

BAB III METODE PENELITIAN

3.1. Lokasi Pengujian

Penelitian dilakukan di Biro Sistem Informasi (BSI) Universitas Muhammadiyah Yogyakarta Jl. Brawijaya, Geblagan, Tamantirto, Kasihan, Bantul, Daerah Istimewa Yogyakarta 55183.



Gambar 3.1. Lokasi pengujian berdasarkan pemetaan

3.2 Peralatan dan Bahan Yang Digunakan

- **Perangkat Keras (Hardware)**

Adapun perangkat keras yang digunakan untuk menunjang penelitian sebagai berikut:

1. Satu buah Wemos D1 mini
2. Satu buah LCD Oled 128x64
3. Satu buah DHT22
4. Satu buah Power Supply 5V
5. Satu buah Thermometer (Extech)
6. Satu buah modem mifi
7. Satu unit Laptop

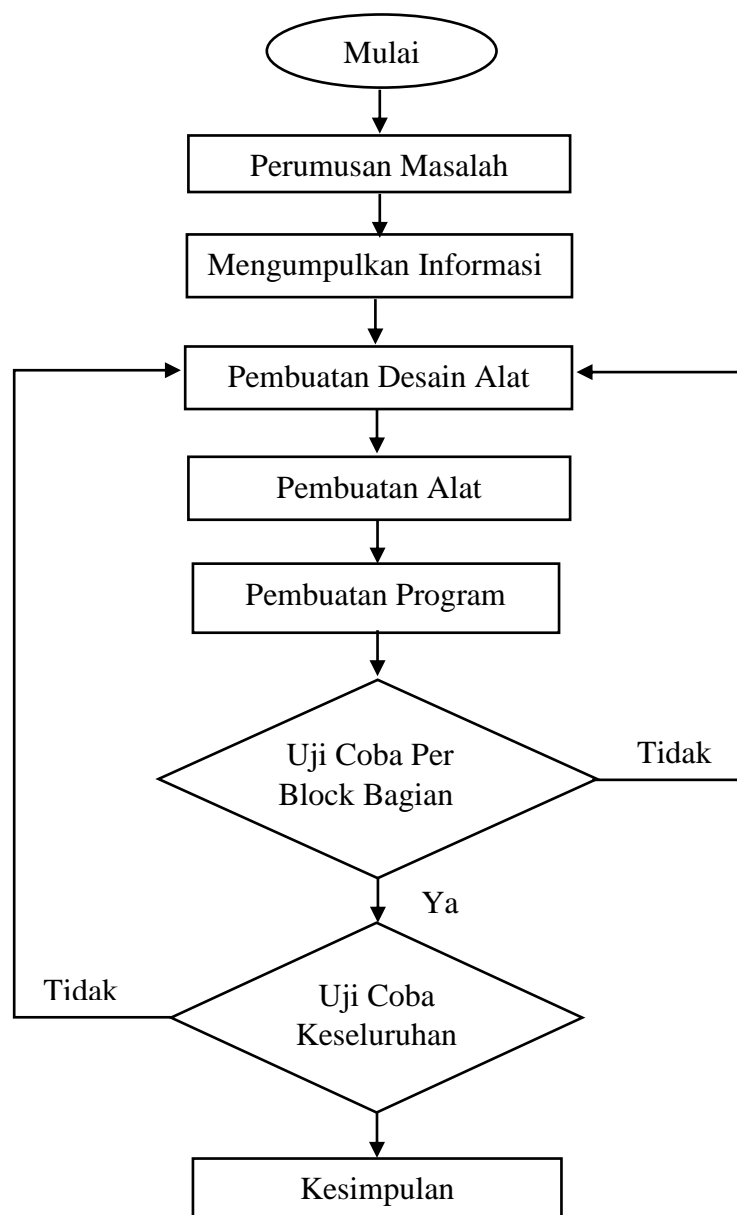
- **Perangkat Lunak (Software)**

Perangkat lunak yang digunakan untuk menunjang penelitian yaitu :

- a. Arduino IDE
- b. Microsoft Word 2013

3.3 Diagram Alur Jalannya Penelitian

Dalam penelitian ini terdapat beberapa proses atau tahapan yang akan di kerjakan, tahapan tersebut diperlihatkan pada *flowchart* berikut ini.



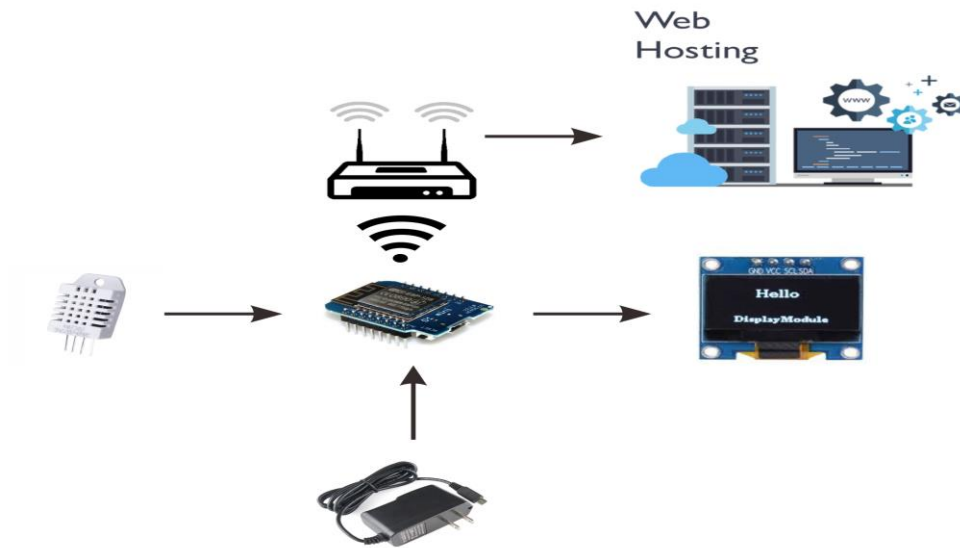
Gambar 3.2. *Flowchart* Jalanya Penelitian

Keterangan :

1. Perumusan Masalah: digunakan untuk menentukan judul dari beberapa faktor penggabungan sistem modifikasi dengan *existing* dan membatasi pokok permasalahan yang akan dibahas.
2. Pengumpulan Informasi: dilakukan dengan mengumpulkan data-data dan informasi dari buku maupun internet yang berkaitan dengan bahan komponen yang akan digunakan dalam alat ini.
3. Pembuatan Desain Alat: dilakukan dengan membuat rancangan atau desain alat yang bertujuan untuk langkah selanjutnya, yaitu pembuatan alat sehingga dapat diketahui bentuk gambar yang sesungguhnya dari alat ini.
4. Pembuatan Alat: setelah bahan-bahan dan komponen terkumpul maka dilakukan proses pembuatan alat sesuai dengan desain yang sudah dibuat.
5. Pembuatan Program: setelah pembuatan desain alat selanjutnya membuat program sehingga alat dapat bekerja sesuai dengan yang diinginkan.
6. Uji Coba Per Block Bagian: pengujian perblok bagian rangkaian yang terpisah sebelum.
7. Uji Coba Keseluruhan: pengujian dilakukan setelah semua bagian rangkaian telah digabungkan menjadi satu dan alat yang siap dicoba untuk pengambilan data.
8. Kesimpulan: berisi hasil akhir yang di peroleh setelah penelitian selesai.

3.4 Perancangan *Hardware*

Sistem yang di buat pada penelitian ini menggunakan beberapa komponen *hardware* atau perangkat keras yang saling berkaitan dan terhubung dalam sebuah sistem monitoring pendeteksi suhu dan kelembaban ruang server berbasis sensor suhu, gambar 3.3 memperhatikan diagram perancangan sistem yang dibuat pada penelitian ini.



Gambar 3.3 Diagram Perancangan *Hardware*

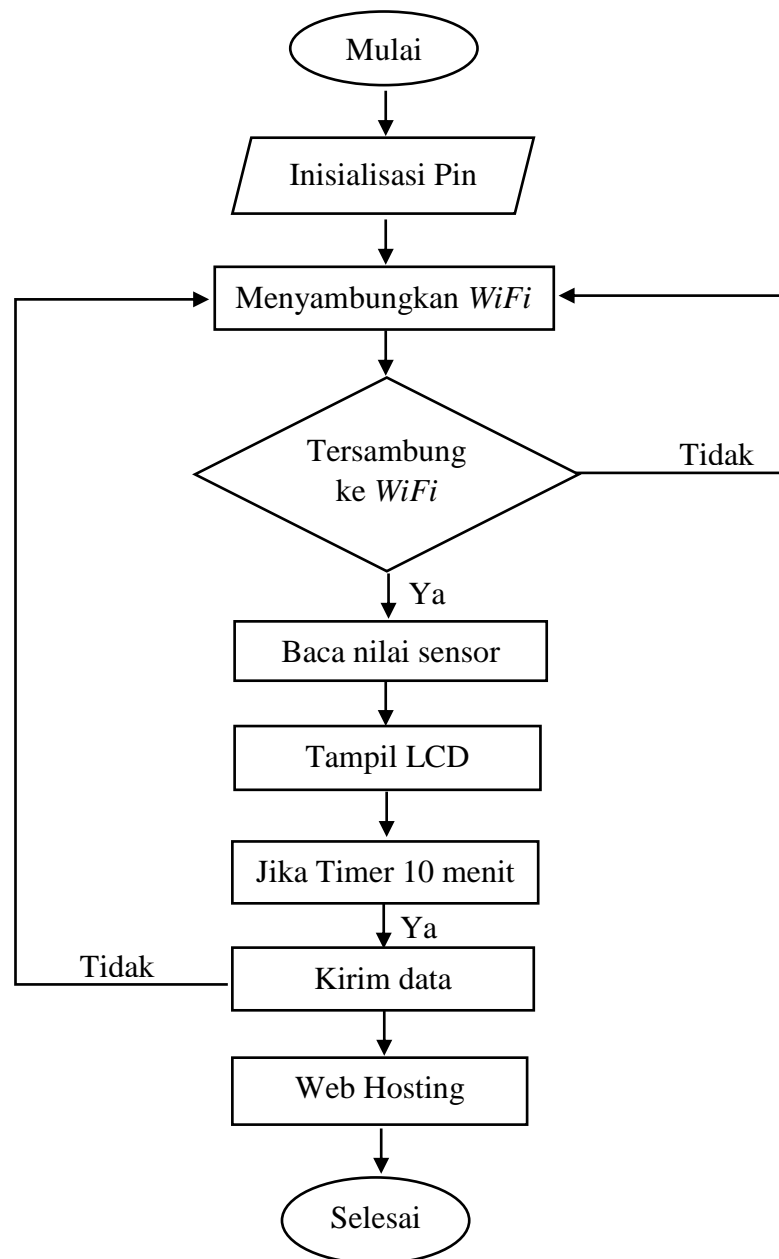
Komponen yang digunakan dalam perancangan *hardware* ini adalah wemos D1 mini yang merupakan mikrokontroler dengan dua buah input yaitu 5V dan 3,3V yang dihasilkan oleh catu daya, serta delapan input digital. Selanjutnya, ada satu buah sensor DHT22 yang diberi daya sebesar 3,3V berfungsi untuk mendeteksi suhu dan kelembaban. Sensor DHT22 dapat mengirimkan data ke mikrokontroler Wemos D1 mini melalui *shield wemos* DHT22 pin D4 (digital 4) dan data yang terbaca kemudian muncul pada layar LCD Oled. Penggunaan LCD oled karena hanya membutuhkan dua input digital maupun analog, selain itu LCD oled juga membutuhkan daya 5V. Input LCD Oled tersambung pada *shield wemos* LCD, pada pin D1 (digital 1) dan D2 (digital 2).

Pada diagram perancangan, mikrokontroler wemos D1 mini dihidupkan oleh catu daya yang memiliki daya sebesar 5V. Mikrokontroler wemos D1 mini juga harus tersambung dengan *WiFi* untuk dapat membaca data sensor. *WiFi* akan tersambung otomatis ke mikrokontroler wemos D1 mini ketika ID dan *password* sesuai dengan program yang telah dibuat sebelumnya. Ketika *WiFi* tidak tersambung akan muncul status *disconnect* pada layar LCD, tetapi jika *WiFi* tersambung maka akan muncul status *connected* pada layar LCD. Apabila *WiFi* dan mikrokontroler D1 mini

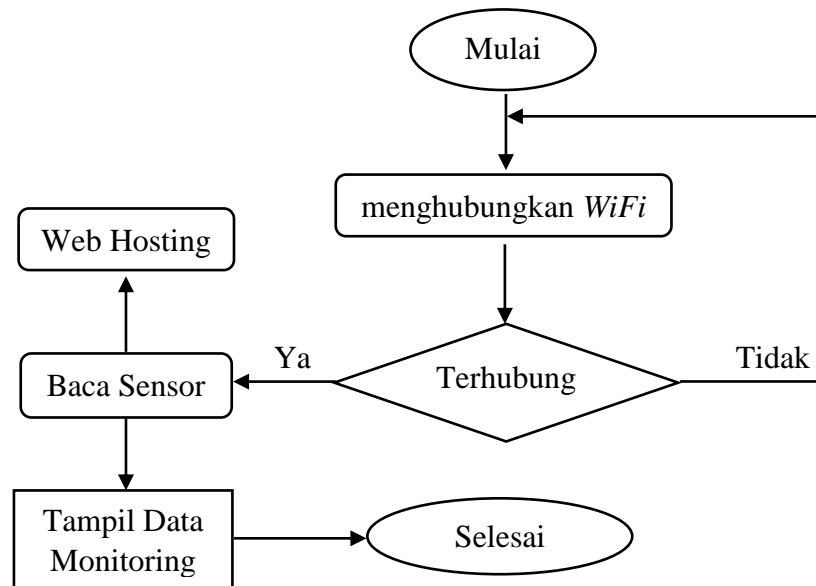
sudah tersambung dan menunjukkan status *connected* pada LCD, maka mikrokontroler D1 mini dapat mengirim perintah pada sensor DHT22 untuk membaca suhu dan kelembaban ruang server. Data suhu ruang server akan selalu muncul pada layar LCD dibawah status *connected*. Data terakhir suhu dan kelembaban ruang server yang muncul pada menit ke 10 akan dikirimkan ke web hosting pada alamat sunupambudi.esy.es.

3.5 Perancangan *Software*

Pada penelitian ini, perancangan dan pembuatan *software* dibagi dalam dua bagian utama yaitu *software* pada sistem monitoring yang dilihat pada LCD Oled dan sistem antarmuka yang dapat di akses menggunakan PC, Laptop dan perangkat elektronik yang terkoneksi dengan internet sebagai sistem monitoring pada gambar 3.4 memperlihatkan flowchart perancangan *software* sistem monitoring dan gambar 3.5 memperlihatkan perancangan *software* antarmuka monitoring.



Gambar 3.4 Flowchart *Software* Sistem Monitoring



Gambar 3.5 Perancangan *Software* Antar Muka Monitoring

Pada penelitian ini, perancangan antarmuka dibangun berdasarkan bahasa pemrograman Arduino IDE & Pemrograman Hyper *Text Markup Language* (HTML) yang telah dilengkapi dengan library agar dapat terkoneksi dengan *hardware* yang telah dibuat.

3.6 Langkah- langkah Pengujian Monitoring Suhu dan Kelembaban

3.6.1 Persiapan Standar

- a. Persiapan Perlengkapan Pengujian
 - 1) Alat monitoring
 - 2) Catu daya
 - 3) *WiFi*
 - 4) Thermometer
 - 5) Form data hasil pengujian
- b. Pemeriksaan Data Pengujian
 - 1) Tanggal pengambilan data
 - 2) Nama tempat pengambilan data
 - 3) Merek, tipe thermometer

3.6.2 Pengujian Kerja

Langkah pengujian monitoring suhu dan kelembaban ruang server berbasis IoT (*Internet of Things*) :

1. Menghubungkan catu daya dengan alat monitoring suhu dan kelembaban ruang server
2. Hidupkan *WiFi*
3. Memastikan kondisi *WiFi* terkoneksi dengan alat
4. Mencatat hasil pembacaan alat setiap 10 menit
5. Mencatat suhu ruang setiap 10 menit menggunakan thermometer

3.7 Pembuatan Program Monitoring Suhu dan Kelembaban

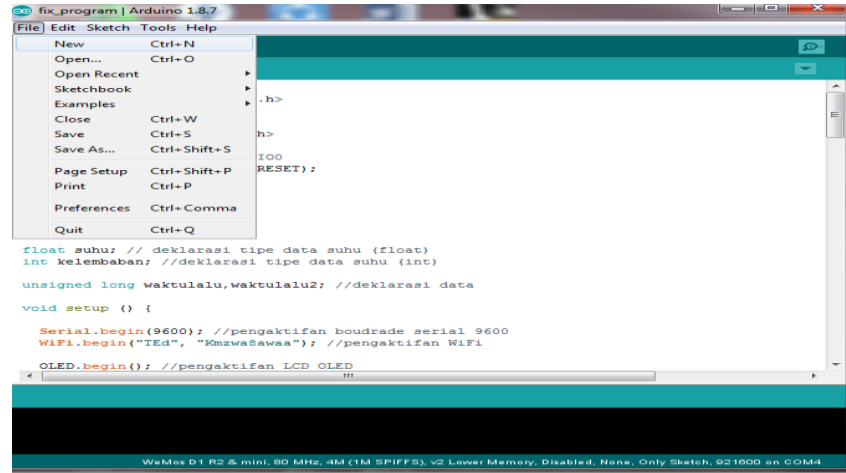
Berikut adalah langkah awal pembuatan program pada Arduino IDE :

1. Membuka program aplikasi Arduino Klik start → klik icon “ Arduino”
lihat gambar 3.6.



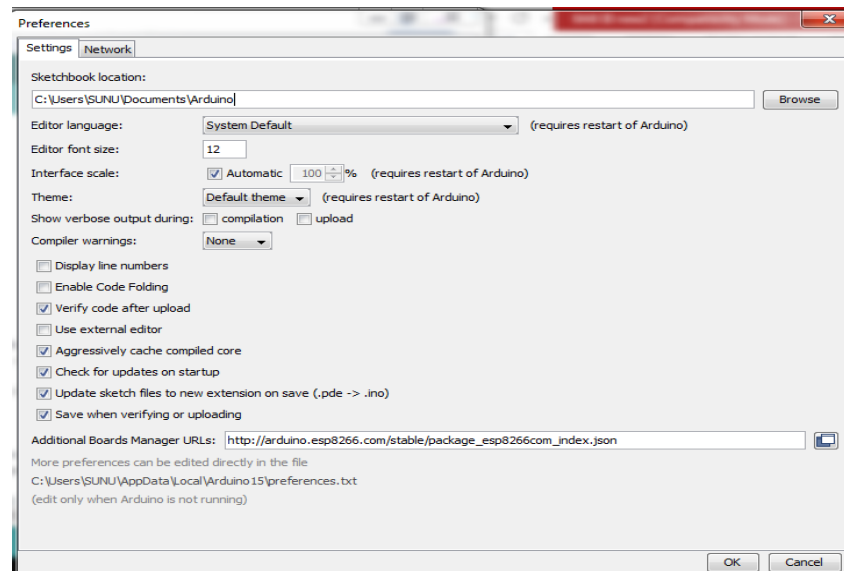
Gambar 3.6 Program Aplikasi Arduino

2. Setelah tampilan program aplikasi Arduino terbuka, selanjutnya membuka *new project*. Untuk lebih jelasnya lihat gambar 3.7.



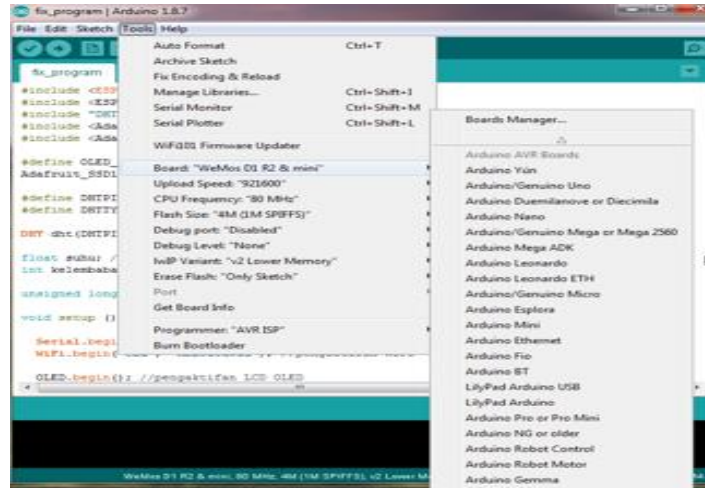
Gambar 3.7 *New Project* Arduino.

3. Setelah itu pilih File lalu klik Preferences, agar wemos dapat terbaca pada bord program arduino isikan alamat sebagai berikut http://arduino.esp8266.com/stable/package_esp8266com_index.json untuk dapat terkoneksi, seperti yang ditunjukkan pada gambar 3.8.



Gambar 3.8 Setting *Bord* Wemos

4. Setelah itu klik Tools pada bord klik Wemos D1 R2 & mini agar dapat membaca program arduino yang telah di buat sepeterti ditunjukkan pada Gambar 3.9.



Gambar 3.9 Setting Bord Manager

Demikian adalah langkah dalam membuat file program monitoring suhu dan kelembaban pada Arduino IDE. Selanjutnya dapat memasukan *library* yang dibutuhkan seperti library LCD OLED, DHT22, Wemos, sesuai program yang akan dibuat agar dapat mendukung berjalannya alat.

Untuk memulai program menggunakan Arduino IDE digunakan Arduino > ESP8266 > DHT22 > ESP8266HTTPClient.h. Kemudian ditambah library DHT22 dan LCD Oled dari Adafruit. Karena DHT22 sudah ber-output digital maka menggunakan pin digital sebagai outputnya, D4 sebagai input digital untuk DHT22. Untuk connect LCD menggunakan LCD oled yang menggunakan 1 input dan 1 output digital, program monitoring suhu dan kelembaban pada tabel 3.1.

Tabel 3.1 Program monitoring Suhu dan Kelembaban

```
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include "DHT.h"
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define OLED_RESET 0 // GPIO0
Adafruit_SSD1306 OLED(OLED_RESET);

#define DHTPIN D4
#define DHTTYPE DHT22
```

Program ini ada dua tipe yaitu suhu dan kelembaban yang bertujuan agar dapat membedakan pembacaan sensor. `Waktulalu`, `waktulalu2` bertujuan untuk mengirim data-data sebelumnya, agar data yang baru tidak bertabrakan dengan data sebelumnya.

Tabel 3.1 Lanjutan Program monitoring Suhu dan Kelembaban

```
DHT dht(DHTPIN, DHTTYPE);

float suhu; // deklarasi tipe data suhu (float)
int kelembaban; //deklarasi tipe data suhu (int)

unsigned long waktulalu,waktulalu2; //deklarasi data
```

Selanjutnya memulai *void setup*. Pada Serial begin, *baud rate* diatur ke 9600 cukup untuk mengirim data ke server. *WiFi begin* merupakan program untuk mengakses *WiFi* agar wemos dapat terkoneksi internet dan dapat mengirimkan data yang terbaca oleh sensor. Ketika wemos belum terkoneksi dengan *WiFi* LCD oled menampilkan status menghubungkan dan ketika *WiFi* tersambung akan merubah status menjadi tersambung dan *connected*.

Tabel 3.1 Lanjutan Program monitoring Suhu dan Kelembaban

```

void setup () {
  Serial.begin(9600); //pengaktifan boudrade serial 9600
  WiFi.begin("TED", "Kmwza8awaa"); //pengaktifan WiFi

  OLED.begin(); //pengaktifan LCD OLED
  OLED.clearDisplay(); //menghapus layar LCD Oled
  OLED.setTextWrap(false);
  OLED.setTextSize(1);
  OLED.setTextColor(WHITE);
  OLED.clearDisplay();
  OLED.println("menyambungkan");
  OLED.display();
  while (WiFi.status() != WL_CONNECTED) {
    delay(100);
    Serial.print("Connecting..");
    OLED.print(".");
    OLED.display();
  }
}

```

Berikutnya mengatur *void loop* untuk perulangan program. Pada tampilan LCD oled menampilkan suhu dan kelembaban dengan jeda waktu 10 menit untuk *upload* ke server. Ketika *WiFi* tersambung maka LCD akan berubah menjadi *connected* di bawahnya akan menampilkan suhu °C dan kelembaban %. Ketika *WiFi* terputus akan berubah menjadi *disconnect*.

Tabel 3.1 Lanjutan Program monitoring Suhu dan Kelembaban

```

void loop() {
  if (WiFi.status() == WL_CONNECTED) {
    sensor();
    OLED.clearDisplay();
    OLED.setTextSize(1);
    OLED.setTextColor(WHITE);
    OLED.setCursor(0,0);
    OLED.println("connected");
    OLED.print("suhu = "); OLED.print(suhu);OLED.println("C");
    OLED.print("kelembaban = ");
    OLED.print(kelembaban);OLED.println("% ");

    if(millis()-waktulalu2>600000)
    {
      kirim();
      waktulalu2=millis();
      for(int i=0; i<20; i++){
        OLED.print(">");delay(10);
        OLED.display();
      }
    }
    OLED.display();
    delay(500);
  }
  else{
    WiFi.begin("TED", "Kmw8awaa");
    while (WiFi.status() != WL_CONNECTED) {
      sensor();
      OLED.clearDisplay();
      OLED.setTextSize(1);
      OLED.setTextColor(WHITE);
      OLED.setCursor(0,0);

      OLED.println("disconnect");
      OLED.print("suhu = "); OLED.print(suhu);OLED.println("°C");
      OLED.print("kelembaban = ");
      OLED.print(kelembaban);OLED.println("% ");
      OLED.display();
      delay(200);
      Serial.print("Connecting..");
    }
  }
}

```

Kemudian untuk mengakses sensor diberi jeda 5 detik untuk membaca suhu dan kelembaban ketika tidak terhubung dalam serial monitor akan *failed* membaca sensor DHT.

Tabel 3.1 Lanjutan Program Monitoring Suhu dan Kelembaban

```
void sensor()
{
  if(millis()-waktulalu >3000){
    kelembaban = dht.readHumidity();
    suhu = dht.readTemperature();
    if (isnan(suhu) || isnan(kelembaban) ) {
      Serial.println("Failed to read from DHT sensor!");
      return;
    }
    Serial.print(suhu); Serial.print(" "); Serial.println(kelembaban);
    waktulalu=millis();
  }
```

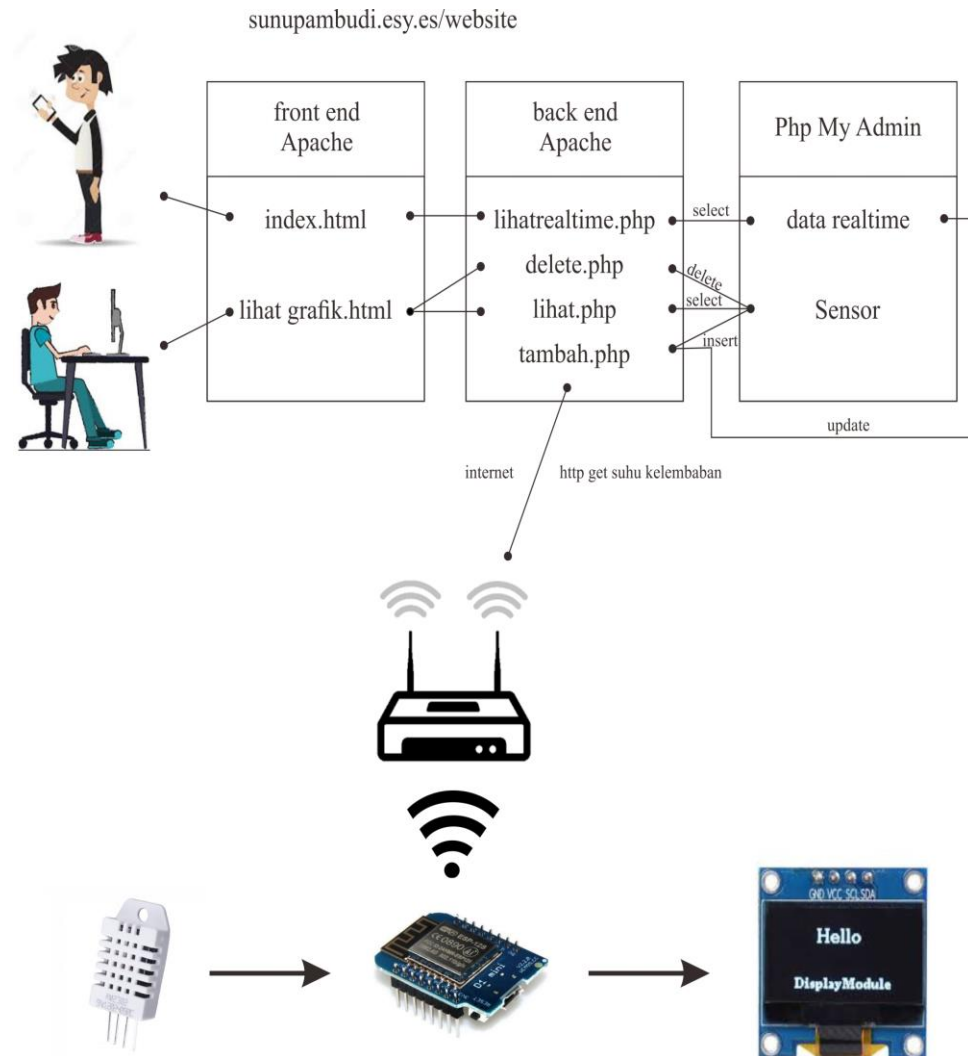
Data di kirim ke server melalui link www.sunupambudi.esy.es menggunakan *String link*. *Tambah.php* merupakan tempat untuk memasukkan data. Data tersebut kemudian dikirim ke database MySQL.

Tabel 3.1 Lanjutan Program monitoring Suhu dan Kelembaban

```
void kirim(){
  if (WiFi.status() == WL_CONNECTED) {
    String link = "http://sunupambudi.esy.es/tambah.php?suhu=";
    link += suhu;
    link += "&kelembaban=";
    link += kelembaban;
    HTTPClient http;
    http.begin(link);
    int httpCode = http.GET(); //Send the request
    Serial.println(link);
  }
  http.end();
}
```

3.8 Pembuatan Program Antarmuka Web

Dalam membuat program antar muka ini diperlukan beberapa proses awal yang harus dilakukan dalam membuat tampilan antarmuka web. Proses pada website ada *fronted* (tampilan depan), *backend* (tampilan belakang) dan database pada PhpMyAdmin. Seperti yang ditunjukkan pada gambar :



Gambar 3.10 Sistem Monitoring Suhu dan Kelembaban Pada Web

3.8.1 Index.html (fronted)

Tabel 3.2 Program Index.html

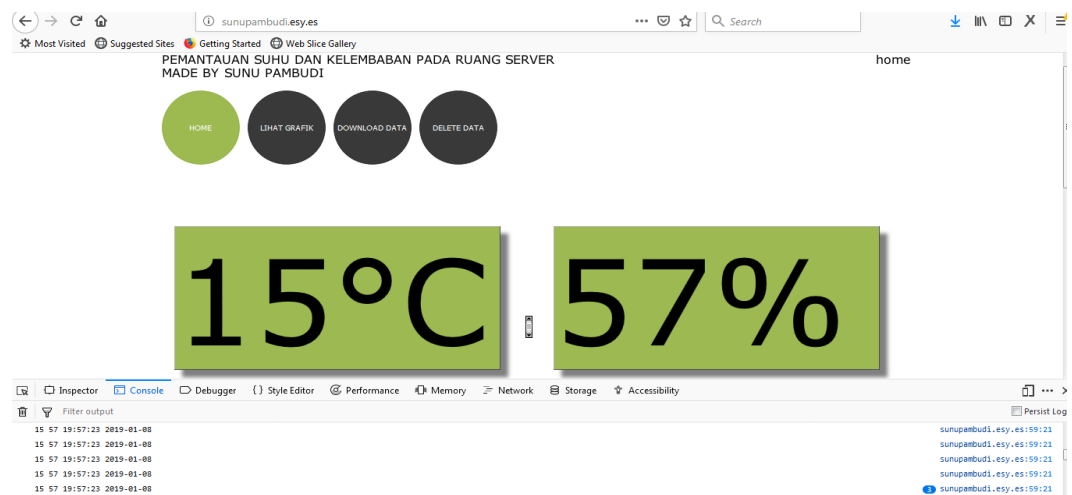
```

<script type="text/javascript">
$.ready(function(){
var suhu,kelembaban,jam,tanggal;
    setInterval(function(e){

        $.ajax({
            'type': 'GET',
            'url': 'lihatrealtime.php',
            'dataType': 'json',
            'data': $(this).serialize(),
            'success': function(data){
                suhu = data.suhu;
                kelembaban = data.kelembaban;
                jam = data.jam;
                tanggal = data.tanggal;
                $('#suhu').html("<p class=\"Body\"><span class=\"C-1\">"+suhu+"°C</span></p>");
                $('#kelembaban').html("<p class=\"Body\"><span class=\"C-1\">"+kelembaban+"%</span></p>");
                $('#waktu').html("<p class=\"Body\"><span class=\"C-2\">Update terakhir jam "+jam+" , tanggal "+tanggal +"</span></p>");
            }
        });
        console.log(suhu+" "+kelembaban+" "+jam+" "+tanggal);
    },500);
});
</script>

```


Program diatas berfungsi mengambil data dari `datarealtime.php` untuk ditampilkan ke *user* yang mengakses data tersebut, data diambil setiap 500ms. Tampilan program dapat di lihat pada gambar 3.11.



Gambar 3.11 Tampilan program `index.html`

3.8.2 `Lihatrealtime.php` (*backend*)

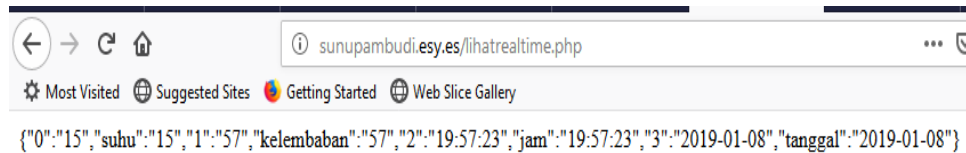
Tabel 3.3 Program `lihatrealtime.php`

```
<?php

$db=mysqli_connect("localhost","u349353524_data","12345678","u34935
3524_data") or die("gagal");

$query = mysqli_query($db,"SELECT * FROM `data realtime` WHERE
1") or die("salah");
$lokasi = mysqli_fetch_array($query);
echo json_encode($lokasi);
?>
```

Program diatas bertujuan untuk mengambil semua data dari *database* pada *realtime* dan menampilkan dalam bentuk json. Seperti pada gambar 3.12.



Gambar 3.12 Tampilan lihatrealtime.php

3.8.3 Lihatgrafik.html (*frontend*)

Tabel 3.4 Program lihatgrafik.html

```
<script>
document.getElementById("tanggal1").value = format(new Date(), 'yyyy-
MM-dd');
document.getElementById("tanggal2").value = format(new Date(), 'yyyy-
MM-dd');
var tanggal1 = format(new Date(), 'yyyy-MM-dd');
var tanggal2 = format(new Date(), 'yyyy-MM-dd');
function cari() {
    tanggal1 = document.getElementById("tanggal1").value;
    tanggal2 = document.getElementById("tanggal2").value;
    console.log(tanggal1+" "+tanggal2);
}
var ctx1;
var myLineChart1;
var ctx2;
var myLineChart2;
```

Program ini terdiri dari 3 fungsi sistem yang terdiri :

Tabel 3.4 Program lihatgrafik.html (lanjutan)

```
$(document).ready(function(){
    setInterval(function(e){
        console.log(tanggal1+" "+tanggal2);
        $.ajax({
            'type': 'GET',
            'url':
'lihat.php?tanggal1='+tanggal1+'&tanggal2='+tanggal2,
            'dataType': 'json',
            'data': $(this).serialize(),
            'success': function(data){
```

Fungsi 1 program fungsi ajax melakukan perintah ke lihat.php untuk mengambil data.

Tabel 3.4 Program lihatgrafik.html (lanjutan)

```

var akhir = Object.keys(data).length;
var awal = akhir - 100;
var x1 = []var x2 = [];
var y = [];

if ( awal < 0){awal = 0;}
console.log(awal+" "+akhir);
for ( var i =awal; i < akhir; i++)
{
    var jadwal = data[i].tanggal + " " + data[i].jam;
    x1.push(data[i].suhu);
    x2.push(data[i].kelembaban);
    y.push(jadwal);
    tabel += "<tr style=\"height:30px;\"><td class=\"TC-2\"><p class=\"Table-Body\"><span class=\"C-2\">";
    tabel += data[i].tanggal;
    tabel += "</span></p></td><td class=\"TC-2\"><p class=\"Table-Body\"><span class=\"C-2\">";
    tabel += data[i].jam;
    tabel += "</span></p></td><td class=\"TC-2\"><p class=\"Table-Body\"><span class=\"C-2\">";
    tabel += data[i].suhu;
    tabel += "</span></p></td><td class=\"TC-2\"><p class=\"Table-Body\"><span class=\"C-2\">";
    tabel += data[i].kelembaban;
    tabel += "</span></p></td><td class=\"TC-2\"><p class=\"Table-Body\"><span class=\"C-2\">";
    tabel += "<a href=\"delete.php?tanggal="
    tabel += data[i].tanggal;
    tabel += "&jam=";
    tabel += data[i].jam;
    tabel += "> delete </a>";
    tabel += "</span></p></td></tr>";
}
tabel += "</table>";
console.log(tabel);
$('#tabelku').html(tabel);

```

Fungsi 2 program tersebut berfungsi untuk membuat data yang diperoleh dari fungsi 1 menjadi sebuah table dan terdapat program yang mengakses delete.php untuk keperluan menghapus data.

Tabel 3.4 Program lihatgrafik.html (lanjutan)

```
ctx1 = document.getElementById("chartsuhu");

myLineChart1 = new Chart(ctx1, {

  type: 'line',
  options: {
    animation: {
      duration: 0
    }
  },
  data: {
    labels: y,
    datasets: [{
      data: x1,
      label: "suhu",
      borderColor: "#3e95cd",
      fill: false
    } ] }
});

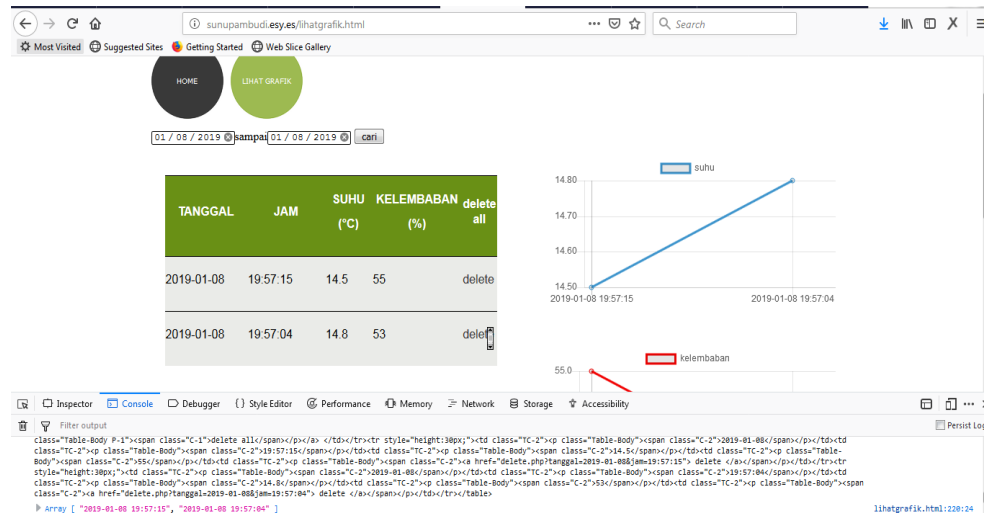
ctx2 = document.getElementById("chartkelembaban");

myLineChart2 = new Chart(ctx2, {

  type: 'line',
  options: {
    animation: {
      duration: 0
    }
  },
  data: {
    labels: y,
    datasets: [{
      data: x2,
      label: "kelembaban",
      borderColor: "#f20404",
      fill: false
    } ]
  }
});

console.log(y);
```

Fungsi 3 program ini berfungsi untuk membuat data yang diperoleh dari fungsi 1 menjadi sebuah *chart*. Program diatas menggunakan fungsi *interval* yang akan mengakses data setiap 2000ms.



Gambar 3.13 Tampilan lihatgrafik.html

3.8.4 Lihat.php (backend)

Tabel 3.5 Program lihat.php

```
<?php
$db=mysqli_connect("localhost","u349353524_data","12345678","u34935
3524_data") or die("gagal");
if(isset($_GET['tanggal1']) && isset($_GET['tanggal2']))
{
$query = mysqli_query($db,"SELECT * FROM sensor WHERE
tanggal>=$_GET[tanggal1] AND tanggal<=$_GET[tanggal2] ORDER
BY tanggal DESC,jam DESC ");
$array=array();
$n = 0;
while ($lokasi = mysqli_fetch_array($query)){
    $n +=1;
    $array2=array();
    $array2['id']=$n;
    $array2['suhu']=$lokasi['suhu']      ;
    $array2['kelembaban']= $lokasi['kelembaban'];
    $array2['tanggal']= $lokasi['tanggal'];
    $array2['jam']= $lokasi['jam'];
```

Tabel 3.5 Program lihat.php (Lanjutan)

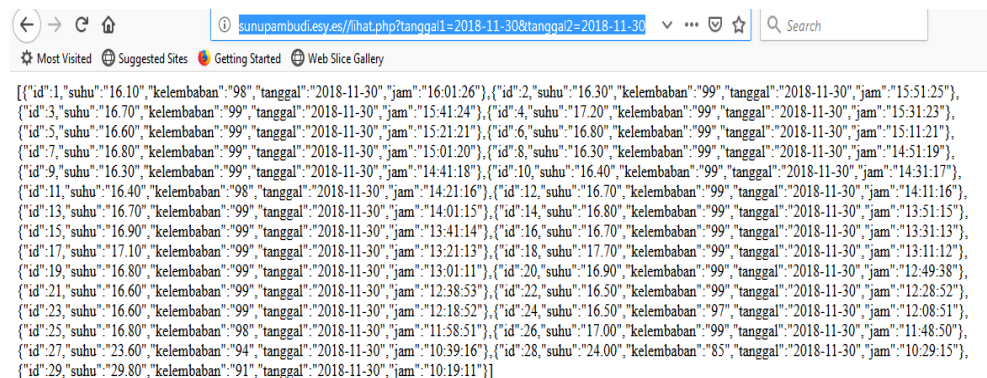
```

        $array[]=$array2;
    }
    echo json_encode($array);
}
else
{
$query = mysqli_query($db,"SELECT * FROM sensor WHERE 1
ORDER BY tanggal DESC ");
$array=array();
$n = 0;
while ($lokasi = mysqli_fetch_array($query)){
    $n +=1;
    $array2=array();
    $array2['id']= $n;
    $array2['suhu']=$lokasi['suhu']      ;
    $array2['kelembaban']= $lokasi['kelembaban'];
    $array2['tanggal']= $lokasi['tanggal'];
    $array2['jam']= $lokasi['jam'];
    $array[]=$array2;
}
echo json_encode($array);
}

?>

```

Program diatas bertujuan untuk mengambil semua data dari *database*, pada table sensor sesuai dengan tanggal yang diberikan dan menampilkannya dalam bentuk json (aplikasi) dapat dilihat pada gambar dibawah ini :



Gambar 3.13 Tampilan lihat.php

3.8.5 Delete.php (*backend*)

Tabel 3.6 Program delete.php

```
<?php
$db=mysqli_connect("localhost","u349353524_data","12345678","u34935
3524_data") or die("gagal");

if(isset($_GET['tanggal']) && isset($_GET['jam']))
{
    mysqli_query($db,"DELETE FROM sensor WHERE
tanggal=$_GET[tanggal] AND jam=$_GET[jam]")or die ("gagal");
}
if(isset($_GET['hapus']))
{
    mysqli_query($db,"DELETE FROM sensor WHERE 1 ")or die ("gagal");
}
header("Location: http://sunupambudi.esy.es/lihatgrafik.html");
?>
```

Program diatas bertujuan untuk *delete* data pada *database* tabel sensor dengan tanggal dan jam yang diberikan.

3.8.6 Tambah.php (*backend*)

Tabel 3.7 Program tambah.php

```
<?php
date_default_timezone_set('Asia/Jakarta');
$tanggal=date("Y/m/d");
$jam=date("H:i:sa");
$db=mysqli_connect("localhost","u349353524_data","12345678","u34935
3524_data") or die("gagal");
if(isset($_GET['suhu']))
{
    mysqli_query($db,"INSERT INTO `sensor`(`suhu`,
`kelembaban`,`tanggal`,`jam`) VALUES
($_GET[suhu],$_GET[kelembaban], $tanggal, $jam)") or die("gagal
insert");
    mysqli_query($db,"UPDATE `data realtime` SET
suhu=$_GET[suhu],kelembaban=$_GET[kelembaban],jam=$_jam',
tanggal=$tanggal' ") or die("gagal realtime");
    echo "haloo ",$_GET['suhu'];
}
}
```

Program diatas bertujuan untuk menambahkan data pada tabel sensor dan memperbaharui data pada tabel data realtime sesuai dengan data yang diberikan.

3.8.7 Wemos Program (Arduino.ide)

Tabel 3.8 Wemos Program

```
if (WiFi.status() == WL_CONNECTED) {  
  
    String link = "http://sunupambudi.esy.es/tambah.php?suhu=";  
    link += suhu;  
    link += "&kelembaban=";  
    link += kelembaban;  
  
    HTTPClient http;  
    http.begin(link);  
    int httpCode = http.GET(); //Send the request  
    Serial.println(link);  
  
    http.end();  
}
```

Program diatas bertujuan untuk mengirimkan data sensor wemos ke tambah.php.