

## BAB V

### IMPLEMENTASI DAN PENGUJIAN SISTEM

#### 5.1 HASIL IMPLEMENTASI

Pada tahap ini penulis mengimplementasikan hasil rancangan yang telah dibuat. Adapun hasil implementasi dari penelitian ini di jelaskan seperti berikut ini.

#### 5.2 HASIL PERANCANGAN PERANGKAT KERAS

Berikut ditampilkan hasil rancangan perangkat keras *water resistant* untuk perkebunan karet:



Gambar 5.1 Alat *watter resistant* getah pohon karet

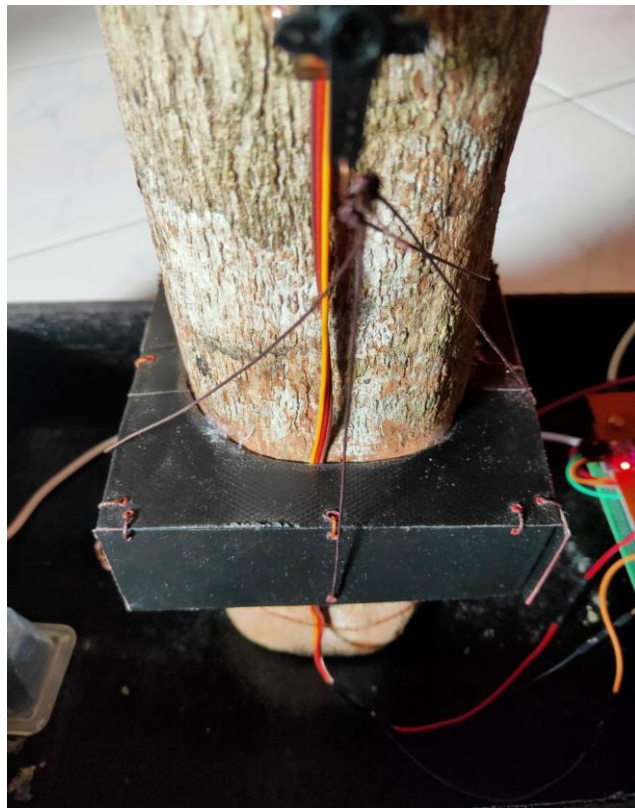
Dari gambar 5.1 terlihat bentuk fisik hasil rancangan alat *water resistant* untuk pekebunan karet secara keseluruhan. Peneliti menggunakan sensor rintik hujan *raindrop* untuk membaca kondisi cuaca dengan posisi paling atas bertujuan untuk membaca terjadinya hujan sebelum sampai di penampang getah karet dan menggunakan motor servo untuk menggerakkan penutup penampang getah karet. Alat *watter resistant* getah pohon karet ini akan bekerja secara otomatis jika sensor *raindrop* mendeteksi adanya air hujan. Dengan terdeteksinya air hujan maka motor servo akan bekerja secara otomatis untuk menggerakkan penutup getah pohon karet. Lalu untuk lebih memastikan sensor bekerja atau tidak, maka nilai sensor dapat di ketahui melalui web.

### 5.2.1 Implementasi Alat Jika Tidak Terdeteksi Air Hujan



**Gambar 5.2** Sensor *raindrop* tidak mendeteksi adanya air hujan

Pada gambar 5.2 di atas terlihat tidak ada air yang mengenai permukaan sensor *raindrop* sehingga dia mengirimkan data ke esp 8266 bahwa tidak ada air yang terdeteksi.



**Gambar 5.3 Keadaan penutup tidak bekerja jika tidak terjadi hujan**

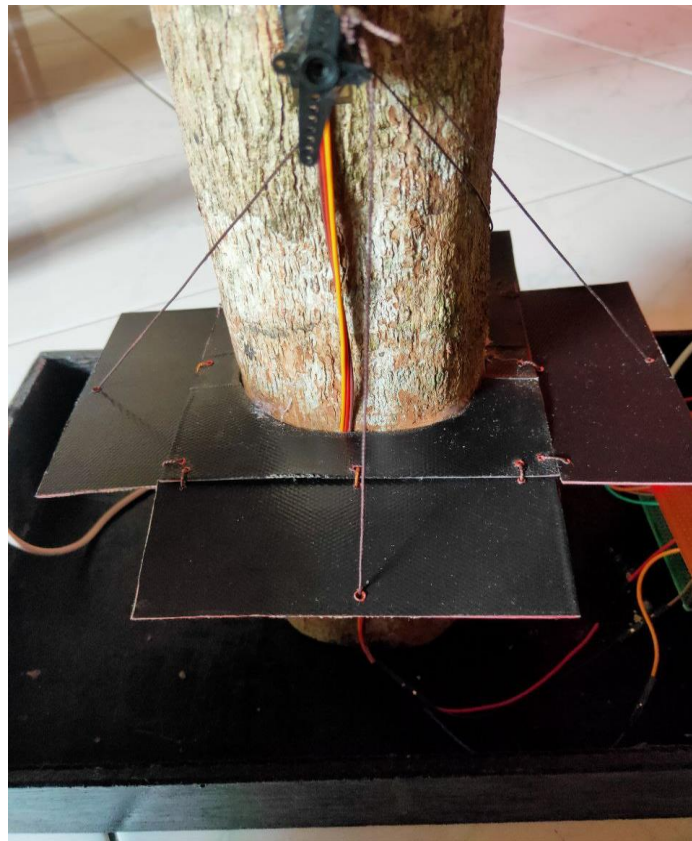
Dari gambar 5.3 terlihat motor servo tidak bekerja sehingga penutup getah karet tidak membuka, karena esp8266 mengirimkan data ke servo jika tidak adanya air hujan yang terdeteksi.

### 5.2.2 Implementasi Alat Jika Terdeteksi Air Hujan



**Gambar 5.4** Sensor *raindrop* mendeteksi adanya air hujan

Pada gambar 5.4 di atas terlihat ada air yang mengenai permukaan sensor *raindrop* sehingga dia mengirimkan data ke esp8266 bahwa ada air yang terdeteksi dan akan diteruskan ke motor servo untuk bekerja menggerakkan penutup getah karet.



**Gambar 5.5 Keadaan penutup bekerja jika terjadi hujan**

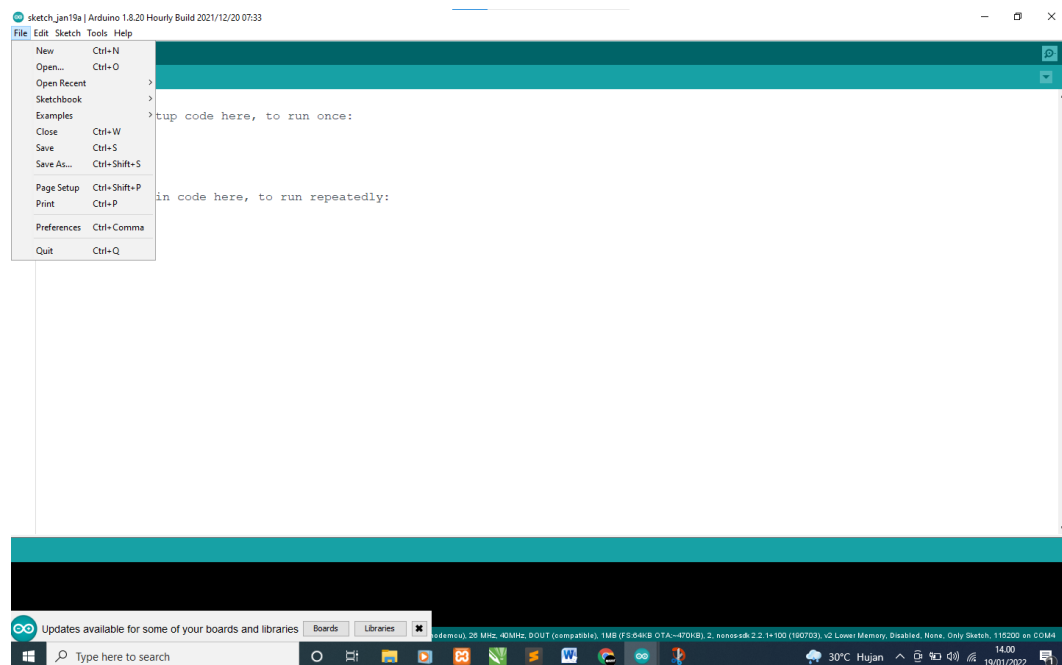
Dari gambar 5.5 terlihat motor servo bekerja sehingga penutup getah karet membuka untuk menghalang getah karet terkenannya air hujan. Esp 8266 mengirimkan data ke servo untuk berkerja , karena sensor *raindrop* mendeteksi adanya air hujan.

## 5.3 PENGUJIAN PERANGKAT LUNAK

### 5.3.1 *Arduino Ide*

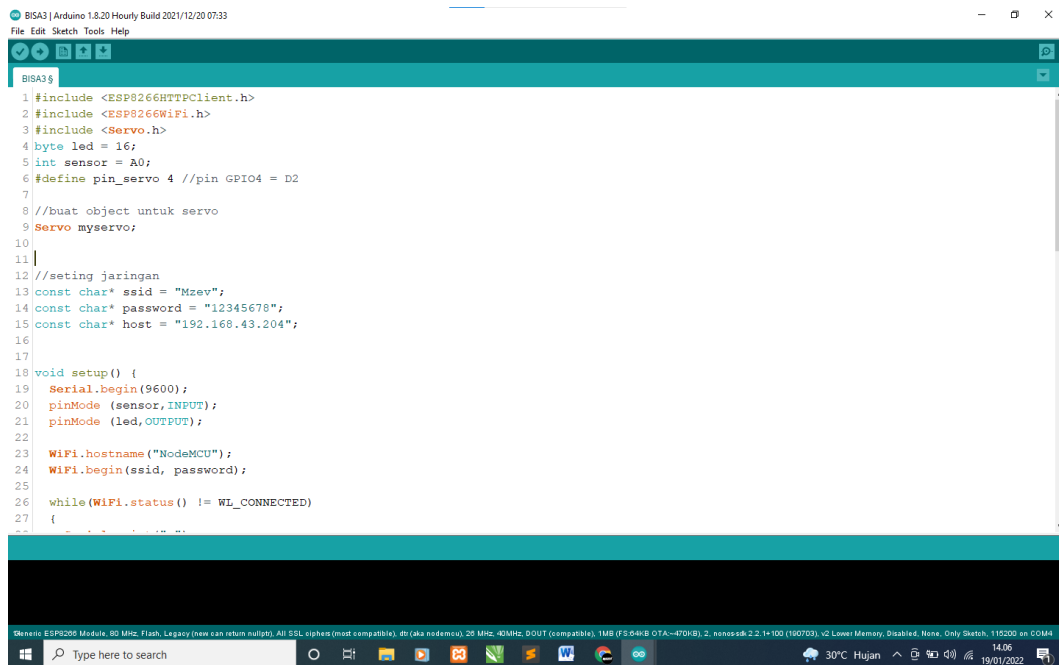
*Arduino Ide* digunakan untuk membuat *sketch* program, *Arduino IDE* ini berguna untuk mengedit, membuat, meng-*upload* ke *board* yang ditentukan, dan meng-*coding* program tertentu. Pada penelitian ini arduino ide digunakan untuk memprogram *ESP8266*.

Cara pertama yaitu dengan menginstal seperti biasa software *Arduino Ide*, jika sudah lalu buka software arduino ide. Selanjutnya untuk membuat sebuah *project* baru silahkan pilih menu *file* lalu pilih *New*, maka nanti akan ada lembar halaman kosong tempat untuk membuat program.



**Gambar 5.6** Tampilan awal *arduino ide*

Selanjutnya membuat listing program sesuai dengan perintah yang diinginkan dan juga harus memperhatikan beberapakode program agar nantinya tidak terjadi error yang mengakibatkan program tidak jalan.



```

1 #include <ESP8266HTTPClient.h>
2 #include <ESP8266WiFi.h>
3 #include <Servo.h>
4 byte led = 16;
5 int sensor = A0;
6 #define pin_servo 4 //pin GPIO4 = D2
7
8 //buat object untuk servo
9 Servo myservo;
10
11
12 //seting jaringan
13 const char* ssid = "Mzev";
14 const char* password = "12345678";
15 const char* host = "192.168.43.204";
16
17
18 void setup() {
19   Serial.begin(9600);
20   pinMode (sensor, INPUT);
21   pinMode (led, OUTPUT);
22
23   WiFi.hostname("NodeMCU");
24   WiFi.begin(ssid, password);
25
26   while(WiFi.status() != WL_CONNECTED)
27   {

```

ESP8266 Module, 80 MHz, Flash, Legacy (new can return multiple), All SSL cipher (most compatible), 40 (aka nodemcu), 20 MHz, 40MHz, DOUT (compatible), 1MB (FS 0-4MB OTA~470KB), 2, nvsflash 2.2, 1+100 (160703), v2 Lower Memory, Disabled, None, Only Sketch, 115200 on COM4

Type here to search 30°C Hujan 14:06 19/01/2022

**Gambar 5.7** Tampilan listing program *arduino ide*

Langkah selanjutnya jika program sudah selesai dibuat maka kita perlu memasukan program tersebut ke mikrokontroler esp8266. Caranya yaitu dengan mengupload program tersebut dan tunggu prosesnya sampai selesai dan pastikan tidak ada program yang error.

```

51 Serial.println("Connection failed.");
52 return;
53 }
54
55 //baca air
56 String Link;
57 HTTPClient http;
58 Link = "http://192.168.43.204/SK/bacaair.php?sensor=" + String(data_sensor);
59
60 http.begin(Link);
61 http.GET();
62 http.end();
63
64
65 if(data_sensor < 1000 )
66 {
67   myservo.write(180);
68   Serial.println("Servo On");
69 }
70
71 else if (data_sensor >1000 )
72 {
73   myservo.write(0);
74   Serial.println("Servo Off");
75 }
76
77 delay(1000);
78 }
  
```

Done uploading.  
Leaving...  
Hard resetting via RTS pin...

Senoeng ESP8266 Module, 80 MHz, Flash, Legacy (new can return nullptr), All SSL cipher (most compatible), dt (aka nodemcu), 20 MHz, 40MHz, DOUT (compatible), 1MB (FS 5.0MB OTA~470KB), 3, nonosdk 2.2.1+100 (190703), v2 Lower Memory, Disabled, None, Only Sketch, 115200 on COM4

**Gambar 5.8** Tampilan proses upload program *Arduino Ide*

Untuk melihat program yang sudah di upload berjalan dengan baik maka bisa di lihat melalui *serial monitor* untuk mengetahui nilai sensor.

```

1 #include <ESP8266HTTPClient.h>
2 #include <ESP8266WiFi.h>
3 #include <Servo.h>
4 byte led = 16;
5 int sensor = A0;
6 #define pin_servo 4 //
7
8 //buat object untuk servo
9 Servo myservo;
10
11
12 //setting jaringan
13 const char* ssid = "M...";
14 const char* password = "...";
15 const char* host = "192.168.43.204";
16
17
18
19
20 void setup() {
21   Serial.begin(9600);
22   pinMode (sensor, INPUT);
23   pinMode (led, OUTPUT);
24
25   WiFi.hostname("NodeMCU");
26   WiFi.begin(ssid, password);
27
  
```

Servo Off  
1024  
Servo Off  
1024  
Servo Off  
1024  
Servo Off  
1024  
Servo Off  
1024  
Servo Off  
1024  
Servo Off  
1024  
Servo Off  
1024  
Servo Off  
1024

Autoscroll  Show timestamp  
No line ending | 9600 baud | Clear output

Senoeng ESP8266 Module, 80 MHz, Flash, Legacy (new can return nullptr), All SSL cipher (most compatible), dt (aka nodemcu), 20 MHz, 40MHz, DOUT (compatible), 1MB (FS 5.0MB OTA~470KB), 3, nonosdk 2.2.1+100 (190703), v2 Lower Memory, Disabled, None, Only Sketch, 115200 on COM4

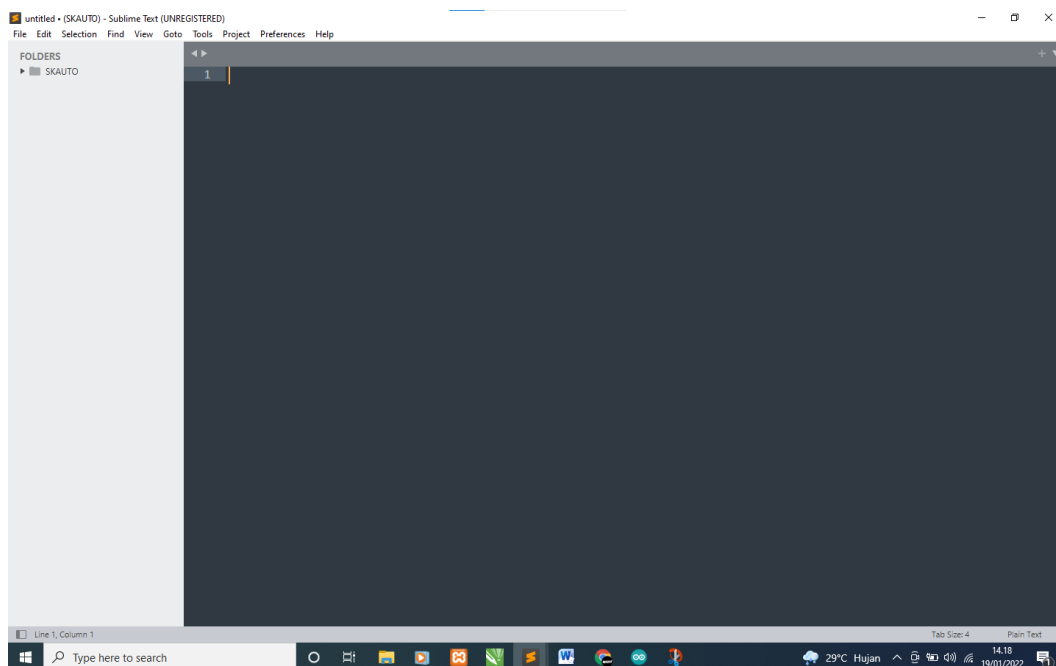
**Gambar 5.9** Tampilan *serial monitor* arduino ide



### 5.3.2 *Sublim Text*

*Sublim text* digunakan sebagai *software text* editor yang sangat berguna untuk menulis sejumlah code serta mampu membuka berbagai macam jenis file. Dimana pada penelitian ini digunakan untuk membuat web.

Cara yang pertama dengan menginstal software *sublim text* seperti biasa, lalu buka software tersebut maka nanti hanya ada halaman kosong yang belum ada kode program yang di buat, dimana halaman tersebut merupakan tempat untuk membuat program nantinya.



**Gambar 5.10** Tampilan awal *sublim text*

Selanjutnya yaitu membuat program untuk pembuatan webnya, dan juga perlu memperhatikan beberapa kode program agar nantinya tidak terjadi eror pada halaman web.

```

4  $sql = mysqli_query($koneksi, "SELECT * FROM tb_sensor");
5  $data = mysqli_fetch_array($sql);
6
7  $nilai = $data['nilaiair'];
8  $servo = $data['servo'];
9
10 }
11
12 <!doctype html>
13 <html lang="en">
14
15 <!-- Required meta tags -->
16 <meta charset="utf-8">
17 <meta name="viewport" content="width=device-width, initial-scale=1">
18 <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
19 <script type="text/javascript" src="jquery/jquery.min.js"></script>
20
21 <!-- Bootstrap CSS -->
22 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
  integrity="sha384-1BmE4KwBq78iYhF1dvKuhFTAU6auU8tT94WrHftjDbrCEXSU1oBoqy12QvZ6jW3" crossorigin="
  anonymous">
23
24 <title>Web Skripsi</title>
25
26 <script type="text/javascript">
27
28   function ubahposisi(value)
29   {
30     document.getElementById("posisi").innerHTML = value ;
31     //ajak untuk merubah nilai status servo
32     var xmlhttp = new XMLHttpRequest();
33
34     xmlhttp.onreadystatechange = function()
35     {
36       if (xmlhttp.readyState == 4 && xmlhttp.status == 200)
37       {
38         //ambil respon dari web setelah berhasil merubah nilai
39         document.getElementById("posisi").innerHTML = xmlhttp.responseText;

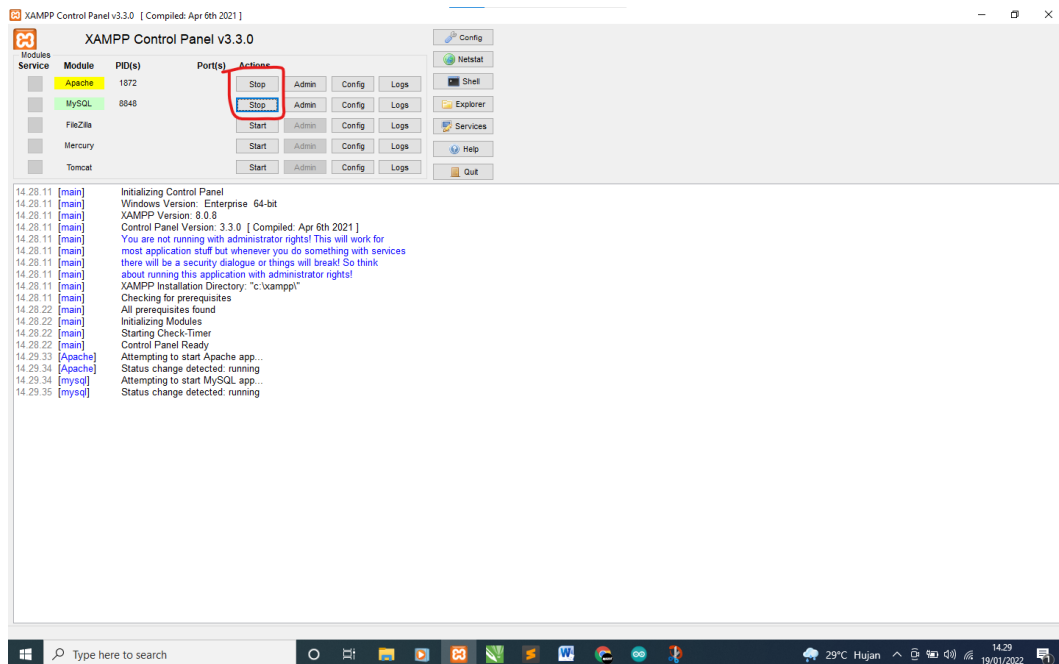
```

**Gambar 5.11** Tampilan program web pada *sublim text*

### 5.3.3 Xampp

*Xampp* digunakan untuk menjadi server lokal yang menyimpan data sebuah website dan juga web server. *XAMPP* digunakan untuk menguji kinerja fitur dari website tersebut dan menampilkan konten website tersebut tanpa memerlukan koneksi internet.

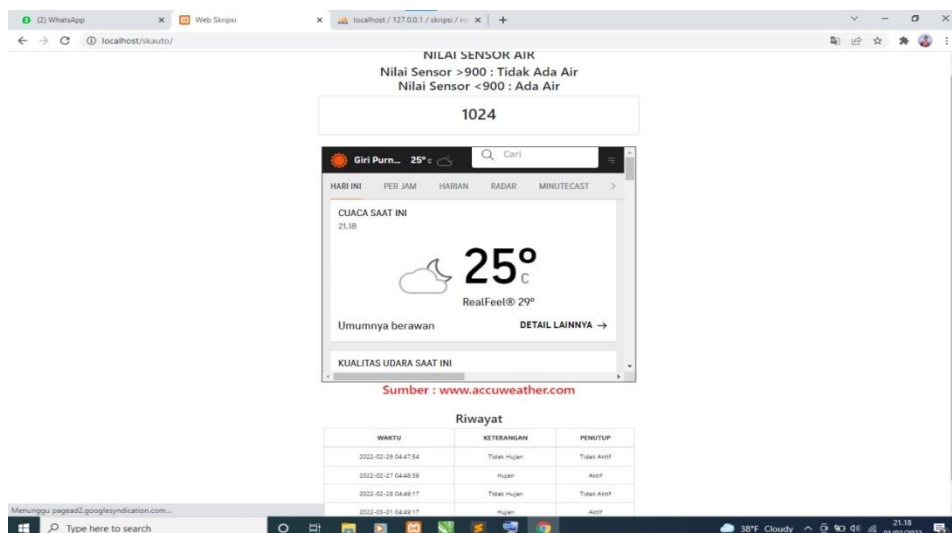
Langkah pertama dengan menginstal software *Xampp* seperti biasa, lalu jika setelah selesai akan tampil menu awal dari software *Xamp*. Untuk menjalankan software ini fitur yang perlu di aktifkan yaitu Apache dan Mysql , maka dari itu harus mengaktifkan terlebih dahulu.



Gambar 5.12 Tampilan *software xampp*

### 5.3.4 Web

Fungsi utama dari *web* disini yaitu digunakan untuk menampilkan data nilai sensor raindrop untuk dapat dilihat pada tampilan *web*, dan juga *web* digunakan untuk menampilkan informasi prakiraan cuaca.



Gambar 5.13 Tampilan halaman *web*

## 5.4 PENGUJIAN PERANGKAT KERAS

### 5.4.1 Pengujian Tegangan Sumber

Tahap pertama yang dilakukan adalah pengujian tegangan sumber, yang mana tegangan sumber di hasilkan dari adaptor.

**Table 5.1 Pengujian Tegangan Sumber**

| <b>Sumber Arus</b> | <b>Tegangan <i>Input</i></b> | <b>Tegangan <i>Output</i></b> |
|--------------------|------------------------------|-------------------------------|
| <b>Adaptor</b>     | <b>5 V</b>                   | <b>5 V</b>                    |

### 5.4.2 Pengujian Tegangan ESP 8266

Setelah melakukan pengujian tegangan sumber, selanjutnya menguji tegangan ESP 8266.

**Table 5.2 Pengujian tegangan esp 8266**

| <b>Sumber</b>   | <b>Tegangan <i>Input</i></b> | <b>Tegangan <i>Output</i></b> |
|-----------------|------------------------------|-------------------------------|
| <b>ESP 8266</b> | <b>5 V</b>                   | <b>3.3 V</b>                  |

### 5.4.3 Pengujian Tegangan Senror *Raindrop*

Setelah melakukan pengujian tegangan ESP 8266, selanjutnya pengujian tegangan untuk sensor *Raindrop*.

**Table 5.3 Pengujian tegangan sensor *raindrop***

| <b>Sumber</b>          | <b>Tegangan <i>Input</i></b> | <b>Tegangan <i>Output</i></b> |
|------------------------|------------------------------|-------------------------------|
| <b><i>Raindrop</i></b> | <b>3.3 V</b>                 | <b>3.3 V</b>                  |

#### 5.4.4 Pengujian Motor Servo

Pengujian selanjutnya yaitu pengujian pada gerak motor servo untuk menggerakkan penutup.

**Table 5.4 Pengujian motor servo**

| Percobaan | Nilai Sensor | Motor Servo | Posisi Servo |
|-----------|--------------|-------------|--------------|
| 1         | 1024         | OFF         | 0 Derajat    |
| 2         | 900          | ON          | 180 Derajat  |
| 3         | 1005         | OFF         | 0 Derajat    |
| 4         | 590          | ON          | 180 Derajat  |

#### 5.4.5 Pengujian Sensor *Raindrop*

Selanjutnya yaitu pengujian pada sensor *Raindrop* untuk mengukur adanya air hujan atau tidak.

**Table 5.5 Pengujian sensor *raindrop***

| Percobaan | Kondisi | Nilai Sensor | Akurasi |
|-----------|---------|--------------|---------|
| 1         | Embun   | 1001         | 90%     |
| 2         | Hujan   | 600          | 95%     |
| 3         | Kering  | 1024         | 100%    |

## 5.5 ANALISA SISTEM SECARA KESELURUHAN

Untuk mendeteksi apabila terjadi kesalahan setelah uji coba, maka perlu dilakukan analisa rangkaian secara keseluruhan. Dari seluruh proses yang telah dilakukan, baik pengujian perangkat keras maupun perangkat lunak, dapat dikatakan bahwa alat ini dapat berfungsi sebagaimana yang penulis inginkan. Proses pembacaan sensor *raindrop* tidak terjadi kesalahan pembacaan data, dan pada motor servo penutup getah karet juga berfungsi dengan baik, lalu nilai sensor bisa di lihat pada halaman *web* beserta informasi cuaca.

Pengujian ini dilakukan untuk menunjukkan bahwa sistem watter resistant getah pohon karet dapat bekerja sesuai dengan tujuan dari pembuatan. Pengujian ini dilakukan dengan cara sebagai berikut :

1. Hal pertama yang dilakukan adalah menguji sensor *raindrop*, jika permukaan sensor terkena air hujan maka sensor akan mengirim nilai kepada ESP 8266.
2. Selanjutnya ESP 8266 akan melanjutkan mengirim perintah ke motor servo untuk menggerakkan penutup getah pohon karet agar tidak terkena air hujan.
3. Selanjutnya untuk bisa melihat nilai sensor tersebut maka ESP 8266 mengirim nilainya ke halaman *web*.
4. Pada halaman *web* bisa digunakan untuk melihat nilai yang dikirimkan oleh sensor *raindrop* dan juga bisa melihat informasi prakiraan cuaca.

**Table 5.6 Pengujian keseluruhan**

| <b>Percobaan</b> | <b>Nilai Sensor<br/>Raindrop</b> | <b>Motor Servo</b> | <b>Posisi Servo</b> | <b>Penutup<br/>Getah Karet</b> |
|------------------|----------------------------------|--------------------|---------------------|--------------------------------|
| <b>1</b>         | <b>1024</b>                      | <b>OFF</b>         | <b>0 Derajat</b>    | <b>Tidak<br/>Bekerja</b>       |
| <b>2</b>         | <b>850</b>                       | <b>ON</b>          | <b>180 Derajat</b>  | <b>Bekerja</b>                 |
| <b>3</b>         | <b>760</b>                       | <b>ON</b>          | <b>180 Derajat</b>  | <b>Bekerja</b>                 |
| <b>4</b>         | <b>970</b>                       | <b>OFF</b>         | <b>0 Derajat</b>    | <b>Tidak<br/>Bekerja</b>       |
| <b>5</b>         | <b>612</b>                       | <b>ON</b>          | <b>180 Derajat</b>  | <b>Bekerja</b>                 |