

Team ID: PNT2022TMID50102

Date:17/11/2022

Configuration of Node-Red to send commands to IBM cloud

ibmiot out node I used to send data from Node-Red to IBM Watson device. So, after adding it to the flow we need to configure it with credentials of our Watson device.

The screenshot shows the 'Edit ibmiot in node' configuration window in Node-RED. The window has a title bar with the Node-RED logo and a tab labeled 'Node-RED: node-red-hdyfv-20...'. Below the title bar is a URL bar showing 'ed/#flow/c7ddb1462b8a000c'. The main content area is titled 'Edit ibmiot in node' and contains several configuration fields. At the top right of the main area are three buttons: 'Delete', 'Cancel', and 'Done'. Below these is a 'Properties' section with a gear icon and a search icon. The configuration fields are as follows:

- Authentication:** A dropdown menu set to 'API Key'.
- API Key:** A text input field containing 'IBMIOT APIKEY' with a search icon to its left and a pencil icon to its right.
- Input Type:** A dropdown menu set to 'Device Event'.
- Device Type:** A text input field with a checkbox labeled 'All or' and the value 'abcd'.
- Device Id:** A text input field with a checkbox labeled 'All or' and the value '7654321'.
- Event:** A text input field with a checkbox labeled 'All or' and the value ' '.
- Format:** A text input field with a checkbox labeled 'All or' and the value 'json'.
- QoS:** A dropdown menu set to '0'.
- Name:** A text input field containing 'IBM IoT'.

At the bottom left of the window is a checkbox labeled 'Enabled' which is currently checked.

Here we add two buttons in UI

1 -> for motor on

2 -> for motor off

We used a function node to analyse the data received and assign command to each number.

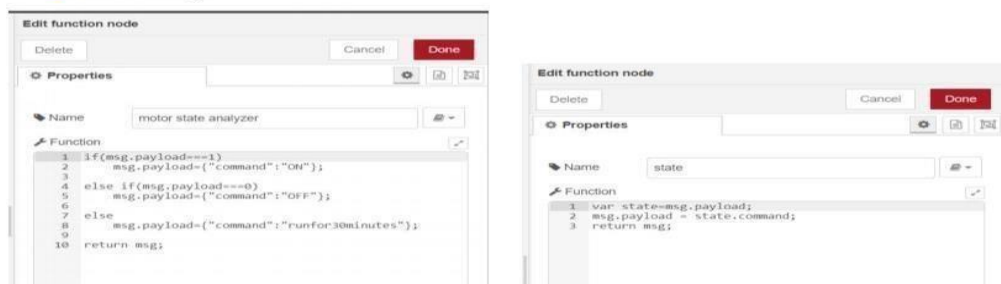
The Java script code for the analysis is:

```
if(msg.payload===1)
msg.payload={"command": "ON"};
else if(msg.payload===0)
msg.payload={"command": "OFF"};
```

Then we use another function node to parse the data and get the command and represent it visually with text node.

The Java script code for that function node is:

```
var state=msg.payload;
msg.payload = state.command;
return msg;
```

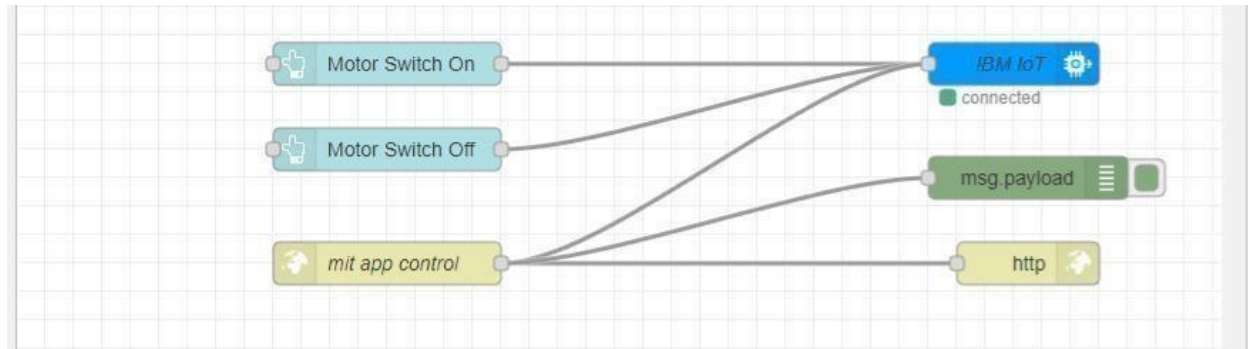


The above images show the java script codes of analyser and state function nodes.

Then we add edit json node to the conversion between JSON string & object and finally connect it to IBM IoT Out.



Edit JSON node needs to be configured like this



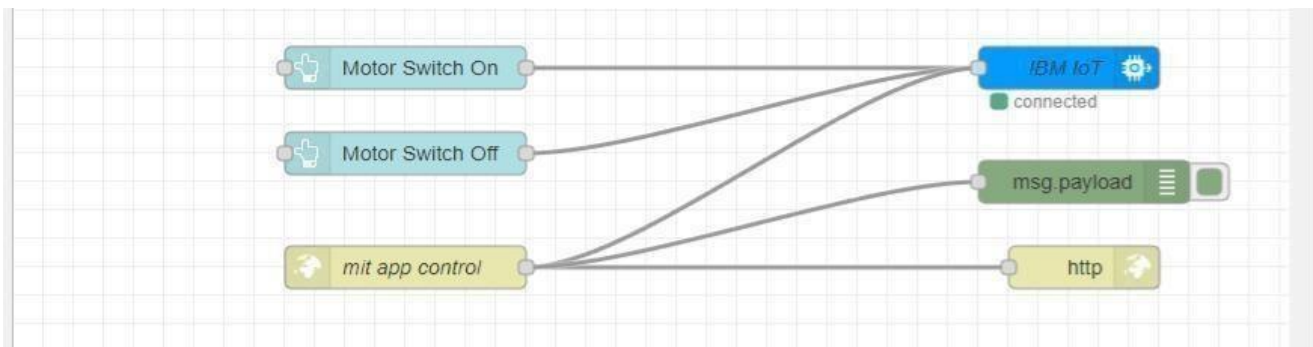
This is the program flow for sending commands to IBM cloud.

Adjusting User Interface

In order to display the parsed JSON data a Node-Red dashboard is created

Here we are using Gauges, text and button nodes to display in the UI and helps to monitor the parameters and control the farm equipment.

Below images are the Gauge, text and button node configurations.



Complete Program Flow

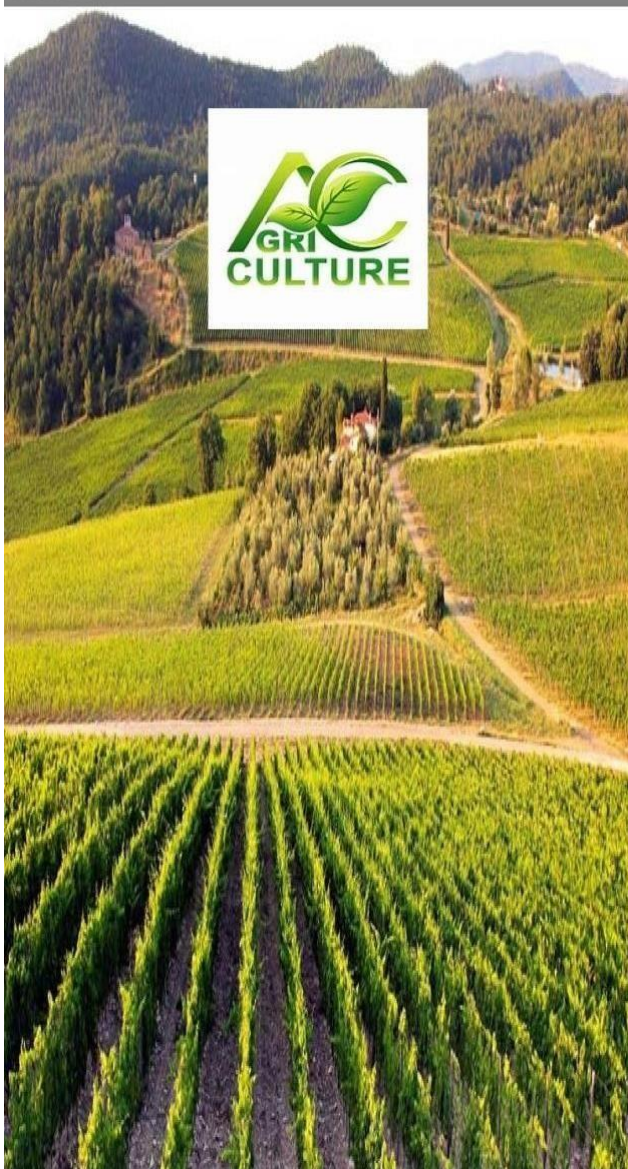


MOBILE APP WEB :



BLOCK DIAGRAM

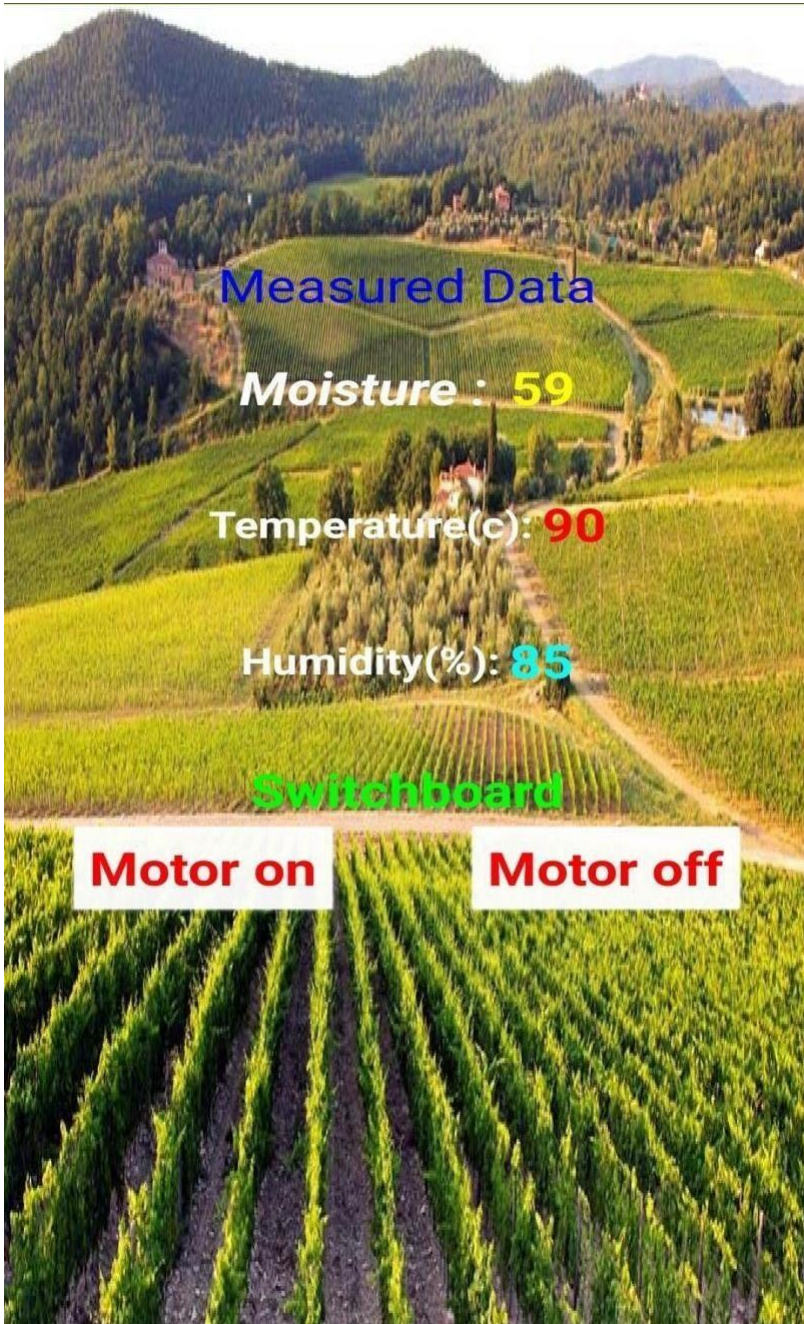
Screen1



SCREEN – 1



SCREEN - 2



SCREEN - 3

Web APP UI Home Tab

