

Sistem Pendukung Keputusan Penentuan Prioritas Produk untuk Kegiatan Promosi Menggunakan Metode Multi-Objective Optimization By Ratio Analysis (MOORA) di DRW SKINCARE Banjarmasin.

Aplikasi berbasis web yang dikembangkan menggunakan **Laravel 13**. README ini dibuat khusus untuk menjelaskan terkait sistem yang dibangun.

Sistem ini dibuat untuk membantu pengguna dalam melakukan penilaian produk berdasarkan beberapa kriteria, seperti jumlah penjualan, tingkat permintaan, stok produk, harga jual, serta kriteria lain yang dapat diatur melalui sistem. Hasil akhir dari proses ini berupa peringkat produk yang dapat dijadikan dasar dalam menentukan produk prioritas untuk kegiatan promosi.

1. Daftar Isi

- [Gambaran Umum](#)
 - [Kebutuhan Sistem](#)
 - [Instalasi Sistem](#)
 - [Konfigurasi Database](#)
 - [Menjalankan Sistem](#)
 - [Akun Default](#)
 - [Alur Pemakaian Sistem](#)
 - [Penjelasan Metode MOORA](#)
 - [Struktur File Terkait SPK](#)
 - [Troubleshooting](#)
 - [Perintah Ringkas Instalasi](#)
-

2. Gambaran Umum

Sistem ini digunakan untuk membantu proses pengambilan keputusan dalam menentukan prioritas produk promosi. Sebelum adanya sistem, proses penentuan produk prioritas dapat dilakukan secara manual berdasarkan pengalaman, pengamatan lapangan, diskusi tim, atau pertimbangan umum. Cara tersebut dapat menimbulkan perbedaan penilaian karena setiap produk memiliki nilai kriteria yang berbeda.

Dengan adanya sistem ini, proses penilaian dilakukan secara lebih terstruktur karena sistem menghitung setiap produk berdasarkan kriteria, bobot, dan tipe atribut yang telah ditentukan. Sistem kemudian menghasilkan nilai akhir dan urutan ranking produk berdasarkan metode MOORA.

Pada modul ini terdapat dua role pengguna utama:

Role	Keterangan
Admin	Mengakses dashboard, Mengelola kriteria, Mengelola data produk, proses perhitungan, melihat hasil perhitungan, dan mengelola riwayat perhitungan.

Role	Keterangan
Manajer	Mengakses dashboard, Mengelola kriteria, Memproses perhitungan, Melihat hasil perhitungan, dan Mengelola riwayat perhitungan.

3. Kebutuhan Sistem

Sebelum menjalankan sistem, pastikan perangkat sudah memiliki kebutuhan berikut:

Komponen	Versi yang Disarankan
PHP	>= 8.3
Composer	2.x
Node.js	>= 18
NPM	Mengikuti versi Node.js
Database	MySQL direkomendasikan
Browser	Google Chrome, Microsoft Edge, atau browser modern lain

Ekstensi PHP yang disarankan aktif:

- `pdo_mysql`
- `mbstring`
- `openssl`
- `tokenizer`
- `xml`
- `ctype`
- `json`
- `fileinfo`
- `zip`

4. Instalasi Sistem

4.1 Ekstrak atau Clone Project

Jika project berbentuk file ZIP, ekstrak terlebih dahulu. Setelah itu masuk ke folder project:

```
cd drw-analytics-main
```

Jika menggunakan Git, jalankan:

```
git clone <url-repository>
cd drw-analytics-main
```

4.2 Install Dependency Backend

Jalankan perintah berikut untuk menginstal dependency Laravel:

```
composer install
```

4.3 Install Dependency Frontend

Jalankan perintah berikut untuk menginstal dependency frontend:

```
npm install
```

4.4 Salin File Environment

Salin file `.env.example` menjadi `.env`:

```
cp .env.example .env
```

Untuk pengguna Windows Command Prompt, gunakan:

```
copy .env.example .env
```

4.5 Generate Application Key

Jalankan perintah berikut:

```
php artisan key:generate
```

Application key digunakan Laravel untuk kebutuhan enkripsi dan keamanan sistem.

5. Konfigurasi Database

5.1 Konfigurasi MySQL/MariaDB

Buka file `.env`, lalu sesuaikan bagian konfigurasi database seperti berikut:

```
APP_NAME="DRW Analytics"  
APP_ENV=local  
APP_DEBUG=true  
APP_URL=http://localhost:8000
```

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=drw_analytics
DB_USERNAME=root
DB_PASSWORD=
```

Sesuaikan **DB_USERNAME** dan **DB_PASSWORD** dengan konfigurasi database pada perangkat masing-masing.

5.2 Buat Database Kosong

Buat database baru dengan nama:

```
CREATE DATABASE drw_analytics;
```

Database dapat dibuat melalui phpMyAdmin, Laragon, XAMPP, MySQL Workbench, atau terminal MySQL.

5.3 Jalankan Migration

Setelah database dibuat dan file **.env** sudah disesuaikan, jalankan migration:

```
php artisan migrate
```

Perintah ini akan membuat tabel-tabel yang dibutuhkan oleh modul SPK, antara lain:

- **user**
- **produk**
- **kategori_produk**
- **kriteria**
- **nilai_produk**
- **input_permintaan**
- **perhitungan**
- **hasil_perhitungan**
- **detail_perhitungan**

Migration juga membuat akun default untuk role Admin dan Manajer.

5.4 Jalankan Seeder Kategori Produk

Untuk mengisi data kategori produk, jalankan:

```
php artisan db:seed --class=KategoriProdukSeeder
```

Catatan: Perintah tersebut digunakan untuk menjalankan seeder kategori produk saja.

5.5 Import Backup SQL Opsional

Project menyediakan file backup SQL:

```
spk_moora_backup.sql
```

File tersebut dapat diimport ke MySQL apabila ingin menggunakan struktur atau data contoh yang sudah tersedia. Jika menggunakan file backup, pastikan database tujuan sudah dibuat terlebih dahulu.

6. Menjalankan Sistem

6.1 Menjalankan Server Laravel

Jalankan perintah berikut:

```
php artisan serve
```

Server Laravel akan berjalan pada alamat:

```
http://localhost:8000
```

6.2 Menjalankan Vite untuk Frontend

Buka terminal baru, lalu jalankan:

```
npm run dev
```

Perintah ini digunakan agar asset CSS dan JavaScript dapat berjalan dalam mode development.

6.3 Menjalankan Semua Proses Sekaligus

Jika tersedia script development pada project, sistem juga dapat dijalankan dengan:

```
composer run dev
```

6.4 Build Asset Frontend

Jika sistem akan dijalankan tanpa mode development, build asset terlebih dahulu:

```
npm run build
```

Setelah server berjalan, buka browser dan akses:

```
http://localhost:8000
```

Sistem akan menampilkan halaman login.

7. Akun Default

Setelah migration berhasil dijalankan, sistem menyediakan akun default berikut:

Role	Username	Password
Admin	admin	admin123
Manajer	manajer	drw2025

Penting: Sistem login menggunakan **username** dan **password**, bukan email. Password default sebaiknya diganti apabila sistem digunakan pada lingkungan produksi.

8. Alur Pemakaian Sistem

8.1 Login ke Sistem

1. Buka browser dan akses <http://localhost:8000>.
2. Masukkan username dan password sesuai akun yang tersedia.
3. Setelah login berhasil, pengguna akan diarahkan ke halaman dashboard.

8.2 Dashboard

Menu: </dashboard>

Dashboard menampilkan ringkasan informasi yang berkaitan dengan data pada sistem. Halaman ini dapat diakses oleh Admin dan Manajer.

8.3 Kelola Kriteria

Menu: </kelola-kriteria>

Menu ini digunakan untuk mengatur kriteria yang menjadi dasar penilaian produk. Data yang dapat dikelola meliputi:

- Nama kriteria.
- Tipe atribut, yaitu **Benefit** atau **Cost**.
- Bobot kriteria dalam persen.

- Sumber data, yaitu **Excel** atau **Manual**.
- Nama kolom Excel apabila nilai kriteria diambil dari file Excel.

Ketentuan kriteria:

- Total bobot seluruh kriteria aktif harus berjumlah **100%**.
- Bobot antara satu kriteria dengan kriteria yang lain tidak diperbolehkan sama.
- Kriteria bertipe **Benefit** berarti semakin besar nilai kriteria, maka semakin baik.
- Kriteria bertipe **Cost** berarti semakin kecil nilai kriteria, maka semakin baik.
- Kriteria dengan sumber data **Excel** akan mengambil nilai dari data import produk.
- Kriteria dengan sumber data **Manual** akan diisi melalui menu Input Permintaan.

8.4 Kelola Data Produk

Menu: **/data-produk**

Menu ini hanya dapat diakses oleh Admin. Admin dapat melakukan beberapa aktivitas berikut:

- Menambah data produk secara manual.
- Mengedit data produk.
- Menghapus data produk.
- Mengimport data produk dari file Excel.
- Melihat preview data sebelum import disimpan ke database.

Status data produk terdiri dari:

Status	Keterangan
Lengkap	Semua nilai kriteria yang dibutuhkan sudah tersedia.
Belum Lengkap	Masih ada nilai kriteria yang belum tersedia.

Produk hanya dapat diproses dalam perhitungan apabila data kriteria yang dibutuhkan sudah lengkap.

8.5 Input Permintaan

Menu: **/input-permintaan**

Menu Input Permintaan digunakan untuk memilih produk yang akan dimasukkan ke proses perhitungan dan mengisi nilai kriteria manual. Ketentuannya:

- Nilai manual menggunakan skala **1–5**.
- Pemilihan produk dibatasi maksimal **5 produk per kategori**.
- Produk yang dipilih dan memiliki data lengkap akan masuk ke proses perhitungan.

8.6 Hitung SPK

Menu: **/hitung-spk**

Langkah penggunaan:

1. Masuk ke menu **Menghitung Prioritas**.

2. Isi nama periode data, misalnya **Promosi Juni 2026**.
3. Periksa bobot kriteria yang ditampilkan oleh sistem.
4. Jika diperlukan, bobot dapat diubah sementara untuk sesi perhitungan tersebut.
5. Klik tombol **Mulai Perhitungan**.
6. Sistem akan melakukan proses perhitungan menggunakan metode MOORA.
7. Sistem menampilkan hasil ranking produk berdasarkan nilai akhir Yi.

Syarat perhitungan:

- Total bobot kriteria harus **100%**.
- Bobot antara satu kriteria dengan kriteria yang lain tidak diperbolehkan sama.
- Minimal terdapat **2 produk** dengan data lengkap.
- Setiap produk harus memiliki nilai pada seluruh kriteria yang digunakan.

8.7 Hasil Perhitungan

Route:

```
/perhitungan/{id}/hasil
```

Halaman hasil perhitungan menampilkan:

- Periode data.
- Daftar produk yang dihitung.
- Ranking produk.
- Nilai akhir Yi.
- Matriks keputusan.
- Matriks normalisasi.
- Detail nilai per kriteria.

Produk dengan nilai Yi tertinggi menjadi produk dengan prioritas promosi paling tinggi.

8.8 Riwayat Perhitungan

Menu: **/riwayat**

Menu Riwayat Perhitungan menyimpan seluruh proses perhitungan yang pernah dilakukan. Pengguna dapat:

- Melihat daftar riwayat perhitungan.
- Membuka kembali detail hasil perhitungan.
- Menghapus riwayat perhitungan tertentu.

Riwayat menyimpan data perhitungan pada saat proses dilakukan, sehingga hasil lama tetap dapat dilihat meskipun data master berubah.

9. Penjelasan Metode MOORA

Metode **MOORA** digunakan untuk menyelesaikan masalah pengambilan keputusan multikriteria. Pada sistem ini, alternatif yang dinilai adalah produk skincare di DRW SKINCARE Banjarmasin, sedangkan kriterianya adalah aspek penilaian seperti penjualan, permintaan, stok, harga, dan kriteria lain yang ditentukan pengguna.

Tahapan perhitungan MOORA pada sistem adalah sebagai berikut:

9.1 Pembentukan Matriks Keputusan

Sistem membentuk matriks keputusan dari nilai setiap produk terhadap setiap kriteria. Setiap baris menunjukkan produk, sedangkan setiap kolom menunjukkan kriteria.

9.2 Normalisasi Matriks

Normalisasi dilakukan agar nilai setiap kriteria memiliki skala yang sebanding. Rumus normalisasi yang digunakan adalah:

$$X_{ij}^* = X_{ij} / \sqrt{(\sum X_{ij}^2)}$$

Keterangan:

- X_{ij}^* = nilai normalisasi alternatif ke-i pada kriteria ke-j.
- X_{ij} = nilai awal alternatif ke-i pada kriteria ke-j.
- $\sqrt{(\sum X_{ij}^2)}$ = akar jumlah kuadrat nilai pada kriteria ke-j.

9.3 Perkalian Bobot

Setelah normalisasi, sistem mengalikan nilai normalisasi dengan bobot masing-masing kriteria.

$$\text{Nilai terbobot} = \text{nilai normalisasi} \times \text{bobot kriteria}$$

9.4 Perhitungan Nilai Y_i

Nilai akhir atau nilai optimasi dihitung dengan cara mengurangi total nilai cost dari total nilai benefit:

$$Y_i = \text{total nilai benefit} - \text{total nilai cost}$$

Keterangan:

- Kriteria **Benefit** dijumlahkan karena semakin besar nilainya semakin baik.
- Kriteria **Cost** dikurangkan karena semakin kecil nilainya semakin baik.

9.5 Pemeringkatan Produk

Produk diurutkan berdasarkan nilai Y_i dari yang tertinggi hingga terendah. Produk dengan nilai Y_i tertinggi menjadi prioritas utama untuk dipromosikan.

Jika terdapat nilai Yi yang sama, sistem menggunakan aturan tambahan, yaitu:

1. Total benefit tertinggi.
2. Total cost terendah.
3. Nama produk secara alfabetis.

10. Struktur File Terkait SPK

Berikut struktur file utama yang berkaitan dengan modul SPK:

```
app/Http/Controllers/  
├─ AuthController.php  
├─ DashboardController.php  
├─ ProdukController.php  
├─ KriteriaController.php  
├─ InputPermintaanController.php  
└─ PerhitunganController.php  
  
app/Http/Middleware/  
└─ RoleMiddleware.php  
  
app/Models/  
├─ User.php  
├─ Produk.php  
├─ KategoriProduk.php  
├─ Kriteria.php  
├─ NilaiProduk.php  
├─ InputPermintaan.php  
├─ Perhitungan.php  
├─ HasilPerhitungan.php  
└─ DetailPerhitungan.php  
  
resources/views/  
├─ auth/  
│   └─ login.blade.php  
└─ spk/  
    ├─ dashboard.blade.php  
    ├─ data-produk.blade.php  
    ├─ import-preview.blade.php  
    ├─ kelola-kriteria.blade.php  
    ├─ input-permintaan.blade.php  
    ├─ hitung-spk.blade.php  
    ├─ hasil-perhitungan.blade.php  
    └─ riwayat.blade.php  
  
database/migrations/  
├─ 2026_04_01_000001_create_user_table.php  
├─ 2026_04_01_000002_create_produk_table.php  
├─ 2026_04_01_000003_create_kriteria_table.php  
├─ 2026_04_01_000005_create_nilai_produk_table.php  
└─ 2026_04_01_000006_create_input_permintaan_table.php
```

```

├── 2026_04_01_000007_create_perhitungan_table.php
├── 2026_04_01_000008_create_hasil_perhitungan_table.php
├── 2026_04_01_000009_create_detail_perhitungan_table.php
├── 2026_05_14_131014_create_kategori_produk_and_add_id_kategori.php

database/seeders/
├── KategoriProdukSeeder.php

```

11. Troubleshooting

Masalah	Penyebab	Solusi
Halaman tidak bisa dibuka	Server Laravel belum berjalan	Jalankan <code>php artisan serve</code>
Tampilan CSS/JS tidak muncul	Vite belum berjalan atau asset belum dibuild	Jalankan <code>npm run dev</code> atau <code>npm run build</code>
Error koneksi database	Konfigurasi <code>.env</code> belum sesuai	Periksa <code>DB_CONNECTION</code> , <code>DB_DATABASE</code> , <code>DB_USERNAME</code> , dan <code>DB_PASSWORD</code>
Tabel belum tersedia	Migration belum dijalankan	Jalankan <code>php artisan migrate</code>
Login gagal	Username/password salah atau akun belum tersedia	Gunakan akun default setelah migration berhasil
Total bobot kriteria harus 100%	Jumlah bobot kriteria belum tepat	Atur kembali bobot pada menu Kelola Kriteria
Hitung SPK gagal	Produk lengkap kurang dari 2 atau nilai kriteria belum lengkap	Lengkapi data produk dan input permintaan terlebih dahulu
Import Excel gagal	Format kolom Excel tidak sesuai	Sesuaikan nama kolom Excel dengan nama kolom yang terdaftar pada kriteria
Seeder gagal	Seeder yang dijalankan tidak sesuai	Jalankan <code>php artisan db:seed --class=KategoriProdukSeeder</code>
Error <code>could not find driver</code>	Ekstensi database PHP belum aktif	Aktifkan <code>pdo_mysql</code> untuk MySQL atau <code>pdo_sqlite</code> untuk SQLite
Data lama masih terbaca setelah ubah <code>.env</code>	Cache konfigurasi belum dibersihkan	Jalankan <code>php artisan config:clear</code> dan <code>php artisan cache:clear</code>

12. Perintah Ringkas Instalasi

Berikut ringkasan perintah instalasi dari awal:

```
cd drw-analytics-main
composer install
npm install
cp .env.example .env
php artisan key:generate
php artisan migrate
php artisan db:seed --class=KategoriProdukSeeder
npm run build
php artisan serve
```

Untuk mode development dengan Vite:

```
php artisan serve
npm run dev
```

Atau gunakan:

```
composer run dev
```

13. Identitas Pengembang

Alya Yulhadi

NIM: 2207412026

Politeknik Negeri Jakarta

README ini dibuat sebagai panduan instalasi dan penggunaan **Sistem Pendukung Keputusan (SPK)** untuk penentuan prioritas produk promosi menggunakan metode **MOORA**.