

Smart Monitoring Temperature and Humidity of the Room Server

Disusun oleh:

Candra Wijaya Mj	3311811055
Retno Anggi Syahputri	3311811061
Feren Anindita Salsabila Boru Purba	3311811076

Disusun untuk memenuhi salah satu syarat kelulusan
matakuliah IF418 Proyek Akhir 1



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BATAM
BATAM
2020**

DAFTAR ISI

Halaman Judul	i
Daftar Isi	ii
Daftar Gambar	iii
Daftar Tabel.....	iv
Kata Pengantar	v
 BAB I PENDAHULUAN.....	 1
1.1 Latar Belakang	1
1.2 Manfaat dan Tujuan.....	2
 BAB II SPESIFIKASI KEBUTUHAN DAN ALAT	 3
2.1 NodeMCU ESP8266	3
2.2 Sensor DHT22.....	5
2.3 Grafana	6
2.4 Eclipse Mosquitto.....	6
2.5 Apache	7
2.6 Mysql	8
2.7 Arduino IDE.....	8
2.8 NodeJS	9
2.9 Flowchart	10
 BAB III PENGODINGAN.....	 11
3.1 Script pada NodeMcu ESP 8266.....	113
3.2 Script Mqtt.js.....	15
3.3 Script Bottelegram.js	16
 BAB IV IMPLEMENTASI DAN PENGUJIAN.....	 17
4.1 Implementasi	17
4.2 Pengujian Monitoring Suhu dan Kelembapan	19
4.3 Pengujian Notifikasi Telegram	23
 BAB V KESIMPULAN	 255
5.1 Kesimpulan	255
 Daftar Pustaka	 26

DAFTAR GAMBAR

Gambar 1 Module NodeMCU ESP 8266.....	4
Gambar 2 Sensor DHT22 pins	5
Gambar 3 Flowchart Alur	10
Gambar 4 Alur Pengodingan.....	11
Gambar 5 Script NodeMcu ESP8266	11
Gambar 6 Script NodeMcu ESP8266	12
Gambar 7 Script NodeMcu ESP8266	12
Gambar 8 Script Mqtt.js	13
Gambar 9 Script Mqtt.js	14
Gambar 10 Script Mqtt.js	14
Gambar 11 Script Bottelegram.js	15
Gambar 12 Script Bottelegram.js	16
Gambar 13 Script Grafana.....	16
Gambar 14 Rangkaian Alat.....	17
Gambar 15 Alur Impelentasi Perangkat Keras	18
Gambar 16 Alur Impelentasi Perangkat Lunak	18
Gambar 17 Serial Monitor Arduino IDE	19
Gambar 18 Script Mqtt.js menerima data	19
Gambar 19 Data pada Mysql.....	20
Gambar 20 Data Sources Grafana	20
Gambar 21 Membuat grafik Grafana.....	21
Gambar 22 Grafik Grafana.....	21
Gambar 23 Grafik Grafana.....	22
Gambar 24 Manage User Grafana	22
Gambar 25 History Grafana	23
Gambar 26 Console Log Bottelegram.js.....	24
Gambar 27 Notifikasi Bot Telegram	24

DAFTAR GAMBAR

Tabel 1 Level Suhu Data Center.....	1
-------------------------------------	---

KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas segala rahmat-Nya sehingga laporan yang berjudul “Smart Monitoring Temperature and Humidity of the Room Server” ini dapat tersusun hingga selesai. Laporan ini disusun untuk memenuhi tugas Proyek Akhir 1. Kami mengucapkan terima kasih atas bantuan dari pihak yang telah berkontribusi dalam menyelesaikan laporan ini.

Harapan kami semoga laporan ini dapat menambah pengetahuan dan pengalaman dalam bidang IoT. Untuk kedepannya dapat memperbaiki sistem yang telah dibuat maupun menambah isi laporan agar menjadi lebih baik lagi.

Karena keterbatasan pengetahuan maupun pengalaman kami, kami yakin masih banyak kekurangan dalam laporan ini. Oleh karena itu kami mengharapkan saran dan kritik yang membangun demi kesempurnaan laporan ini.

Batam, 2 Mei 2020

Penyusun,

BAB I PENDAHULUAN

1.1 Latar Belakang

Ruang server adalah ruang yang menyimpan seluruh data perusahaan atau lembaga lainnya secara terpusat. Tetapi ada masalah yang timbul yaitu kondisi panas pada ruang server membuat kinerjanya menjadi lambat bahkan bisa membuat server mati. Oleh karena itu diperlukan sistem IoT (Internet of Things) untuk dapat memantau temperatur dan kelembapan pada ruang server.

Smart Monitoring Temperature and Humidity of The Room Server adalah sistem yang dapat memantau suhu dan kelembapan pada ruang server. Sistem tersebut merupakan sistem yang menggunakan perangkat IoT (Internet of Things) yaitu NodeMCU ESP8266 dan sensor suhu dan kelembapan ruang yaitu Dht22, yang berfungsi untuk membaca data kondisi suhu dan kelembapan pada ruang server. Kemudian mengirimkan data tersebut di dalam database. Sistem ini juga dapat memberikan notifikasi secara otomatis apabila data suhu di ruang server melebihi batas suhu yang telah ditentukan melalui Telegram.

Standard suhu ruang server pada umumnya adalah 18 sampai 27°C. Berikut adalah table rekomendasi level suhu data center yang ada dunia.

Data Center	Rendah (°C)	Tinggi (°C)	Optimal
Cisco	18	27	-
Dell	24	26	-
HP	18	27	-
IBM	18	27	-
Oracle	21	23	22

Tabel 1. Level Suhu Data Center

1.2 Manfaat dan Tujuan

Rumusan Adapun tujuan dari proyek akhir ini yaitu merancang dan membangun “Smart Monitoring Temperature and Humidity of The Room Server” yang dapat :

- Merancang dan membangun Sistem Smart Monitoring Temperature and Humidity of the Room Server.
- Memantau suhu dan kelembapan pada ruang server menggunakan perangkat IoT.
- Memberikan Notifikasi Ketika suhu sudah mencapai batas maksimal yang ditentukan.

BAB II

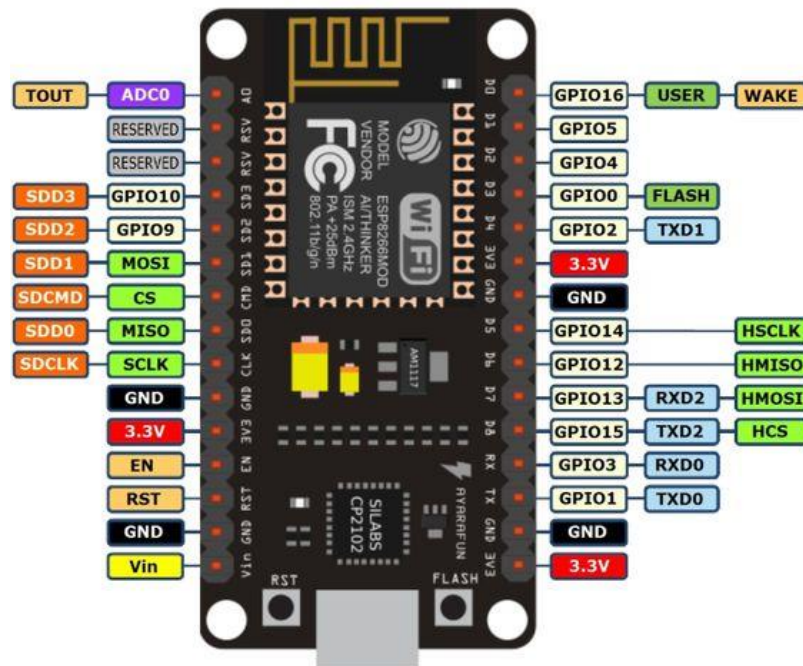
SPESIFIKASI KEBUTUHAN DAN ALAT

2.1 NodeMCU ESP8266

NodeMCU merupakan sebuah open source platform IoT dan pengembangan yang menggunakan bahasa pemrograman Lua untuk membantu dalam membuat prototype produk IoT atau bisa dengan memakai sketch dengan arduino IDE. NodeMCU berukuran panjang 4.83cm, lebar 2.54cm, dan berat 7 gram. Board ini sudah dilengkapi dengan fitur WiFi dan Firmwarenya yang bersifat opensource.

Spesifikasi yang dimiliki oleh NodeMCU sebagai berikut :

1. Board ini berbasis ESP8266 serial WiFi SoC (Single on Chip) dengan onboard USB to TTL. Wireless yang digunakan adalah IEEE 802.11b/g/n.
2. Tantalum capacitor 100 micro farad dan 10 micro farad.
3. 3.3v LDO regulator.
4. Blue led sebagai indikator.
5. Cp2102 usb to UART bridge.
6. Tombol reset, port usb, dan tombol flash.
7. Terdapat 9 GPIO yang di dalamnya ada 3 pin PWM, 1 x ADC Channel, dan pin RX TX
8. 3 pin ground.
9. S3 dan S2 sebagai pin GPIO
10. S1 MOSI (Master Output Slave Input) yaitu jalur data dari master dan masuk ke dalam slave, sc cmd/sc.
11. S0 MISO (Master Input Slave Input) yaitu jalur data keluar dari slave dan masuk ke dalam master.
12. SK yang merupakan SCLK dari master ke slave yang berfungsi sebagai clock.
13. Pin Vin sebagai masukan tegangan.
14. Built in 32-bit MCU.



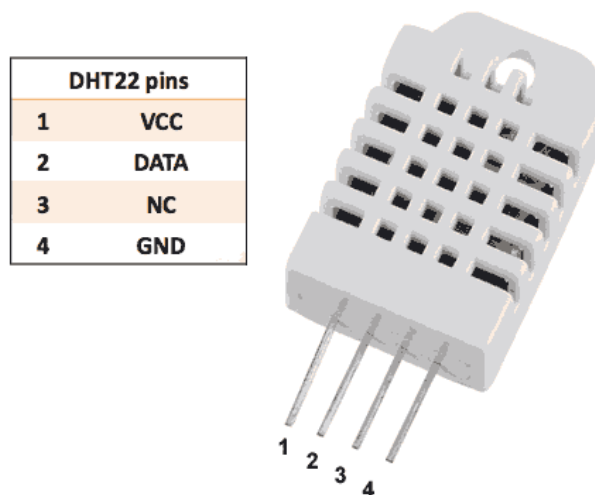
Gambar 1. Module NodeMCU ESP 8266

1. RST : berfungsi mereset modul
2. ADC: Analog Digital Converter. Rentang tegangan masukan 0-1v, dengan skup nilai digital 0-1024
3. EN: Chip Enable, Active High
4. IO16 :GPIO16, dapat digunakan untuk membangunkan chipset dari mode deep sleep
5. IO14 : GPIO14; HSPI_CLK
6. IO12 : GPIO12: HSPI_MISO
7. IO13: GPIO13; HSPI_MOSI; UART0_CTS
8. VCC: Catu daya 3.3V (VDD)
9. CS0 :Chip selection
10. MISO : Slave output, Main input
11. IO9 : GPIO9
12. IO10 GBIO10
13. MOSI: Main output slave input
14. SCLK: Clock

- 15. GND: Ground
- 16. IO15: GPIO15; MTDO; HSPICS; UART0_RTS
- 17. IO2 : GPIO2; UART1_TXD
- 18. IO0 : GPIO0
- 19. IO4 : GPIO4
- 20. IO5 : GPIO5
- 21. RXD : UART0_RXD; GPIO3
- 22. TXD : UART0_TXD; GPIO1

2.2 Sensor DHT22

DHT-22 atau AM2302 adalah sensor suhu dan kelembaban, sensor ini memiliki keluaran berupa sinyal digital dengan konversi dan perhitungan dilakukan oleh MCU 8-bit terpadu. Sensor tersebut memiliki kalibrasi akurat dengan kompensasi suhu ruang penyesuaian dengan nilai koefisien tersimpan dalam memori OTP terpadu. Sensor DHT22 memiliki rentang pengukuran suhu dan kelembaban yang luas, DHT22 mampu mentransmisikan sinyal keluaran melewati kabel hingga 20 meter sehingga sesuai untuk ditempatkan di mana saja, tetapi jika kabel yang panjang di atas 2 meter harus ditambahkan buffer capacitor 0,33 μ F antara pin#1 (VCC) dengan pin#4 (GND).



Gambar 2. Sensor DHT22 pins

Spesifikasi Teknis DHT22 / AM-2302 yaitu:

1. Catu daya: 3,3 - 6 Volt DC (tipikal 5 VDC)
2. Sinyal keluaran: digital lewat bus tunggal dengan kecepatan 5 ms/operasi
3. Elemen pendeteksi: kapasitor polimer (polymer capacitor)
4. Jenis sensor: kapasitif (capacitive sensing)
5. Rentang deteksi kelembapan : 0-100% RH (akurasi $\pm 2\%$ RH)
6. Rentang deteksi suhu : -40° - $+80^{\circ}$ Celcius (akurasi $\pm 0,5^{\circ}\text{C}$)
7. Resolusi sensitivitas : 0,1%RH; 0,1 $^{\circ}\text{C}$
8. Histeresis kelembaban: $\pm 0,3\%$ RH
9. Stabilitas jangka panjang: $\pm 0,5\%$ RH / tahun
10. Periode pemindaian rata-rata: 2 detik
11. Ukuran: 25,1 x 15,1 x 7,7 mm
12. Hubungkan pin#2 (data) dari sensor ini dengan pin Digital I/O pada MCU (Microcontroller Unit).

2.3 Grafana

Grafana adalah multi-platform open source analytics dan perangkat lunak visualisasi interaktif yang tersedia sejak 2014. Grafana menyediakan grafik, grafik, dan peringatan untuk web ketika terhubung ke sumber data yang didukung dapat diperluas melalui sistem plug-in . Pengguna akhir dapat membuat dashboard pemantauan yang kompleks menggunakan pembuat kueri interaktif.

Sebagai alat visualisasi, Grafana adalah komponen yang populer dalam memonitoring, sering digunakan dalam kombinasi dengan basis data deret waktu seperti Prometheus dan Graphite, platform pemantauan seperti Sensu, Icinga, Zabbix , Netdata, PRTG, dan sumber data lainnya.

2.4 Eclipse Mosquitto

Eclipse Mosquitto adalah broker pesan open source (berlisensi EPL / EDL) yang mengimplementasikan protokol MQTT versi 5.0, 3.1.1 dan 3.1. Mosquitto

ringan dan cocok untuk digunakan pada semua perangkat mulai dari komputer papan tunggal berdaya rendah hingga server penuh.

Protokol MQTT menyediakan metode ringan untuk melakukan pengiriman pesan menggunakan model publikasikan atau berlangganan untuk perpesanan Internet of Things seperti dengan sensor daya rendah atau perangkat seluler seperti telepon, komputer tertanam atau mikrokontroler.

2.5 Apache

Apache merupakan web server yang paling banyak dipergunakan di Internet. Program ini pertama kali didesain untuk sistem operasi lingkungan UNIX. Namun demikian, pada beberapa versi berikutnya Apache mengeluarkan programnya yang dapat dijalankan di Windows NT.

Berdasarkan sejarahnya, Apache dimulai oleh veteran developer NCSA httpd (National Center for Supercomputing Application). Saat itu pengembangan NCSA httpd sebagai web server mengalami stagnasi. ROB MC COOL meninggalkan NCSA dan memulai sebuah proyek baru bersama para webmaster lainnya, menambal bug, dan menambahkan fitur pada NCSA httpd. Mereka mengembangkan program ini lewat mailing list. Dengan berpijak pada NCSA httpd versi 1.3, Team Apache mengeluarkan rilis pertama kali secara resmi Apache versi 0.6.2.

Beberapa dukungan Apache yaitu :

1. Kontrol Akses. Kontrol ini dapat dijalankan berdasarkan nama host atau nomor IP
2. CGI (Common Gateway Interface), yang paling terkenal untuk digunakan adalah perl (Practical Extraction and Report Language)
3. PHP (Personal Home Page/PHP Hypertext Processor); program dengan metode semacam CGI, yang memproses teks dan bekerja di server. Apache mendukung PHP dengan menempatkannya sebagai salah satu modulnya (mod_php). Hal ini membuat kinerja PHP menjadi lebih baik
4. SSI (Server Side Includes)

2.6 Mysql

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multialur, multipengguna, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU General Public License (GPL), tetapi ada juga lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL.

2.7 Arduino IDE

IDE (Integrated Development Environment) atau secara bahasa merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui software tersebut Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dinamakan melalui sintaks pemrograman. Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino (Sketch) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. IC mikrokontroler Arduino telah ditanamkan suatu program bernama Bootlader yang berfungsi sebagai penengah antara compiler Arduino dengan mikrokontroler. IDE yang diperuntukkan untuk membuat perintah atau source code, melakukan pengecekan kesalahan, kompilasi, upload program, dan menguji hasil kerja arduino melalui serial monitor.

Menu menu yang ada pada sketch Arduino IDE yaitu:

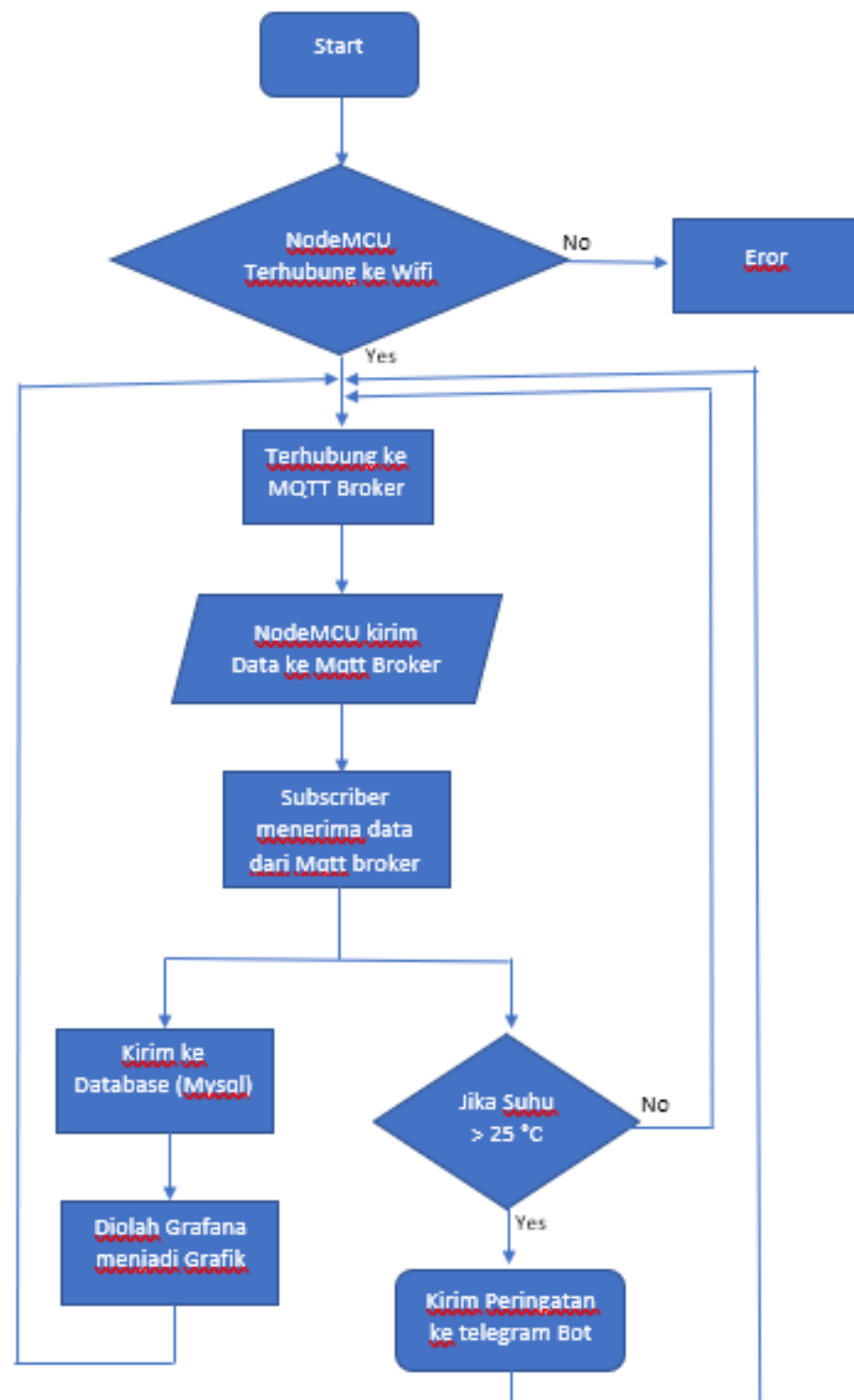
1. Verify berfungsi untuk melakukan checking kode yang dibuat apakah sudah sesuai dengan kaidah pemrograman yang ada atau belum.
2. Upload berfungsi untuk melakukan kompilasi program atau kode yang dibuat menjadi bahasa yang dapat dipahami oleh mesin atau Arduino.
3. New berfungsi untuk membuat Sketch baru.
4. Open berfungsi untuk membuka sketch yang pernah dibuat dan membuka kembali untuk dilakukan editing atau upload ulang ke Arduino.

5. Save berfungsi untuk menyimpan Sketch yang telah dibuat.
6. Serial Monitor berfungsi untuk membuka serial monitor. Serial monitor merupakan jendela yang menampilkan data yang dikirimkan atau dipertukarkan antara arduino dengan sketch pada port serialnya.
7. File

2.8 NodeJS

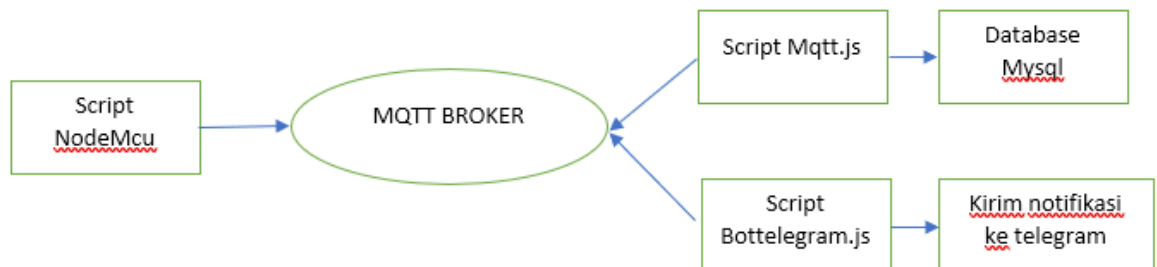
Node.js adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis web dan ditulis dalam sintaks bahasa pemrograman JavaScript. JavaScript dikenal sebagai bahasa pemrograman yang berjalan di sisi client / browser, maka adanya Node.js untuk melengkapi peran JavaScript sehingga berlaku sebagai bahasa pemrograman yang berjalan di sisi server, seperti halnya PHP, Ruby, Perl, dan sebagainya. Node.js dapat berjalan di sistem operasi Windows, Mac OS X dan Linux tanpa perlu ada perubahan kode program. Node.js memiliki pustaka server HTTP sendiri sehingga memungkinkan untuk menjalankan server web tanpa menggunakan program server web seperti Apache atau Nginx.

2.9 Flowchart



Gambar 3. Flowchart Alur

BAB III PENGODINGAN



Gambar 4. Alur Pengodingan

3.1 Script pada NodeMcu ESP 8266

```

_01.temp_sensor_01
#include <Adafruit_MQTT.h>
#include <Adafruit_MQTT_Client.h>

// Code based on
// https://github.com/adafruit/DHT-sensor-library/blob/master/examples/DHTtester/DHTtester.ino
// https://gist.github.com/igrr/7f7e7973366fc01d6393
// https://github.com/iot-playground/Arduino/blob/master/ESP8266ArduinoIDE/DS18B20_temperature_sensor/DS18B20_temperature_sensor.ino

// esp8266 + dht22 + mqtt

#include "DHT.h"
#include <PubSubClient.h>
#include <ESP8266WiFi.h>

const char* ssid = "CANDRA";
const char* password = "Alaskakil2";

char* topic = "sensor";
char* server = "192.168.1.11";
char* hellotopic = "hello_topic";

#define DHTPIN 14 // what pin we're connected to
#define DHTTYPE DHT22 // DHT 22 (AM2302)
#define REPORT_INTERVAL 30 // in sec

String clientName;
DHT dht(DHTPIN, DHTTYPE, 14);
WiFiClient wifiClient;
PubSubClient client(server, 1883, wifiClient);

float oldH ;
  
```

Gambar 5. Script NodeMcu ESP8266

Script pada gambar 5 berfungsi untuk memasukkan library mqtt, sensor dht, koneksi ke wifi, alamat ip mqtt broker dan topic mqtt.


```

_01.temp_sensor_01
void loop() {

    float h = dht.readHumidity();
    float t = dht.readTemperature();
    float f = dht.readTemperature(true);

    if (isnan(h) || isnan(t) || isnan(f)) {
        Serial.println("Failed to read from DHT sensor!");
        return;
    }

    float hi = dht.computeHeatIndex(f, h);

    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.print(" %\t");
    Serial.print("Temperature: ");
    Serial.print(t);
    Serial.print(" *C ");
    Serial.print(f);
    Serial.print(" *F\t");
    Serial.print("Heat index: ");
    Serial.print(hi);
    Serial.println(" *F");

    String payload = "";
    payload += h;
    payload += ",";
    payload += t;
    payload += "\n";
}

```

Gambar 6. Script NodeMcu ESP8266

Fungsi Script pada gambar 6 berfungsi untuk membaca suhu dan kelembapan pada sensor.

```

_01.temp_sensor_01
...

if (t != oldT || h != oldH )
{
    sendTemperature(payload);
    oldT = t;
    oldH = h;
}

int cnt = REPORT_INTERVAL;

while (cnt--)
    delay(1000);
}

void sendTemperature(String payload) {
    if (!client.connected()) {
        if (client.connect((char*) clientName.c_str())) {
            Serial.println("Connected to MQTT broker again");
            Serial.print("Topic is: ");
            Serial.println(topic);
        }
        else {
            Serial.println("MQTT connect failed");
            Serial.println("Will reset and try again...");
            abort();
        }
    }

    if (client.connected()) {
        Serial.print("Sending payload: ");
    }
}

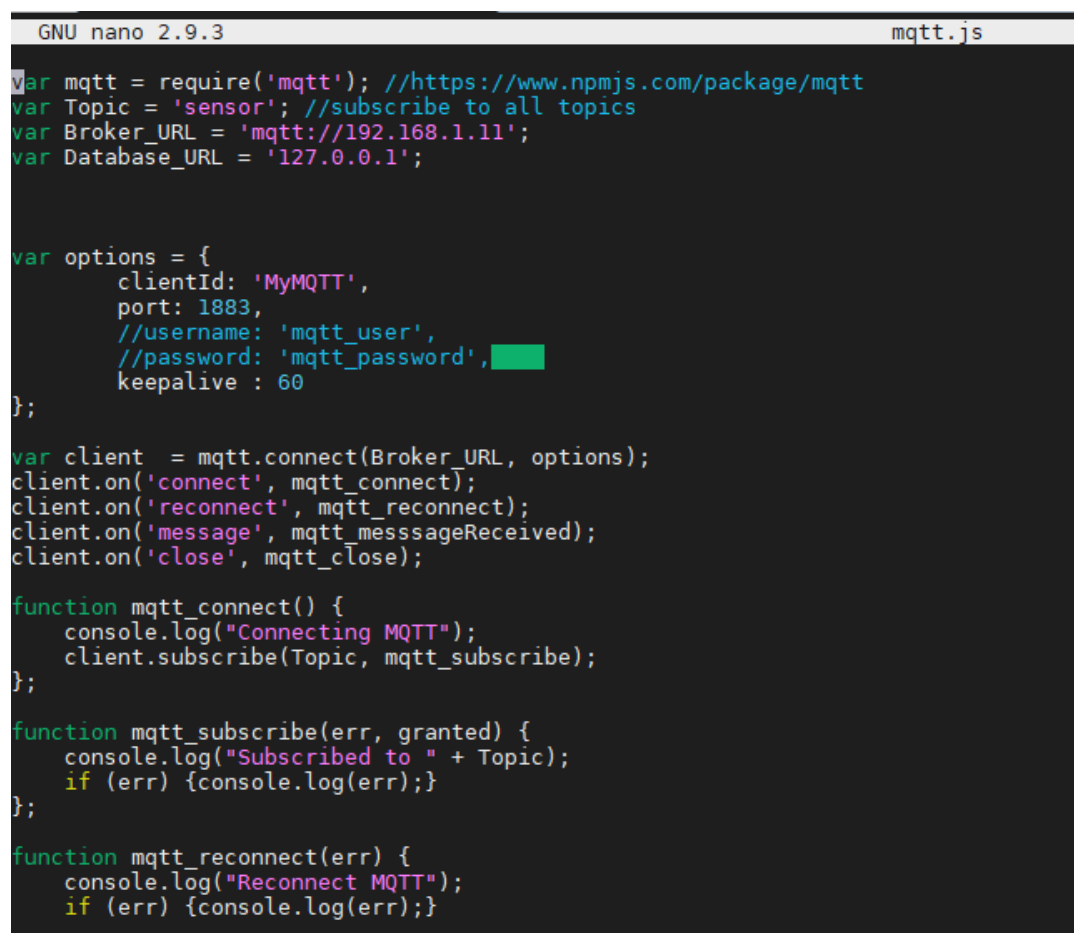
```

Gambar 7. Script NodeMcu ESP826

Fungsi Script pada gambar 7 berfungsi untuk mengirim data sensor yang sudah dibaca ke Mqtt Broker.

3.2 Script Mqtt.js

Script di bawah ini berfungsi untuk subscribe topic pada mqtt broker, gunanya untuk mendapatkan data suhu dan kelembapan yang dikirim oleh NodeMcu tadi, dan data langsung dikirim ke database mysql, Script ini berjalan pada server.



```
var mqtt = require('mqtt'); //https://www.npmjs.com/package/mqtt
var Topic = 'sensor'; //subscribe to all topics
var Broker_URL = 'mqtt://192.168.1.11';
var Database_URL = '127.0.0.1';

var options = {
  clientId: 'MyMQTT',
  port: 1883,
  //username: 'mqtt_user',
  //password: 'mqtt_password',
  keepalive : 60
};

var client = mqtt.connect(Broker_URL, options);
client.on('connect', mqtt_connect);
client.on('reconnect', mqtt_reconnect);
client.on('message', mqtt_messageReceived);
client.on('close', mqtt_close);

function mqtt_connect() {
  console.log("Connecting MQTT");
  client.subscribe(Topic, mqtt_subscribe);
};

function mqtt_subscribe(err, granted) {
  console.log("Subscribed to " + Topic);
  if (err) {console.log(err);}
};

function mqtt_reconnect(err) {
  console.log("Reconnect MQTT");
  if (err) {console.log(err);}
}
```

Gambar 8. Script Mqtt.js

Script pada gambar 8 untuk memasukkan library Mqtt dan membuat fungsi connect ke Mqtt broker

```

GNU nano 2.9.3                                mqtt.js
//receive a message from MQTT broker
function mqtt_messageReceived(topic, message, packet) {
    var message_str = message.toString(); //convert byte array to string
    console.log("message to string", message_str);
    message_str = message_str.replace(/\n$/, ''); //remove new line
    //message_str = message_str.toString().split(",");
    console.log("message to params array", message_str);
    //payload syntax: clientID,topic,message
    if (message_str.length == 0) {
        console.log("Invalid payload");
    } else {
        insert_message(topic, message_str, packet);
        //console.log(message_arr);
    }
};

function mqtt_close() {
    //console.log("Close MQTT");
};

////////////////////////////////////
//////////////////////////////////// MYSQL //////////////////////////////////
////////////////////////////////////
var mysql = require('mysql'); //https://www.npmjs.com/package/mysql
//Create Connection
var connection = mysql.createConnection({
    host: '127.0.0.1',
    user: "candra",
    password: "candramj123",
    database: "PA"
});

connection.connect(function(err) {
    if (err) throw err;

```

Gambar 9. Script Mqtt.js

Script pada gambar 9 untuk menerima data dari Mqtt Broker, dan dibawahnya script untuk koneksi ke database

```

connection.connect(function(err) {
    if (err) throw err;
    //console.log("Database Connected!");
});

//insert a row into the tbl_messages table
function insert_message(topic, message_str, packet) {
    var message_arr = extract_string(message_str); //split a string into an array
    var hum= message_arr[0];
    var temp = message_arr[1];
    var date_insert= new Date();
    var sql = "INSERT INTO ?? (?,?,?,?) VALUES (?,?,,?)";
    var params = ['grafana', 'hum', 'temp', 'topic', 'date_insert', hum, temp, topic, date_insert];
    sql = mysql.format(sql, params);

    connection.query(sql, function (error, results) {
        if (error) throw error;
        console.log("Message added: " + message_str);
    });
};

//split a string into an array of substrings
function extract_string(message_str) {
    var message_arr = message_str.split(","); //convert to array
    return message_arr;
};

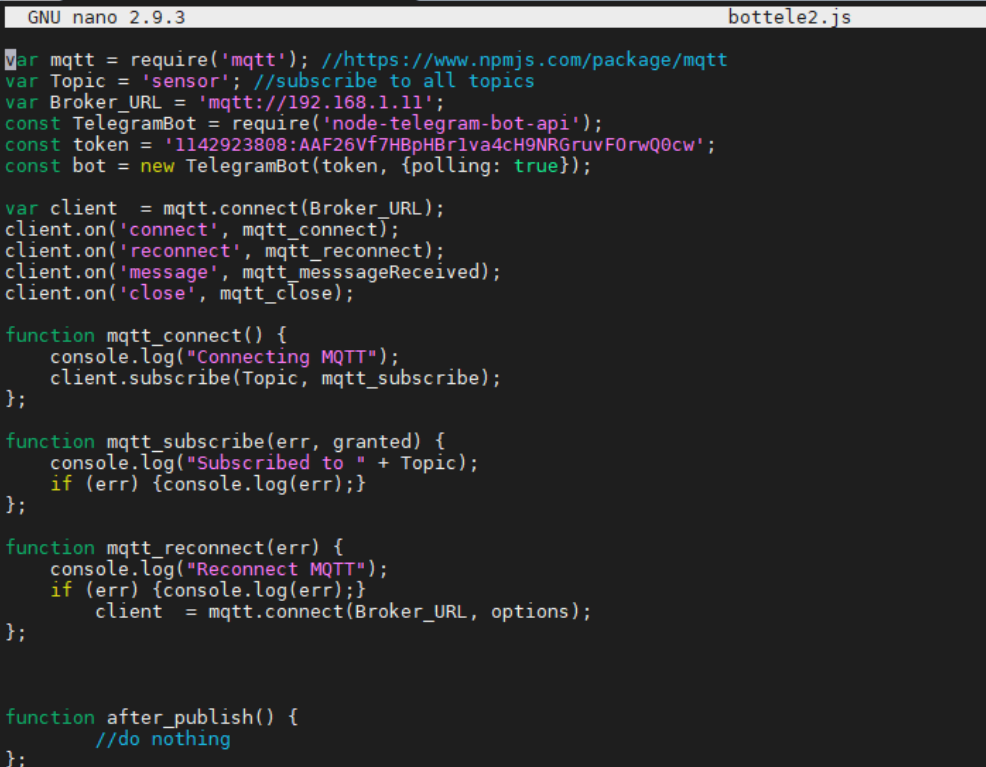
```

Gambar 10. Script Mqtt.js

Script pada gambar 10 untuk mengirim data ke database, data akan diubah menjadi array terlebih dahulu dan dikirim ke database.

3.3 Sript Bottelegram.js

Pada script ini untuk notifikasi peringatan yang akan dikirim ke Telegram, bisa ke Telegram Pribadi atau pun grub, jika suhu sudah mencapai batas yang ditentukan maka notifikasi akan dikirim oleh bot telegram yang sudah dibuat.



```
GNU nano 2.9.3 bottele2.js
var mqtt = require('mqtt'); //https://www.npmjs.com/package/mqtt
var Topic = 'sensor'; //subscribe to all topics
var Broker_URL = 'mqtt://192.168.1.11';
const TelegramBot = require('node-telegram-bot-api');
const token = '1142923808:AAF26Vf7HBpHBr1va4cH9NRGruvF0rwQ0cw';
const bot = new TelegramBot(token, {polling: true});

var client = mqtt.connect(Broker_URL);
client.on('connect', mqtt_connect);
client.on('reconnect', mqtt_reconnect);
client.on('message', mqtt_messageReceived);
client.on('close', mqtt_close);

function mqtt_connect() {
  console.log("Connecting MQTT");
  client.subscribe(Topic, mqtt_subscribe);
};

function mqtt_subscribe(err, granted) {
  console.log("Subscribed to " + Topic);
  if (err) {console.log(err);}
};

function mqtt_reconnect(err) {
  console.log("Reconnect MQTT");
  if (err) {console.log(err);}
  client = mqtt.connect(Broker_URL, options);
};

function after_publish() {
  //do nothing
};
```

Gambar 11. Script Bottelegram.js

Script pada gambar 11 untuk memasukkan library Mqtt dan bot telegram dan juga fungsi untuk connect ke Mqtt broker.

```

GNU nano 2.9.3      bottele2.js
};

//receive a message from MQTT broker
function mqtt_messageReceived(topic, message, packet) {
  var message_str = message.toString(); //convert byte array to string
  console.log(message.toString());
  var message_temp = message_str.slice(6, 8);
  console.log("Temp : ", message_temp);
  var d = new Date();
  var n = d.getHours();
  var m = d.getMinutes();

  if (n == 10 && m == 1){
    bot.sendMessage(316190685, "Notif everyday, Server Room Temperature is " + message_temp + ' C' );
  }else if (n == 10 && m == 3){
    bot.sendMessage(316190685, "Notif everyday, Server Room Temperature is " + message_temp + ' C' );
  }

  if (message_temp == 33){
    bot.sendMessage(316190685, "Your Server Room Temperature is " + message_temp + ' C' );
  }else if (message_temp >= 35){
    bot.sendMessage(316190685, "Warning Your Server Room Temperature is " + message_temp + ' C' );
  }
};

function mqtt_close() {
  //console.log("Close MQTT");
};

```

Gambar 12. Script Bottelegram.js

Script pada gambar 12 untuk menerima data dari Mqtt broker dan langsung di kirim ke bot telegram jika memenuhi kondisi yang sudah ditentukan.

3.4 Script Pada Grafana

```

A
SELECT
  UNIX_TIMESTAMP(date_insert) as time_sec,
  temp as value,
  'temp' as metric
FROM grafana
WHERE $__timeFilter(date_insert)
ORDER BY date_insert ASC

```

Format as Time series Query Builder Show Help Generated SQL

Gambar 13. Script Grafana

Script Pada Gambar 13 untuk membuat data yang ada didalam database menjadi grafik.

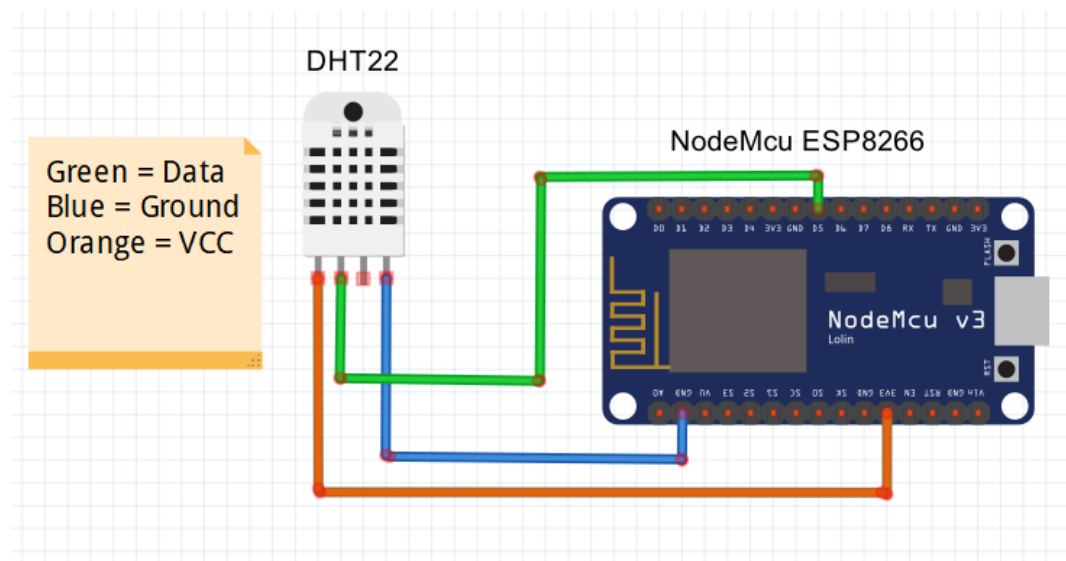
BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi

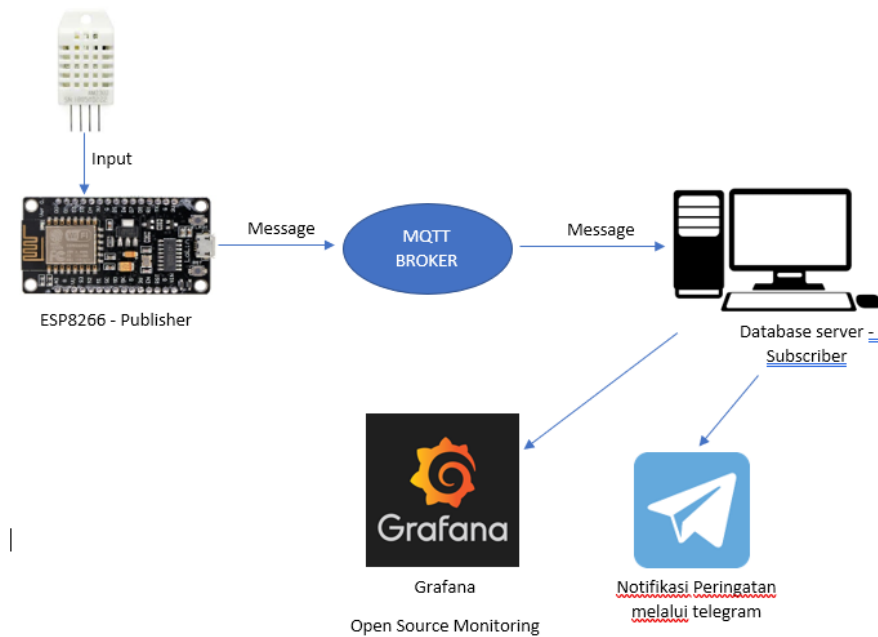
Pembahasan Implementasi merupakan kegiatan akhir dari proses penerapan sistem baru, dimana tahap ini merupakan tahap dimana sistem siap untuk dioperasikan dan dapat dipandang sebagai usaha untuk mewujudkan sistem yang telah dirancang.

a) Implementasi Perangkat Keras



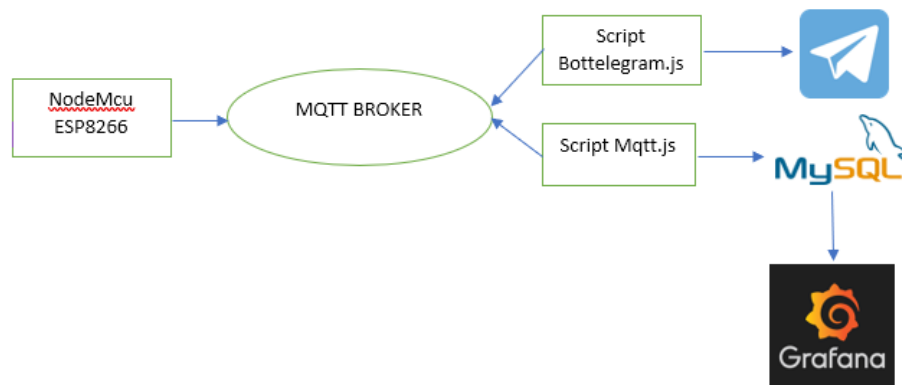
Gambar 14. Rangkaian Alat

Pada gambar 14 terdapat NodeMcu ESP8266 dan DHT22 sensor suhu dan kelembapan, agar NodeMcu bisa menerima data sensor dari DHT22 harus menghubungkan kedua alat tersebut dengan kabel jumper dan mencocokkan masing-masing port sesuai dengan fungsinya seperti di gambar 14 . Kabel warna orange adalah VCC atau tegangan 3,3 volt, kabel warna hijau adalah data sensor dan kabel warna biru adalah ground atau netral.



Gambar 15. Alur Impelentasi Perangkat Keras

b) Implementasi Perangkat Lunak



Gambar 16. Alur Impelentasi Perangkat Lunak

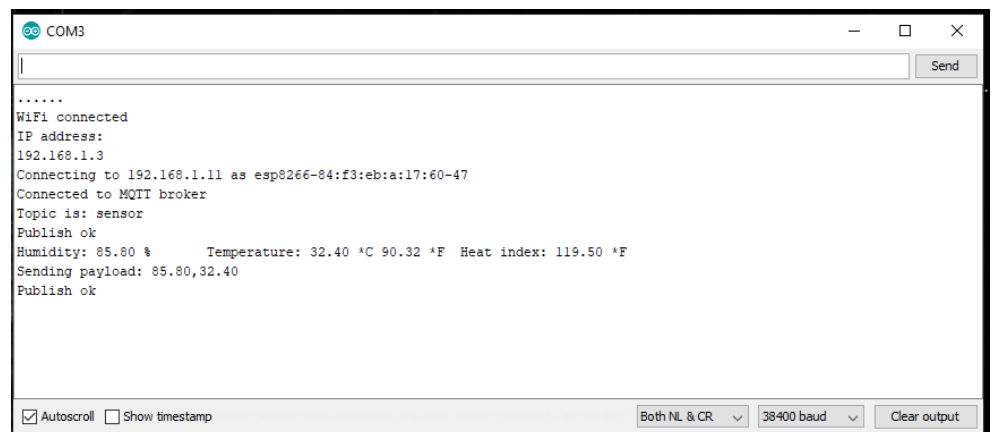
Pada gambar 16 terdapat alur perangkat lunak yang berjalan, data yang berjalan akan seperti gambar di atas, dari NodeMCU mengirim ke Mqtt Broker, selanjutnya Script Bottelegram.js dan script Mqtt.js yang berjalan pada server akan menerima data dari Mqtt Broker. Pada Script Bottelegram

terdapat pengkondisian apabila suhu mencapai batas yang ditentukan akan mengirim notifikasi ke Telegram. Pada Script Mqtt.js data akan dikirim ke Mysql dan diubah menjadi Grafik Oleh Grafana.

4.2 Pengujian Monitoring Suhu dan Kelembapan

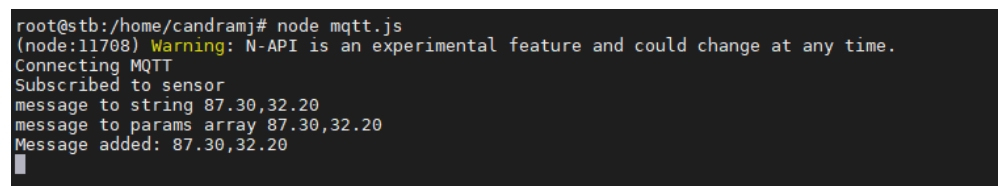
Pada pengujian ini akan melihat proses data dari sensor sampai ke grafana yaitu:

1. Pada Arduino IDE, Setelah upload Script NodeMcu ESP8266 maka dapat melihat data yang diperoleh dari sensor DHT22 dan dikirim ke Mqtt Broker dengan Topic “sensor”.

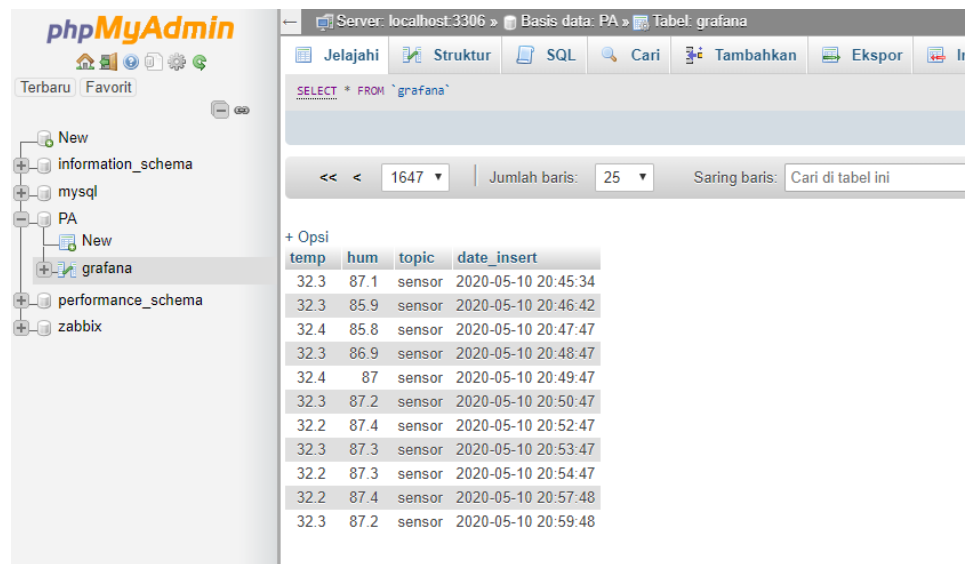


Gambar 17. Serial Monitor Arduino IDE

2. Setelah data sampai ke Mqtt Broker dengan Topic “sensor”, maka script Mqtt.js yang berjalan di server akan menerima data tersebut, karena script Mqtt.js juga mengikuti topic “sensor” dan data akan langsung dikirim ke Database Mysql, seperti gambar di bawah



Gambar 18. Script Mqtt.js menerima data



Server: localhost:3306 » Basis data: PA » Tabel: grafana

SELECT * FROM `grafana`

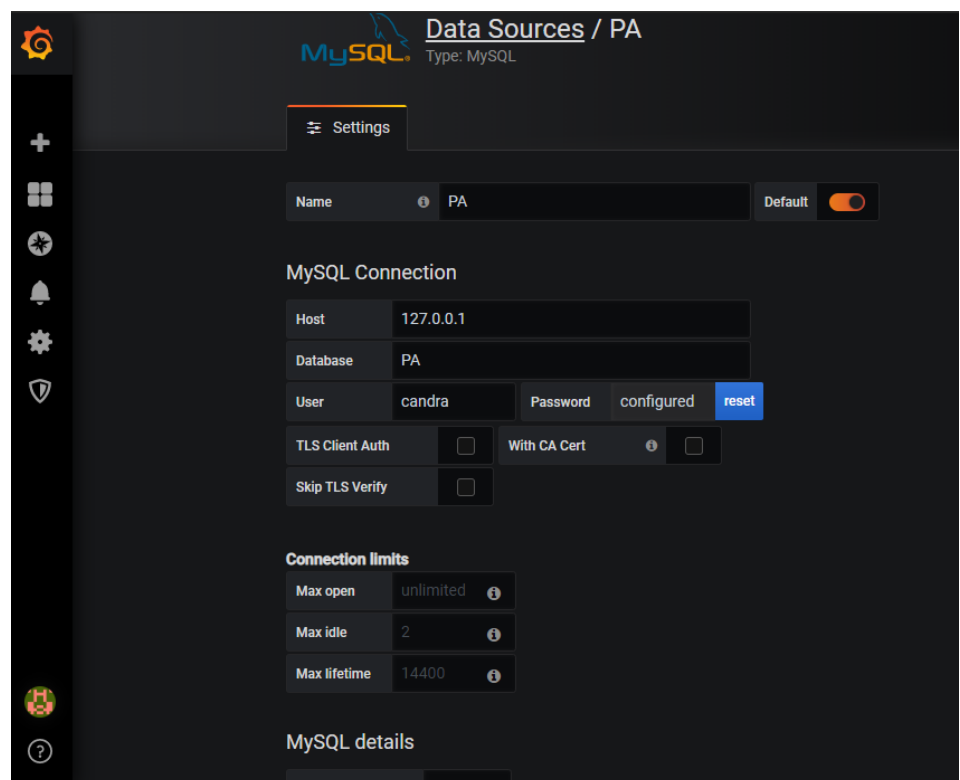
<< < 1647 Jumlah baris: 25 Saring baris: Cari di tabel ini

+ Opsi

temp	hum	topic	date_insert
32.3	87.1	sensor	2020-05-10 20:45:34
32.3	85.9	sensor	2020-05-10 20:46:42
32.4	85.8	sensor	2020-05-10 20:47:47
32.3	86.9	sensor	2020-05-10 20:48:47
32.4	87	sensor	2020-05-10 20:49:47
32.3	87.2	sensor	2020-05-10 20:50:47
32.2	87.4	sensor	2020-05-10 20:52:47
32.3	87.3	sensor	2020-05-10 20:53:47
32.2	87.3	sensor	2020-05-10 20:54:47
32.2	87.4	sensor	2020-05-10 20:57:48
32.3	87.2	sensor	2020-05-10 20:59:48

Gambar 19. Data pada Mysql

- Selanjutnya Pada Grafana ditambahkan dulu Data Sources Mysql dan disetting sesuai dengan Mysql yang akan digunakan.



MLSQL Data Sources / PA
Type: MySQL

Settings

Name PA Default ☒

MySQL Connection

Host 127.0.0.1

Database PA

User candra Password configured reset

TLS Client Auth ☐ With CA Cert ☐

Skip TLS Verify ☐

Connection limits

Max open unlimited

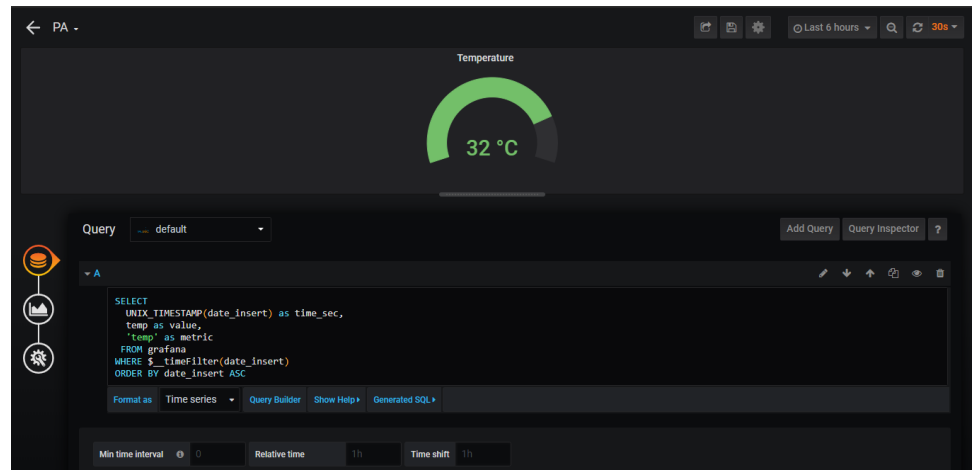
Max idle 2

Max lifetime 14400

MySQL details

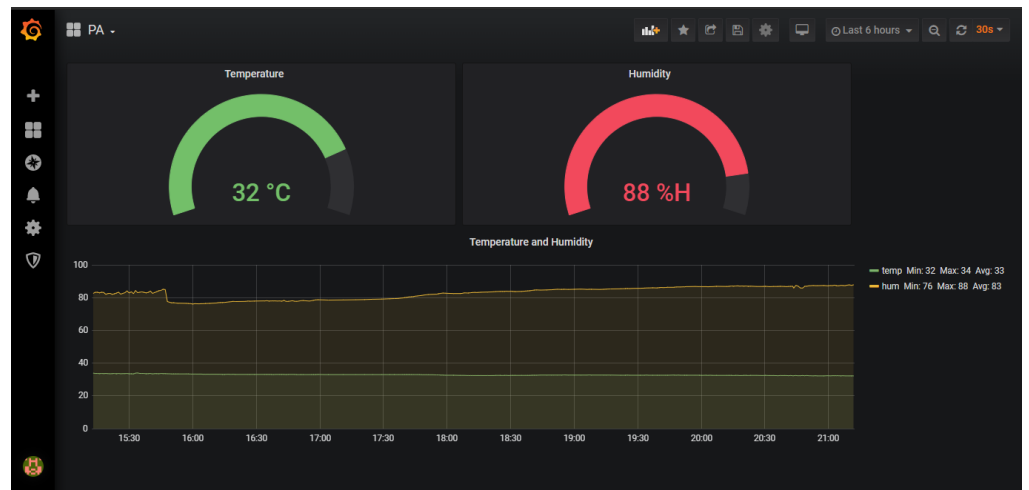
Gambar 20. Data Sources Grafana

4. Sesudah menambahkan Data Sources selanjutnya membuat Grafik yang mengambil data dari Database tadi dan ada script Grafana untuk membaca data database dan diubah menjadi grafik.



Gambar 21. Membuat grafik Grafana

5. Berikut adalah tampilan grafik pada Grafana.



Gambar 22. Grafik Grafana

6. Dapat menambah grafik untuk monitoring jaringan, server ataupun storage. Sehingga monitoring menjadi satu dan mudah.



Gambar 23. Grafik Grafana

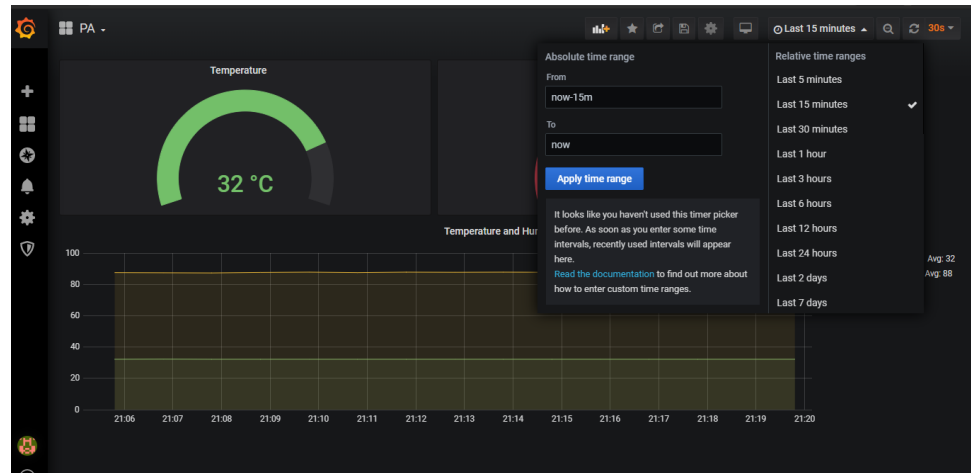
7. Dapat mengelola user seperti dibawah ini, seperti user grafana@grafana.com adalah viewer.

The Configuration page shows the Users tab for the organization 'Main Org'. The table below lists the current users:

Login	Email	Name	Seen	Role
admin	admin@localhost		4m	Admin
guest@grafana.com	guest@grafana.com	guest	14d	Viewer

Gambar 24. Manage User Grafana

8. Pada Grafana dapat melihat history dari suhu dan kelembapan, fitur tersebut ada di kanan atas pada layer



Gambar 25. History Grafana

4.3 Pengujian Notifikasi Telegram

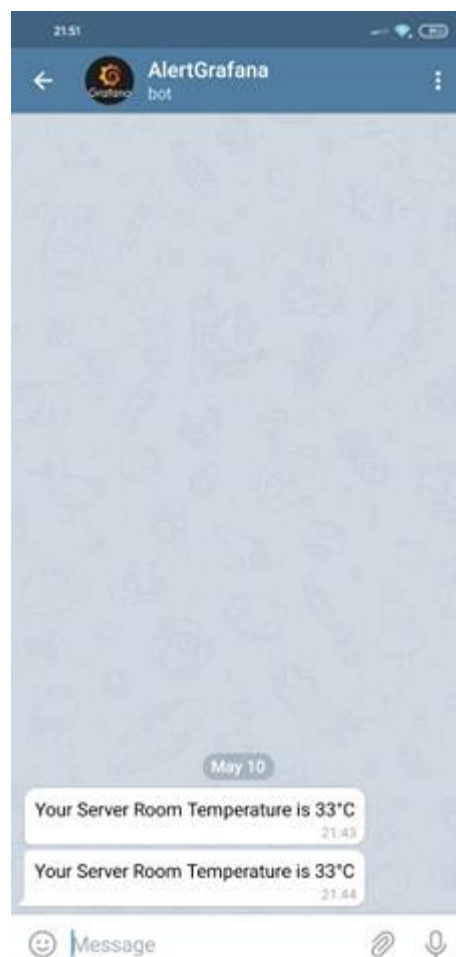
Berikut merupakan proses bagaimana terjadinya notifikasi peringatan Telegram yaitu:

1. Pada Script Bottelegram.js fungsinya sama dengan Mqtt.js yaitu menerima data dari Mqtt broker, bedanya di Script Bottelegram.js terdapat pengkondision apabila suhu mencapai batas yang ditentukan akan mengirim notifikasi ke telegram.
2. Jika script bottelegram.js dijalankan pengkondisian yang dibuat untuk pengujian ini adalah ketika suhu > 32 maka akan mengirimkan notifikasi ke telegram. Ketika suhu > 32 maka akan ada console.log (Alert Send to telegram), Ketika suhu ≥ 32 maka notifikasi tidak akan terkirim.

```
root@stb:/home/candramj# node bottele2.js
node-telegram-bot-api deprecated Automatic enabling of cancellation of promises is deprecated.
In the future, you will have to enable it yourself.
See https://github.com/yagop/node-telegram-bot-api/issues/319. module.js:652:30
(node:29767) Warning: N-API is an experimental feature and could change at any time.
Connecting MQTT
Subscribed to sensor
99.90,33.60
Temp : 33
Alert Send to Telegram
91.90,33.20
Temp : 33
Alert Send to Telegram
92.60,32.20
Temp : 32
92.80,32.20
Temp : 32
```

Gambar 26. Console Log Bottelegram.js

3. Berikut adalah notifikasi dari bot telegram.



Gambar 27. Notifikasi Bot Telegram

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dapat disimpulkan dari hasil implementasi dan pengujian “Smart Monitoring Temperatu and Humidity of the Room Server” yang telah dilakukan yaitu :

1. Dapat memonitoring temperature dan kelembapan pada ruang server.
2. Dapat memberi peringatan pada situasi tertentu pada ruang server berupa notifikasi. Sangat berguna apabila terjadi apa-apa pada server ditengah malam atau dihari libur.

DAFTAR PUSTAKA

Dwi Ely Kurniawan, Mohd Iqbal, John Friadi, Rohmat Indra Borman and Rio Rinaldi. (2019). Smart Monitoring Temperature and Humidity of the Room Server Using Raspberry Pi and Whatsapp Notifications. *Journal of Physics: Conference Series*, Volume 1351.

Temperature upload over MQTT using ESP8266 and DHT22 sensor. (n.d.). Retrieved March 16, 2020, from Things Board: <https://thingsboard.io/docs/samples/esp8266/temperature/>

Using MySQL in Grafana. (n.d.). Retrieved March 16, 2020, from Grafana: <https://grafana.com/docs/grafana/latest/features/datasources/mysql/>

What is the correct temperature for a server room? (2020 , May 3). Retrieved from Server Fault: <https://serverfault.com/questions/11145/what-is-the-correct-temperature-for-a-server-room>