

# **ANALISA SRS APLIKASI MyHerbalCare**

## **PEMROGRAMAN WEB**



Universitas  
**Esa Unggul**

**Disusun Oleh:**

**Nama:** Faiz Izaz Fathoni

**NIM:** 20230801081

**PRODI:** Teknik Informatika

Universitas Esa Unggul

2025

# Software Requirement Specification (SRS)

Judul Aplikasi: MyHerbalCare

## 1 Pendahuluan

### 1.1 Latar Belakang

Dalam beberapa tahun terakhir, terjadi peningkatan minat masyarakat terhadap pengobatan alternatif, khususnya penggunaan tanaman herbal. Namun, informasi mengenai tanaman herbal yang benar dan terverifikasi masih sulit diakses oleh masyarakat umum. MyHerbalCare dikembangkan sebagai solusi digital berbasis web untuk memberikan informasi tersebut secara akurat, cepat, dan mudah diakses. Aplikasi ini menyediakan database tanaman herbal, penyakit yang dapat disembuhkan, serta hubungan antar keduanya. Platform ini dapat digunakan oleh masyarakat umum sebagai referensi awal, dan oleh admin sebagai pengelola konten melalui dashboard khusus.

## 2 Ruang Lingkup

Berikut ruang lingkup Aplikasi MyHerbalCare, yaitu:

- Penyimpanan dan pengelolaan data tanaman herbal dan penyakit.
- Relasi many-to-many antara herbal dan penyakit, memungkinkan satu herbal menangani banyak penyakit, dan satu penyakit memiliki banyak solusi herbal.
- Sistem autentikasi dan manajemen akun admin serta user.
- Penyajian informasi herbal kepada pengguna melalui antarmuka publik.
- API publik untuk pengambilan data (dengan otentikasi API Key) oleh pihak ketiga atau pengembang lain.

## 3 Kebutuhan Fungsional

### **Admin:**

- Login
- Mengelola daftar Penyakit (CRUD)
- Mengelola daftar Herbal (CRUD)
- Menghubungkan data herbal dan penyakit (Many-to-Many)
- Mengelola akun user

### **User:**

- Login
- Menelusuri daftar tanaman herbal berdasarkan kategori atau nama penyakit.
- Melihat deskripsi tanaman herbal dan penyakit terkait.
- Mengakses informasi tanpa perlu melakukan input data
- Melihat hasil hubungan data herbal dan penyakit yang dibuat oleh admin pada tabel penyakit

#### 4 Kebutuhan Non-Fungsional

- Performa cepat dalam menampilkan dan mencari data.
- Responsif di perangkat desktop dan mobile.
- Aman dengan validasi input dan autentikasi admin.
- Tersedia dokumentasi API untuk pengembang.

#### 5 Spesifikasi Teknologi

Komponen	Teknologi
Framework	Laravel 12
Admin Panel	Filament v3
Database	MySQL
Bahasa	PHP (Backend), JavaScript (Interaktif), HTML & CSS (Frontend)
Swagger	L5-Swagger untuk dokumentasi API
Autentikasi	API Key (untuk API), Laravel Auth (untuk admin/user)

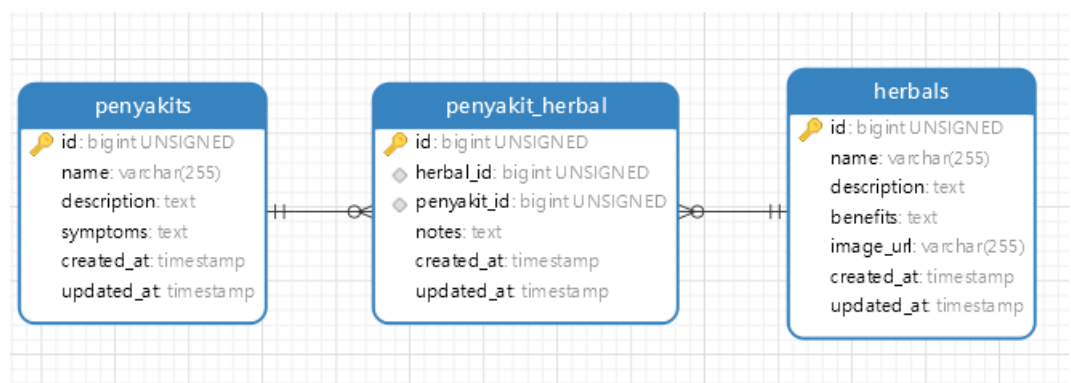
#### 6 Analisa Basis Data

##### 6.1 Tujuan

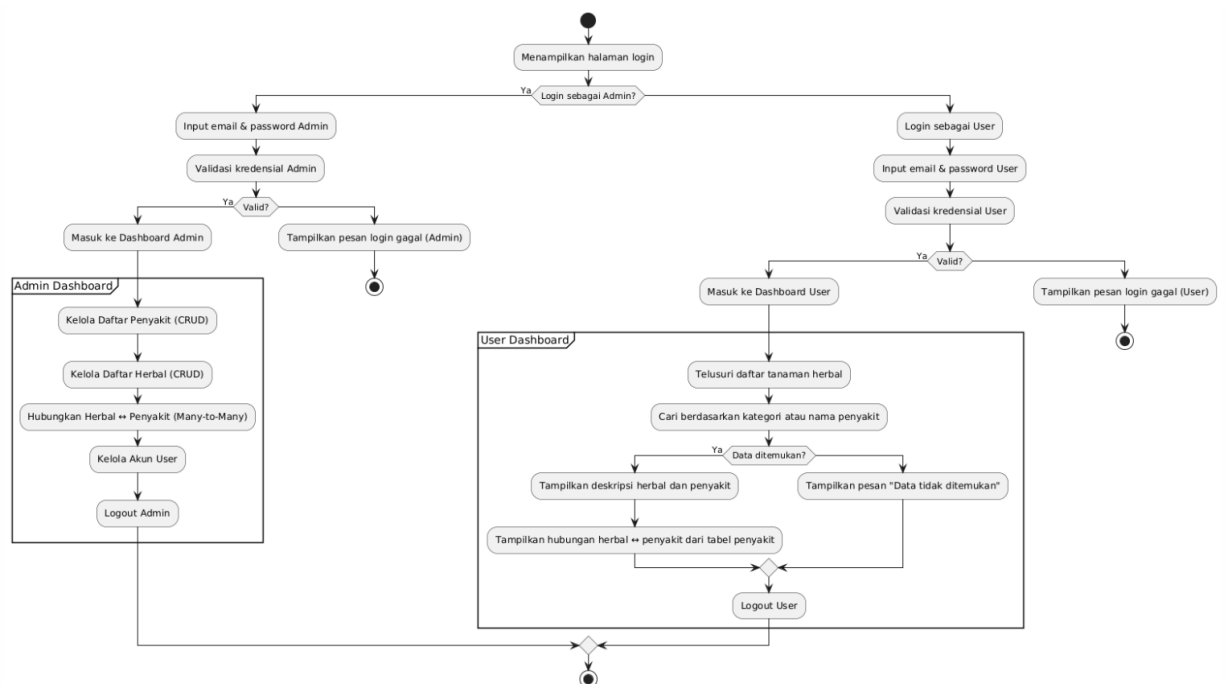
Sistem MyHerbalCare membutuhkan struktur basis data yang mampu menyimpan informasi tanaman herbal, penyakit, dan relasi di antara keduanya. Tujuan utama perancangan database adalah untuk memastikan integritas data, kemudahan akses, serta dukungan terhadap relasi many-to-many antara entitas herbal dan penyakit.

##### 6.2 Struktur Tabel Utama dan Relasinya

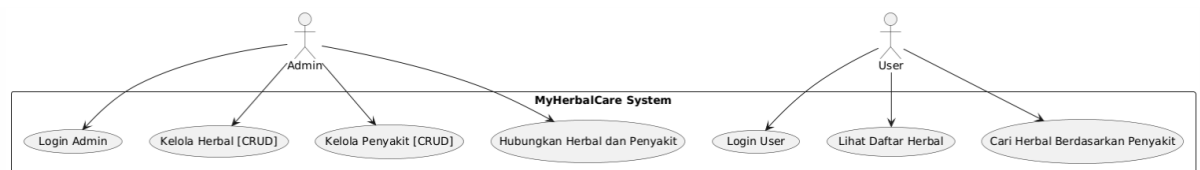
Pada proyek ini terdapat 3 tabel seperti herbals, penyakits, dan penyakit\_herbal. Berikut gambar relasi dari ketiga tabel tersebut:



## 7 Flowchart MyHerbalCare



## 8 Usecase Diagram – MyHerbalCare System



## 9 Api

```

9 Route::middleware([ApiKeyMiddleware::class])>group(function () {
10
11     // Herbal routes
12     Route::prefix('herbals')>group(function () {
13         Route::get('/', [HerbalController::class, 'index']);
14         Route::post('/', [HerbalController::class, 'store']);
15         Route::get('/{id}', [HerbalController::class, 'show']);
16         Route::put('/{id}', [HerbalController::class, 'update']);
17         Route::delete('/{id}', [HerbalController::class, 'destroy']);
18     });
19
20     // Penyakit routes
21     Route::prefix('penyakit')>group(function () {
22         Route::get('/', [PenyakitController::class, 'index']);
23         Route::post('/', [PenyakitController::class, 'store']);
24         Route::get('/{id}', [PenyakitController::class, 'show']);
25         Route::put('/{id}', [PenyakitController::class, 'update']);
26         Route::delete('/{id}', [PenyakitController::class, 'destroy']);
27     });
28 });
  
```

Pada gambar diatas, dapat terlihat contoh implementasi routing untuk sebuah API menggunakan framework Laravel. Routing ini dikelompokkan dalam sebuah middleware `ApiKeyMiddleware` yang berfungsi untuk memverifikasi API key sebelum mengizinkan akses ke endpoint yang dilindungi. Hal ini penting untuk keamanan, memastikan hanya klien yang memiliki API key yang valid yang dapat mengakses API tersebut.

Dalam kode tersebut, terdapat dua kelompok route utama: ``herbals`` dan ``penyakits``. Setiap kelompok route didefinisikan dengan prefix yang sesuai, yaitu ``/herbals`` dan ``/penyakits``. Setiap kelompok menyediakan operasi CRUD (Create, Read, Update, Delete) yang diimplementasikan melalui metode HTTP standar seperti GET, POST, PUT, dan DELETE. Misalnya, route GET ``/herbals`` digunakan untuk mengambil daftar herbal, sedangkan POST ``/herbals`` untuk menambahkan data baru. Demikian pula, route PUT dan DELETE digunakan untuk memperbarui dan menghapus data berdasarkan ID.

Namun, terdapat kesalahan sintaks dalam penulisan parameter ID, di mana seharusnya menggunakan kurung kurawal (``{id}``) sebagai placeholder, bukan tanda kurung siku (``[id]``). Selain itu, beberapa baris route tidak diakhiri dengan semikolon (``;`), yang dapat menyebabkan error saat dijalankan. Dengan perbaikan sintaks dan penambahan semikolon, kode ini akan berfungsi dengan baik untuk mengelola resource herbal dan penyakit secara terstruktur dan aman.