

LAPORAN JOBSHEET MINGGU 14
INTERNET OF THINGS
KONFIGURASI SMART DEVICE



Disusun Oleh :

RAJENDRA RAKHA ARYA PRABASWARA

(1941720080)

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

Praktikum

1. MQTT Node-RED

Walaupun beberapa protokol yang dapat disupport oleh Node-RED, akan tetapi pada kesempatan kali ini protokol yang digunakan MQTT. Untuk menggunakan protokol MQTT, pada Node-RED dashboard ketika melakukan instalasi sudah termasuk di dalamnya. Ikut langkah-langkah di bawah ini untuk mulai praktikum

1. Silakan seret node `inject` ke worksheet, kemudian ubahlah nilai properties seperti pada gambar berikut

The screenshot shows the 'Edit inject node' dialog box in Node-RED. The 'Name' field is 'publish message'. The 'msg. topic' field is 'room/suhu'. The 'Repeat' section is set to 'interval' with a value of '3' and unit 'seconds'. The 'Inject once after' checkbox is unchecked.

2. Seret juga node `function` ke worksheet, sesuaikan propertiesnya seperti pada gambar berikut

Edit function node

Delete
Cancel
Done

Properties

Name
Name

Setup
On Start
On Message
On Stop

```

1 msg.payload=Math.floor(Math.random()*100);
2 return msg;

```

- Jangan lupa seret juga node `mqtt out` pada kategori network, tambahkan server broker agar bisa publish data dengan cara klik icon pensil. Konfigurasinya adalah sebagai berikut adalah sebagai berikut

Edit mqtt out node > Add new mqtt-broker config node

Cancel
Add

Properties

Name
Mqtt Server AWS

Connection
Security
Messages

Server
broker.sinaungoding.com
Port
1883

☒ Connect automatically
☐ Use TLS

Protocol
MQTT V3.1.1

Client ID
Leave blank for auto generated

Keep Alive
60

Session
☒ Use clean session

Edit mqtt out node

Properties

Server: Mqtt Server AWS

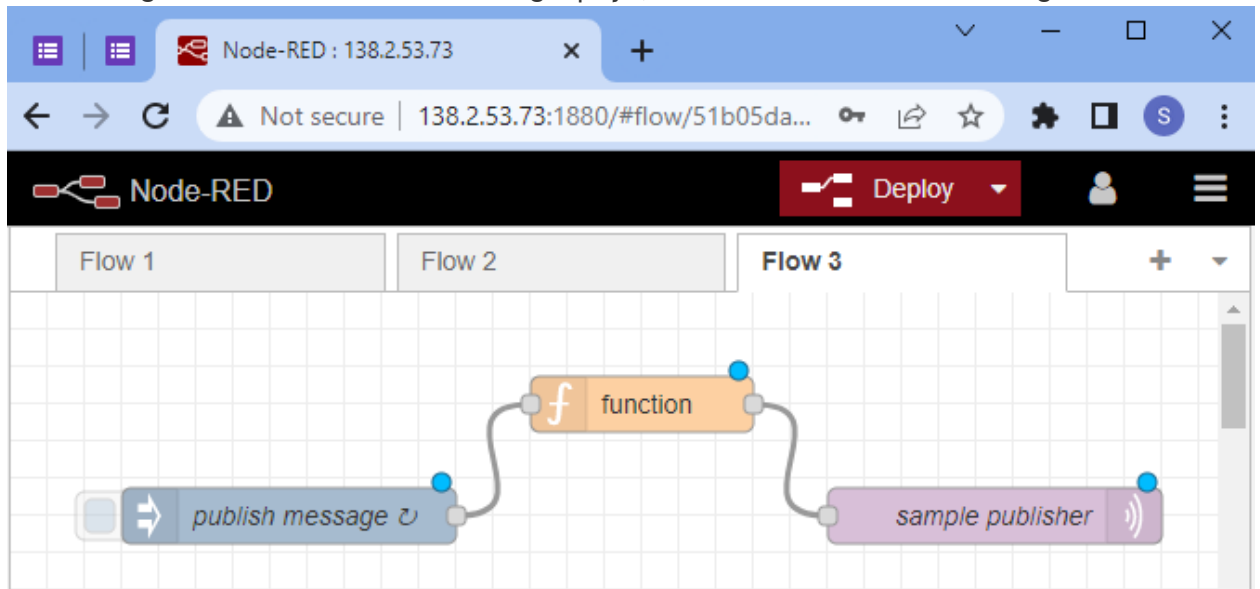
Topic: room/suhu

QoS: 2
 Retain:

Name: sample publisher

Tip: Leave topic, qos or retain blank if you want to set them via msg properties.

Perhatian gambar berikut untuk flow lengkapnya, setelah semua node dihubungkan.



4. Tambahkan node `mqtt in` ke worksheet, sesuaikan konfigurasi sebagai berikut

Edit mqtt in node

Delete Cancel Done

Properties

Server: Mqtt Server AWS

Action: Subscribe to single topic

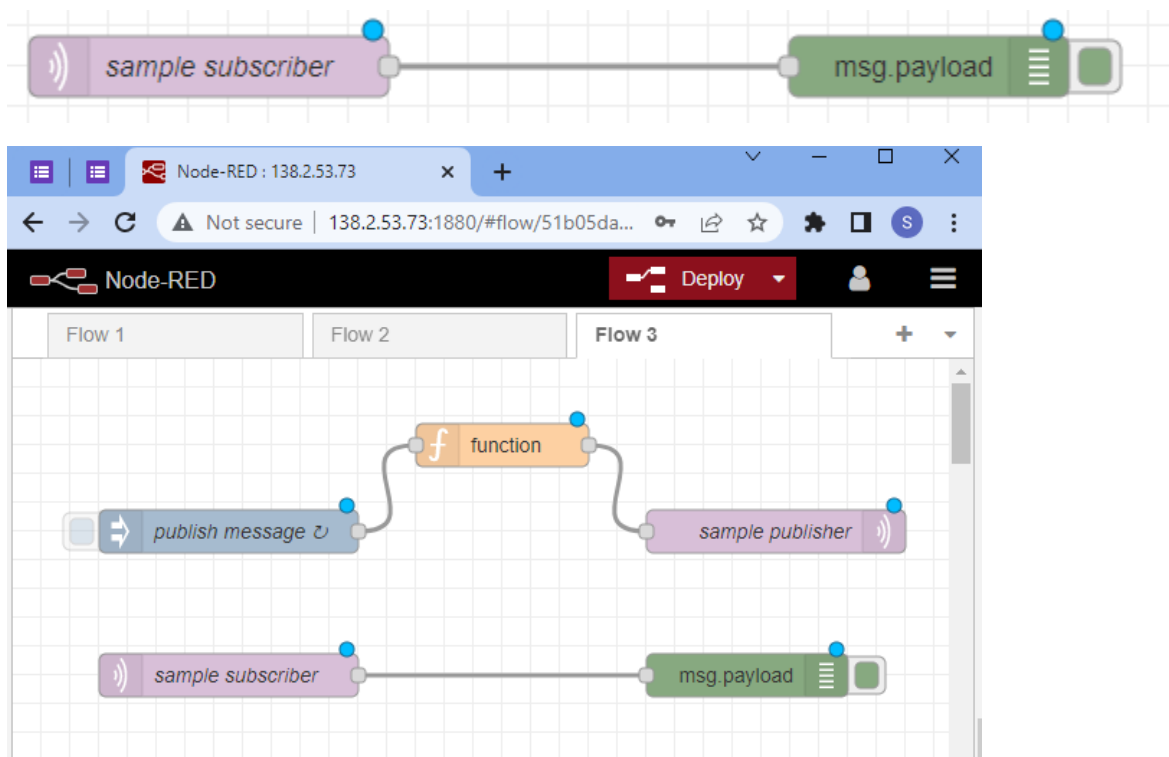
Topic: suhu/room

QoS: 2

Output: auto-detect (string or buffer)

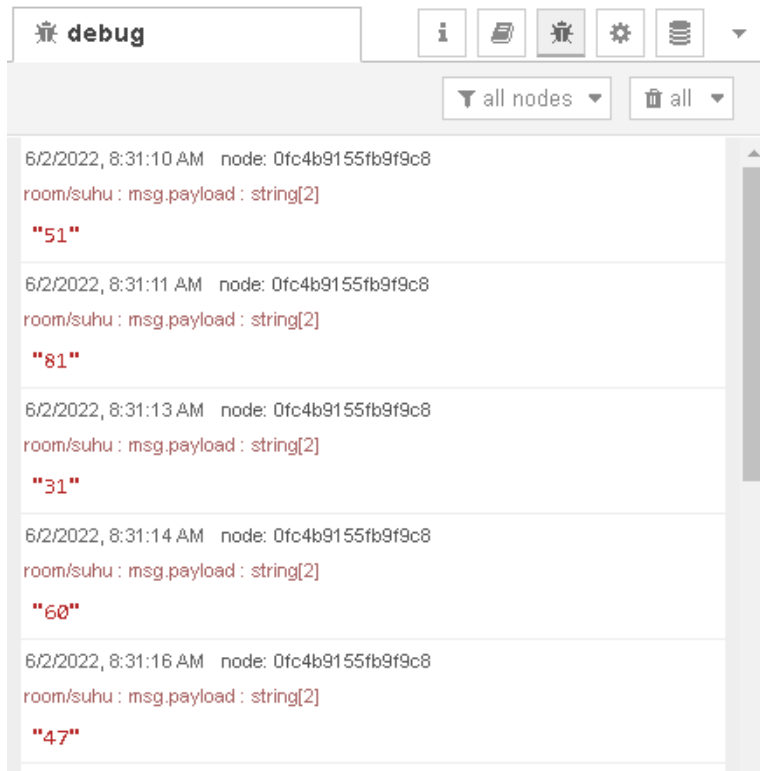
Name: sample subscriber

5. Tambahkan node terakhir yaitu node debug, sementara untuk kongfigurasinya tidak perlu disesuaikan. Hubungkan kedua node tersebut sehingga menjadi sebagai berikut



Verifikasi Hasil Percobaan

Setelah dilakukan deploy, kemudian klik tab debug atau icon kotak seharusnya adalah sebagai berikut. Adapun untuk nilainya pasti berbeda



Pertanyaan

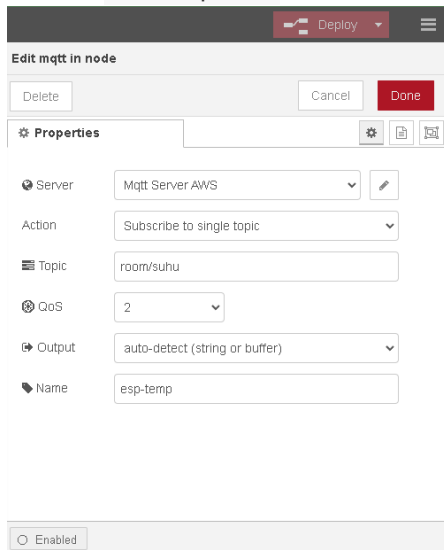
1. Pada node inject, pada properties Repeat dengan nilai interval. Apakah fungsinya?
JAWAB: Berfungsi untuk mengirimkan public message setiap 3 detik dengan mengatur payload kedalam public message dan direpeat 3 detik.
2. Apakah yang dimaksud dengan baris kode `msg.payload=Math.floor(Math.random()*100);`?
JAWAB: Maksud dari kode program itu adalah untuk menghasilkan angka acak antara angka 0 - 99 yang akan diteruskan kepada node grafik.
3. Bagian node mqtt out, apakah fungsi qos dengan nilai 2?

JAWAB: Berfungsi untuk memastikan setiap pesan hanya diterima satu kali oleh penerima.

2. Menghubungkan Smart Device Node-RED

Pada praktikum kali ini akan dibuat sebuah dashboard untuk menampilkan suhu yang dikirimkan oleh ESP-8266 menggunakan protokol MQTT. Langkah awal kita perlu menyiapkan Node-RED selanjutnya kita buat kode di sisi ESP-8266.

1. Silakan buat flow baru dengan cara klik tombol plus(+), tambahkan terlebih dahulu node `mqtt` in ke worksheet dengan konfigurasi sebagai berikut

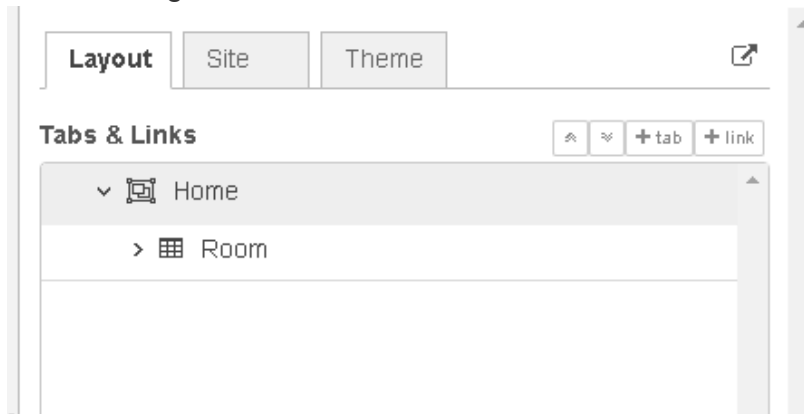


The screenshot shows the 'Edit mqtt in node' configuration window in Node-RED. At the top, there are buttons for 'Delete', 'Cancel', and 'Done'. Below is the 'Properties' section with the following settings:

- Server: Mqtt Server AWS
- Action: Subscribe to single topic
- Topic: room/suhu
- QoS: 2
- Output: auto-detect (string or buffer)
- Name: esp-temp

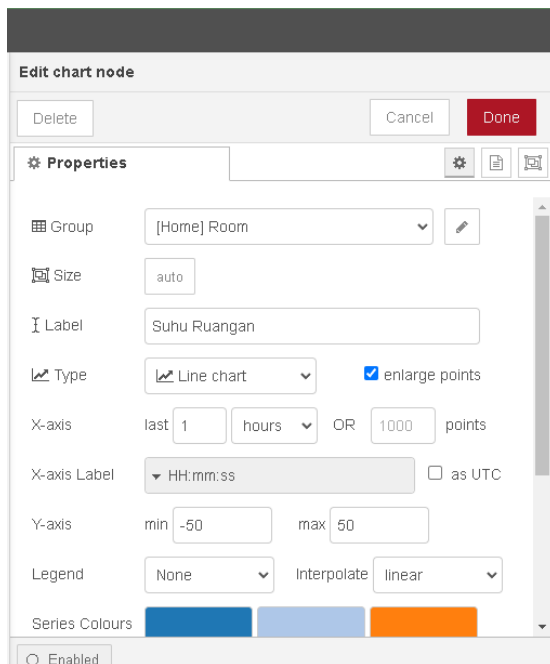
At the bottom, there is an 'Enabled' checkbox which is currently checked.

2. Buatlah dashboard dengan tab Site dengan title `Node-RED Dashboard` dengan layout adalah sebagai berikut

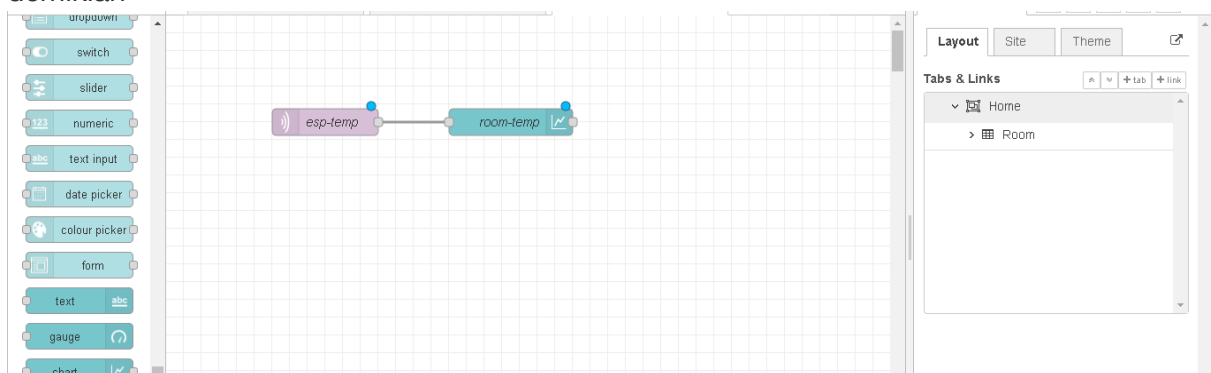


The screenshot shows the 'Layout' tab of the Node-RED Dashboard configuration. It features three tabs: 'Layout', 'Site', and 'Theme'. Below the tabs is a 'Tabs & Links' section with buttons for 'Home', 'Room', '+ tab', and '+ link'. The 'Home' tab is selected and expanded, showing a sub-tab 'Room'.

3. Tambahkan node `chart` dan sesuaikan konfigurasinya menjadi sebagai berikut
Jangan lupa pada bagian Name diisikan dengan `room-temp`.



4. Hubungkan kedua node tersebut dan Deploy, tampilan dashboard secara utuh menjadi demikian



Langkah terakhir yang perlu dilakukan yaitu membuat kode smart device, ESP8266. Tambahkan atau pasang sensor DHT11 yang rencananya akan kita tampilkan pada dashboard yang sebelumnya telah kita buat. Kode lengkapnya adalah sebagai berikut

```

main.cpp x platformio.ini PIO Home
vs-program14 > src > main.cpp > loop()
1 #include <Arduino.h>
2 #include <ESP8266WiFi.h>
3 #include <PubSubClient.h>
4 #include <SimpleDHT.h>
5
6 // hp
7 // const char *ssid = "od3ng";
8 // const char *password = "0d3n9bro";
9
10 // kampus
11 const char *ssid = "Smart Parking"; // sesuaikan dengan username wifi
12 const char *password = "5m4rT_P4rk1Ng"; // sesuaikan dengan password wifi
13 const char *mqtt_server = "broker.hivemq.com"; // isikan server broker
14
15 WiFiClient espClient;
16 PubSubClient client(espClient);
17
18 SimpleDHT11 dht11(D7);
19
20 long now = millis();
21 long lastMeasure = 0;
22 String macAddr = "";

```



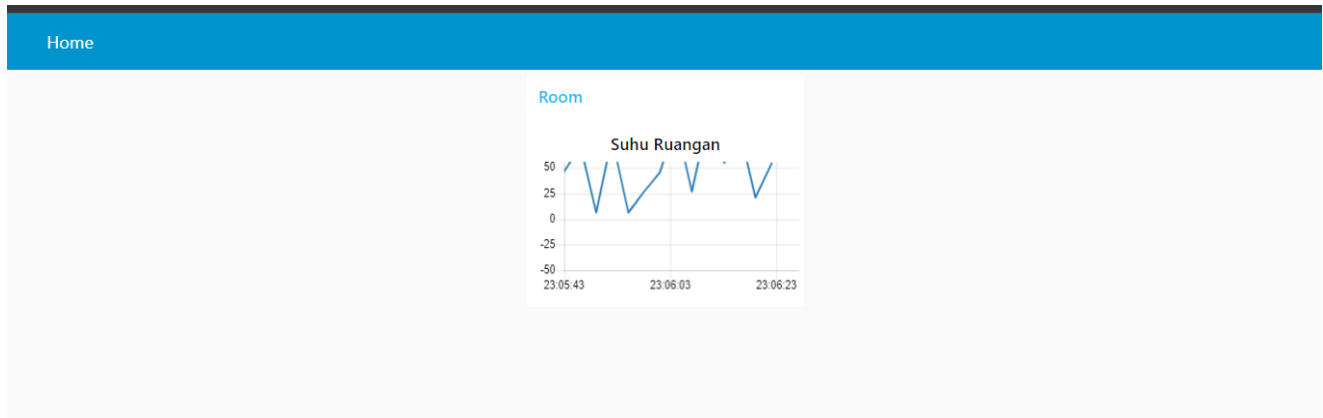
```
main.cpp x platformio.ini PIO Home
vs-program14 > src > main.cpp > loop()

24 void setup_wifi()
25 {
26     delay(10);
27     Serial.println();
28     Serial.print("Connecting to ");
29     Serial.println(ssid);
30     Wifi.begin(ssid, password);
31     while (Wifi.status() != WL_CONNECTED)
32     {
33         delay(500);
34         Serial.print(".");
35     }
36     Serial.println("");
37     Serial.print("Wifi connected - ESP IP address: ");
38     Serial.println(Wifi.localIP());
39     macAddr = Wifi.macAddress();
40     Serial.println(macAddr);
41 }
42
43 void reconnect()
44 {
45     while (!client.connected())
46     {
47         Serial.print("Attempting MQTT connection...");
48         if (client.connect(macAddr.c_str()))
49         {
50             Serial.println("connected");
51         }
52         else
53         {
54             Serial.print("Failed, rc=");
55             Serial.print(client.state());
56             Serial.println(" try again in 5 seconds");
57             delay(5000);
58         }
59     }
60 }
61
62 void setup()
63 {
64
65
66
67
68
69
70 void loop()
71 {
72     if (!client.connected())
73     {
74         reconnect();
75     }
76     if (!client.loop())
77     {
78         client.connect(macAddr.c_str());
79     }
80     now = millis();
81     if (now - lastMeasure > 5000)
82     {
83         lastMeasure = now;
84         int err = SimpleDHTerrSuccess;
85
86         byte temperature = 0;
87         byte humidity = 0;
88         if ((err = dht11.read(&temperature, &humidity, NULL)) != SimpleDHTerrSuccess)
89         {
90             Serial.print("Pembacaan DHT11 gagal, err=");
91             Serial.println(err);
92             delay(1000);
93             return;
94         }
95         static char temperatureTemp[7];
96         dtostrf(temperature, 4, 2, temperatureTemp);
97         Serial.println(temperatureTemp);
98
99         client.publish("room/suhu", temperatureTemp); // agar lebih unix silakan tambahkan NIM ex: 0001/room/suhu
100     }
101 }
```

Upload kode di atas ke ESP8266 Anda dan amati hasilnya.

Verifikasi Hasil Percobaan

Hasil percobaan tersebut dapat dilihat pada serial monitor dan Node-RED melalui browser, kurang lebih seperti gambar di bawah



Pertanyaan

1. Modifikasi program di ESP8266 di atas agar bisa melakukan subscribe dengan topik room/lamp?

