

# **LAPORAN AKHIR TUGAS BESAR SISTEM TERTANAM**

## **PERANCANGAN SISTEM PENJEMUR KOPI OTOMATIS MENGUNAKAN FUZZY LOGIC MAMDANI**

Diajukan Kepada Pengampu Mata Kuliah Sistem Tertanam  
untuk Memenuhi Tugas Besar



**Dosen Pengampu :**

**Ilham Firman Ashari, S.Kom., M.T**

**Disusun Oleh:**

<b>Syabana Minggu Noviantosa</b>	<b>118140122</b>
<b>Solafide Zamili</b>	<b>121140201</b>
<b>Marchel Ferry Timoteus Samosir</b>	<b>121140195</b>
<b>Hamas Azhar</b>	<b>119140138</b>

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**INSTITUT TEKNOLOGI SUMATERA**

**2023**

# DAFTAR ISI

<b>DAFTAR ISI</b>	<b>2</b>
<b>BAB I</b>	<b>2</b>
<b>PENDAHULUAN</b>	<b>2</b>
1.1 Latar Belakang	3
1.2 Rumusan Masalah	3
1.3 Tujuan Perancangan	4
<b>BAB II</b>	<b>5</b>
<b>DASAR TEORI</b>	<b>5</b>
2.1 Arduino IDE	5
2.2 Breadboard	5
2.3 Kabel Jumper	5
2.4 Sensor Suhu dan Kelembaban (DHT11)	5
2.5 Motor Servo	6
2.6 Sensor BMP180	6
2.7 Sensor MD0127	6
2.8 Mini ZVS	6
2.9 Raspberry pi	6
<b>BAB III</b>	<b>7</b>
<b>METODE</b>	<b>7</b>
3.1 Desain Perangkat (Hardware & Software)	7
3.1.1 Flowchart	8
3.1.2 Gambaran Umum Sistem	9
3.1.3 Schematic Diagram	10
3.2 Skema Pengujian	11
3.2.1 Skema Pengujian perangkat keras	11
3.2.2 Skema Pengujian Fuzzy Logic	12
<b>BAB IV</b>	<b>13</b>
<b>HASIL DAN PEMBAHASAN</b>	<b>13</b>
4.1 Kode program	13
Kode Arduino	13
Kode Raspberry pi	15
4.2 Hasil Pengujian	22
4.2.1 Skema Pengujian perangkat keras	22
4.2.2 Skema Pengujian Fuzzy Logic	24
<b>BAB V</b>	<b>25</b>
<b>KESIMPULAN</b>	<b>25</b>
5.1 Kesimpulan	25
5.2 Saran	25
<b>DAFTAR PUSTAKA</b>	<b>26</b>
<b>LOGSHEET</b>	<b>27</b>
<b>BIODATA TIM</b>	<b>28</b>

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Indonesia merupakan salah satu penghasil kopi terbesar di dunia. Menurut Badan Pusat Statistik, produksi kopi Indonesia akan mencapai 794,8 ribu ton pada tahun 2022. Lampung merupakan salah satu daerah di Indonesia yang menghasilkan kopi. Provinsi Lampung adalah salah satu provinsi yang memiliki komoditas hasil perkebunan paling besar di Indonesia yaitu perkebunan kopi. Luas areal perkebunan kopi Provinsi merupakan terbesar nomor 2 di Indonesia setelah Sumatera Selatan yang memiliki luas 161,4 ribu(Ha) dan penyumbang kontribusi terbesar nomor 2 terhadap produksi kopi Indonesia pada tahun 2017 sebesar 17,44% [7]. Kabupaten Lampung Barat menyumbang produksi kopi paling besar di Provinsi Lampung mencapai 51.284,6 ton per tahun [8].

Pada pengolahan biji kopi, proses penjemuran merupakan bagian yang penting dalam prosedurnya. Proses penjemuran dapat mempengaruhi kualitas dari biji kopi. Petani yang menggunakan metode tradisional, harus selalu memperhatikan panas matahari dan mengantisipasi terjadinya hujan saat penjemuran.

Lampung Barat memiliki suhu udara yang terbilang dingin dengan rata-rata suhu 21,68 °C dan memiliki kelembaban udara 90,11 % pada tahun 2016 [9]. Dengan suhu dan kelembaban udara seperti itu sangat cocok untuk tanaman kopi. Namun Lampung Barat juga terkenal dengan curah hujan yang terbilang tinggi bahkan pada Bulan Februari 2016 mencapai 446,00 mm<sup>3</sup> [9]. Hal ini akan menyulitkan terutama pada proses penjemuran atau pengeringan biji kopi, sehingga perlu adanya teknologi yang dapat mengatasi hal tersebut agar hasil dari produksi kopi dapat maksimal.

Maka dari permasalahan diatas, akan di rancang Sistem Penjemur Kopi Otomatis dengan Fuzzy Logic Mamdani. Sistem Penjemur Kopi Otomatis ini memiliki sensor yang dapat membantu penjemuran biji kopi.

Dalam sistem ini, akan menerapkan beberapa sensor dan juga aktuator, antara lain sensor BMP180 untuk membaca tekanan udara, sensor DHT11 untuk membaca suhu dan kelembaban udara, sensor LDR untuk membaca cerah atau mendungnya sinar matahari, sensor MD0127 untuk membaca air hujan yang turun baik dalam keadaan mendung ataupun cerah, aktuator Servo untuk menggerakkan kanopi, dan aktuator Mini ZVS sebagai pemanas bilik untuk mengeringkan kopi. Sistem ini akan ditenagai dengan Raspberry Pi 2 untuk mengolah data dengan Fuzzy Logic Mamdani yang nantinya output yang dihasilkan digunakan untuk menggerakkan aktuator dan Arduino Uno untuk mengatur sensor. Sistem ini juga akan menggunakan Firebase sebagai database real-time yang nantinya akan ditampilkan ke dalam sebuah aplikasi seperti website atau mobile.

### **1.2 Rumusan Masalah**

1. Bagaimana Sistem Penjemur Kopi Otomatis bisa menjadi solusi untuk para petani ketika menghadapi musim pancaroba ?
2. Apa kendala utama yang dihadapi petani kopi Lampung dalam menjalankan penjemuran biji kopi secara tradisional, khususnya terkait dengan keterbatasan mengantisipasi cuaca dan mengelola proses penjemuran?

3. Bagaimana implementasi Sistem Penjemur Kopi Otomatis dengan Fuzzy Logic Mamdani dapat meningkatkan efisiensi dan kualitas penjemuran biji kopi di Lampung serta membantu petani mengatasi tantangan yang ada?

### **1.3 Tujuan Perancangan**

1. Merancang sistem penjemur kopi otomatis untuk meningkatkan efisiensi penjemuran dengan penyesuaian otomatis terhadap kondisi cuaca.
2. Menciptakan sistem yang menjaga kualitas biji kopi pada tingkat optimal melalui penggunaan Fuzzy Logic Mamdani.
3. Sistem dirancang untuk membantu petani mengatasi tantangan cuaca, terutama curah hujan tinggi, dengan sensor real-time dan kontrol otomatis pada penutupan kanopi.

## **BAB II**

### **DASAR TEORI**

#### **2.1 Arduino IDE**

Arduino IDE (Integrated Development Environment) adalah lingkungan pengembangan perangkat lunak yang digunakan untuk memprogram mikrokontroler Arduino. Arduino adalah platform open-source yang digunakan untuk membuat berbagai proyek elektronik dan prototipe yang berbasis mikrokontroler. Arduino IDE menyediakan alat untuk menulis, mengedit, mengkompilasi, dan mengupload kode program ke mikrokontroler Arduino [1].

#### **2.2 Breadboard**

Breadboard adalah perangkat elektronik yang digunakan untuk merakit prototipe sirkuit elektronik tanpa menggunakan solder. Breadboard sering digunakan oleh para pengembang, mahasiswa, atau hobiis dalam proses pengujian dan eksperimen dengan komponen elektronik sebelum merakitnya menjadi sirkuit yang lebih permanen. Breadboard biasanya terdiri dari suatu papan datar berbentuk persegi atau persegi panjang dengan banyak lubang kecil yang tersusun dalam matriks atau grid. Setiap lubang pada breadboard biasanya terhubung dengan jalur logam di bawah permukaan, yang digunakan untuk menghubungkan komponen elektronik dan membuat sirkuit [2].

#### **2.3 Kabel Jumper**

Kabel jumper adalah kabel pendek yang digunakan dalam elektronika dan komputer untuk menghubungkan komponen elektronik atau sirkuit secara sementara atau dalam merakit prototipe sirkuit. Kabel jumper biasanya terbuat dari kawat tipis yang dilapisi dengan isolasi plastik dan memiliki konektor di kedua ujungnya untuk memudahkan pemasangan. Kabel jumper umumnya digunakan untuk menghubungkan pin atau kaki-kaki komponen elektronik, seperti mikrokontroler, sensor, modul, atau komponen lainnya yang memerlukan koneksi antara satu dengan yang lain [3]. Kabel jumper dapat digunakan untuk menghubungkan komponen pada breadboard, PCB (Printed Circuit Board), atau modul pengembangan seperti Arduino atau Raspberry Pi [3].

#### **2.4 Sensor Suhu dan Kelembaban (DHT11)**

Sensor DHT11 adalah sensor yang digunakan untuk mengukur suhu dan kelembaban udara dalam satu paket sensor. Sensor ini cukup populer digunakan dalam proyek elektronik dan Internet of Things (IoT) karena harga yang terjangkau dan kemudahan penggunaannya. Sensor DHT11 dihubungkan ke mikrokontroler, seperti Arduino, Raspberry Pi, atau platform lainnya, untuk membaca data suhu dan kelembaban yang dihasilkan oleh sensor tersebut. DHT11 cukup mudah digunakan dengan berbagai library atau kode sumber yang telah tersedia, sehingga membuatnya populer dalam proyek-proyek elektronik dan IoT yang memerlukan pemantauan suhu dan kelembaban. Penting memperhatikan batasan dan ketelitian dari sensor DHT11, serta mengikuti petunjuk penggunaan yang diberikan oleh produsen untuk memperoleh hasil pengukuran yang akurat. Sensor ini digunakan untuk mendeteksi tingkat kelembaban pada tempat sampah. Sensor ini akan mentrigger notifikasi ke cloud telegram jika kelembaban terlalu tinggi karena terdapat potensi untuk menyebabkan penyakit [2].

## 2.5 Motor Servo

Motor servo adalah jenis motor yang digunakan untuk menggerakkan mekanisme yang memerlukan kontrol posisi atau sudut yang presisi. Motor servo biasanya digunakan dalam berbagai aplikasi seperti robotika, kendali otomatis, model pesawat terbang, kendaraan remote control, dan lain sebagainya. Motor servo dapat digunakan dalam berbagai proyek elektronik atau mekanik untuk mengontrol gerakan presisi, seperti mengendalikan posisi sudut dari suatu perangkat, memutar objek dalam sudut tertentu, atau mengontrol pergerakan robot. Penting untuk memahami karakteristik dan spesifikasi motor servo yang digunakan dalam proyek Anda, serta mengikuti petunjuk penggunaan yang diberikan oleh produsen untuk memastikan penggunaan yang aman dan optimal [5].

## 2.6 Sensor BMP180

Sensor BMP180 adalah sensor tekanan atmosfer yang dapat mengukur tekanan, suhu, dan ketinggian udara di sekitar modul. Sensor ini menggunakan antarmuka I2C untuk berkomunikasi dengan Arduino atau mikrokontroler lainnya. Sensor ini memiliki rentang pengukuran tekanan dari 300 hPa sampai 1100 hPa dengan akurasi hingga 0.02 hPa

## 2.7 Sensor MD0127

Raindrop sensor adalah sebuah alat yang dapat mendeteksi hujan atau adanya cuaca hujan yang berada di sekitarnya, sensor ini dapat digunakan sebagai switch, saat adanya tetesan air hujan yang jatuh melewati raining board yang terdapat pada sensor, selain itu raindrop sensor dapat juga digunakan untuk mengukur intensitas curah hujan [6].

## 2.8 Mini ZVS

Aktuator Mini ZVS adalah papan pemanas induksi dengan teknologi Zero Voltage Switching (ZVS) yang berukuran kecil. Papan ini biasanya digunakan dalam aplikasi DIY (Do It Yourself) atau proyek-proyek pengembangan elektronik. Teknologi ZVS memastikan saklar diaktifkan atau dinonaktifkan pada saat tegangan nol untuk meningkatkan efisiensi dan mengurangi rugi daya.

## 2.9 Raspberry pi

Raspberry pi adalah sebuah SBC (single-board computer) seukuran kartu kredit. raspberry pi telah dilengkapi dengan semua fungsi layaknya sebuah komputer lengkap, menggunakan SOC(System-on-a-Chip) ARM yang dikemas dan diintegrasikan di atas PCB(papan sirkuit). Raspberry Pi ini mampu bekerja layaknya komputer pada umumnya dengan kemampuan untuk menjalankan sistem operasi Linux dan aplikasinya seperti LibreOffice, multimedia (audio dan video), peramban web, ataupun programming [10]

Raspberry Pi dapat berfungsi sebagai komputer mandiri namun tidak dapat menggantikan komputer tradisional karena memiliki keterbatasan. Raspberry Pi memiliki beberapa pengaplikasian diantaranya dapat digunakan sebagai perangkat sistem *smart home*, Asisten AI, *Live Bot*, Kamera keamanan dengan *motion capture*, dan *Zero-Powered Smartphone*. [11]

## BAB III

### METODE

#### 3.1 Desain Perangkat (Hardware & Software)

Dalam mengimplementasikan sistem Tempat Sampah Pintar (Smartbin), terdapat desain perangkat yang dibagi menjadi *Hardware* dan *Software* dalam perancangan sistem. *Hardware* yang digunakan sebagai berikut:

Tabel 1.1 *Hardware* yang digunakan dalam rangkaian

No	Nama	Jenis	Jumlah
1	BMP180	Sensor	1
2	DHT11	Sensor	1
3	MD0127	Sensor	1
4	Servo	Aktuator	2
5	Mini ZVS Induction Heater	Aktuator	1
6	Raspberry Pi 2	Microcontroller	1
7	Arduino Uno Rev 3	Microcontroller	1
8	Large Breadboard	-	2
9	Kabel Jumper	-	±20

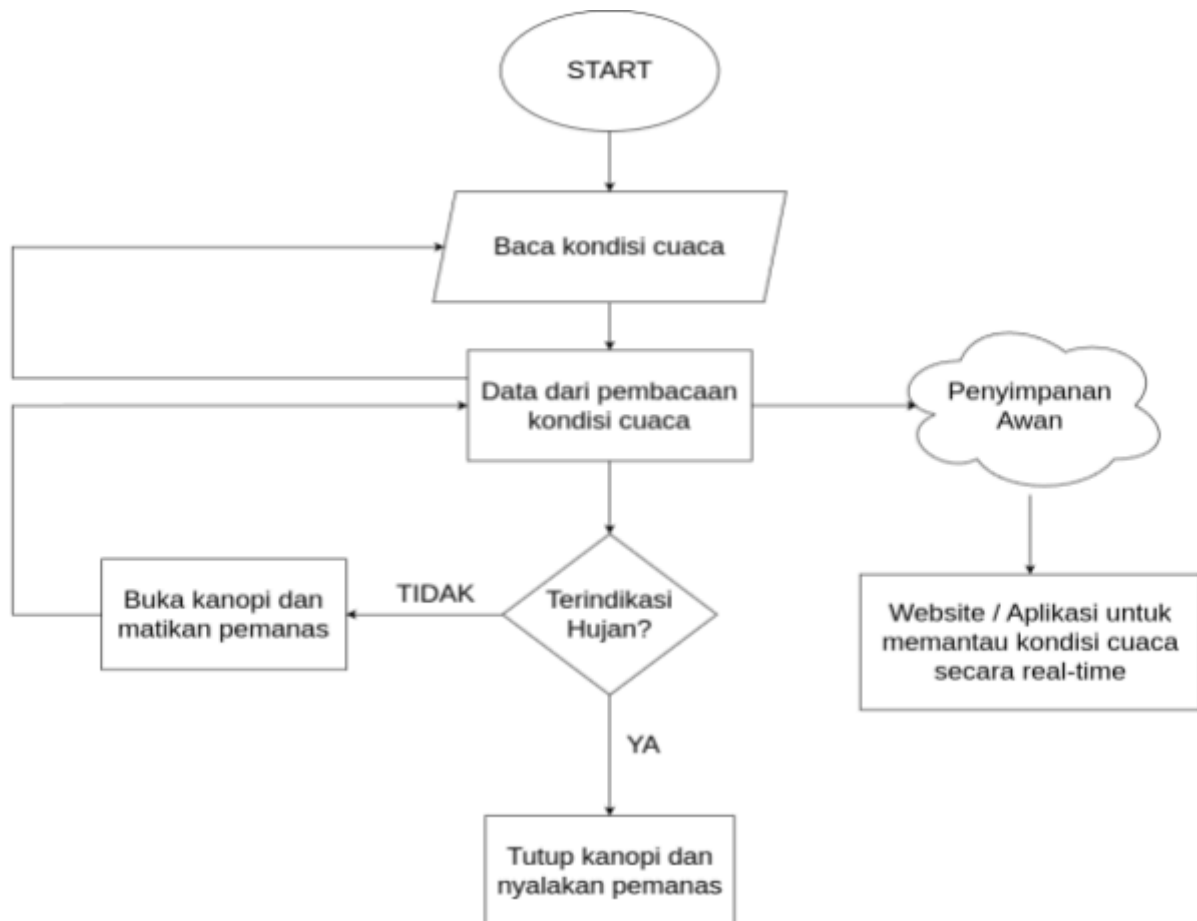
Adapun *software* yang digunakan sebagai berikut:

Tabel 1.2 *Software* yang digunakan dalam rangkaian

No	Nama
1	Geany

2	Arduino IDE
3	Firebase

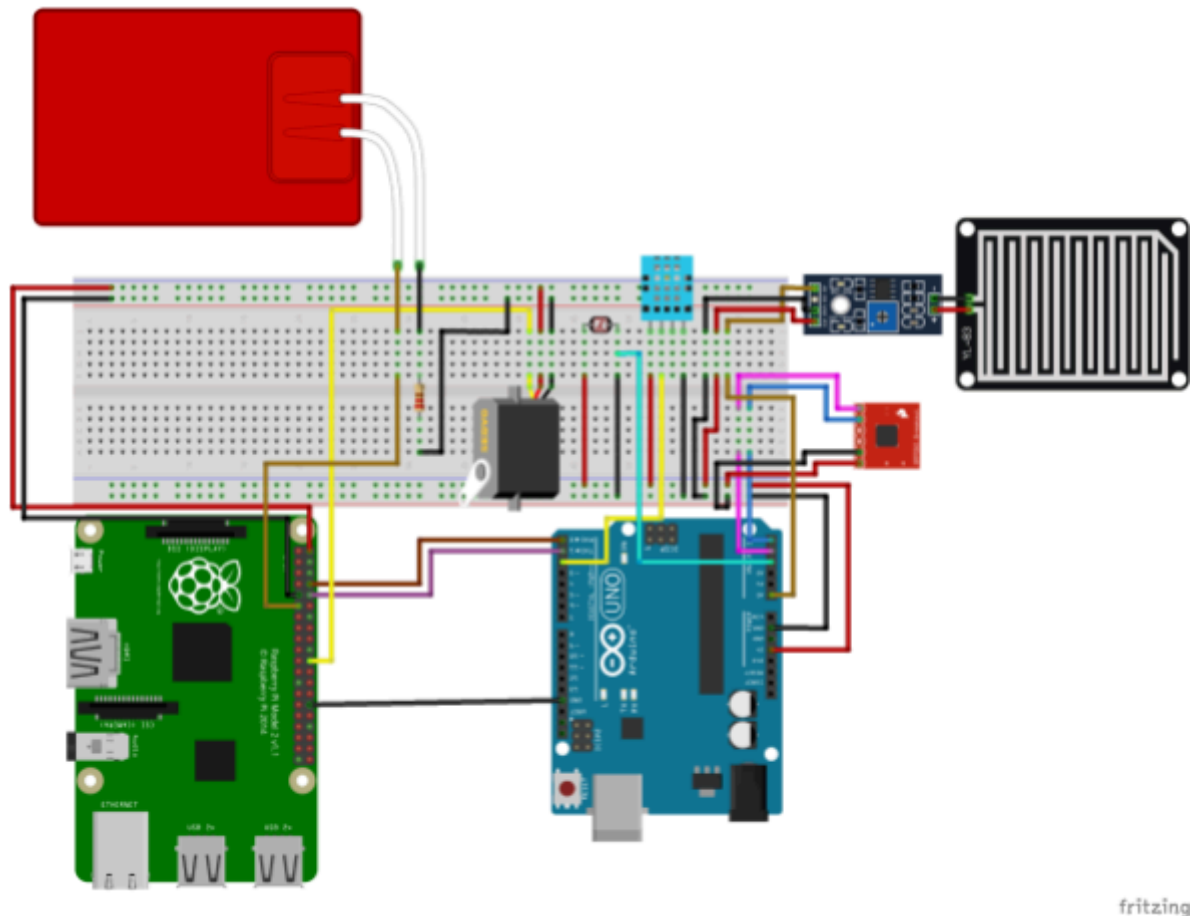
### 3.1.1 Flowchart



Gambar 1. Flowchart



### 3.1.2 Gambaran Umum Sistem



Gambar 2 Gambaran Umum Sistem

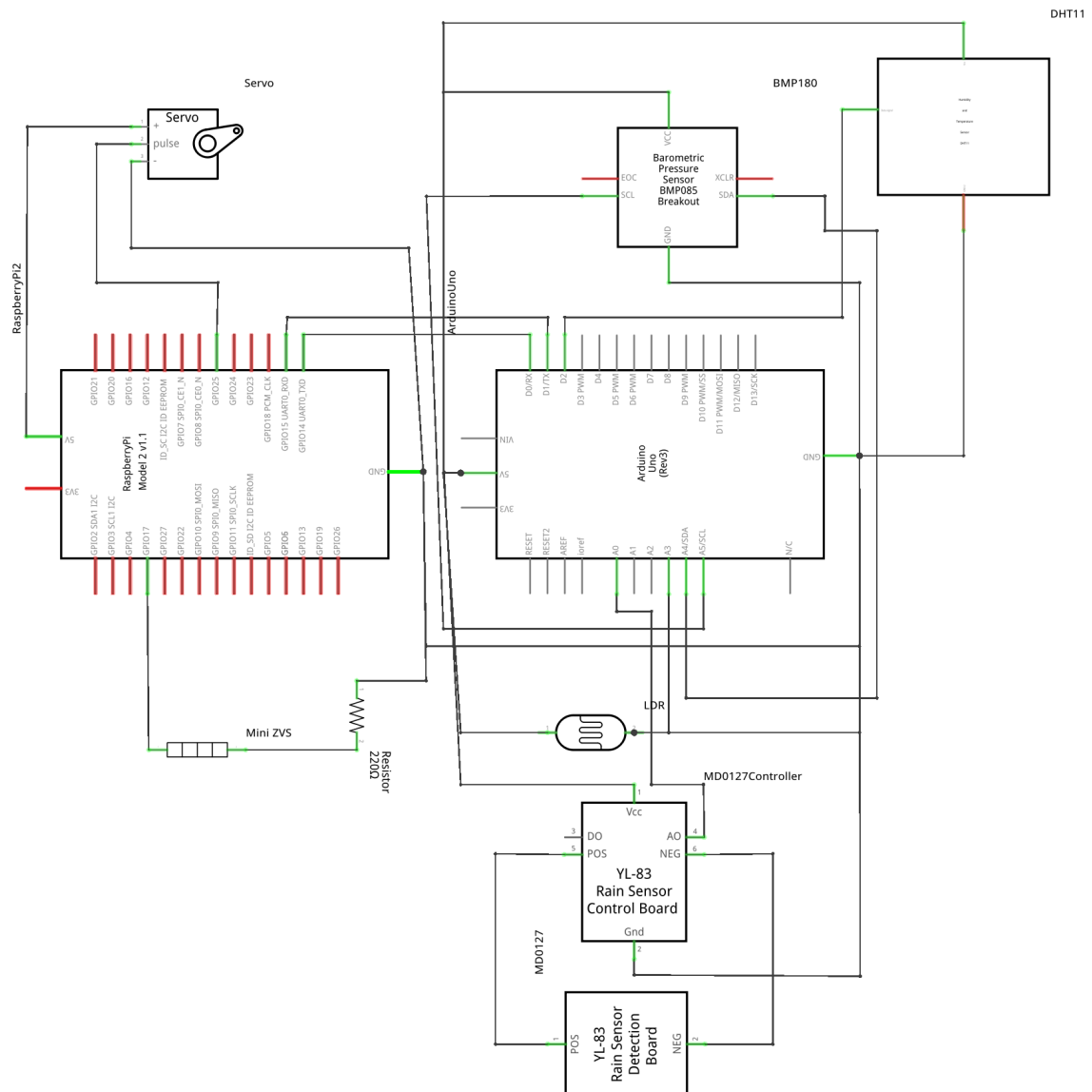
Dalam sistem ini, akan menerapkan beberapa sensor dan juga aktuator, antara lain sensor BMP180 untuk membaca tekanan udara, sensor DHT11 untuk membaca suhu dan kelembaban udara, sensor LDR untuk membaca cerah atau mendungnya sinar matahari, sensor MD0127 untuk membaca air hujan yang turun baik dalam keadaan mendung ataupun cerah, aktuator Servo untuk menggerakkan kanopi, dan aktuator Mini ZVS sebagai pemanas bilik untuk mengeringkan kopi. Sistem ini akan ditenagai dengan Raspberry Pi 2 untuk mengolah data dengan Fuzzy Logic Mamdani yang nantinya output yang dihasilkan digunakan untuk menggerakkan aktuator dan Arduino Uno untuk mengatur sensor. Sistem ini juga akan menggunakan Firebase sebagai database real-time yang nantinya akan ditampilkan ke dalam sebuah aplikasi seperti website atau mobile.

Sistem penjemur kopi otomatis memiliki dua microcontroller, yaitu Raspberry Pi 2 dan Arduino Uno. Dua microcontroller tersebut dibedakan berdasarkan penugasan. Arduino Uno bertanggung jawab dalam membaca sensor - sensor yang terpasang di sistemnya sendiri lalu mengirim data hasil pembacaan ke Raspberry Pi 2, sedangkan Raspberry Pi 2 bertanggungjawab dalam mengolah data hasil kiriman dari Arduino Uno untuk pengambilan keputusan untuk tindakan terhadap aktuator yang terpasang di sistemnya sendiri dan juga mengirim data ke penyimpanan awan, yaitu Firebase.

Pada tahap awal, Arduino Uno akan membaca kondisi cuaca di sekitar wadah untuk menjemur kopi melalui sensor tekanan udara (BMP180), sensor suhu dan kelembaban udara (DHT11), sensor

deteksi kecerahan di ruangan terbuka (LDR), dan sensor deteksi rintik hujan (MD0127). Data tersebut akan dikirimkan ke Raspberry Pi 2. Setelah proses pengiriman data, Raspberry Pi 2 mengirimkan data terlebih dahulu ke penyimpanan awan Firebase yang nantinya akan dipergunakan untuk website atau aplikasi. Lalu, data hasil dari pembacaan empat sensor, diproses melalui Fuzzy Logic Mamdani dikarenakan kompleksitas suatu kondisi cuaca. Di dalam Fuzzy Logic Mamdani, terdapat aturan - aturan khusus yang sudah ditetapkan. Ketika Raspberry Pi 2 mengindikasikan bahwa cuaca tersebut hujan, maka kanopi akan tertutup dan pemanas bilik akan dinyalakan. Ketika Raspberry Pi 2 mengindikasikan bahwa cuaca tersebut cerah, maka kanopi akan terbuka dan pemanas bilik akan dimatikan.

### 3.1.3 Schematic Diagram



fritzing

Gambar 3 Schematic Diagram

### 3.2 Skema Pengujian

#### 3.2.1 Skema Pengujian perangkat keras

Nomor Pengujian	Kelas Uji	Butir Uji	Indikator keberhasilan
P-01	Sensor suhu dan kelembaban	Mendeteksi suhu untuk mengetahui keadaan suhu sekitar	Sensor dapat mengirimkan 1 data suhu setiap 1 detik selama 10 detik
		Mendeteksi kelembaban udara untuk mengetahui keadaan kelembaban udara sekitar	Sensor dapat mengirimkan 1 data suhu setiap 1 detik selama 10 detik
P-02	Sensor tekanan udara	Mendeteksi tekanan udara untuk mengetahui kondisi tekanan udara sekitar	Sensor dapat mengirimkan 1 data tekanan udara setiap 1 detik selama 10 detik.
		Menguji keakuratan sensor tekanan udara.	Sensor dapat menghasilkan nilai tekanan udara dengan $\leq$ % relative error
P-03	Atap tertutup/terbuka	Menguji tingkat akurasi atap	Atap akan menutup/membuka sesuai dengan kondisi cuaca saat pengujian, dengan tingkat akurasi $\geq$ 60 %

P-04	Pemanas wadah jemuran	Menghasilkan panas ketika atap tertutup	Ketika atap tertutup, maka aktuator akan dihidupkan untuk memanaskan wadah tempat penjemuran kopi
------	-----------------------	---	---

### 3.2.2 Skema Pengujian Fuzzy Logic

Nomor Pengujian	Kelas Uji	Butir Uji	Indikator keberhasilan
P-05	Fungsionalitas Fuzzy	Mengolah data dari sensor tekanan udara, sensor suhu dan kelembaban dengan algoritma fuzzy logic.	Dapat menampilkan data fuzzy sesuai dengan referensi pada output Arduino IDE. Membuka/menutup atap penjemur biji kopi otomatis dengan data hasil fuzzy logic.
		Membuka/menutup atap penjemur biji kopi otomatis dengan data hasil fuzzy logic.	Dapat membuka/menutup atap sesuai dengan hasil output fuzzy. Dapat membuka/menutup atap sesuai dengan hasil output fuzzy.

## BAB IV

# HASIL DAN PEMBAHASAN

### 4.1 Kode program

#### Kode Arduino

```
1.  #include <DHT.h>
2.  #include <SPI.h>
3.  #include <Adafruit_BMP280.h>
4.
5.  // Definisi pin untuk sensor DHT
6.  #define DHTPIN 2
7.  #define DHTTYPE DHT11
8.
9.  // Definisi pin untuk sensor BMP280
10. #define BMP_SCK (13)
11. #define BMP_MISO (12)
12. #define BMP_MOSI (11)
13. #define BMP_CS (10)
14.
15. // Definisi pin untuk sensor hujan
16. #define RAINPIN A2
17.
18. // Inisialisasi objek sensor DHT dan BMP280
19. DHT dht(DHTPIN, DHTTYPE);
20. Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO, BMP_SCK);
21.
22. void setup() {
23.     // Memulai sensor DHT
24.     dht.begin();
25.
```

```
26. // Memulai sensor BMP280

27. bmp.begin();

28.

29. // Mengatur pin untuk sensor hujan sebagai input

30. pinMode(RAINPIN, INPUT);

31.

32. // Mengatur parameter sampling dan standby untuk sensor BMP280

33. bmp.setSampling(Adafruit_BMP280::MODE_NORMAL, Adafruit_BMP280::SAMPLING_X2, Adafruit_BMP280::SAMPLING_X16,
Adafruit_BMP280::FILTER_X16, Adafruit_BMP280::STANDBY_MS_500);

34.

35. // Memulai komunikasi serial dengan baud rate 9600

36. Serial.begin(9600);

37. }

38.

39. void loop() {

40. // Mengecek apakah ada data yang tersedia di Serial

41. if (Serial.available() > 0) {

42. // Membaca satu baris data dari Serial

43. String line = Serial.readStringUntil('\n');

44.

45. // Memeriksa apakah baris data adalah "OK"

46. if (line == "OK") {

47. // Membaca kelembaban dan suhu dari sensor DHT

48. float h = dht.readHumidity();

49. float t = bmp.readTemperature();

50. float T = dht.readTemperature();

51.

52. // Membaca tekanan atmosfer dari sensor BMP280

53. float p = bmp.readPressure();

54.

55. // Membaca nilai basah (wet) dari sensor hujan

56. float w = 1023 - analogRead(RAINPIN);
```

```

57.
58.      // Menampilkan data kelembaban, suhu, tekanan, dan nilai basah melalui Serial
59.      Serial.print("hum:");
60.      Serial.print(h);
61.      Serial.print(",tmpBMP:");
62.      Serial.print(t);
63.      Serial.print(",tmpDHT:");
64.      Serial.print(T);
65.      Serial.print(",pre:");
66.      Serial.print(p);
67.      Serial.print(",wet:");
68.      Serial.print(w);
69.      Serial.print("\n");
70.
71.      // Memberi jeda 500 milidetik
72.      delay(500);
73.  }
74.  }
75.  }
76.

```

## Kode Raspberry pi

```

# Import library yang diperlukan
from serial import Serial
from time import sleep
from os import system
import pyrebase
import json
import numpy as np
import skfuzzy as fuzz
from skfuzzy import control as ctrl
from gpiozero import Servo

```

```

import RPi.GPIO as GPIO

# Definisi pin GPIO pada Raspberry Pi
PINSERVO = 12
PINZVS = 24

# Konfigurasi Firebase dari file JSON
config = json.load(open("./firebaseConfig.json"))
firebase = pyrebase.initialize_app(config)
auth = firebase.auth()
user = auth.sign_in_with_email_and_password("this_is_email@mail.com", "this_is_password")
db = firebase.database()

# Kelas untuk pemrosesan fuzzy logic
class AIProcessing :
    def __init__(self) :
        # Mendefinisikan variabel fuzzy logic
        self.__hum = ctrl.Antecedent(np.arange(20, 91, 1), 'hum')
        self.__tmp = ctrl.Antecedent(np.arange(0, 51, 1), 'tmp')
        self.__pre = ctrl.Antecedent(np.arange(900, 1101, 1), 'pre')
        self.__wet = ctrl.Antecedent(np.arange(0, 101, 1), 'wet')
        self.__out = ctrl.Consequent(np.arange(0, 101, 1), 'out')

        # Menginisialisasi fungsi keanggotaan
        self.__membership()

        # Menetapkan aturan fuzzy
        self.__setRules()

    def __membership(self) :
        # Menentukan fungsi keanggotaan untuk setiap variabel
        self.__hum["low"] = fuzz.trimf(self.__hum.universe, [20, 20, 65])
        self.__hum["medium"] = fuzz.trimf(self.__hum.universe, [20, 60, 80])
        self.__hum["high"] = fuzz.trimf(self.__hum.universe, [75, 90, 90])

        self.__tmp["low"] = fuzz.trimf(self.__tmp.universe, [0, 0, 22])

```



```

self.__tmp["medium"] = fuzz.trimf(self.__tmp.universe, [0, 21, 27])
self.__tmp["high"] = fuzz.trimf(self.__tmp.universe, [26, 50, 50])

self.__pre["low"] = fuzz.trimf(self.__pre.universe, [900, 900, 938])
self.__pre["medium"] = fuzz.trimf(self.__pre.universe, [900, 935, 1011])
self.__pre["high"] = fuzz.trimf(self.__pre.universe, [1003, 1100, 1100])

self.__wet["low"] = fuzz.trimf(self.__wet.universe, [0, 0, 50])
self.__wet["high"] = fuzz.trimf(self.__wet.universe, [45, 100, 100])

self.__out["low"] = fuzz.trimf(self.__out.universe, [0, 0, 50])
self.__out["high"] = fuzz.trimf(self.__out.universe, [45, 100, 100])

def __setRules(self) :
    # Menetapkan aturan fuzzy
    rules = [
        ctrl.Rule(self.__hum["low"] & self.__pre["low"] & self.__tmp["low"] & self.__wet["low"],
self.__out["low"]),
        ctrl.Rule(self.__hum["low"] & self.__pre["low"] & self.__tmp["low"] & self.__wet["high"],
self.__out["low"]),
        ctrl.Rule(self.__hum["low"] & self.__pre["medium"] & self.__tmp["low"] & self.__wet["low"],
self.__out["high"]),
        ctrl.Rule(self.__hum["low"] & self.__pre["medium"] & self.__tmp["low"] & self.__wet["high"],
self.__out["low"]),
        ctrl.Rule(self.__hum["low"] & self.__pre["high"] & self.__tmp["low"] & self.__wet["low"],
self.__out["high"]),
        ctrl.Rule(self.__hum["low"] & self.__pre["high"] & self.__tmp["low"] & self.__wet["high"],
self.__out["low"]),
        ctrl.Rule(self.__hum["medium"] & self.__pre["low"] & self.__tmp["low"] & self.__wet["low"],
self.__out["low"]),
        ctrl.Rule(self.__hum["medium"] & self.__pre["low"] & self.__tmp["low"] & self.__wet["high"],
self.__out["low"]),
        ctrl.Rule(self.__hum["medium"] & self.__pre["medium"] & self.__tmp["low"] &
self.__wet["low"], self.__out["high"]),
        ctrl.Rule(self.__hum["medium"] & self.__pre["medium"] & self.__tmp["low"] &
self.__wet["high"], self.__out["low"]),
        ctrl.Rule(self.__hum["medium"] & self.__pre["high"] & self.__tmp["low"] & self.__wet["low"],
self.__out["high"]),

```

[illegible]

[illegible]

```

        ctrl.Rule(self.__hum["high"] & self.__pre["high"] & self.__tmp["high"] & self.__wet["low"],
self.__out["high"]),

        ctrl.Rule(self.__hum["high"] & self.__pre["high"] & self.__tmp["high"] & self.__wet["high"],
self.__out["low"])

    ]

    boxCtrl = ctrl.ControlSystem(rules)
    self.__boxSim = ctrl.ControlSystemSimulation(boxCtrl)

def AICompute(self, dataDICT) :
    # Menghitung output fuzzy berdasarkan input
    self.__boxSim.input['hum'] = dataDICT["hum"]
    self.__boxSim.input['tmp'] = dataDICT["tmp"]
    self.__boxSim.input['pre'] = dataDICT["pre"]
    self.__boxSim.input['wet'] = dataDICT["wet"]

    self.__boxSim.compute()

    return round(self.__boxSim.output['out'], 2)

# Program utama
if __name__ == "__main__" :
    # Membuat objek AIProcessing
    callAI = AIProcessing()

    # Menginisialisasi servo dan MiniZVS
    servo = Servo(PINSERVO)
    GPIO.setmode(GPIO.BCM)
    GPIO.setup(PINZVS, GPIO.OUT)

    # Membuka koneksi serial dengan Arduino
    serial = Serial('/dev/ttyACM0', 9600, timeout=1)
    serial.reset_input_buffer()

    try :
        while True :
            # Membersihkan layar terminal

```

```

system("clear")

# Mengirim perintah "OK" ke Arduino
serial.write(b"OK\n")

# Membaca data dari Arduino
if serial.in_waiting > 0:
    line = serial.readline().decode('utf-8').rstrip()
    dataRAW = line.split(",")
    dataFINAL = {}
    dataFINAL["tmp"] = 0

# Memproses dan menyimpan data sensor
for i in range(len(dataRAW)) :
    temp = dataRAW[i].split(":")
    if temp[0] == "tmpBMP" or temp[0] == "tmpDHT" :
        dataFINAL["tmp"] += float(temp[1])
    elif temp[0] == "pre" :
        dataFINAL[temp[0]] = round(float(temp[1]) / 100, 2)
    elif temp[0] == "wet" :
        dataFINAL[temp[0]] = round((float(temp[1]) / 1023) * 100.00, 2)
    else :
        dataFINAL[temp[0]] = round(float(temp[1]), 2)
dataFINAL["tmp"] = round(float(dataFINAL["tmp"]) / 2, 2)

# Menghitung output fuzzy menggunakan metode AI
dataFINAL["fuzz"] = callAI.AICompute(dataFINAL)

# Kontrol servo dan GPIO berdasarkan output fuzzy
if dataFINAL["fuzz"] < 50.00 :
    servo.min()
    GPIO.output(PINZVS, 1)
else :
    servo.max()
    GPIO.output(PINZVS, 0)

```

```

# Mengirim data ke database Firebase
db.child(user["localId"]).set(dataFINAL)

# Menampilkan data di terminal
print(dataRAW)
print("")
print(dataFINAL)

# Menunggu selama 0.5 detik
sleep(0.5)
except Exception as error:
    # Menutup koneksi dan menampilkan pesan kesalahan
    servo.close()
    GPIO.cleanup()
    serial.close()
    print(error)

```

## 4.2 Hasil Pengujian

### 4.2.1 Skema Pengujian perangkat keras

Nomor Pengujian	Kelas Uji	Butir Uji	Hasil	Kendala
P-01	Sensor suhu dan kelembaban	Mendeteksi suhu untuk mengetahui keadaan suhu sekitar	Berhasil	-
		Mendeteksi kelembaban udara untuk mengetahui keadaan kelembaban udara sekitar	Berhasil	-

P-02	Sensor tekanan udara	Mendeteksi tekanan udara untuk mengetahui kondisi tekanan udara sekitar	Berhasil	-
		Menguji keakuratan sensor tekanan udara.	Berhasil	-
P-03	Atap tertutup/terbuka	Menguji tingkat akurasi atap	Berhasil	-
P-04	Pemanas wadah jemuran	Menghasilkan panas ketika atap tertutup	Berhasil	besar sumber dayanya (arus) tidak mencukupi sesuai cakupan arus yang dibutuhkan, sehingga sedikit lama untuk mengeluarkan panas untuk wadah

#### 4.2.2 Skema Pengujian Fuzzy Logic

Nomor Pengujian	Kelas Uji	Butir Uji	Hasil	Kendala
-----------------	-----------	-----------	-------	---------

P-05	Fungsionalitas Fuzzy	Mengolah data dari sensor tekanan udara, sensor suhu dan kelembaban dengan algoritma fuzzy logic.	Berhasil	-
		Membuka/menutup atap penjemur biji kopi otomatis dengan data hasil fuzzy logic.	Berhasil	-



## **BAB V**

### **KESIMPULAN**

#### **5.1 Kesimpulan**

Berdasarkan latar belakang, permasalahan, dan tujuan perancangan yang telah diuraikan serta perancangan sistem, dapat disimpulkan bahwa Sistem Penjemur Kopi Otomatis dengan Fuzzy Logic Mamdani merupakan solusi inovatif untuk meningkatkan efisiensi dan kualitas penjemuran biji kopi di Lampung, Indonesia. Produksi kopi yang tinggi di Lampung, khususnya di Kabupaten Lampung Barat, memberikan peluang untuk mengoptimalkan proses penjemuran agar hasil yang dihasilkan lebih baik.

Kendala yang dihadapi petani kopi di Lampung, khususnya terkait keterbatasan prakiraan cuaca dan pengelolaan proses pengeringan tradisional, menjadi motivasi utama pengembangan sistem otomatis yang mampu beradaptasi dengan perubahan kondisi cuaca, terutama saat musim pancaroba.

Implementasi sensor BMP180, DHT11, LDR, dan MD0127, bersama dengan aktuator servo dan mini ZVS untuk membuat sistem penjemur kopi otomatis ini memberikan solusi untuk memecahkan masalah yang dihadapi para petani. Dengan Fuzzy logic sebagai algoritma untuk mengendalikan aktuator dan menghasilkan keputusan berdasarkan data yang diperoleh dari sensor seperti buka tutup kanopi dan proses penjemuran kopi yang dilakukan secara otomatis berdasarkan cuaca yang dideteksi oleh sensor.

Dengan demikian, Sistem Penjemur Kopi Otomatis ini dapat dianggap sebagai langkah maju dalam teknologi pertanian, memberikan solusi cerdas untuk meningkatkan hasil dan kualitas produksi kopi di Lampung, sekaligus meningkatkan kesejahteraan petani melalui peningkatan efisiensi dalam proses pertanian mereka.

#### **5.2 Saran**

Berikut merupakan saran dalam pengembangan kerja praktek selanjutnya, yaitu:

1. Pada penelitian yang akan datang, sistem bisa dilakukan uji coba secara langsung dengan para petani guna mengetahui dampak atau masalah yang terjadi di lapangan.
2. Sistem yang dirancang bisa menggunakan sensor atau aktuator yang lebih canggih atau lebih baik dari penelitian ini
3. Mengembangkan aplikasi antarmuka pengguna untuk memudahkan petani dalam mengoperasikan dan memantau sistem dari jauh.

## DAFTAR PUSTAKA

- [1] Muhammad J.(2021). Perancangan Purwarupa Atap Penjemur Kopi Otomatis dengan Arduino Mega menggunakan Fuzzy Logic Mamdani (Studi Kasus : Lampung Barat). Institut Teknologi Sumatera.
- [2] Djuandi F.(2011). Pengenalan Arduino.www. tobuku.com
- [3] Asyiva Nurbaeti dkk(2021) Rancang Bangun alat Pengering Biji Kopi berbasis Internet of Things, Jurnal Jaringan Telekomunikasi, volume 11, hal. 74-80.
- [4] Abdul Latief A(2019). Rancang Bangun Smart Coffee Roasters Berbasis Mikrokontroler, Jurnal IT Media Informasi IT STMIK Handayani. volume 10 no 1
- [5] Friyogi T, dkk (2020) Percancangan, Implementasi Monitoring dan Kontrol Alat pemanggang kopi, volume 12 no 2.
- [6] Katyal, A., Yadav, R., & Pandey, M.. Wireless Arduino Based Weather Station. International Journal of Advanced Research in Computer and Communication Engineering. Retrieved November 12, 2023, from <https://www.ijamtes.org/gallery/29.%20mar%20ijamtes%20-%20317.pdf>
- [7] E. D. Martauli, “Analysis of Coffee Production in Indonesia Analisis Produksi Kopi Di Indonesia,” vol. 01, no. 02, p. 2, 2018.
- [8] BPS, Ed., “Statistik Kopi Indonesia,” 2017.
- [9] L. Barat and D. Angka, “Kabupaten Lampung Barat Dalam Angka 2018,” 2018.
- [10] I. Dharma Wijaya, U. Nurhasan, and M. Agung Barata, “IMPLEMENTASI RASPBERRY PI UNTUK RANCANG BANGUN SISTEM KEAMANAN PINTU RUANG SERVER DENGAN PENGENALAN WAJAH MENGGUNAKAN METODE TRIANGLE FACE,” malang, Nov. 2017.
- [11] H. D. Ghae, Dr. L. Solanki, G. Sahu “A Review Paper on Raspberry Pi and its Applications”, International Journal of Advances in Engineering and Management (IJAEM), vol. 2, no. 12, 225-227, 2021
- [12] Aulia Ullah, Oktaf B.K., Imam Santoso, “Fuzzy Logic Implementation to Control Temperature and Humidity in a Bread Proofing Machine”, volume 1 no.2 , hal.66-74, September 2018.
- [13] Muhammad Yusvin M, Rama Okta W., “Implementasi Sistem Monitoring Deteksi Hujan dan Suhu Berbasis Sensor secara Real Time”, volume 20, no.1, 2017.
- [14] Ridho T.S., Kusnadi , Tito Sudiarto. “PROTOTYPE SISTEM KEMANAN BUKA TUTUP ATAP JEMURAN OTOMATIS MENGGUNAKAN SENSOR AIR DAN LIGHT DEPENDENT RESISTOR (LDR) BERBASIS ARDUINO”. volume 8, no.2, hal.161-172, Nov 2018.
- [15] Refi Amaliani, Abdul J.L., Imran Lubis (2021). “Rancang Bangun Miniatur Stasiun Cuaca Untuk Monitoring Curah Hujan, Suhu dan Kelembaban Udara Area lokal Menggunakan berbasis IoT

## LOGSHEET

No	Tanggal	Kegiatan Kehadiran Lampiran
1	2 Des 2023	Pembelian alat
2	9 Des 2023	Progress pembuatan alat
3	11 Des 2023	Progress pembuatan alat
4	14 Des 2023	Progress pembuatan alat dan laporan akhir
5	15 Des 2023	Progress laporan akhir

## BIODATA TIM

<b>Nama</b>	<b>NIM</b>	<b>Prodi</b>
Syabana Minggu Noviantosa	118140122	Teknik informatika
Solafide Zamili	121140201	Teknik Informatika
Marchel Ferry Timoteus Samosir	121140195	Teknik Informatika
Hamas Azhar	119140138	Teknik Informatika