

PRAKTIKUM
SISTEM CERDAS DAN PENDUKUNG KEPUTUSAN
SEMESTER GENAP T.A 2024/2025
LAPORAN PROYEK AKHIR



DISUSUN OLEH :

NIM	:	123230185 123230210
NAMA	:	ATIQA DESYTA ZAHRANI SALSABILLA RIZKY SETYABUDI
PLUG	:	IF-G
NAMA ASISTEN	:	ARVIDION HAVAS OKTAVIAN PANCA AULIA RAHMAN

PROGRAM STUDI INFORMATIKA
JURUSAN INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"
YOGYAKARTA
2025

HALAMAN PENGESAHAN LAPORAN PROYEK AKHIR

Disusun oleh :

Atiqa Desyta Zahrani 123230185
Salsabilla Rizky Setyabudi 123230210

Telah Diperiksa dan Disetujui oleh Asisten Praktikum Sistem Cerdas dan
Pendukung Keputusan
Pada Tanggal :

Asisten Praktikum

Asisten Praktikum

Arvidion Havas Oktavian
NIM. 123220067

Panca Aulia Rahman
NIM. 123200099

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa yang senantiasa mencerahkan rahmat dan hidayah-Nya sehingga saya dapat menyelesaikan praktikum Sistem Cerdas dan Pendukung Keputusan serta laporan proyek akhir praktikum yang berjudul Rekomendasi Tanaman Pertanian Menggunakan Fuzzy Logic Berdasarkan Karakteristik Tanah dan Iklim. Adapun laporan ini berisi tentang proyek akhir yang saya pilih dari hasil pembelajaran selama praktikum berlangsung.

Tidak lupa ucapan terimakasih kepada asisten dosen yang selalu membimbing dan mengajari saya dalam melaksanakan praktikum dan dalam menyusun laporan ini. Laporan ini masih sangat jauh dari kesempurnaan, oleh karena itu kritik serta saran yang membangun saya harapkan untuk menyempurnakan laporan akhir ini.

Atas perhatian dari semua pihak yang membantu penulisan ini, saya ucapkan terimakasih. Semoga laporan ini dapat dipergunakan seperlunya.

Yogyakarta, 2 Juni 2025

Penyusun I

Penyusun II

Atiqa Desyta Zahrani

NIM. 123230185

Salsabilla Rizky Setyabudi

NIM. 123230210

DAFTAR ISI

HALAMAN PENGESAHAN.....	1
LAPORAN PROYEK AKHIR.....	1
KATA PENGANTAR.....	2
DAFTAR ISI.....	3
BAB I.....	5
PENDAHULUAN.....	5
1.1 Latar Belakang Masalah.....	5
1.2 Tujuan Proyek Akhir.....	5
1.3 Manfaat Proyek Akhir.....	5
BAB II.....	7
PEMBAHASAN.....	7
2.1 Dasar Teori.....	7
2.2 Deskripsi Umum Proyek Akhir.....	7
2.3 Inti Pembahasan.....	8
BAB III.....	30
JADWAL PENGERJAAN DAN PEMBAGIAN TUGAS.....	30
3.1 Jadwal Penggerjaan.....	30
3.2 Pembagian Tugas.....	30
BAB IV.....	31
KESIMPULAN DAN SARAN.....	31
4.1 Kesimpulan.....	31
4.2 Saran.....	31
DAFTAR PUSTAKA.....	32

**Rekomendasi Tanaman Pertanian Menggunakan Fuzzy
Logic Berdasarkan Karakteristik Tanah dan Iklim**

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam dunia pertanian, pemilihan jenis tanaman yang tepat sangat bergantung pada kondisi lingkungan seperti pH tanah, suhu udara, kelembaban, dan curah hujan. Namun, banyak petani masih menentukan pilihan secara manual atau berdasarkan pengalaman, tanpa analisis data yang akurat. Hal ini sering menyebabkan hasil panen yang tidak optimal, bahkan gagal panen. Untuk mengatasi ketidakpastian dalam penentuan tanaman yang cocok, dibutuhkan sistem cerdas yang mampu mengolah data karakteristik lahan dan memberikan rekomendasi secara fleksibel. Salah satu metode yang sesuai untuk menangani data yang bersifat tidak pasti dan bervariasi adalah logika fuzzy.

1.2 Tujuan Proyek Akhir

Membangun sebuah sistem pendukung keputusan berbasis logika fuzzy yang mampu merekomendasikan jenis tanaman pertanian secara otomatis berdasarkan data karakteristik lahan, seperti pH tanah, suhu, kelembaban, dan curah hujan. Sistem ini diharapkan dapat membantu petani dalam mengambil keputusan yang lebih tepat, dengan memanfaatkan metode Fuzzy Mamdani untuk mengolah data yang bersifat tidak pasti. Selain itu, proyek ini juga bertujuan untuk menunjukkan penerapan metode fuzzy dalam bidang pertanian secara praktis dengan menggunakan dataset nyata dari Kaggle.

1.3 Manfaat Proyek Akhir

- 1) Membantu Pengambilan Keputusan Petani : Sistem ini dapat memberikan rekomendasi tanaman yang sesuai berdasarkan kondisi lahan dan iklim, sehingga membantu petani dalam mengambil keputusan yang lebih tepat dan berbasis data, bukan hanya berdasarkan intuisi atau pengalaman.
- 2) Mengurangi Risiko Kegagalan Panen : Dengan memilih tanaman yang sesuai dengan karakteristik tanah dan cuaca, risiko gagal panen dapat diminimalkan. Ini berdampak langsung pada peningkatan produktivitas dan hasil pertanian.
- 3) Penerapan Teknologi Cerdas di Sektor Pertanian : Proyek ini menunjukkan penerapan metode logika fuzzy sebagai bagian dari teknologi pertanian modern (smart farming), yang dapat mendorong inovasi dan efisiensi dalam pengelolaan lahan pertanian.
- 4) Sebagai Contoh Implementasi Fuzzy Logic : Sistem ini dapat dijadikan contoh nyata penerapan metode Fuzzy Mamdani dalam kehidupan sehari-hari, khususnya dalam bidang pertanian, sehingga bermanfaat sebagai bahan pembelajaran bagi mahasiswa dan pengembangan teknologi.
- 5) Meningkatkan Kesadaran akan Pentingnya Data : Penggunaan dataset dari Kaggle menunjukkan bahwa keputusan yang baik dapat diambil dari

pengolahan data yang akurat. Ini bisa meningkatkan kesadaran petani atau pelaku pertanian terhadap pentingnya pengumpulan dan analisis data lahan.

BAB II

PEMBAHASAN

2.1 Dasar Teori

a. Sistem Rekomendasi dalam Bidang Pertanian

Sistem rekomendasi adalah sistem yang dirancang untuk memberikan saran atau alternatif pilihan berdasarkan sejumlah kriteria atau kondisi. Dalam pertanian, sistem ini dapat digunakan untuk membantu petani memilih jenis tanaman yang sesuai dengan kondisi lahan dan iklim lokal guna meningkatkan hasil panen dan efisiensi produksi.

b. Karakteristik Tanah dan Iklim

Kondisi tanah dan iklim sangat menentukan kesesuaian jenis tanaman. Parameter-parameter utama yang diperhatikan meliputi:

- Tanah :
 - pH Tanah : Menentukan tingkat keasaman yang mempengaruhi ketersediaan unsur hara.
 - Tekstur Tanah : Mempengaruhi kapasitas penyerapan air dan udara.
 - Kandungan Hara : Seperti nitrogen, fosfor, dan kalium.
- Iklim :
 - Suhu : Berpengaruh pada metabolisme dan fotosintesis tanaman.
 - Curah Hujan : Mempengaruhi kebutuhan air tanaman.
 - Kelembaban Udara : Berperan dalam penguapan dan transpirasi.

c. Fuzzy Logic

Fuzzy logic adalah metode pengambilan keputusan yang dapat menangani ketidakpastian dan ketidaktegasan data. Tidak seperti logika biner yang hanya mengenal 0 dan 1, fuzzy logic memperbolehkan nilai tengah di antara keduanya. Komponen utama fuzzy logic meliputi:

- Fuzzifikasi: Mengubah input numerik menjadi himpunan fuzzy.
- Aturan Dasar: Menentukan logika hubungan antar variabel.
- Inferensi: Proses penarikan kesimpulan dari aturan fuzzy.
- Defuzzifikasi: Mengubah output fuzzy menjadi output numerik/tegas.

Fuzzy logic sangat sesuai digunakan dalam sistem berbasis lingkungan yang datanya tidak pasti atau bervariasi, seperti tanah dan iklim.

2.2 Deskripsi Umum Proyek Akhir

Proyek akhir ini bertujuan untuk merancang dan membangun sistem rekomendasi tanaman pertanian berbasis fuzzy logic. Sistem ini akan menerima input berupa parameter karakteristik tanah (seperti pH, tekstur) dan iklim (seperti suhu, curah hujan, kelembaban), lalu memberikan output berupa rekomendasi jenis tanaman yang sesuai untuk kondisi tersebut.

Sistem ini ditujukan untuk membantu petani, penyuluh pertanian, atau instansi pertanian dalam menentukan tanaman yang optimal ditanam di suatu wilayah. Dengan memanfaatkan fuzzy logic, sistem dapat mempertimbangkan ketidakpastian atau variasi kondisi lingkungan secara fleksibel dan cerdas.

Proyek akan diimplementasikan dalam bentuk perangkat lunak sederhana menggunakan bahasa pemrograman (misalnya Python) dengan antarmuka pengguna berbasis web atau desktop.

2.3 Inti Pembahasan

a. Identifikasi Masalah

Petani sering menghadapi kesulitan dalam memilih jenis tanaman yang sesuai dengan kondisi tanah dan iklim lokal karena informasi teknis yang terbatas dan kondisi lingkungan yang dinamis. Tidak adanya sistem bantuan yang bersifat fleksibel juga menyulitkan dalam pengambilan keputusan yang optimal.

b. Solusi yang Diberikan

Solusi yang ditawarkan adalah sistem rekomendasi berbasis fuzzy logic yang mampu memproses data tanah dan iklim untuk menghasilkan saran tanaman yang cocok. Sistem ini fleksibel karena menggunakan pendekatan logika fuzzy yang bisa menangani ketidakpastian input.

c. Langkah-langkah Pengembangan

1. Pengumpulan Data Parameter Lingkungan: Mengumpulkan rentang nilai pH tanah, suhu, kelembaban, dan curah hujan.
2. Perancangan Fuzzy System:
 - a. Menentukan input variabel dan membership function untuk tiap parameter.
 - b. Menentukan output variable berupa tanaman yang direkomendasikan.
 - c. Menyusun rule base berdasarkan pengetahuan pakar atau referensi ilmiah.
3. Implementasi Sistem : Menggunakan bahasa pemrograman untuk membangun sistem (misalnya Python + Streamlit).
4. Uji Coba dan Evaluasi : Menggunakan data nyata untuk menguji akurasi rekomendasi dan keandalan sistem.

d. Contoh Aturan Fuzzy (Rule Base)

Contoh aturan logika fuzzy dalam sistem ini:

- IF pH tanah asam AND suhu tinggi AND curah hujan tinggi THEN tanaman yang direkomendasikan adalah padi.

- IF pH tanah netral AND suhu sedang AND kelembaban rendah THEN tanaman yang direkomendasikan adalah jagung.

e. Manfaat Sistem

- Membantu pengambilan keputusan berbasis data.
- Meningkatkan produktivitas pertanian.
- Mengurangi risiko kegagalan panen akibat salah pilih tanaman.
- Edukatif bagi petani dalam memahami pentingnya kondisi tanah dan iklim.

f. Perhitungan Algoritma Fuzzy

- Langkah 1 : Fuzzifikasi - Mengubah input numerik menjadi himpunan fuzzy.
 - Fuzzy Set : Temperature

```
temperature = ctrl.Antecedent(np.arange(0, 51, 1), 'temperature')

temperature['low'] = fuzz.trimf(temperature.universe, [0, 0, 25])
temperature['medium'] = fuzz.trapmf(temperature.universe, [20, 23, 27, 30])
temperature['high'] = fuzz.trimf(temperature.universe, [35, 50, 50])
```

→ Low : trimf [0, 0, 25]

$$\mu_{\text{low}}(x) = \begin{cases} 1, & 0 \leq x \leq 0 \\ \frac{25-x}{25-0}, & 0 < x < 25 \\ 0, & x \geq 25 \end{cases}$$

→ Medium : trapmf [20, 23, 27, 30]

$$\mu_{\text{medium}}(x) = \begin{cases} 0, & x \leq 20 \\ \frac{x-20}{23-20}, & 20 < x < 23 \\ 1, & 23 \leq x \leq 27 \\ \frac{30-x}{30-27}, & 27 < x < 30 \\ 0, & x \geq 30 \end{cases}$$

→ High : trimf [35, 50, 50]

$$\mu_{\text{high}}(x) = \begin{cases} 0, & x \leq 35 \\ \frac{x-35}{50-35}, & 35 < x < 50 \\ 1, & x = 50 \end{cases}$$

- Fuzzy Set : Humidity

```
humidity = ctrl.Antecedent(np.arange(0, 101, 1), 'humidity')

humidity['low'] = fuzz.trimf(humidity.universe, [0, 0, 50])
humidity['medium'] = fuzz.trapmf(humidity.universe, [40, 50, 70, 80])
humidity['high'] = fuzz.trimf(humidity.universe, [70, 100, 100])
```

→ Low : trimf [0, 0, 50]

$$\mu_{\text{low}}(x) = \begin{cases} 1, & x = 0 \\ \frac{50-x}{50}, & 0 < x < 50 \\ 0, & x \geq 50 \end{cases}$$

→ Medium : trapmf [40, 50, 70, 80]

$$\mu_{\text{medium}}(x) = \begin{cases} 0, & x \leq 40 \\ \frac{x-40}{50-40}, & 40 < x < 50 \\ 1, & 50 \leq x \leq 70 \\ \frac{80-x}{80-70}, & 70 < x < 80 \\ 0, & x \geq 80 \end{cases}$$

→ High : trimf [70, 100, 100]

$$\mu_{\text{high}}(x) = \begin{cases} 0, & x \leq 70 \\ \frac{x-70}{100-70}, & 70 < x < 100 \\ 1, & x = 100 \end{cases}$$

- Fuzzy Set : Ph

```
ph = ctrl.Antecedent(np.arange(0, 10.1, 0.1), 'ph')

ph['acidic'] = fuzz.trimf(ph.universe, [3, 3, 6])
ph['neutral'] = fuzz.trapmf(ph.universe, [6.0, 6.3, 6.7, 7.0])
ph['alkaline'] = fuzz.trimf(ph.universe, [7, 10, 10])
```

→ Acidic : trimf [3, 3, 6]

$$\mu_{\text{acidic}}(x) = \begin{cases} 1, & x = 3 \\ \frac{6-x}{6-3}, & 3 < x < 6 \\ 0, & x \geq 6 \end{cases}$$

→ Neutral : trapmf [6.0, 6.3, 6.7, 7.0]

$$\mu_{\text{neutral}}(x) = \begin{cases} 0, & x \leq 6.0 \\ \frac{x-6.0}{6.3-6.0}, & 6.0 < x < 6.3 \\ 1, & 6.3 \leq x \leq 6.7 \\ \frac{7.0-x}{7.0-6.7}, & 6.7 < x < 7.0 \\ 0, & x \geq 7.0 \end{cases}$$

→ Alkaline : trimf [7, 10, 10]

$$\mu_{\text{alkaline}}(x) = \begin{cases} 0, & x \leq 7 \\ \frac{x-7}{10-7}, & 7 < x < 10 \\ 1, & x = 10 \end{cases}$$

- Fuzzy Set : Rainfall

```
rainfall = ctrl.Antecedent(np.arange(0, 301, 1), 'rainfall')

rainfall['low'] = fuzz.trimf(rainfall.universe, [0, 0, 150])
rainfall['medium'] = fuzz.trapmf(rainfall.universe, [80, 110, 170, 200])
rainfall['high'] = fuzz.trimf(rainfall.universe, [200, 300, 300])
```

→ Low : trimf [0, 0, 150]

$$\mu_{\text{low}}(x) = \begin{cases} 1, & x = 0 \\ \frac{150-x}{150}, & 0 < x < 150 \\ 0, & x \geq 150 \end{cases}$$

→ Medium : trapmf [80, 110, 170, 200]

$$\mu_{\text{medium}}(x) = \begin{cases} 0, & x \leq 80 \\ \frac{x-80}{110-80}, & 80 < x < 110 \\ 1, & 110 \leq x \leq 170 \\ \frac{200-x}{200-170}, & 170 < x < 200 \\ 0, & x \geq 200 \end{cases}$$

→ High : trimf [200, 300, 300]

$$\mu_{\text{high}}(x) = \begin{cases} 0, & x \leq 200 \\ \frac{x-200}{300-200}, & 200 < x < 300 \\ 1, & x = 300 \end{cases}$$

- Fuzzy Set : Tanaman

```

tanaman = ctrl.Consequent(np.arange(0, 101, 1), 'tanaman')

tanaman['rice'] = fuzz.trimf(tanaman.universe, [0, 0, 33])
tanaman['maize'] = fuzz.trimf(tanaman.universe, [25, 50, 75])
tanaman['chickpea'] = fuzz.trimf(tanaman.universe, [67, 100, 100])

```

→ Rice : trimf [0, 0, 33]

$$\mu_{\text{rice}}(x) = \begin{cases} 1, & x = 0 \\ \frac{33-x}{33}, & 0 < x < 33 \\ 0, & x \geq 33 \end{cases}$$

→ Maize : trimf [25, 50, 75]

$$\mu_{\text{maize}}(x) = \begin{cases} 0, & x \leq 25 \\ \frac{x-25}{50-25}, & 25 < x < 50 \\ \frac{75-x}{75-50}, & 50 < x < 75 \\ 0, & x \geq 75 \end{cases}$$

→ Chickpea : trimf [67, 100, 100])

$$\mu_{\text{chickpea}}(x) = \begin{cases} 0, & x \leq 67 \\ \frac{x-67}{100-67}, & 67 < x < 100 \\ 1, & x = 100 \end{cases}$$

Berikut untuk source code bagian fuzzifikasi :

```

tanaman_sim.input['temperature'] = input_temp
tanaman_sim.input['humidity'] = input_hum
tanaman_sim.input['ph'] = input_ph
tanaman_sim.input['rainfall'] = input_rain

```

- Langkah 2 : Aturan Dasar

```

# Aturan fuzzy Rice(Padi), Maize(Jagung), Chickpea(Kacang Arab)
rules = [
    # Rules for Rice (Padi) - generally likes moderate to high
    # humidity and rainfall, varying temps, and acidic to neutral pH
    ctrl.Rule(temperature['low'] & humidity['medium'] &
              ph['acidic'] & rainfall['medium'], tanaman['rice']),
    ctrl.Rule(temperature['low'] & humidity['high'] &
              ph['acidic'] & rainfall['medium'], tanaman['rice']),
    ctrl.Rule(temperature['low'] & humidity['high'] &
              ph['neutral'] & rainfall['high'], tanaman['rice']),
    ctrl.Rule(temperature['medium'] & humidity['high'] &

```

```

ph['acidic'] & rainfall['high'], tanaman['rice']),
    ctrl.Rule(temperature['medium'] & humidity['medium'] &
ph['neutral'] & rainfall['medium'], tanaman['rice']),
    ctrl.Rule(temperature['low'] & humidity['low'] &
ph['neutral'] & rainfall['low'], tanaman['rice']),


        # Rules for Maize (Jagung) - generally likes moderate
temperature, humidity, and rainfall, neutral pH
        ctrl.Rule(temperature['medium'] & humidity['medium'] &
ph['neutral'] & rainfall['medium'], tanaman['maize']),
        ctrl.Rule(temperature['medium'] & humidity['medium'] &
ph['neutral'] & rainfall['low'], tanaman['maize']),
        ctrl.Rule(temperature['medium'] & humidity['medium'] &
ph['neutral'] & rainfall['high'], tanaman['maize']),
        ctrl.Rule(temperature['medium'] & humidity['low'] &
ph['neutral'] & rainfall['medium'], tanaman['maize']),
        ctrl.Rule(temperature['high'] & humidity['medium'] &
ph['neutral'] & rainfall['medium'], tanaman['maize']),
        ctrl.Rule(temperature['low'] & humidity['medium'] &
ph['neutral'] & rainfall['medium'], tanaman['maize']),


        # Rules for Chickpea (Kacang Arab) - generally likes
warmer, drier conditions and alkaline soil
        ctrl.Rule(temperature['high'] & humidity['low'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
        ctrl.Rule(temperature['high'] & humidity['low'] &
ph['alkaline'] & rainfall['medium'], tanaman['chickpea']),
        ctrl.Rule(temperature['high'] & humidity['medium'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
        ctrl.Rule(temperature['medium'] & humidity['low'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
        ctrl.Rule(temperature['high'] & humidity['high'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
        ctrl.Rule(temperature['high'] & humidity['high'] &
ph['acidic'] & rainfall['medium'], tanaman['chickpea']),
    ]

```

- Langkah 3 : Inferensi

```
tanaman_sim.compute()
```

- Langkah 4 : Defuzzifikasi

```
output_tanaman = tanaman_sim.output['tanaman']
```

g. Listing Program dan Hasil Program

Source code program :

```
import streamlit as st
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import skfuzzy as fuzz
from skfuzzy import control as ctrl
import hashlib
import json
import os
import base64

st.set_page_config(page_title="Fuzzy Crop Recommendation",
layout="centered")

# ----- BACKGROUND -----
def get_base64_of_bin_file(bin_file):
    with open(bin_file, 'rb') as f:
        data = f.read()
    return base64.b64encode(data).decode()

def set_bg_from_local(image_file):
    bin_str = get_base64_of_bin_file(image_file)
    page_bg_img = f'''
<style>
[data-testid="stApp"] {{
    background-image: url("data:image/png;base64,{bin_str}");
    background-size: cover;
    background-position: center;
    background-repeat: no-repeat;
    background-attachment: fixed;
}}
</style>
'''
    st.markdown(page_bg_img, unsafe_allow_html=True)

# Set background (ganti path sesuai file kamu)
set_bg_from_local("D:/prak scpk/data.jpg")

# ----- USER LOGIN -----
USER_FILE = "users.json"

def load_users():
    if os.path.exists(USER_FILE):
        with open(USER_FILE, "r") as f:
            return json.load(f)
    return {}

def save_users(users):
```

```

with open(USER_FILE, "w") as f:
    json.dump(users, f)

def hash_password(password):
    return hashlib.sha256(password.encode()).hexdigest()

users = load_users()

if 'logged_in' not in st.session_state:
    st.session_state.logged_in = False
if 'page' not in st.session_state:
    st.session_state.page = 'login'
if 'username' not in st.session_state:
    st.session_state.username = ''

def show_login():
    st.title("Login")
    username = st.text_input("Username")
    password = st.text_input("Password", type="password")
    if st.button("Login"):
        if username in users and users[username] == hash_password(password):
            st.session_state.logged_in = True
            st.session_state.username = username
            st.success(f"Selamat datang, {username}!")
            st.rerun()
        else:
            st.error("Username atau password salah.")
    st.write("Belum punya akun?")
    if st.button("Daftar sekarang"):
        st.session_state.page = 'signup'
        st.rerun()

def show_signup():
    st.title("Daftar Akun Baru")
    new_user = st.text_input("Username baru")
    new_pass = st.text_input("Password baru", type="password")
    if st.button("Daftar"):
        if new_user.strip() == "" or new_pass.strip() == "":
            st.warning("Username dan password tidak boleh kosong.")
        elif new_user in users:
            st.warning("Username sudah terdaftar.")
        else:
            users[new_user] = hash_password(new_pass)
            save_users(users)
            st.success("Akun berhasil dibuat! Silakan login.")
            st.session_state.page = 'login'
            st.rerun()
    if st.button("Kembali ke login"):
        st.session_state.page = 'login'
        st.rerun()
# ----- DASHBOARD & FUZZY ----- #

```

```

def show_dashboard():
    st.sidebar.write(f"👤 User: **{st.session_state.username}**")
    if st.sidebar.button("Logout"):
        st.session_state.logged_in = False
        st.session_state.page = 'login'
        st.session_state.username = ''
        st.rerun()

    menu = st.sidebar.selectbox("📚 Menu", ["Lihat Dataset", "Input dan Hasil Fuzzy"])

    @st.cache_data
    def load_data():
        return pd.read_csv("dataset_fuzzy_3_tanaman.csv")

    try:
        data = load_data()
        data_loaded = True
    except Exception as e:
        st.warning("⚠️ Gagal memuat dataset.")
        st.error(f"Detail error: {e}")
        data_loaded = False

    if menu == "Lihat Dataset":
        st.title("📋 Dataset Fuzzy Crop")
        if data_loaded:
            num_rows = st.sidebar.number_input("Jumlah baris", 5, len(data), 10)
            st.dataframe(data.head(num_rows))
            with st.expander("📊 Statistik"):
                st.write("Ukuran:", data.shape)
                st.write(data.describe())
                st.write("Missing Values:")
                st.write(data.isnull().sum())
        else:
            st.title("⚡ Input dan Rekomendasi Fuzzy")

        col1, col2 = st.columns(2)
        with col1:
            input_temp = st.slider("🌡️ Temperature (°C)", 0.0, 50.0, 25.0)
            input_ph = st.slider("🧪 pH", 0.0, 10.0, 5.0)
        with col2:
            input_hum = st.slider("💧 Humidity (%)", 0.0, 100.0, 50.0)
            input_rain = st.slider("🌧️ Rainfall (mm)", 0.0, 300.0, 150.0)

        # Setup fuzzy variables and membership functions
        temperature = ctrl.Antecedent(np.arange(0, 51, 1), 'temperature')
        humidity = ctrl.Antecedent(np.arange(0, 101, 1), 'humidity')
        ph = ctrl.Antecedent(np.arange(0, 10.1, 0.1), 'ph')

```

```

rainfall = ctrl.Antecedent(np.arange(0, 301, 1), 'rainfall')

tanaman = ctrl.Consequent(np.arange(0, 101, 1), 'tanaman')

temperature['low'] = fuzz.trimf(temperature.universe, [0, 0,
25])
temperature['medium'] = fuzz.trapmf(temperature.universe, [20,
23, 27, 30])
temperature['high'] = fuzz.trimf(temperature.universe, [35, 50,
50])

humidity['low'] = fuzz.trimf(humidity.universe, [0, 0, 50])
humidity['medium'] = fuzz.trapmf(humidity.universe, [40, 50,
70, 80])
humidity['high'] = fuzz.trimf(humidity.universe, [70, 100,
100])

ph['acidic'] = fuzz.trimf(ph.universe, [3, 3, 6])
ph['neutral'] = fuzz.trapmf(ph.universe, [6.0, 6.3, 6.7, 7.0])
ph['alkaline'] = fuzz.trimf(ph.universe, [7, 10, 10])

rainfall['low'] = fuzz.trimf(rainfall.universe, [0, 0, 150])
rainfall['medium'] = fuzz.trapmf(rainfall.universe, [80, 110,
170, 200])
rainfall['high'] = fuzz.trimf(rainfall.universe, [200, 300,
300])

tanaman['rice'] = fuzz.trimf(tanaman.universe, [0, 0, 33])
tanaman['maize'] = fuzz.trimf(tanaman.universe, [25, 50, 75])
tanaman['chickpea'] = fuzz.trimf(tanaman.universe, [67, 100,
100])

# Aturan fuzzy (Tambahkan rules untuk cakupan yang lebih baik)
rules = [
    # Rules for Rice (Padi) - generally likes moderate to high
    # humidity and rainfall, varying temps, and acidic to neutral pH
    ctrl.Rule(temperature['low'] & humidity['medium'] &
    ph['acidic'] & rainfall['medium'], tanaman['rice']),
    ctrl.Rule(temperature['low'] & humidity['high'] &
    ph['acidic'] & rainfall['medium'], tanaman['rice']),
    ctrl.Rule(temperature['low'] & humidity['high'] &
    ph['neutral'] & rainfall['high'], tanaman['rice']),
    ctrl.Rule(temperature['medium'] & humidity['high'] &
    ph['acidic'] & rainfall['high'], tanaman['rice']),
    ctrl.Rule(temperature['medium'] & humidity['medium'] &
    ph['neutral'] & rainfall['medium'], tanaman['rice']),
    ctrl.Rule(temperature['low'] & humidity['low'] &
    ph['neutral'] & rainfall['low'], tanaman['rice']),

    # Rules for Maize (Jagung) - generally likes moderate
    # temperature, humidity, and rainfall, neutral pH
    ctrl.Rule(temperature['medium'] & humidity['medium'] &

```

```

ph['neutral'] & rainfall['medium'], tanaman['maize']),
    ctrl.Rule(temperature['medium'] & humidity['medium'] &
ph['neutral'] & rainfall['low'], tanaman['maize']),
    ctrl.Rule(temperature['medium'] & humidity['medium'] &
ph['neutral'] & rainfall['high'], tanaman['maize']),
    ctrl.Rule(temperature['medium'] & humidity['low'] &
ph['neutral'] & rainfall['medium'], tanaman['maize']),
    ctrl.Rule(temperature['high'] & humidity['medium'] &
ph['neutral'] & rainfall['medium'], tanaman['maize']),
    ctrl.Rule(temperature['low'] & humidity['medium'] &
ph['neutral'] & rainfall['medium'], tanaman['maize']),


        # Rules for Chickpea (Kacang Arab) - generally likes
warmer, drier conditions and alkaline soil
    ctrl.Rule(temperature['high'] & humidity['low'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
    ctrl.Rule(temperature['high'] & humidity['low'] &
ph['alkaline'] & rainfall['medium'], tanaman['chickpea']),
    ctrl.Rule(temperature['high'] & humidity['medium'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
    ctrl.Rule(temperature['medium'] & humidity['low'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
    ctrl.Rule(temperature['high'] & humidity['high'] &
ph['alkaline'] & rainfall['low'], tanaman['chickpea']),
    ctrl.Rule(temperature['high'] & humidity['high'] &
ph['acidic'] & rainfall['medium'], tanaman['chickpea']),
    ]


tanaman_ctrl = ctrl.ControlSystem(rules)
tanaman_sim = ctrl.ControlSystemSimulation(tanaman_ctrl)

try:
    tanaman_sim.input['temperature'] = input_temp
    tanaman_sim.input['humidity'] = input_hum
    tanaman_sim.input['ph'] = input_ph
    tanaman_sim.input['rainfall'] = input_rain
    tanaman_sim.compute()

    output_tanaman = tanaman_sim.output['tanaman']
    if output_tanaman < 33.0:
        label = "rice"
    elif output_tanaman < 66.0:
        label = "maize"
    else:
        label = "chickpea"

    st.success(f"🌿 Rekomendasi tanaman terbaik: **{label}**")
    st.caption(f"Nilai fuzzy output: {output_tanaman:.2f}")

def plot_var(variable, label, nilai_vertikal=None):
    plt.figure(figsize=(8, 4))
    for term_name, term_obj in variable.terms():

```

```

        plt.plot(variable.universe, term_obj.mf,
label=term_name)
            if nilai_vertikal is not None:
                plt.axvline(x=nilai_vertikal, color='r',
linestyle='--', label=f"{label} input")
                plt.title(f"Fungsi Keanggotaan - {label}")
                plt.xlabel(label)
                plt.ylabel("Derajat Keanggotaan")
                plt.legend()
                plt.grid(True)
                st.pyplot(plt.gcf())
                plt.close()

            with st.expander("🔍 Keanggotaan Input"):
                tab1, tab2, tab3, tab4 = st.tabs(["Temperature",
"Humidity", "pH", "Rainfall"])
                with tab1:
                    plot_var(temperature, "Temperature (°C)",
input_temp)
                with tab2:
                    plot_var(humidity, "Humidity (%)", input_hum)
                with tab3:
                    plot_var(ph, "pH", input_ph)
                with tab4:
                    plot_var(rainfall, "Rainfall (mm)", input_rain)

            with st.expander("☒ Fungsi Keanggotaan Output"):
                plt.figure(figsize=(8, 4))
                for term_name, term_obj in tanaman.terms.items():
                    plt.plot(tanaman.universe, term_obj.mf,
label=term_name)
                    plt.axvline(x=output_tanaman, color='purple',
linestyle='--', label='Output')
                    plt.title("Fungsi Keanggotaan - Tanaman
Recommendation")
                    plt.xlabel("Output")
                    plt.ylabel("Derajat Keanggotaan")
                    plt.legend()
                    plt.grid(True)
                    st.pyplot(plt.gcf())
                    plt.close()

        except Exception as e:
            st.error(f"Terjadi kesalahan pada perhitungan fuzzy: {e}")

# ----- MAIN -----
if not st.session_state.logged_in:
    if st.session_state.page == 'login':
        show_login()
    elif st.session_state.page == 'signup':
        show_signup()
else:

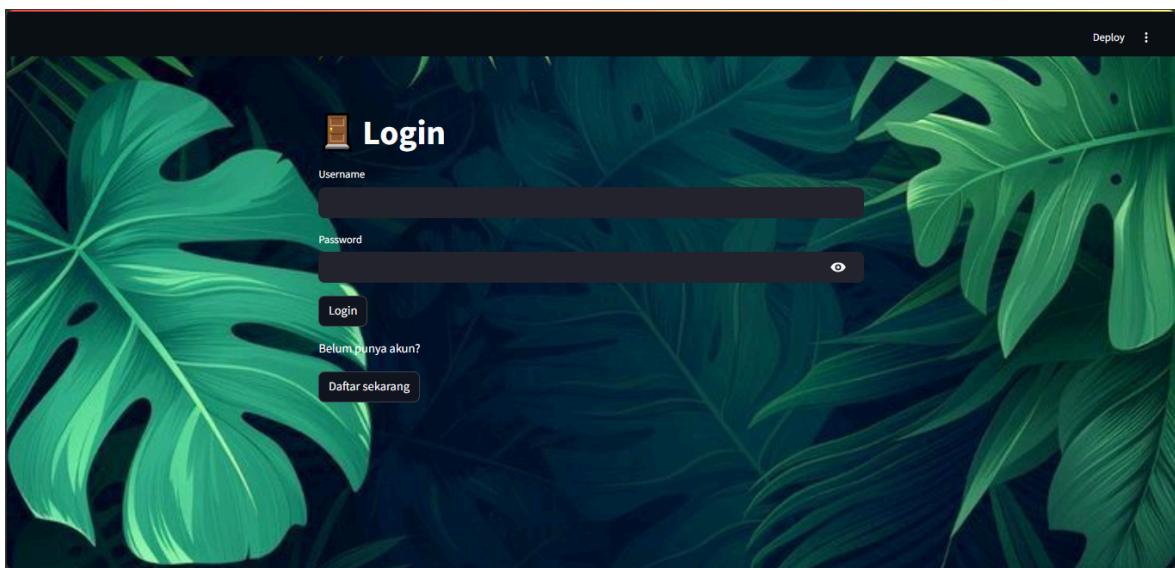
```

```
show_dashboard()
```

2.1. Source Code Program

Hasil Program :

Di halaman ini, terdapat kolom untuk memasukkan username dan password, serta tombol untuk melakukan login. Jika pengguna belum memiliki akun, mereka dapat menekan tombol "**Daftar sekarang**" yang akan mengarahkan ke halaman **Pendaftaran**. Pada halaman ini, pengguna dapat membuat akun baru dengan memasukkan username dan password, lalu akun tersebut akan disimpan dalam file `users.json` secara otomatis.



Gambar 2.3.1. Login Program

Setelah berhasil login, pengguna akan masuk ke halaman **Dashboard** yang memiliki menu navigasi di bagian sidebar. Di sidebar ini, terdapat dua pilihan utama: "**Lihat Dataset**" dan "**Input dan Hasil Fuzzy**". Jika memilih "Lihat Dataset", pengguna dapat melihat isi data tanaman yang digunakan sebagai dasar sistem fuzzy, termasuk jumlah baris yang ingin ditampilkan, serta statistik deskriptif seperti rata-rata, nilai maksimum, dan informasi tentang data yang hilang. Dataset ini dibaca dari file CSV dengan nama `dataset_fuzzy_3_tanaman.csv`.

Dataset Fuzzy Crop

	N	P	K	temperature	humidity	ph	rainfall	label
0	90	42	43	20.8797	82.0027	6.503	202.9355	rice
1	85	58	41	21.7705	80.3196	7.0381	226.6555	rice
2	60	55	44	23.0045	82.3208	7.8402	26.9642	rice
3	74	35	40	26.4911	80.1584	6.9804	242.864	rice
4	78	42	42	20.1302	81.6049	7.6285	262.7173	rice
5	69	37	42	23.058	83.3701	7.0735	251.055	rice
6	69	55	38	22.7088	82.6394	5.7008	271.3249	rice
7	94	53	40	20.2777	82.8941	5.7186	241.9742	rice
8	89	54	38	24.5159	83.5352	6.6853	230.4462	rice
9	68	58	38	23.224	83.0332	6.3363	221.2092	rice

Statistik

Ukuran: (300, 300)

	N	P	K	temperature	humidity	ph	rainfall
count	300	300	300	300	300	300	300
mean	65.9133	54.6033	46.5267	21.6005	54.7418	6.6692	133.669
std	21.8744	12.1619	25.2312	2.8895	27.9415	0.8329	75.9337
min	20	35	15	17.025	14.258	5.0063	40.6517
25%	52	44.75	22	15.0791	18.2591	6.0593	77.1982
50%	68.5	55	40	21.3244	65.3038	6.5373	90.6152
75%	83.25	61	77	24.2616	80.9085	7.1566	203.4017
max	200	80	85	26.93	84.9691	8.8687	298.5861

Missing Values:

0	
N	0
P	0
K	0
temperature	0
humidity	0
ph	0
rainfall	0
label	0

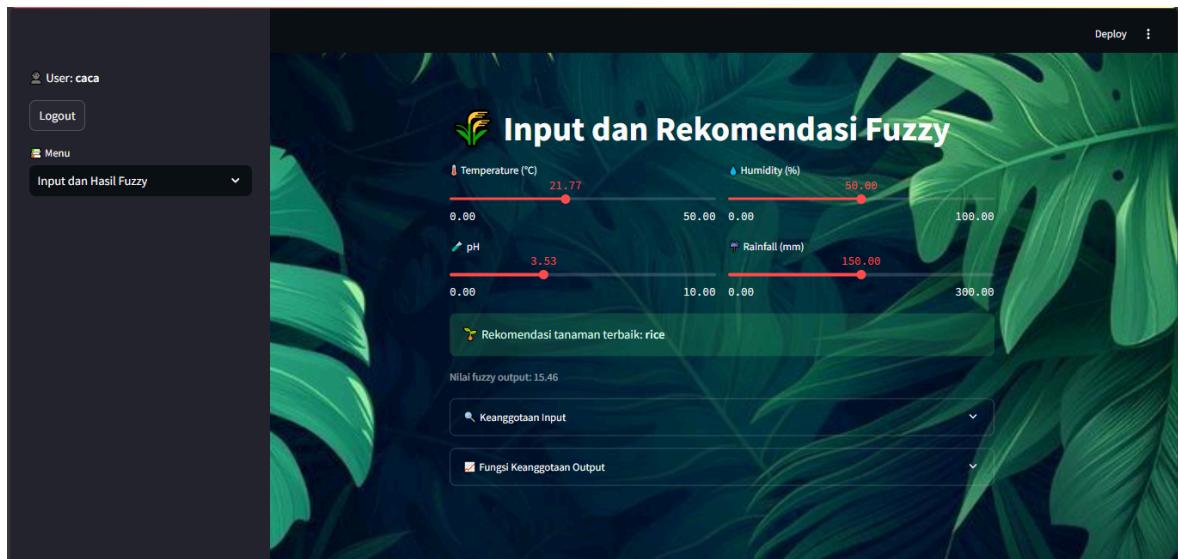
Gambar 2.3.2 Lihat Dataset

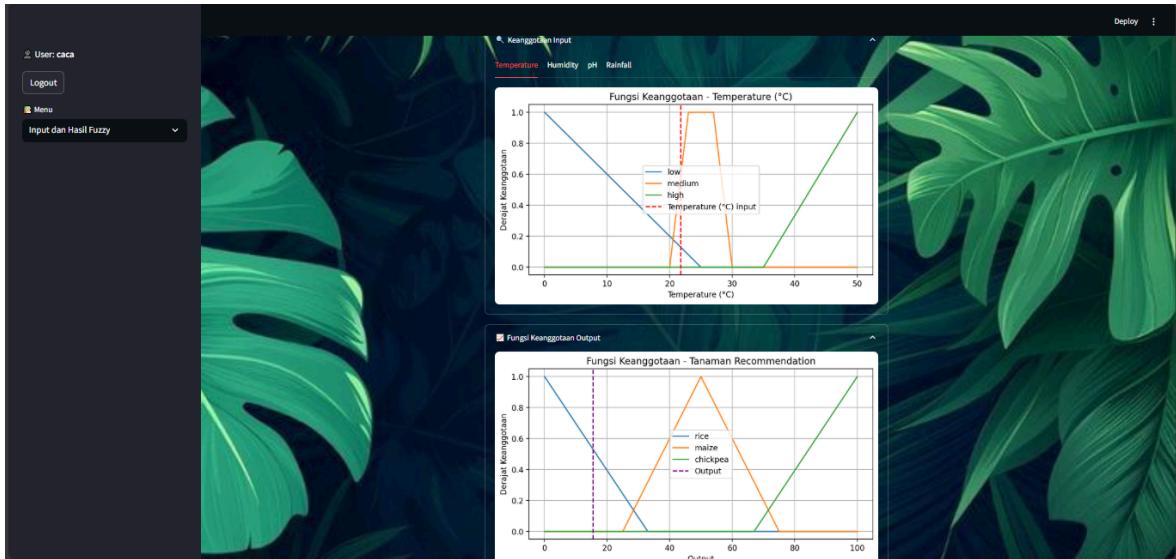
Menu kedua, yaitu "**Input dan Hasil Fuzzy**", adalah inti dari aplikasi ini. Di halaman ini, pengguna dapat memasukkan nilai-nilai untuk suhu (temperature), kelembaban (humidity), pH tanah, dan curah hujan (rainfall) menggunakan slider. Setelah input dimasukkan, sistem fuzzy akan melakukan perhitungan berbasis aturan yang telah ditentukan sebelumnya. Nilai-nilai input akan difuzzifikasi ke dalam derajat keanggotaan (μ) dari masing-masing kategori, lalu diproses melalui aturan fuzzy Mamdani yang melibatkan kombinasi kondisi-kondisi seperti "jika suhu rendah dan kelembaban tinggi dan pH asam dan curah hujan sedang maka tanam padi", dan seterusnya.



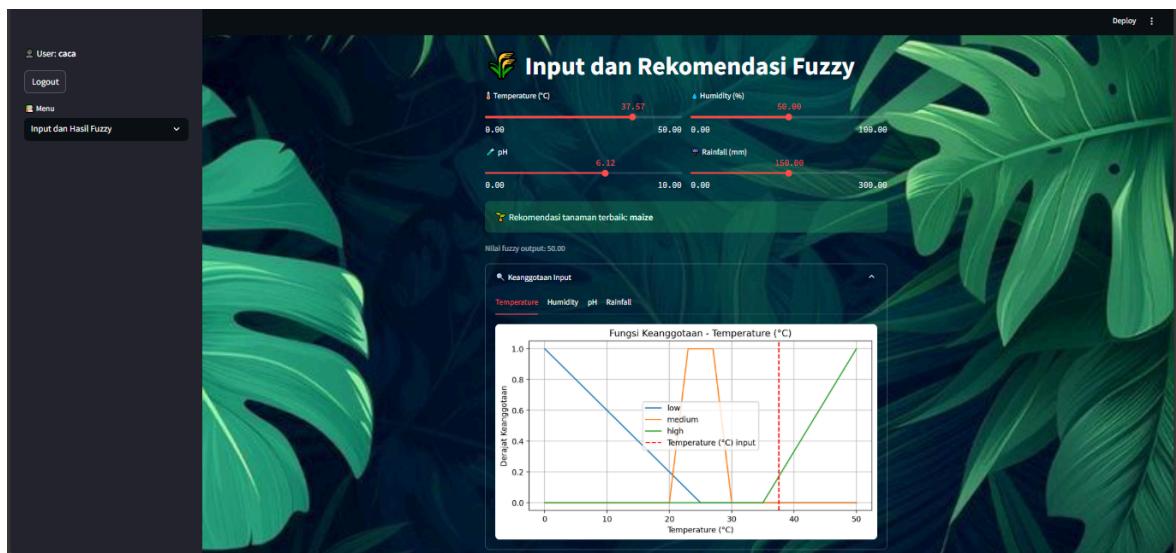
Gambar 2.3.3 Input dan Hasil Fuzzy

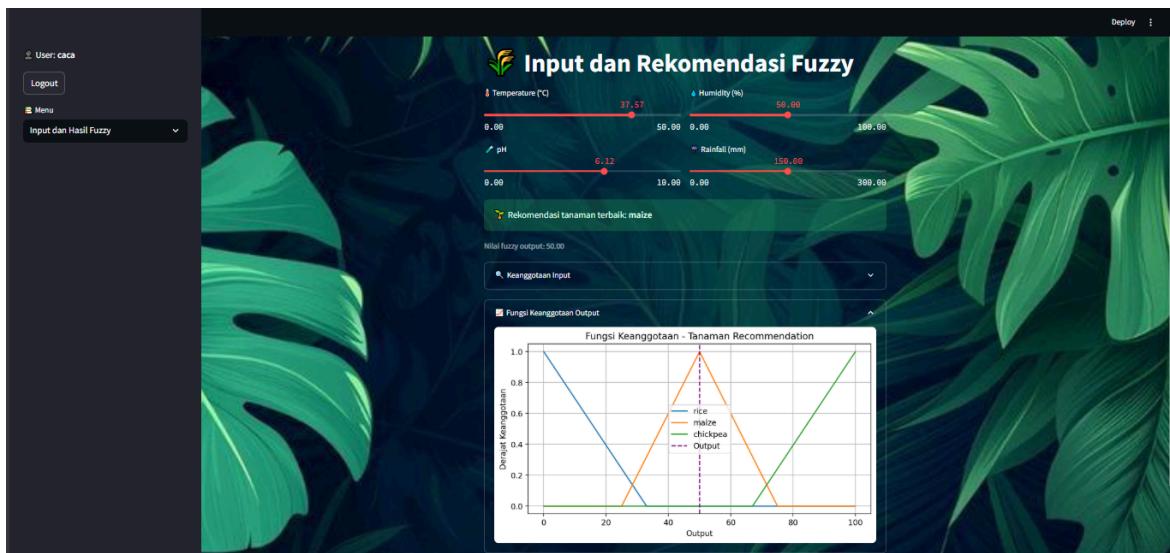
Setelah aturan-aturan dievaluasi, sistem akan melakukan **defuzzifikasi** menggunakan metode **centroid** untuk menghasilkan nilai numerik output. Nilai ini kemudian direkomendasikan menjadi jenis tanaman terbaik untuk ditanam, yaitu salah satu dari **rice (padi)**, **maize (jagung)**, atau **chickpea (kacang arab)**, tergantung dari rentang nilai output fuzzy. Selain hasil rekomendasi, program juga menampilkan grafik-grafik visualisasi fungsi keanggotaan untuk masing-masing input dan output, lengkap dengan garis vertikal penanda nilai input pengguna.



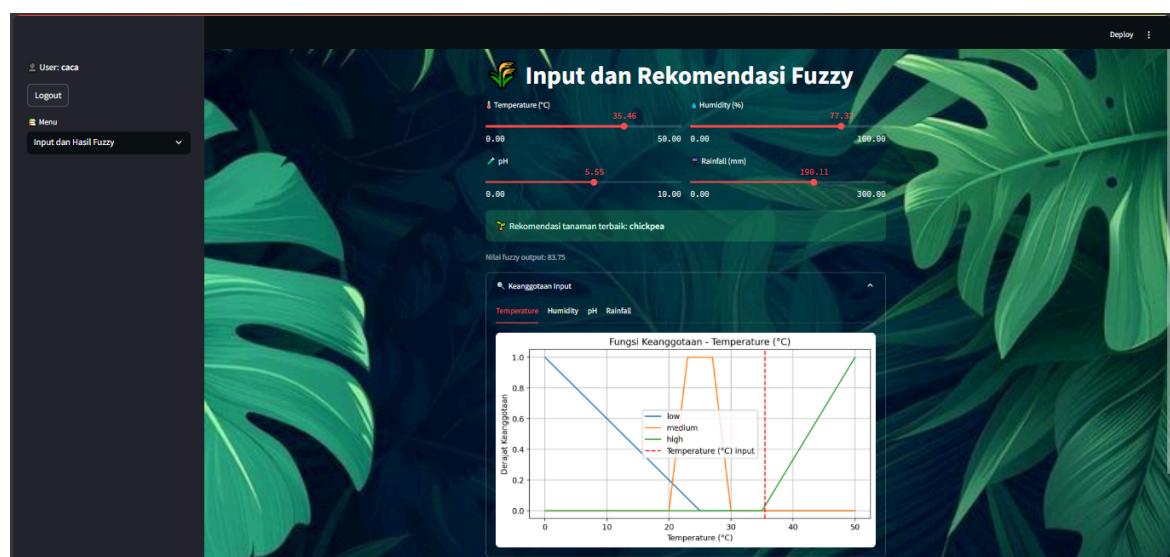


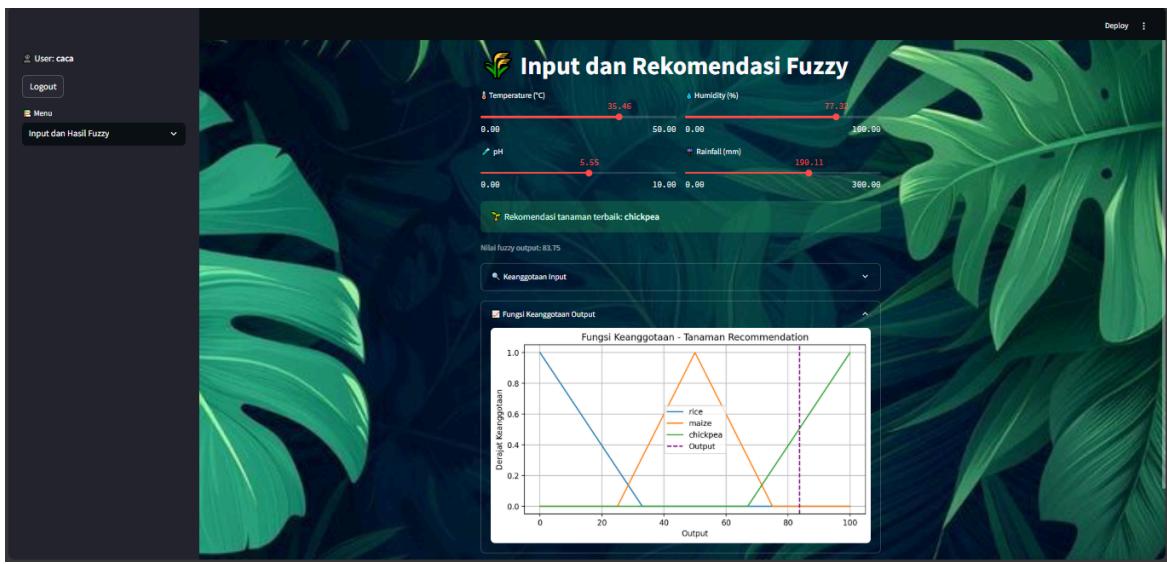
Gambar 2.3.4 defuzzifikasi Rice (Padi)





Gambar 2.3.4 defuzzifikasi maize (jagung)





Gambar 2.3.4 defuzzifikasi chickpea (kacang arab)

Pengguna juga dapat menekan tombol **Logout** yang tersedia di sidebar untuk keluar dari sistem dan kembali ke halaman login.

BAB III
JADWAL PENGERJAAN DAN PEMBAGIAN TUGAS

3.1 Jadwal Pengerjaan

Tabel 3.1 Jadwal Pengerjaan

No	Kegiatan	2025			
		Mei			
		1	2	3	4
1	Menentukan tema dan dataset				
2	Pembuatan Program				
3	Pembuatan Laporan				

3.2 Pembagian Tugas

Tabel 3.2 Pembagian Tugas

No	Kegiatan	Penanggung Jawab
1	Menentukan tema dan dataset	Atiqa Desyta Zahrani Salsabilla Rizky Setyabudi
2	Pembuatan Program	Atiqa Desyta Zahrani Salsabilla Rizky Setyabudi
3	Pembuatan Laporan	Atiqa Desyta Zahrani Salsabilla Rizky Setyabudi

BAB IV

KESIMPULAN DAN SARAN

4.1 Kesimpulan

Sistem rekomendasi tanaman berbasis fuzzy logic yang dikembangkan berhasil memberikan rekomendasi tanaman berdasarkan karakteristik lingkungan. Sistem ini telah diuji menggunakan dataset nyata dan memberikan hasil yang cukup akurat serta fleksibel. Dengan antarmuka berbasis web menggunakan Streamlit, aplikasi ini mudah digunakan dan interaktif.

4.2 Saran

Pengembangan lebih lanjut dapat mencakup penambahan jenis tanaman, validasi menggunakan data lapangan, serta pengembangan aplikasi ke versi mobile.

DAFTAR PUSTAKA

Dataset

Atharva Ingle. *Crop Recommendation Dataset*. Diakses dari:

<https://www.kaggle.com/datasets/atharvaingle/crop-recommendation-dataset>

Buku

1. Buku modul Praktikum Sistem Cerdas dan Pendukung Keputusan.