

Develop The Web Application Using Node-RED

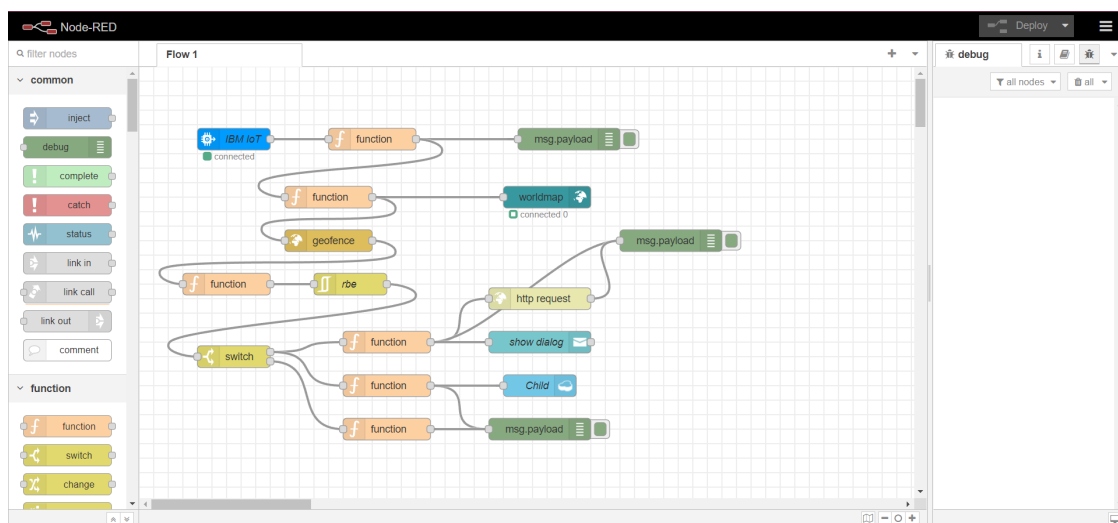
IoT Based Safety Gadget for Child Safety Monitoring and Notification

Team ID : PNT2022TMID27117

Aim: Develop the web application using Node-RED

Steps Followed:

- Opened a Node-RED project



- Added code to get child location in python

```
child.py - C:\Users\Anu\AppData\Local\Programs\Python\Python37\child.py (3.7.0)
File Edit Format Run Options Window Help

import json
import wiotp.sdk.device
import time
myConfig = {

    "identity":{
        "orgId": "4o1qab",
        "typeId": "TestDeviceType",
        "deviceId": "12345"
    },
    "auth": {
        "token": "pnhXvztN-sRMKvshxyi"
    }
}

client= wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
    name = "Smartbridge"
    #in area location

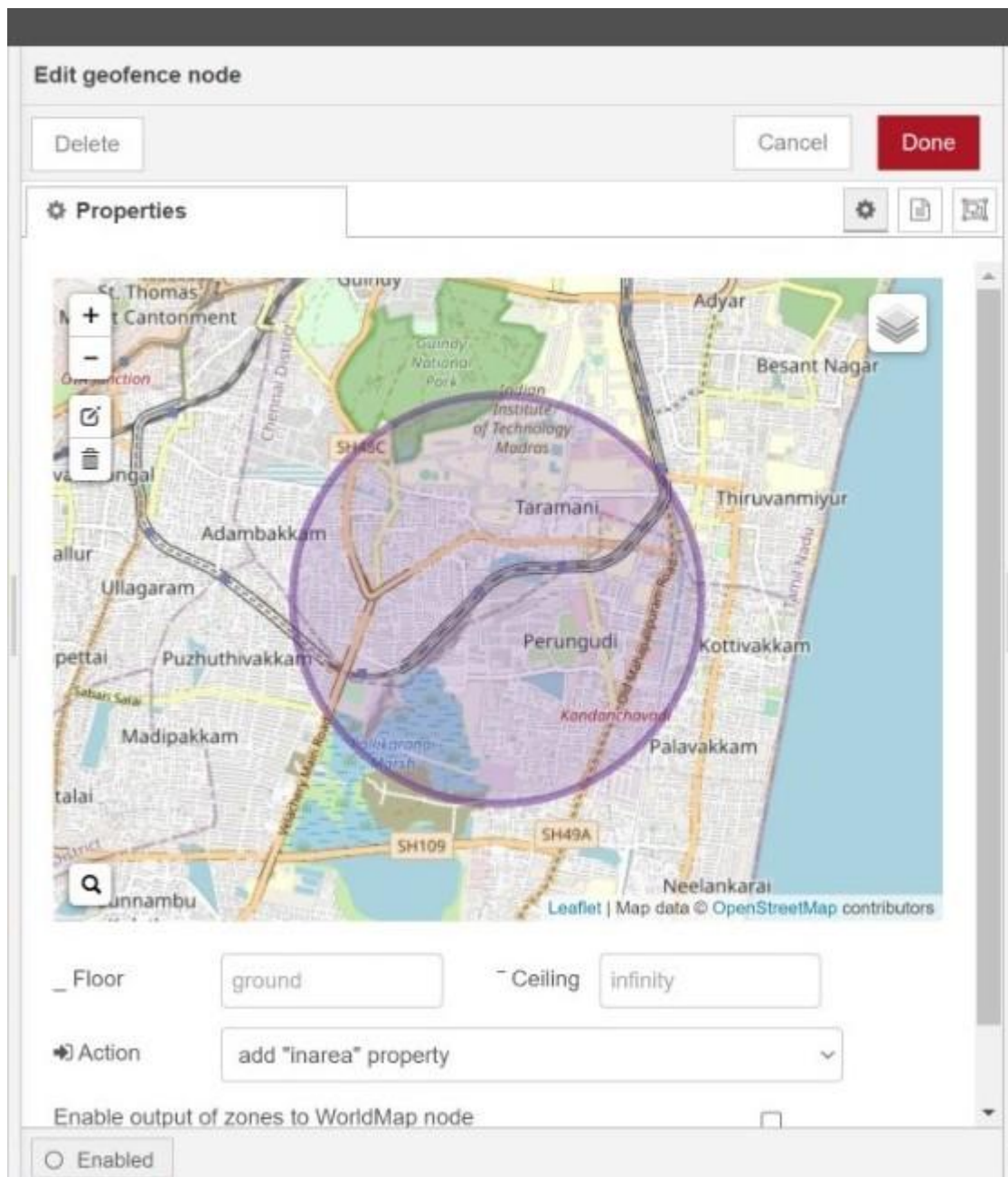
    latitude = 17.4225176
    longitude = 78.5456842

    #out area location

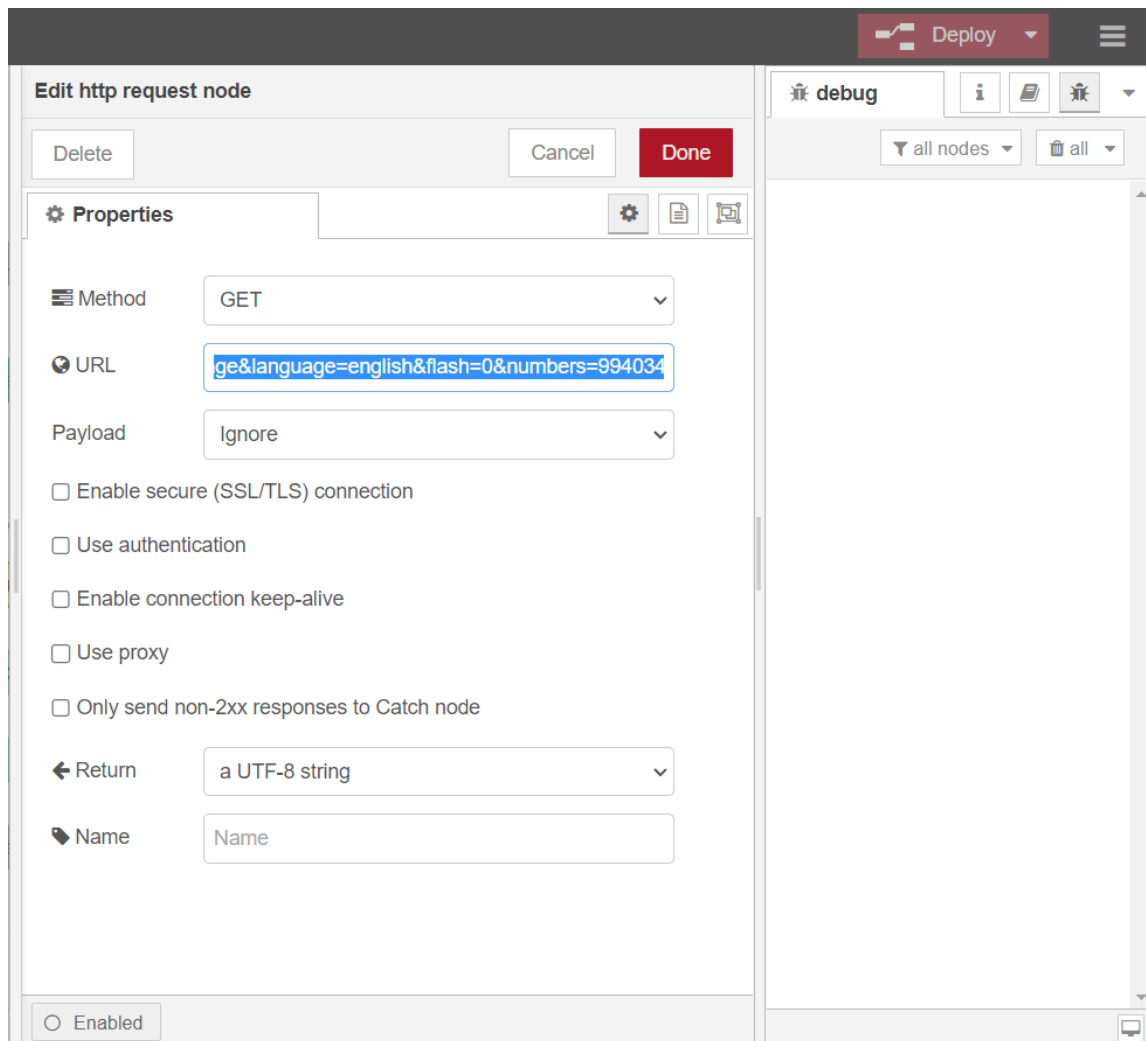
    #latitude 17.4219272
    #longitude 78.5488783
    myData={'name': name, 'lat':latitude, 'lon': longitude}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Data published to IBM IoT platform: ",myData)
    time.sleep(5)

client.disconnect()
```

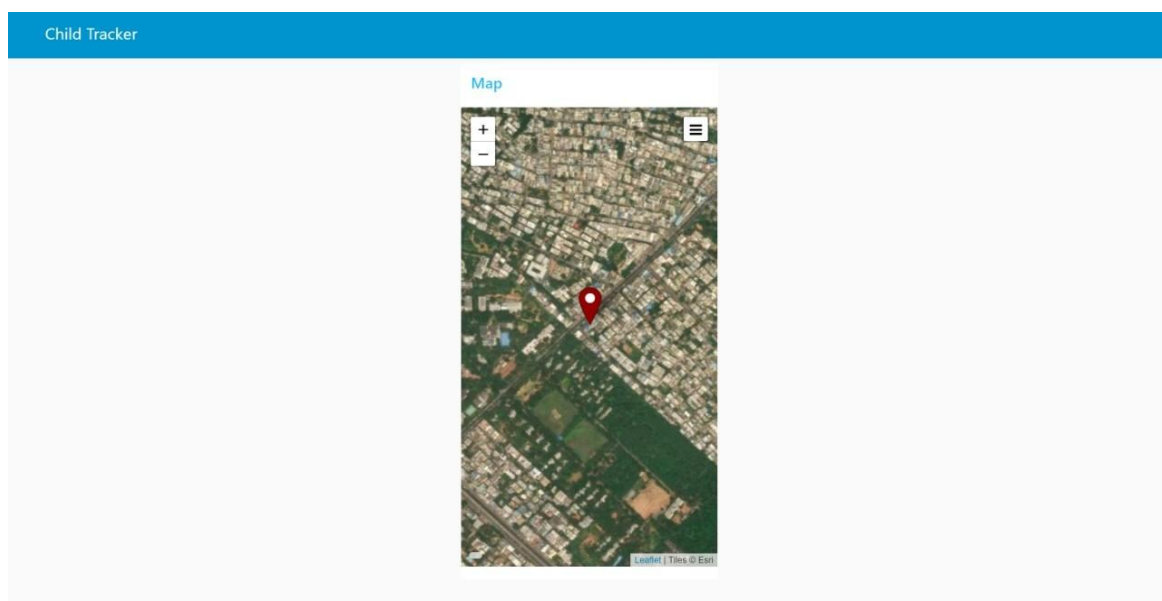
- Created the GeoFence



- Editing the HTTP Request URL



- Located the child



- Created the geofence node

Edit geofence node

Delete Cancel Done

Properties

St. Thomas Cantonment, Adyar, Besant Nagar, Thiruvannamiyur, Kottivakkam, Palavakkam, Neelankarai, SH109, SH49A, Kandanachavadi, Perungudi, Taramani, Indian Institute of Technology Madras, Guindy National Park, Adambakkam, Ullagaram, Puzhuthivakkam, Madipakkam, Talai, Annambur, Leaflet | Map data © OpenStreetMap contributors

_ Floor: ground Ceiling: infinity

Action: add "inarea" property

Enable output of zones to WorldMap node: ☐

☐ Enabled

debug

all nodes all

msg.payload: Object

```
{
  name: "smartbridge",
  lat: 17.4219272,
  lon: 78.5400782
}
```

11/11/2022, 12:45:24 PM node: f2f2649a.0d0d98
 iot-2/type/TestDeviceType/id/12345/evt/status/fmt/json :
 msg.payload: Object

```
{
  name: "Smartbridge",
  lat: 17.4225176,
  lon: 78.5456842
}
```

11/11/2022, 12:45:26 PM node: f2f2649a.0d0d98
 iot-2/type/TestDeviceType/id/12345/evt/event_1/fmt/json :
 msg.payload: Object

```
{
  name: "smartbridge",
  lat: 17.4219272,
  lon: 78.5400782
}
```

11/11/2022, 12:45:29 PM node: f2f2649a.0d0d98
 iot-2/type/TestDeviceType/id/12345/evt/event_1/fmt/json :
 msg.payload: Object

```
{
  name: "smartbridge",
  lat: 17.4219272,
  lon: 78.5400782
}
```

11/11/2022, 12:45:29 PM node: f2f2649a.0d0d98
 iot-2/type/TestDeviceType/id/12345/evt/status/fmt/json :
 msg.payload: Object

```
{
  name: "Smartbridge",
  lat: 17.4225176,
  lon: 78.5456842
}
```

11/11/2022, 12:45:32 PM node: f2f2649a.0d0d98
 iot-2/type/TestDeviceType/id/12345/evt/event_1/fmt/json :
 msg.payload: Object

```
{
  name: "smartbridge",
  lat: 17.4219272,
  lon: 78.5400782
}
```

- Python script sending requests to IBM Cloud

```
child.py - C:\Users\Anu\AppData\Local\Programs\Python\Python37\child.py (3.7.0)
File Edit Format Run Options Window Help
import json
import wiotp.sdk.device
import time
myConfig = {
    "identity": {
        "orgId": "401qab",
        "typeId": "TestDeviceType",
        "deviceId": "12345"
    },
    "auth": {
        "token": "pnhXvzN-sWkRvshxyl"
    }
}
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

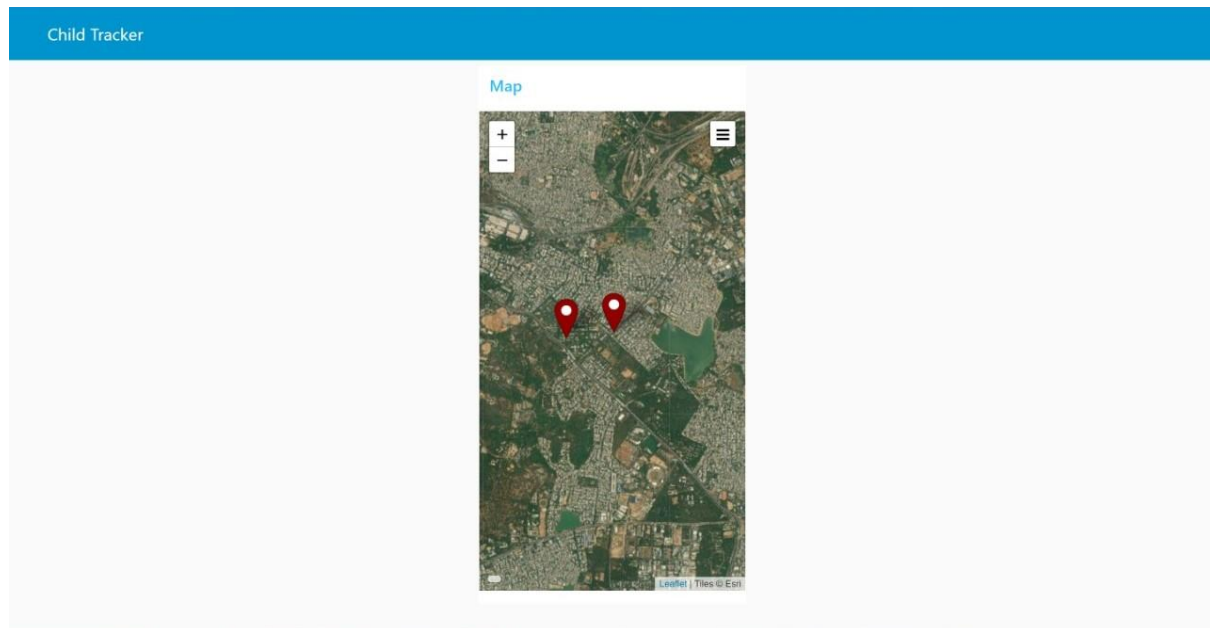
while True:
    name = "Smartbridge"
    #in area location
    #latitude = 17.4225176
    #longitude = 78.5456842

    #out area location
    latitude = 17.4219272
    longitude = 78.5400782
    myData = {'name': name, 'lat': latitude, 'lon': longitude}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Data published to IBM IoT platform: ", myData)
    time.sleep(5)

client.disconnect()
```

Ln: 28 Col: 18

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



Result: Successfully developed the web application using Node-RED