

MIKROPROSESORDAN ANTARMUKA 2
LAPORAN UJIAN AKHIR SEMESTER



OLEH :

Hendri Darmawan (1210181044)
M. Irsyad Akmal N. (1210181045)
Athiya Refi Hanianty (1210181046)
Zulfatun Naimah (1210181047)

KELAS:

3-D4 Telekomunikasi B

DOSEN PENGAMPU :

Mohamad Ridwan

PROGRAM STUDI TEKNIK TELEKOMUNIKASI
DEPARTEMEN TEKNIK ELEKTRO
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
2020/2021

LAPORAN UJIAN AKHIR SEMESTER

I. DASAR TEORI

A. DHT 11

Sensor DHT11 adalah module sensor yang berfungsi untuk mensensing objek suhu dan kelembaban yang memiliki output tegangan analog yang dapat diolah lebih lanjut menggunakan mikrokontroler. Module sensor ini tergolong kedalam elemen resistif seperti perangkat pengukur suhu seperti contohnya yaitu NTC. Kelebihan dari module sensor ini dibanding module sensor lainnya yaitu dari segi kualitas pembacaan data sensing yang lebih responsif yang memiliki kecepatan dalam hal sensing objek suhu dan kelembaban, dan data yang terbaca tidak mudah terinterferensi. Sensor DHT11 pada umumnya memiliki fitur kalibrasi nilai pembacaan suhu dan kelembaban yang cukup akurat. Penyimpanan data kalibrasi tersebut terdapat pada memori program OTP yang disebut juga dengan nama koefisien kalibrasi.

- **Cara kerja Sensor DHT11**

Sensor DHT11 merupakan serangkaian komponen sensor dan IC controller yang dikemas dalam satu paket. Sensor ini ada yang memiliki 4 pin ada pula yang 3 pin. Tapi tidak menjadi masalah karena dalam penerapannya tidak ada perbedaan. Didalam bodi sensor yang berwarna biru atau putih terdapat sebuah Resistor dengan tipe NTC (*Negative Temperature Coefficient*).

Resistor jenis ini memiliki karakteristik dimana nilai resistansinya berbanding terbalik dengan kenaikan suhu. Artinya, semakin tinggi suhu ruangan maka nilai resistansi NTC akan semakin kecil. Sebaliknya nilai resistansi akan meningkat ketika suhu disekitar sensor menurun.

Selain itu didalamnya terdapat sebuah sensor kelembaban dengan karakteristik resistif terhadap perubahan kadar air di udara. Data dari kedua sensor ini diolah didalam IC controller. IC controller ini akan mengeluarkan output data dalam bentuk *single wire bidirectional*.

- **Spesifikasi Sensor DHT11**

1. Tegangan Input 3-5V
2. Arus 0.3mA, Idle 60uA
3. Periode sampling 2 detik
4. Output data serial
5. Resolusi 16bit
6. Temperatur antara 0°C sampai 50°C (akurasi 1°C)
7. Kelembapan antara 20% sampai 95% (akurasi 5%)

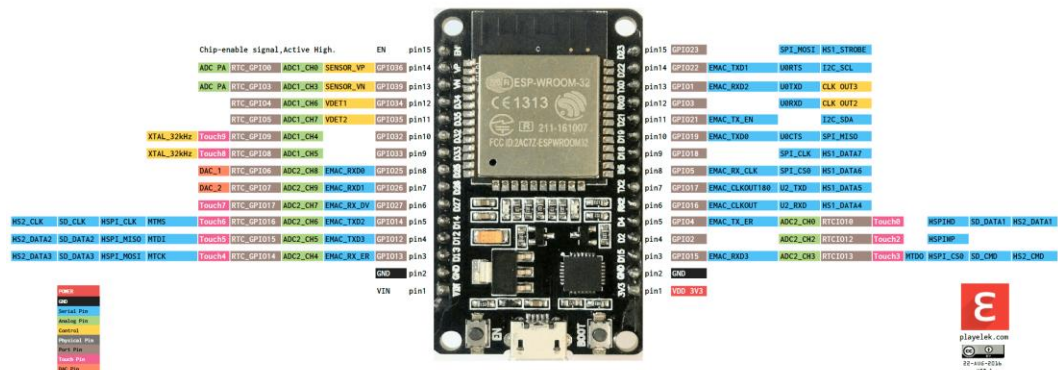
- **Susunan Pin**

Sensor DHT11 memiliki 2 versi, yaitu versi 4 pin dan versi 3 pin. Tidak ada perbedaan karakteristik dari 2 versi ini. Pada versi 4 pin, Pin 1 adalah tegangan sumber, berkisar antara 3V sampai 5V. Pin 2 adalah data keluaran (output) . Pin ke 3 adalah pin NC (normal y close) alias tidak digunakan dan pin ke 4 adalah Ground. Sedangkan pada versi 3 kaki, pin 1 adalah VCC antara 3V sampai 5V, pin 2 adalah data keluaran dan pin 3 adalah Ground.

B. ESP 32

ESP32 adalah mikrokontroler yang dikenalkan oleh Espressif System merupakan penerus dari mikrokontroler ESP8266. Pada mikrokontroler ini sudah tersedia modul WiFi dalam chip sehingga sangat mendukung untuk membuat sistem aplikasi Internet of Things. terlihat pada gambar di atas merupakan pin out dari ESP32. Pin tersebut dapat dijadikan input atau output untuk menyalakan LCD, lampu, bahkan untuk menggerakan motor DC.

DOIT ESP32 DEVKIT V1 PINOUT



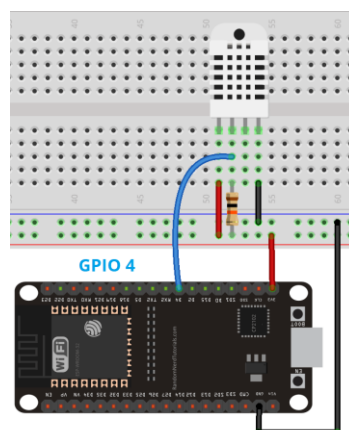
- Pada pin out tersebut terdiri dari :
 - a) 18 ADC (Analog Digital Converter, berfungsi untuk merubah sinyal analog ke digital)
 - b) 2 DAC (Digital Analog Converter, kebalikan dari ADC)
 - c) 16 PWM (Pulse Width Modulation)
 - d) 10 Sensor sentuh
 - e) 2 jalur antarmuka UART
 - f) pin antarmuka I2C, I2S, dan SPI

ESP32 ini memiliki tegangan operasi 3.3V berbeda dengan mikrokontroler ATmega pada Arduino Uno, jadi untuk membuat suatu rangkaian elektronik menggunakan ESP32 harus di perhatikan bahwa suplay listrik pada *rangkaian* tidak boleh lebih dari 3.3V semisal 5v apa lagi 9v. Jika tegangan di atas 3.3v diberikan pada rangkaian yang menggunakan ESP32 tentu akan merusak ESP32 dan membuat rangkaian elektronik tersebut mengeluarkan asap.

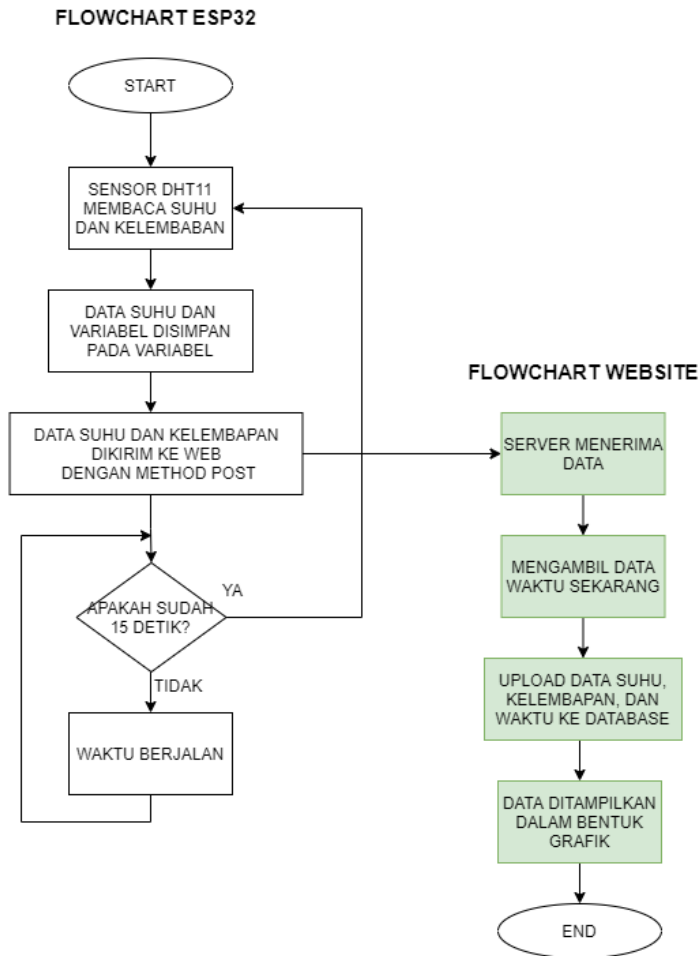
II. ALAT YANG DIGUNAKAN

1. ESP32
2. DHT11
3. Resistor 10k Ohm
4. Project Board
5. Kabel Jumper
6. Kabel USB

III. RANGKAIAN



IV. FLOWCHART



V. SOURCE CODE

• Source Code ESP32

```
#include <HTTPClient.h>

#include "DHT.h"
#include <WiFi.h>

#define DHTTYPE DHT11
#define DHTPIN 4
#define RETRY_LIMIT 20
DHT dht(DHTPIN, DHTTYPE);
const char* ssid = "myRouter";
```

```

const char* password = "myPassword";

void setup()
{
  dht.begin();
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP Address");
  Serial.println(WiFi.localIP());
}

void loop()
{
  int rtl = RETRY_LIMIT;
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  delay(500);
  while (isnan(t) || isnan(h)) {
    Serial.println("Check sensor again - " + rtl);
    h = dht.readHumidity();
    t = dht.readTemperature();
    delay(500);
    if (--rtl < 1) {

```

```

    ESP.restart();
}
}

HTTPClient http;
//membuat koneksi dengan server
http.begin("http://kado.atwebpages.com/iot/upload.php");
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
//format data yang dikirim
int httpResponseCode = http.POST("temperature=" + String(t) + "&humidity=" +
String(h));

if (httpResponseCode > 0) {
    //cek respon http
    String response = http.getString();
    Serial.println(httpResponseCode);
    Serial.println(response);
}
else {
    Serial.print("Error on sending post");
    Serial.println(httpResponseCode);
}
//mengakhiri koneksi http
http.end();

Serial.println("Temp = " + String(t));
Serial.println("humidity = " + String(h));

delay(15000);
}

```

- **Source Code config.php**

```
<?php
define('DB_SERVER', 'fdb26.awardspace.net');
define('DB_USER', '3444711_jastip');
define('DB_PASSWORD', 'keepitsecret');
define('DB_NAME', '3444711_jastip');
define('TB_ENV', 'iot');
?>
```

- **Source Code upload.php**

```
<?php
include('config.php');
$conn = mysqli_connect(DB_SERVER, DB_USER, DB_PASSWORD, DB_NAME)
or die("Unable to connect to MySQL");

if (mysqli_real_escape_string($conn, $_POST['temperature']) == NULL
||mysqli_real_escape_string($conn, $_POST['temperature']) == NAN){
    $temperature="NULL";
}else{
    $temperature=mysqli_real_escape_string($conn, $_POST['temperature']);
}

if (mysqli_real_escape_string($conn, $_POST['humidity']) == NULL){
    $humidity="NULL";
}else{
    $humidity=mysqli_real_escape_string($conn, $_POST['humidity']);
}

$logdate= date("Y-m-d H:i:s");

$insertSQL="INSERT into ".TB_ENV." (logdate,temperature,humidity) values
('".$logdate."','".$temperature."','".$humidity."')";

mysqli_query($conn,$insertSQL) or die("INSERT Query has Failed - ".$insertSQL );
?>
```


- **Source Code chart.php**

```

<?php
$koneksi=mysqli_connect("fdb26.awardspace.net","3444711_jastip","keepitsecret","34
44711_jastip");
$no = 0;
$menu = array();
$sql = "SELECT * FROM iot ORDER BY logdate ASC";
$qry = mysqli_query($koneksi, $sql);
while ($row = mysqli_fetch_array($qry)) {
    $temp[] = $row['2'];
    $hum[] = $row['3'];
    $date[] = $row['1'];
    $lasttemp = end($temp);
    $lasthum = end($hum);
}

$array = join(" ", $hum);
$array = join(" ", $temp);
$dat = join(" ", $date);
?>

<html lang="en">

<head>
    <meta http-equiv="refresh" content="60" />
    <title>Grafik Pembacaan Suhu Kelompok 3</title>
    <script
        type="text/javascript"
        src="https://code.jquery.com/jquery -
1.11.1.min.js"></script>
    <script
        type="text/javascript"
        src="https://code.highcharts.com/highcharts.js"></script>

```

```

<script                                                                    type="text/javascript"
src="https://code.highcharts.com/modules/exporting.js"></script>
<link href="jquery-gauge.css" type="text/css" rel="stylesheet">
<script type="text/javascript">
    var chart;
    $(document).ready(function() {
        chart = new Highcharts.Chart({
            chart: {
                renderTo: 'container',
                zoomType: 'xy'
            },
            title: {
                text: 'Grafik Suhu dan Kelembapan'
            },
            subtitle: {
                text: 'by: Kelompok 3'
            },
            xAxis: [{
                categories: ['<?php echo $dat; ?>'],
                labels: {
                    rotation: -45,
                    align: 'right',
                    style: {
                        fontSize: '13px',
                        fontFamily: 'Verdana, sans-serif'
                    }
                }
            }],
            yAxis: [{ // Primary yAxis
                labels: {
                    formatter: function() {

```

```

        return this.value + "";
    },
    style: {
        color: '#89A54E'
    }
},
title: {
    text: 'Suhu dan Kelembapan',
    style: {
        color: '#4572A7'
    }
}
}, { // Secondary yAxis
    title: {
        text: "",
        style: {
            color: '#4572A7'
        }
    },
    labels: {
        formatter: function() {
            return this.value + "";
        },
        style: {
            color: '#4572A7'
        }
    },
    opposite: true
}],
tooltip: {
    formatter: function() {

```

```

        return " + this.x + ' Nilai= ' + this.y;
    }
},
legend: {
    layout: 'vertical',
    align: 'left',
    x: 100,
    verticalAlign: 'top',
    y: 0,
    floating: true,
    backgroundColor: '#FFFFFF'
},
series: [{
    name: 'Humidity',
    color: '#4572A7',
    type: 'spline',
    data: [<?php echo $array; ?>]
}, {
    name: 'Temperature',
    color: '#658475',
    type: 'spline',
    data: [<?php echo $array; ?>]
}]
});
});
</script>
<style>
.demo1 {
    width: 40vw;
    height: 40vw;
    box-sizing: border-box;

```

```
float: left;
margin: 20px;
margin-bottom: 0px;
}
```

```
.demo2 {
width: 40vw;
height: 40vw;
box-sizing: border-box;
float: right;
margin: 20px;
margin-bottom: 0px;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div style="width: 100%; height: 100%;">
```

```
<div class="gauge1 demo1"></div>
```

```
<div class="gauge2 demo2"></div>
```

```
<div style="left:250px;top:0.2vw;position: absolute;">Temperature</div>
```

```
<div style="right:255px;top:0.2vw;position: absolute;">Humidity</div>
```

```
</div>
```

```
<div id="container" style="width: 100%; height: 100%;"></div>
```

```
<script type="text/javascript" src="jquery-gauge.min.js"></script>
```

```
<script>
```

```
// first example
```

```
$('.gauge1').gauge({
```

```
values: {
```

```
0: '0°C',
```

```
10: '5°C',
```

```

    20: '10°C',
    30: '15°C',
    40: '20°C',
    50: '25°C',
    60: '30°C',
    70: '35°C',
    80: '40°C',
    90: '45°C',
    100: '50°C'
  },
  colors: {
    0: 'blue',
    20: '#666',
    40: 'green',
    60: 'orange',
    80: 'red',
  },
  angles: [
    90,
    360
  ],
  lineWidth: 10,
  arrowWidth: 20,
  arrowColor: '#666',
  inset: true,
  value: <?= $lasttemp ?> / 50 * 100
});

```

```

// second example
$('.gauge2').gauge({
  values: {

```

```

        0: '0%',
        10: '10%',
        20: '20%',
        30: '30%',
        40: '40%',
        50: '50%',
        60: '60%',
        70: '70%',
        80: '80%',
        90: '90%',
        100: '100%'
    },
    colors: {
        80: '#666',
        50: '#378618',
        20: '#ffa500',
        0: '#f00'
    },
    angles: [
        180,
        450
    ],
    lineWidth: 10,
    arrowWidth: 20,
    arrowColor: 'blue',
    inset: true,

    value: <?= $lasthum ?>

    });
</script>
</body>

```

</html>

VI. HASIL PERCOBAAN

- Tampilan Struktur Tabel

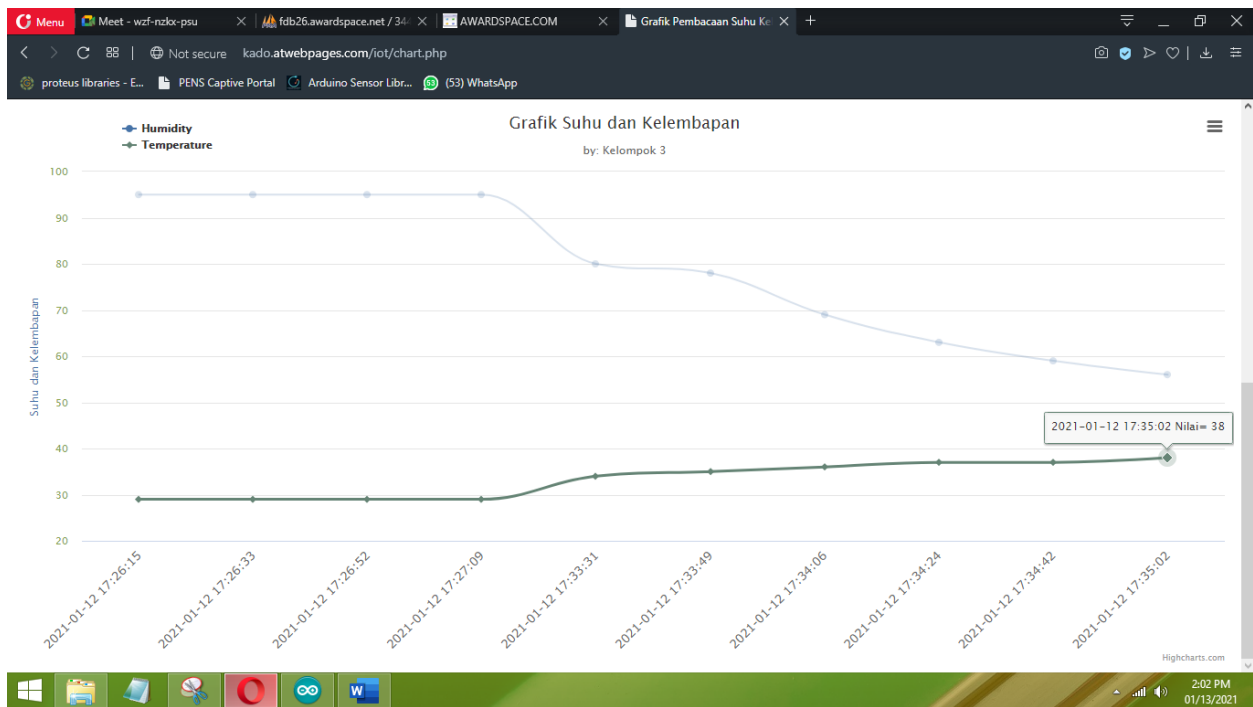
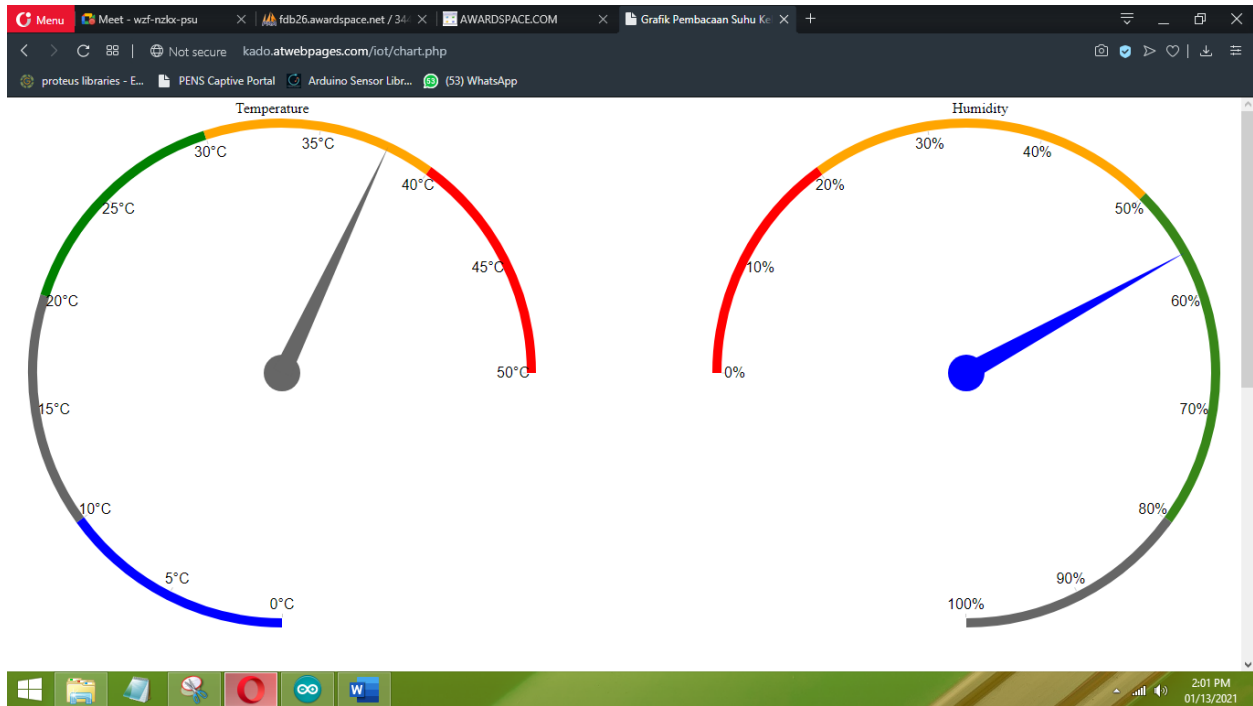
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 id	int(200)			No	None		AUTO_INCREMENT	Change
<input type="checkbox"/>	2 logdate	datetime			No	None			Change
<input type="checkbox"/>	3 temperature	float			No	None			Change
<input type="checkbox"/>	4 humidity	float			No	None			Change

- Tampilan Isi Tabel

				id	logdate	temperature	humidity
<input type="checkbox"/>	Edit	Copy	Delete	461	2021-01-12 17:26:15	29	95
<input type="checkbox"/>	Edit	Copy	Delete	462	2021-01-12 17:26:33	29	95
<input type="checkbox"/>	Edit	Copy	Delete	463	2021-01-12 17:26:52	29	95
<input type="checkbox"/>	Edit	Copy	Delete	464	2021-01-12 17:27:09	29	95
<input type="checkbox"/>	Edit	Copy	Delete	465	2021-01-12 17:33:31	34	80
<input type="checkbox"/>	Edit	Copy	Delete	466	2021-01-12 17:33:49	35	78
<input type="checkbox"/>	Edit	Copy	Delete	467	2021-01-12 17:34:06	36	69
<input type="checkbox"/>	Edit	Copy	Delete	468	2021-01-12 17:34:24	37	63
<input type="checkbox"/>	Edit	Copy	Delete	469	2021-01-12 17:34:42	37	59
<input type="checkbox"/>	Edit	Copy	Delete	470	2021-01-12 17:35:02	38	56

- **Tampilan Web**

Dapat diakses di: <http://kado.atwebpages.com/iot/chart.php>



VII. CARA KERJA DAN ANALISA

Perancangan kali ini membuat sebuah teknologi jaringan elektronik yang terintegrasi antara perangkat elektronik dan peralatan-peralatan rumah tangga sehingga keseluruhan rumah dapat diawasi secara terpusat sebagai sebuah mesin. Salah satu fokusnya adalah fitur untuk mengukur suhu dan kelembapan ruangan dengan dimonitor dari website. Cara kerja dari rancangan ini adalah sensor DHT11 pada pin GPIO 4 ESP32 membaca suhu dan kelembapan ruangan. Terkadang ESP32 mengalami kesalahan dalam membaca suhu dan kelembapan dengan menghasilkan data pembacaan berupa NaN. Hal tersebut dapat diatasi dengan melakukan restart ESP32. Dalam firmware kami sudah ada pengkondisian tersebut sehingga ESP32 akan melakukan restart otomatis hingga didapatkan data pembacaan yang sesuai dengan kondisi ruangan. ESP32 mendapatkan nilai pembacaan sensor dan melakukan upload nilai tersebut ke web dengan metode POST hal ini dilakukan oleh syntax `http.POST("temperature=" + String(t) + "&humidity=" + String(h))` ke alamat <http://kado.atwebpages.com/iot/upload.php>. Pada syntax php `upload.php` nilai tersebut diterima dan disimpan variabel POST lalu dilakukan penambahan perlindungan atau mencegah Sql Injection dengan fungsi `mysqli_real_escape_string()`. Hal ini mencegah karakter-karakter unik atau karakter khusus sebelum mengirim query ke mysql yang dapat membahayakan data termasuk dari serangan SQL injection. Berikutnya php akan mendapatkan waktu pada saat data tersebut diterima.

Kemudian data suhu, kelembapan, dan waktu dimasukkan ke tabel database bernama `iot` dengan perintah `insert`. Nilai pembacaan pada database tersebut ditampilkan pada halaman website dalam bentuk meter dan grafik menggunakan library javascript. Data pembacaan suhu dan kelembapan terakhir ditampilkan pada meter yang terdapat indicator suhu dan kelembapan yang sesuai pada saat itu yaitu didapatkan dari array terakhir yang diambil dari database. Website monitoring suhu tersebut dapat diakses di <http://kado.atwebpages.com/iot/chart.php>. Website tersebut terupdate otomatis setiap 60 detik sekali sehingga user tidak perlu melakukan refresh secara manual untuk memperbarui data.

Video demo projek dapat dilihat di: <https://youtu.be/aO7MtPGjquc>