

# DEVELOP A WEB APPLICATION USING NODE-RED SERVICE

## USE DASHBOARD NODES FOR CREATING UI (WEB APP)

<b>Date</b>	15 November 2022
<b>Team ID</b>	PNT2022TMID07171
<b>Project Name</b>	Smart Solutions for Railways
<b>Maximum Marks</b>	8 Marks

## Node RED Design

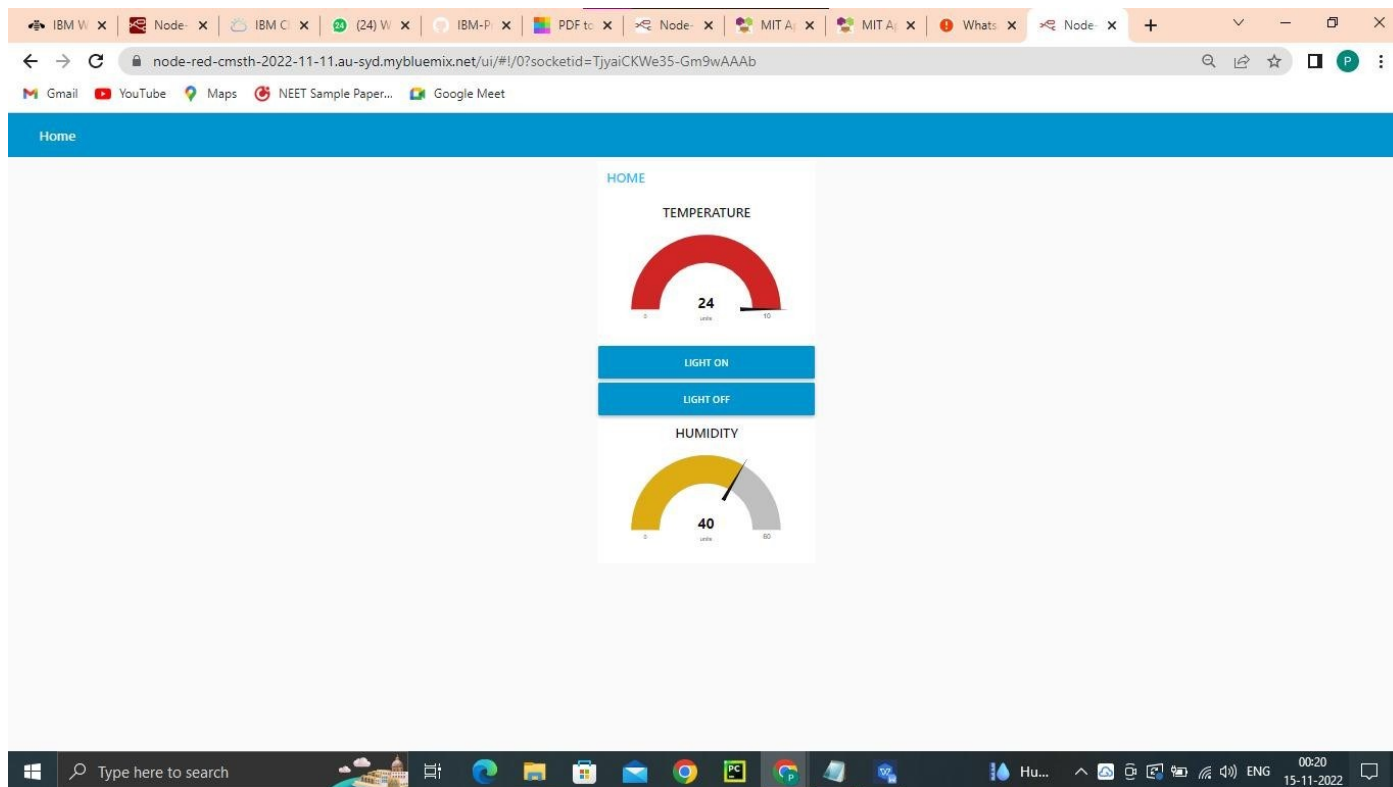
The screenshot displays the Node-RED web interface in a browser. The address bar shows the URL: `node-red-cmsth-2022-11-11.au-syd.mybluemix.net/red/#flow/e0f03146a28ca8d7`. The interface includes a left sidebar with node categories (common, function, etc.), a central workspace with two flows, and a right sidebar with a debug console.

**Flow 1:** This flow starts with an inject node connected to a function node. The function node has two outputs: one to a 'Humid' node and another to a 'temp' node. Both 'Humid' and 'temp' nodes are connected to a 'msg payload' node, which then connects to 'HUMIDITY' and 'TEMPERATURE' dashboard nodes respectively.

**Flow 2:** This flow starts with a 'LIGHT ON' and 'LIGHT OFF' switch. Both switches are connected to a function node. The function node has two outputs: one to a 'msg payload' node and another to a 'command' node. The 'command' node is connected to an 'http' node. There is also a 'GET /data' node connected to a 'function' node, which is connected to an 'http' node.

**Debug Console:** The debug console on the right shows a series of messages. It includes timestamps, node IDs, and payloads. The payloads are strings like 'lighton', 'lightoff', and 'lighton'. The messages are grouped by node, showing the flow of data through the system.

# Node RED WEBPAGE



## MIT APP TO GET DATA AND TURN ON /OFF LIGHT

