

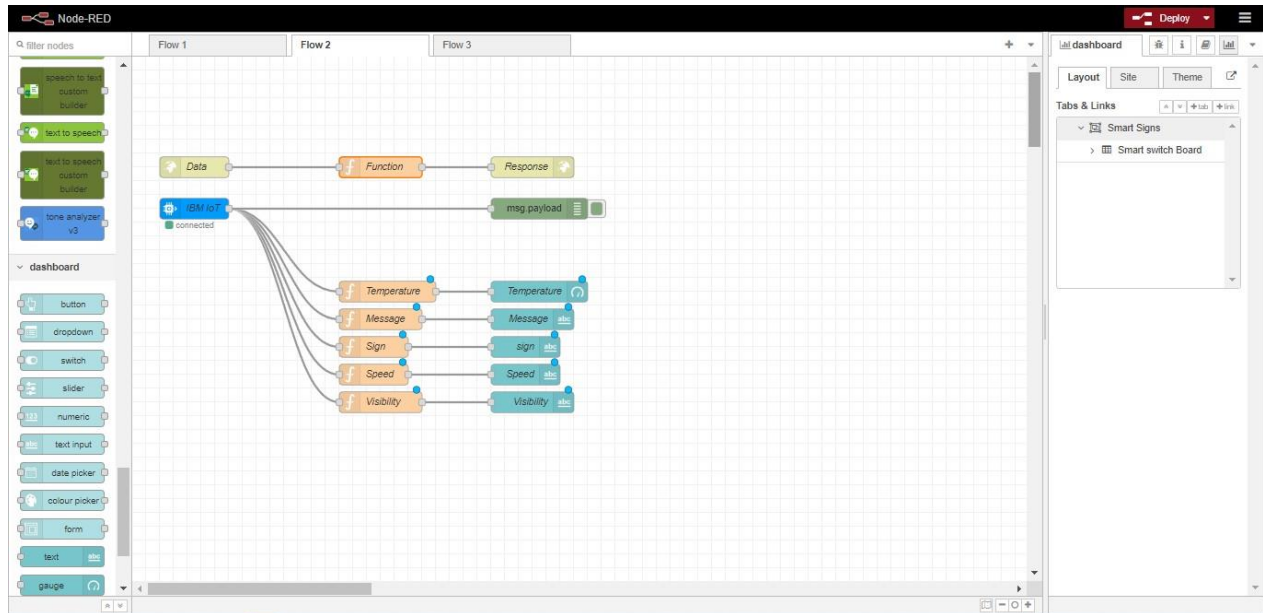
## PROJECT DEVELOPMENT PHASE

Sprint - 4

DATE	16/11/2022
TEAM ID	PNT2022TMID29476
PROJECTNAME	SIGNS WITH SMART CONNECTIVITY FOR BETTER ROADSAFETY

## DEVELOPIN GROUTE BASED ON THE PROGRAM :

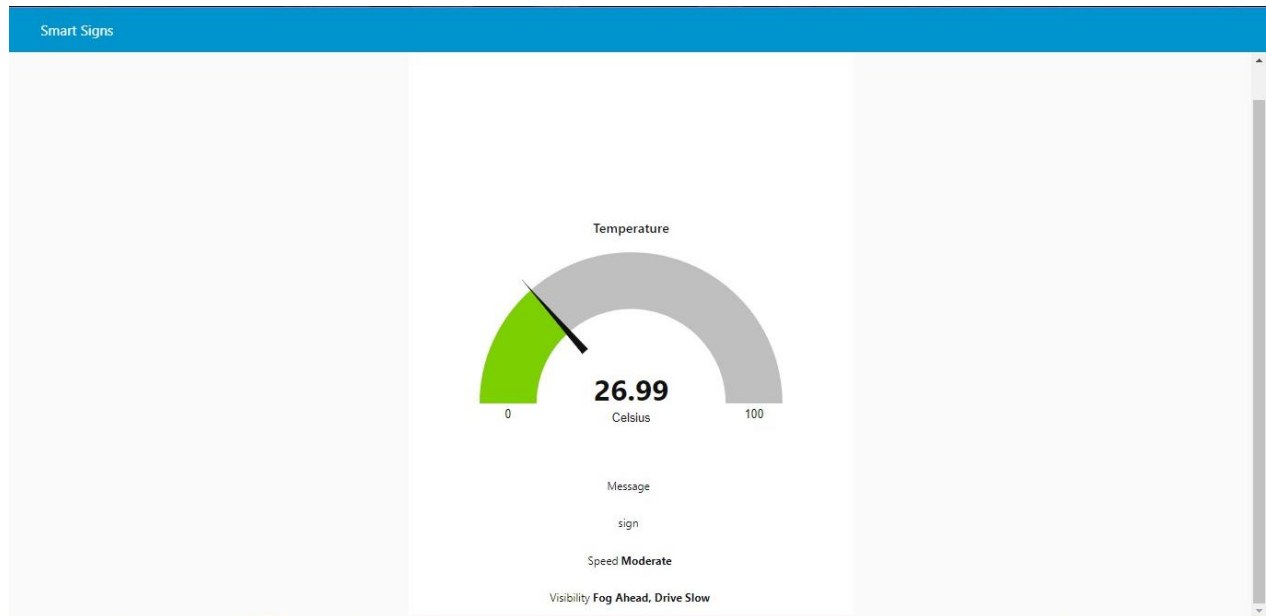
Here based on the project the routing is developed by using appropriate nodes.



## OUTPUT FOR ODERED :

After making the proper connection between nodes the deploy button is enabled and the result is displayed on the node-reddashboard.

Its hows theres ultina diagrammatic structure :



CODE IN PYTHON IDLE:

PROGRAM :

```
randomSensorData.py - C:\Users\paul\OneDrive\Desktop\randomSensorData.py (3.7.0)
File Edit Format Run Options Window Help

import wiotp.sdk.device
import time
import random
import ibmiotf.application
import ibmiotf.device
import requests, json

myConfig = {
    #Configuration
    "identity": {
        "orgId": "xfxck9s",
        "typeId": "NodeMCU",
        "deviceId": "6385476398"
    },
    #API Key
    "auth": {
        "token": "9384731286"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    mcmd.data['command']

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

#OpenWeatherMap Credentials
BASE_URL = "https://api.openweathermap.org/data/2.5/weather?"
CITY = "Nagercoil"
URL = BASE_URL + "q=" + "chennai" + "&appid=" + "01df65417ab3968e3fc2a38c4aee27bb"

while True:
    response = requests.get(URL)
    if response.status_code == 200:
        data = response.json()
        main = data['main']
        temperature = main['temp']
        humidity = main['humidity']
        pressure = main['pressure']
        report = data['visibility']

    #message part
    msg=random.randint(0,5)
```

```
randomSensorData.py - C:\Users\paul\OneDrive\Desktop\randomSensorData.py (3.7.0)
File Edit Format Run Options Window Help

msg=random.randint(0,5)
if msg==1:
    message="SLOW DOWN , SCHOOL IS NEAR"
elif msg==3:
    message="SLOW DOWN , HOSPITAL NEARBY"
elif msg==4:
    message="NEED HELP, POLICE STATION NEARBY"
else:
    message=""

#Speed part
speed=random.randint(0,150)
if speed>100:
    speedMsg="SLOW DOWN , Speed Limit Exceeded"
elif speed>=60 and speed<100:
    speedMsg="Moderate Speed"
else:
    speedMsg=""

#Sign part
sign=random.randint(0,5)
if sign==1:
    signMsg="Right Diversion ->"
elif sign==3:
    signMsg="Left Diversion <-"
elif sign==5:
    signMsg="U Turn"
else:
    signMsg=""

#Visibility
if temperature<=50:
    visibility="Fog Ahead, Drive Slow"
else:
    visibility="Clear Weather"

else:
    print("Error in the HTTP request")
myData={'Temperature':temperature, 'Message':message, 'Sign':signMsg, 'Speed':speedMsg, 'Visibility':visibility}
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
print("Published data Successfully: %s" % myData)
client.commandCallback = myCommandCallback
time.sleep(5)
client.disconnect()
```

**Program used in the code:**

```
import wiotp.sdk.device
importrtti
```

```

import random
import ibmiotf.application
import ibmiotf.device
import requests, json

myConfig = {
    "#Configuration": {
        "identity": {
            "orgId": "xfxok9",
            "typeId": "NodeMCU",
            "deviceId": "6385476358"
        },
        "#APIKey": {
            "auth": {
                "token": "9384731286"
            }
        }
    }
}

```

```

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m = cmd.data['command']

```

```

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

```

```

# OpenWeatherMap
Credentials_BASE_URL = "https://api.openweathermap.org/data/2.5/weather?"

```

CITY="Chennai"

URL=BASE\_URL+"q="+ "chennai"+"&appid="+ "01df65417ab3968e3fc2a38c4aee27bb"

while True:

    response=requests.get(URL)

    if response.status\_code==200: data=r

        esponse.json()

        main = data['main']temperature

        = main['temp']humidity =

        main['humidity']pressure=main[

        'pressure']report=data['visibility'

        ]

    #messge

    partmsg=random.randint(0,5)i

    fmsg==1:

        message="SLOWDOWN,SCHOOLISNEAR"

    elifmsg==3:

        message="SLOWDOWN,HOSPITALNEARBY"

    elifmsg==5:

        message="NEEDHELP,POLICESTATIONNEARBY"

    else:

        message=""

    #Speed

    partspeed=random.randint(0,150)if

    speed>=100:

        speedMsg="SLOWDOWN,SpeedLimitExceeded"elifspeed>=6

    0andspeed<100:

```

        speedMsg="Moderate
Speed"else:
        speedMsg=""

    #Sign
    partsign=random.randint(0,5)i
    fsign==1:
        signMsg="Right Diversion -
>"elifsign==3:
        signMsg="Left Diversion <-
"elifsign==5:
        signmsg="UTurn"els
    e:
        signMsg=""

    #Visibility
    iftemperature<=50:
        visibility="FogAhead,DriveSlow"else:
        visibility="ClearWeather"

else:
    print("Error in the HTTP request")myData={'Temperature':temperature,
'Message':message,
'Sign':signMsg, 'Speed':speedMsg,
'Visibility':visibility}client.publishEvent(eventId="status",msgFormat="json",
data=myData, qos=0, onPublish=None)print("Published
data Successfully: %s",
myData)client.commandCallback=myCommandCallbac
ktime.sleep(5)

```

The output of the code was displayed in python idle shell mode.

10

---