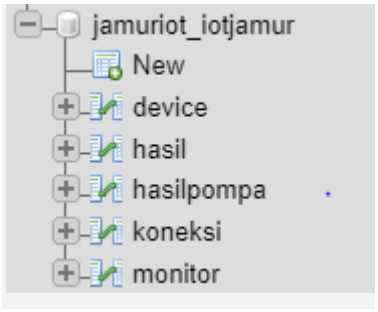
BAB V. IMPLEMENTASI DAN PENGUJIAN

5.1 Implementasi

Setelah dilakukan perancangan sistem, maka selanjutnya adalah implementasi sesuai dengan perancangan yang dilakukan. Pada bagian ini menjelaskan tentang hasil dari sistem yang telah dibangun. Implementasi dijelaskan secara detail secara visual dengan tampilan gambar dan potongan kode program atau *listing code*, sebagai berikut:

5.1.1 Implementasi Database

Implementasi database sesuai dengan perancangan menggunakan database *MySQL* yang digunakan untuk menyimpan data dalam sistem, sebagai berikut:



Gambar 5.1 Implementasi Tabel Database

5 tabel yang diimplementasikan dalam sistem tersebut yaitu, tabel device, hasil(hasil kipas), hasilpompa, koneksi, dan monitor.

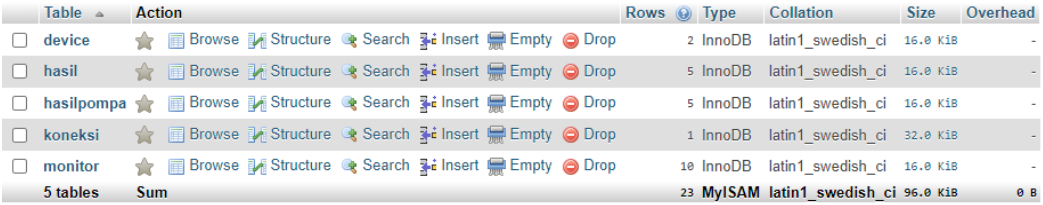
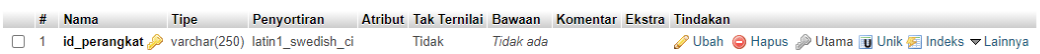


Table	Action	Rows	Type	Collation	Size	Overhead
device	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
hasil	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
hasilpompa	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
koneksi	Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	32.0 KiB	-
monitor	Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16.0 KiB	-
5 tables	Sum	23	MyISAM	latin1_swedish_ci	96.0 KiB	0 B

Gambar 5.2 Detail Tabel Database

Pada Gambar 5.3 merupakan tabel device yang digunakan untuk menyimpan data-data perangkat. Dalam tabel tersebut terdapat atribut id_perangkat (*primary key*).



#	Nama	Tipe	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id_perangkat	varchar(250)	latin1_swedish_ci		Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Lainnya

Gambar 5.3 Implementasi Tabel Device

Pada Gambar 5.4 merupakan tabel koneksi yang digunakan untuk menyimpan data-data perangkat yang terhubung dengan website. Dalam tabel tersebut terdapat id (*primary key*), nama_perangkat, dan id_perangkat.

#	Nama	Tipe	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(11)			Tidak	*Tidak ada*		AUTO_INCREMENT	Ubah Hapus Utama Unik Lainnya
2	nama_perangkat	varchar(250)	latin1_swedish_ci		Tidak	*Tidak ada*			Ubah Hapus Utama Unik Lainnya
3	id_perangkat	varchar(250)	latin1_swedish_ci		Tidak	*Tidak ada*			Ubah Hapus Utama Unik Lainnya

Gambar 5.4 Implementasi Tabel Koneksi

Pada Gambar 5.5 merupakan tabel monitor yang digunakan untuk menampung data mentah yang nantinya akan diolah ke dalam perhitungan fuzzy. Dalam tabel tersebut terdapat atribut id (*primary key*), id_perangkat (*foreign key*), temperature, dan humidity.

#	Nama	Tipe	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(11)			Tidak	*Tidak ada*		AUTO_INCREMENT	Ubah Hapus Utama Unik Indeks Lainnya
2	id_perangkat	varchar(250)	latin1_swedish_ci		Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Lainnya
3	temperature	int(11)			Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Lainnya
4	humidity	int(11)			Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Lainnya

Gambar 5.5 Implementasi Tabel Monitor

Pada Gambar 5.6 merupakan tabel hasil (hasil kipas) yang berisi data dari hasil perhitungan fuzzy untuk kipas. Dalam tabel tersebut terdapat atribut id (*primary key*), id_perangkat (*foreign key*), temperature, humidity, dan hasil hitung kipas.

#	Nama	Tipe	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(11)			Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Spasial Lainnya
2	id_perangkat	varchar(250)	latin1_swedish_ci		Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Spasial Lainnya
3	tempereature	int(11)			Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Spasial Lainnya
4	humidity	int(11)			Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Spasial Lainnya
5	hasil_hitung	float			Tidak	*Tidak ada*			Ubah Hapus Utama Unik Indeks Spasial Lainnya

Gambar 5.6 Implementasi Tabel Hasil Kipas

Pada Gambar 5.7 merupakan tabel hasil pompa yang berisi data dari hasil perhitungan fuzzy untuk pompa. Dalam tabel tersebut terdapat atribut id (*primary key*), id_perangkat (*foreign key*), temperature, humidity, dan hasil hitung pompa.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐	2	id_perangkat	varchar(250)	latin1_swedish_ci		No	None			Change Drop More
☐	3	temperature	int(11)			No	None			Change Drop More
☐	4	humidity	int(11)			No	None			Change Drop More
☐	5	hasilpompa	float			No	None			Change Drop More

Gambar 5.7 Implementasi Tabel Hasil Pompa

5.1.2 Implementasi Kode Program

Implementasi beberapa potongan kode program alur proses kerja yang ada dalam sistem sesuai dengan perancangan yang telah dilakukan sebelumnya. Implementasi berdasarkan proses analisis dan perancangan dijelaskan secara detail, sebagai berikut:

a. Publish-Subscribe Data Di ESP32

Pada potongan kode program di bawah ini terdapat fungsi callback yang akan dipanggil oleh MQTT ketika menerima data. Pada bagian fungsi loop, terdapat potongan kode program yang berfungsi untuk melakukan publish data.

```
void callback(char* topic, byte* payload, unsigned int length) {

  if (top.equals("nodemcu/smanualkps")) {
    manual = "";
    for (int i = 0; i < length; i++) {
      manual = (char)payload[i];
    }
  }
}

void loop() {
  // put your main code here, to run repeatedly:
    if (client.publish(macTpc, String(mAc).c_str())) {
    }
    // Again, client.publish will return a boolean value depending
on whether it succeded or not.
    // If the message failed to send, we will try again, as the
connection may have broken.
    else {
      client.connect(clientID);
      delay(10);  //  This  delay  ensures  that  client.publish
doesn't clash with the client.connect call
```

```
       client.publish(maTopic, String(mAc).c_str());
     }
```

b. Publish Data Dari Website

Pada potongan kode program di bawah ini terdapat fungsi yang berguna untuk mempublish data dari website.

```
function publishToMQTT_Pompa(message) {
      var device = <?php echo
json_encode($_SESSION["id_perangkat"]);?>;
     message = new Paho.MQTT.Message(device);
     message.destinationName = "nodemcu/pompa";
     client.send(message);
}

function publishToMQTT_Kipas(){
     var device = <?php echo
json_encode($_SESSION["id_perangkat"]);?>;
     message = new Paho.MQTT.Message(device);
     message.destinationName = "nodemcu/kipas";
     client.send(message);
}
```

c. Perhitungan Fuzzy Sugeno

Potongan kode program di bawah ini berisi perhitungan dari metode yang menghasilkan keputusan untuk menjalankan aktuator secara otomatis.

```
<?php
    function findMin($x, $y){
        if($x <= $y){
            return $x;
        } else{
            return $y;
        }
    }

    function tempDingin($temperature){
        if($temperature <= 0){
            return 1;
        } elseif ($temperature > 0 && $temperature < 20){
            return (20 - $temperature) / (20-0);
        } else{
            return 0;
        }
```

```
}

function tempNormal($temperature){
    if($temperature > 15 && $temperature <= 22.5){
        return ($temperature - 15) / (22.5-15);
    } elseif($temperature > 22.5 && $temperature < 30){
        return (30 - $temperature) / (30-22.5);
    } else {
        return 0;
    }
}
```

```
function tempPanas($temperature){
    if($temperature <= 25){
        return 0;
    } elseif ($temperature > 25 && $temperature < 40){
        return ($temperature - 25) / (40-25);
    } else{
        return 1;
    }
}

function humKering($humidity){
    if($humidity <= 0){
        return 1;
    } elseif ($humidity > 0 && $humidity <= 35){
        return (35 - $humidity) / (35-0);
    } else{
        return 0;
    }
}

function humNormal($humidity){
    if($humidity > 25 && $humidity <= 52.5){
        return ($humidity - 25) / (52.5-25);
    } elseif($humidity > 52.5 && $humidity < 80){
        return (80 - $humidity) / (80-52.5);
    } else{
        return 0;
    }
}

function humBasah($humidity){
    if($humidity <= 70){
        return 0;
    } elseif ($humidity > 70 && $humidity < 100){
        return (100 - $humidity) / (100-70);
    } else{
        return 1;
    }
}

function pompaCepat($alfa){
    if($alfa <= 0){
        return 0;
    } else{

       return 4;
    }
```

```
    }

    function pompaSedang($alfa){
        if($alfa <= 0){
            return 0;
        } else{
            return 10;
        }
    }
```

```
     function pompaLama($alfa){
        if($alfa <= 0){
            return 0;
        } else{
            return 14;
        }
    }

    function kipasCepat($beta){
        if($beta <= 0){
            return 0;
        } else{
            return 10;
        }
    }

    function kipasSedang($beta){
        if($beta <= 0){
            return 0;
        } else{
            return 25;
        }
    }

    function kipasLama($beta){
        if($beta <= 0){
            return 0;
        } else{
            return 40;
        }
    }

    function perhitungan1($temperature, $humidity){
        $alfa[0] = findMin(tempDingin($temperature),
humKering($humidity));
        $z[0] = pompaSedang($alfa[0]);

        $alfa[1] = findMin(tempDingin($temperature),
humNormal($humidity));
        $z[1] = pompaCepat($alfa[1]);

        $alfa[2] = findMin(tempDingin($temperature),
humBasah($humidity));
        $z[2] = pompaCepat($alfa[2]);

        $alfa[3] = findMin(tempNormal($temperature),
humKering($humidity));
        $z[3] = pompaSedang($alfa[3]);
```

```
        $alfa[4] = findMin(tempNormal($temperature),
humNormal($humidity));
        $z[4] = pompaSedang($alfa[4]);

        $alfa[5] = findMin(tempNormal($temperature),
humBasah($humidity));
        $z[5] = pompaCepat($alfa[5]);
```

```
        $alfa[6] = findMin(tempPanas($temperature),
humKering($humidity));
        $z[6] = pompaLama($alfa[6]);

        $alfa[7] = findMin(tempPanas($temperature),
humNormal($humidity));
        $z[7] = pompaSedang($alfa[7]);

        $alfa[8] = findMin(tempPanas($temperature),
humBasah($humidity));
        $z[8] = pompaSedang($alfa[8]);


        $temp_1 = 0;
        $temp_2 = 0;
        $hasil = 0;

        for($i = 0; $i < 9; $i++){
            $temp_1 = $temp_1 + $alfa[$i] * $z[$i];
            // echo $temp_1;
            // echo "<br>";
            $temp_2 = $temp_2 + $alfa[$i];
            // echo $temp_2 ;
            // echo "<br>";
        }
        $hasil = $temp_1 / $temp_2;
        return $hasil;
    }

    function perhitungan2($temperature, $humidity){
        $beta[0] = findMin(tempDingin($temperature),
humKering($humidity));
        $zz[0] = kipasCepat($beta[0]);

        $beta[1] = findMin(tempDingin($temperature),
humNormal($humidity));
        $zz[1] = kipasSedang($beta[1]);

        $beta[2] = findMin(tempDingin($temperature),
humBasah($humidity));
        $zz[2] = kipasLama($beta[2]);

        $beta[3] = findMin(tempNormal($temperature),
humKering($humidity));
        $zz[3] = kipasSedang($beta[3]);
```

```
        $beta[4] = findMin(tempNormal($temperature),
humNormal($humidity));
        $zz[4] = kipasCepat($beta[4]);

        $beta[5] = findMin(tempNormal($temperature),
humBasah($humidity));
        $zz[5] = kipasLama($beta[5]);
```

```
        $beta[6] = findMin(tempPanas($temperature),
humKering($humidity));
        $zz[6] = kipasSedang($beta[6]);

        $beta[7] = findMin(tempPanas($temperature),
humNormal($humidity));
        $zz[7] = kipasSedang($beta[7]);

        $beta[8] = findMin(tempPanas($temperature),
humBasah($humidity));
        $zz[8] = kipasLama($beta[8]);


        $temp_3 = 0;
        $temp_4 = 0;
        $hasil1 = 0;

        for($i = 0; $i < 9; $i++){
            $temp_3 = $temp_3 + $beta[$i] * $zz[$i];
            // echo $temp_3;
            // echo "<br>";
            $temp_4 = $temp_4 + $beta[$i];
            // echo $temp_4 ;
            // echo "<br>";
        }

        $hasil1 = $temp_3 / $temp_4;
        return $hasil1;
    }

?>
```

d. Subscribe Data dan Mengirim Data Ke Database Di Python

Pada kode program di bawah ini berisi subscribe data dari ESP32 dan mengirimkan data ke database. Selain itu, terdapat potongan kode program untuk men-trigger halaman perhitungan metode agar aktuator berjalan secara otomatis.

```
driver = webdriver.Chrome(executable_path="C:\\chromedriver.exe",
options=chrome_options)
driver.get('http://jamuriot.my.id/sugeno.php')

def on_connect(client, userdata, flags, rc):
        # """The callback for when the client receives a CONNACK
response from the server."""
        # print('Connected with result code' +str(rc))
        if rc == 0:
```

```
            client.connected_flag = True  # set flag
            print("connected OK")
            client.subscribe([(MQTT_TOPIC,0), (MQTT_TOPIC1,0)])
        else:
            print("Bad connection Returned code=", rc)

def on_message(client, userdata, msg):
```

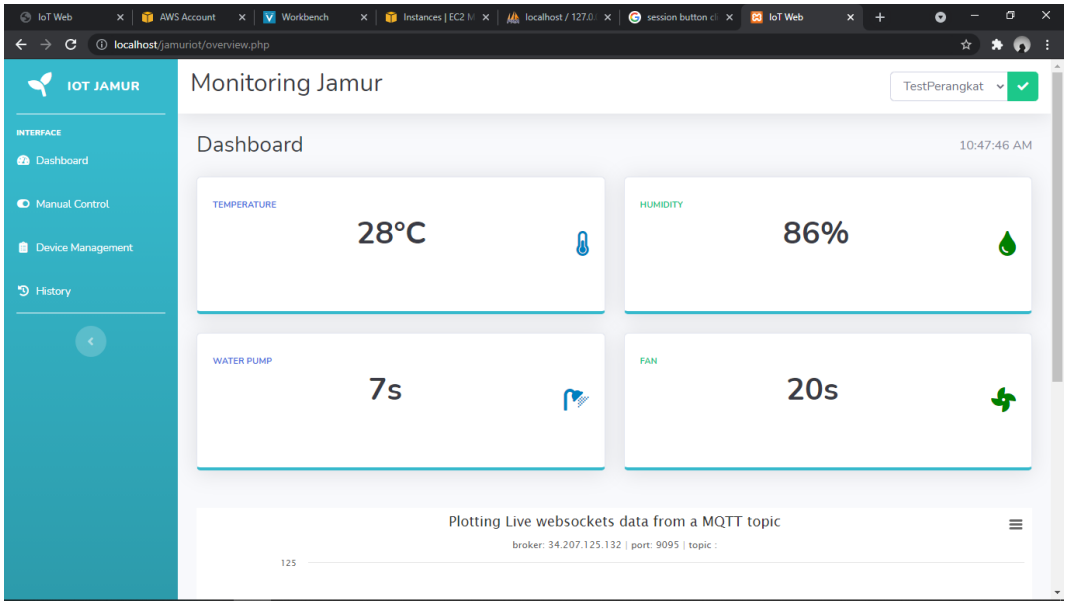
```
    if (temperature == 0 or humidity == 0 or mac == ''):
        if (msg.topic == 'jamur/temperature'):
                temperature = msg.payload.decode("utf-8")
                tempT = msg.topic
        elif (msg.topic == 'jamur/humidity'):
                humT = msg.topic
                humidity = msg.payload.decode("utf-8")
        elif (msg.topic == 'jamur/mac'):
                macAd = msg.topic
                mac = msg.payload.decode("utf-8")

    if (temperature != 0 and humidity != 0 and mac != ''):
        data1 = [(tempT, temperature), (humT, humidity), (macAd,
mac)]
        r =
requests.post('http://192.168.1.6/Jamuriot/sendData.php', data1)
```

5.1.3 Implementasi Tampilan Sistem

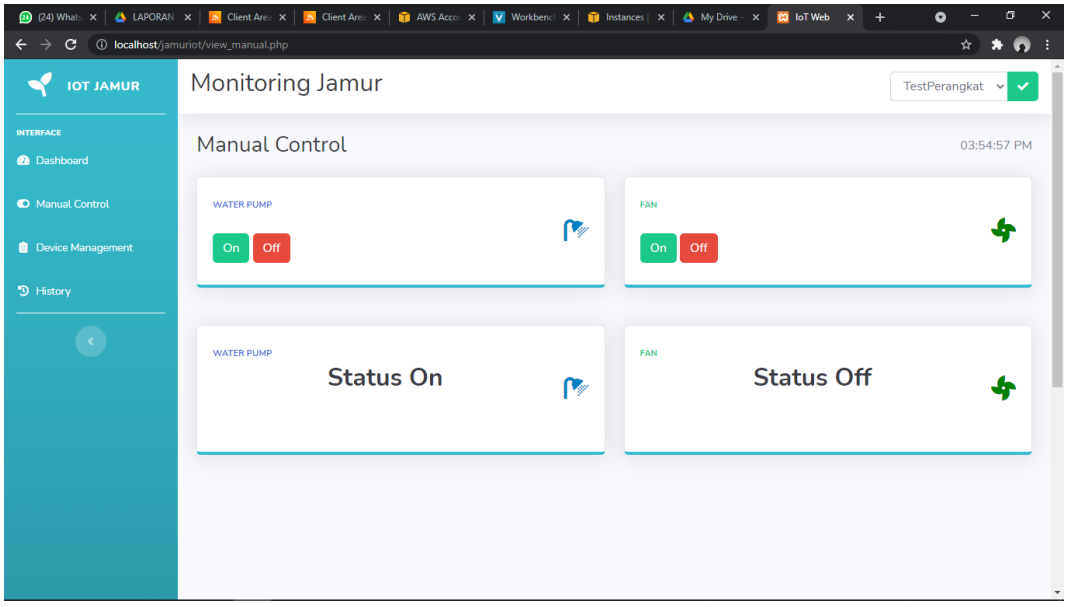
Implementasi *user interface* dari sistem sesuai dengan perancangan desain tampilan yang dilakukan sebelumnya, sebagai berikut:

Gambar 5.8 merupakan halaman utama untuk melihat grafik dan memonitoring data suhu udara, dan kelembaban udara serta juga menampilkan hasil durasi perhitungan fuzzy logic. Data-data tersebut bersifat dinamis mengikuti penyesuaian data yang dipublish oleh ESP32.



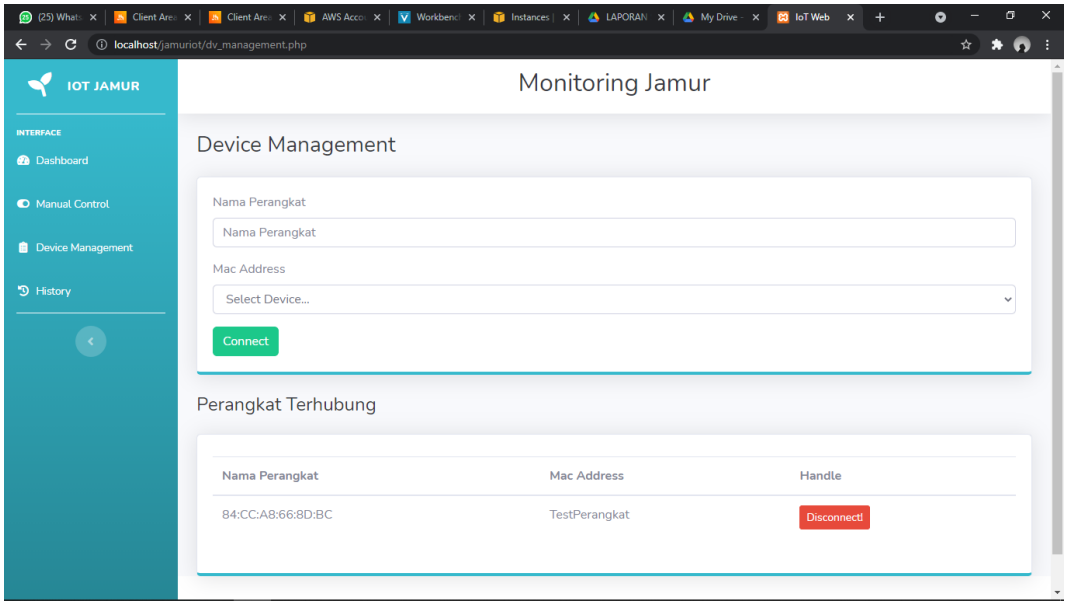
Gambar 5.8 Tampilan Dashboard

Gambar 5.9 merupakan tampilan halaman manual control, terdapat *button* untuk menghidupkan atau mematikan aktuator secara manual dan juga status aktuator yang on atau off.



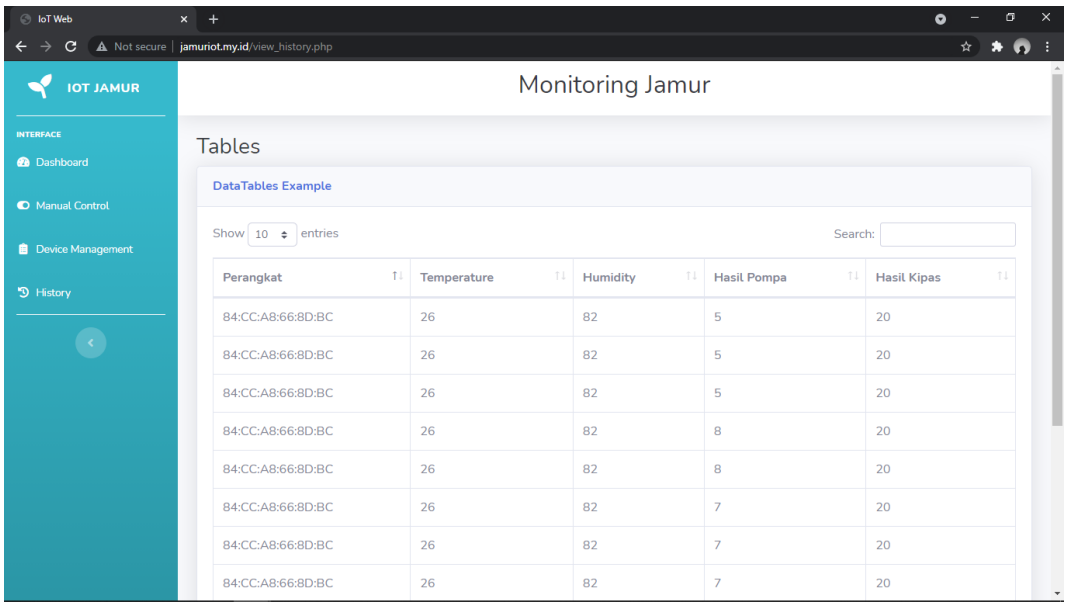
Gambar 5.9 Tampilan Manual Control

Gambar 5.10 merupakan tampilan untuk menghubungkan perangkat ke website yang di dalamnya terdapat form nama dan pilihan perangkat yang akan dikoneksikan. Selain itu, pada tampilan ini terdapat tabel yang menampilkan perangkat mana saja yang sudah terkoneksi ke website.



Gambar 5.10 Tampilan Device Management

Pada Gambar 5.11 digunakan untuk menampilkan data monitoring suhu, kelembaban dan juga hasil fuzzy untuk pompa dan kipas.



Gambar 5.11 Tampilan Data History

5.1.4 Implementasi Hardware

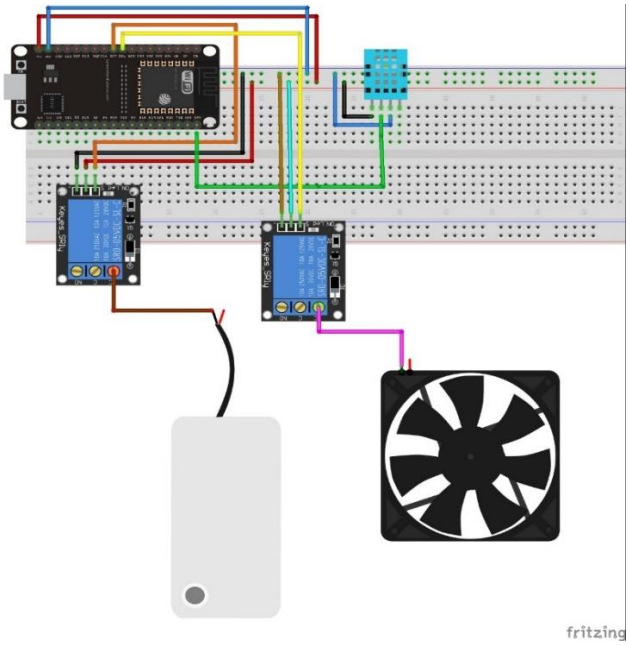
Implementasi hardware dilakukan dengan menggambarkan penerapan hardware yaitu rangkaian ESP32 dengan sensor dan relay.

a. Rangkaian ESP32 dengan sensor dan relay

Sistem monitoring suhu kelembaban otomatis tanaman jamur ini dengan metode fuzzy menggunakan mikrokontroller ESP32. ESP32 untuk membaca data

dari sensor DHT11 untuk kemudian value dari sensor tersebut dikirimkan ke RaspberryPi.

Rangkaian ESP32 terhubung dengan sensor DHT11 menggunakan kabel jumper. Untuk penggunaan pin vin dan ground terhubung secara parallel agar kebutuhan pin daya dan ground terpenuhi. Rangkaian ditunjukkan pada Gambar 5.12.



Gambar 5.12 Rangkaian Hardware

5.2 Pengujian

Pada tahap ini, akan dilakukan pengujian menggunakan teknik *blackbox*. Dimana sistem akan melakukan pengujian terhadap kinerja sistem ini. Adapun dua jenis pengujian yang dilakukan, yaitu:

1. Pengujian Perangkat Keras

Pada tahap ini dilakukan pengujian apakah sensor yang digunakan dapat memberikan nilai suhu udara, dan kelembaban udara dengan tepat dan mengirimkannya ke ESP32. Jika sensor tidak dapat memberikan nilai input dan tidak dapat mengirimkan ke ESP32, maka akan dilakukan perangkaian ulang agar mendapatkan nilai sensor yang tepat.

2. Pengujian Perangkat Lunak

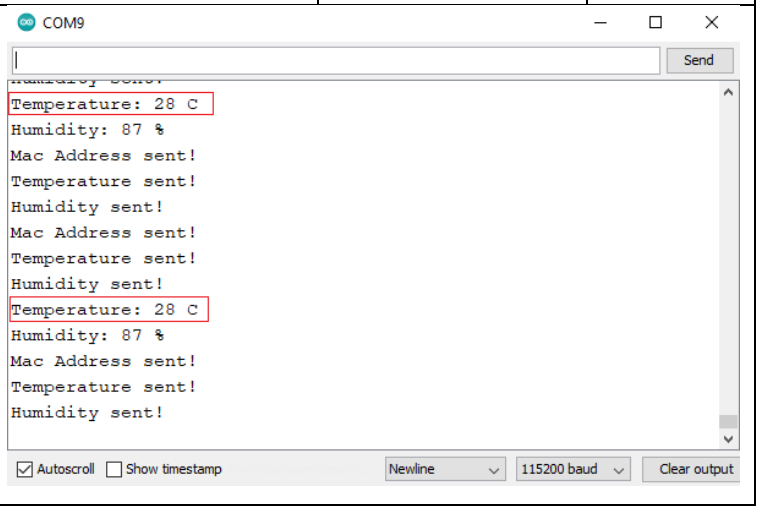
Pada tahap ini, dilakukan pengujian sistem. Apakah sistem ini dapat melakukan penerimaan data dari ESP32 dan menampilkannya pada pengguna. Apabila sistem belum menerima data sensor dari ESP32, maka dilakukan pemantauan terhadap pengiriman dari perangkat keras ke dalam perangkat lunak. Kemudian perangkat lunak akan melakukan analisa untuk memberikan *output* durasi penyiraman dan durasi kipas menggunakan metode *Fuzzy* berdasarkan nilai suhu udara, dan kelembaban udara.

a. Deteksi Suhu Udara

Nama kolom	Keterangan
Id	#1
Nama	Deteksi Suhu Udara
Deskripsi	Use case ini mendeskripsikan tentang pengujian deteksi suhu udara
Goals	Alat dapat memberikan nilai suhu udara dengan akurat
Actor	ESP32 dan sensor DHT11
Normal Flow	1. ESP32 melakukan pembacaan nilai analog suhu udara 2. ESP32 menyimpan data suhu udara
Alternative Flow	-
Include	-

Tabel 5.1 Use Case Testing Deteksi Suhu Udara

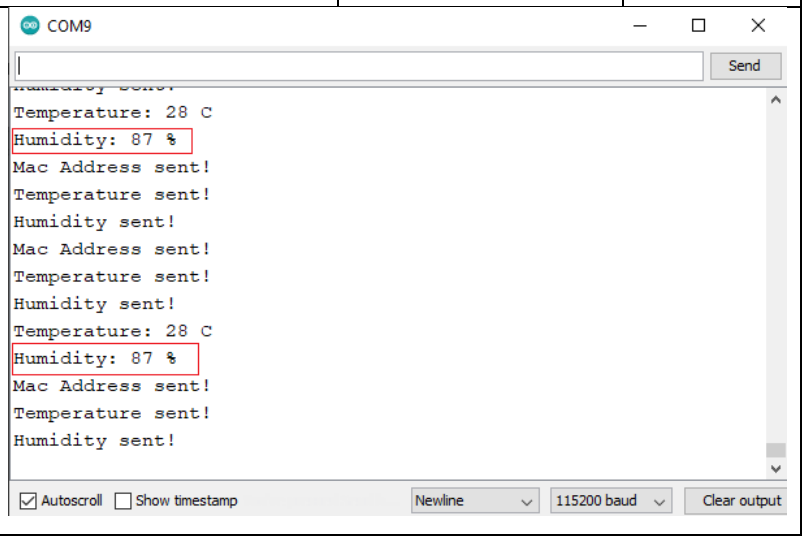
No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	ESP32 melakukan pembacaan nilai analog suhu udara	ESP32 dapat memberikan nilai analog suhu udara	Sesuai yang diharapkan	Valid

2.	ESP32 menyimpan data suhu udara	ESP32 menyimpan data suhu udara pada suatu variabel	Sesuai yang diharapkan	Valid
Gambar			

b. Deteksi Kelembaban Udara

Nama kolom	Keterangan
Id	#2
Nama	Deteksi Kelembaban Udara
Deskripsi	Use case ini mendeskripsikan tentang pengujian deteksi kelembaban udara
Goals	Alat dapat memberikan nilai kelembaban udara dengan akurat
Actor	ESP32 dan sensor DHT11
Normal Flow	1. ESP32 melakukan pembacaan nilai analog kelembaban udara 2. ESP32 menyimpan data kelembaban udara
Alternative Flow	-
Include	-

Tabel 5.2 Use Case Testing Deteksi Kelembaban Udara

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	ESP32 melakukan pembacaan nilai analog kelembaban udara	ESP32 dapat memberikan nilai analog kelembaban udara	Sesuai yang diharapkan	Valid
2.	ESP32 menyimpan data kelembaban udara	ESP32 menyimpan data kelembaban udara pada suatu variabel	Sesuai yang diharapkan	Valid
	Gambar			

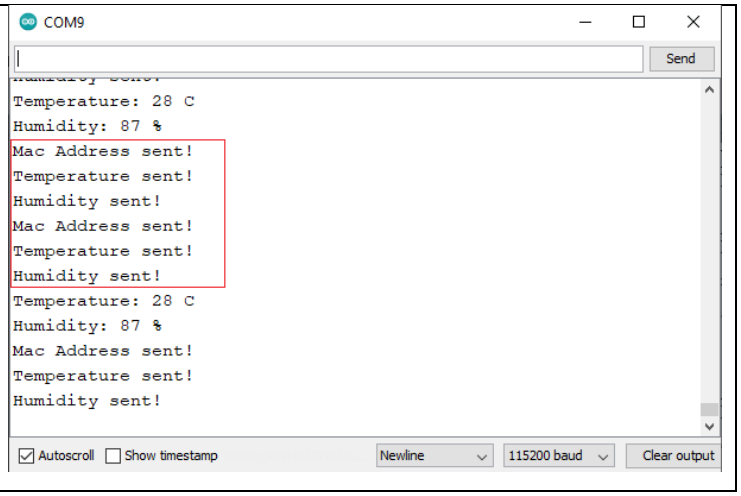
c. Pengiriman Nilai Sensor

Nama kolom	Keterangan
Id	#3
Nama	Pengiriman Nilai Sensor
Deskripsi	Use case ini mendeskripsikan tentang pengujian pengiriman nilai sensor

Goals	Alat dapat melakukan pengiriman data sensor dengan ESP32 ke Raspberry Pi
Actor	ESP32 dan Raspberry Pi
Normal Flow	1. ESP32 mengkoneksikan jaringan wifi dan mendapatkan alamat ip 2. ESP32 mengkoneksikan ke MQTT broker 3. ESP32 mengirimkan data sensor ke Raspberry Pi
Alternative Flow	1.1 ESP32 gagal mengkoneksikan dengan jaringan wifi 1.1.1 Menampilkan pesan berupa string titik “.” 1.1.2 Menghubungkan ulang pada jaringan wifi 1.1.3 ESP32 gagal terkoneksi dengan wifi selama 1 menit 2.1 ESP32 gagal mengkoneksikan ke MQTT broker 2.1.1 Menampilkan pesan gagal terkoneksi 2.1.2 Menghubungkan ulang ke MQTT broker 2.1.3 ESP32 gagal terkoneksi dengan MQTT broker selama 1 menit
Include	Suhu Udara dan Kelembaban Udara

Tabel 5.3 Use Case Testing Pengiriman Nilai Sensor

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	ESP32 mengkoneksikan dengan jaringan wifi	ESP32 dapat terhubung dengan jaringan wifi	Sesuai yang diharapkan	Valid
2.	ESP32 mengkoneksikan ke MQTT broker	ESP32 berhasil terkoneksi dengan MQTT broker	Sesuai yang diharapkan	Valid
3.	ESP32 mengirimkan data sensor ke Raspberry Pi	ESP32 dapat melakukan pengiriman data ke Raspberry Pi menggunakan MQTT	Sesuai yang diharapkan	Valid

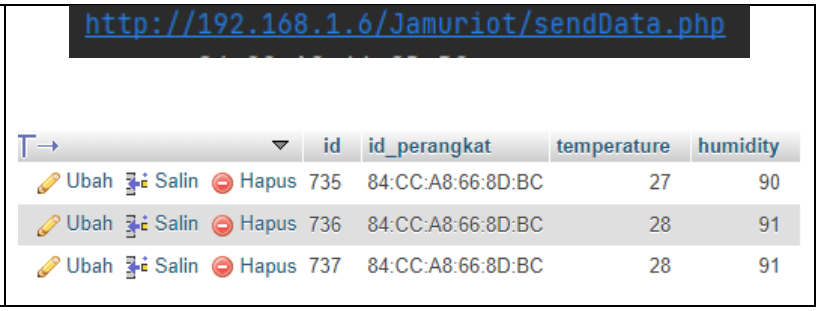
Gambar	

d. Manajemen Database

Nama kolom	Keterangan
Id	#4
Nama	Manajemen Database
Deskripsi	Use case ini mendeskripsikan tentang proses insert pada database
Goals	Sistem dapat menyimpan data ke database
Actor	Sistem
Normal Flow	1. Sistem menyimpan data ke dalam database
Alternative Flow	1. Sistem gagal menyimpan data sensor
Include	Pengiriman Nilai Sensor

Tabel 5.4 Use Case Testing Manajemen Database

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	Sistem menyimpan data sensor	Sistem dapat menyimpan data sensor pada tabel monitoring	Sesuai yang diharapkan	Valid

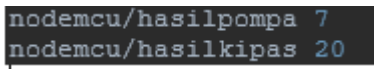
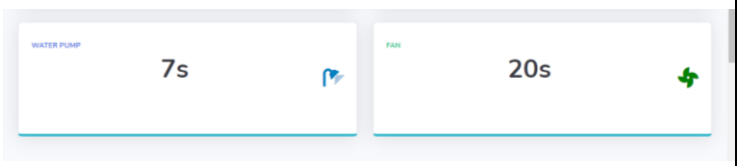
Gambar	

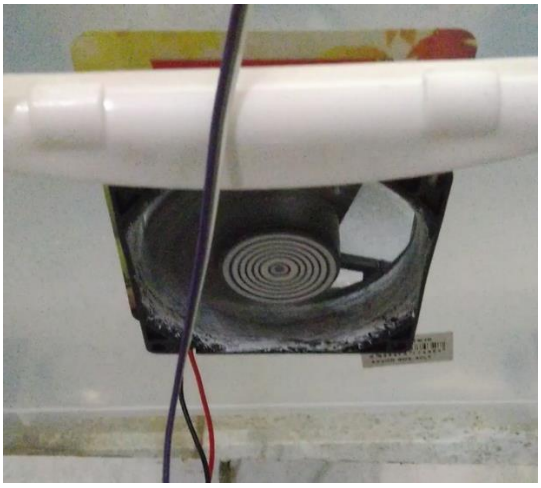
e. Perhitungan Fuzzy Sugeno

Nama kolom	Keterangan
Id	#5
Nama	Perhitungan Fuzzy Sugeno
Deskripsi	Use case ini mendeskripsikan tentang proses perhitungan durasi penyiraman menggunakan inputan suhu udara dan kelembaban udara.
Goals	Sistem dapat menghitung dan memberikan keputusan durasi penyiraman dan durasi kipas dengan baik
Actor	Sistem dan MySQL
Normal Flow	1. Sistem melakukan pengelompokkan data menggunakan himpunan fuzzy 2. Sistem melihat data inferensi dari database 3. Sistem melakukan perhitungan defuzzyfikasi 4. Sistem memberikan response kepada ESP32
Alternative Flow	3.1 Sistem memberikan keputusan durasi penyiraman dan durasi kipas 4.1 Sistem mengirimkan keputusan durasi penyiraman dan durasi kipas pada ESP32
Include	Manajemen Database

Tabel 5.5 Use Case Testing Perhitungan Fuzzy Sugeno

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	Sistem melakukan pengelompokka	Sistem dapat mengelompokkan data	Sesuai yang diharapkan	Valid

	n data menggunakan himpunan fuzzy	menggunakan himpunan fuzzy dengan tepat		
2.	Sistem melihat data inferensi dari database	Sistem dapat memberikan rule aksi yang harus dilakukan sistem	Sesuai dengan yang diharapkan	Valid
3.	Sistem melakukan perhitungan defuzzyfikasi	Sistem dapat melakukan perhitungan defuzzyfikasi	Seusai dengan yang diharapkan	Valid
4.	Sistem memberikan response ke ESP32	Sistem dapat memberikan respon balik ke ESP32	Sesuai dengan yang diharapkan	Valid
Gambar		  Ketika hasil defuzzyfikasi dikirim maka relay akan hidup.		

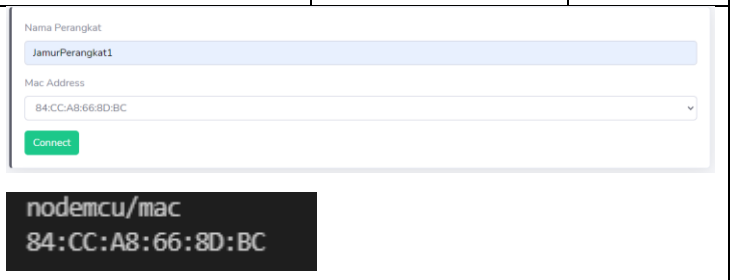
	  

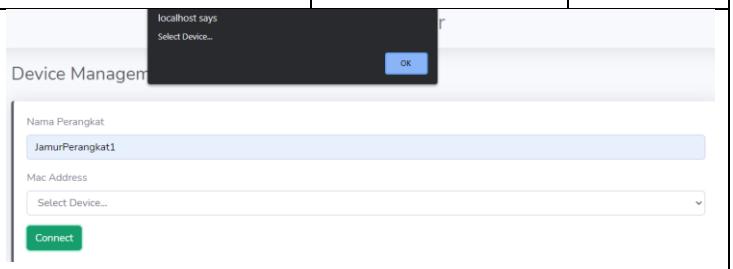
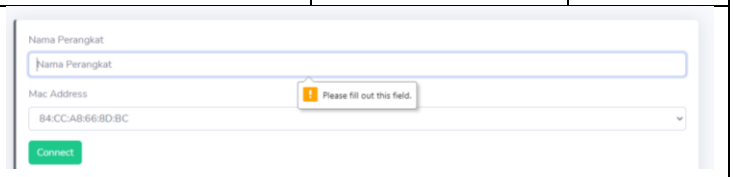
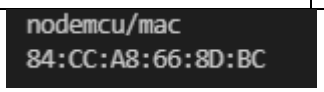
f. Connect/Disconnect Perangkat

Nama kolom	Keterangan
Id	#6
Nama	Connect/Disconnect Perangkat
Deskripsi	Use case ini mendeskripsikan tentang pengujian connect/disconnect perangkat dengan website
Goals	ESP32 dapat terhubung ataupun tidak dengan website
Actor	User, Sistem, dan ESP32
Normal Flow	1. User menginputkan nama dan memilih device yang akan dihubungkan ke website 2. User melakukan klik button “Connect!” 3. Sistem memberikan respon balik ke ESP32 4. ESP32 terhubung dengan website

Alternative Flow	1.1 Sistem berhasil memberikan respon balik ke ESP32 1.1.1 ESP32 menerima pesan mac address dan “1” 1.2 Sistem gagal memberikan respon balik ke ESP32 1.2.1 Memunculkan validasi apabila ada form input yang belum terisi 1.2.2 Memunculkan alert apabila perangkat yang dipilih sudah terhubung dengan website 1.1 ESP32 menerima pesan MAC Address dan “1” 4.1.1 MAC Address sesuai dengan yang dimiliki ESP32 dan melakukan publish data 4.1.2 MAC Address tidak sesuai dengan yang dimiliki ESP32, tidak melakukan publish data
Include	-

Tabel 5.6 Use Case Testing Connect/Disconnect Perangkat

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	User melakukan input data nama dan memilih device, kemudian menekan button “Connect!”	Mengirimkan pesan “1” dan MAC Address	Sesuai yang diharapkan	Valid
	Gambar			
2.	User melakukan input data nama dan tidak memilih device kemudian	Sistem memunculkan pesan bahwa form tidak boleh ada yang kosong	Seusai dengan yang diharapkan	Valid

	menekan button "Connect!”			
	Gambar			
3.	User melakukan tidak melakukan input data nama dan memilih device kemudian menekan button “Connect!”	Sistem memunculkan pesan bahwa form tidak boleh ada yang kosong	Sesuai dengan yang diharapkan	Valid
	Gambar			
4.	Sistem memberikan respon balik ke ESP32	Sistem dapat memberikan respon balik ke ESP32	Sesuai dengan yang diharapkan	Valid
	Gambar			
5.	ESP32 terhubung ke website	ESP32 dapat terhubung ke website	Sesuai dengan yang diharapkan	Valid
	Gambar			

g. Monitoring Suhu Udara

Nama kolom	Keterangan
Id	#7
Nama	Monitoring Suhu Udara
Deskripsi	Use case ini mendeskripsikan tentang proses user memonitoring suhu udara
Goals	Sistem dapat memberikan data suhu udara secara realtime
Actor	Sistem dan User
Normal Flow	1. Sistem menampilkan halaman dashboard 2. Sistem menampilkan data suhu udara berupa angka dan dalam bentuk grafik
Alternative Flow	-
Include	Connect/Disconnect Perangkat

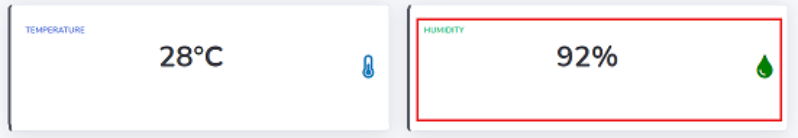
Tabel 5.7 Use Case Testing Monitoring Suhu Udara

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	Tampilkan halaman dashboard	Sistem dapat menampilkan halaman dashboard	Sesuai yang diharapkan	Valid
2.	Pilih Perangkat dan Klik button “Select”	Sistem dapat menampilkan data suhu udara dan grafik secara real time	Sesuai dengan yang diharapkan	Valid
Gambar				

h. Monitoring Kelembaban Udara

Nama kolom	Keterangan
Id	#8
Nama	Monitoring Kelembaban Udara
Deskripsi	Use case ini mendeskripsikan tentang proses user memonitoring kelembaban udara
Goals	Sistem dapat memberikan data kelembaban udara secara realtime
Actor	Sistem dan User
Normal Flow	1. Sistem menampilkan halaman dashboard 2. Sistem menampilkan data kelembaban udara berupa angka dan dalam bentuk grafik
Alternative Flow	-
Include	Connect/Disconnect Perangkat

Tabel 5.8 Use Case Testing Monitoring Kelembaban Udara

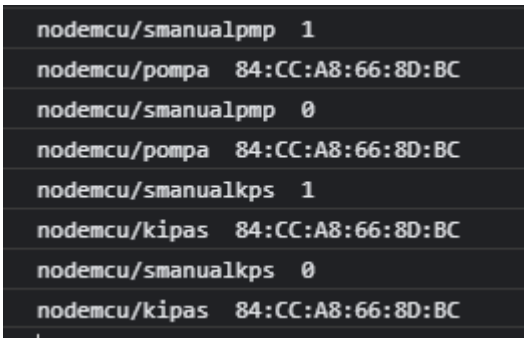
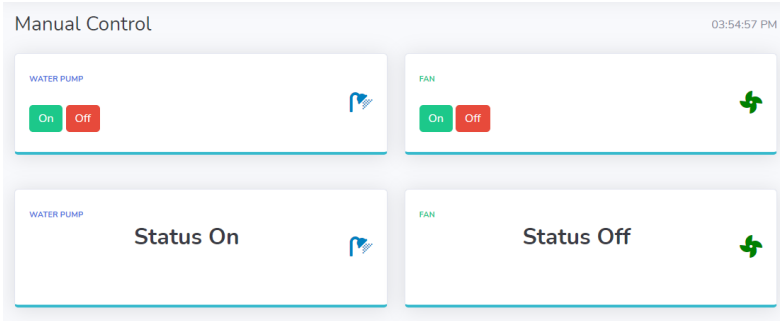
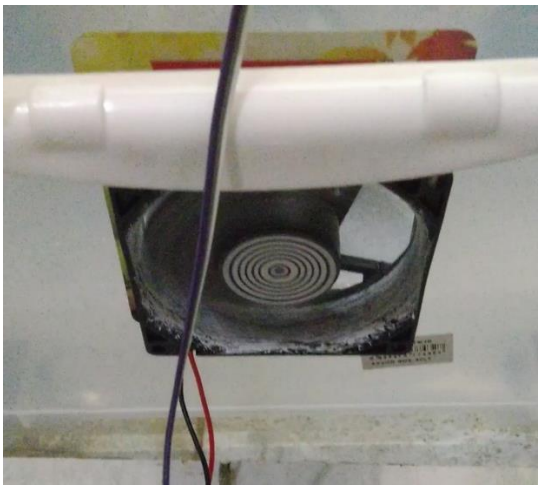
No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	Tampilkan halaman dashboard	Sistem dapat menampilkan halaman dashboard	Sesuai yang diharapkan	Valid
2.	Pilih Perangkat dan Klik button “Select”	Sistem dapat menampilkan data kelembaban udara dan grafik secara real time	Sesuai dengan yang diharapkan	Valid
Gambar				

i. Manual Control Aktuator

Nama kolom	Keterangan

Id	#9
Nama	Manual Control Aktuator
Deskripsi	Use case ini mendeskripsikan tentang proses user melakukan manual control aktuator
Goals	Sistem dapat menghidupkan atau mematikan aktuator
Actor	User, Sistem dan ESP32
Normal Flow	1. Sistem menampilkan halaman dashboard 2. User melakukan manual kontrol aktuator
Alternative Flow	-
Include	Connect/Disconnect Perangkat

Tabel 5.9 Use Case Testing Manual Control Aktuator

No	Test Case	Hasil yang diharapkan	Hasil yang didapatkan	Status
1.	Tampilkan halaman dashboard	Sistem dapat menampilkan halaman dashboard	Sesuai yang diharapkan	Valid
2.	Pilih Perangkat dan Klik button “Select”. Klik button ON/OFF pada card Water Pump dan Fan	User dapat melakukan manual control actuator dan sistem akan menampilkan status dari pompa atau kipas	Sesuai dengan yang diharapkan	Valid

Gambar