

Project Development Phase Delivery of Sprint 3

Date	15 November 2022
Team ID	PNT2022TMID14459
Project Name	SmartFarmer – IoT Enabled Smart Farming Application

Configuration of Node-Red to send commands to IBM cloud

The screenshot shows the 'Edit ibmiot in node' configuration panel in Node-Red. The panel has a title bar with 'Delete', 'Cancel', and 'Done' buttons. Below the title bar is a 'Properties' tab with a settings icon, a copy icon, and a refresh 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 on the left and an edit icon on the right.
- Input Type:** A dropdown menu set to 'Device Event'.
- Device Type:** A checkbox labeled 'All or' followed by a text input field containing 'abcd'.
- Device Id:** A checkbox labeled 'All or' followed by a text input field containing '7654321'.
- Event:** A checkbox labeled 'All or' followed by a text input field containing '+'.
- Format:** A checkbox labeled 'All or' followed by a text input field containing 'json'.
- QoS:** A dropdown menu set to '0'.
- Name:** A text input field containing 'IBM IoT'.

At the bottom of the panel, there is a checkbox labeled 'Enabled' which is currently unchecked.

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 analyses is:

```
if(msg.payload===1)
```

```
msg.payload={"command": "ON"};
```

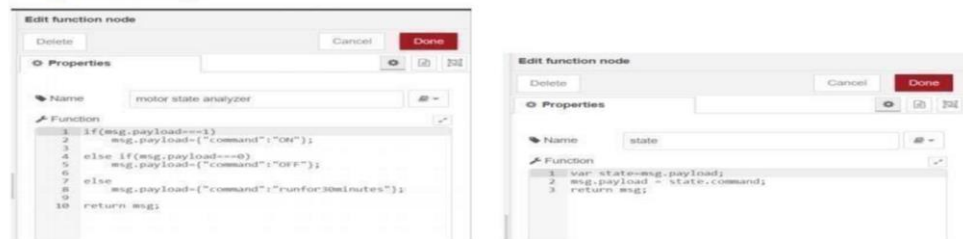
```
else if(msg.payload===0)
```

```
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.

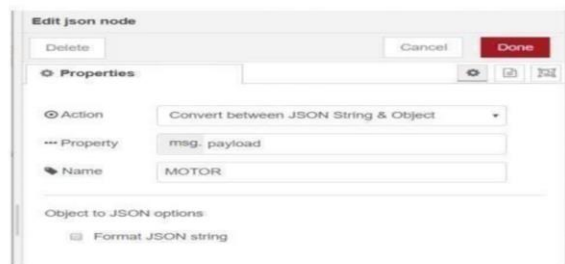
The Java script code for that function node is:

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

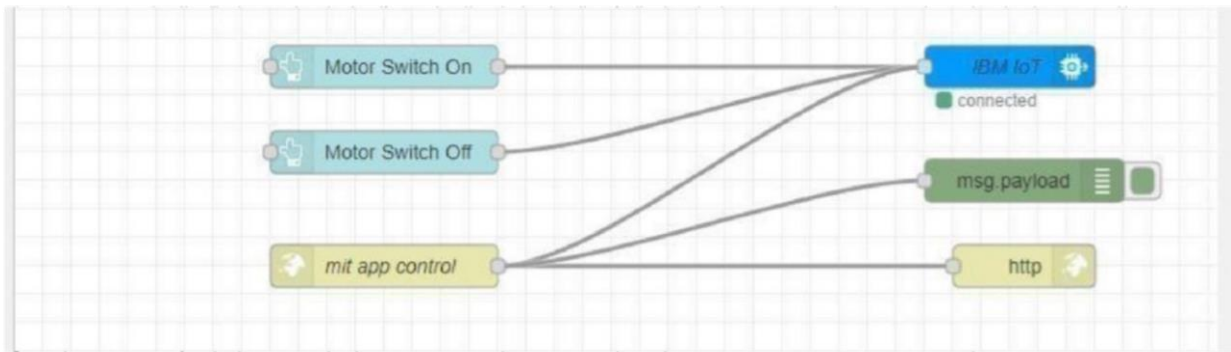


The above images show the java script codes of analyser and state function nodes.

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



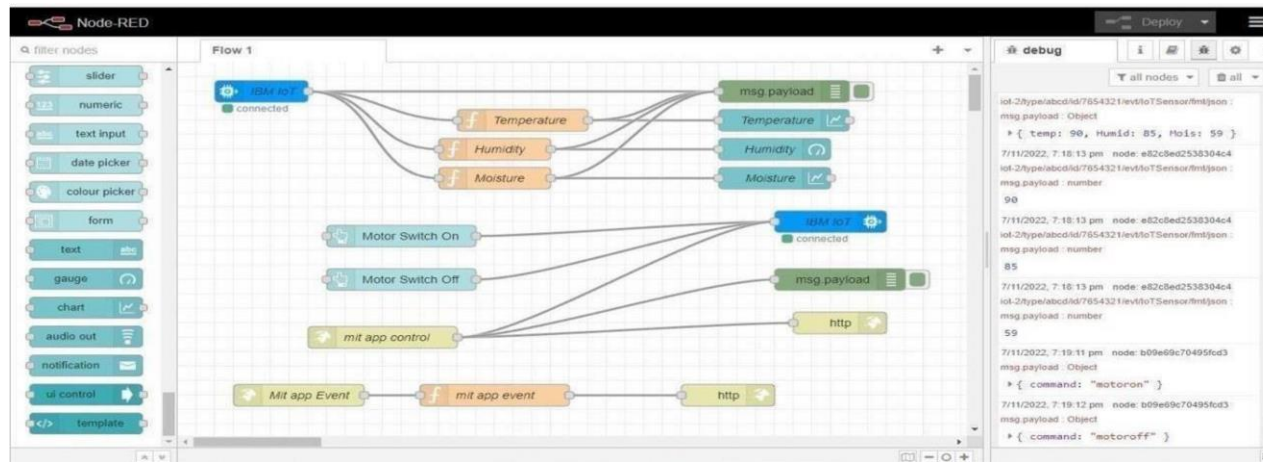
Edit JSON node needs to be configured like this



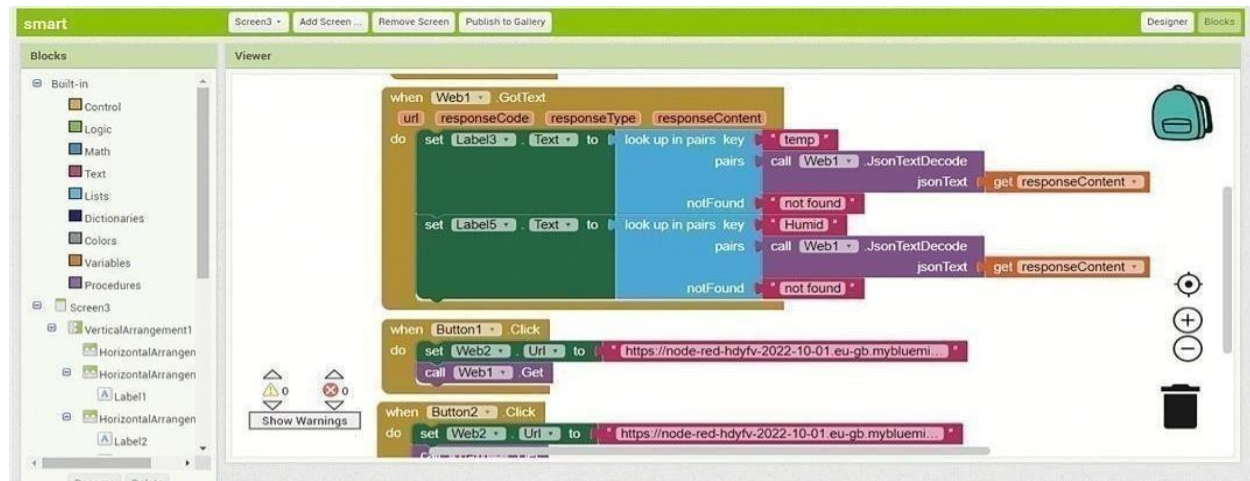
ADJUSTING USER INTERFACE

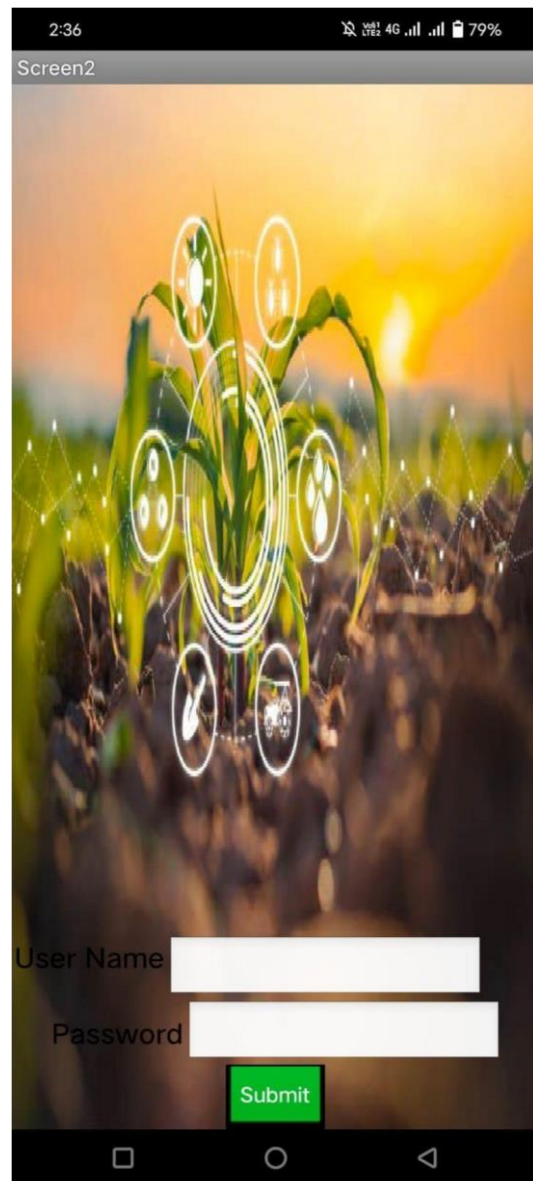
A screenshot of the Node-RED web interface. The main workspace shows a flow with an 'IBM IoT' node connected to 'Motor Switch On' and 'Motor Switch Off' nodes, and a 'mit app control' node. The 'Edit chart node' panel is open on the right, showing properties for a chart node. The 'Label' is 'Temperature', 'Type' is 'Line chart', 'X-axis' is 'last 5 minutes', 'X-axis Label' is 'HH:mm:ss', 'Y-axis' is 'min max', 'Legend' is 'None', and 'Series Colours' are set to blue, green, and red. A color picker is open over the 'Series Colours' section, showing a gradient from yellow to black and RGB values of 221, 192, and 3.

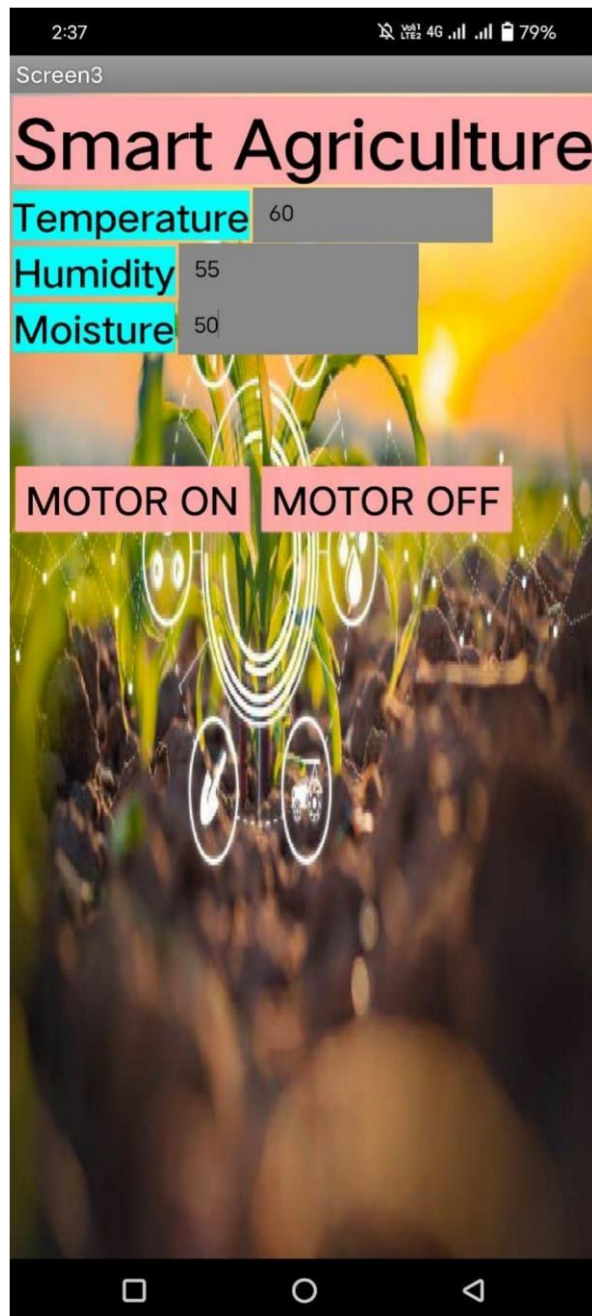
COMPLETE PROGRAM FLOW



MOBILE APP







WEB APP UI

