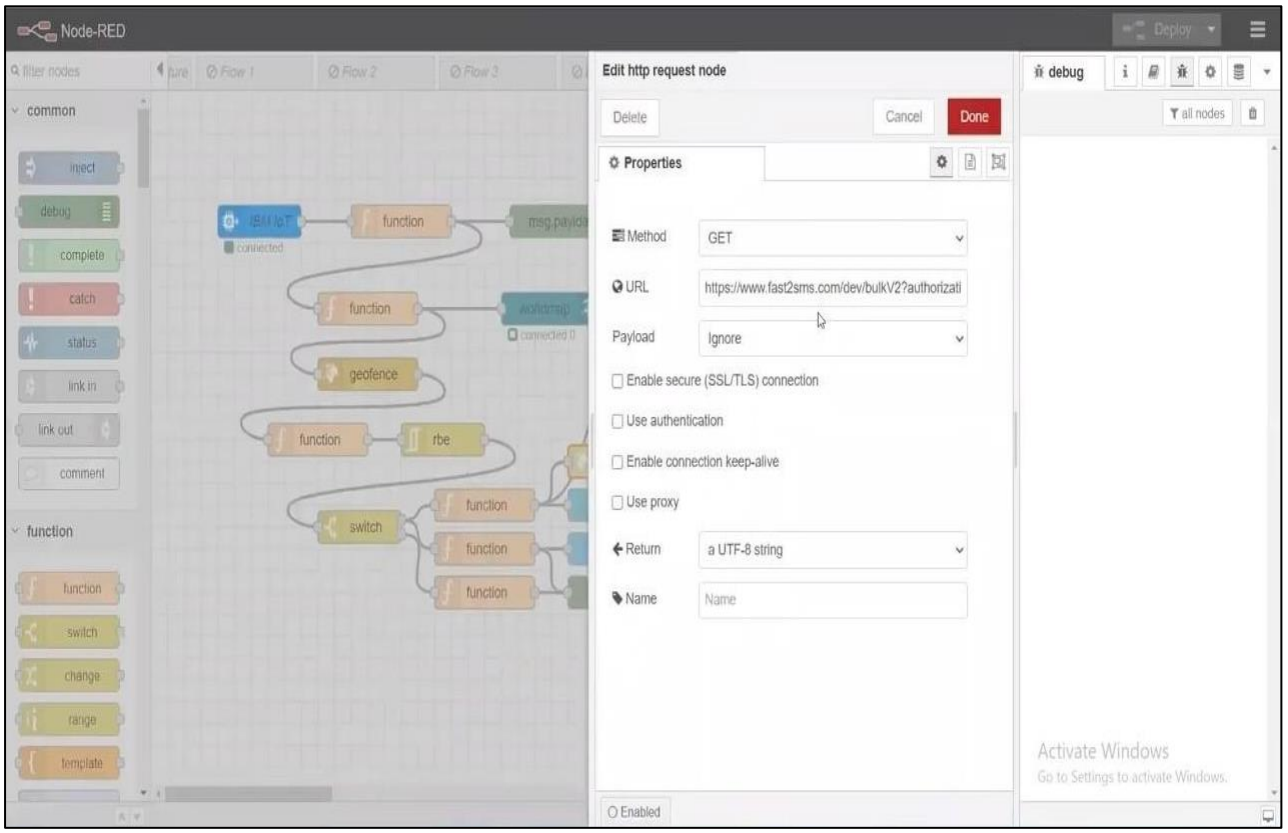
- Develop a code to get child Location using Python

```
"geofence Coding.py - C:/Users/HOME/Documents/7th SEM/geofence Coding.py (3.7.0)"
File  Edit  Format  Run  Options  Window  Help
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
organization = "op5ijq"
deviceType = "priya0210devicetype"
deviceId = "priya0210deviceid"
authMethod = "use-token-auth"
authToken = "aWgbI$!h_bEOTlxmjW"
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    else :
        print ("led is off")

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()


deviceCli.connect()

while True:


        temp=random.randint(0,100)
        Humid=random.randint(0,100)

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

        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(1)

        deviceCli.commandCallback = myCommandCallback

deviceCli.disconnect()
```

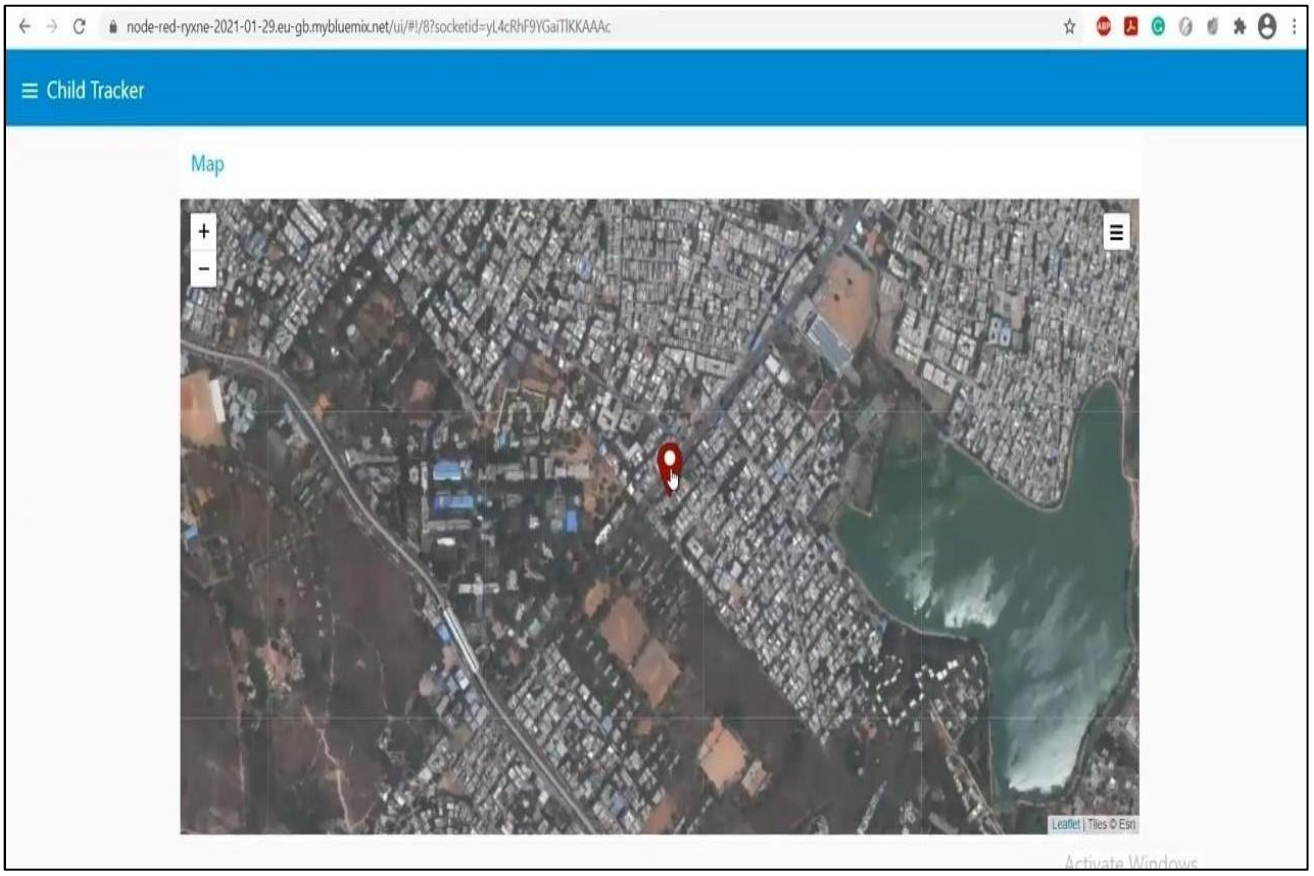
- Create **GEOFENCE** mode



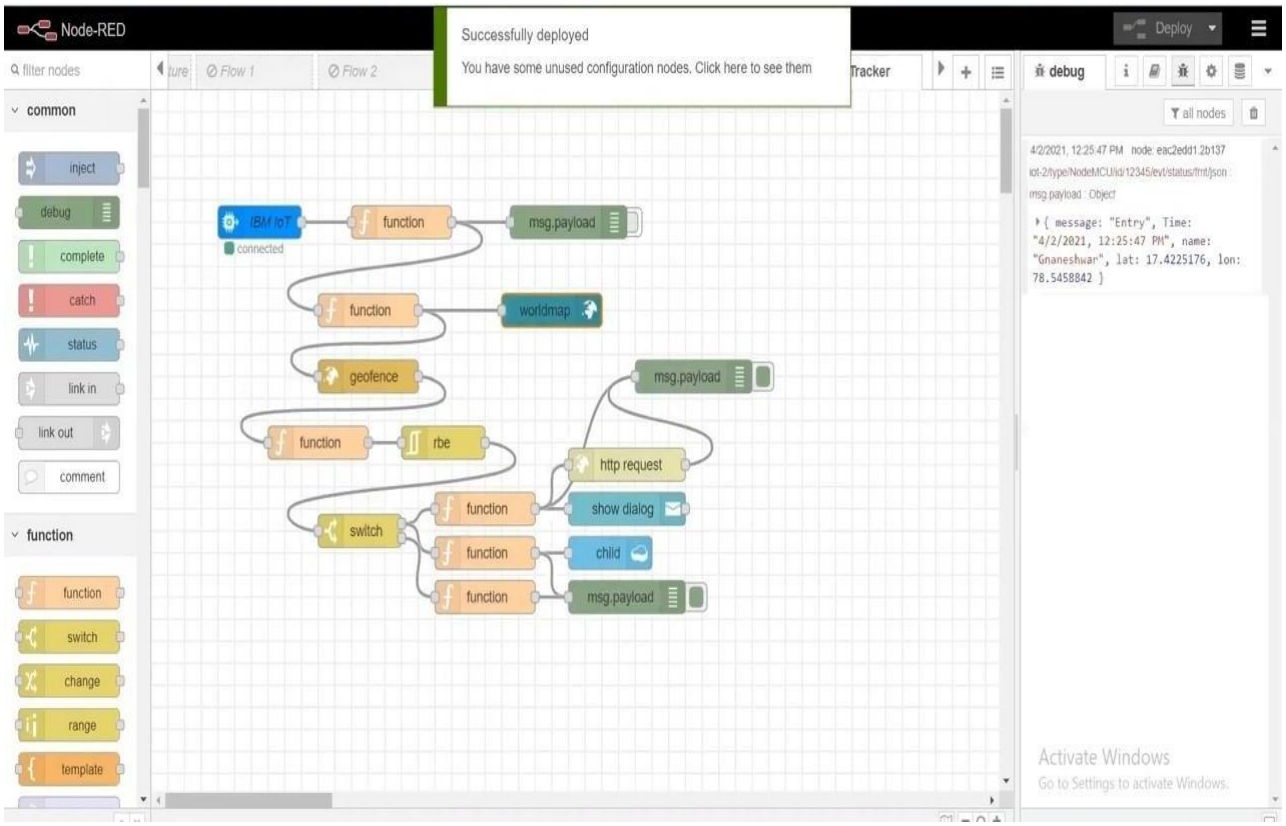
- Edit the **HTTP Request URL**

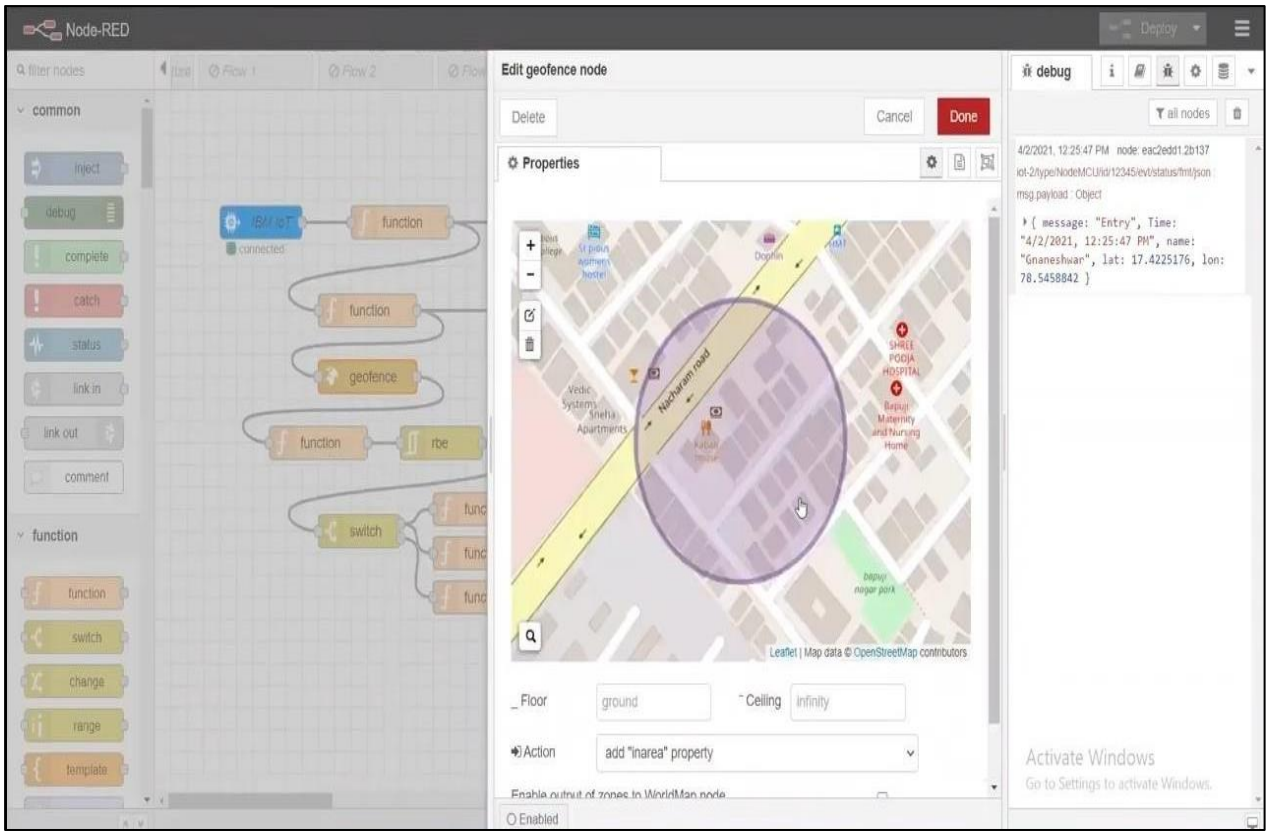


- Located the Child's location

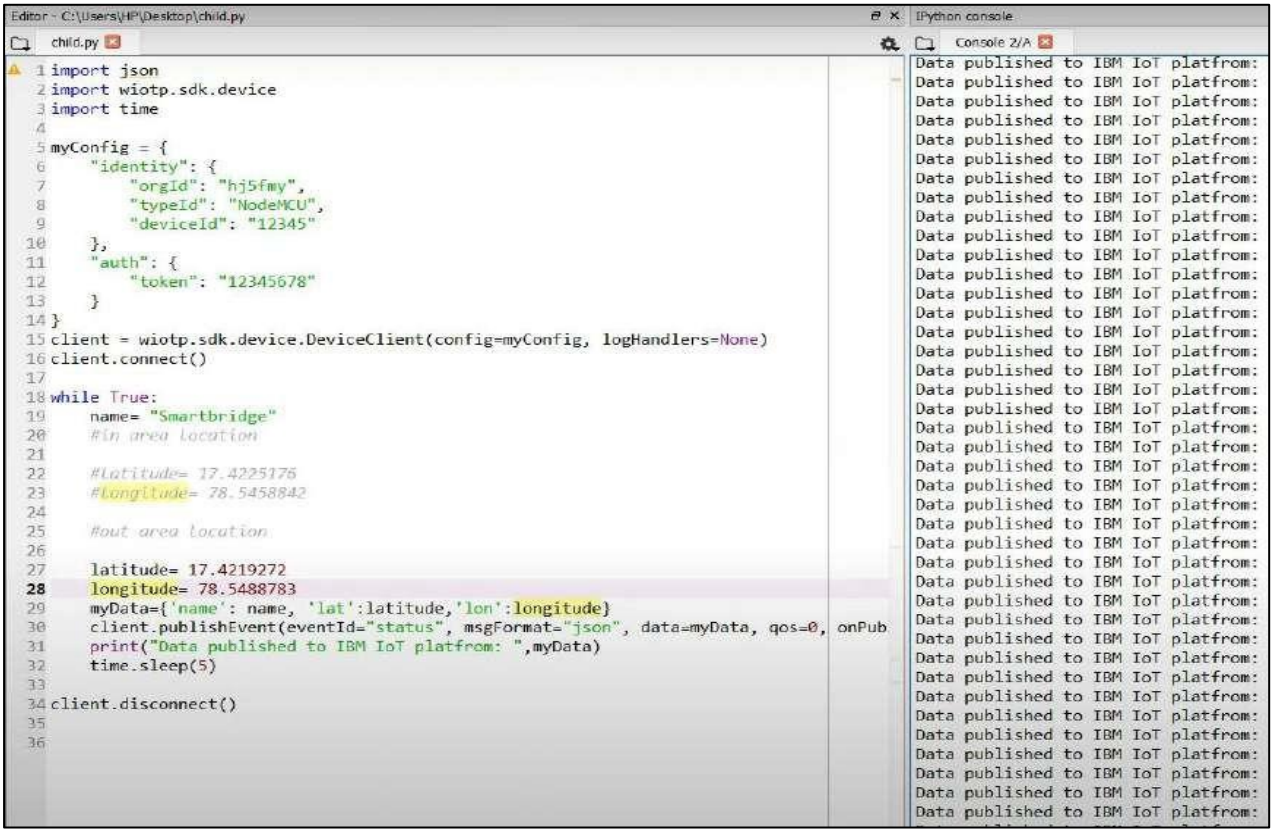


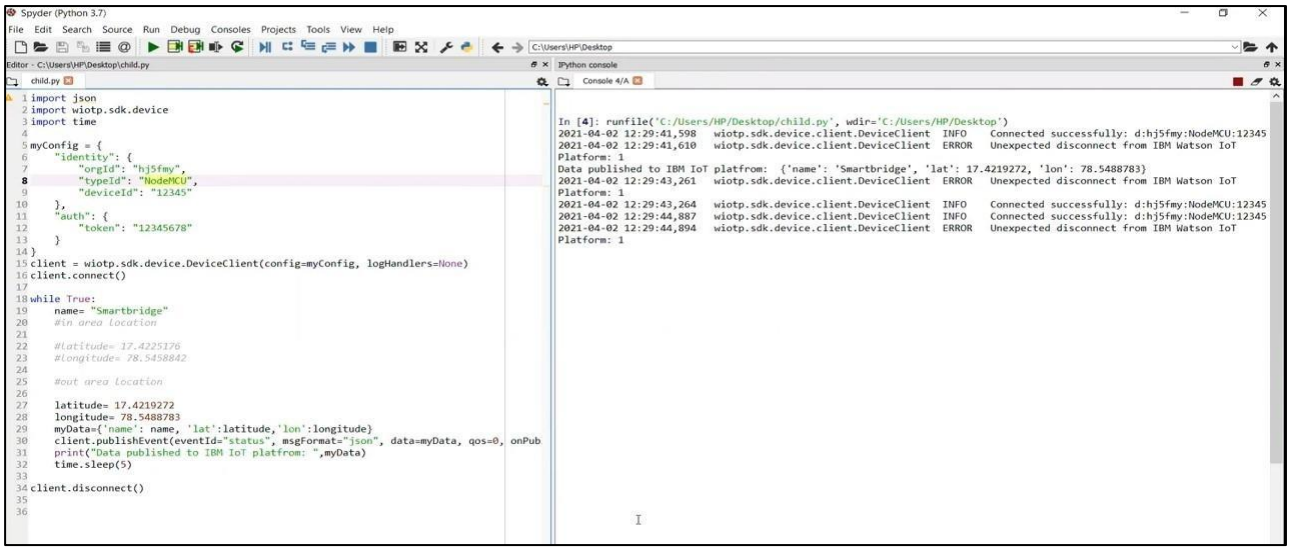
- **GEOFENCE** node created



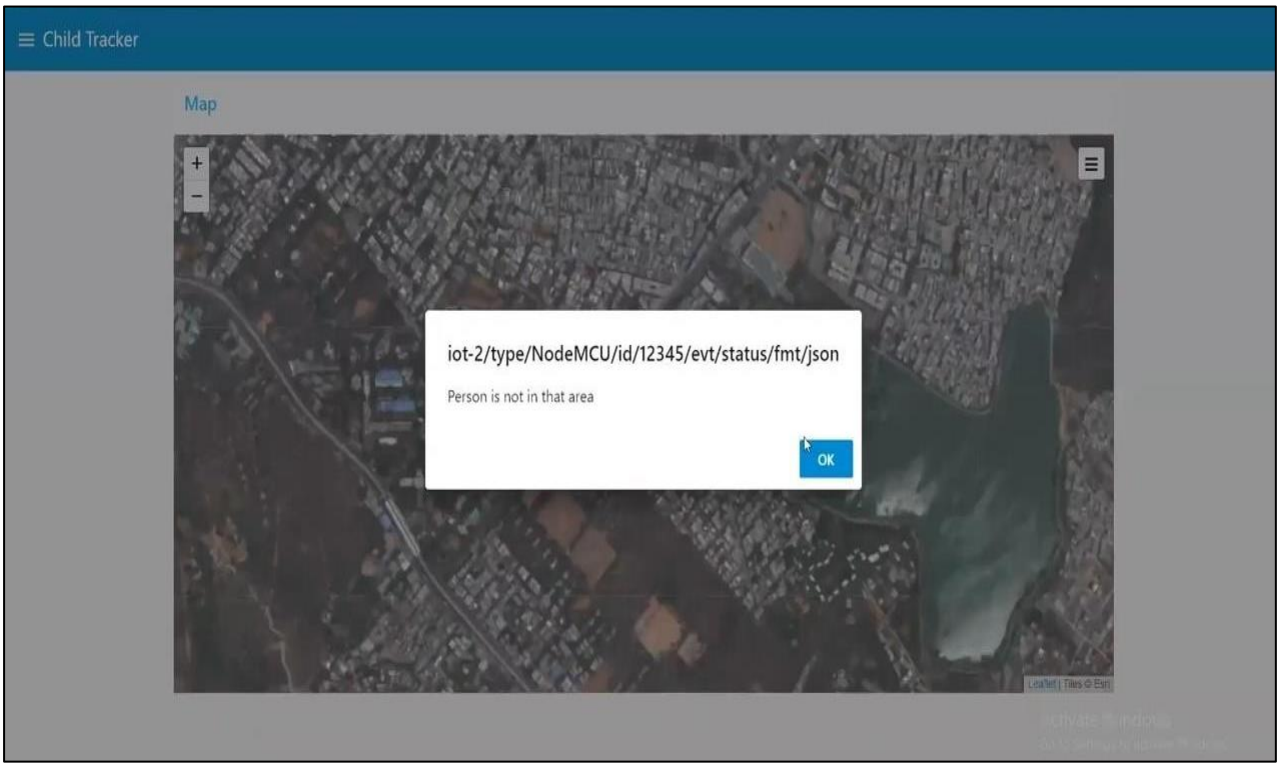


- Requests are being send to **IBM -CLOUD**





- After running the script, the web UI shows “Person is not in theparticular area”



RESULT:

Thus the web-application using node-red was developed successfully.