

## BAB V. IMPLEMENTASI DAN PENGUJIAN

### 5.1 Implementasi

Implementasi merupakan penerapan terhadap rancangan dari konsep – konsep yang telah direncanakan sebelumnya. Pada bagian ini menjelaskan mengenai implementasi perangkat keras, implementasi database, implementasi antarmuka pengguna dan implementasi bahasa program.

#### 5.1.1 Implementasi Perangkat Keras

Pada implementas perangkat keras tahap implementasi terdiri dari 2 bagian meliputi implementasi prototipe dan implementasi embedded, penjelasan sebagai berikut.

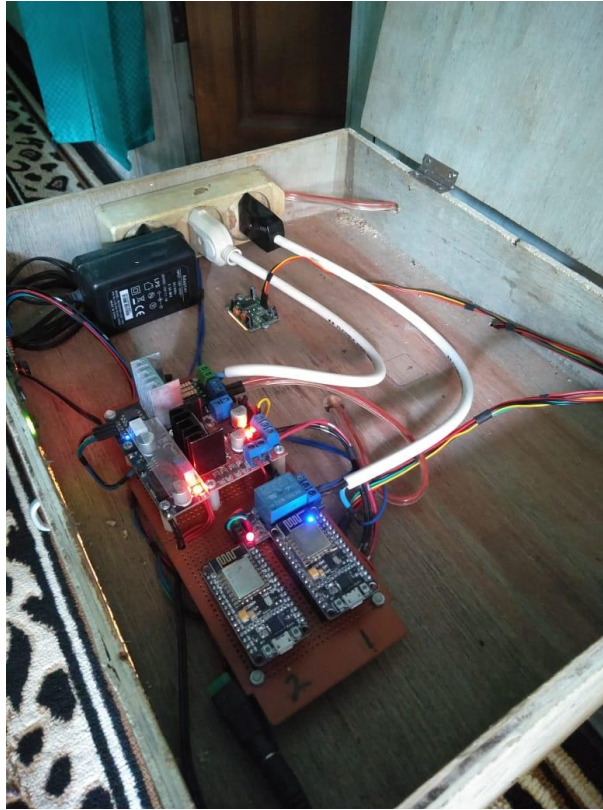
##### 5.1.1.1 Implementasi Prototipe

Prototipe inkubator dibuat berbentuk kubus menggunakan bahan dasar kayu triplek dengan ukuran 40cm x 30cm x 50cm. Pada bagian pintu inkubator diberi lubang dan dipasang akrilik berukuran 33cm x 20cm guna agar telur dapat dilihat dari luar. Pada bagian atas terdapat ruang berukuran 40cm x 30cm x 6cm guna untuk menempatkan sistem embedded didesain dapat terbuka dan tertutup. Seperti yang ditunjukkan pada Gambar 5.1.



Gambar 5. 1 Implementasi Prototype Inkubator

Penempatan alat ditempatkan pada bagian atas inkubator, atap didesain dapat terbuka dan tertutup agar memudahkan jika terjadi masalah teknis pada sistem embedded. Berikut adalah peletakan alat yang ditunjukkan pada Gambar



Gambar 5. 2 Penempatan *Embedded System* pada Prototype

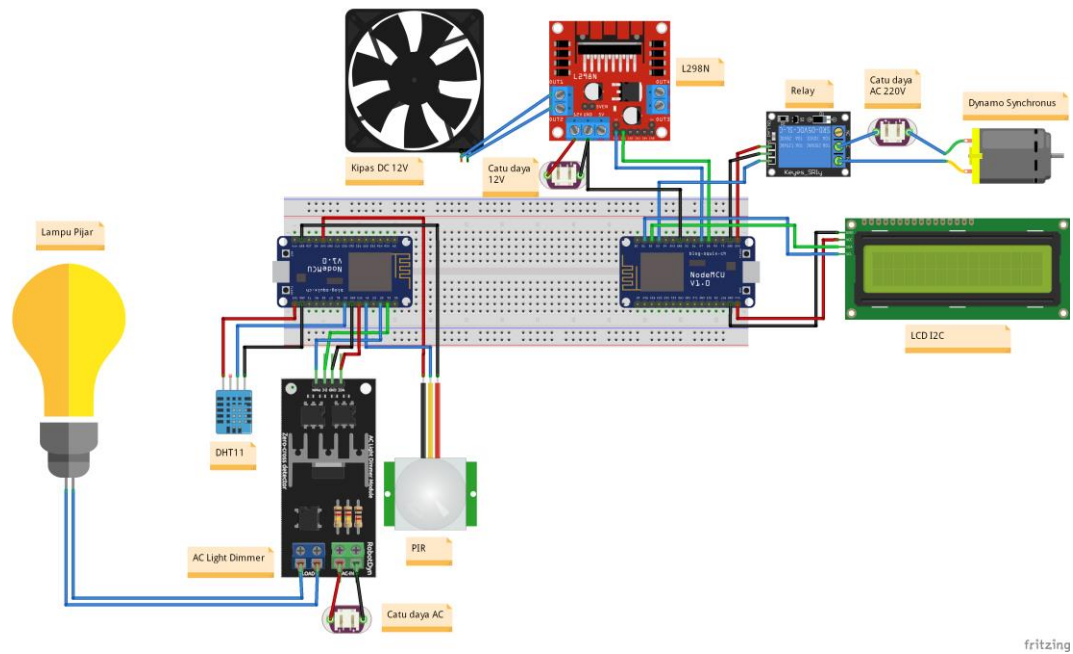
Penempatan sensor dan aktuator berada dalam ruang inkubator sensor dht11 ditempatkan didekat telur puyuh, sensor pir ditempatkan diatas atap menghadap kebawah, aktuator lampu berada diatas menghadap kebawah, kipas ditempatkan pada dinding belakang dalam inkubator, dan *dynamo synchronus* ditempatkan menempel pada dinding bagian samping dalam inkubator. Berikut adalah peletakan sensor dan aktuator dalam inkubator yang ditunjukkan pada Gambar



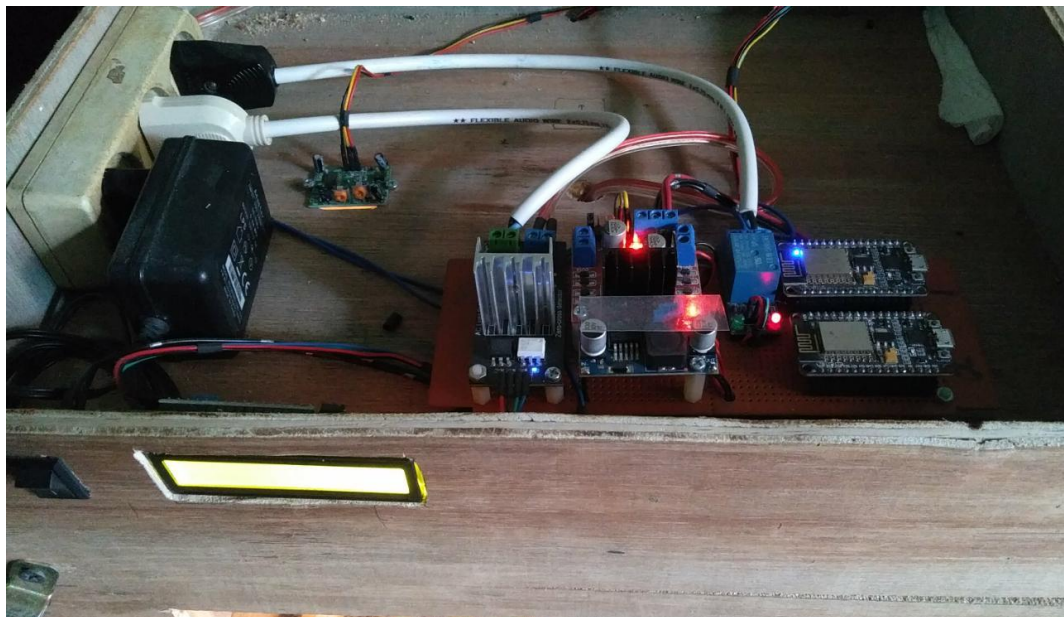
Gambar 5. 3 Penempatan Sensor dan Aktuator Pada Prototipe

#### 5.1.1.2 Implementasi Embedded Sistem

Implementasi embedded sistem merupakan pembuatan dari beberapa rangkaian yang digabung menjadi satu kesatuan. Embedded sistem terdiri dari 2 buah NodeMCU, 2 buah sensor yaitu dht11 dan pir, relay 1 channel, *AC light dimmer*, modul i298n, modul *stepdown* 5v, lcd I2C, *dynamo synchronus*, 1 buah kipas dc 12v, 1 buah lampu, dan adaptor 12v. Rangkaian embedded sistem ditunjukkan pada Gambar 5.4 dan Implementasi embedded sistem ditunjukkan pada Gambar 5.5.



Gambar 5. 4 Rangkaian Embedded Sistem



Gambar 5. 5 Implementasi Embedded System pada Prototype

### 5.1.2 Implementasi Database

Implementasi database dilakukan dengan menggunakan rancangan database yang telah dibuat sebelumnya. Dibuat sebuah database dengan nama 'smart\_inkubator' dan memiliki 3 tabel meliputi tabel 'device', tabel 'manage', dan tabel 'sensor'. Pada implementasi ini menggunakan MySQL sebagai database dan PhpMyAdmin sebagai tool untuk mengelola database dalam *localhost*. Struktur

tabel 'device' ditunjukkan pada Gambar 5.6, struktur tabel 'manage' ditunjukkan pada Gambar 5.7, dan struktur tabel 'sensor' ditunjukkan pada Gambar 5.8.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	mac_address	varchar(25)	latin1_swedish_ci	No	None			Change  Drop  More
<input type="checkbox"/>	3	nama_device	varchar(11)	latin1_swedish_ci	No	None			Change  Drop  More
<input type="checkbox"/>	4	topicCol1	varchar(25)	latin1_swedish_ci	No	None			Change  Drop  More

Gambar 5. 6 Gambar Struktur Tabel Device

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	jenis_telur	varchar(11)	latin1_swedish_ci	No	None			Change  Drop  More
<input type="checkbox"/>	3	banyak_telur	int(11)		No	None			Change  Drop  More
<input type="checkbox"/>	4	tanggal_masuk	date		No	None			Change  Drop  More
<input type="checkbox"/>	5	tanggal_menetas	date		No	None			Change  Drop  More

Gambar 5. 7 Gambar Struktur Tabel Manage

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	suhu	float		No	None			Change  Drop  More
<input type="checkbox"/>	3	kelembapan	float		No	None			Change  Drop  More
<input type="checkbox"/>	4	fuzzy_lampu	float		No	None			Change  Drop  More
<input type="checkbox"/>	5	fuzzy_kipas	float		No	None			Change  Drop  More
<input type="checkbox"/>	6	waktu	time		No	None			Change  Drop  More
<input type="checkbox"/>	7	tanggal	date		No	None			Change  Drop  More

Gambar 5. 8 Gambar Struktur Tabel Sensor

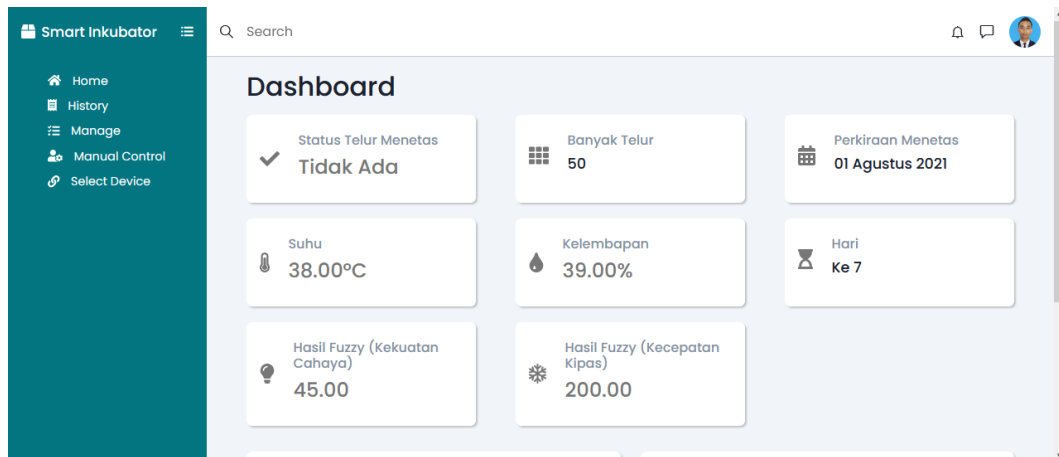
### 5.1.3 Implementasi Antarmuka Pengguna

Implementasi dari tampilan antarmuka pengguna atau *userinterface* dibuat menggunakan bahasa pemrograman PHP dan dikombinasikan dengan HTML, CSS, dan JavaScript untuk mendukung tampilan *website* agar dapat diakses dengan nyaman melalui *web browser* dan mobile.

#### 5.1.3.1 Halaman *Dashboard*

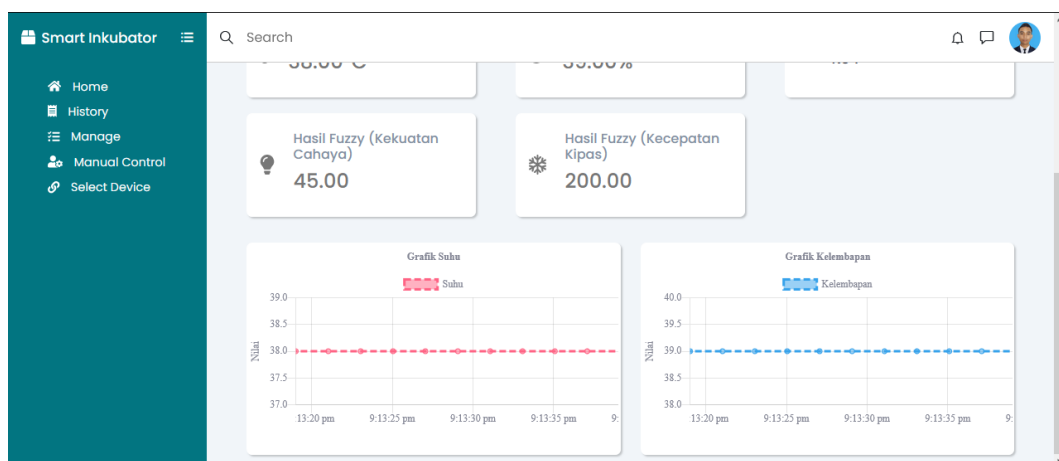
Pada halaman *dashboard* menampilkan beberapa informasi, pada bagian atas menampilkan status telur menetas, suhu, kelembapan, kekuatan lampu (hasil *fuzzy* lampu), dan kecepatan kipas (hasil *fuzzy* kipas) pada ruang inkubator secara *realtime* juga terdapat informasi tambahan seperti banyak telur, tanggal perkiraan menetas, , hari ke- i, yang diambil dari database. Halaman *dashboard* bagian atas ditunjukkan pada Gambar 5.9.





Gambar 5. 9 Halaman *Dashboard* Bagian Atas

Pada halaman *dashboard* bagian bawah menampilkan 2 buah chart dari data suhu dan kelembapan *realtime* pada ruang inkubator. Halaman *dashboard* bagian bawah ditunjukkan pada Gambar 5.10.



Gambar 5. 10 Halaman *Dashboard* Bagian Bawah

### 5.1.3.2 Halaman *History*

Pada halaman *history* menampilkan informasi data yang tersimpan dalam database selama ruang inkubator menyala. Informasi yang ditampilkan meliputi suhu, kelembapan, kekuatan lampu, kecepatan kipas, waktu, dan tanggal dalam bentuk tabel. Data tersebut ditampilkan agar pengguna dapat mengetahui dan melihat *record* data sebelumnya. Halaman *history* data ditunjukkan pada Gambar 5.11.

No ↑↓	Suhu ↑↓	Kelembapan ↑↓	Kekuatan Lampu ↑↓	Kecepatan Kipas ↑↓	Waktu ↑↓	Tanggal ↑↓
1	37	49	50.33	235.2	21:52:41	16 Juli 2021
2	37	47	50.33	228.27	22:07:42	16 Juli 2021
3	37	45	50.33	221.33	22:22:43	16 Juli 2021
4	38	45	50.33	217.87	22:37:43	16 Juli 2021
5	38	45	45	216	22:52:44	16 Juli 2021

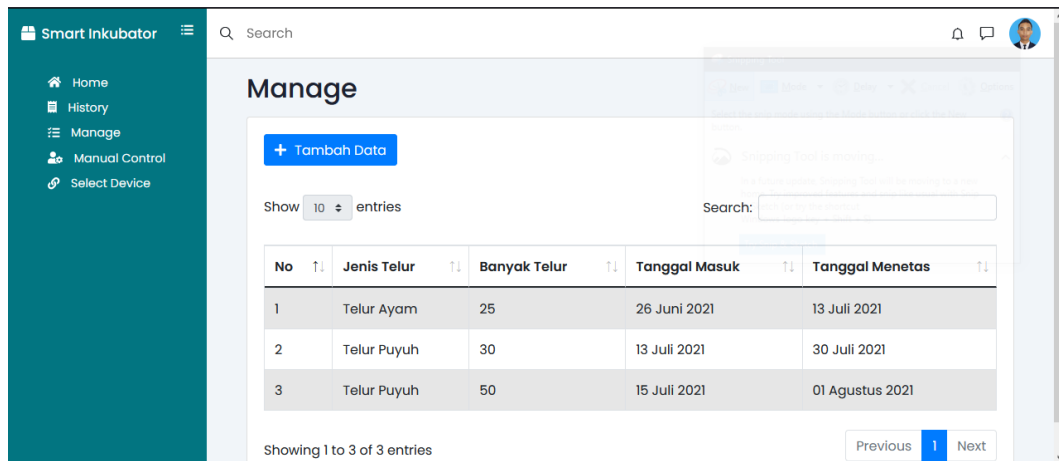
Gambar 5. 11 Halaman *History* Data

### 5.1.3.3 Manage

Pada halaman *manage* menampilkan informasi data mengenai telur yang telah masuk dalam ruang inkubator. Ketika pengguna memasukkan telur baru kedalam ruang inkubator, pengguna dapat menambahkan data mengenai telur yang akan dimasukkan guna sebagai arsip dan catatan. Penambahan data dengan cara menekan *button* tambah data kemudian akan muncul dialog form seperti pada Gambar 5.12.

Gambar 5. 12 Dialog Form

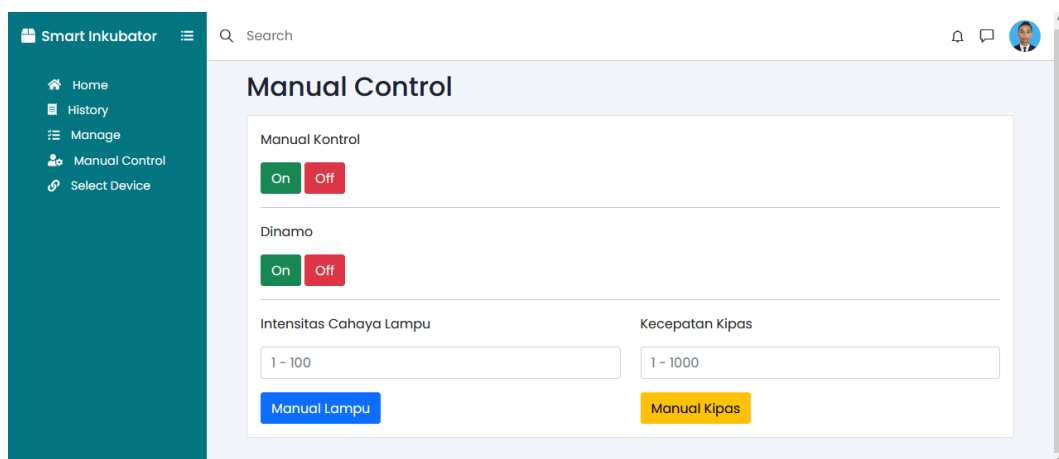
Data tersebut akan dimasukkan kedalam database. Informasi data yang ditampilkan meliputi jenis telur, banyak telur, tanggal masuk, dan tanggal menetas. Halaman manage ditunjukkan pada Gambar 5.13.



Gambar 5. 13 Halaman *Manage*

#### 5.1.3.4 Manual Control

Pada halaman manual kontrol terdapat button on/off untuk mengaktifkan dan mematikan kontrol manual terhadap alat. Kontrol manual meliputi kontrol on/off dinamo, intensitas cahaya lampu, dan kecepatan kipas. Halaman manual kontrol ditunjukkan pada Gambar 5.14.

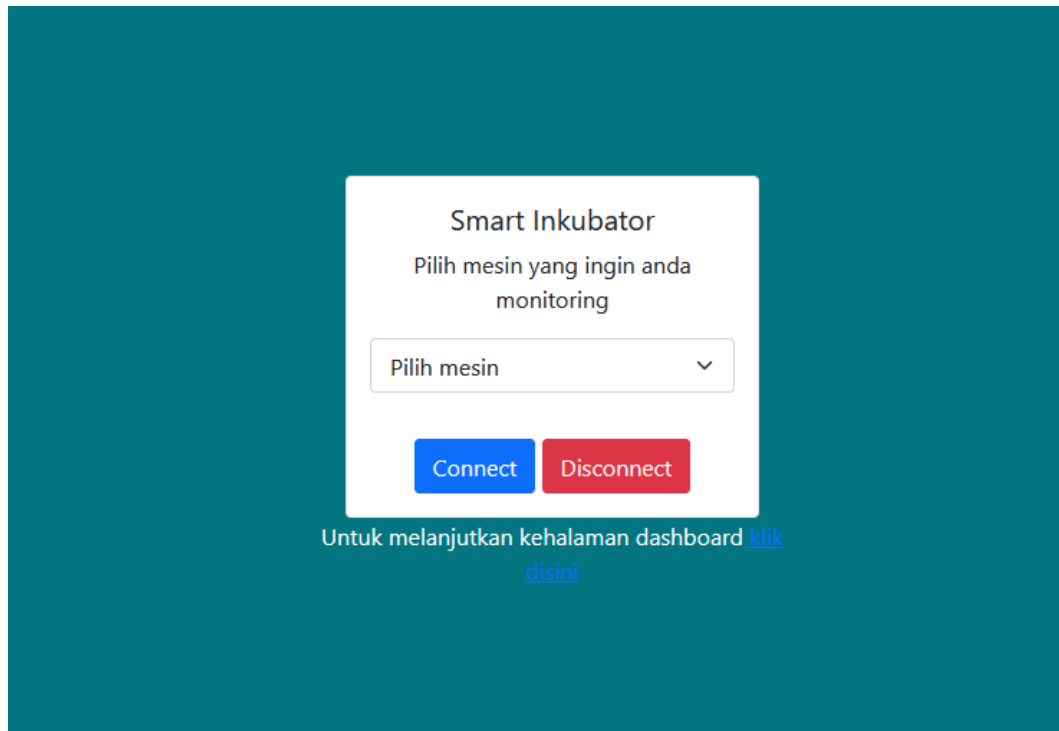


Gambar 5. 14 Halaman *Manual Control*



### 5.1.3.5 Select Device

Pada halaman *select device* pengguna dapat melakukan *connect/disconnect device* dengan cara memilih *device* mana yang akan dimonitoring pada form dan *connect* maka data informasi akan muncul pada *dashboard* sesuai dengan mesin yang dipilih. Halaman *select device* ditunjukkan pada Gambar 5.15.



Gambar 5. 15 Halaman *Select Device*

### 5.1.4 Implementasi Kode Program

Pada bagian implementasi kode program berisi *sourcecode* dari perangkat sensor, proses pengolahan data dan konfigurasi sistem sehingga hasil pembacaan dapat di analisa.

#### 5.1.4.1 Kode Program Pembacaan Sensor dan *Publish/Subscribe* Data

Kode program pembacaan sensor digunakan untuk membaca data yang kemudian akan diolah oleh mikrokontroller, bersamaan dengan itu data hasil pembacaan akan di-*publish* menggunakan protokol mqtt. Berikut potongan kode program pembacaan sensor dan *publish/subscribe* data ditunjukkan pada Tabel 5.1.

Tabel 5. 1 Kode Program Pembacaan Sensor dan *Publish/Subscribe* Data

```
#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <SimpleTimer.h>;
```

```

#include <RBDdimmer.h>

dht DHT;

void BacaDHT() {
  DHT.read11(sensor);
  suhu = DHT.temperature;
  rh = DHT.humidity;
  Serial.print("Kelembaban udara = ");
  Serial.print(rh);
  Serial.print("% ");
  Serial.print("Suhu = ");
  Serial.print(suhu);
  Serial.println(" C ");

  //Publish Mqtt
  String Ssuhu = String(suhu);
  String Skelembapan = String(rh);
  tempSuhu = Ssuhu;
  tempKelembapan = Skelembapan;
  client.publish(connectTopicSuhu, tempSuhu.c_str(), true);
  client.publish(connectTopicKelembapan, tempKelembapan.c_str(),
true);
}

void DeteksiTelur() {
  int val = digitalRead(PIR_MOTION_SENSOR);
  String txtHigh = "Ada";
  String txtLow = "Tidak Ada";
  Serial.println(PIR_MOTION_SENSOR);
  if (val == HIGH) {
    Serial.println("Motion detected!");
    // String StrPirHigh = String(outLampu);
    client.publish(connectTopicPIR, txtHigh.c_str(), true);
  } else {
    Serial.println("No Motion detected!");
    client.publish(connectTopicPIR, txtLow.c_str(), true);
  }
}

void setup() {
  Serial.begin(115200);
  pinMode(PIR_MOTION_SENSOR, INPUT);
  setup_wifi();
  client.setServer(mqtt_server, port);
  client.setCallback(MQTTcallback);
  //Dimmer
  dimmer.begin(NORMAL_MODE, ON);
  WaktuBacaDHT.setInterval(2000);
  WaktuFuzzy.setInterval(3000);
  WaktuDeteksiTelur.setInterval(1000);
  WaktuLampu.setInterval(1000);
  WaktuManual.setInterval(2000);
  WaktuKirimDataSql.setInterval(60000 * 15);
}

void loop() {
  if (!client.connected()) {
    reconnect();
  }
}

```

```

} else {
  client.loop();
  if (WaktuManual.isReady()) {
    if (pesanManualValue == 1) {
      if (pesanManualLampu > 0) {
        int outVal = pesanManualLampu;
        dimmer.setPower(outVal);
      }
    } else {
      if (WaktuBacaDHT.isReady()) {
        BacaDHT();
        WaktuBacaDHT.reset();
      }
      if (WaktuFuzzy.isReady()) {
        Fuzzy();
        WaktuFuzzy.reset();
      }
      if (WaktuDeteksiTelur.isReady()) {
        DeteksiTelur();
        WaktuDeteksiTelur.reset();
      }

      if (WaktuLampu.isReady()) {
        Lampu();
        WaktuLampu.reset();
      }

      if (WaktuKirimDataSql.isReady()) {
        KirimDataSql();
        WaktuKirimDataSql.reset();
      }
    }
  }
}
}

```

#### 5.1.4.2 Kode Program Koneksi Wifi

Kode program koneksi wifi digunakan agar mikrokontroller dapat terkoneksi dengan wifi yang dituju dan dapat melakukan pengiriman data. Berikut potongan kode program koneksi wifi ditunjukkan pada Tabel 5.2.

Tabel 5. 2 Kode Program Koneksi Wifi

```

#include <ESP8266WiFi.h>

const char* ssid = "Wifi Id";
const char* password_wifi = "semangatbaru";

void setup_wifi() {
  //setting koneksi
  WiFi.hostname("NodeMCU");
  WiFi.begin(ssid, password_wifi);

  //cek koneksi
  while (WiFi.status() != WL_CONNECTED)

```

```

{
  Serial.print(".");
  delay(500);
}

//jika berhasil konek
// digitalWrite(PIN_LED, HIGH);
Serial.println("WiFi connected");
Serial.println(WiFi.localIP());
}

void setup() {
  Serial.begin(115200);
  setup_wifi();
}

```

#### 5.1.4.3 Kode Program MQTT pada Embedded

Kode program mqtt pada embedded digunakan untuk pengiriman data sensor maupun data perhitungan kepada subscriber menggunakan protokol mqtt dan melalui broker. Potongan kode program mqtt pada embedded ditunjukkan pada Tabel 5.3.

Tabel 5. 3 Kode Program MQTT pada Embedded

```

#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <SimpleTimer.h>;
#include <RBDDimmer.h>

const char* ssid = "Wifi Id";
const char* password_wifi = "semangatbaru";

#define mqtt_server "u70d942b.ap-southeast-1.emqx.cloud"
#define port 15861

#define username "admin"
#define password "admin"
#define clientNr "01"

// Some project settings
#define slash "/"
#define topicConnect "/ESP8266/connected"
#define topicIn "/ESP8266/inTopic"
#define GPIO0 0

#define concat(first, second) first second

const char* connectTopicSuhu = "stwo.royalheros@gmail.com/suhu";
const char* connectTopicKelembapan =
"stwo.royalheros@gmail.com/kelembapan";
const char* connectTopicFuzzyLampu =
"stwo.royalheros@gmail.com/fuzzyLampu";
const char* connectTopicFuzzyKipas =
"stwo.royalheros@gmail.com/fuzzyKipas";

```

```

const char* connectTopicPIR = "stwo.royalheros@gmail.com/PIR";
const char* connectTopicManualLampu =
"stwo.royalheros@gmail.com/manualLampu";
const char* connectTopicDevice =
"stwo.royalheros@gmail.com/device";
const char* connectTopicNamaDevice =
"stwo.royalheros@gmail.com/namaDevice";
const char* inTopic = concat(concat(slash, username), topicIn);
const char* client_id = "NodeMCU1";
WiFiClient espClient;
PubSubClient client(espClient);
char msg[50];
int value = 0;

void MQTTcallback(char* topic, byte* payload, unsigned int
length) {
  if (strcmp(topic, "stwo.royalheros@gmail.com/manualLampu") ==
0) {
    for (int i = 0; i < length; i++) {
      Serial.print((char)payload[i]);
      payload[length] = '\0';
      pesanManualLampu = atoi((char* )payload);
    }
  }

  if (strcmp(topic, "stwo.royalheros@gmail.com/manualValue") ==
0) {
    for (int i = 0; i < length; i++) {
      Serial.print((char)payload[i]);
      payload[length] = '\0';
      pesanManualValue = atoi((char* )payload);
    }
  }
}

void reconnect() {
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    ++value;
    if (client.connect(client_id, username, password)) {
      String macAdd = WiFi.macAddress();
      String namaDevice = "Mesin 1";
      Serial.println("connected");
      client.publish(connectTopicDevice, macAdd.c_str(), true);
      client.publish(connectTopicNamaDevice, namaDevice.c_str(),
true);
      client.subscribe("stwo.royalheros@gmail.com/manualValue");
      client.subscribe("stwo.royalheros@gmail.com/manualLampu");
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      delay(5000);
    }
  }
}

void setup() {
  Serial.begin(115200);

```

```

    setup_wifi();
    client.setServer(mqtt_server, port);
    client.setCallback(MQTTcallback);
}

void loop() {
    if (!client.connected()) {
        reconnect();
    }
}

```

#### 5.1.4.4 Kode Program MQTT pada *Website*

Kode program mqtt pada *website* digunakan untuk menerima dan mengirim data. Data yang diterima dapat secara langsung ditampilkan pada halaman utama *website*. Data yang dikirim dapat direspon oleh device lain yang mensubscribe topik yang sama. Berikut potongan kode program mqtt pada *website* ditunjukkan pada Tabel 5.4.

Tabel 5. 4 Kode Program MQTT pada *Website*

```

var messagePayloadSuhu = 0;
var messagePayloadKelembapan = 0;
var messagePayloadFuzzyLampu = 0;
var messagePayloadFuzzyKipas = 0;
var messagePayloadDevice;
var messagePayloadNamaDevice;
var macAddress;
var namaDevice;

// var messagePayloadKelembapan = 0;
// Called after form input is processed
function startConnect() {
    // Generate a random client ID
    clientID = "mqttProses";

    // Fetch the hostname/IP address and port number from the
    form
    // host = "maqiatto.com";
    host = "u70d942b.ap-southeast-1.emqx.cloud";
    // port = 8883;           //port maqiatto
    // port = 9001;           //port mosquitto local
    port = 8083;

    // Initialize new Paho client connection
    client = new Paho.MQTT.Client(host, Number(port), clientID);

    // Connect the client, if successful, call onConnect
    function
    client.connect({
        userName: "admin",
        password: "admin",
        onSuccess: onConnect,
    });
}

```



```

    // Set callback handlers
    client.onConnectionLost = onConnectionLost;
    client.onMessageArrived = onMessageArrived;
}

// Called when the client connects
function onConnect() {
    // Fetch the MQTT topic from the form
    topicSuhu = "stwo.royalheros@gmail.com/suhu";
    topicKelembapan = "stwo.royalheros@gmail.com/kelembapan";
    topicFuzzyLampu = "stwo.royalheros@gmail.com/fuzzyLampu";
    topicFuzzyKipas = "stwo.royalheros@gmail.com/fuzzyKipas";
    topicPIR = "stwo.royalheros@gmail.com/PIR";
    topicManualValue = "stwo.royalheros@gmail.com/manualValue";
    topicDevice = "stwo.royalheros@gmail.com/device";
    topicNamaDevice = "stwo.royalheros@gmail.com/namaDevice";

    // Print output for the user in the messages div
    // document.getElementById("pesansuhu").innerHTML +=
    '<span>Subscribing to: ' + topic + '</span><br/>';

    // Subscribe to the requested topic
    client.subscribe(topicSuhu);
    client.subscribe(topicKelembapan);
    client.subscribe(topicFuzzyLampu);
    client.subscribe(topicFuzzyKipas);
    client.subscribe(topicPIR);
    client.subscribe(topicManualValue);
    client.subscribe(topicDevice);
    client.subscribe(topicNamaDevice);
}

// Called when the client loses its connection
function onConnectionLost(responseObject) {
    console.log("onConnectionLost: Connection Lost");
    if (responseObject.errorCode !== 0) {
        console.log("onConnectionLost: " +
responseObject.errorMessage);
    }
}

// Called when a message arrives
// var suhu = "";
// var kelembapan = "";
function onMessageArrived(message) {
    // console.log("onMessageArrived: " +
message.payloadString);
    if (message.destinationName == topicSuhu) {
        // suhu = message.payloadString;
        document.getElementById("suhu").innerHTML = '<span>' +
message.payloadString + '&degC</span>';
        console.log("onMessageArrivedSuhu: " +
message.payloadString);
        messagePayloadSuhu = parseFloat(message.payloadString);
    }
    if (message.destinationName == topicKelembapan) {
        // kelembapan = message.payloadString;
        document.getElementById("kelembapan").innerHTML =
'<span>' + message.payloadString + '%</span>';
    }
}

```

```

        console.log("onMessageArrivedKel: " +
message.payloadString);
        messagePayloadKelembapan =
parseFloat(message.payloadString);
    }
    if (message.destinationName == topicFuzzyLampu) {
        // kelembapan = message.payloadString;
        document.getElementById("fuzzyLampu").innerHTML =
'<span>' + message.payloadString + '</span>';
        console.log("onMessageArrivedFuzzyLampu: " +
message.payloadString);
        messagePayloadFuzzyLampu =
parseFloat(message.payloadString);
    }
    if (message.destinationName == topicFuzzyKipas) {
        // kelembapan = message.payloadString;
        document.getElementById("fuzzyKipas").innerHTML =
'<span>' + message.payloadString + '</span>';
        console.log("onMessageArrivedFuzzyKipas: " +
message.payloadString);
        messagePayloadFuzzyKipas =
parseFloat(message.payloadString);
    }
    if (message.destinationName == topicPIR) {
        // kelembapan = message.payloadString;
        document.getElementById("PIR").innerHTML = '<span>' +
message.payloadString + '</span>';
        console.log("onMessageArrivedPIR: " +
message.payloadString);
    }

    updateScroll(); // Scroll to bottom of window
}

```

#### 5.1.4.5 Kode Program Koneksi Http Database

Kode program koneksi http database digunakan untuk mengkoneksikan mikrokontroller menuju database dimana data suhu, kelembapan, hasil *fuzzy* lampu, dan hasil *fuzzy* kipas akan disimpan pada database menggunakan protokol. Berikut potongan kode program mqtt pada *website* ditunjukkan pada Tabel 5.5.

Tabel 5. 5 Kode Program Koneksi Http Database

```

#include <ESP8266HTTPClient.h>
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <SimpleTimer.h>;
#include <RBDdimmer.h>

WiFiClient espClient;
PubSubClient client(espClient);
char msg[50]; // message to publish
int value = 0; // connection attempt

//Cek koneksi http
void cekKoneksi() {

```

```

//proses cek koneksi server
WiFiClient client1;
if (!client1.connect(host, 80))
{
  Serial.println("Connection Failed");
  return; //jika tidak terkoneksi ulang dari awal
}
}

void KirimDataSql() {
  //proses pengiriman data ke server
  cekKoneksi();
  String Link;
  HTTPClient http;
  Link = "http://192.168.1.115/Smart-
Inkubator/kirimdata.php?suhu=" + String(suhu) + "&kelembapan=" +
String(rh) + "&fuzzy_lampu=" + String(outLampu) +
"&fuzzy_kipas=" + String(outKipas);
  http.begin(Link);
  //mode
  http.GET();
  http.end();
}

void setup() {
  Serial.begin(115200);
  setup_wifi();
  client.setServer(mqtt_server, port);
  client.setCallback(MQTTcallback);
  WaktuKirimDataSql.setInterval(60000 * 15);
}

void loop() {
  if (!client.connected()) {
    reconnect();
  } else {
    client.loop();
    if (WaktuKirimDataSql.isReady()) {
      KirimDataSql();
      WaktuKirimDataSql.reset();
    }
  }
}

```

#### 5.1.4.6 Kode Program Proses Perhitungan *Fuzzy* dan Publish Data

Kode program proses perhitungan *fuzzy* dan publish data digunakan untuk memproses data dari sensor menggunakan metode *fuzzy* segeno yang mana hasil perhitungan akan di-*publish* sebagai nilai dari aktuator. Berikut potongan kode program proses perhitungan *fuzzy* dan publish data ditunjukkan pada Tabel 5.6.

Tabel 5. 6 Kode Program Proses Perhitungan *Fuzzy* dan Publish Data

```

int outVal = 0;
dimmerLamp dimmer(outputPin, zerocross);

```

```

SimpleTimer WaktuBacaDHT, WaktuFuzzy, WaktuDeteksiTelur,
WaktuKirimDataSql, WaktuLampu, WaktuManual;

String tempSuhu, tempKelembapan, tempLampu, tempKipas;
int pesanManualValue = 0;
int pesanManualLampu = 0;

dht DHT;
//-----
//Variabel Fuzzy
float suhu, rh, outLampu, outKipas;

//Input
float shDingin, shNormal, shPanas;
float rhKering, rhNormal, rhBasah;

//Output
float pelan = 200;
float Kipasnormal = 260;
float cepat = 300;

float terang = 85;
float normal = 45;
float redup = 30;

//Variabel rule fuzzy
float rule1, rule2, rule3, rule4, rule5, rule6, rule7, rule8,
rule9;

// Fuzzyfikasi Suhu
unsigned char suhuDingin() {
    if (suhu < 32) {
        shDingin = 1;
    }
    else if (suhu >= 32 && suhu <= 38) {
        shDingin = (38 - suhu) / (38 - 32);
    }
    else if (suhu > 38) {
        shDingin = 0;
    }
    return shDingin;
}
unsigned char suhuNormal() {
    if (suhu < 32) {
        shNormal = 0;
    }
    else if (suhu >= 32 && suhu <= 38) {
        shNormal = (suhu - 32) / (38 - 32);
    }
    else if (suhu > 38 && suhu <= 44) {
        shNormal = (44 - suhu) / (44 - 38);
    }
    else if (suhu > 44) {
        shNormal = 0;
    }
    return shNormal;
}
unsigned char suhuPanas () {

```

```

    if (suhu < 38 ) {
        shPanas = 0;
    }
    else if (suhu >= 38 && suhu <= 44) {
        shPanas = (suhu - 38) / (44 - 38);
    }
    else if (suhu > 44) {
        shPanas = 1;
    }
    return shPanas;
}

//Fuzzifikasi Kelembaban
unsigned char kelKering () {
    if (rh < 45 ) {
        rhKering = 1;
    }
    else if (rh >= 45 && rh <= 55) {
        rhKering = (55 - rh) / (55 - 45);
    }
    else if (rh > 55) {
        rhKering = 0;
    }
    return rhKering;
}

unsigned char kelNormal () {
    if (rh < 45) {
        rhNormal = 0;
    }
    else if (rh >= 45 && rh <= 55) {
        rhNormal = (rh - 45) / (55 - 45);
    }
    else if (rh > 55 && rh <= 65) {
        rhNormal = (65 - rh) / (65 - 55);
    }
    else if (rh > 65) {
        rhNormal = 0;
    }
    return rhNormal;
}

unsigned char kelBasah () {
    if (rh < 55) {
        rhBasah = 0;
    }
    else if (rh >= 55 && rh <= 65) {
        rhBasah = (rh - 55) / (65 - 55);
    }
    else if (rh > 65) {
        rhBasah = 1;
    }
    return rhBasah;
}

//Fuzzifikasi
void fuzzifikasi() {
    suhuDingin();
    suhuNormal();
    suhuPanas();
    kelKering();
    kelNormal();
}

```

```

    kelBasah();
}
// Rule
void fuzzy_rule () {
    float jml_rule[8];
    float SumA = 0;
    fuzzifikasi();
    if (shDingin >= 0 && rhKering >= 0) {
        rule1 = min(shDingin, rhKering);
        jml_rule[0] = rule1;
    }
    if (shNormal >= 0 && rhNormal >= 0) {
        rule2 = min(shNormal, rhNormal);
        jml_rule[1] = rule2;
    }
    if (shPanas >= 0 && rhBasah >= 0) {
        rule3 = min(shPanas, rhBasah);
        jml_rule[2] = rule3;
    }
    if (shDingin >= 0 && rhNormal >= 0) {
        rule4 = min(shDingin, rhNormal);
        jml_rule[3] = rule4;
    }
    if (shNormal >= 0 && rhKering >= 0) {
        rule5 = min(shNormal, rhKering);
        jml_rule[4] = rule5;
    }
    if (shPanas >= 0 && rhKering >= 0) {
        rule6 = min(shPanas, rhKering);
        jml_rule[5] = rule6;
    }
    if (shDingin >= 0 && rhBasah >= 0) {
        rule7 = min(shDingin, rhBasah);
        jml_rule[6] = rule7;
    }
    if (shNormal >= 0 && rhBasah >= 0) {
        rule8 = min(shNormal, rhBasah);
        jml_rule[7] = rule8;
    }
    if (shPanas >= 0 && rhNormal >= 0) {
        rule9 = min(shPanas, rhNormal);
        jml_rule[8] = rule9;
    }
}

//defuzifikasi lampu
float weightLamp = ((rule1 * terang) + (rule2 * normal) +
(rule3 * redup) + (rule4 * terang) + (rule5 * normal) + (rule6 *
redup) + (rule7 * terang) + (rule8 * normal) + (rule9 * redup));
float averageLamp = ((jml_rule[0] + jml_rule[1] + jml_rule[2]
+ jml_rule[3] + jml_rule[4] + jml_rule[5] + jml_rule[6] +
jml_rule[7] + jml_rule[8] ));
outLampu = weightLamp / averageLamp;

//defuzifikasi kipas
float weightFan = ((rule1 * pelan) + (rule2 * Kipasnormal) +
(rule3 * cepat) + (rule4 * Kipasnormal) + (rule5 * pelan) +
(rule6 * pelan) + (rule7 * cepat) + (rule8 * cepat) + (rule9 *
Kipasnormal));

```



```

    float averageFan = ((jml_rule[0] + jml_rule[1] + jml_rule[2] +
jml_rule[3] + jml_rule[4] + jml_rule[5] + jml_rule[6] +
jml_rule[7] + jml_rule[8] ));
    outKipas = weightFan / averageFan;
}

void Fuzzy() {
    fuzzy_rule();

    //Publish Mqtt
    String StrLampu = String(outLampu);
    String StrKipas = String(outKipas);
    tempLampu = StrLampu;
    tempKipas = StrKipas;
    client.publish(connectTopicFuzzyLampu, tempLampu.c_str(),
true);
    client.publish(connectTopicFuzzyKipas, tempKipas.c_str(),
true);
}

void setup() {
    Serial.begin(115200);
    // USE_SERIAL.begin(9600);
    // pinMode(PIN_LED, OUTPUT);
    pinMode(PIR_MOTION_SENSOR, INPUT);
    setup_wifi();
    client.setServer(mqtt_server, port);
    client.setCallback(MQTTcallback);

    //Dimmer
    dimmer.begin(NORMAL_MODE, ON);

    WaktuBacaDHT.setInterval(2000);
    WaktuFuzzy.setInterval(3000);
    WaktuDeteksiTelur.setInterval(1000);
    WaktuLampu.setInterval(1000);
    WaktuManual.setInterval(2000);
    WaktuKirimDataSql.setInterval(60000 * 15);
}

void loop() {
    if (!client.connected()) {
        reconnect();
    } else {
        client.loop();
        if (WaktuManual.isReady()) {
            if (pesanManualValue == 1) {
                if (pesanManualLampu > 0) {
                    int outVal = pesanManualLampu;
                    dimmer.setPower(outVal);
                }
            } else {
                if (WaktuBacaDHT.isReady()) {
                    BacaDHT();
                    WaktuBacaDHT.reset();
                }
            }
            if (WaktuFuzzy.isReady()) {
                Fuzzy();
            }
        }
    }
}

```

```

        WaktuFuzzy.reset();
    }
    if (WaktuDeteksiTelur.isReady()) {
        DeteksiTelur();
        WaktuDeteksiTelur.reset();
    }
    if (WaktuLampu.isReady()) {
        Lampu();
        WaktuLampu.reset();
    }
    if (WaktuKirimDataSql.isReady()) {
        KirimDataSql();
        WaktuKirimDataSql.reset();
    }
}
}
}
}
}
}
}

```

### 5.1.5 Implementasi Metode *Fuzzy* Sugeno

Proses perhitungan menggunakan metode *fuzzy* sugeno dijalankan di NodeMCU yang hasilnya akan digunakan sebagai kontrol aktuator, berikut merupakan penguraian alur perhitungan *fuzzy* sugeno dengan asumsi data yang didapat oleh sensor dht11 yaitu suhu 34°C dan kelembapan 56%.

#### 1. Proses Fuzzifikasi

Proses fuzzifikasi digunakan untuk mengubah inputan sensor ke dalam himpunan *fuzzy*. Berikut adalah pengelompokan himpunan keanggotaan *fuzzy* dari variable suhu dan kelembaban ruang inkubator.

##### a. Himpunan Keanggotaan Suhu

Data himpunan keanggotaan suhu merupakan kombinasi data yang didapat dari jurnal referensi dan wawancara yang ditunjukkan pada Tabel 5.7 dan menghasilkan range suhu yang memiliki tiga himpunan ditunjukkan pada Tabel 5.8.

Tabel 5. 7 Data Suhu

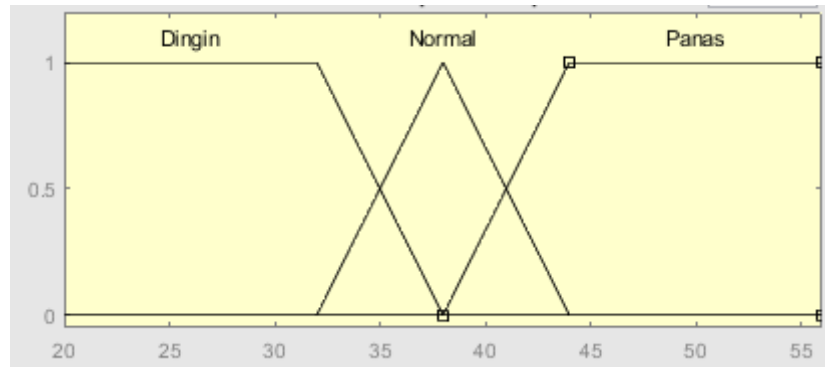
Jurnal Referensi		Wawancara	
Suhu	36 °C - 39°C	Suhu	37 °C - 38°C

Tabel 5. 8 Himpunan Keanggotaan Suhu

Himpunan Keanggotaan Fuzzy	Range Nilai
Dingin	20 °C – 38 °C

Normal	32 °C – 44 °C
Panas	38 °C – 56 °C

Representasi dari keanggotaan *fuzzy* dengan variable suhu dapat dilihat pada Gambar 5.16.



Gambar 5. 16 Representasi Himpunan Keanggotaan Suhu

Dengan asumsi bahwa nilai suhu udara adalah 34°C maka termasuk kedalam normal, sehingga perhitungannya sebagai berikut :

$$\begin{aligned} \bullet \quad \mu_{\text{Dingin}} &= \begin{cases} 1; & x < 32 \\ \frac{(38-x)}{(38-32)}; & x \geq 32 \text{ dan } x \leq 38 \\ 0; & x > 38 \end{cases} \\ \bullet \quad \mu_{\text{Normal}} &= \begin{cases} 0; & x < 32 \text{ atau } x > 44 \\ \frac{(x-38)}{(38-32)}; & x \geq 32 \text{ dan } x \leq 38 \\ \frac{(44-x)}{(44-38)}; & x > 38 \text{ dan } x \leq 44 \end{cases} \\ \bullet \quad \mu_{\text{Panas}} &= \begin{cases} 0; & x < 38 \\ \frac{(x-38)}{(44-38)}; & x \geq 38 \text{ dan } x \leq 44 \\ 1; & x > 44 \end{cases} \end{aligned}$$

Maka diperoleh hasil fuzzifikasi suhu udara sebagai berikut :

- $\mu_{\text{Dingin}} \quad [34] = 0.67$
- $\mu_{\text{Normal}} \quad [34] = 0.33$
- $\mu_{\text{Panas}} \quad [34] = 0$

b. Himpunan Keanggotaan Kelembaban

Data himpunan keanggotaan kelembaban merupakan kombinasi data yang didapat dari jurnal referensi dan wawancara yang ditunjukkan pada Tabel 5.9 dan menghasilkan range kelembaban yang memiliki tiga himpunan ditunjukkan pada Tabel 5.10.

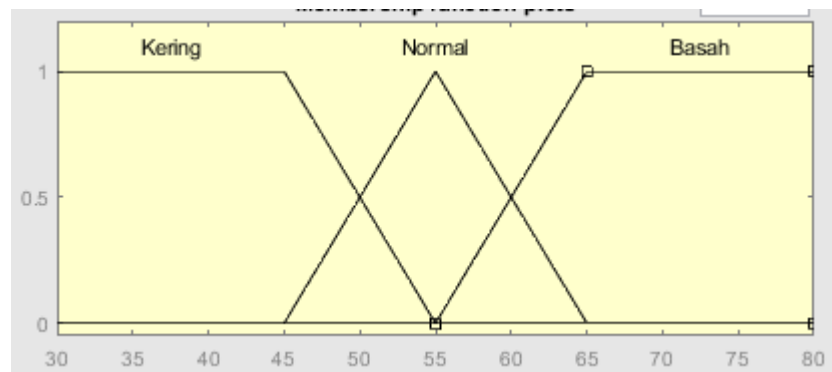
Tabel 5. 9 Data Kelembaban

Jurnal Referensi		Wawancara	
Kelembaban	55%	Kelembaban	70%

Tabel 5. 10 Himpunan Keanggotaan Kelembaban

Himpunan Keanggotaan Fuzzy	Range Nilai
Kering	30% – 55%
Normal	45% – 65%
Basah	55% – 80%

Representasi dari keanggotaan *fuzzy* dengan variable kelembaban dapat dilihat pada Gambar 5.17.



Gambar 5. 17 Representasi Himpunan Keanggotaan Kelembaban

Dengan asumsi bahwa nilai suhu udara adalah 56% maka termasuk kedalam normal, sehingga perhitungannya sebagai berikut :

$$\bullet \quad \mu_{\text{Kering}} = \begin{cases} 1; & x < 45 \\ \frac{(55-x)}{(55-45)}; & x \geq 45 \text{ dan } x \leq 55 \\ 0; & x > 55 \end{cases}$$

$$\begin{aligned} \bullet \quad \mu_{\text{Normal}} &= \begin{cases} 0; & x < 45 \text{ atau } x > 65 \\ \frac{(x-45)}{(55-45)}; & x \geq 45 \text{ dan } x \leq 55 \\ \frac{(65-x)}{(65-55)}; & x > 55 \text{ dan } x \leq 65 \end{cases} \\ \bullet \quad \mu_{\text{Basah}} &= \begin{cases} 0; & x < 55 \\ \frac{(x-55)}{(65-55)}; & x \geq 55 \text{ dan } x \leq 65 \\ 1; & x > 65 \end{cases} \end{aligned}$$

Maka diperoleh hasil fuzzifikasi suhu udara sebagai berikut :

- $\mu_{\text{Kering}} \quad [56] = 0$
- $\mu_{\text{Normal}} \quad [56] = 0.90$
- $\mu_{\text{Basah}} \quad [56] = 0.10$

## 2. Proses Fungsi Implikasi Inferensi

Berdasarkan proses fuzzifikasi, maka diperoleh basis aturan *fuzzy* yang akan ditunjukkan pada Tabel 5.11.

Tabel 5. 11 Basis Aturan *Fuzzy*

Rule	Suhu	Kelembapan	Lampu	Kipas
R1	Dingin	Kering	Terang	Pelan
R2	Normal	Normal	Normal	Normal
R3	Panas	Basah	Redup	Cepat
R4	Dingin	Normal	Terang	Normal
R5	Normal	Kering	Normal	Pelan
R6	Panas	Kering	Redup	Pelan
R7	Dingin	Basah	Terang	Cepat
R8	Normal	Basah	Normal	Cepat
R9	Panas	Normal	Redup	Normal

Pada metode *fuzzy sugeno*, nilai *output* berupa nilai konstan sehingga telah ditentukan nilai kondisi dari *output* seperti pada Tabel 5.12.

Tabel 5. 12 Nilai Konstan Kondisi *Output*

Variabel Kondisi	Nilai
Terang	85
Normal	45
Redup	30
Cepat	300
Normal	260
Pelan	200

Fungsi Implikasi menggunakan perhitungan minimum (MIN), dengan mengambil nilai terkecil dari hasil fuzzifikasi suhu dan kelembapan.

$$a_i = \mu_{A_1}(X) \cap \mu_{B_1}(X) \cap \mu_{C_1}(X) = \text{MIN}\{\mu_{A_1}(X), \mu_{B_1}(X), \mu_{C_1}(X)\}$$

Dalam penentuan status *output* menggunakan Metode *fuzzy sugeno* Order Nol yaitu nilai K sebagai konstanta atau nilai tegas.

$$\text{IF } (x_1 \text{ is } A_1) \cdot (x_2 \text{ is } A_2) \cdot \dots \dots \dots \cdot (x_n \text{ is } A_n) \text{ THEN } z = k$$

Keterangan :

- $x_n$  = *variable input*
- $A_n$  = himpunan keanggotaan

Berikut adalah basis aturan untuk penentuan *output* dari intensitas cahaya lampu dan kecepatan kipas.

- [R1] IF Suhu Dingin AND Kelembapan Kering THEN Lampu Terang AND Kipas Pelan
- $$\alpha - \text{predikat1} = \mu_{\text{Suhu Dingin}} \cap \mu_{\text{Kelembapan Kering}}$$
- $$= \text{MIN} (0 ; 0)$$
- $$= 0$$



- [R2] IF Suhu Normal AND Kelembapan Normal THEN Lampu Normal AND Kipas Normal

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Normal}} \cap \mu_{\text{Kelembapan Normal}} \\ &= \text{MIN} (0.33 ; 0.90) \\ &= 0.33\end{aligned}$$

- [R3] IF Suhu Panas AND Kelembapan Basah THEN Lampu Redup AND Kipas Cepat

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Panas}} \cap \mu_{\text{Kelembapan Basah}} \\ &= \text{MIN} (0 ; 0) \\ &= 0\end{aligned}$$

- [R4] IF Suhu Dingin AND Kelembapan Normal THEN Lampu Terang AND Kipas Normal

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Dingin}} \cap \mu_{\text{Kelembapan Normal}} \\ &= \text{MIN} (0.67 , 0.90) \\ &= 0.67\end{aligned}$$

- [R5] IF Suhu Normal AND Kelembapan Kering THEN Lampu Normal AND Kipas Pelan

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Normal}} \cap \mu_{\text{Kelembapan Kering}} \\ &= \text{MIN} (0 ; 0) \\ &= 0\end{aligned}$$

- [R6] IF Suhu Panas AND Kelembapan Kering THEN Lampu Redup AND Kipas Pelan

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Panas}} \cap \mu_{\text{Kelembapan Kering}} \\ &= \text{MIN} (0 ; 0) \\ &= 0\end{aligned}$$

- [R7] IF Suhu Dingin AND Kelembapan Basah THEN Lampu Terang AND Kipas Cepat

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Dingin}} \cap \mu_{\text{Kelembapan Basah}} \\ &= \text{MIN} (0.67 ; 0.10) \\ &= 0.10\end{aligned}$$

- [R8] IF Suhu Normal AND Kelembapan Basah THEN Lampu Normal AND Kipas Cepat

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Dingin}} \cap \mu_{\text{Kelembapan Kering}} \\ &= \text{MIN} (0.33 ; 0.10) \\ &= 0.10\end{aligned}$$

- [R9] IF Suhu Panas AND Kelembapan Normal THEN Lampu Redup AND Kipas Normal

$$\begin{aligned}\alpha - \text{predikat1} &= \mu_{\text{Suhu Panas}} \cap \mu_{\text{Kelembapan Normal}} \\ &= \text{MIN} (0 ; 0) \\ &= 0\end{aligned}$$

### 3. Defuzzifikasi

Proses defuzzifikasi bertujuan untuk menghasilkan konstanta atau nilai tegas menggunakan rumus rata-rata terbobot (weight average) sebagai berikut :

$$WA = \frac{a_1 z_1 + a_2 z_2 + a_3 z_3 + \dots a_i z_i}{a_1 + a_2 + a_3 + \dots a_i}$$

Keterangan :

$a_i$  : adalah  $\alpha$  – predikat ke – i.

$z_i$  : adalah konsekuensi ke – i.

Sehingga hasil rata – ratanya adalah :

$$\begin{aligned}WA_{\text{Lampu}} &= \frac{(0,33 \times 45) + (0,67 \times 85) + (0,10 \times 85) + (0,10 \times 45)}{(0,33 + 0,67 + 0,10 + 0,10)}\end{aligned}$$

$$= 70,6$$

$$WA_{Kipas}$$

$$= \frac{(0,33 \times 260) + (0,67 \times 260) + (0,10 \times 300) + (0,10 \times 300)}{(0,33 + 0,67 + 0,10 + 0,10)}$$

$$= 266,6$$

Berdasarkan perhitungan manual menggunakan Metode *fuzzy* sugeno Orde Nol dengan inputan Suhu Udara 34°C dan Kelembaban 56%, menghasilkan nilai *output* untuk lampu 70,6 dan kipas 266,6. Nilai *output* tersebut yang akan digunakan untuk mengatur intensitas cahaya lampu dan kecepatan.

## 5.2 Pengujian

Tahap pengujian berisikan tentang skenario tentang pengujian terhadap sistem untuk mengetahui apakah sistem sudah berjalan sesuai dengan yang diharapkan atau tidak. Pengujian pada bab ini meliputi pengujian *hardware*, pengujian *software*, pengujian metode, pengujian sistem keseluruhan.

### 5.2.1 Pengujian *Hardware*

#### 5.2.1.1 Pengujian Sensor DHT11 dan PIR

Pengujian sensor suhu, kelembaban, pancaran infra merah dan gerakan dilakukan untuk menguji fungsionalitas apakah bekerja dengan baik untuk mendapatkan kondisi prototype. Pengujian ditunjukkan pada Tabel 5.13.

Tabel 5. 13 Pengujian Sensor DHT11 dan PIR

No.	Skenario	Harapan Hasil	Keterangan
1.	Pembacaan data sensor suhu, kelembaban udara, pancaran infra merah, dan gerakan	Sensor mampu membaca, memberikan dan menampilkan data dari sensor ke dalam serial monitor	[✓] Sesuai
			[ ] Tidak Sesuai

### 5.2.1.2 Pengujian Modul Relay

Pengujian modul relay dilakukan untuk menguji fungsionalitas apakah modul relay dapat bekerja dengan baik pada pengaturan on maupun off. Pengujian ditunjukkan pada Tabel 5.14.

Tabel 5. 14 Pengujian Modul Relay

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian untuk pengaturan logika nyala dan mati pada <i>dynamo synchronus</i>	Modul dapat menyambungkan dan memutus tegangan untuk <i>dynamo synchronus</i>	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai

### 5.2.1.3 Pengujian Modul AC *Light Dimmer*

Pengujian modul AC *light dimmer* dilakukan untuk menguji fungsionalitas apakah bekerja dengan baik untuk pengaturan kecerahan lampu yang dihasilkan. Pengujian ditunjukkan pada Tabel 5.15.

Tabel 5. 15 Pengujian Modul AC *Light Dimmer*

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian untuk pengaturan intensitas cahaya yang dihasilkan lampu pijar	Modul dapat mengatur intensitas cahaya lampu pijar sesuai dengan nilai yang diberikan	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai

### 5.2.1.4 Pengujian Modul L298N

Pengujian modul l298n dilakukan untuk menguji fungsionalitas apakah bekerja dengan baik untuk pengaturan kecepatan yang dihasilkan oleh kipas. Pengujian ditunjukkan pada Tabel 5.16.

Tabel 5. 16 Pengujian Modul L298N

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian untuk pengaturan kecepatan yang dihasilkan oleh kipas	Modul dapat mengatur kecepatan kipas sesuai dengan nilai yang diberikan	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai

#### 5.2.1.5 Pengujian Lampu Pijar

Pengujian lampu dilakukan untuk menguji fungsionalitas apakah bekerja dengan baik sebagai sumber panas dari ruang inkubator prototype. Pengujian ditunjukkan pada Tabel 5.17.

Tabel 5. 17 Pengujian Lampu Pijar

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian kestabilan intensitas cahaya yang dihasilkan lampu pijar	Cahaya yang dihasilkan dapat stabil dan tidak terdapat kendala saat melakukan pengaturan intensitas cahaya	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai

#### 5.2.1.6 Pengujian Kipas

Pengujian kipas dilakukan untuk menguji fungsionalitas apakah bekerja dengan baik sebagai kontrol kelembapan dan perataan panas pada ruang inkubator. Pengujian ditunjukkan pada Tabel 5.18.

Tabel 5. 18 Pengujian Kipas

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian kipas sebagai kontrol kelembapan dan perataan panas pada prototipe inkubator	Kecepatan kipas dapat diatur sesuai dengan kondisi pada ruang inkubator	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai

### 5.2.1.7 Pengujian *Dynamo Synchronus*

Pengujian *dynamo synchronus* dilakukan untuk menguji fungsionalitas apakah bekerja dengan baik sebagai penggerak untuk memutar telur. Pengujian ditunjukkan pada Tabel 5.19.

Tabel 5. 19 Pengujian *Dynamo Synchronus*

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian <i>dynamo synchronus</i> sebagai penggerak untuk menarik rak telur pada prototipe inkubator	<i>Dynamo synchronus</i> dapat berputar dan dapat menarik rak telur agar telur dapat berputar	[✓] Sesuai
			[ ] Tidak Sesuai

### 5.2.2 Pengujian *Software*

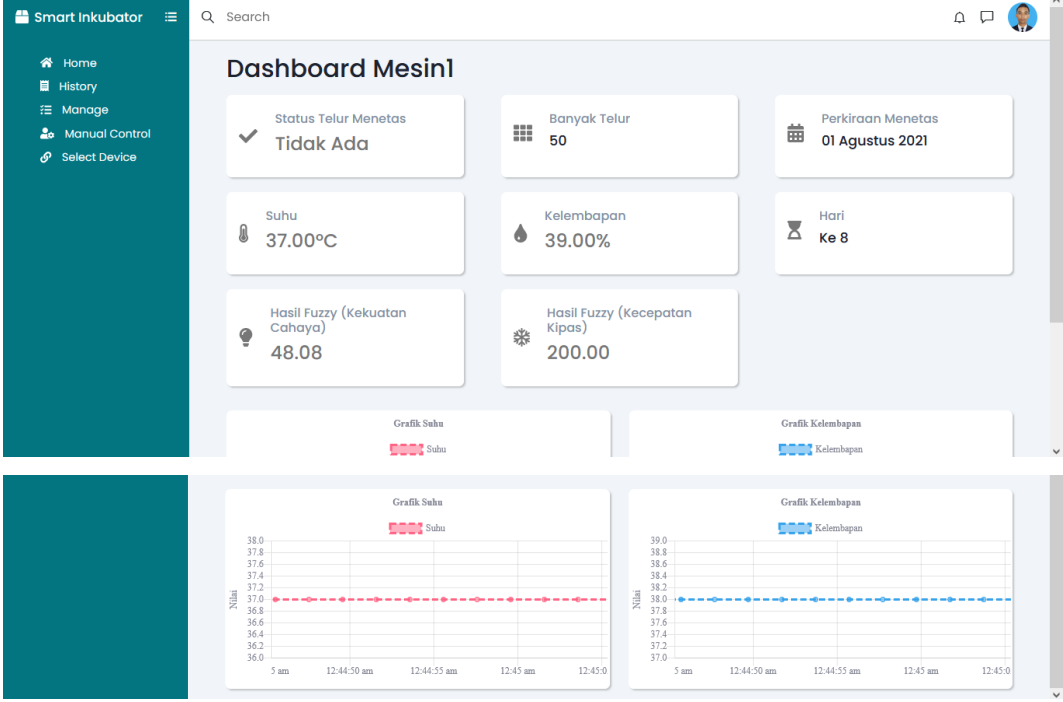
#### 5.2.2.1 Pengujian *Monitoring Website*

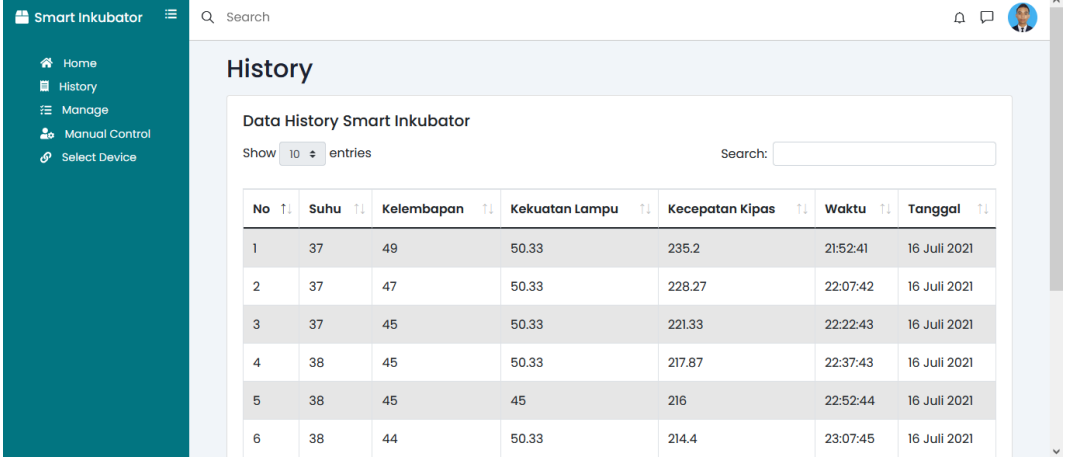
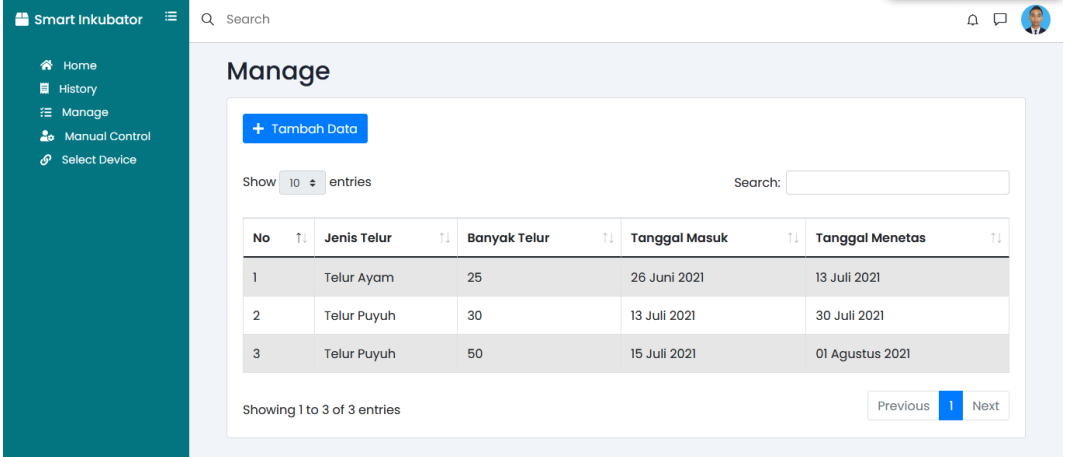
Pengujian *monitoring website* dilakukan untuk menguji fungsionalitas apakah sistem *monitoring* pada *website* dapat berjalan sesuai dengan kebutuhan. Pengujian ditunjukkan pada Tabel 5.20.

Tabel 5. 20 Pengujian *Monitoring Website*

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian pada halaman <i>dashboard website</i>	Halaman <i>dashboard website</i> menampilkan data status telur menetas, banyak telur, tanggal perkiraan menetas, suhu, kelembaban, hari ke -i, hasil <i>fuzzy</i> lampu, dan hasil <i>fuzzy</i> kipas serta <i>realtime chart</i> dari sensor <i>input</i> .	[✓] Sesuai
			[ ] Tidak Sesuai

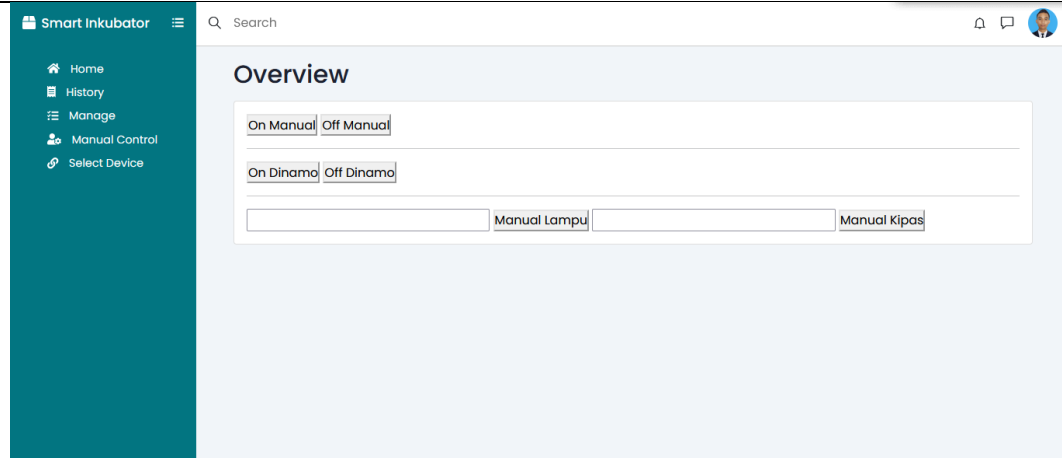


No.	Skenario	Harapan Hasil	Keterangan
<b>Gambar Halaman Dashboard</b>			
			
2.	Pengujian halaman histori	Halaman <i>history website</i> menampilkan data yang tersimpan dari database serta menggunakan fungsi pencarian dan filter jumlah data yang tampil	<input checked="" type="checkbox"/> Sesuai  <input type="checkbox"/> Tidak Sesuai
<b>Gambar Halaman Histori</b>			

No.	Skenario	Harapan Hasil	Keterangan																																																	
	 <p><b>Smart Inkubator</b> Search</p> <p>History</p> <p>Data History Smart Inkubator</p> <p>Show 10 entries Search:</p> <table border="1"> <thead> <tr> <th>No</th> <th>Suhu</th> <th>Kelembapan</th> <th>Kekuatan Lampu</th> <th>Kecepatan Kipas</th> <th>Waktu</th> <th>Tanggal</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>37</td> <td>49</td> <td>50.33</td> <td>235.2</td> <td>21:52:41</td> <td>16 Juli 2021</td> </tr> <tr> <td>2</td> <td>37</td> <td>47</td> <td>50.33</td> <td>228.27</td> <td>22:07:42</td> <td>16 Juli 2021</td> </tr> <tr> <td>3</td> <td>37</td> <td>45</td> <td>50.33</td> <td>221.33</td> <td>22:22:43</td> <td>16 Juli 2021</td> </tr> <tr> <td>4</td> <td>38</td> <td>45</td> <td>50.33</td> <td>217.87</td> <td>22:37:43</td> <td>16 Juli 2021</td> </tr> <tr> <td>5</td> <td>38</td> <td>45</td> <td>45</td> <td>216</td> <td>22:52:44</td> <td>16 Juli 2021</td> </tr> <tr> <td>6</td> <td>38</td> <td>44</td> <td>50.33</td> <td>214.4</td> <td>23:07:45</td> <td>16 Juli 2021</td> </tr> </tbody> </table>			No	Suhu	Kelembapan	Kekuatan Lampu	Kecepatan Kipas	Waktu	Tanggal	1	37	49	50.33	235.2	21:52:41	16 Juli 2021	2	37	47	50.33	228.27	22:07:42	16 Juli 2021	3	37	45	50.33	221.33	22:22:43	16 Juli 2021	4	38	45	50.33	217.87	22:37:43	16 Juli 2021	5	38	45	45	216	22:52:44	16 Juli 2021	6	38	44	50.33	214.4	23:07:45	16 Juli 2021
No	Suhu	Kelembapan	Kekuatan Lampu	Kecepatan Kipas	Waktu	Tanggal																																														
1	37	49	50.33	235.2	21:52:41	16 Juli 2021																																														
2	37	47	50.33	228.27	22:07:42	16 Juli 2021																																														
3	37	45	50.33	221.33	22:22:43	16 Juli 2021																																														
4	38	45	50.33	217.87	22:37:43	16 Juli 2021																																														
5	38	45	45	216	22:52:44	16 Juli 2021																																														
6	38	44	50.33	214.4	23:07:45	16 Juli 2021																																														
3.	Pengujian Halaman Manage	Halaman manage dapat menambahkan data manage dan menampilkan data telur yang masuk dalam inkubator	<input checked="" type="checkbox"/> Sesuai <input type="checkbox"/> Tidak Sesuai																																																	
<b>Gambar Halaman Manage</b>																																																				
	 <p><b>Smart Inkubator</b> Search</p> <p>Manage</p> <p>+ Tambah Data</p> <p>Show 10 entries Search:</p> <table border="1"> <thead> <tr> <th>No</th> <th>Jenis Telur</th> <th>Banyak Telur</th> <th>Tanggal Masuk</th> <th>Tanggal Menetas</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Telur Ayam</td> <td>25</td> <td>26 Juni 2021</td> <td>13 Juli 2021</td> </tr> <tr> <td>2</td> <td>Telur Puyuh</td> <td>30</td> <td>13 Juli 2021</td> <td>30 Juli 2021</td> </tr> <tr> <td>3</td> <td>Telur Puyuh</td> <td>50</td> <td>15 Juli 2021</td> <td>01 Agustus 2021</td> </tr> </tbody> </table> <p>Showing 1 to 3 of 3 entries Previous 1 Next</p>			No	Jenis Telur	Banyak Telur	Tanggal Masuk	Tanggal Menetas	1	Telur Ayam	25	26 Juni 2021	13 Juli 2021	2	Telur Puyuh	30	13 Juli 2021	30 Juli 2021	3	Telur Puyuh	50	15 Juli 2021	01 Agustus 2021																													
No	Jenis Telur	Banyak Telur	Tanggal Masuk	Tanggal Menetas																																																
1	Telur Ayam	25	26 Juni 2021	13 Juli 2021																																																
2	Telur Puyuh	30	13 Juli 2021	30 Juli 2021																																																
3	Telur Puyuh	50	15 Juli 2021	01 Agustus 2021																																																
4.	Halaman Manual Kontrol	Halaman manual kontrol <i>website</i> dapat melakukan kontrol alat secara manual melalui <i>website</i> . Kontrol manual meliputi on / off	<input checked="" type="checkbox"/> Sesuai <input type="checkbox"/> Tidak Sesuai																																																	

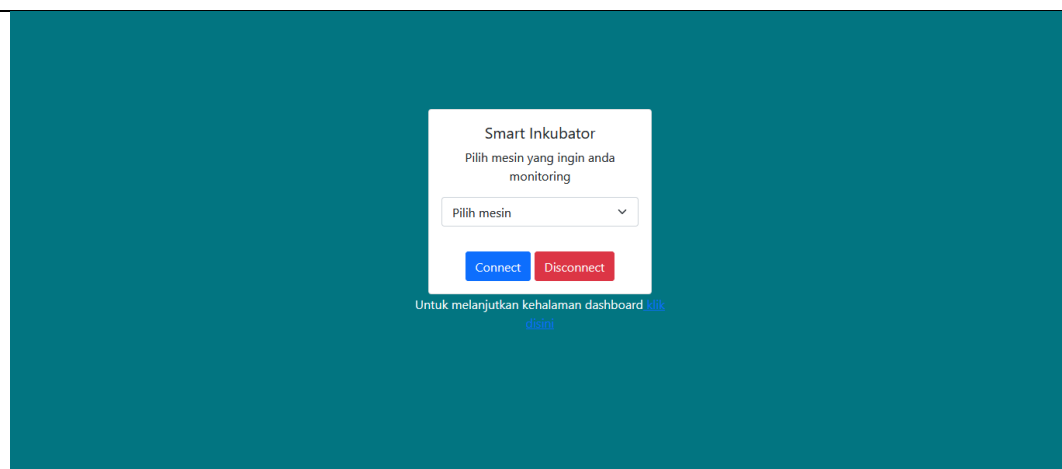
No.	Skenario	Harapan Hasil	Keterangan
		<i>dynamo</i> , mengatur intensitas cahaya lampu, dan kecepatan kipas	

### Gambar Halaman Manual Kontrol



5.	Halaman <i>Select Device</i>	Halaman <i>select device</i> dapat memilih <i>device</i> mana yang ingin dimonitoring dan akan ditampilkan pada <i>dashboard</i>	[✓] Sesuai
			[ ] Tidak Sesuai

### Gambar Halaman *Select Device*

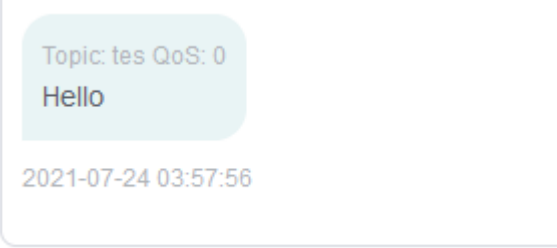


### 5.2.2.2 Pengujian Protokol pada Sistem Aplikasi

Pengujian protokol dilakukan untuk menguji fungsionalitas apakah *website* dapat terkoneksi dengan broker dan publish serta subscribe data. Pengujian ditunjukkan pada Tabel 5.21.

Tabel 5. 21 Pengujian Protokol pada Sistem Aplikasi

No.	Skenario	Harapan Hasil	Keterangan																															
1.	Pengujian protokol mqtt pada sistem dapat terkoneksi ke broker	NodeMCU maupun sistem <i>website</i> dapat terkoneksi ke mqtt broker	[✓] Sesuai																															
			[ ] Tidak Sesuai																															
<b>Gambar Pengujian Protokol MQTT</b>																																		
<table border="1"> <thead> <tr> <th>Client ID</th> <th>Username</th> <th>IP Address</th> <th>Keepalive</th> <th>Protocol</th> <th>Connect Status</th> <th>Connection At</th> </tr> </thead> <tbody> <tr> <td>mqttProses</td> <td>admin</td> <td>125.166.0.179:2719 8</td> <td>60</td> <td>MQTT</td> <td>Connected</td> <td>2021-07-24 03:26 0</td> </tr> <tr> <td>NodeMCU2</td> <td>admin</td> <td>125.166.0.179:1966 4</td> <td>15</td> <td>MQTT</td> <td>Connected</td> <td>2021-07-23 22:48 2</td> </tr> <tr> <td>NodeMCU1</td> <td>admin</td> <td>125.166.0.179:2704 0</td> <td>15</td> <td>MQTT</td> <td>Connected</td> <td>2021-07-23 20:36 6</td> </tr> </tbody> </table>							Client ID	Username	IP Address	Keepalive	Protocol	Connect Status	Connection At	mqttProses	admin	125.166.0.179:2719 8	60	MQTT	Connected	2021-07-24 03:26 0	NodeMCU2	admin	125.166.0.179:1966 4	15	MQTT	Connected	2021-07-23 22:48 2	NodeMCU1	admin	125.166.0.179:2704 0	15	MQTT	Connected	2021-07-23 20:36 6
Client ID	Username	IP Address	Keepalive	Protocol	Connect Status	Connection At																												
mqttProses	admin	125.166.0.179:2719 8	60	MQTT	Connected	2021-07-24 03:26 0																												
NodeMCU2	admin	125.166.0.179:1966 4	15	MQTT	Connected	2021-07-23 22:48 2																												
NodeMCU1	admin	125.166.0.179:2704 0	15	MQTT	Connected	2021-07-23 20:36 6																												
2.	Pengujian sistem untuk melakukan publish data	NodeMCU maupun <i>Website</i> dapat melakukan publish data melalui mqtt broker dengan topic yang sama	[✓] Sesuai																															
			[ ] Tidak Sesuai																															
<b>Gambar Pengujian Sistem Publish Data</b>																																		
<p><b>Publisher</b></p> <p>Topic: tes      QoS: 0      Select Connection: tes publish      <input type="checkbox"/> Retain      <b>Publish</b></p> <p>Payload: JSON</p> <p>1 Hello</p>																																		

3.	Pengujian sistem untuk melakukan subscribe data	NodeMCU maupun <i>Website</i> dapat melakukan subscribe data melalui mqtt broker dengan topic yang sama	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai
<b>Gambar Pengujian Sistem Subscribe Data</b>			
			

### 5.2.2.3 Pengujian Database

Pengujian penyimpanan database dilakukan untuk menguji fungsionalitas apakah sistem dapat menyimpan data yang meliputi data sensor, hasil *fuzzy*, device, dan manage data kedalam database. Pengujian ditunjukkan pada Tabel 5.22.

Tabel 5. 22 Pengujian Database

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian penyimpanan data sensor dan hasil <i>fuzzy</i> pada database	Data dapat tersimpan ke dalam database	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai
<b>Gambar Pengujian Simpan Data Sensor dan Hasil Fuzzy</b>			

id	suhu	kelembapan	fuzzy_lampu	fuzzy_kipas	waktu	tanggal
201	37	49	50.33	235.2	21:52:41	2021-07-16
202	37	47	50.33	228.27	22:07:42	2021-07-16
203	37	45	50.33	221.33	22:22:43	2021-07-16
204	38	45	50.33	217.87	22:37:43	2021-07-16
205	38	45	45	216	22:52:44	2021-07-16
206	38	44	50.33	214.4	23:07:45	2021-07-16
207	38	44	50.33	214.4	23:22:45	2021-07-16
208	38	43	50.33	210.93	23:37:46	2021-07-16
209	38	43	45	212	23:52:46	2021-07-16
210	37	42	45	212	00:07:48	2021-07-17
211	38	42	45	208	00:22:48	2021-07-17

2.	Pengujian penyimpanan data detail mikrokontroller pada database	Data dapat tersimpan ke dalam database	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai

**Gambar Pengujian Simpan Data Detail NodeMCU**

id	mac_address	nama_device	topicCol1
2	48:3F:DA:0D:85:98	Mesin1	Mesin1

3.	Pengujian penyimpanan data manajemen pada database	Data dapat tersimpan ke dalam database	<input checked="" type="checkbox"/> Sesuai
			<input type="checkbox"/> Tidak Sesuai

**Gambar Pengujian Simpan Data Manajemen**

id	jenis_telur	banyak_telur	tanggal_masuk	tanggal_keluar
8	Telur Ayam	25	2021-06-26	2021-07-13
9	Telur Puyuh	30	2021-07-13	2021-07-30
10	Telur Puyuh	50	2021-07-15	2021-08-01

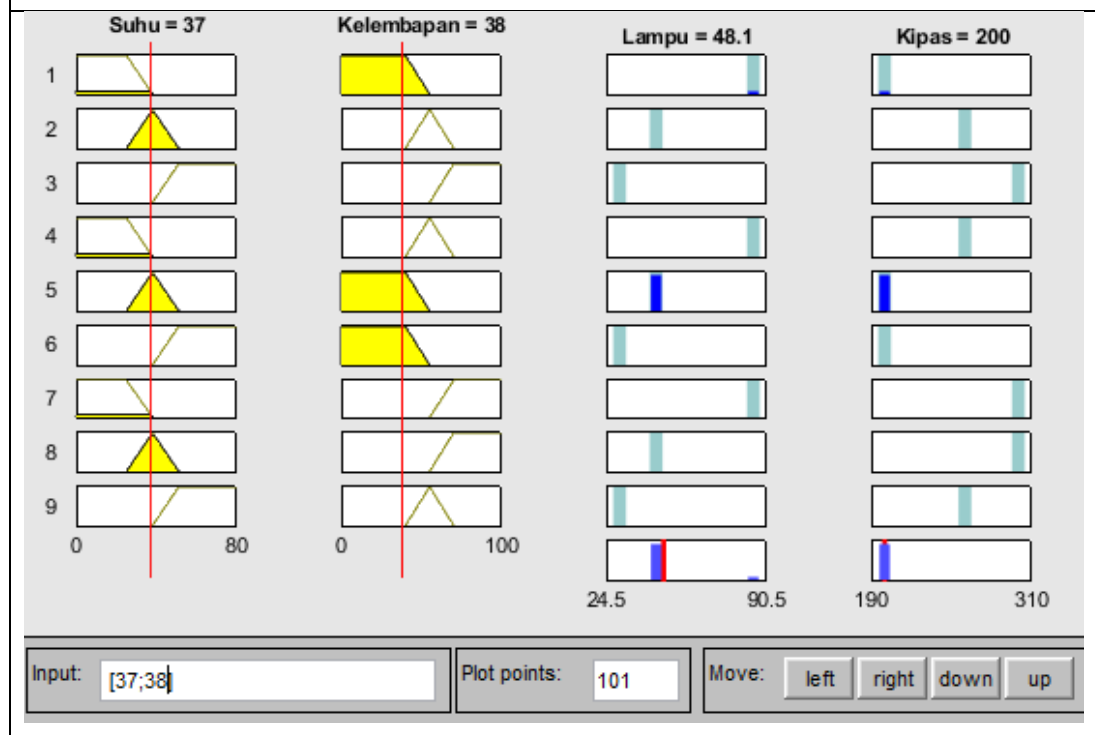
### 5.2.3 Pengujian Metode *Fuzzy* Sugeno

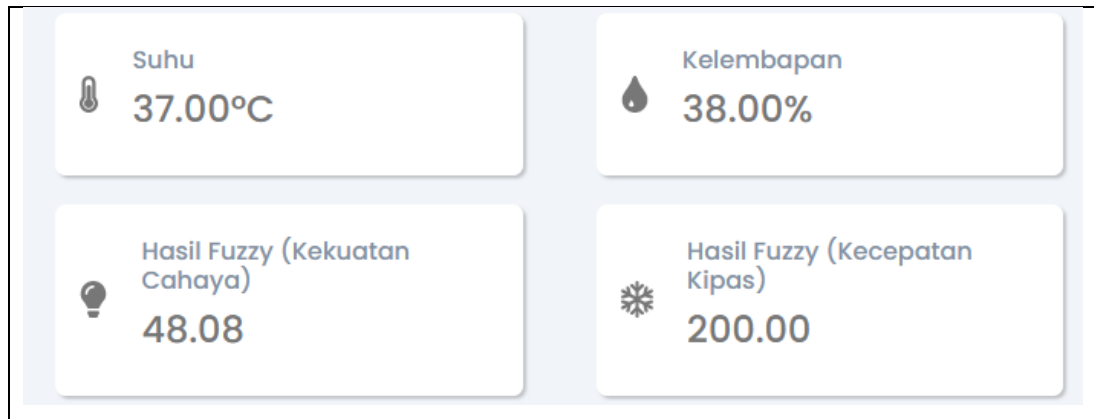
Pengujian metode *fuzzy* sugeno dilakukan sebagai perbandingan perhitungan antara sistem dengan *fuzzy* pada matlab sehingga perhitungan pada matlab dapat menjadi patokan dalam perhitungan pada program untuk keberhasilan pengimplementasian *fuzzy* sugeno sebagai metode yang digunakan pada alat. Pengujian ditunjukkan pada Tabel 5.23.

Tabel 5. 23 Pengujian Metode *Fuzzy* Sugeno

No.	Skenario	Harapan Hasil	Keterangan
1.	Pengujian metode <i>fuzzy</i> sugeno pada program	Perhitungan <i>fuzzy</i> sugeno pada program mendapatkan nilai yang sama dengan perhitungan pada matlab	[✓] Sesuai
			[ ] Tidak Sesuai

Gambar Pengujian Metode Fuzzy Sugeno





#### 5.2.4 Pengujian Sistem Keseluruhan

Pengujian sistem dilakukan untuk mengetahui apakah sistem berjalan sesuai dengan rencana yang diharapkan atau tidak. Pengujian dilakukan dengan cara metode *Black Box* yaitu pengujian berdasarkan aplikasi tanpa melihat struktur atau cara kerja internal. Pengujian yang dilakukan dapat dilihat pada Tabel 5.24.

Tabel 5. 24 Pengujian Sistem Keseluruhan

No	Skenario	Keterangan
1	Pembacaan sensor terhadap suhu, kelembaban, pancaran infra merah, dan gerakan pada ruang inkubator	<input checked="" type="checkbox"/> Sesuai
		<input type="checkbox"/> Tidak Sesuai
2	Menampilkan data suhu, kelembaban, pancaran infra merah dan gerakan pada <i>dashboard</i> secara <i>realtime</i>	<input checked="" type="checkbox"/> Sesuai
		<input type="checkbox"/> Tidak Sesuai
3	Menampilkan data suhu, dan kelembaban ruang inkubator secara <i>realtime</i> pada chart	<input checked="" type="checkbox"/> Sesuai
		<input type="checkbox"/> Tidak Sesuai
4	Menampilkan hasil perhitungan <i>fuzzy</i> pada <i>dashboard</i> secara <i>realtime</i>	<input checked="" type="checkbox"/> Sesuai
		<input type="checkbox"/> Tidak Sesuai
5	Melihat data record suhu, kelembaban, hasil <i>fuzzy</i> lampu, dan <i>fuzzy</i> kipas pada halaman <i>history</i>	<input checked="" type="checkbox"/> Sesuai
		<input type="checkbox"/> Tidak Sesuai
6.	Melihat dan menambahkan data pada halaman <i>menage</i>	<input checked="" type="checkbox"/> Sesuai
		<input type="checkbox"/> Tidak Sesuai
7.		<input checked="" type="checkbox"/> Sesuai



No	Skenario	Keterangan
	Mengatur intensitas cahaya dan kecepatan kipas sesuai <i>output</i> yang dilakukan <i>fuzzy</i>	[ ] Tidak Sesuai
8.	Menyalakan dan mematikan <i>dynamo synchronus</i> sesuai dengan jadwal	[✓] Sesuai
		[ ] Tidak Sesuai
9.	Menyalakan atau mematikan aktuator menggunakan kontrol manual	[✓] Sesuai
		[ ] Tidak Sesuai
10.	Mengatur kecerahan lampu menggunakan kontrol manual	[✓] Sesuai
		[ ] Tidak Sesuai
11.	Mengatur kecepatan kipas menggunakan kontrol manual	[✓] Sesuai
		[ ] Tidak Sesuai
12.	Menyalakan dan mematikan relay menggunakan kontrol manual	[✓] Sesuai
		[ ] Tidak Sesuai

Pengujian keseluruhan dilakukan dengan mencoba satu persatu fitur dan fungsi dari sistem yang telah selesai dibuat.