

SPRINT 2 : Creating a successful Web application based on Node-Red to gather data from IoT device.

Team ID	PNT2022TMID51059
Project Title	Gas Leakage Monitoring & Alerting System For Industries
Date	07 Nov,2022

The screenshot displays the Node-RED web interface in a browser. The main workspace shows a flow named 'Flow 1' with the following components:

- IBM IoT Node:** A blue node labeled 'connected' that triggers the flow.
- Function Nodes:** Four orange nodes labeled 'Humidity', 'Hazardous_gas', 'Temperature', and 'Pressure' are connected to the IBM IoT node.
- Output Nodes:** Four teal nodes labeled 'Humidity_Value', 'Hazardous_gas', 'Temperature', and 'Pressure' are connected to their respective function nodes.
- msg.payload Node:** A green node that receives data from all four output nodes.
- HTTP Node:** A yellow node labeled 'http' is connected to the 'msg.payload' node.
- Value Node:** An orange node labeled 'Value' is connected to the 'http' node.
- GET Node:** A yellow node labeled '[get] /data' is connected to the 'Value' node.

The right-hand side of the interface shows the 'debug' console with a list of messages. The messages are JSON objects representing IoT data points, including 'Humidity', 'Temperature', 'Pressure', and 'Hazardous_gas'.

```
11/19/2022, 9:45:44 AM node: d8b73c2da06569ae
iot-2/type/GOT/id/GOT_1/evt/event_1/fmt/json : msg.payload :
number
72

11/19/2022, 9:45:44 AM node: d8b73c2da06569ae
iot-2/type/GOT/id/GOT_1/evt/event_1/fmt/json : msg.payload :
number
27

11/19/2022, 9:45:44 AM node: d8b73c2da06569ae
iot-2/type/GOT/id/GOT_1/evt/event_1/fmt/json : msg.payload :
number
92

11/19/2022, 9:45:44 AM node: d8b73c2da06569ae
iot-2/type/GOT/id/GOT_1/evt/event_1/fmt/json : msg.payload :
number
73

11/19/2022, 9:45:44 AM node: d8b73c2da06569ae
iot-2/type/GOT/id/GOT_1/evt/event_1/fmt/json : msg.payload :
Object
{ Hazardousgas: 41, Temperature: 60, Humidity: 14, Pressure: 75 }

11/19/2022, 9:45:44 AM node: d8b73c2da06569ae
iot-2/type/GOT/id/GOT_1/evt/event_1/fmt/json : msg.payload :
number
```

Device Type: GOT

Events 1

Device actions

New event type +

Event type name

event_1

Send

Schedule

7

Every Minute

Payload

Specify the event payload in the editor window or by uploading a [CSV file](#).

0 {

1 "Hazardousgas": random(0,100),

2 "Temperature" : random(0,100),

3 "Humidity" : random(0,100),

4 "Pressure" : random(0,100)

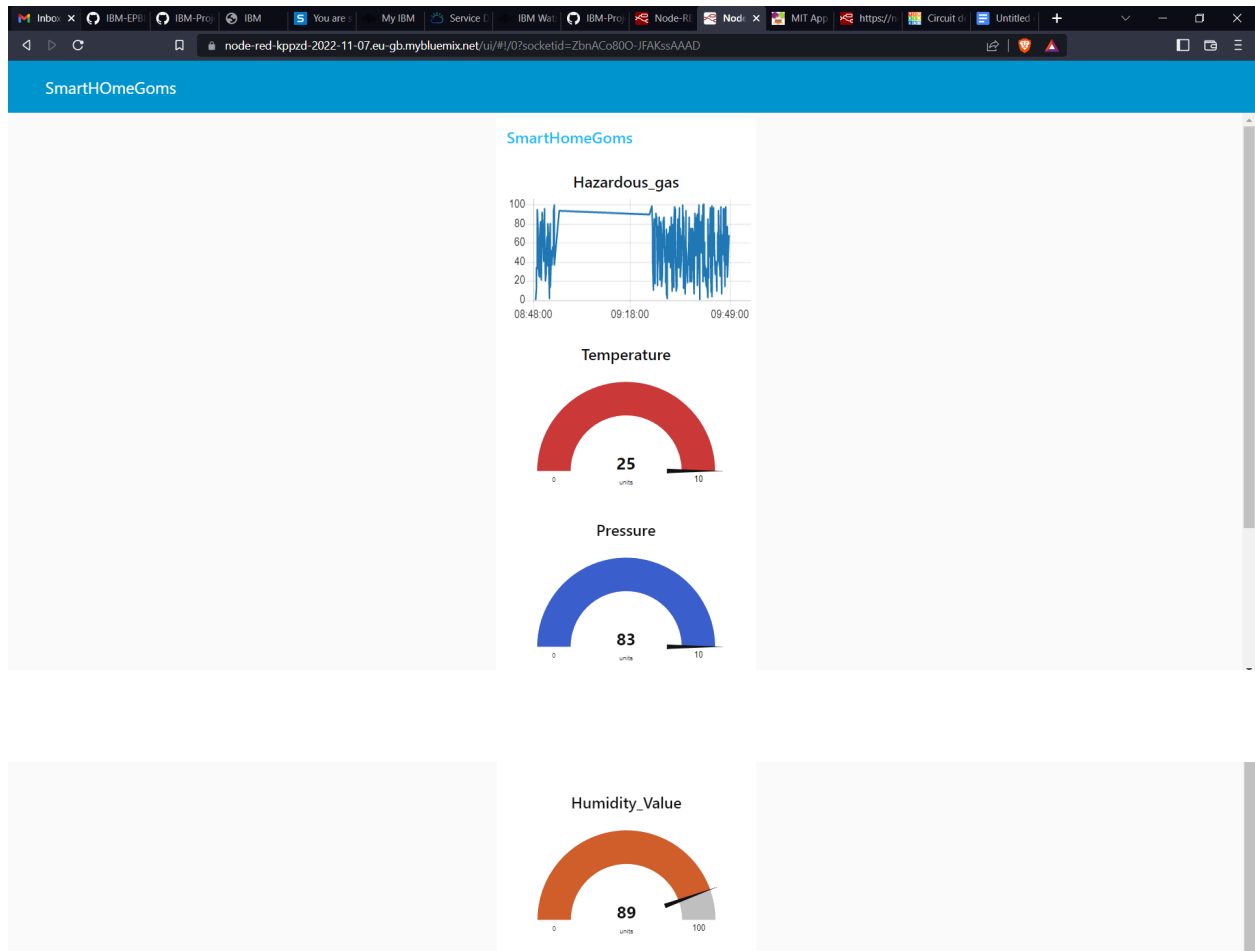
5 }

Upload a CSV file

Cancel

Save

The source code is given above in the image .



This is the expected output.

The data has been gathered and displayed successfully.