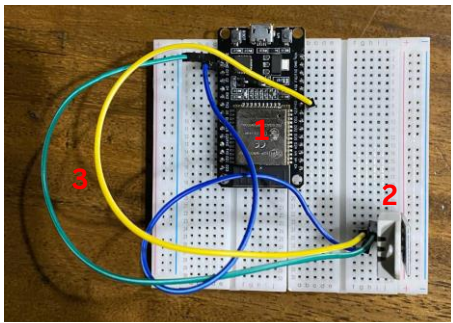
Projek UTS
2023/2024

TUGAS UTS:
“SISTEM BERBASIS MIKROPROSESOR”
SEMESTER GENAP 2023/2024

Ryan Krishandi Lukito (22/497249/TK/54488)
Yefta Nathaniel Wibowo (22/492878/TK/53954)

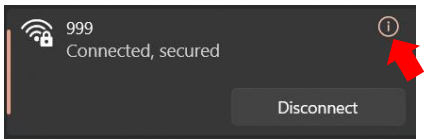
Konfigurasi Rangkaian ESP32 dan DHT-22

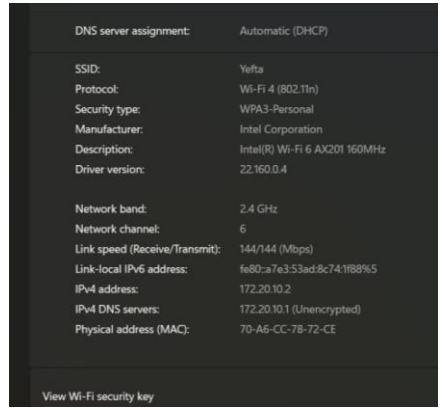
1. ESP32 ditempatkan di bagian mana saja pada breadboard.
2. Sensor DHT-22 diletakkan pada posisi vertikal agar tidak terjadi short circuit. Sensor ini digunakan untuk me-retrieve temperature dan humidity secara realtime.
3. Kabel jumper dipasangkan pada pin 3v3, Ground, dan pin 26 pada ESP32. Kemudian, kabel pin 3v3 disambungkan pada pin positive, kabel Ground disambungkan pada pin negative, dan kabel pin 26 pada pin data dari sensor DHT-22.
4. Sambungkan ESP32 pada laptop dengan menggunakan kabel micro-usb dan siapkan Arduino IDE.



IP Address Laptop dari Access Point

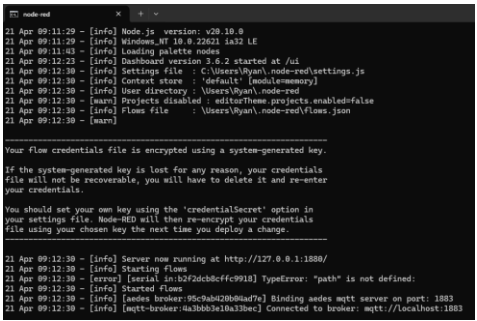
1. Hubungkan laptop anda dengan koneksi wifi yang biasa digunakan
2. Untuk mendapatkan alamat ip, gunakan command prompt dengan memasukkan command “netsh wlan show profile”.
3. Kemudian lihat pada bagian IPv4 Address.
4. IP Address juga bisa didapatkan dengan menekan lambang “i” pada wifi.

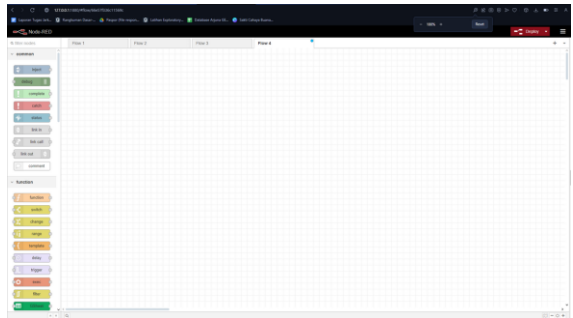




Node Aedes MQTT Broker

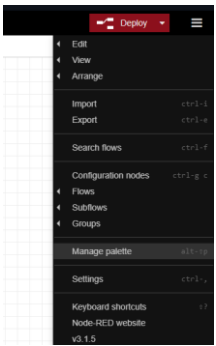
Aktifkan node-red pada command prompt dan buka node-red menggunakan chrome atau search engine lainnya.

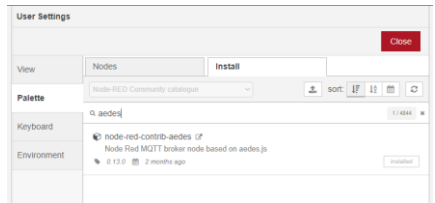




Node Aedes MQTT Broker

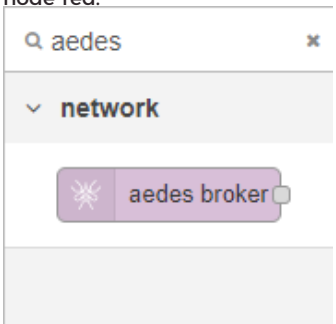
Install Node Aedes MQTT Broker pada **manage palletes**

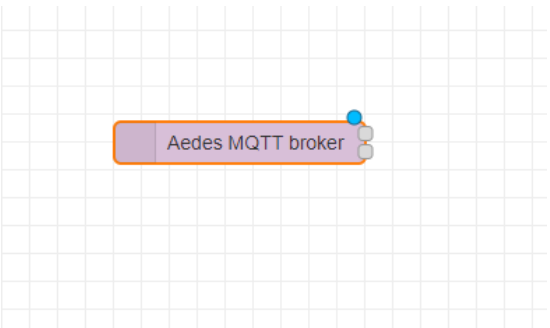




Node Aedes MQTT Broker

Setelah proses instalasi selesai, pilih Aedes MQTT Broker pada filter nodes dan drag ke dalam canvas node-red.





Koneksi Antar DHT-22 dengan ESP32

1. Install library DHT pada Library Manager Arduino IDE.
2. Import library DHT dengan command “#include <DHT.h>” (tanpa tanda petik).
3. Tulis line code di bawah ini untuk mengaktifkan dan mendapatkan bacaan temperature serta humidity dari DHT-22

```
11  DHT sensor_dht(DHT_PIN,DHT22);
```

```
83  float temp = sensor_dht.readTemperature();
84  float hum = sensor_dht.readHumidity();
85
86  doc["Temperature"] = temp;
87  doc["Humidity"] = hum;
88
89  char jsonString[100];
90  serializeJson(doc, jsonString);
91  client.publish("/yefta_sensor/data", jsonString);
92
93  Serial.println("Data Published: ");
94  Serial.println(jsonString);
95
96  delay(2000);
```

sketch_mar22a.ino

```
1  #include <WiFi.h>
2  #include <PubSubClient.h>
3  #include <DHT.h>
4  #include <ArduinoJson.h>
```

Hasil Serial Monitor

Output Serial Monitor ×

Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')

```
Data Published:
{"Temperature":30.60000038,"Humidity":86}
Data Published:
{"Temperature":30.60000038,"Humidity":85.90000153}
Data Published:
{"Temperature":30.60000038,"Humidity":85.90000153}
Data Published:
{"Temperature":30.70000076,"Humidity":85.90000153}
Data Published:
{"Temperature":30.70000076,"Humidity":86.09999847}
```

Koneksi Antar WiFi dengan ESP32

1. Install library WiFi pada Library Manager Arduino IDE.
2. Import library WiFidengan command “#include <WiFi.h>” (tanpa tanda petik).
3. Tulis line code seperti pada slide berikutnya ini untuk mengaktifkan fitur WiFi pada ESP32.

sketch_mar22a.ino

```
1  #include <WiFi.h>
2  #include <PubSubClient.h>
3  #include <DHT.h>
4  #include <ArduinoJson.h>
```

Koneksi Antar WiFi dengan ESP32

```
const int DHT_PIN = 26;
const char* ssid = "Yefta"; /// wifi ssid
const char* password = "12345678";
const char* mqtt_server = "172.20.10.2";// mosquitto server url

DHT sensor_dht(DHT_PIN,DHT22);
WiFiClient espClient;
PubSubClient client(espClient);
unsigned long lastMsg = 0;
float temp = 0;
float hum = 0;

StaticJsonDocument<100> doc;
```

```
void setup_wifi() {
  delay(10);
  Serial.println();
  Serial.print("Wifi terkoneksi ke : ");
  Serial.println(ssid);

  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  randomSeed(micros());

  Serial.println("");
  Serial.println("WiFi berhasil terkoneksi");
  Serial.print("Alamat IP : ");
  Serial.println(WiFi.localIP());
}

void callback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  for (int i = 0; i < length; i++) {
    Serial.print((char)payload[i]);
  }
}

void reconnect() {
  while (!client.connected()) {
    Serial.print("Baru melakukan koneksi MQTT ...");
    String clientId = "ESP32Client-";
    clientId += String(random(0xffff), HEX);
    if (client.connect(clientId.c_str())) {
      Serial.println("Connected");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      delay(5000);
    }
```

```
Output  Serial Monitor ×
Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')

.....
WiFi berhasil terkoneksi
Alamat IP : 172.20.10.8
Baru melakukan koneksi MQTT ...Connected
```

Koneksi Antar MQTT Server dengan ESP32

1. Install library PubSubClient pada Library Manager Arduino IDE.
2. Import library WiFidengan command “#include <PubSubClient.h>” (tanpa tanda petik).
3. Tulis line code di bawah ini untuk mengaktifkan fitur PubSubClient pada ESP32. Command lengkap tersedia pada potongan code berikut.

sketch_mar22a.ino

```
1   #include <WiFi.h>
2   #include <PubSubClient.h>
3   #include <DHT.h>
4   #include <ArduinoJson.h>
```

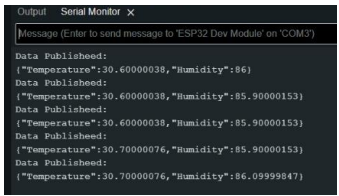
```
13  PubSubClient client(espClient);
```

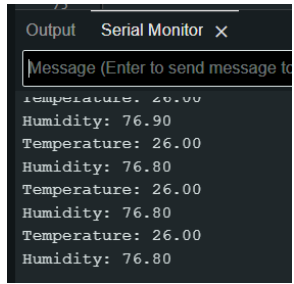
```
50  void reconnect() {
51    while (!client.connected()) {
52      Serial.print("Baru melakukan koneksi MQTT ...");
53      String clientId = "ESP32Client-";
54      clientId += String(random(0xffff), HEX);
55      if (client.connect(clientId.c_str())) {
56        Serial.println("Connected");
57      } else {
58        Serial.print("failed, rc=");
59        Serial.print(client.state());
60        Serial.println(" try again in 5 seconds");
61        delay(5000);
62      }
63    }
64  void setup() {
65    //pinMode(2, OUTPUT);
66    Serial.begin(115200);
67    sensor_dht.begin();
68    setup_wifi();
69    client.setServer(mqtt_server, 1883);
70    client.setCallback(callback);
71
72  }
```

```
73  void loop() {
74    if (!client.connected()) {
75      reconnect();
76    }
77    client.loop();
78
79  unsigned long now = millis();
80  if (now - lastMsg > 2000) { //perintah publish data
81    lastMsg = now;
82
83    float temp = sensor_dht.readTemperature();
84    float hum = sensor_dht.readHumidity();
85
86    doc["Temperature"] = temp;
87    doc["Humidity"] = hum;
88
89    char jsonString[100];
90    serializeJson(doc, jsonString);
91    client.publish("/yefta_sensor/data", jsonString);
92
93    Serial.println("Data Published: ");
94    Serial.println(jsonString);
95
96    delay(2000);
97  }
98  }
```

Koneksi Antar MQTT Server dengan ESP32

1. Berikut data dalam betuk topik dan message secara langsung
2. Berikut data dalam betuk JSON





Sistem Berbasis Mikroprosesor

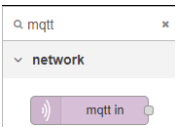
Projek UTS
2023/2024

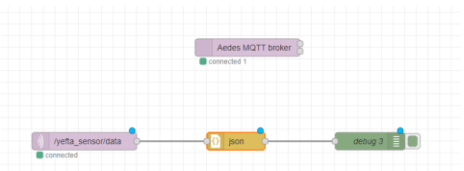
Koneksi Antar MQTT Server dengan ESP32

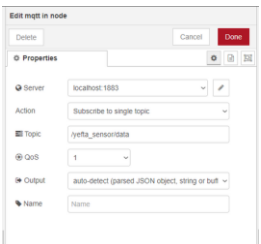
Kode lengkap untuk menghubungkan ESP32 denga Wi-Fi dan output temperature dan humidity dalam bentuk JSON dapat dilihat pada link berikut: https://bit.ly/3wcrSdh

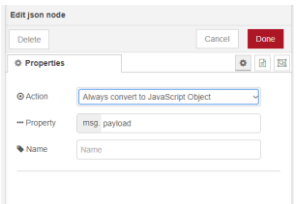
Debug Node-Red dengan MQTT

1. Buatlah rangkaian node pada node-red seperti gambar di samping dengan node aedes, json, dan mqtt in.
2. Jangan lupa atur konfigurasi node seperti di bawah.







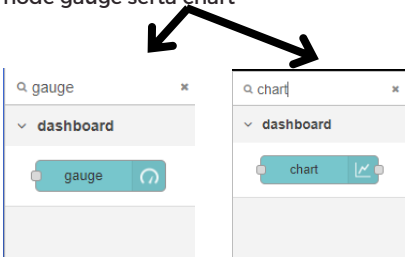


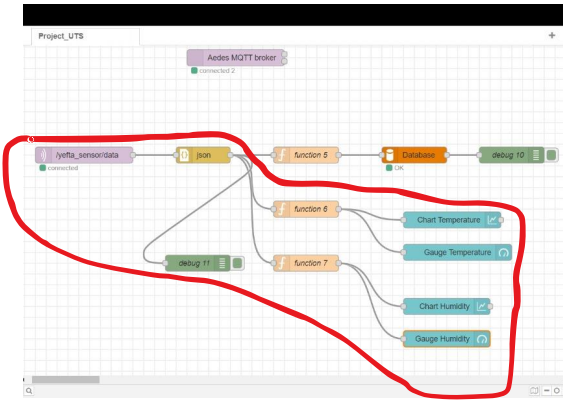
Debug Node-Red dengan MQTT

Klik “Deploy” dan apabila keluaran sudah seperti gambar di bawah, maka proses retrieve data dari ESP32 ke MQTT Node-Red sudah berhasil.

Implementasi Dashboard Gauge dan Chart

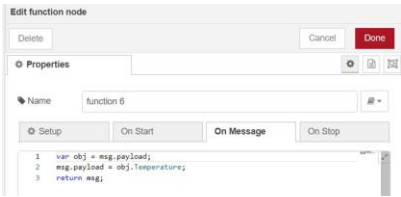
Untuk membuat Gauge dan Chart pada Dashboard Node-red, buatlah rangkaian seperti gambar di samping (bagian yang dilingkari warna merah) dengan tambahan node gauge serta chart



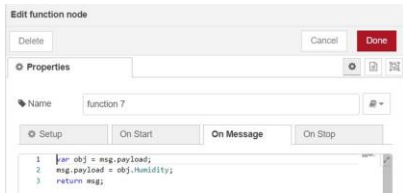


Implementasi Dashboard Gauge dan Chart

Masukkan kode di bawah pada Function 6



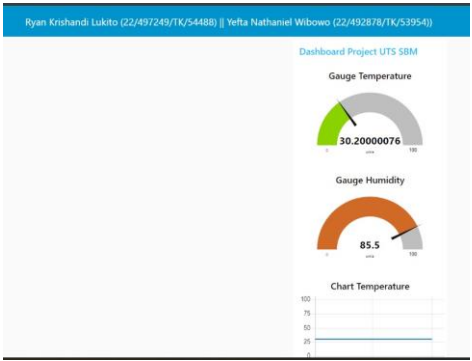
Masukkan kode di bawah pada Function 7

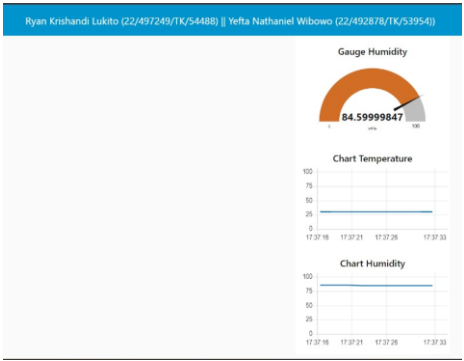


Isi dari node gauge dan chart tidak perlu diubah (kecuali ingin mengubah posisi gauge dan chart).

Implementasi Dashboard Gauge dan Chart

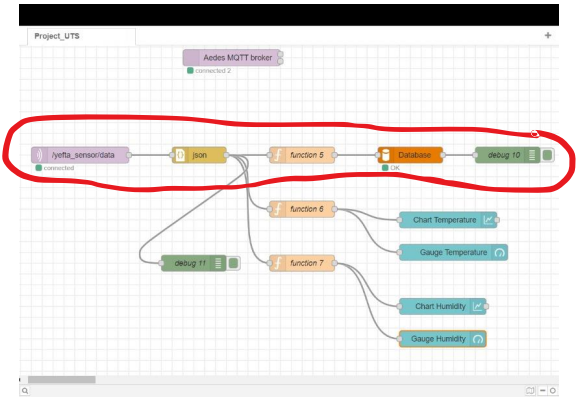
Berikut hasil akhir dari implementasi Gauge dan Chart pada Dashboard Node-red





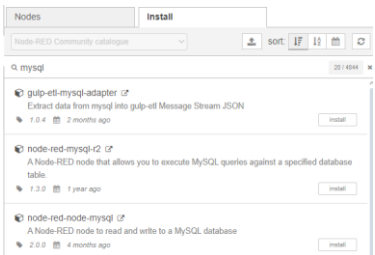
Implementasi MySQL pada Node-red

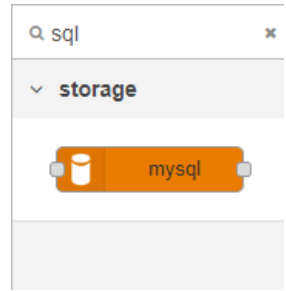
Untuk menambahkan data ke dalam database MySQL melalui node-red, buatlah rangkaian seperti gambar di samping (bagian yang dilingkari warna merah)



Node MySQL untuk Database

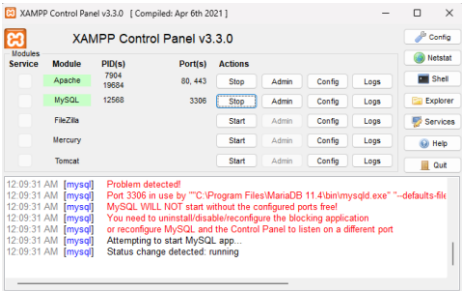
Install node database untuk memasukkan data dari ESP32 ke dalam database. Dalam penugasan ini, database yang digunakan adalah MySQL berbasis xampp.





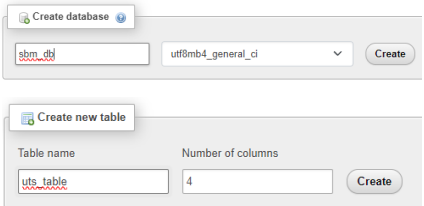
Implementasi MySQL pada Node-red

Install XAMPP dan jalankan Apache serta MySQL. Setelahnya, tulis "localhost/phpmyadmin" pada search engine yang biasa anda gunakan.



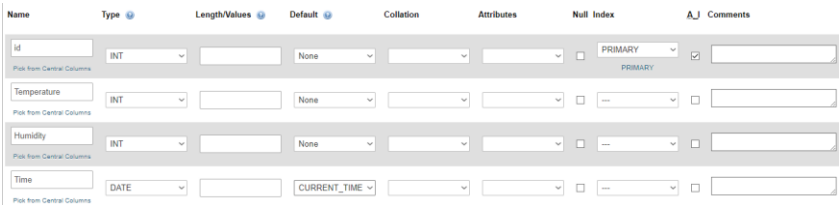
Implementasi MySQL pada Node-red

Di dalam “localhost/phpmyadmin”, buat sebuah database bernama “sbm_db” dan tabel bernama “uts_table”.



Implementasi MySQL pada Node-red

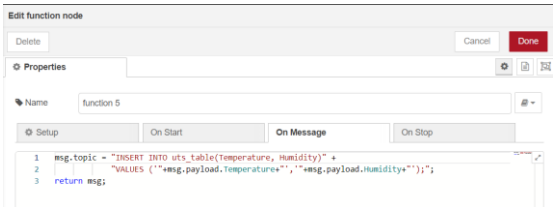
Isikan “uts_table” dengan beberapa atribut seperti id, Temperature, Humidity, dan Time.



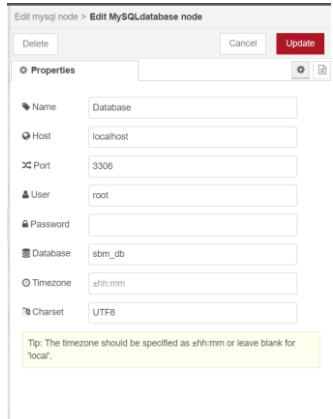
Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I	Comments
id Pick from Central Columns	INT		None			☐	PRIMARY PRIMARY	☑	
Temperature Pick from Central Columns	INT		None			☐	---	☐	
Humidity Pick from Central Columns	INT		None			☐	---	☐	
Time Pick from Central Columns	DATE		CURRENT_TIME			☐	---	☐	

Implementasi MySQL pada Node-red

Masukkan kode di bawah pada node Function 5

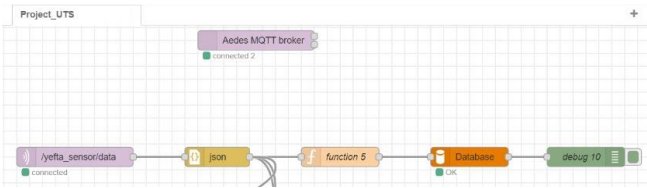


Sesuaikan konfigurasi pada node database



Implementasi MySQL pada Node-red

Berikut rangkaian node untuk menghubungkan MySQL dengan ESP32 pada Node-red



Implementasi MySQL pada Node-red

Klik “Deploy” apabila seluruh konfigurasi nodes sudah sesuai. Kemudian, lihat pesan pada debug. Apabila keluarannya seperti pada gambar disamping, maka data dari DHT-22 sudah berhasil dimasukkan ke dalam MySQL

Implementasi MySQL pada Node-red

Berikut tampilan pada phpmyadmin apabila data dari DHT-22 berhasil di-retrieve ke dalam database MySQL.

Pada bagian “Time” terlihat bahwa data masuk ke dalam database dalam rentang waktu 2 detik (2000ms).