

| | |
|--------------|---|
| DATE | 16 TH November 2022 |
| TEAM ID | PNT2022TMID36126 |
| PROJECT NAME | GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES |

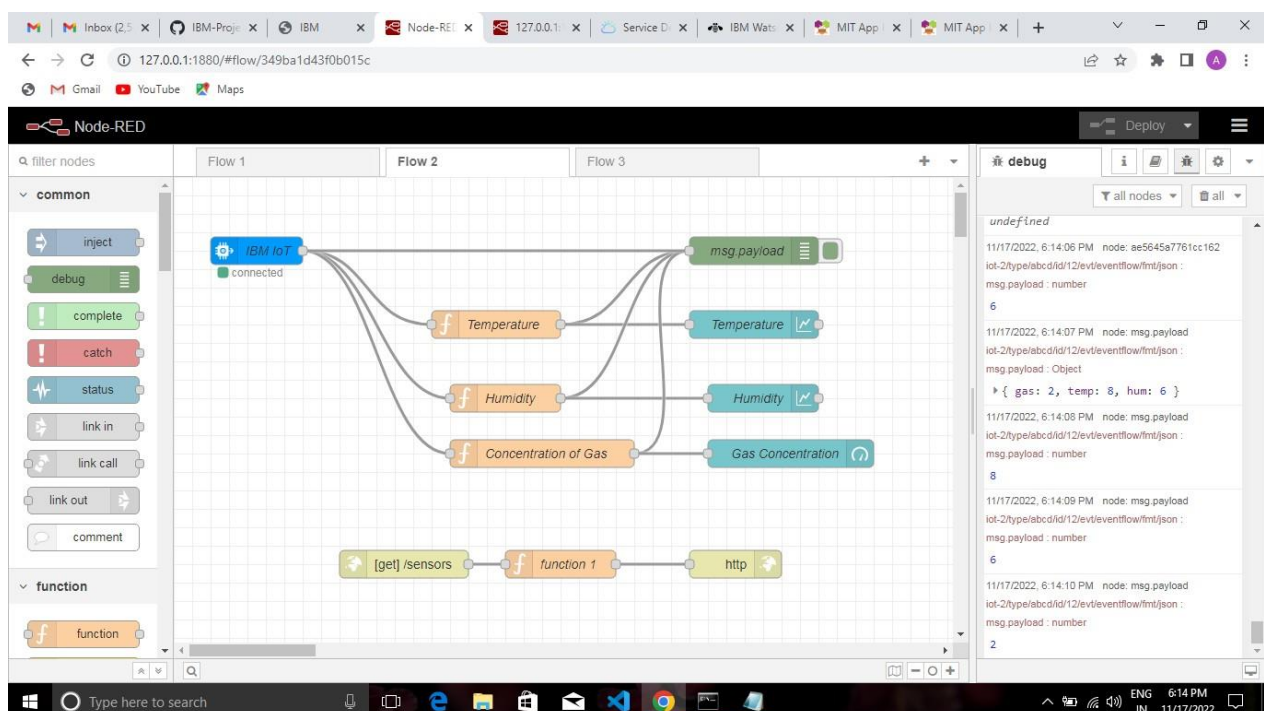
Task:

To receive the data from Node-RED to the Mobile App developed Using MIT App Inventor.

Procedure:

1. In Node-RED **http** node is added to post the data gathered from IBM Watson IoT Platform to a webpage.
2. Screen – 3 is developed to gather data from the website where the data is published by Node-RED.
3. In case of abnormal environmental conditions. An alert message is generated by MIT App.

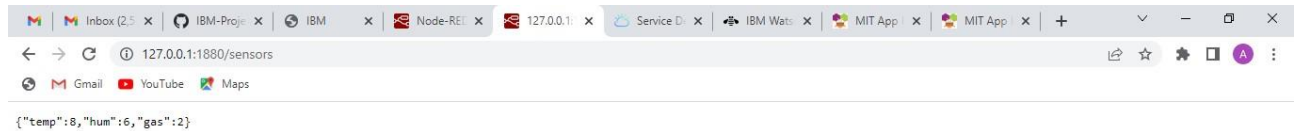
1. Node-RED:



Function1 in node-red:

This function is responsible to post the sensor data to the webpage from where the data is gathered by MIT App.

2. Web page:



3. MIT App:

Blocks for screen 3 has been developed.

Blocks are capable of generating notification when,

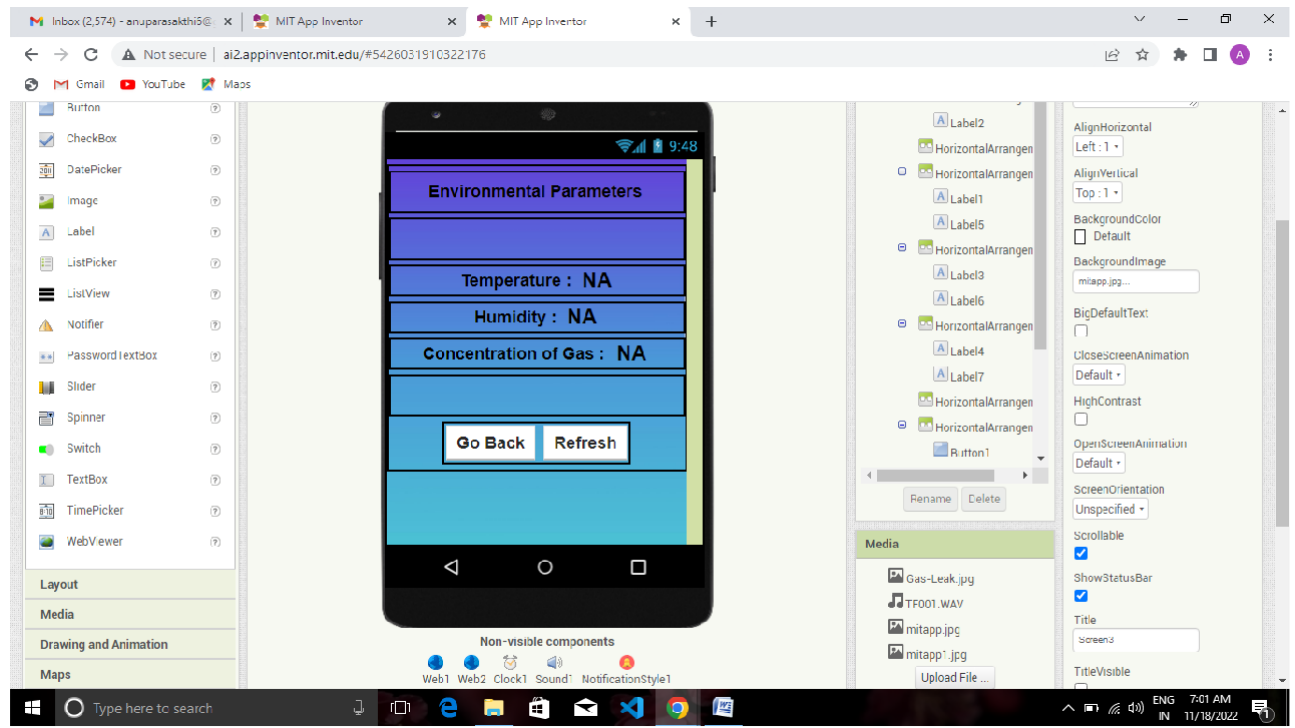
Temperature is greater than 40°C

Humidity is greater than 30% Concentration of gas is

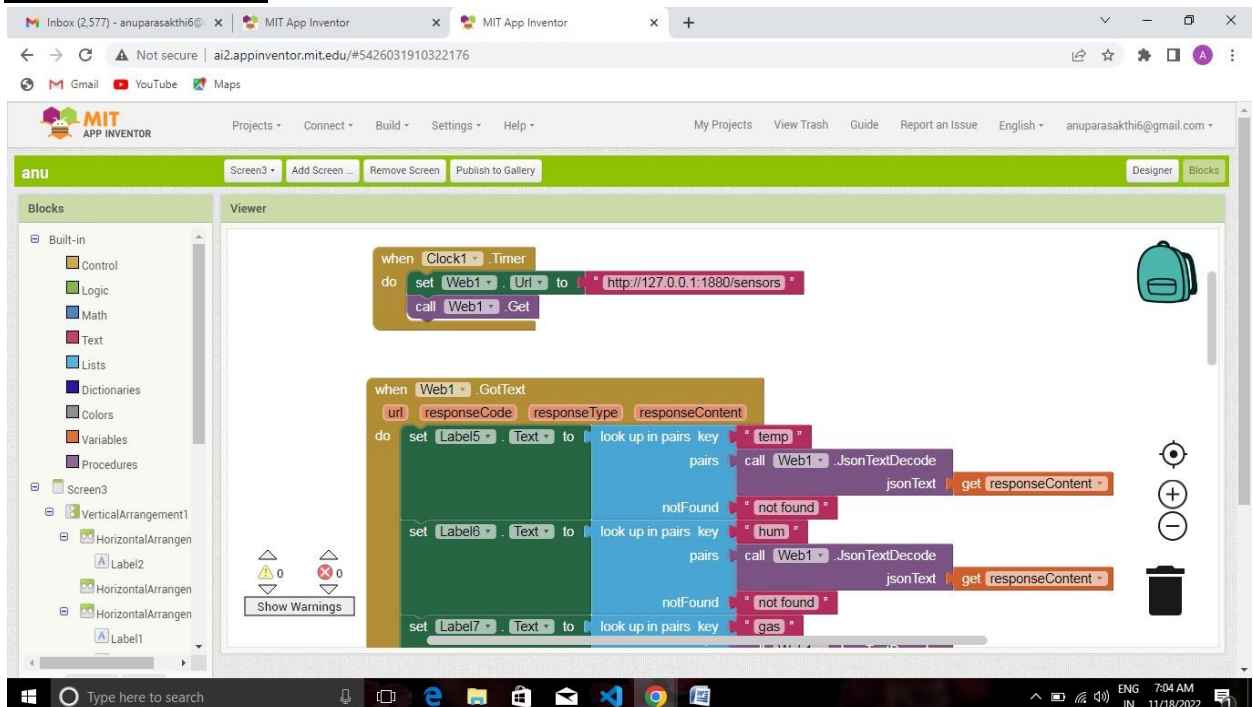
greater than 50%

Not only notification, this also rings an alarm whenever abnormal

Screen3:



Screen3 Blocks:



Screen 3 (Blocks):

The screenshot shows the MIT App Inventor web interface for a project named 'anu'. The 'Blocks' palette on the left lists various categories: Built-in, Control, Logic, Math, Text, Lists, Dictionaries, Colors, Variables, and Procedures. Under 'Screen3', there are three 'HorizontalArrangement' blocks containing 'Label2', 'Label1', and 'Label1' respectively. The 'Viewer' pane displays the following code blocks:

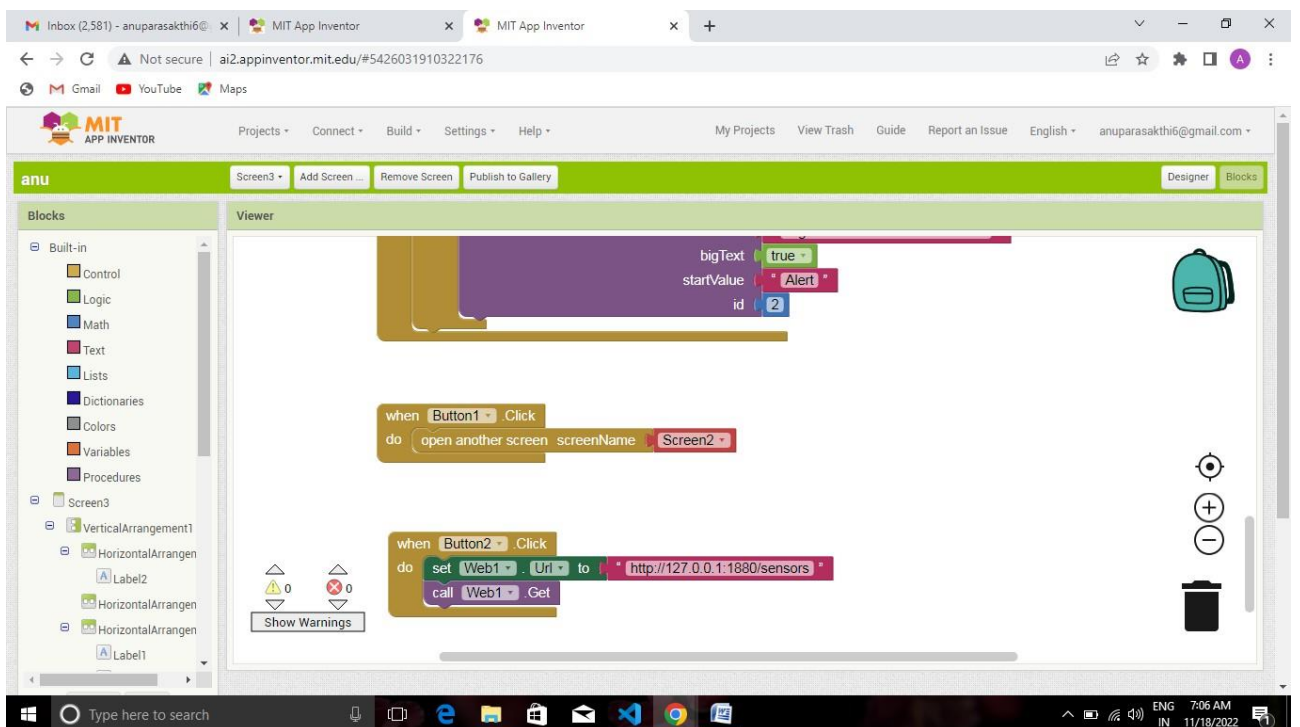
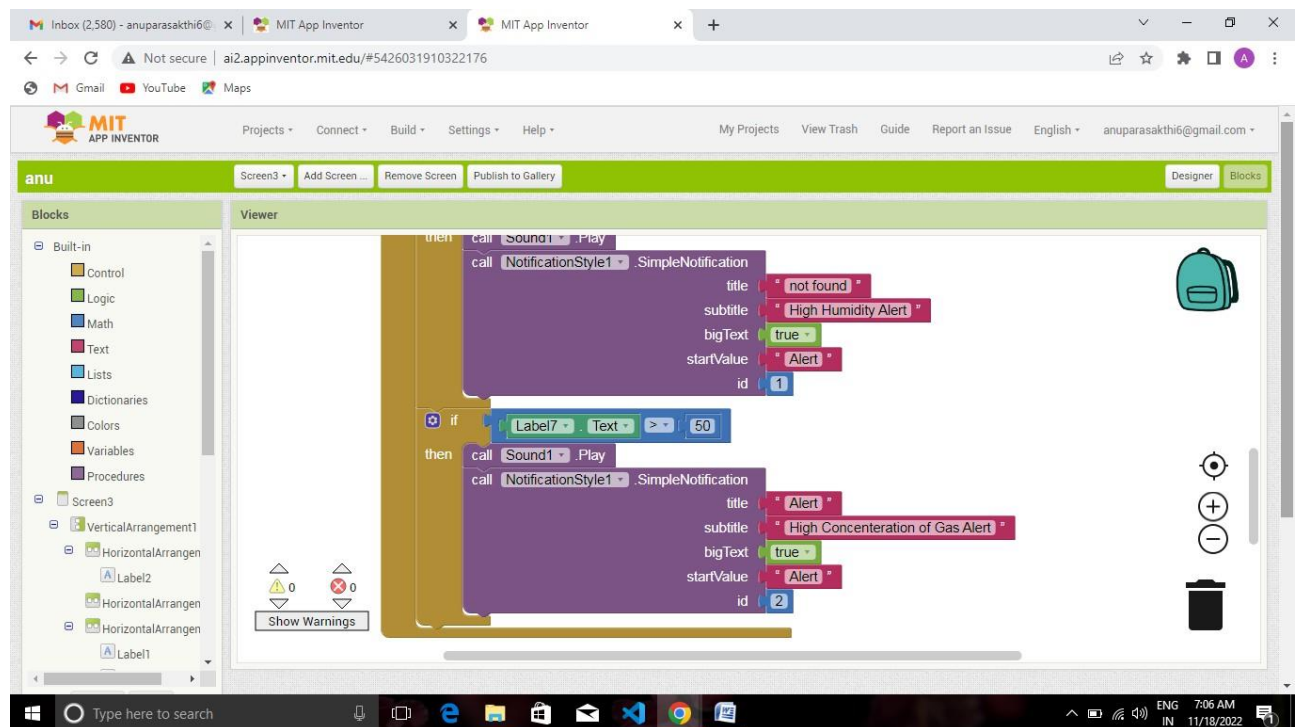
- when Web1 GotText** (event handler)
 - do** block containing:
 - set Label5 Text to** (look up in pairs key: "temp", pairs: call Web1.JsonTextDecode, jsonText: get responseContent)
 - set Label6 Text to** (look up in pairs key: "hum", pairs: call Web1.JsonTextDecode, jsonText: get responseContent)
 - set Label7 Text to** (look up in pairs key: "gas", pairs: call Web1.JsonTextDecode, jsonText: get responseContent)
 - if** (Label5 Text > 40) then:
 - call Sound1.Play
 - call NotificationStyle1.SimpleNotification

The interface also shows a 'Show Warnings' button and a 'Designer' tab.

The screenshot shows the MIT App Inventor web interface for the same project 'anu'. The 'Blocks' palette is the same. The 'Viewer' pane displays the following code blocks:

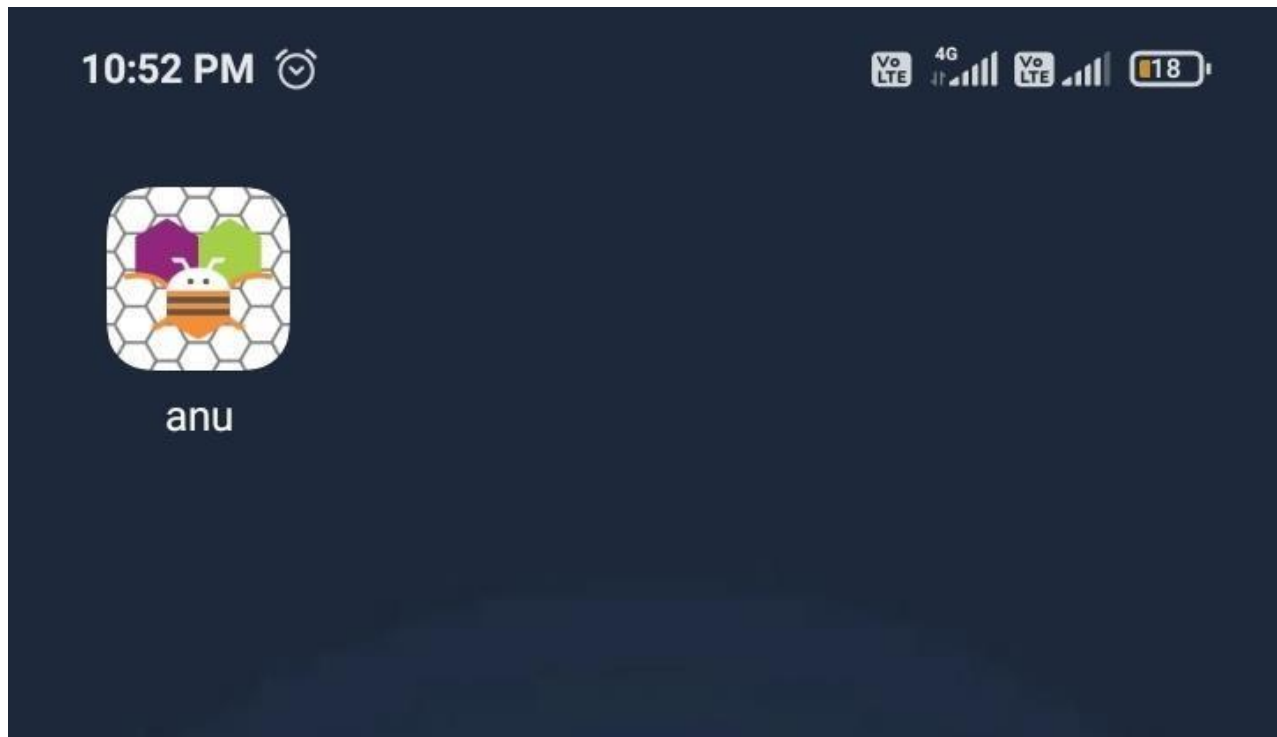
- if** (Label5 Text > 40) then:
 - call Sound1.Play
 - call NotificationStyle1.SimpleNotification (title: "Alert", subtitle: "High Temperature Alert", bigText: true, startValue: 1, id: 0)
- if** (Label6 Text > 30) then:
 - call Sound1.Play
 - call NotificationStyle1.SimpleNotification (title: "not found", subtitle: "High Humidity Alert", bigText: true, startValue: "Alert", id: 1)

The interface also shows a 'Show Warnings' button and a 'Designer' tab.



These above written blocks are responsible for bringing the data to the mobile app and to generate alarms incase of emergency.

Application in Mobile:



Screen3 in Mobile:

