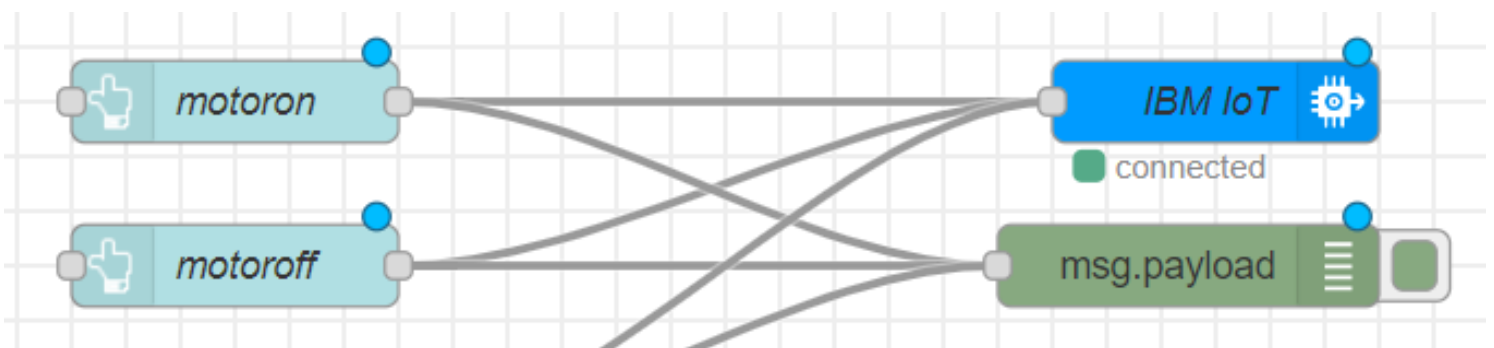


SPRINT- 3

Date	12 NOVEMBER 2022
Team ID	PNT2022TMID04663
Project Name	Smart Farmer-IoT Enabled smartFarming Application

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.
- Here we add two buttons in UI



- Edit the properties of ibmiot out node

The screenshot shows the Node-RED interface with the 'Edit ibmiot in node' dialog box open. The dialog box has a 'Delete' button, 'Cancel' button, and a red 'Done' button. Below these are tabs for 'Properties', 'Code', and 'Help'. The 'Properties' tab is active, showing the following configuration options:

- Authentication:** API Key (dropdown)
- API Key:** 5477d0eca8d5337d (text input with edit icon)
- Input Type:** Device Event (dropdown)
- Device Type:** All or + (dropdown)
- Device Id:** All or device id e.g. ab12cd231a21 (dropdown)
- Event:** All or + (dropdown)
- Format:** All or json (dropdown)
- QoS:** 0 (dropdown)
- Name:** IBM IoT (text input)
- Service:** registered (text input)

At the bottom of the dialog, there is a note: "Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to". Below this note is a checkbox labeled 'Enabled'.

The background flow diagram shows a sequence of nodes: 'ibmiot in' (connected), followed by 'Temp', 'Humid', 'Moisture', 'function', 'motoron', 'motoroff', and 'command'.

- Edit the properties of Motor ON Button node

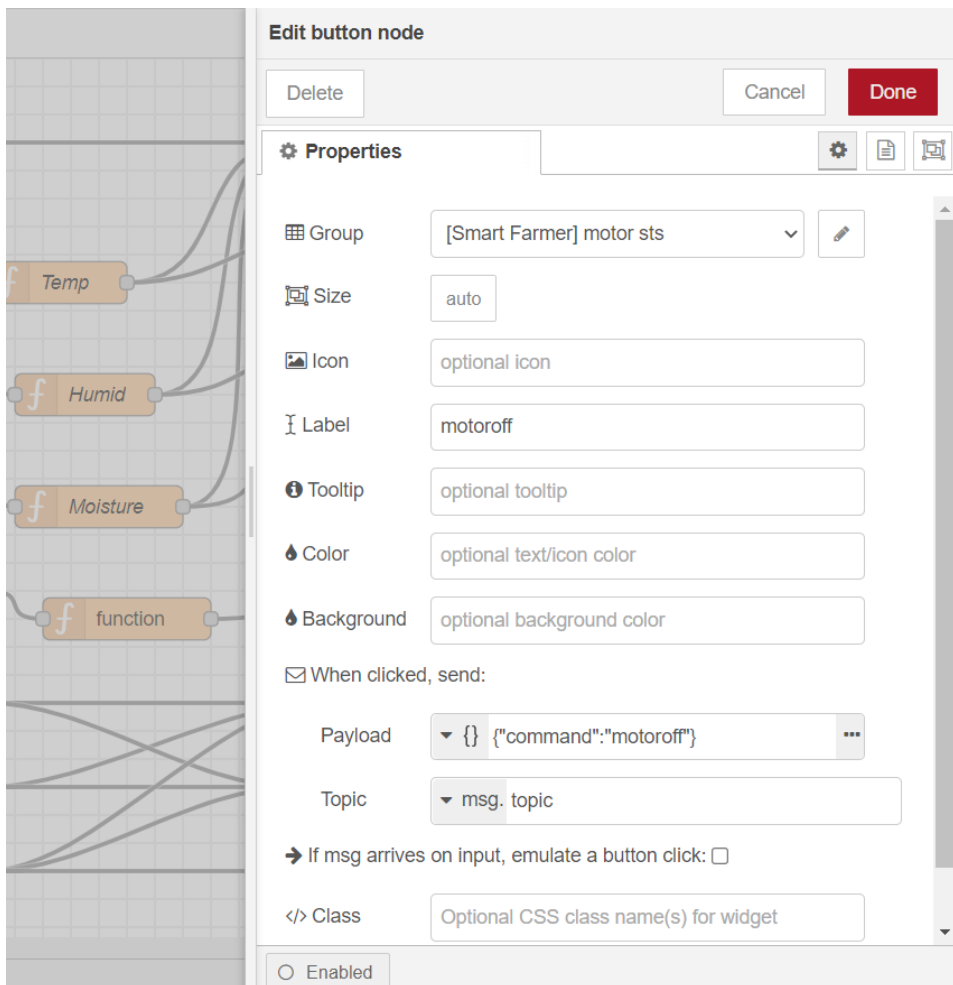
The screenshot shows the Node-RED interface with the 'Edit button node' dialog box open. The dialog box has a 'Delete' button, 'Cancel' button, and a red 'Done' button. Below these are tabs for 'Properties', 'Code', and 'Help'. The 'Properties' tab is active, showing the following configuration options:

- Group:** [Smart Farmer] motor sts (dropdown)
- Size:** auto (text input)
- Icon:** optional icon (text input)
- Label:** motoron (text input)
- Tooltip:** optional tooltip (text input)
- Color:** optional text/icon color (text input)
- Background:** optional background color (text input)
- When clicked, send:**
 - Payload:** {} ("command":"motoron") (text input with expand icon)
 - Topic:** msg. topic (text input)
- If msg arrives on input, emulate a button click:** ☐
- Class:** Optional CSS class name(s) for widget (text input)

At the bottom of the dialog, there is a checkbox labeled 'Enabled'.

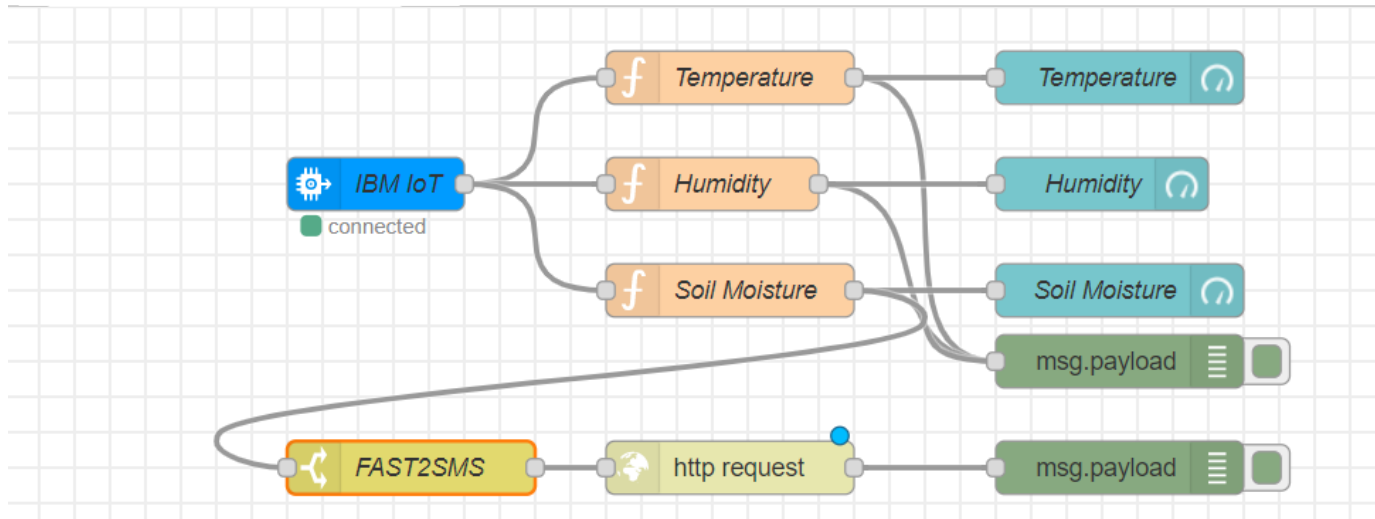
The background flow diagram shows a sequence of nodes: 'Temp', 'Humid', 'Moisture', 'function', 'motoron', 'motoroff', and 'command'.

- Edit the properties of Motor OFF Button node



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

Configuration of Node – Red with FAST2SMS :



- Edit the properties of FAST2SMS Switch node
 - If Soil Moisture is Less than 50, it will send the message to the particular number.

Edit switch node

Delete Cancel Done

Properties

Name FAST2SMS

Property msg. payload

☐ <= 50 → 1

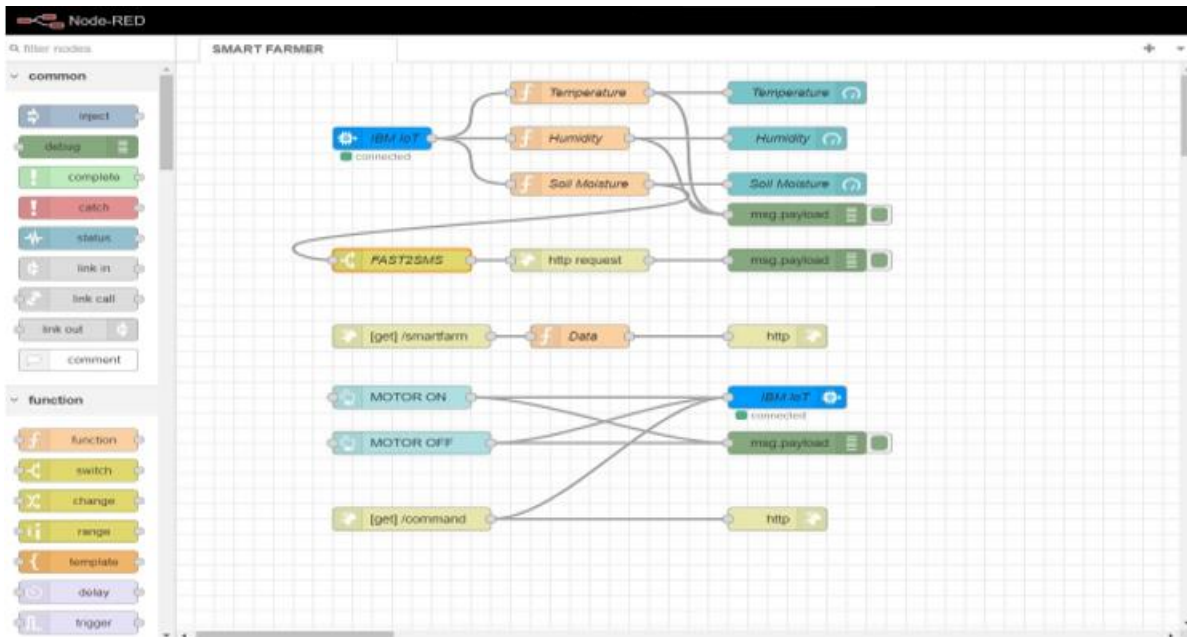
+ add

checking all rules

☐ recreate message sequences

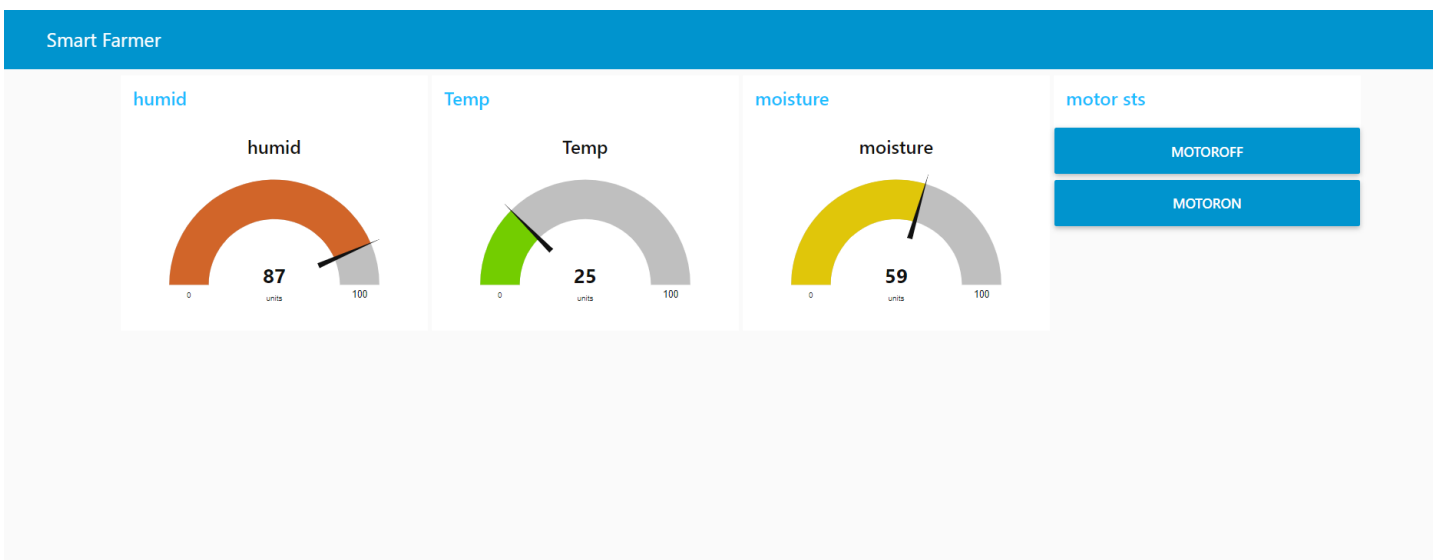
Enabled

Node – Red Complete Program Flow:



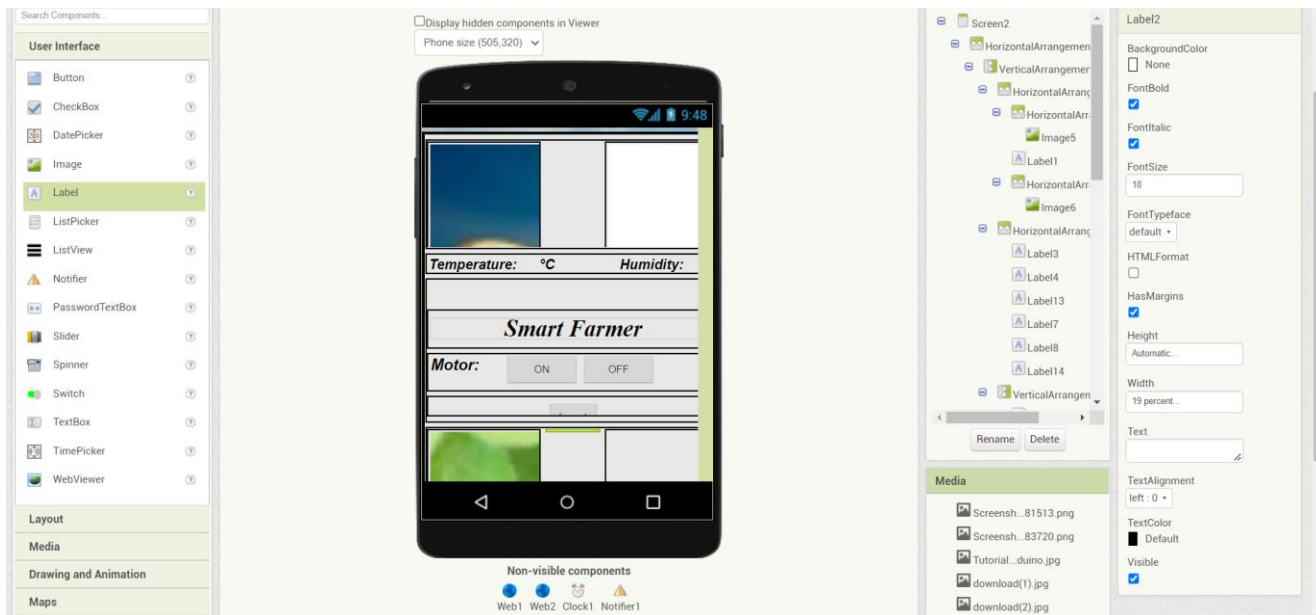
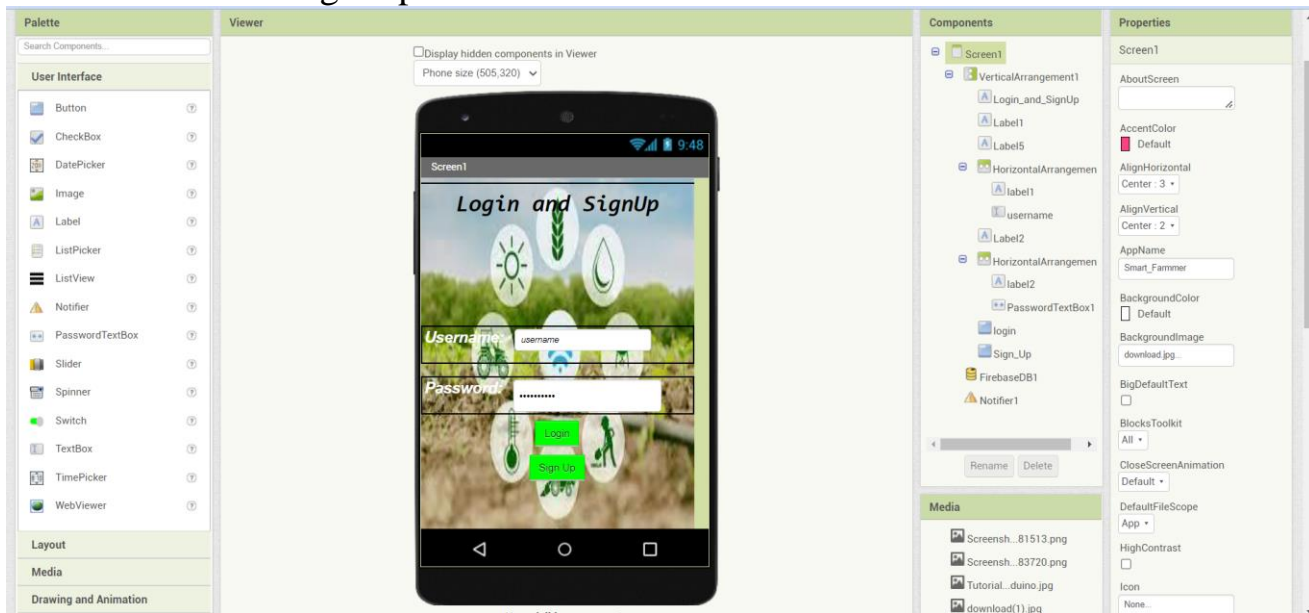
Node – Red Dashboard (WEB APP UI):

- In order to display the parsed JSON data a Node-Red dashboard is created.
- Below images are the Gauge, text and button node configurations.



MOBILE APP :

- MIT APP INVENTOR
 - Designer part



○ Blocks Part

MIT APP INVENTOR

Projects Connect Build Settings Help

My Projects View Trash Guide Report an Issue English nithinbasketball@gmail.com

Smart_Farmer Screen2 Add Screen ... Remove Screen Publish to Gallery Designer Blocks

Blocks

- Built-in
 - Control
 - Logic
 - Math
 - Text
 - Lists
 - Dictionaries
 - Colors
 - Variables
 - Procedures
- Screen2
 - HorizontalArrangement
 - VerticalArrangement
 - HorizontalArrangement
 - HorizontalArrangement
 - Image5
 - Label1

Viewer

when Clock1 -> Timer

do

- set Web1 -> Uri to https://node-red-mefew-2022-11-16.mybluemix.net/
- call Web1 -> Get

when Web1 -> GotText

do

- responseCode
- look up in pairs key temp
- call Web1 -> JsonTextDecode
- not found
- set Label9 -> Text to look up in pairs key Humid
- call Web1 -> JsonTextDecode
- not found
- set Label9 -> Text to look up in pairs key moisture
- call Web1 -> JsonTextDecode
- not found

when Button1 -> Click

do

- set Web2 -> Uri to https://node-red-mefew-2022-11-16.mybluemix.net/
- call Web2 -> Get
- call Notifier1 -> ShowAlert
- notice Motor is On!
- set HorizontalArrangement7 -> Image to Screenshots_20221119_183720.png
- set Label10 -> Text to ON

when Button2 -> Click

do

- set Web2 -> Uri to https://node-red-mefew-2022-11-16.mybluemix.net/
- call Web2 -> Get
- call Notifier1 -> ShowAlert
- notice Motor is On!
- set HorizontalArrangement7 -> Image to images2.jpg
- set Label10 -> Text to OFF

Show Warnings

Button function Blocks

MIT APP INVENTOR

Projects Connect Build Settings Help

My Projects View Trash Guide Report an Issue English nithinbasketball@gmail.com

Smart_Farmer Screen1 Add Screen ... Remove Screen Publish to Gallery Designer Blocks

Blocks

- Built-in
 - Control
 - Logic
 - Math
 - Text
 - Lists
 - Dictionaries
 - Colors
 - Variables
 - Procedures
- Screen1
 - VerticalArrangement1
 - Login_and_SignUp
 - Label1
 - Label5
 - HorizontalArrangement
 - Label1

Viewer

when login -> Click

do

- if
 - is empty username -> Text or is empty PasswordTextBox1 -> Text
 - then call Notifier1 -> ShowAlert
 - notice Enter the username and pass
 - else call FirebaseDB1 -> StoreValue
 - tag username -> Text
 - valueToStore PasswordTextBox1 -> Text
 - call Notifier1 -> ShowAlert
 - notice SignUp Successful

when Sign Up -> Click

do

- call FirebaseDB1 -> GetValue
- tag username -> Text
- valueIfTagNotThere NA

when FirebaseDB1 -> DataChanged

do

- if
 - get tag == username -> Text and get value == PasswordTextBox1 -> Text
 - then open another screen screenName Screen2
 - else call Notifier1 -> ShowAlert
 - notice Username and password is wrong!!

Show Warnings

○ Mobile App

SCREEN 1



SCREEN 2

