

## MODUL 6 – VISUALISASI DENGAN PLATFORM NODE-RED

### 1.1. Tujuan

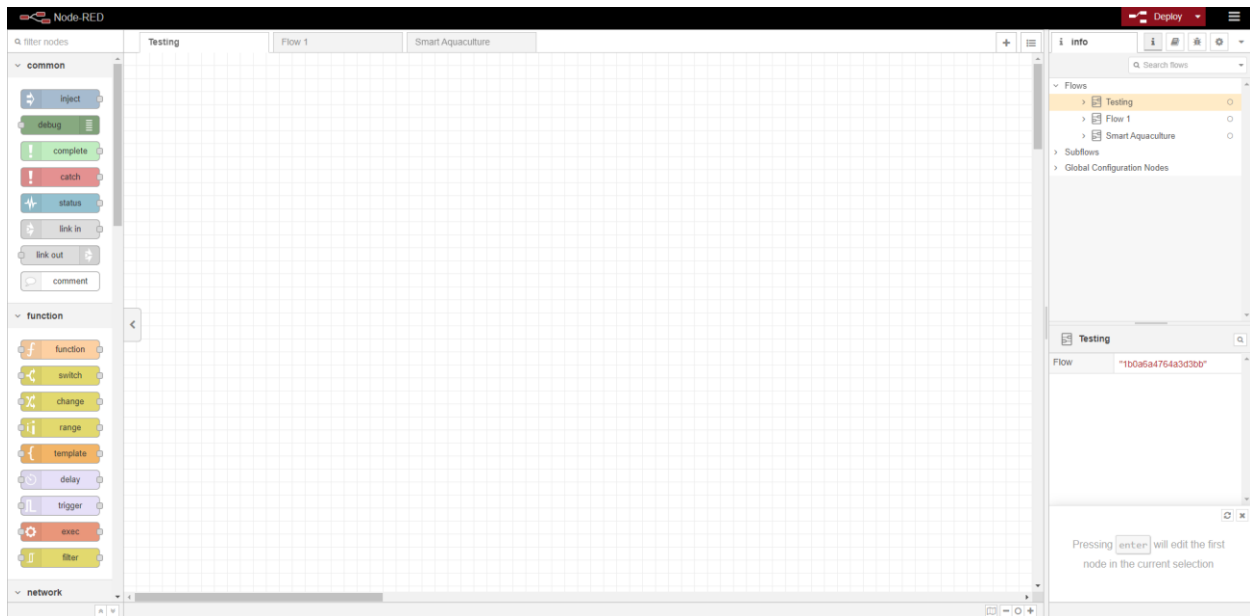
- Mahasiswa mampu untuk menampilkan data yang dikirimkan oleh Node secara realtime.
- Mahasiswa mampu untuk menampilkan data dalam bentuk grafik.
- Mahasiswa mampu untuk menyimpan data yang dikirimkan oleh Node.
- Mahasiswa mampu untuk menampilkan data yang tersimpan dalam bentuk grafik.

### 1.2. Praktikum

#### 1.2.1. Platform IoT

Sekarang ini dukungan untuk teknologi IoT sudah mulai banyak. Salah satunya adalah beberapa platform dan tools yang menyediakan konektifitas antar node IoT. Menampilkan data dan mengontrol data sudah dapat dilakukan oleh platform-platform tersebut. Platform tersebut diantaranya adalah Node-Red, Blink, ThingSpeak, ThingsBoard, KaaIoT, Openremote dan lain sebagainya. Pada praktikum ini kita akan menggunakan salah satu dari platform atau tools tersebut yaitu Node-Red.

Node-Red merupakan platform yang menyediakan macam-macam pengolahan data hingga visualisasi data dengan menggunakan diagram blok. Berikut ini adalah tampilan antarmuka Node-Red (Gambar 1).



Gambar 1. Antarmuka Node-Red

Untuk dapat menggunakan Node-Red kalian bisa melakukan instalasi Node-Red tersebut melalui website official-nya disini (<https://nodered.org/docs/getting-started/windows>).

## Quick Start

### 1. Install Node.js

Download the latest 14.x LTS version of Node.js from the official **Node.js home page**. It will offer you the best version for your system.

Run the downloaded MSI file. Installing Node.js requires local administrator rights; if you are not a local administrator, you will be prompted for an administrator password on install. Accept the defaults when installing. After installation completes, close any open command prompts and re-open to ensure new environment variables are picked up.

Once installed, open a command prompt and run the following command to ensure Node.js and npm are installed correctly.

Using Powershell: `node --version; npm --version`

Using cmd: `node --version && npm --version`

You should receive back output that looks similar to:

```
v14.17.2.0
6.14.13
```

### 2. Install Node-RED

Installing Node-RED as a global module adds the command `node-red` to your system path. Execute the following at the command prompt:

```
npm install -g --unsafe-perm node-red
```

### 3. Run Node-RED

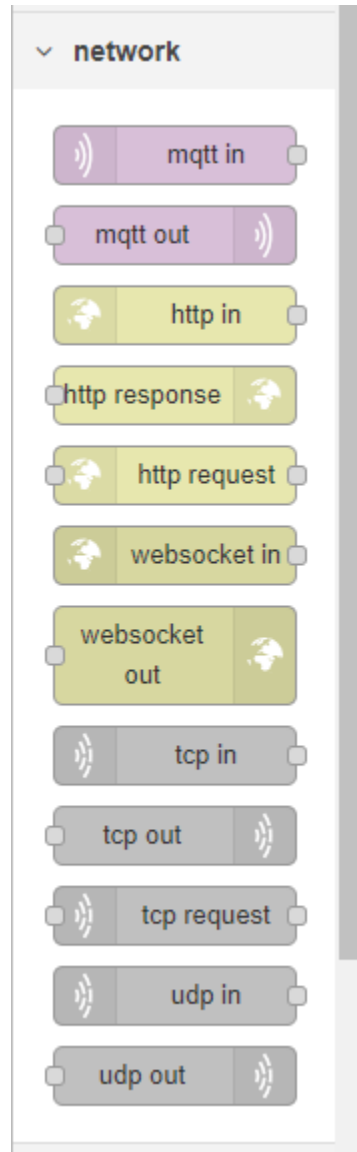
Once installed, you are ready to **run Node-RED**.

Gambar 2. Langkah-langkah menginstall Node-Red

### 1.2.2. Pengenalan Node-Red

Pada dasarnya platform merupakan suatu aplikasi yang dirancang untuk dapat digunakan secara dinamis agar dapat digunakan untuk membangun suatu sistem dalam bentuk yang lain dan kegunaan yang lain sesuai user yang menggunakannya. Node-Red disini memiliki fitur

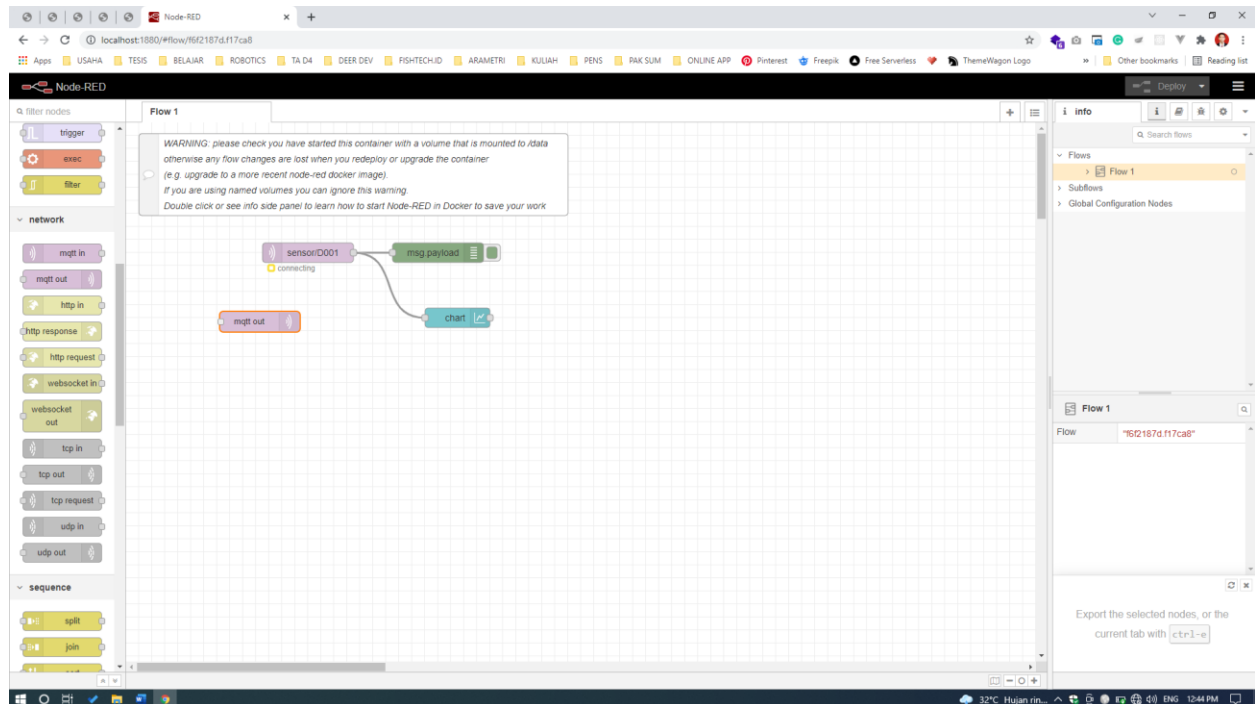
konektifitas yang dapat terhubung dengan perangkat IoT melalui beberapa protocol komunikasi. Salah satunya yang akan dipraktekan adalah protocol komunikasi MQTT. Letak blok diagram komunikasi MQTT berada pada group network seperti berikut.



*Gambar 3. Komponen group network*

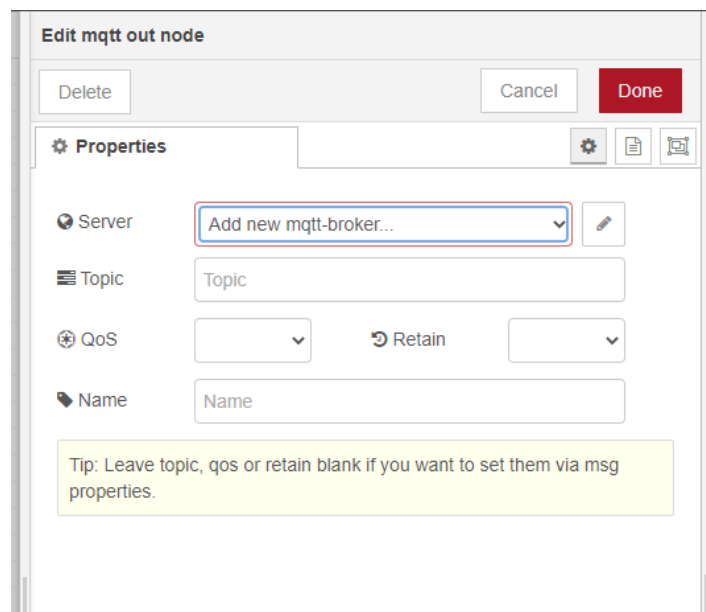
Blok *mqtt in* digunakan untuk melakukan subscribe sedangkan *mqtt out* digunakan untuk melakukan publish data.

Cara menggunakan blok-blok komponen pada Node-Red tersebut cukup dengan melakukan drag-and-drop komponen tersebut.



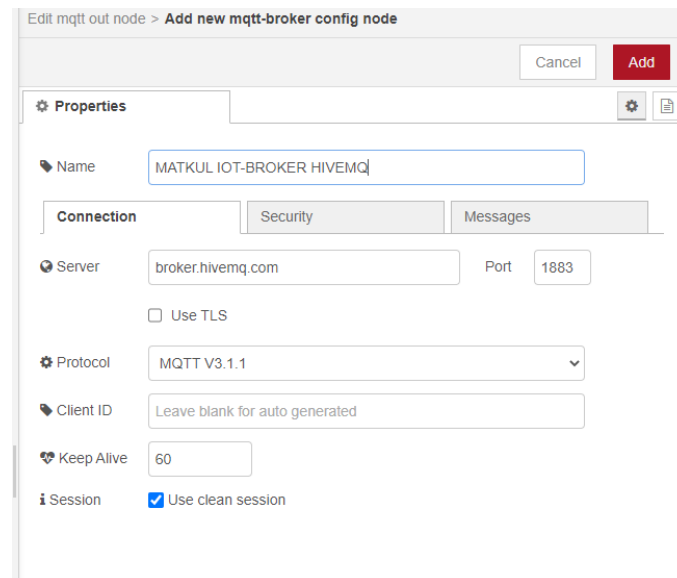
Gambar 4. Drag and Drop Komponen

Membuat koneksi ke MQTT broker dapat dilakukan dengan melakukan double klik pada komponen *mqtt in* atau *mqtt out*, selanjutnya akan keluar window seperti berikut ini.



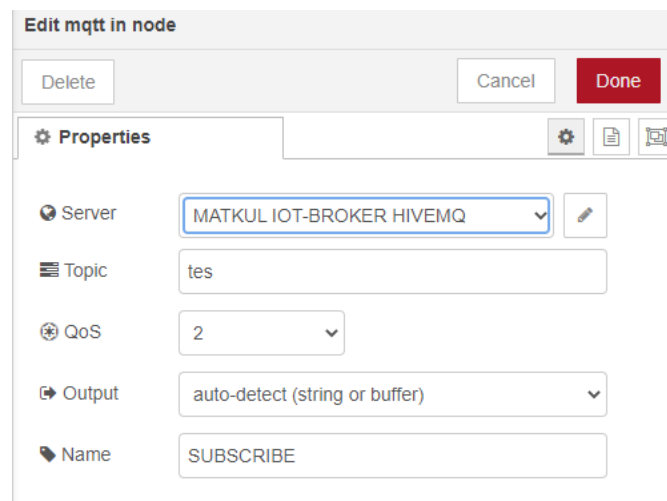
Gambar 5. Membuat koneksi ke Broker

Klik pada icon pencil maka akan memunculkan konfigurasi koneksi ke Broker seperti berikut ini. Selanjutnya klik Add untuk melakukan simpan settingan tersebut.



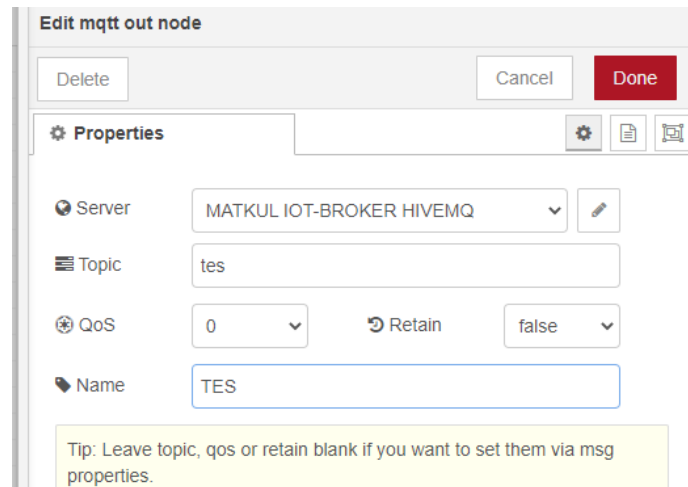
*Gambar 6. Konfigurasi server Broker*

Subscribe merupakan cara untuk mendapatkan data yang dikirimkan oleh masing-masing node yang bertindak sebagai publisher. Dalam Node-Red untuk melakukan subscriber dapat dilakukan dengan menggunakan komponen mqtt in. berikut ini adalah konfigurasi untuk melakukan subscribe.



*Gambar 7. Konfigurasi sebagai subscriber*

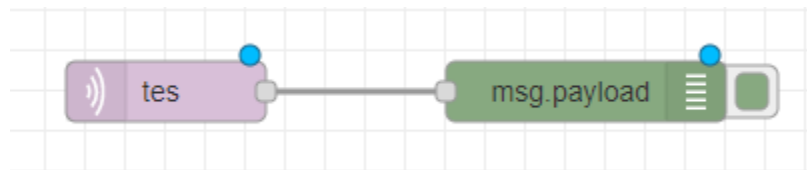
Sedangkan apabila ingin mengirimkan data yaitu dilakukan dengan melakukan publish data dengan menggunakan komponen mqtt out seperti berikut ini.



Gambar 8. Konfigurasi sebagai publisher

### 1.2.3. Menampilkan Data yang dikirim Publisher di Node-Red

Untuk menampilkan data yang di kirimkan oleh publisher dapat dilakukan dengan menampilkannya pada window debugger. Berikut ini adalah diagramnya.



Gambar 9. Menampilkan data yang dipublish pada debugger

Komponen  dihubungkan ke . atur konfigurasi pada komponen mqtt in pada broker yang sudah disiapkan dari proses sebelumnya seperti berikut ini.

Edit mqtt in node

Delete Cancel Done

Properties

Server MATKUL IOT

Topic tes

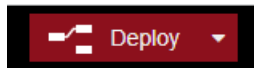
QoS 2

Output auto-detect (string or buffer)

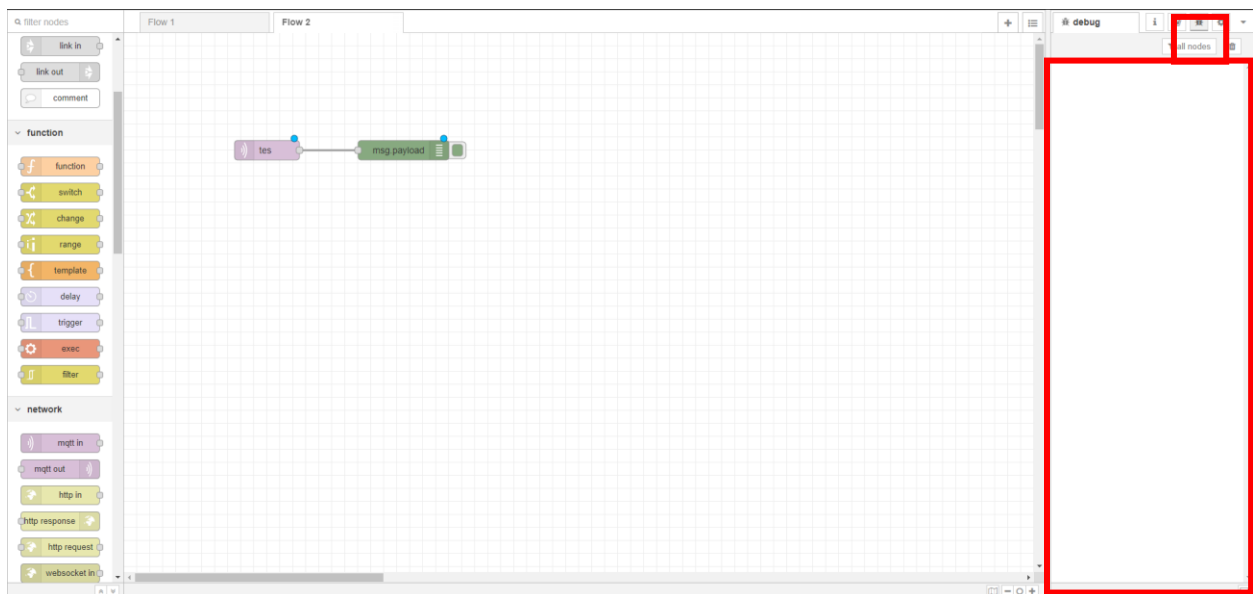
Name Name

Gambar 10. Konfigurasi mqtt in sebagai subscriber

Setelah semua konfigurasi selesai jalankan programnya dengan mengklik tombol **Deploy**

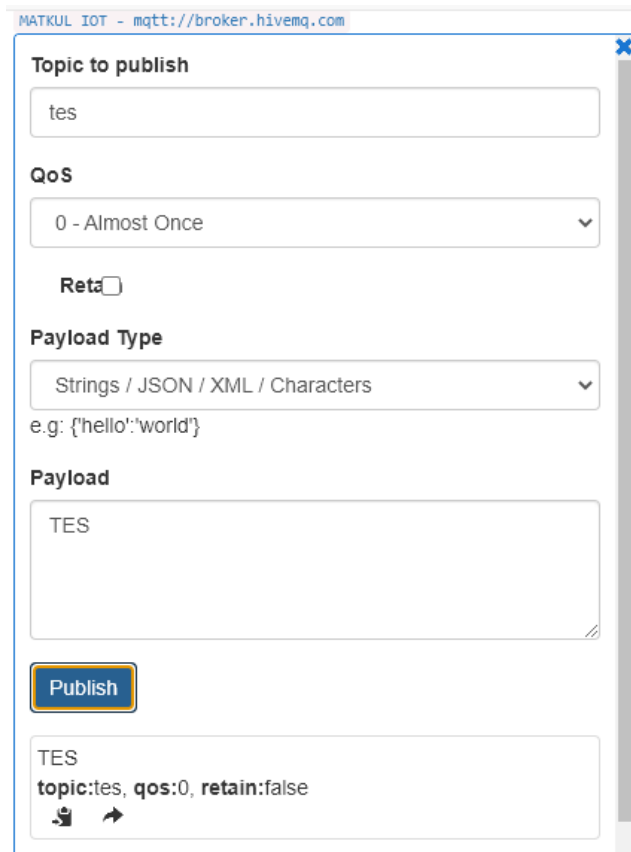


Kemudian buka window debugger pada bagian kanan Node-Red untuk melihat data dari publisher tersebut.



Gambar 11. Window debugger

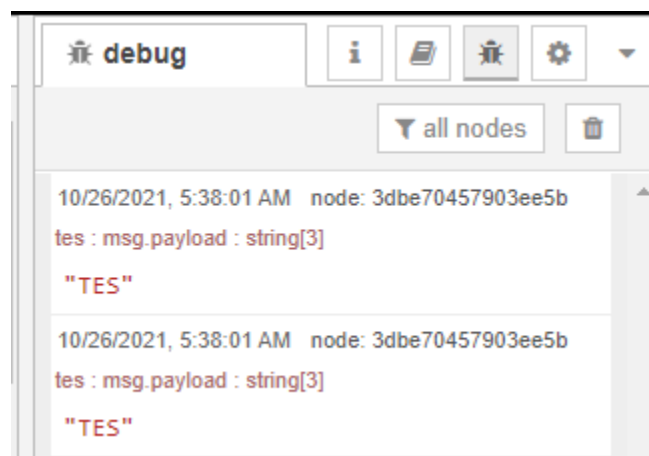
untuk melakukan publish data dapat disimulasikan dengan menggunakan MQTT Box seperti berikut ini.



The screenshot shows the MQTT Box web interface with the following fields and values:

- Topic to publish:** tes
- QoS:** 0 - Almost Once
- Retain:** ☐
- Payload Type:** Strings / JSON / XML / Characters
- Payload:** TES
- Buttons:** Publish
- Summary:** TES, topic:tes, qos:0, retain:false


Gambar 12. Simulasi publish data

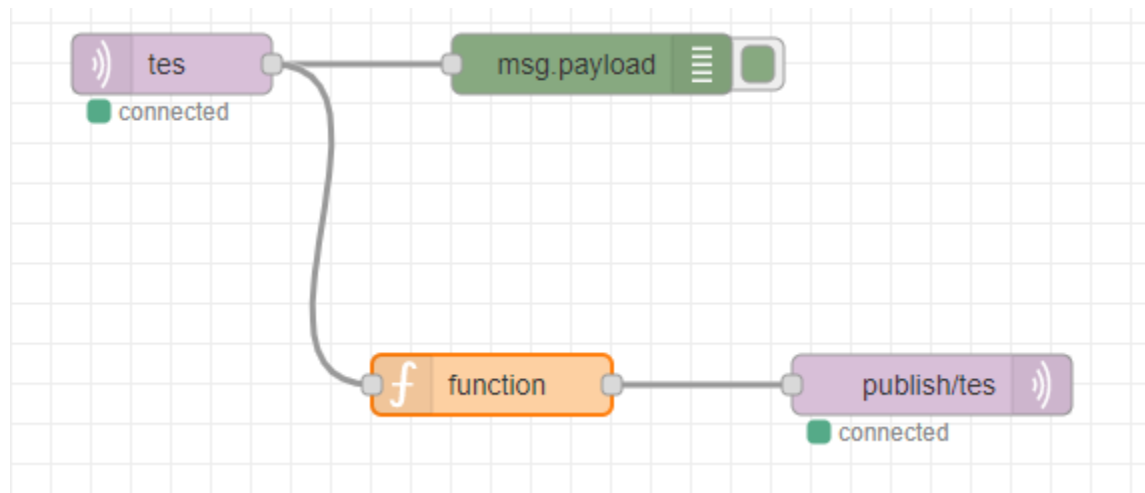


Gambar 13. Melihat hasil data yang dikirimkan oleh publisher



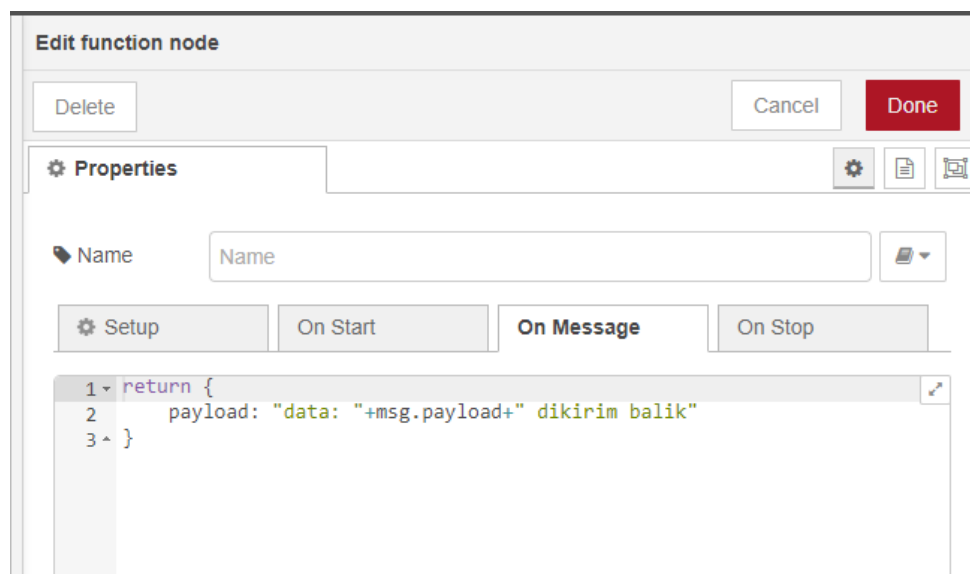
#### 1.2.4. Publish data di Node-Red

Pada Node-Red dapat juga melakukan publish data dengan menggunakan komponen mqtt out . Berikut ini adalah cara untuk melakukan publish data dengan menggabungkan script menggunakan javascript.



Gambar 14. Diagram publish data dari data yang disubscribe

Komponen yang digunakan adalah komponen function dan mqtt out. Komponen function digunakan untuk mensisipkan script dalam bahasa javascript berfungsi untuk mengolah data/mentransformasikan sedangkan komponen mqtt out berfungsi sebagai pengirim (publisher) data dari komponen function. Double klik pada komponen function dan isikan program seperti berikut.



Gambar 15. Isi program pada komponen function

Delete
Cancel
Done

Properties

Server
MATKUL IOT

Topic
publish/tes

QoS
Retain

Name
Name

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

Gambar 16. Konfigurasi komponen mqtt out

MATKUL IOT - mqtt://broker.hivemq.com

Topic to publish
tes

QoS
0 - Almost Once

Retain
☐

Payload Type
Strings / JSON / XML / Characters

e.g: {'hello':'world'}

Payload
OKE

Publish

OKE
topic:tes, qos:0, retain:false

publish/tes

data: OKE dikirim balik
qos : 0, retain : false, cmd : publish, dup : false, topic : publish/tes, messageid : , length : 36

data: OKE dikirim balik
qos : 0, retain : false, cmd : publish, dup : false, topic : publish/tes, messageid : , length : 36

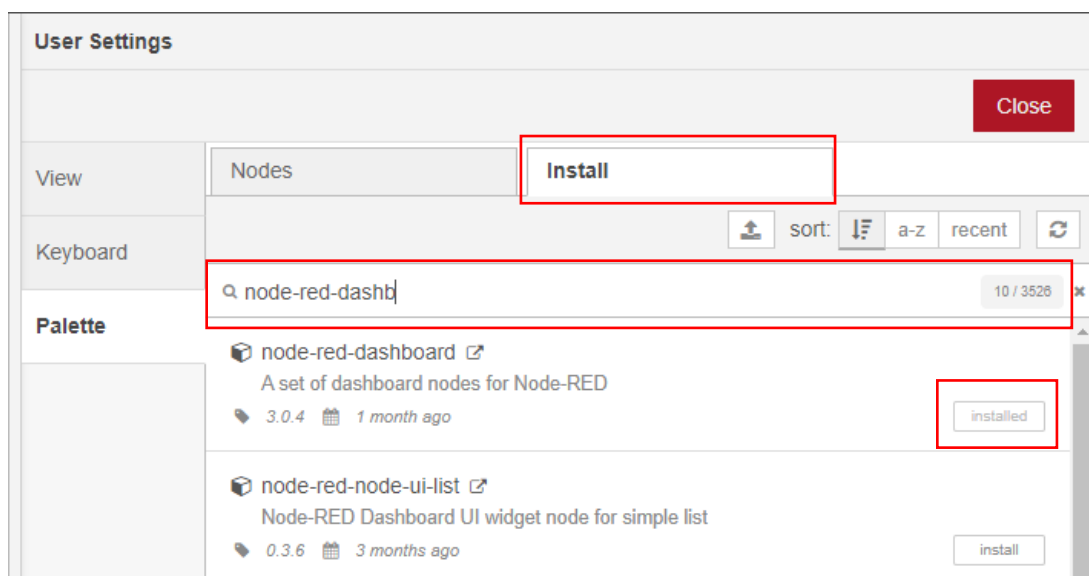
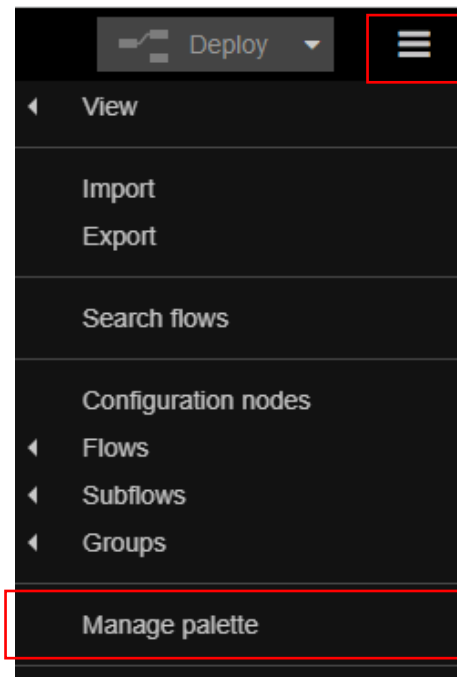
data: OKE dikirim balik
qos : 0, retain : false, cmd : publish, dup : false, topic : publish/tes, messageid : , length : 36

data: OKE dikirim balik
qos : 0, retain : false, cmd : publish, dup : false, topic : publish/tes, messageid : , length : 36

Gambar 17. Hasil simulasi pengiriman data dan subscribe data

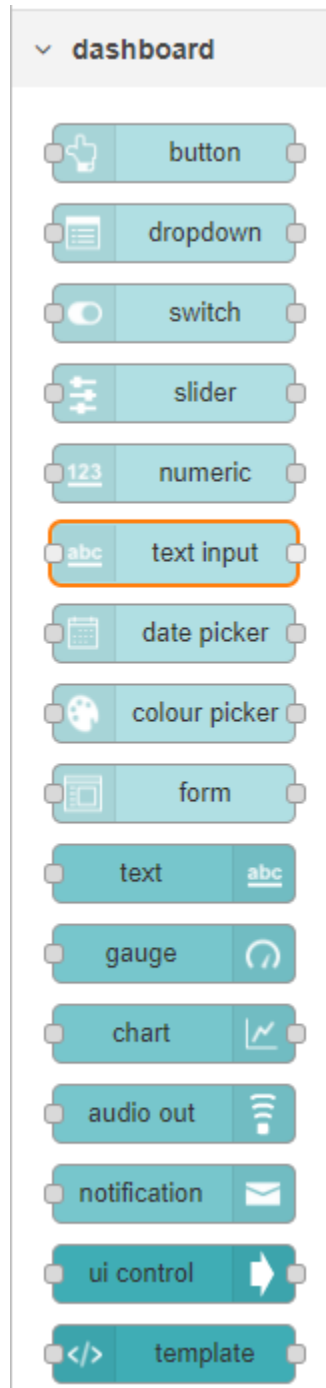
### 1.2.5. Visualisasi data

Dalam Node-Red banyak sekali plugin atau palette tambahan yang dapat digunakan untuk pengembangan lebih advance lagi. Salah satunya disini kita akan menggunakan palette tambahan untuk dapat memvisualisasikan data dalam bentuk grafik. Berikut cara menambahkan palette tersebut.



Gambar 18. Langkah-langkah menginstall palette ui dashboard

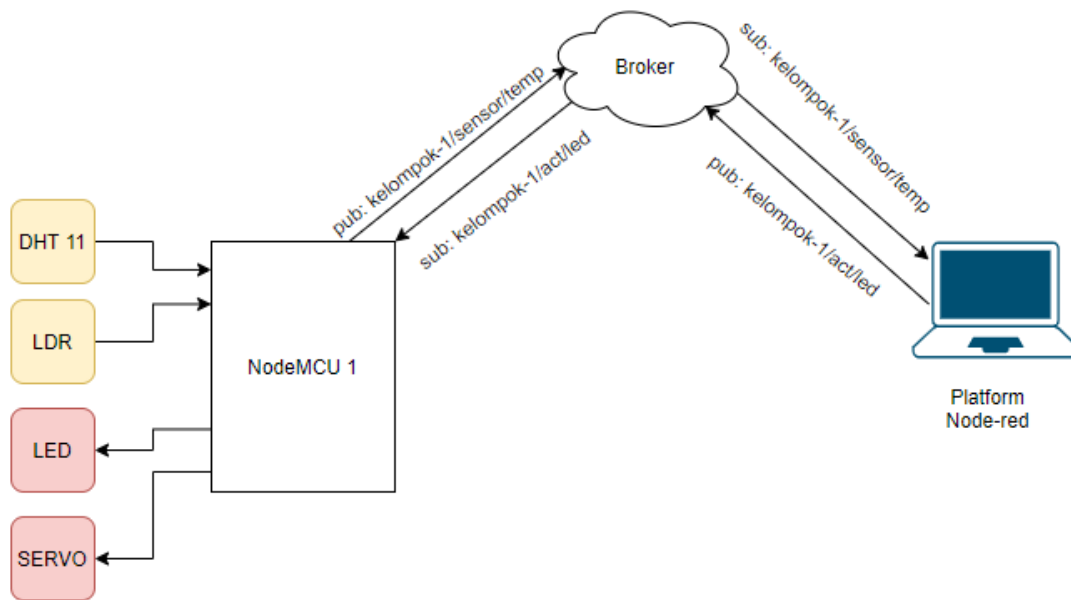
Setelah proses instalasi selesai maka dapat kita jumpai komponen baru paling bawah berupa komponen dashboard seperti berikut ini.



Gambar 19. Komponen dashboard

### 1.3.Tugas

1. Dokumentasikan setiap langkah-langkah instalasi Node-Red berserta menjalankan Node-Red.
2. Menampilkan data dalam bentuk grafik Gauge untuk masing-masing paramter sensor yaitu Humidity, Temperature dan LDR.
3. Melakukan hal yang sama seperti praktikum sebelumnya namun menggantikan kontrol sistemnya menggunakan platform Node-Red dengan ketentuan berikut ini.



NodeMCU 1 bertugas sebagai pengambil data sensor dan terhubung dengan LED. Sedangkan Device Node-Red bertugas sebagai pemroses data yang dikirimkan oleh NodeMCU 1 melalui jaringan dengan protokol MQTT. Berikut ini adalah Rule dari kasus tersebut.

No	Sensor	Servo	LED
1	Temperature > 29	90 derajat	-
2	Temperature <= 29	0 derajat	-
3	LDR mendeteksi cahaya		ON
4	LDR tidak mendeteksi cahaya		OFF