

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Dalam kehidupan sehari-hari, suhu dan kelembapan udara merupakan dua faktor penting yang mempengaruhi kenyamanan manusia. Ketika suhu meningkat dan udara menjadi lembap, ruangan akan terasa pengap dan membuat aktivitas menjadi terganggu. Sebaliknya, ketika suhu terlalu rendah dengan kelembapan yang tidak terkontrol dapat menimbulkan rasa tidak nyaman bahkan berdampak pada kesehatan. Oleh karena itu, pengaturan suhu dan kelembapan dapat dikatakan sebagai aspek penting untuk menciptakan lingkungan yang nyaman.

Salah satu solusi yang sering digunakan untuk menjaga kenyamanan ruangan adalah pendinginan ruangan menggunakan alat, salah satu contohnya adalah kipas angin. Namun, dalam pengoperasiannya masih banyak yang mengatur kecepatan kipas tersebut secara manual. Hal inilah yang memerlukan intervensi terus-menerus dari pengguna, sehingga kurang efisien dan sering kali tidak responsif terhadap perubahan kondisi lingkungan. Hal ini mendorong perlunya sebuah sistem otomatis yang dapat menyesuaikan kecepatan kipas secara real-time berdasarkan suatu hal yang dapat dijadikan parameter, diantaranya adalah suhu dan kelembapan.

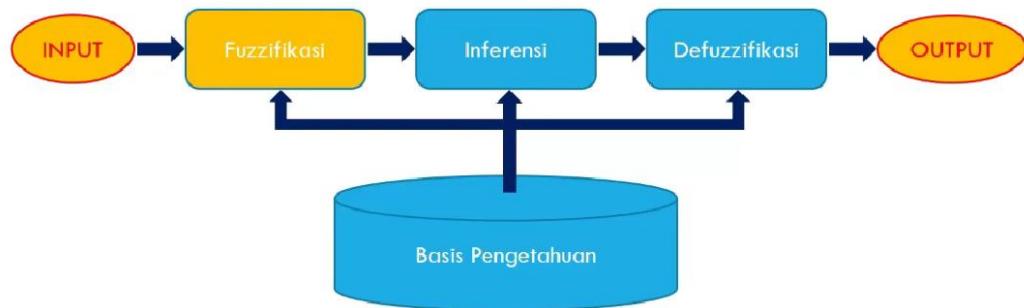
Di sinilah logika fuzzy menjadi solusi yang relevan untuk sistem otomatis yang diperlukan. Berbeda dengan sistem konvensional yang bekerja berdasarkan nilai pasti, logika fuzzy mampu menangani ketidakpastian dalam berbagai data. Misalnya, suhu yang tidak lagi hanya dinyatakan sebagai "19°C" tetapi bisa dikategorikan sebagai "sedikit hangat" atau kelembapan yang dianggap "cukup tinggi". Dengan kemampuan ini, logika fuzzy memungkinkan sistem untuk memberikan keputusan yang lebih fleksibel dan sesuai dengan kondisi sebenarnya.

Dalam laporan ini, diimplementasikannya sebuah sistem pengaturan kecepatan kipas otomatis berbasis logika fuzzy. Sistem ini menggunakan metode Mamdani, yang dikenal dengan kesederhanaannya dalam menerapkan aturan fuzzy dan kemampuannya menghasilkan nilai yang dapat diinterpretasikan dengan lebih akurat. Studi kasus yang diangkat adalah pengaturan kecepatan kipas berdasarkan input suhu (dalam skala derajat Celsius) dan kelembapan udara (dalam persentase), dengan hasil yang diperoleh berupa persentase kecepatan kipas dari rentang 0 hingga 100.

Melalui sistem ini, pengaturan kecepatan kipas seharusnya menjadi lebih responsif, efisien, dan juga mampu memberikan kenyamanan optimal sesuai dengan perubahan kondisi lingkungan terbaru. Selain itu, pendekatan ini juga menunjukkan bagaimana teknologi berbasis logika fuzzy dapat diterapkan secara nyata dalam kehidupan sehari-hari untuk meningkatkan kualitas hidup.

## 1.2 Metodologi

Dalam implementasinya, sistem pengaturan kecepatan kipas otomatis berbasis logika fuzzy ini mencakup beberapa tahapan penting, yaitu proses fuzzifikasi, proses inferensi, dan proses defuzzifikasi. Semua tahapan tersebut dilakukan untuk mengolah data *input* berupa suhu dan kelembapan menjadi *output* berupa kecepatan kipas yang sesuai dengan kondisi lingkungan terbaru. Gambar 1.1 merupakan visualisasi tahapan dari implementasi sistem logika fuzzy.



Gambar 1. 1 Diagram Alur Sistem Pengaturan Kecepatan Kipas Logika Fuzzy

\*Sumber : <https://youtu.be/6szqrV9u9k8>

### a) Fuzzifikasi

Pada tahap ini, nilai *input* numerik atau pasti (*crisp*) suhu dan kelembapan diubah menjadi derajat keanggotaan dalam kategori fuzzy. Suhu dibagi menjadi tiga kategori fuzzy, yaitu dingin, normal dan panas, sedangkan untuk kelembapan dikategorikan menjadi rendah, sedang dan tinggi. Fungsi keanggotaan yang digunakan pada studi kasus ini adalah fungsi segitiga dan trapesium, yang dirancang untuk merepresentasikan setiap kategori fuzzy secara linear.

### b) Inferensi

Tahap ini menggunakan metode inferensi fuzzy berbasis pengetahuan (aturan) atau *rule-based*. Aturan-aturan fuzzy dirancang berdasarkan hubungan yang logis antara data *input* yaitu suhu dan kelembapan serta *output* kecepatan kipas. Inferensi dilakukan menggunakan metode Mamdani dengan perhitungan menggunakan operator logika AND yang mengambil nilai *min* untuk menentukan nilai aturan. Output dari tahap ini berupa nilai fuzzy yang mewakili derajat keanggotaan dalam kategori kecepatan kipas, yaitu lambat, sedang dan cepat.

### c) Defuzzifikasi

Tahap terakhir merupakan defuzzifikasi, proses mengubah *output* fuzzy menjadi nilai asli atau *crisp*. Perhitungan defuzzifikasi yang digunakan untuk metode Mamdani biasanya adalah Centroid, yaitu nilai asli dihitung sebagai

titik pusat area agregasi dari semua fungsi keanggotaan yang berlaku. Hasil defuzzifikasi berupa persentase kecepatan kipas yang sesuai dengan nilai suhu dan kelembapan.

## BAB II

### METODE

#### 2.1 Sistem Inferensi Fuzzy

Sistem Inferensi Fuzzy merupakan sebuah mekanisme pengambilan keputusan yang menggunakan logika fuzzy untuk memetakan *input* dan *output* berdasarkan aturan linguistik. Sistem ini dirancang untuk menangani data yang bersifat tidak pasti atau ambigu, seperti suhu yang “hangat” atau kelembapan yang “cukup tinggi”.

Pada umumnya, sistem inferensi fuzzy terdiri dari tiga tahap utama, yaitu :

1. Fuzzifikasi, mengubah nilai *input* numerik menjadi derajat keanggotaan fuzzy
2. Inferensi, menerapkan aturan fuzzy untuk menghasilkan *output* fuzzy
3. Defuzzifikasi, mengubah *output* fuzzy menjadi nilai numerik atau *crisp*

Metode yang digunakan dalam implementasi ini adalah Sistem Inferensi Fuzzy Mamdani yang merupakan salah satu metode yang paling umum dan paling akurat. Metode ini bekerja menggunakan operator logika AND (*min*) dan OR (*max*) untuk menggabungkan aturan fuzzy.

#### 2.2 Metode Sistem Inferensi Fuzzy

Sistem ini diimplementasikan untuk mengolah dua *input* data yaitu suhu dan kelembapan menjadi *output* berupa kecepatan kipas dengan proses sebagai berikut.

##### 2.2.1 Fuzzifikasi

Fuzzifikasi adalah proses mengubah nilai *input* numerik menjadi nilai fuzzy berdasarkan fungsi keanggotaan. Dalam implementasi ini, fungsi keanggotaan yang digunakan adalah fungsi segitiga dan trapesium yang didefinisikan sebagai berikut

Fungsi Segitiga

$$\mu(x) = \begin{cases} 0, & \text{jika } x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}, & \text{jika } a \leq x \leq b \\ \frac{c-x}{c-b}, & \text{jika } b \leq x \leq c \end{cases}$$

Fungsi Trapesium

$$\mu(x) = \begin{cases} 0, & \text{jika } x \leq a \text{ atau } x \geq d \\ \frac{x-a}{b-a}, & \text{jika } a \leq x \leq b \\ 1, & \text{jika } b \leq x \leq c \\ \frac{d-x}{d-c}, & \text{jika } c \leq x \leq d \end{cases}$$

Untuk fungsi segitiga, terdapat 3 fungsi dan untuk fungsi trapezium terdapat 4 kondisi, masing masing penggunaan fungsi tersebut adalah sebagai berikut :

Fungsi Segitiga	Kondisi 1	$x$ berada di luar rentang fungsi segitiga dan nilai derajat keanggotaan $\mu(x)$ adalah 0 karena tidak termasuk dalam kategori fuzzy
	Kondisi 2	nilai $x$ berada disisi kiri segitiga, naik secara linear dari $a$ hingga puncak $b$ dengan rumus untuk menghitung nilai keanggotaan secara proporsional antara 0 hingga 1
	Kondisi 3	nilai $x$ berada disisi kanan segitiga, turun secara linear dari puncak $b$ hingga $c$ dengan rumus untuk menghitung nilai keanggotaan secara proporsional antara 1 hingga 0
Fungsi Trapesium	Kondisi 1	$x$ berada di luar rentang fungsi trapesium dan nilai derajat keanggotaan $\mu(x)$ adalah 0 karena tidak termasuk dalam kategori fuzzy
	Kondisi 2	nilai $x$ berada disisi kiri trapesium, naik secara linear dari $a$ hingga $b$ dengan rumus untuk menghitung nilai keanggotaan secara proporsional dari 0 hingga 1
	Kondisi 3	nilai $x$ berada dibagian tengah trapezium yang merupakan rentang stabil, nilai keanggotaan tetap maksimum $\mu(x) = 1$
	Kondisi 4	nilai $x$ berada disisi kanan trapesium, turun secara linear dari puncak $c$ hingga $d$ dengan rumus untuk menghitung nilai keanggotaan secara proporsional dari 1 hingga 0

### 2.2.2 Inferensi

Inferensi adalah proses penerapan aturan fuzzy berdasarkan logika linguistik. Dalam sistem ini menggunakan operator AND (*min*) untuk menentukan nilai aturan fuzzy. Berikut aturan fuzzy yang digunakan :

- Aturan 1 : Jika Suhu Dingin AND Kelembapan Rendah, Kecepatan Kipas → Lambat
- Aturan 2 : Jika Suhu Dingin AND Kelembapan Sedang, Kecepatan Kipas → Lambat

- Aturan 3 : Jika Suhu Dingin AND Kelembapan Tinggi, Kecepatan Kipas → Sedang
- Aturan 4 : Jika Suhu Normal AND Kelembapan Rendah, Kecepatan Kipas → Lambat
- Aturan 5 : Jika Suhu Normal AND Kelembapan Sedang, Kecepatan Kipas → Sedang
- Aturan 6 : Jika Suhu Normal AND Kelembapan Tinggi, Kecepatan Kipas → Cepat
- Aturan 7 : Jika Suhu Panas AND Kelembapan Rendah, Kecepatan Kipas → Sedang
- Aturan 8 : Jika Suhu Panas AND Kelembapan Sedang, Kecepatan Kipas → Cepat
- Aturan 9 : Jika Suhu Panas AND Kelembapan Tinggi, Kecepatan Kipas → Cepat

Formula perhitungan yang digunakan inferensi ini menggunakan operator AND adalah sebagai berikut :

$$\mu_{output} = \min(\mu_{suhu}, \mu_{kelembapan})$$

### 2.2.3 Defuzzifikasi dan Agregasi

Defuzzifikasi adalah proses mengubah *output* fuzzy menjadi nilai numerik (*crisp*). Metode perhitungan yang digunakan adalah Centroid atau Center of Gravity (COG) yang dihitung menggunakan formula berikut :

$$Crisp\ Output = \frac{\int \mu(x) \cdot x \, dx}{\int \mu(x) \, dx}$$

Di mana :

- $\mu(x)$  : Derajat keanggotaan hasil dari agregasi
- $x$  : Domain *output* (Kecepatan Kipas dengan rentang 0% – 100%)

Proses ini melibatkan agregasi *output* fuzzy dari semua aturan menjadi satu fungsi keanggotaan yang mencakup semua kategori linguistik (lambat, sedang, cepat).

## **BAB III**

### **HASIL DAN PEMBAHASAN**

#### **3.1 Source Code Program**

Berikut adalah source code dari implementasi Sistem Pengaturan Kecepatan Kipas Otomatis Berdasarkan Suhu dan Kelembapan menggunakan bahasa Python pada Google Collab.

```
import numpy as np
import matplotlib.pyplot as plt

def fuzzifikasi_suhu(suhu):
    dingin = max(0, min(1, (20 - suhu) / 20))
    normal = max(0, min((suhu - 10) / 10, (30 - suhu) / 10))
    panas = max(0, min((suhu - 25) / 15, 1))
    return {'Dingin': dingin, 'Normal': normal, 'Panas': panas}

def fuzzifikasi_kelembapan(kelembapan):
    rendah = max(0, min(1, (50 - kelembapan) / 50))
    sedang = max(0, min((kelembapan - 30) / 20, (70 - kelembapan) / 20))
    tinggi = max(0, min((kelembapan - 60) / 40, 1))
    return {'Rendah': rendah, 'Sedang': sedang, 'Tinggi': tinggi}

def inferensi(suhu_fuzzy, kelembapan_fuzzy):
    aturan = [
        (min(suhu_fuzzy['Dingin'],
kelembapan_fuzzy['Rendah']), "Jika Suhu Dingin dan Kelembapan Rendah --> Lambat"),
        (min(suhu_fuzzy['Dingin'],
kelembapan_fuzzy['Sedang']), "Jika Suhu Dingin dan Kelembapan Sedang --> Lambat"),
        (min(suhu_fuzzy['Dingin'],
kelembapan_fuzzy['Tinggi']), "Jika Suhu Dingin dan Kelembapan Tinggi --> Sedang"),
        (min(suhu_fuzzy['Normal'],
kelembapan_fuzzy['Rendah']), "Jika Suhu Normal dan Kelembapan Rendah --> Lambat"),
        (min(suhu_fuzzy['Normal'],
kelembapan_fuzzy['Sedang']), "Jika Suhu Normal dan Kelembapan Sedang --> Cepat"),
        (min(suhu_fuzzy['Normal'],
kelembapan_fuzzy['Tinggi']), "Jika Suhu Normal dan Kelembapan Tinggi --> Cepat"),
    ]
```

```

        (min(suhu_fuzzy['Panas'],
kelembapan_fuzzy['Rendah']), "Jika Suhu Panas dan Kelembapan
Rendah --> Sedang"),
        (min(suhu_fuzzy['Panas'],
kelembapan_fuzzy['Sedang']), "Jika Suhu Panas dan Kelembapan
Sedang --> Cepat"),
        (min(suhu_fuzzy['Panas'],
kelembapan_fuzzy['Tinggi']), "Jika Suhu Panas dan Kelembapan
Tinggi --> Cepat"),
    ]
hasil = {'Lambat': 0, 'Sedang': 0, 'Cepat': 0}
aturan_berlaku = []
for kekuatan_aturan, deskripsi in aturan:
    if kekuatan_aturan > 0:
        aturan_berlaku.append((deskripsi,
round(kekuatan_aturan, 2)))
    if "Lambat" in deskripsi:
        hasil['Lambat'] = max(hasil['Lambat'],
kekuatan_aturan)
    elif "Sedang" in deskripsi:
        hasil['Sedang'] = max(hasil['Sedang'],
kekuatan_aturan)
    elif "Cepat" in deskripsi:
        hasil['Cepat'] = max(hasil['Cepat'],
kekuatan_aturan)
return hasil, aturan_berlaku

def defuzzifikasi(hasil_fuzzy):
    x = np.linspace(0, 100, 1000)
    lambat = np.maximum(0, np.minimum(1, (50 - x) / 50))
    sedang = np.maximum(0, np.minimum((x - 30) / 20, (70 - x) /
20))
    cepat = np.maximum(0, np.minimum((x - 50) / 50, 1))

    agregasi = np.maximum(hasil_fuzzy['Lambat'] * lambat,
np.maximum(hasil_fuzzy['Sedang'] * sedang,
hasil_fuzzy['Cepat'] * cepat))

    output_crisp = np.sum(agregasi * x) / np.sum(agregasi)
    return round(output_crisp, 2), agregasi, x

# INPUT NILAI SUHU DAN KELEMBAPAN
suhu = float(input("Masukkan nilai Suhu : "))
kelembapan = float(input("Masukkan nilai persentase
Kelembapan : "))

```

```

# FUZZIFIKASI
suhu_fuzzy = fuzzifikasi_suhu(suhu)
kelembapan_fuzzy = fuzzifikasi_kelembapan(kelembapan)
print("\nFuzzifikasi Suhu :")
for key, value in suhu_fuzzy.items():
    print(f" {key} : {value}")
print("\nFuzzifikasi Kelembapan :")
for key, value in kelembapan_fuzzy.items():
    print(f" {key} : {value}")

# INFERENSI
hasil_fuzzy, aturan_berlaku = inferensi(suhu_fuzzy,
kelembapan_fuzzy)
print("\nHasil Inferensi Fuzzy :")
for key, value in hasil_fuzzy.items():
    print(f" {key} : {value}")
print("\nAturan yang Berlaku :")
for deskripsi, kekuatan in aturan_berlaku:
    print(f" {deskripsi}, dengan Hasil AND = {kekuatan}")

# DEFUZZIFIKASI
output_crisp, agregasi, x = defuzzifikasi(hasil_fuzzy)
print("\nKecepatan Kipas Secara Crisp : ", f"{output_crisp}%")

# HASIL KECEPATAN KIPAS SECARA LINGUISTIK
output_linguistik = max(hasil_fuzzy, key=hasil_fuzzy.get)
print("Kecepatan Kipas Secara Linguistik :",
output_linguistik)
print("\n")

# VISUALISASI DERAJAT KEANGGOTAAN SUHU
x_suhu = np.linspace(0, 40, 1000)
plt.figure(figsize=(8, 4))
plt.plot(x_suhu, [max(0, min(1, (20 - xi) / 20)) for xi in
x_suhu], label="Dingin")
plt.plot(x_suhu, [max(0, min((xi - 10) / 10, (30 - xi) / 10)) for
xi in x_suhu], label="Normal")
plt.plot(x_suhu, [max(0, min((xi - 25) / 15, 1)) for xi in
x_suhu], label="Panas")
plt.title("Derajat Keanggotaan Suhu")
plt.xlabel("Suhu (°C)")
plt.ylabel("Derajat Keanggotaan")
plt.legend()
plt.grid()
plt.show()
print("\n")

```

```

# VISUALISASI DERAJAT KEANGGOTAAN KELEMBAPAN
x_kelembapan = np.linspace(0, 100, 1000)
plt.figure(figsize=(8, 4))
plt.plot(x_kelembapan, [max(0, min(1, (50 - xi) / 50)) for xi in x_kelembapan], label="Rendah")
plt.plot(x_kelembapan, [max(0, min((xi - 30) / 20, (70 - xi) / 20)) for xi in x_kelembapan], label="Sedang")
plt.plot(x_kelembapan, [max(0, min((xi - 60) / 40, 1)) for xi in x_kelembapan], label="Tinggi")
plt.title("Derajat Keanggotaan Kelembapan")
plt.xlabel("Kelembapan (%)")
plt.ylabel("Derajat Keanggotaan")
plt.legend()
plt.grid()
plt.show()
print("\n")

# VISUALISASI AGREGASI DAN JUGA DEFUZZIFIKASI
plt.figure(figsize=(8, 4))
plt.plot(x, agregasi, label='Agregasi Fungsi Keanggotaan Inferensi', color='blue')
plt.fill_between(x, 0, agregasi, color='skyblue', alpha=0.5)
plt.title("Agregasi dan Defuzzifikasi")
plt.axvline(output_crisp, color='red', linestyle='--',
label=f'Kecepatan Kipas (Crisp) : {output_crisp}%')
plt.xlabel("Kecepatan Kipas (%)")
plt.ylabel("Derajat Keanggotaan")
plt.legend()
plt.show()

```

### 3.2 Hasil Running Aplikasi

```
☰ Masukkan nilai Suhu : 19
Masukkan nilai persentase Kelembapan : 70

Fuzzifikasi Suhu :
Dingin : 0.05
Normal : 0.9
Panas : 0

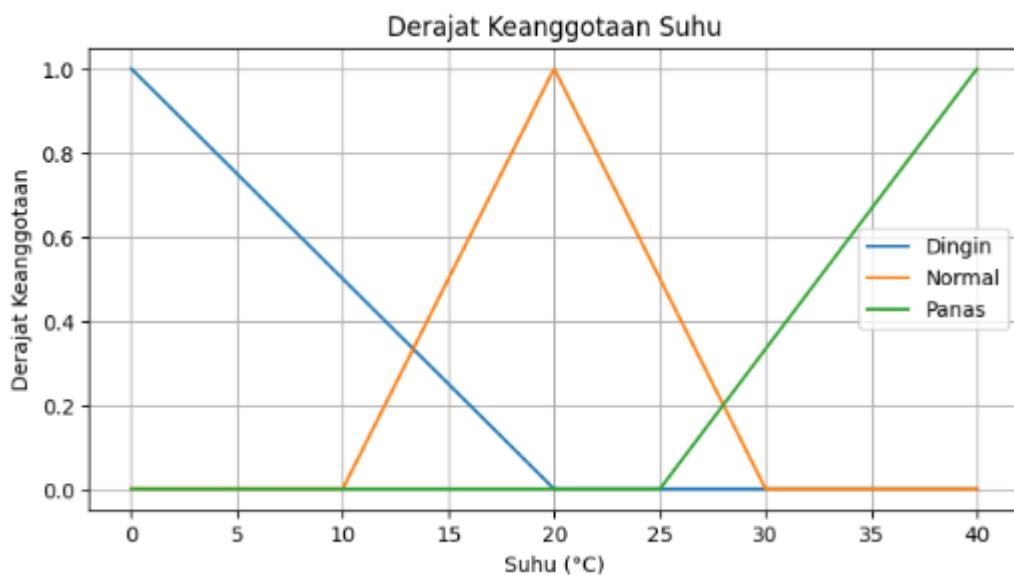
Fuzzifikasi Kelembapan :
Rendah : 0
Sedang : 0
Tinggi : 0.25

Hasil Inferensi Fuzzy :
Lambat : 0
Sedang : 0.05
Cepat : 0.25

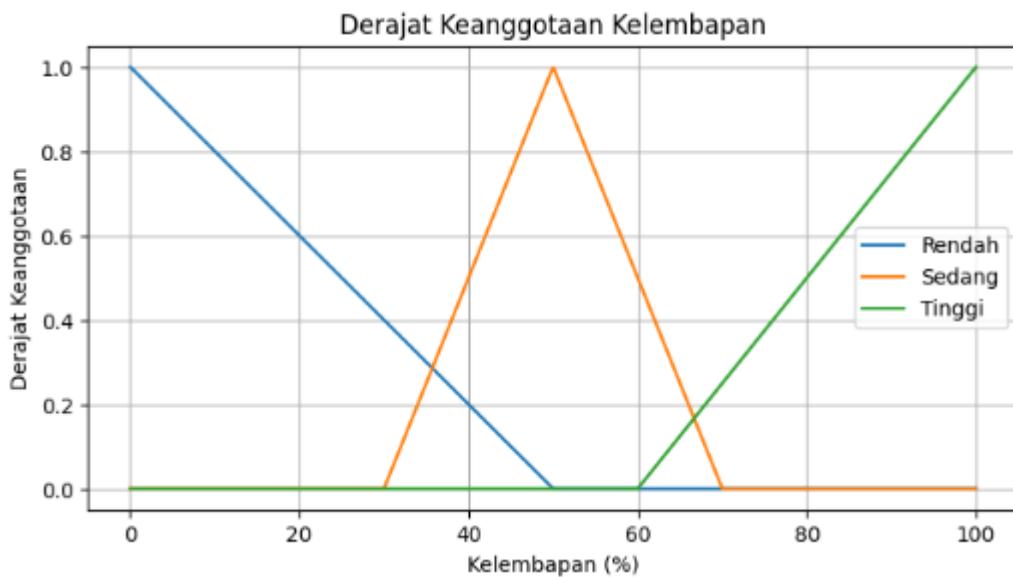
Aturan yang Berlaku :
Jika Suhu Dingin dan Kelembapan Tinggi --> Sedang, dengan Hasil AND = 0.05
Jika Suhu Normal dan Kelembapan Tinggi --> Cepat, dengan Hasil AND = 0.25

Kecepatan Kipas Secara Crisp : 79.73%
Kecepatan Kipas Secara Linguistik : Cepat
```

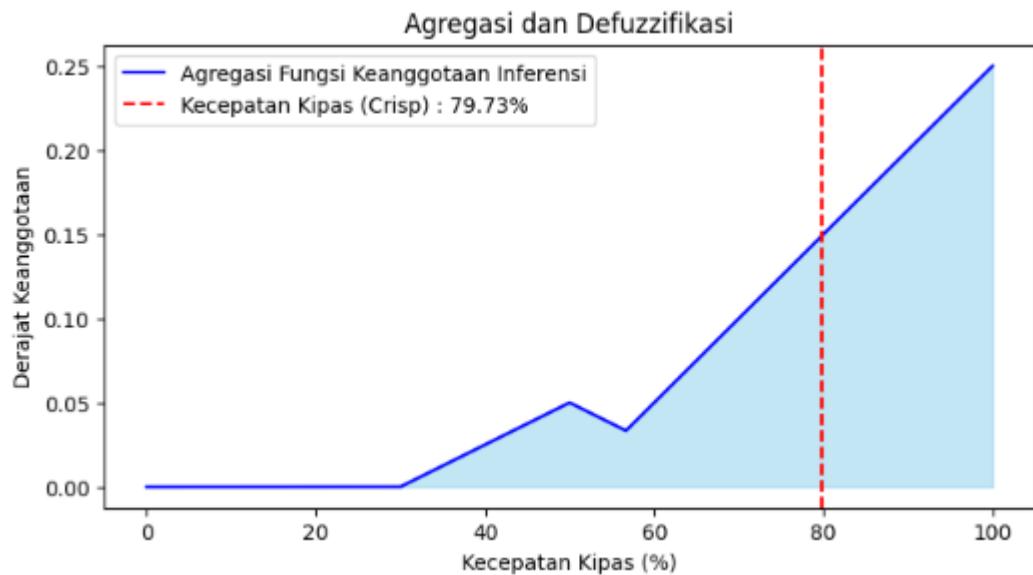
Gambar 3. 1 Output Sistem Pada Console



Gambar 3. 2 Grafik Derajat Keanggotaan Suhu



Gambar 3. 3 Grafik Derajat Keanggotaan Kelembapan



Gambar 3. 4 Grafik Hasil Agregasi dan Defuzzifikasi

## **BAB IV**

### **KESIMPULAN**

Hasil implementasi sistem ini menunjukkan bahwa pengaturan kecepatan kipas otomatis berbasis logika fuzzy dapat diimplementasikan dengan baik menggunakan Sistem Inferensi Fuzzy metode Mamdani. Logika fuzzy memberikan kemampuan untuk menangani data *input* yang bersifat tidak pasti, seperti suhu yang "normal" atau kelembapan yang "tinggi," sehingga sistem mampu memberikan keputusan yang fleksibel dan responsif terhadap perubahan lingkungan terkini.

Proses kerja sistem terdiri dari tiga tahapan utama, yaitu fuzzifikasi, inferensi, dan defuzzifikasi. Pada tahap fuzzifikasi, nilai numerik suhu dan kelembapan diubah menjadi nilai fuzzy dengan menggunakan fungsi keanggotaan segitiga dan trapesium yang dirancang sesuai dengan kategori linguistik seperti "Dingin," "Normal," dan "Panas" untuk suhu, serta "Rendah," "Sedang," dan "Tinggi" untuk kelembapan. Hasil fuzzifikasi ini kemudian digunakan dalam tahap inferensi, di mana aturan berbasis logika linguistik diterapkan menggunakan operator AND (*min*) untuk menentukan derajat keanggotaan *output* fuzzy. Selanjutnya, pada tahap defuzzifikasi, *output* fuzzy ini diubah menjadi nilai numerik (*crisp*) dengan menggunakan metode Centroid, sehingga menghasilkan nilai persentase kecepatan kipas yang presisi.

Percobaan sistem dilakukan dengan berbagai skenario *input* suhu dan kelembapan. Hasil pengujian menunjukkan bahwa sistem mampu menghasilkan keluaran yang akurat, baik dalam bentuk nilai *crisp* maupun linguistik. Implementasi sistem menggunakan bahasa pemrograman Python juga memungkinkan visualisasi proses, termasuk grafik fungsi keanggotaan, hasil agregasi, dan defuzzifikasi, yang memberikan proyeksi menyeluruh tentang mekanisme pengambilan keputusan dalam sistem yang dibangun.

Dengan pendekatan berbasis logika fuzzy ini, sistem pengaturan kecepatan kipas otomatis diharapkan dapat menjadi solusi yang efisien, responsif, dan mudah diadaptasi untuk berbagai kebutuhan, sehingga memberikan kenyamanan optimal bagi penggunanya.