

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



DISUSUN OLEH:

NAMA/NIM : PRISKA NATALIA SEMBIRING (123230055)
DEWA SETYA BAGUS (123230063)
KELAS/PLUG : IF-A
NAMA ASISTEN : WINDY CLAUDIA NAPITULU(123220029)
M VIDYATAMA ADHI KRISNA(123220038)

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

2025

HALAMAN PENGESAHAN

LAPORAN AKHIR

Disusun oleh :

Priska Natalia Sembiring

123230055

Dewa Setya Bagus

123230063

Telah Diperiksa dan Disetujui oleh Asisten Praktikum Implementasi Struktur Data

Pada Tanggal :

Menyetujui,

Asisten Praktikum 1

Asisten Praktikum 2

Windy Claudia Napitupulu

NIM.123220029

Made Vidyatma Adhi Krisna

NIM. 123220038

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa yang senantiasa mencurahkan rahmat dan hidayah-Nya sehingga saya dapat menyelesaikan praktikum Sistem Cerdas dan Pendukung Keputusan serta laporan proyek akhir praktikum yang berjudul Rekomendasi Kuliner Berdasarkan Selera. 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, 1 Juni 2025

Penyusun

DAFTAR ISI

HALAMAN PENGESAHAN	i
LAPORAN AKHIR.....	i
KATA PENGANTAR.....	ii
DAFTAR ISI	iii
BAB I.....	2
PENDAHULUAN	2
1.1 Latar Belakang Masalah	2
1.2 Tujuan Proyek Akhir	2
1.3 Manfaat Proyek Akhir	3
BAB II	4
PEMBAHASAN.....	4
2.1 Dasar Teori	4
2.2 Deskripsi Umum Proyek Akhir	4
2.3 Inti Pembahasan.....	5
2.3.1 Data Awal (Before Preprocessing)	5
2.3.2 Tahap Preprocessing.....	5
2.3.3 Implementasi Algoritma (Weighted Product)	8
2.3.4 Tampilan GUI dengan Streamlit.....	11
BAB III	13
JADWAL Pengerjaan Tugas dan Pembagian Tugas	13
BAB IV	15
KESIMPULAN DAN SARAN	15
4.1 Kesimpulan.....	16
4.2 Saran	16
DAFTAR PUSTAKA.....	17

PEMILIHAN KULINER BERDASARKAN SELERA

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Dalam kehidupan sehari-hari, makanan tidak hanya menjadi kebutuhan dasar, tetapi juga bagian dari gaya hidup dan pengalaman sosial. Banyak orang ingin menikmati makanan yang sesuai dengan selera pribadi mereka, baik berdasarkan rasa favorit seperti manis, pedas, atau gurih, maupun berdasarkan faktor-faktor lain seperti harga, lokasi, dan popularitas. Namun, dengan begitu banyaknya pilihan makanan yang tersedia, menentukan pilihan yang paling cocok bisa menjadi hal yang membingungkan dan memakan waktu.

Melihat kebutuhan tersebut, dibutuhkan suatu sistem cerdas yang mampu memberikan pilihan kuliner secara personal berdasarkan preferensi pengguna. Program ini akan mengintegrasikan berbagai parameter seperti jenis rasa (asam, manis, pedas, gurih, pahit, asin), popularitas, harga, asal kuliner, serta informasi pendukung seperti lokasi, waktu makan, rating, kategori, waktu persiapan, skor WP (Weighted Product), dan tingkat ketersediaan.

1.2 Tujuan Proyek Akhir

Proyek akhir ini memiliki beberapa tujuan utama, yaitu:

1. Merancang dan membangun program rekomendasi kuliner yang berbasis preferensi pengguna.
2. Menerapkan pemilihan berdasarkan berbagai parameter seperti rasa, harga, lokasi, popularitas, dan asal makanan.
3. Menyediakan informasi lengkap dan dapat mengurutkan makanan yang ada sesuai yang diinginkan oleh pengguna.
4. Mengintegrasikan metode pengambilan keputusan seperti Weighted Product (WP) untuk menghasilkan skor akurasi rekomendasi yang lebih baik.
5. Mengembangkan sistem yang responsif dan interaktif, serta mudah digunakan oleh pengguna umum.

1.3 Manfaat Proyek Akhir

Dengan adanya proyek akhir ini, pengguna diharapkan dapat memperoleh kemudahan dalam menemukan kuliner yang sesuai dengan selera mereka secara cepat dan efisien. Sistem rekomendasi yang dikembangkan memungkinkan pengguna untuk menyaring pilihan makanan berdasarkan berbagai kriteria seperti rasa, harga, lokasi, dan popularitas. Selain itu, pengguna juga dapat melihat informasi pendukung seperti rating, waktu persiapan, dan tingkat ketersediaan, sehingga dapat membantu dalam pengambilan keputusan yang lebih tepat sebelum menentukan pilihan makanan.

Bagi pengembang dan akademisi, proyek ini dapat menjadi sarana pembelajaran yang efektif dalam mengembangkan sistem berbasis preferensi pengguna dan penerapan metode pengambilan keputusan seperti Weighted Product (WP). Proyek ini juga dapat dijadikan sebagai referensi atau dasar pengembangan untuk system rekomendasi lainnya di berbagai bidang, seperti pariwisata, pendidikan, atau e-commerce, yang memerlukan pemrosesan data multi-kriteria dan pendekatan yang bersifat personal.

BAB II

PEMBAHASAN

2.1 Dasar Teori

Dalam pengambilan keputusan multikriteria, dibutuhkan metode yang mampu mengevaluasi sejumlah alternatif berdasarkan berbagai atribut atau kriteria yang memiliki bobot kepentingan yang berbeda. Salah satu metode yang sering digunakan adalah Metode Weighted Product (WP).

Metode WP bekerja dengan cara mengalikan nilai setiap atribut pada suatu alternatif yang telah diberi bobot pangkat sesuai dengan tingkat kepentingannya. Artinya, setiap kriteria tidak hanya memengaruhi hasil akhir, tetapi juga dipertimbangkan secara proporsional terhadap prioritas yang diberikan. Atribut yang bersifat benefit (semakin tinggi nilainya semakin baik), seperti rasa atau rating, akan diproses berbeda dari atribut cost (semakin rendah semakin baik), seperti harga atau waktu persiapan. Dengan pendekatan ini, WP memungkinkan kita untuk menghitung nilai total dari setiap alternatif, sehingga alternatif dengan nilai tertinggi dianggap sebagai pilihan paling optimal.

Metode ini telah terbukti efektif dalam berbagai bidang seperti seleksi kandidat, evaluasi proyek, penentuan produk terbaik, hingga sistem rekomendasi, karena mempertimbangkan interaksi antar kriteria secara proporsional dan realistis. Dalam konteks proyek ini, WP digunakan untuk memberikan rekomendasi kuliner dengan mempertimbangkan berbagai aspek seperti selera rasa, harga, lokasi, waktu makan, tingkat ketersediaan, hingga popularitas dan rating. Setiap atribut akan diolah dalam sistem perhitungan WP sehingga menghasilkan skor akhir (*WP Score*) yang menjadi dasar urutan rekomendasi makanan.

Dengan demikian, pendekatan WP dalam proyek ini bukan hanya sebuah pilihan teknis, tetapi juga merupakan konsep yang telah teruji dan terimplementasi luas dalam berbagai skenario pengambilan keputusan nyata, termasuk pada sistem pendukung keputusan (*Decision Support System*).

2.2 Deskripsi Umum Proyek Akhir

Proyek akhir ini bertujuan untuk merancang dan mengembangkan sebuah sistem rekomendasi kuliner yang dapat membantu pengguna dalam menentukan pilihan makanan yang paling sesuai dengan preferensi mereka. Sistem ini akan bekerja dengan mempertimbangkan berbagai kriteria selera, seperti rasa (manis, asam, asin, gurih, pedas, pahit), harga, asal kuliner, lokasi, popularitas, waktu makan, waktu persiapan, ketersediaan, dan rating.

Pengguna akan diberikan antarmuka yang memungkinkan mereka memilih preferensi mereka terhadap masing-masing kriteria. Setelah data preferensi pengguna dikumpulkan, sistem akan menggunakan metode Weighted Product (WP) untuk mengolah data alternatif makanan yang tersedia dan memberikan hasil rekomendasi berupa urutan makanan dengan skor tertinggi. WP digunakan karena kemampuannya dalam menangani banyak kriteria secara proporsional dan menghasilkan peringkat yang adil berdasarkan bobot setiap kriteria.

Setiap rekomendasi yang ditampilkan akan dilengkapi informasi detail yang relevan dan mudah dipahami. Dengan demikian, proyek ini tidak hanya menawarkan fitur rekomendasi, tetapi juga memberikan pengalaman pengguna yang informatif, personal, dan efisien dalam menentukan pilihan kuliner terbaik sesuai kebutuhan mereka.

2.3 Inti Pembahasan

2.3.1 Data Awal (Before Preprocessing)

```

1  Nama,Kategori,Harga,Waktu_Makan,Lokasi,Deskripsi,Rating,Waktu_Persiapan,Popularitas,Ketersediaan
2  Jus Pir,Manis,20000,Semua,Riau,Jus buah pir manis dan kaya serat.,4,5,750,88
3  Jus Anggur,Manis,25000,Semua,Riau,Jus buah anggur manis dan menyegarkan.,4,2,5,800,89
4  Jus Belimbing,Manis,18000,Semua,Riau,Jus buah belimbing manis dan sedikit asam.,3,9,5,700,85
5  Jus Sirsak,Manis,22000,Semua,Riau,Jus buah sirsak manis asam dan creamy.,4,3,5,820,90
6  Jus Sirsak Nata de Coco,Manis,25000,Semua,Riau,Jus sirsak dengan tambahan nata de coco,4,4,8,850,91
7  Es Cincau Hitam,Manis,12000,Semua,Riau,Minuman segar dari cincau hitam dengan santan dan gula merah.,4,5,780,89
8  Es Kacang Hijau,Manis,15000,Semua,Riau,Minuman segar dari kacang hijau rebus santan dan gula.,4,1,10,800,90
9  Es Dawet,Manis,10000,Semua,Jawa Tengah,Minuman segar dari cendol hijau snatan dan gula merah,4,2,8,850,92
10 Es Cendol,Manis,10000,Semua,Jawa Barat,Minuman segar dengan cendol,4,3,8,870,93
11 Es Sarang Burung,Manis,18000,Semua,Riau,Es serut dengan agar-agar yang dibentuk seperti sarang burung,4,1,10,750,88
12 Es Teler Jumbo,Manis,30000,Semua,Jakarta,Es teler dengan porsi lebih besar dan isian melimpah.,4,7,10,960,98
13 Es Oyen,Manis,20000,Semua,Jawa Barat,Mirip es campur dengan alpukat,4,4,8,880,92
14 Es Doger,Manis,15000,Semua,Jawa Barat,Es krim tradisional dari santan dengan roti tawar,4,2,10,800,90
15 Es Goyobod,Manis,15000,Semua,Jawa Barat,Es campur dengan agar-agar,4,1,10,750,88
16 Es Podeng,Manis,18000,Semua,Jakarta,Es puter dengan roti tawar,4,3,10,820,90
17 Es Shanghai,Manis,20000,Semua,Riau,Es campur dengan aneka buah,4,2,10,780,89
18 Es Gempol Pleret,Manis,15000,Semua,Jawa Tengah,Minuman segar dari gempol (bulatan nasi) dan pleret (cendol pink),4,10,700,85
19 Es Campur Buah,Manis,25000,Semua,Riau,Es campur dengan dominasi buah-buahan segar.,4,5,8,900,94
20 Es Campur Cincau,Manis,18000,Semua,Riau,Es campur dengan banyak cincau hitam dan hijau.,4,3,8,850,91
21 Es Timun Suri,Manis,15000,Semua,Riau,Minuman segar dari timun suri manis dan sangat menyegarkan.,4,1,5,780,89
22 Wedang Jahe Susu,Manis,15000,Semua,Riau,Minuman jahe hangat dengan tambahan susu creamy dan pedas.,4,4,5,880,92
23 Wedang Ronde Kuah Kacang,Manis,18000,Camilan,DIY Yogyakarta,Ronde dengan kuah kacang hangat,4,3,15,800,88
24 Sekoteng,Manis,15000,Camilan,Riau,Minuman jahe hangat dengan roti tawar,4,2,10,750,87

```

Gambar 2.1

2.3.2 Tahap Preprocessing

Mengimpor library pandas dan numpy untuk manipulasi data.

```

import pandas as pd
import numpy as np

```

Tabel 2.1

Membaca file CSV ke DataFrame dan menampilkan isi data.

```

MoodRasaDataFleksibel_df = pd.read_csv("MoodRasaDataFleksibel.csv")
MoodRasaDataFleksibel_df

```

Tabel 2.2

Menampilkan informasi tipe data dan jumlah data tiap kolom.

```
MoodRasaDataFleksibel_df.info()
```

Tabel 2.3

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 556 entries, 0 to 555
Data columns (total 10 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Nama             556 non-null   object
1   Kategori         556 non-null   object
2   Harga            556 non-null   int64
3   Waktu_Makan      556 non-null   object
4   Lokasi           556 non-null   object
5   Deskripsi         556 non-null   object
6   Rating           554 non-null   float64
7   Waktu_Persiapan  542 non-null   float64
8   Popularitas      555 non-null   float64
9   Ketersediaan     554 non-null   float64
dtypes: float64(4), int64(1), object(5)
memory usage: 43.6+ KB
```

Gambar 2.2

Mengecek Missing Value.

```
display(MoodRasaDataFleksibel_df.isnull().sum())
rows_with_nan =
MoodRasaDataFleksibel_df[MoodRasaDataFleksibel_df.isnull().any(axis
=1)]
display(rows_with_nan)
display(MoodRasaDataFleksibel_df.isnull())
```

Tabel 2.4

Nama	0
Kategori	0
Harga	0
Waktu_Makan	0
Lokasi	0
Deskripsi	0
Rating	2
Waktu_Persiapan	14
Popularitas	1
Ketersediaan	2
dtype: int64	

	Nama	Kategori	Harga	Waktu_Makan	Lokasi	Deskripsi	Rating	Waktu_Persiapan	Popularitas	Ketersediaan
128	Sambal Lingkung	Pedas	15000	Semua	Bangka Belitung	Sambal khas Bangka Belitung dari ikan cincang ...	15.0	700.0	80.0	NaN
137	Rusip Ikan	Asin	15000	Semua	Bangka Belitung	Fermentasi ikan kecil	3.8	NaN	650.0	75.0
234	Peuyeum Sampeu	Manis	10000	Camilan	Jawa Barat	Tapai singkong	4.0	NaN	700.0	85.0
235	Dodol Garut	Manis	25000	Camilan	Jawa Barat	Makanan manis kenyal dari beras ...	4.5	NaN	900.0	95.0

Gambar 2.3

Mengisi nilai kosong pada kolom numerik dengan rata-rata kolom tersebut.

```
MoodRasaDataFleksibel_df['Rating'] =  
MoodRasaDataFleksibel_df['Rating'].fillna(MoodRasaDataFleksibel_df[  
'Rating'].mean())  
MoodRasaDataFleksibel_df['Waktu_Persiapan'] =  
MoodRasaDataFleksibel_df['Waktu_Persiapan'].fillna(MoodRasaDataFlek  
sibel_df['Waktu_Persiapan'].mean())  
MoodRasaDataFleksibel_df['Popularitas'] =  
MoodRasaDataFleksibel_df['Popularitas'].fillna(MoodRasaDataFleksibe  
l_df['Popularitas'].mean())  
MoodRasaDataFleksibel_df['Ketersediaan'] =  
MoodRasaDataFleksibel_df['Ketersediaan'].fillna(MoodRasaDataFleksib  
el_df['Ketersediaan'].mean())  
MoodRasaDataFleksibel_df
```

Tabel 2.5

Memastikan apakah masih ada nilai kosong setelah proses pengisian.

```
display(MoodRasaDataFleksibel_df.isnull().sum())  
rows_with_nan =  
MoodRasaDataFleksibel_df[MoodRasaDataFleksibel_df.isnull().any(axis  
=1)]  
display(rows_with_nan)
```

Tabel 2.6

```
Nama          0  
Kategori      0  
Harga         0  
Waktu_Makan   0  
Lokasi        0  
Deskripsi     0  
Rating        0  
Waktu_Persiapan 0  
Popularitas   0  
Ketersediaan  0  
dtype: int64
```

Gambar 2.4

Menghapus baris yang masih memiliki nilai kosong.

```
MoodRasaDataFleksibel_df.dropna(inplace=True)
```

```
display(MoodRasaDataFleksibel_df.isnull().sum())
```

Tabel 2.7

Menduplikasi DataFrame bersih ke variabel baru dan menampilkannya.

```
MoodRasaDataFleksibel_df_cleaned = MoodRasaDataFleksibel_df
MoodRasaDataFleksibel_df_cleaned
```

Tabel 2.8

Menduplikasi DataFrame bersih ke variabel baru dan menampilkannya.

```
MoodRasaDataFleksibel_df_cleaned = MoodRasaDataFleksibel_df
MoodRasaDataFleksibel_df_cleaned
```

Tabel 2.9

Menyimpan ke File CSV Baru.

```
import os

os.makedirs("datasets", exist_ok=True)
MoodRasaDataFleksibel_df_cleaned.to_csv("datasets/MoodRasaDataFleksibel_cleaned.csv", index=False)
```

Tabel 2.10

2.3.3 Implementasi Algoritma (Weighted Product)

WP menghitung skor setiap alternatif (misal: makanan) dengan mengalikan setiap nilai kriteria yang sudah dinormalisasi dan dipangkatkan dengan bobotnya:

$$S_i = \prod_{j=1}^n X_{ij}^{W_j}$$

Gambar 2.5

\

Pemetaan Kriteria: Benefit & Cost. Benefit: Semakin besar semakin baik (dibagi nilai maksimum) Cost: Semakin kecil semakin baik (nilai minimum dibagi nilai).

```
total_weight = rating_weight + harga_weight + waktu_weight +
popularitas_weight + ketersediaan_weight
if total_weight > 0:
    custom_weights = {
        'Rating': rating_weight / total_weight,
        'Harga': harga_weight / total_weight,
        'Waktu_Persiapan': waktu_weight / total_weight,
        'Popularitas': popularitas_weight / total_weight,
        'Ketersediaan': ketersediaan_weight / total_weight
    }
else:
    custom_weights = default_weights
    st.markdown(f"**Total Weight:** {total_weight:.2f}
(normalized to 1.0)")
```

Tabel 2.11

Bobot bisa diubah user, namun default-nya seperti tabel di atas.

```
# Default weights for WP calculation
default_weights = {
    'Rating': 0.25,
    'Harga': 0.20,
    'Waktu_Persiapan': 0.15,
    'Popularitas': 0.25,
    'Ketersediaan': 0.15
}
```

Tabel 2.12

Proses Perhitungan WP dari Data Bersih

1. Normalisasi setiap kriteria sesuai tipe (benefit/cost).
2. Pangkatkan hasil normalisasi dengan bobot masing-masing kriteria.
3. Kalikan semua hasil pangkat untuk tiap alternatif.
4. Skalikan hasil (dikali 100) agar lebih mudah dibaca.

```
def calculate_wp_score(data, weights=None):
    if weights is None:
        weights = default_weight
    df = data.copy()
    # Normalisasi
    normalized_rating = df['Rating'] / df['Rating'].max()
    normalized_harga = df['Harga'].min() / df['Harga']
    normalized_waktu = df['Waktu_Persiapan'].min() /
df['Waktu_Persiapan']
    normalized_popularitas = df['Popularitas'] /
df['Popularitas'].max()
    normalized_ketersediaan = df['Ketersediaan'] /
df['Ketersediaan'].max()

    # Perhitungan WP
    wp_score = (
        (normalized_rating ** weights['Rating']) *
        (normalized_harga ** weights['Harga']) *
        (normalized_waktu ** weights['Waktu_Persiapan']) *
        (normalized_popularitas ** weights['Popularitas']) *
        (normalized_ketersediaan ** weights['Ketersediaan'])
    )

    df['WP_Score'] = (wp_score * 100).round(2)
    return df
```

Tabel 2.13

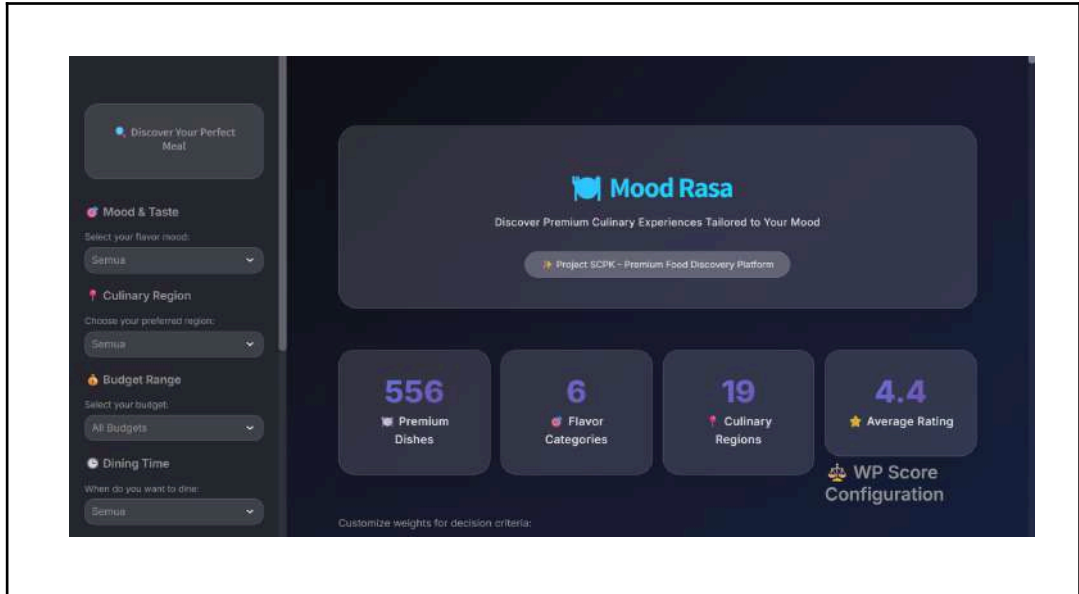
Ranking Final Hasil WP

```
import matplotlib.pyplot as plt
top_n = data.sort_values("WP_Score", ascending=False).head(10)
plt.figure(figsize=(10,5))
plt.barh(top_n['Nama'], top_n['WP_Score'], color='skyblue')
plt.xlabel('WP Score')
plt.title('Top 10 Makanan Berdasarkan WP Score')
plt.gca().invert_yaxis()
plt.show()
```

Tabel 2.14

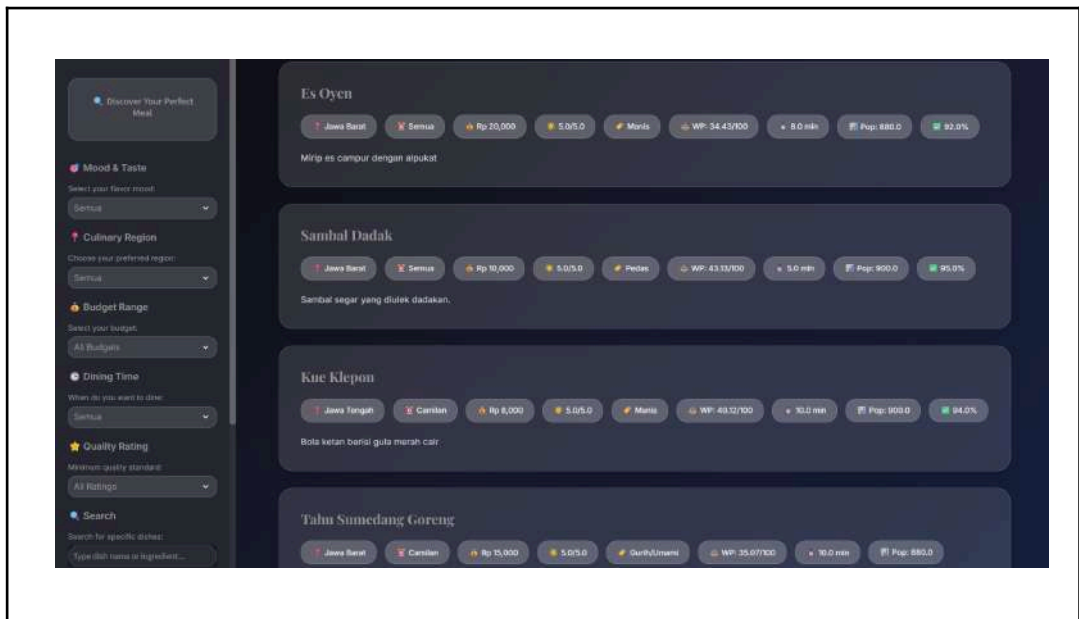
2.3.4 Tampilan GUI dengan Streamlit

Tampilan awal.



Gambar 2.6

Tampilan data bersih.



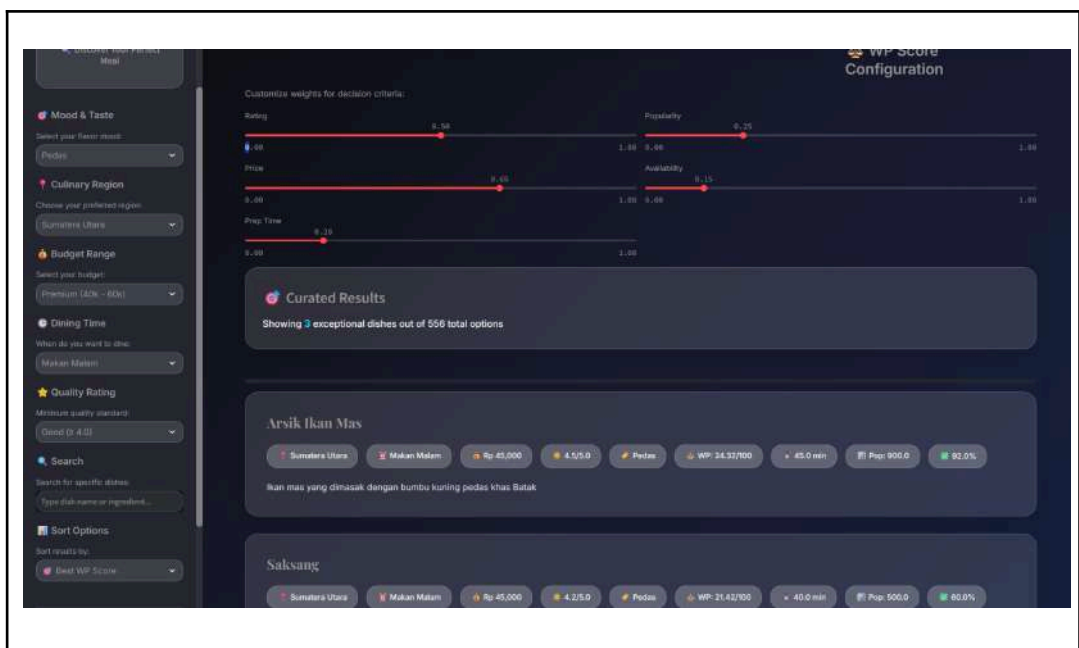
Gambar 2.7

Fitur random pick.



Gambar 2.8

Hasil pilihan yang diberikan.



Gambar 2.9

BAB III

JADWAL Pengerjaan TUGAS DAN PEMBAGIAN TUGAS

3.1 Jadwal Pengerjaan

Proyek akhir ini dikerjakan selama bulan Mei, dengan pembagian tahapan kegiatan mulai dari pengkajian masalah, perancangan sistem, pembuatan program, pencarian data, hingga preprocessing data. Setiap kegiatan disesuaikan dengan waktu yang tersedia dan disusun agar pengerjaan dapat dilakukan secara bertahap dan efisien.

Tabel 3.1 Tabel Jadwal Pengerjaan

No.	Kegiatan	Mei			April			
1.	Mengkaji Masalah	26						
2.	Perancangan Sistem Program	26						
3.	Pembuatan Program	23	25	31				
4.	Membuat dan Mencari Data	26			1			
...	Data Preprocessing				1			

3.2 Pembagian Tugas

Pengerjaan proyek dilakukan secara berkelompok dan pembagian tugas disesuaikan dengan kemampuan dan peran masing-masing anggota. Berikut ini adalah rincian pembagian tugas untuk tiap kegiatan proyek.

Tabel 3.2 Tabel Pembagian Tugas

No	Kegiatan	Penanggung Jawab
1.	Mengkaji Masalah	Dewa, Priska
2.	Perancangan Sistem Program	Dewa, Priska
3.	Membuat Program	Dewa
4.	Membuat dan Mencari Data	Dewa, Priska
5.	Data Preprocessing	Priska

BAB IV

KESIMPULAN DAN SARAN

4.1 Kesimpulan

Proyek akhir ini berhasil menghasilkan sebuah sistem rekomendasi kuliner berbasis preferensi pengguna dengan menggunakan metode Weighted Product (WP). Sistem ini mampu menyaring dan menampilkan rekomendasi makanan yang sesuai dengan selera pengguna berdasarkan berbagai kriteria seperti rasa, harga, rating, popularitas, lokasi, waktu makan, dan ketersediaan. Proses perhitungan yang dilakukan secara otomatis dan objektif memberikan hasil yang akurat dan efisien dalam menentukan pilihan kuliner terbaik.

Metode Weighted Product terbukti efektif dalam menangani data multikriteria karena mempertimbangkan bobot masing-masing atribut dan perbedaan jenis atribut (benefit dan cost). Dengan adanya sistem ini, pengguna dapat terbantu dalam pengambilan keputusan tanpa perlu mencari makanan secara manual atau mencoba secara acak. Selain itu, sistem ini juga memiliki potensi untuk dikembangkan lebih lanjut sebagai bagian dari layanan pencarian atau direktori kuliner berbasis teknologi.

4.2 Saran

Berdasarkan hasil pengerjaan proyek akhir ini, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan selanjutnya:

1. Perluasan data kuliner yang digunakan dalam sistem, agar hasil rekomendasi menjadi lebih bervariasi dan sesuai dengan kondisi di lapangan.
2. Peningkatan tampilan antarmuka pengguna (UI) agar lebih menarik dan mudah dipahami, terutama bagi pengguna yang belum terbiasa dengan teknologi.
3. Uji coba sistem ke lebih banyak pengguna untuk melihat sejauh mana sistem benar-benar membantu dalam memberikan rekomendasi yang sesuai dengan selera.
4. Penambahan kriteria atau filter tambahan, seperti alergi makanan atau preferensi halal/non-halal, agar sistem lebih ramah terhadap kebutuhan pengguna yang beragam.

DAFTAR PUSTAKA

Google Fonts. (2023). *Inter & Playfair Display Font Families*. Diakses dari <https://fonts.google.com/>

Harris, C. R., Millman, K. J., van der Walt, S. J., et al. (2020). *Array Programming with NumPy*. *Nature*, 585, 357–362. <https://doi.org/10.1038/s41586-020-2649-2>

McKinney, W. (2010). *Data Structures for Statistical Computing in Python*. *Proceedings of the 9th Python in Science Conference*, 56–61.

Mozilla Developer Network. (2023). *CSS: Cascading Style Sheets*. Diakses dari <https://developer.mozilla.org/en-US/docs/Web/CSS>

Python Software Foundation. (2023). *Python 3 Documentation*. Diakses dari <https://docs.python.org/3/>

Streamlit Inc. (2019). *Streamlit: The Fastest Way to Build Data Apps*. Diakses dari <https://streamlit.io/>

Triantaphyllou, E. (2000). *Multi-criteria Decision Making Methods: A Comparative Study*. Springer Science & Business Media.

World Wide Web Consortium (W3C). (2021). *HTML5 Specification*. Diakses dari <https://html.spec.whatwg.org/>

Yoon, K. P., & Hwang, C. L. (1995). *Multiple Attribute Decision Making: An Introduction*. SAGE Publications.