

PROJECT DEVELOPMENT PHASE


SPRINT-1

| | |
|--------------|--|
| Date | 12 November 2022 |
| Team ID | PNT2022TMID00923 |
| Project Name | Signs with Smart Connectivity for Better Road Safety |

SPRINT-1:

- In sprint-1, we collect temperature and humidity data from OpenWeatherMap website for a particular city as input.
- We have developed the code for the same and to publish it to IBM IoT Watson, node-red and finally display them in MIT APP INVENTOR.
- We have also implemented a condition such that, if humidity is less than 100, then it displays a warning like "PLEASE SLOW DOWN".

PYTHON CODE:

 *PROJECTFINALDND.py - D:/1ibm/PROJECTFINALDND.py (3.7.0)*

File Edit Format Run Options Window Help

```
#connect and send a datapoint "temp" with value integer value into the cloud as a type of event for every 10 seconds
deviceCli.connect()

while True:

    #get sensor data from DHT11

    a = "https://api.openweathermap.org/data/2.5/weather?q=Chennai,%20IN&appid=e2bea247ed9ad643a04d9a8e55499d5f"
    r=requests.get(url=a)
    data=r.json()

    Temp= data['main']['temp']
    Humd= data['main']['humidity']
    data= {'temp':Temp,'humid':Humd}
    dist=random.randint(0,50)
    dis={'dista':dist}

    if (Humd<100):
        warn={'alert':'PLEASE SLOW DOWN!!!!!!'}
```

IBM IoT WATSON PLATFORM:

The screenshot shows the IBM Watson IoT Platform dashboard. The 'Recent Events' tab is selected, displaying a table of events. A red box highlights three specific events:

| Event | Value | Format | Last Received |
|-----------|-------------------------------------|--------|-------------------|
| IoTSensor | {"dista":1} | json | a few seconds ago |
| IoTSensor | {"alert":"PLEASE SLOW DOWN!!!!!!!"} | json | a few seconds ago |
| IoTSensor | {"temp":300.14,"humid":94} | json | a few seconds ago |
| IoTSensor | {"inst":"stop"} | json | a few seconds ago |
| IoTSensor | {"dista":0} | json | a few seconds ago |

Below the table, it indicates '1 Simulation running'.

NODE-RED:

The screenshot shows the Node-RED interface. The flow 'Flow 1' is active, featuring an 'IBM IoT' node connected to several function nodes for processing temperature, humidity, and alert data. The debug console on the right displays the message payloads, with a red box highlighting the following JSON objects:

```
{ "temp": 300.14, "humid": 94 }
```

```
{ "alert": "PLEASE SLOW DOWN!!!!!!!" }
```

MIT APP INVENTOR:

The screenshot displays the MIT App Inventor web interface in a browser. The address bar shows the URL `ai2.appinventor.mit.edu/#5724286015635456`. The page header includes the MIT App Inventor logo and navigation links: Projects, Connect, Build, Settings, Help, My Projects, View Trash, Guide, Report an Issue, English, and a user email `mail2afreen.j@gmail.com`.

The main workspace is divided into two panels: **Blocks** on the left and **Viewer** on the right. The **Blocks** panel shows a categorized list of components: Built-in (Control, Logic, Math, Text, Lists, Dictionaries, Colors, Variables, Procedures), Screen2, Label1, HorizontalArrangemen, Label2, TextBox1, HorizontalArrangemen, and Label3. The **Viewer** panel displays a visual programming workflow for a web application.

The workflow in the **Viewer** panel is as follows:

- when Web1 . GotText** (Trigger)
- do** (Loop)
- set TextBox1 . Text** to **look up in pairs** key **temp** (Block)
- look up in pairs** pairs **call Web1 . JsonTextDecodeWithDictionaries** **jsonText** **get responseContent** (Block)
- notFound** **not found** (Block)
- set TextBox2 . Text** to **look up in pairs** key **humid** (Block)
- look up in pairs** pairs **call Web1 . JsonTextDecodeWithDictionaries** **jsonText** **get responseContent** (Block)
- notFound** **not found** (Block)
- set TextBox5 . Text** to **look up in pairs** key **alert** (Block)
- look up in pairs** pairs **call Web1 . JsonTextDecodeWithDictionaries** **jsonText** **get responseContent** (Block)
- notFound** **not found** (Block)
- Show Warnings** (Block)

The interface also features a **Designer** tab and a **Blocks** tab. The bottom status bar shows the system clock at 12:57 on 13-11-2022, along with weather information (26°C Cloudy) and various system icons.

USER APP SCREENSHOT:

