

LAMPIRAN

Lampiran 1 Surat Kesepakatan Bimbingan Skripsi

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan dibawah ini.

Pihak Pertama

Nama : Rama Oktabara
NIM : 21090008
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Ir. Ginanjar Wiro Sasmito, S.Kom., M.Kom.
Status : Dosen
NIDN : 0613028601
Jabatan Fungsional : Lektor Kepala
Pangkat/Golongan : Penata Tk. I / III-D

Pada hari ini Jumat tanggal 07 Maret 2025 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan syarat Pihak Pertama wajib melakukan bimbingan Skripsi minimal 8 kali kepada Pihak Kedua. Adapun waktu dan tempat pelaksanaan disepakati antar pihak.
Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 07 Maret 2025

Pihak Pertama



Rama Oktabara

Pihak Kedua



Ir. Ginanjar Wiro Sasmito, S.Kom.,
M.Kom.

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apiliani, S.T., M.Kom.
NIPY. 09.015.225

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan dibawah ini:

Pihak Pertama

Nama	Rama Oktabara
NIM	21090008
Program Studi	Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama	Dega Surono Wibowo, S.T., M.Kom
Status	Dosen
NIDN	0607108202
Jabatan Fungsional	Lektor
Pangkat/Golongan	Penata / III-c

Pada hari ini Jumat tanggal 07 Maret 2025 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat Pihak Kedua melakukan bimbingan satu kali dalam seminggu atau setidaknya tidaknya tiga bimbingan dalam satu bulan (dengan progress), apabila saya tidak memenuhi syarat tersebut maka saya tidak berhak meminta surat rekomendasi mengikuti sidang skripsi dan saya juga berjanji memenuhi persyaratan tersebut dan menyelesaikan penelitian berupa produk dan laporan sesuai jadwal penelitian (tepat waktu).
Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Skripsi

Tegal, 07 Maret 2025

Pihak Pertama



Rama Oktabara

Pihak Kedua



Dega Surono Wibowo, S.T., M.Kom

Mengetahui
Ketua Program Studi Sarjana Terapan Teknik Informatika



Dyah Apriliani, S.T., M.Kom
NIPY 09 015 225

Lampiran 2 Surat Observasi Klinik Dr.Viandini



POLITEKNIK HARAPAN BERSAMA

Sarjana Terapan Teknik Informatika

Nomor : 05.03/TI.PHB/II/2025
Lampiran : -
Hal : Permohonan Izin Observasi
Kepada : Pemilik Klinik Kesehatan Dr.Viandini
Yth. Di Tempat

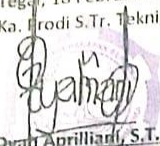
Dengan hormat, mahasiswa dengan identitas berikut ini:

nama : Rama Oktabara
NIM : 21090008
prodi : Sarjana Terapan Teknik Informatika

Bermaksud melakukan penelitian untuk keperluan Skripsi dengan judul "OPTIMALISASI PELAYANAN KESEHATAN MELALUI SISTEM PENDATAAN PASIEN DIGITAL BERBASIS WEB DENGAN FITUR ELECTRONIC MEDICAL RECORD (EMR) DI KLINIK DR. VIANDINI". Kami memohon Bapak/Ibu memberikan izin kepada mahasiswa yang bersangkutan agar memperoleh data, keterangan, dan bahan yang diperlukan.

Demikian permohonan ini disampaikan, atas perhatian kami ucapkan terima kasih.

Tegal, 18 Februari 2025
Ka. Prodi S.Tr. Teknik Informatika,


Dyan Aprilliani, S.T., M.Kom
NIPY : 09.015.725

Lampiran 3 Surat Pernyataan HKI

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini, pemegang hak cipta:

1. N a m a : Rama Oktabara
Kewarganegaraan : Indonesia
Alamat : Jl. Jati Raya No.157 Griya Pelutan Indah,RT/RW(03/03)
52311 ,
Kel.Pelutan,Kec.Pemalang
Kab.Pemalang,Provinsi Jawa Tengah
2. N a m a : Ir. Ginanjar Wiro Sasmito, S.Kom.,M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Kluwut RT.003/002, Kecamatan Bulakamba,
Kabupaten Brebes, Provinsi Jawa Tengah, 52253
3. N a m a : Dega Surono Wibowo, S.T.,M,Kom.
Kewarganegaraan : Indonesia
Alamat : Perumahan Sapphire Regency Blok H, No.1,RT.004,
RW.001,Kelurahan Pulosari,Kecamatan Brebes, 52213

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan:
Berupa : Teknologi Informasi
Berjudul : Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi
Aktivitas Pasien Menggunakan Algoritma K-Nearest Neighbor
☐ Tidak meniru dan tidak sama secara esensial dengan Karya Cipta milik pihak lain atau
obyek kekayaan intelektual lainnya sebagaimana dimaksud dalam Pasal 68 ayat (2);
☐ Bukan merupakan Ekspresi Budaya Tradisional sebagaimana dimaksud dalam Pasal 38;
☐ Bukan merupakan Ciptaan yang tidak diketahui penciptanya sebagaimana dimaksud dalam
Pasal 39;
☐ Bukan merupakan hasil karya yang tidak dilindungi Hak Cipta sebagaimana dimaksud
dalam Pasal 41 dan 42;
☐ Bukan merupakan Ciptaan seni lukis yang berupa logo atau tanda pembeda yang
digunakan sebagai merek dalam perdagangan barang/jasa atau digunakan sebagai lambang
organisasi, badan usaha, atau badan hukum sebagaimana dimaksud dalam Pasal 65 dan;
☐ Bukan merupakan Ciptaan yang melanggar norma agama, norma susila, ketertiban umum,
pertahanan dan keamanan negara atau melanggar peraturan perundang-undangan
sebagaimana dimaksud dalam Pasal 74 ayat (1) huruf d Undang-Undang Nomor 28 Tahun
2014 tentang Hak Cipta.

2. Sebagai pemohon mempunyai kewajiban untuk menyimpan asli contoh ciptaan yang dimohonkan dan harus memberikan apabila dibutuhkan untuk kepentingan penyelesaian sengketa perdata maupun pidana sesuai dengan ketentuan perundang-undangan.
3. Karya Cipta yang saya mohonkan pada Angka 1 tersebut di atas tidak pernah dan tidak sedang dalam sengketa pidana dan/atau perdata di Pengadilan.
4. Dalam hal ketentuan sebagaimana dimaksud dalam Angka 1 dan Angka 3 tersebut di atas saya / kami langgar, maka saya / kami bersedia secara sukarela bahwa:
 - a. permohonan karya cipta yang saya ajukan dianggap ditarik kembali; atau
 - b. Karya Cipta yang telah terdaftar dalam Daftar Umum Ciptaan Direktorat Hak Cipta, Direktorat Jenderal Hak Kekayaan Intelektual, Kementerian Hukum Dan Hak Asasi Manusia R.I dihapuskan sesuai dengan ketentuan perundang-undangan yang berlaku.
 - c. Dalam hal kepemilikan Hak Cipta yang dimohonkan secara elektronik sedang dalam perkara dan/atau sedang dalam gugatan di Pengadilan maka status kepemilikan surat pencatatan elektronik tersebut ditangguhkan menunggu putusan Pengadilan yang berkekuatan hukum tetap.

Demikian Surat pernyataan ini saya/kami buat dengan sebenarnya dan untuk dipergunakan sebagaimana mestinya.

Tegal, 18 Juni 2025



(Rama Oktabara)

(Ir. Ginanjar Wiro Sasmito, S.Kom.,M.Kom.)

(Dega Surono Wibowo, S.T.,M.,Kom..)

Lampiran 4 Surat Pengalihan HKI

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini :

1. Nama : Rama Oktabara
Kewarganegaraan : Indonesia
Alamat : Jl. Jati Raya No.157 Griya Pelutan Indah, RT/RW(03/03) 52311 ,
Kel. Pelutan, Kec. Pemalang Kab. Pemalang, Provinsi Jawa Tengah
2. Nama : Ir. Ginanjar Wiro Sasmito, S.Kom., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Desa Kluwut RT.003/002, Kecamatan Bulakamba, Kabupaten
Brebes, Provinsi Jawa Tengah, 52253
3. Nama : Dega Surono Wibowo, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Perumahan Sapphire Regency Blok H, No.1, RT.004, RW.001,
Kelurahan Pulosari, Kecamatan Brebes, 52213

Adalah Pihak I selaku pencipta, dengan ini menyerahkan karya ciptaan saya kepada :

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)
Politeknik Harapan Bersama
Alamat : Jl. Mataram No. 9 Pasuruan Lor Kota Tegal

Adalah Pihak II selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "Sistem Relam Medis Elektronik Berbasis Web dengan Rekomendasi Aktivitas Pasien Menggunakan Algoritma K-Nearest Neighbor ". untuk didaftarkan di Direktorat Hak Cipta dan Desain Industri, Direktorat Jenderal Kekayaan Intelektual, Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia.

Demikianlah surat pengalihan hak ini kami buat, agar dapat dipergunakan sebagaimana mestinya.

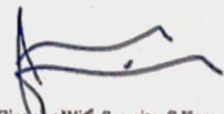
Tegal, 18 Juni 2025

Pemegang Hak Cipta
Kepala P3M

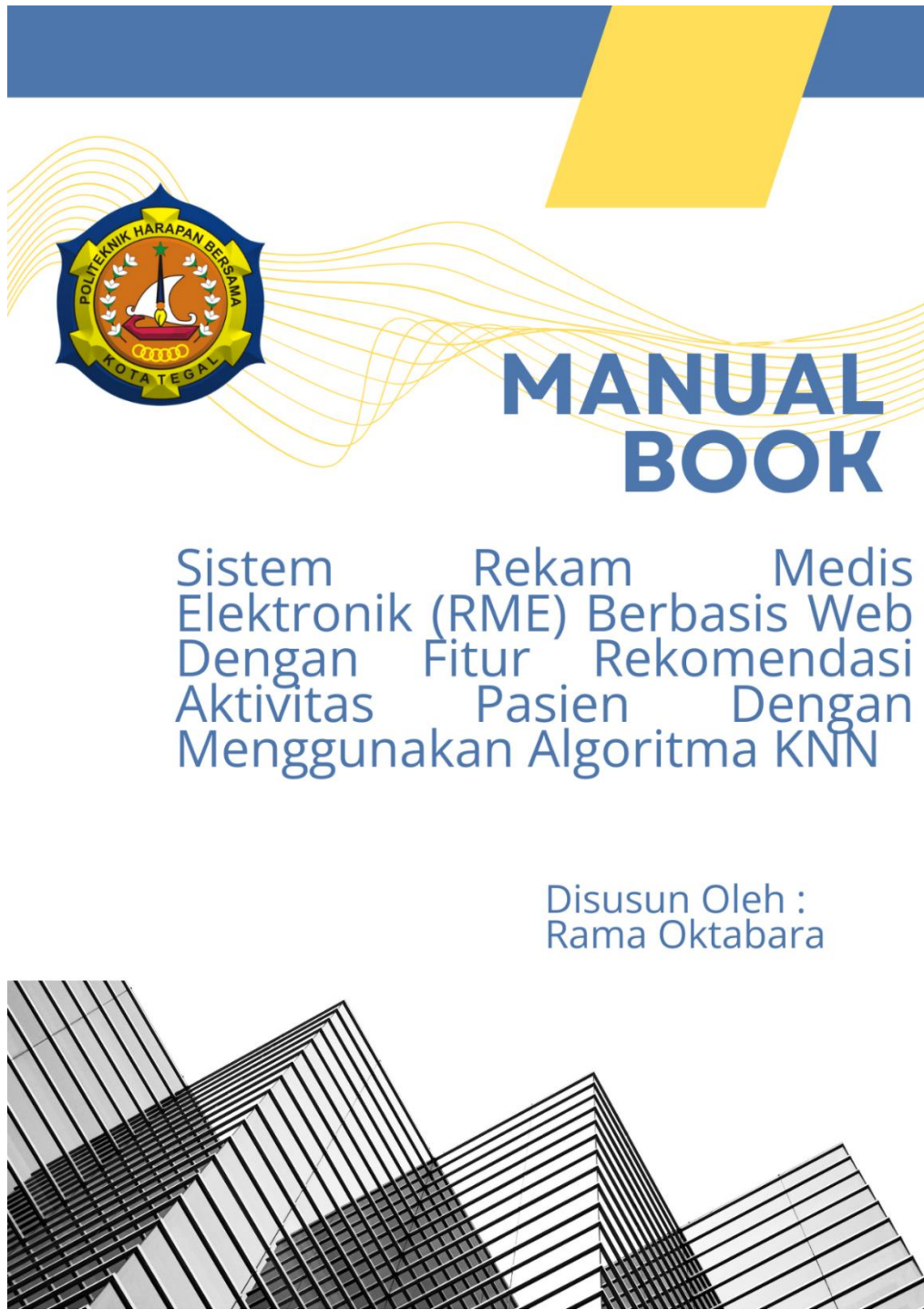
(Muhammad Fala Hidayatullah, S.T., M.Kom.)

Pencipta

(Rama Oktabara)


(Ir. Ginanjar Wiro Sasmito, S.Kom., M.Kom.)


(Dega Surono Wibowo, S.T., M.Kom.)



1. PENDAHULUAN

1.1. Tujuan Pembuatan Dokumen

Dokumen manual book sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN. dibuat dengan tujuan menjelaskan, menggambarkan, panduan penggunaan bagaimana cara penggunaan sistem.

1.2. Deskripsi Umum Sistem

Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN dirancang untuk dapat membantu klinik atau fasilitas kesehatan dalam melakukan pendataan riwayat kesehatan pasien secara digital, cepat, dan terstruktur. Sistem ini tidak hanya mencatat informasi medis, tetapi juga dilengkapi dengan fitur rekomendasi aktivitas pasien yang disesuaikan berdasarkan riwayat medis dan kondisi pasien menggunakan algoritma K-Nearest Neighbor (KNN). Rekomendasi aktivitas tersebut akan dikirimkan secara otomatis melalui aplikasi WhatsApp agar dapat langsung diterima dan dipahami oleh pasien tanpa harus mengakses sistem secara manual. Dengan adanya sistem ini, diharapkan proses pelayanan kesehatan menjadi lebih efisien, personal, dan adaptif terhadap kebutuhan masing-masing pasien.

1.3. Deskripsi Dokumen

Dokumen ini dibuat untuk memberikan panduan penggunaan “Sistem Diagnosis Penyakit Mata Pada Anak Menggunakan Algoritma Yolo V8”. Dokumen ini berisikan informasi sebagai berikut:

1. BAB I.

Berisi informasi umum yang merupakan bagian pendahuluan, yang meliputi tujuan pembuatan dokumen, deskripsi umum sistem serta deskripsi dokumen.

2. BAB II.

Berisi perangkat yang dibutuhkan untuk “Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN” meliputi perangkat lunak dan

perangkat keras.

3. BAB III.

Berisi pengguna manual Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN.

2. PERANGKAT YANG DIBUTUHKAN

2.1. Perangkat Lunak

Perangkat lunak yang digunakan adalah sebagai berikut :

1. Sistem operasi seperti Windows
2. Web browser (Google Chrome, Fierfox, Microsoft Edge, dll)

2.2. Perangkat Keras

Perangkat keras (Hardware) yang digunakan sebagai berikut :

1. Laptop / pc.
2. Mouse.
3. Keyboard

3. FITUR DAN CARA PENGGUNAAN

3.1. Struktur Fitur

Adapun Sturktur halaman pada sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN sebagai berikut :

1. Halaman utama beranda

Pada halaman utama *Dashboard* terdapat informasi pasien yang telah melakukan registrasi per hari, dan juga terdapat chart yang berisi pasien yang sudah melakukan pemeriksaan yang di filter mingguan, bulanan dan tahunan. Pada *Dashboard* juga terdapat chart

penyakit yang sering dialami oleh pasien yang melakukan pemeriksaan.

2. **Halaman Pasien**

Pada halaman ini pengguna dapat melihat tabel yang berisi informasi pasien yang terdaftar di sistem, terdapat juga fitur :

1. Form input biodata pasien baru

3. **Halaman Registrasi**

Pada halaman registrasi pengguna dapat melihat pasien yang sudah didaftarkan oleh staff klinik sesuai nomer urut antrian. Fitur yang terdapat di halaman ini :

1. Form input registrasi
2. Button kajian awal
3. Button rekam medis

4. **Halaman Rekam Medis**

Pada halaman rekam medis pengguna dapat melihat list rekam medis dari pasien, fitur yang terdapat pada halaman ini:

1. Form input rekam medis sesuai pasien yang terdaftar

5. **Halaman Layanan**

Pada halaman ini pengguna dapat melihat list layanan yang akan diberikan kepada pasien, dan juga terdapat fitur form inputan layanan baru.

6. **Halaman Data Obat**

Pada halaman ini pengguna dapat melihat list data obat yang ada di klinik dan juga stok, pengguna juga dapat melakukan form inputan obat baru dan update stok serta harga

7. **Halaman User**

Pada halaman ini pengguna dapat menambah user yang dapat mengakses sistem ini.

8. **Halaman Role**

Pada halaman ini pengguna dapat menambahkan *Role* untuk yang dapat mengakses setiap halaman/fiturnya.

3.2. Penggunaan

Pada bagian ini akan dijelaskan mengenai fitur fitur yang tersedia pada Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN.

1. Cara Membuka Sistem

Untuk menjalankan Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN. Langkah langkah yang harus dilakukan pengguna adalah sebagai berikut :

1. Buka alamat domain website melalui web browser seperti Google Chrome, Firefox dan sejenisnya.
2. Tekan enter setelah mengetik alamat web dipencarian browser.
3. Setelah itu akan muncul halaman login untuk mengakses websitenya.

2. Halaman Pengguna

Pada sistem terdapat halaman pengguna yang terdiri dari berbagai fitur sebagai berikut :

1. Fitur Login

Fitur login ini digunakan untuk masuk kedalam sistem sesuai hak akses. Disini tidak terdapat menu registrasi karena pembuatan akun akan dilakukan oleh admin.

Laravel
Masuk ke akun Anda

Alamat email*

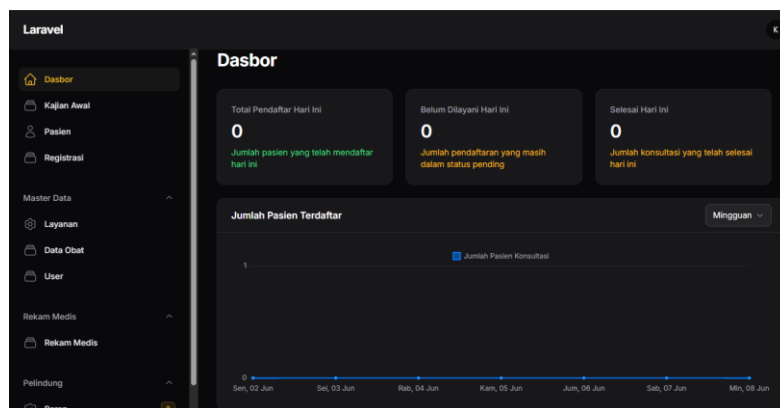
Kata sandi*

☐ Ingat saya

Masuk

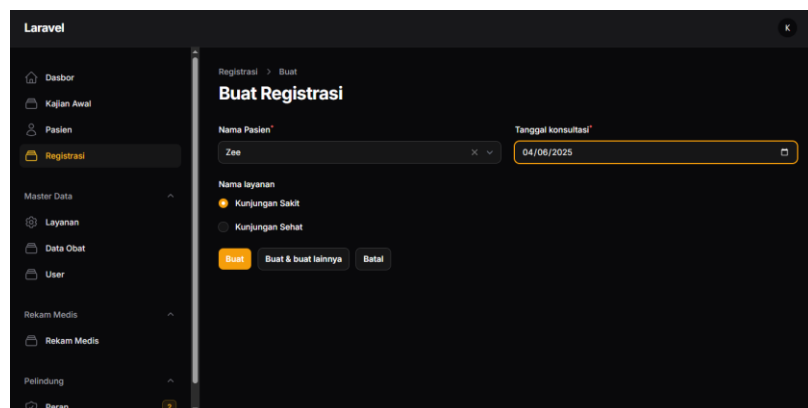
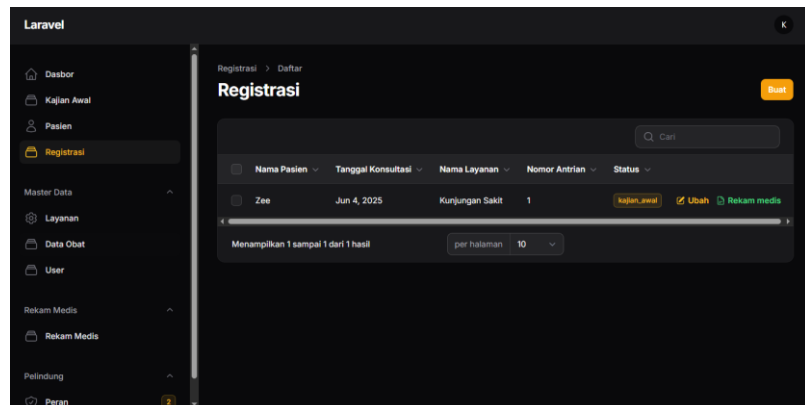
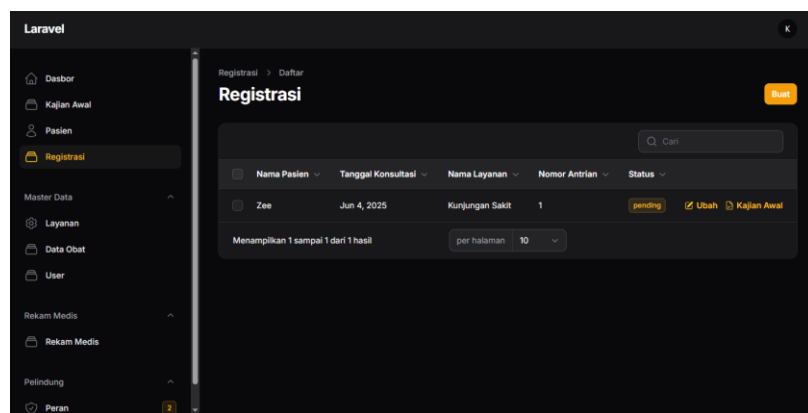
2. Fitur *Dashboard*

Setelah melakukan login pengguna akan dibawa kehalaman *Dashboard* yang berisi pasien yang sudah melakukan registrasi per hari. Lalu ada chart untuk melihat banyaknya pasien yang melakukan pemeriksaan diklinik dan banyaknya penyakit yang diderita pasien.



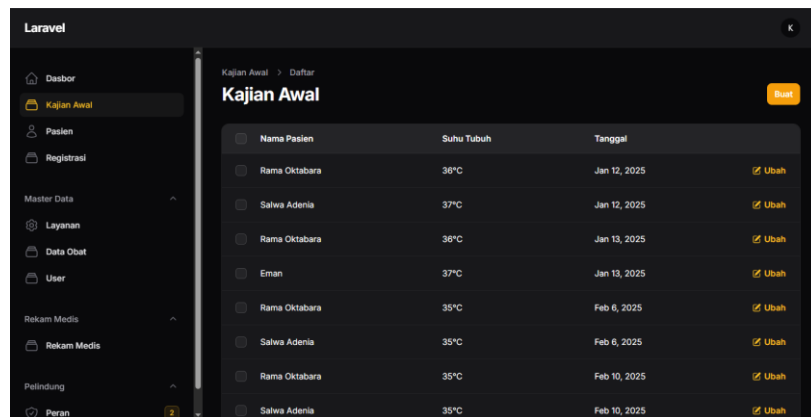
4. Fitur Registrasi

Fitur ini pengguna dapat melakukan pembuatan registrasi/urutan dari pasien yang berkunjung per-harinya. Setelah melakukan input registrasi, pada tabel registrasi terdapat list antrian. Dan pada tabel registrasi kolom status pengguna dapat melihat status pasien, sudah dilakukan pemeriksaan atau belum. Disini juga terdapat tombol untuk pengisian kajian awal lalu tombol rekam medis ketika sudah diisi kajian awal.

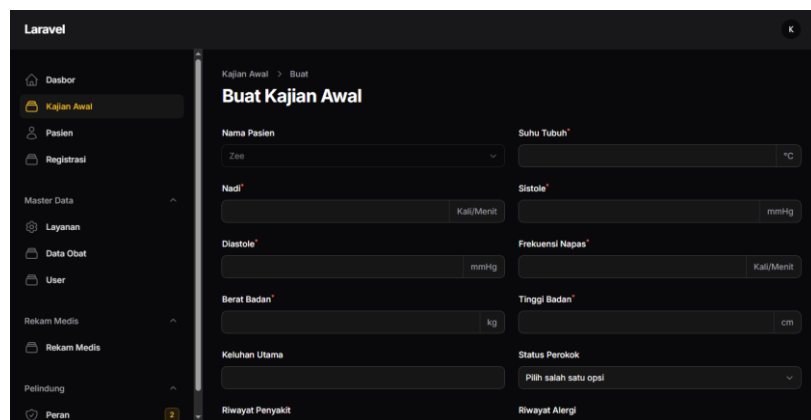


5. Fitur Kajian Awal

Fitur Kajian awal ini pengguna dapat melihat list kajian awal dari pasien yang sudah melakukan pendaftaran. Untuk pengisian kajian awal dapat melalui button Kajian Awal yang terdapat pada tabel registrasi. Karena pengisian kajian awal harus ada pasien yang terdaftar diregistrasi.



☐	Nama Pasien	Suhu Tubuh	Tanggal	
☐	Rama Oktabara	36°C	Jan 12, 2025	Ubah
☐	Salwa Adenia	37°C	Jan 12, 2025	Ubah
☐	Rama Oktabara	38°C	Jan 13, 2025	Ubah
☐	Eman	37°C	Jan 13, 2025	Ubah
☐	Rama Oktabara	35°C	Feb 6, 2025	Ubah
☐	Salwa Adenia	35°C	Feb 6, 2025	Ubah
☐	Rama Oktabara	35°C	Feb 10, 2025	Ubah
☐	Salwa Adenia	35°C	Feb 10, 2025	Ubah



Buat Kajian Awal

Nama Pasien: Zee Suhu Tubuh: °C

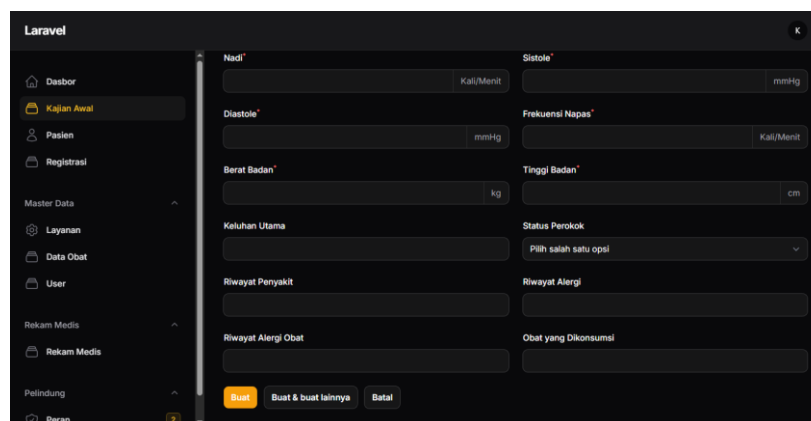
Nadi: Kali/Menit Sistol: mmHg

Diastole: mmHg Frekuensi Napas: Kali/Menit

Berat Badan: kg Tinggi Badan: cm

Keluhan Utama: Status Perokok: Pilih salah satu opsi

Riwayat Penyakit: Riwayat Alergi:



Buat Kajian Awal

Nadi: Kali/Menit Sistol: mmHg

Diastole: mmHg Frekuensi Napas: Kali/Menit

Berat Badan: kg Tinggi Badan: cm

Keluhan Utama: Status Perokok: Pilih salah satu opsi

Riwayat Penyakit: Riwayat Alergi:

Riwayat Alergi Obat: Obat yang dikonsumsi:

[Buat](#) [Buat & buat lainnya](#) [Batal](#)

6. Fitur Rekam Medis

Fitur ini pengguna dapat menginputkan rekam medis pasien yang sudah terdaftar di registrasi dan sudah dilakukan inputan kajian awal. Pada form inputan rekam medis juga terdapat history rekam medis pasien yang sebelumnya guna penunjang rekam medis pasien.

The top screenshot displays the 'Buat Rekam Medis' (Create Medical Record) form. It features a sidebar with navigation links: Dasbor, Kajian Awal, Pasien, Registrasi, Master Data (Layanan, Data Obat, User), Rekam Medis (highlighted), Pelindung, and Peran. The main content area includes a 'Nama Pasien' dropdown menu with 'Zee' selected. Below this is the 'Kajian Awal' (Initial Assessment) section, which contains input fields for: Suhu Tubuh (35), Nadi (100), Sistol (120), Diastole (90), Frekuensi Napas (100), Berat Badan (56), Tinggi Badan (165), and Keluhan Utama (sakit perut bagian kanan).

The bottom screenshot shows the 'Riwayat Rekam Medis' (Medical Record History) section. It includes a 'Status Perokok' dropdown menu with 'Bukan Perokok' selected. Below this are input fields for 'Riwayat Alergi' and 'Riwayat Alergi Obat'. The 'Obat yang Dikonsumsi' (Medications Consumed) section is currently empty. The 'Riwayat Rekam Medis' section includes a 'Filter Tanggal' (Filter Date) dropdown menu with 'Pilih Tanggal' (Select Date) and '27 Mei 2025' selected. The 'Detail Rekam Medis' (Medical Record Detail) section is also visible, showing a 'Tanggal' (Date) input field.

Laravel

27 Mei 2025

Detail Rekam Medis

Tanggal
27/05/2025 04:10

Suhu Tubuh (°C) 35 Nadi (x/menit) 90 Sistol (mmHg) 110 Diastole (mmHg) 100

Frekuensi Napas (x/menit)
90

Diagnosa
Hipertensi

Obat
Metformin (1)

Laravel

Rekam Medis

Diagnosa*
Maag (GERD)

Rekomendasi Aktivitas Scroll untuk melihat seluruh rekomendasi
Jalan cepat 20 menit pagi & sore (total >150 menit/egg); hindari olahraga berat langsung setelah makan. Dianjurkan: Roti gandum halus, pepaya matang, telur rebus; infus (air hangat + jahe muda). Dihindari: Makanan pedas, saus sambal, jeruk nipis, minuman es.

Layanan

Daftar Layanan*

Pilih Layanan* Subtotal Layanan
Pemeriksaan x Rp 0

Tambah Layanan

Laravel

Obat

Daftar Obat*

Pilih Obat* Jumlah* Subtotal
Promag 1 Rp 0

Tambah Obat

Total Harga
Rp 0

Status pulang*

Buat Buat & buat lainnya Batal

7. Fitur Layanan

Fitur layanan menampilkan layanan/tindakan yang ada dalam klinik. Dan dapat menambah layanan baru dan update harga.

Laravel

Layanan > Daftar

Layanan

Sort

☐	Nama Layanan	Harga	Tipe Layanan	
☐	Infuse Vitamin	Rp 150.000	Rawat Jalan	☒ Ubah
☐	Cek Kolesterol	Rp 25.000	Rawat Jalan	☒ Ubah
☐	Perawatan Luka Kecil	Rp 35.000	Rawat Jalan	☒ Ubah
☐	Perawatan Luka Sedang	Rp 45.000	Rawat Jalan	☒ Ubah
☐	Perawatan Luka Besar	Rp 55.000	Rawat Jalan	☒ Ubah
☐	Jahit Luka 1-3	Rp 60.000	Rawat Jalan	☒ Ubah
☐	Jahit Luka 4-6	Rp 90.000	Rawat Jalan	☒ Ubah
☐	Jahit Luka 7-10	Rp 120.000	Rawat Jalan	☒ Ubah

Laravel

Layanan > Buat

Buat Layanan

Nama layanan

Harga

Tipe layanan

☐ Rawat Jalan

☐ Sakit Inap

8. Fitur Data Obat

Halaman ini menampilkan list obat dan stok obat yang ada dalam klinik. Dan dapat menambah juga mengupdate stok dan harga obat.

Laravel

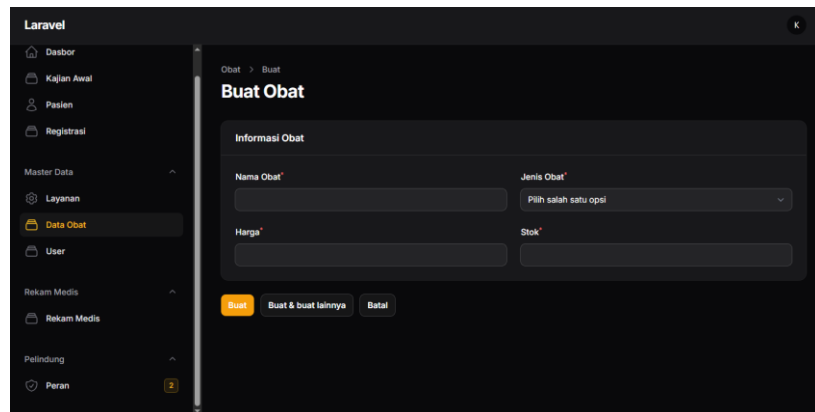
Obat > Daftar

Obat

Search

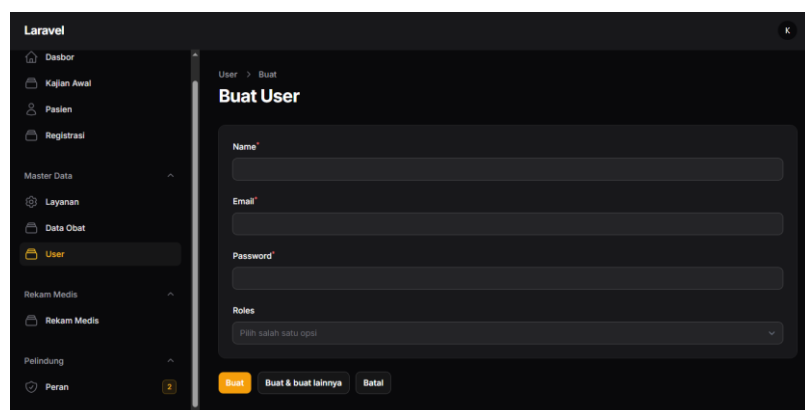
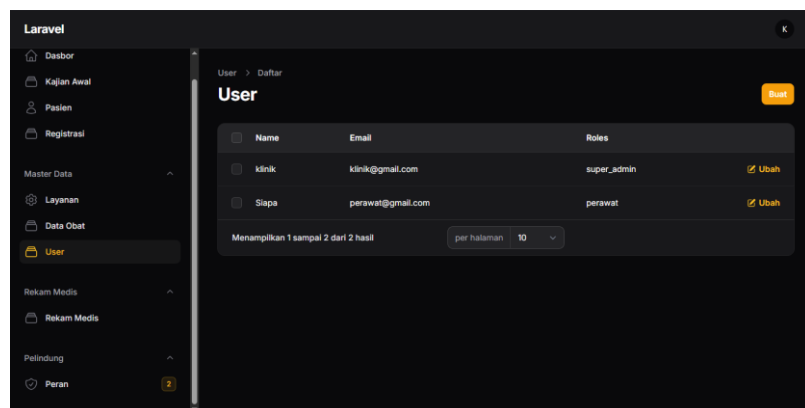
6

☐	Nama Obat	Jenis Obat	Harga	Stok	
☐	Promag	Padat	Rp 10.000	7	☒ Ubah ☐ Hapus
☐	Flu	Cair	Rp 25.000	42	☒ Ubah ☐ Hapus
☐	Syr Flutamol	Cair	Rp 25.000	6	☒ Ubah ☐ Hapus
☐	Syr Ambroxol	Cair	Rp 50.000	8	☒ Ubah ☐ Hapus
☐	Syr Anacetin	Cair	Rp 25.000	10	☒ Ubah ☐ Hapus
☐	Syr Etamox	Cair	Rp 25.000	10	☒ Ubah ☐ Hapus
☐	Amlodipin	Padat	Rp 10.000	10	☒ Ubah ☐ Hapus



9. Fitur User

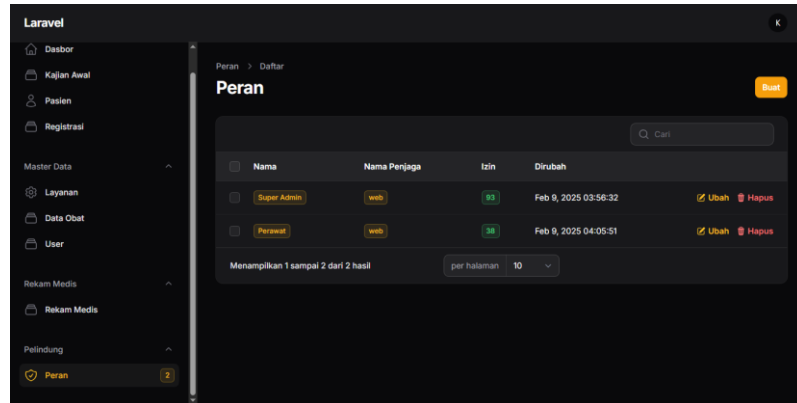
Halaman ini menampilkan dan menambah user yang dapat mengakses website, halaman ini hanya dapat diakses oleh pengguna dengan *Role* super admin.



10. Fitur Role/Peran

Halaman ini menampilkan *Role*/peran yang ada di website ini. Halaman ini hanya dapat diakses oleh super admin, pada halaman

ini pengguna dapat memberikan izin akses fitur pada setiap *Rolenya*



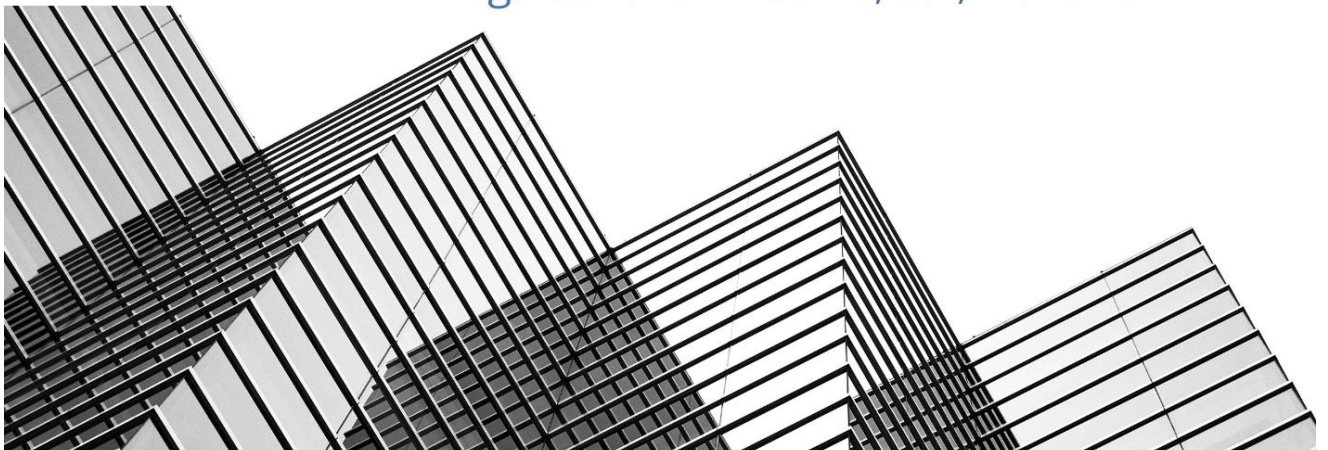


DOKUMEN TEKNIKAL

Sistem Rekam Medis
Elektronik (RME) Berbasis Web
Dengan Fitur Rekomendasi
Aktivitas Pasien Dengan
Menggunakan Algoritma KNN

Disusun Oleh :
Rama Oktabara

Ir. Ginanjar Wiro Sasmito, S.Kom, M.Kom.
Dega Surono Wibowo, S.T, M.Kom.



1. Model Pasien

```
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Model;
6  use Illuminate\Database\Eloquent\Factories\HasFactory;
7  use Carbon\Carbon;
8  use Illuminate\Support\Facades\Log;
9
10 class Pasien extends Model
11 {
12     //
13     use HasFactory;
14     protected $guarded = [];
15     protected $casts = [
16         'tanggal_lahir' => 'date',
17     ];
18
19     public function getUmurAttribute()
20     {
21         try {
22             if (!$this->tanggal_lahir) {
23                 throw new \Exception('Tanggal lahir tidak tersedia');
24             }
25
26             return (int) Carbon::parse($this->tanggal_lahir)->diffInYears(now());
27
28         } catch (\Exception $e) {
29             Log::error('Error menghitung umur: ' . $e->getMessage(), [
30                 'pasien_id' => $this->id,
31                 'tanggal_lahir' => $this->tanggal_lahir
32             ]);
33             return 0;
34         }
35     }
36 }
37
38
```

Kode pada model pasien berfungsi untuk mengelola data pasien di dalam aplikasi rekam medis. Model ini merepresentasikan tabel `pasien` di database sehingga setiap instansinya mencerminkan baris data pasien yang berisi informasi seperti nama lengkap, tanggal lahir, jenis kelamin, alamat, nomor telepon, dan email. Dengan menggunakan properti `$guarded = []`, model ini memungkinkan penyimpanan dan pembaruan semua kolom secara massal (mass-assignment) ketika memanggil metode seperti `Pasien::create($data)`, asalkan validasi data sudah dilakukan pada level controller.

Di bagian `$casts`, kolom `tanggal_lahir` dikonversi menjadi objek **Carbon** setiap kali dibaca. Artinya, ketika aplikasi mengambil nilai `tanggal_lahir` dari database, langsung dihasilkan objek tanggal yang memudahkan manipulasi, misalnya saat memformat tanggal untuk ditampilkan di tampilan pengguna. Lebih lanjut, terdapat metode `getUmurAttribute()` yang secara otomatis menghitung umur pasien berdasarkan nilai `tanggal_lahir`: pertama ia memeriksa apakah kolom tersebut tidak kosong, kemudian menghitung selisih tahun antara tanggal lahir dan waktu sekarang menggunakan fungsi `Carbon`. Jika tanggal lahir tidak tersedia atau tidak valid, kode akan mencatat kesalahan ke log aplikasi dan mengembalikan nilai `0` sebagai umur.

Dengan demikian, di bagian controller atau tampilan (Blade), kita cukup memanggil properti `$pasien->umur` untuk mendapatkan umur pasien dalam satuan tahun tanpa harus menulis ulang logika perhitungan tanggal di berbagai tempat. Apabila suatu saat terdapat pasien yang belum memiliki nilai tanggal lahir, aplikasi tidak akan gagal, melainkan hanya menampilkan umur `0` dan pencatat log akan memberi tahu developer bahwa ada data tanggal lahir yang perlu diperbaiki. Secara keseluruhan, model ini memastikan data pasien tersimpan dengan benar, mempermudah konversi tanggal, dan memberi cara yang konsisten untuk menampilkan informasi umur pasien di seluruh bagian aplikasi.

2. Model Registrasi

```
1  <?php
2  namespace App\Models;
3
4  use Illuminate\Database\Eloquent\Model;
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Carbon\Carbon;
7
8  class Registrasi extends Model
9  {
10     use HasFactory;
11
12     protected $guarded = [];
13
14     protected static function boot()
15     {
16         parent::boot();
17
18         static::creating(function ($registrasi) {
19
20             $tanggal = Carbon::parse($registrasi->tanggal_konsultasi)->toDateString();
21
22             $nomorAntrianTerakhir = self::where('tanggal_konsultasi', $tanggal)
23                 ->max('nomor_antrian');
24
25             $registrasi->nomor_antrian = $nomorAntrianTerakhir ? $nomorAntrianTerakhir + 1 : 1;
26         });
27     }
28
29     public function pasien()
30     {
31         return $this->belongsTo(Pasien::class, 'pasien_id');
32     }
33
34     protected $fillable = ['pasien_id', 'status', 'nomor_antrian', 'tanggal_konsultasi', 'nama_layanan'];
35 }
36
37
38
39
```

Penjelasan Per bagian:

1. Representasi Tabel registrasis di Database

Model Registrasi mewarisi kemampuan Eloquent, sehingga setiap instansinya merepresentasikan baris pada tabel registrasis (konvensi Laravel: nama tabel bentuk jamak). Dengan properti \$fillable, model ini hanya mengizinkan kolom yang tercantum di dalam \$fillable untuk diisi secara massal saat membuat atau memperbarui data, sehingga kolom lain di luar daftar tersebut tidak akan terisi tanpa izin eksplisit.

2. protected \$guarded = [] dan protected \$fillable

- `protected $guarded = []` menyatakan tidak ada kolom yang dilarang untuk mass-assignment.
- `protected $fillable = [...]` selanjutnya menentukan kolom apa saja yang boleh diisi secara massal (misalnya saat memanggil `Registrasi::create($data)`). Hanya kolom di dalam array `$fillable` yang akan diizinkan.

3. Method `boot()` untuk Menetapkan Nomor Antrian Otomatis

Pada event creating, sebelum data Registrasi disimpan:

1. Mengonversi `tanggal_konsultasi` ke Format Y-m-d
 - `Carbon::parse($registrasi->tanggal_konsultasi)->toDateString()` memastikan perbandingan hanya pada tingkat hari (tanpa jam).
2. Mengambil Nomor Antrian Tertinggi pada Hari yang Sama
 - `self::where('tanggal_konsultasi', $tanggal)->max('nomor_antrian')` mencari nomor antrian tertinggi untuk semua entri registrasi dengan tanggal konsultasi yang sama.
3. Menetapkan Nomor Antrian Baru
 - Jika sudah ada nomor antrian sebelumnya, maka `nomor_antrian` baru di-set ke nilai tertinggi + 1. Jika belum ada, di-set ke 1. Dengan ini, setiap pasien yang mendaftar pada hari yang sama akan mendapatkan nomor urut antrian berurutan.
4. Relasi Eloquent ke Model Pasien
 - Menunjukkan bahwa setiap entri Registrasi terkait dengan satu entri Pasien melalui kolom `pasien_id`.
 - Memanggil `$registrasi->pasien` akan menghasilkan objek Pasien yang bersangkutan.
5. Kolom-kolom di `$fillable`
 - `pasien_id`: ID pasien yang mendaftar.

- status: Status pendaftaran (misalnya 'pending', 'selesai', 'batal').
- nomor_antrian: Nomor urut antrian yang dihasilkan otomatis pada event creating.
- tanggal_konsultasi: Tanggal (dan bisa juga waktu) pasien dijadwalkan konsultasi.
- nama_layanan: Jenis layanan atau poliklinik yang dipilih pasien (misalnya 'Umum', 'Spesialis', dll.).

Ringkasan kegunaan model Registrasi:

- Menyimpan Data Pendaftaran Kunjungan Pasien
Model ini menampung informasi pendaftaran, seperti pasien_id, tanggal_konsultasi, dan nama_layanan.
- Menetapkan Nomor Antrian Otomatis
Saat membuat data baru, nomor antrian diatur berdasarkan urutan pendaftaran pada hari yang sama, tanpa perlu dihitung manual di controller.
- Relasi Eloquent
 - pasien(): Mengambil data pasien terkait pada setiap entri registrasi.

3. Model Kajian Awal

```
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Model;
6  use Illuminate\Database\Eloquent\Factories\HasFactory;
7
8  class KajianAwal extends Model
9  {
10     use HasFactory;
11
12     protected $guarded = [];
13
14     public function pasien()
15     {
16         return $this->belongsTo(Pasien::class, 'pasien_id', 'id');
17     }
18
19
20     protected static function booted()
21     {
22         static::creating(function ($kajianAwal) {
23
24             if ($kajianAwal->pasien) {
25                 $kajianAwal->nama_lengkap = $kajianAwal->pasien->nama_lengkap;
26             }
27             $registrasi = Registrasi::find($kajianAwal->registrasi_id);
28             if ($registrasi) {
29                 $registrasi->status = 'kajian_awal';
30                 $registrasi->save();
31             }
32         });
33     }
34
35     public static function existsForToday($pasienId)
36     {
37         return self::where('pasien_id', $pasienId)
38             ->whereDate('created_at', now()->toDateString())
39             ->exists();
40     }
41     public function registrasi()
42     {
43         return $this->belongsTo(Registrasi::class);
44     }
45     protected $attributes = [
46         'sistole' => 0,
47         'diastole' => 0,
48         'nadi' => 0,
49         'frekui_napas' => 0
50     ];
51
52     protected $fillable = ['registrasi_id', 'suhu_tubuh', 'nama_lengkap', 'nadi', 'sistole', 'diastole',
53         'frekui_napas', 'berat_badan', 'tinggi_badan',
54         'keluhan_utama', 'riwayat_penyakit', 'riwayat_alergi', 'riwayat_pengobatan', 'riwayat_alergi_obat',
55         'obat_dikonsumsi', 'status_perokok', 'pasien_id'];
56 }
57
```

Penjelasan bagian-per-bagian:

1. Representasi Tabel kajian_awals di Database

Model KajianAwal mewarisi kemampuan Eloquent dari Laravel, sehingga setiap instansinya akan mewakili baris pada tabel kajian_awals (konvensi Laravel: nama tabel bentuk jamak). Dengan properti \$fillable, model ini hanya mengizinkan kolom yang tercantum di dalam \$fillable untuk diisi secara massal saat membuat atau memperbarui data, sehingga kolom lain di luar daftar tersebut tidak akan terisi tanpa izin eksplisit.

2. Isi \$guarded = []

Menyatakan bahwa tidak ada atribut yang dilarang untuk mass-assignment, namun karena kita mendefinisikan \$fillable setelahnya, maka kolom yang benar-benar boleh diisi massal adalah yang ada di dalam \$fillable. Ini menegaskan pemisahan antara atribut yang boleh dan tidak boleh diisi oleh user.

3. Relasi Eloquent ke Model Pasien

- Menunjukkan bahwa setiap entri KajianAwal terkait dengan satu entri Pasien melalui kolom pasien_id.
- Saat memanggil \$kajianAwal->pasien, Laravel otomatis mengambil objek Pasien yang dimiliki.

4. Booted Method untuk Menyalin Nama Pasien dan Mengubah Status Registrasi

Saat membuat (creating) data KajianAwal baru:

1. Menyalin Nama Lengkap Pasien

- Jika relasi pasien berhasil ditemukan, maka kolom nama_lengkap otomatis diisi dari \$kajianAwal->pasien->nama_lengkap. Hal ini berguna agar tabel kajian_awals memiliki salinan nama lengkap, memudahkan pencarian tanpa perlu JOIN ke tabel pasiens.

2. Mengubah Status di Tabel Registrasi

- Mencari entri Registrasi yang memiliki id sama dengan `registrasi_id` milik `$kajianAwal`.
- Jika ditemukan, nilai kolom status pada entri Registrasi diubah menjadi `'kajian_awal'` untuk menandakan pasien sedang dalam tahap pemeriksaan awal, kemudian disimpan kembali ke database.

5. Method `existsForToday($pasienId)`

- Digunakan untuk mengecek apakah pasien dengan `pasien_id` tertentu sudah memiliki entri `KajianAwal` pada tanggal hari ini (`created_at`).
- Jika sudah pernah membuat kajian awal hari ini, method ini mengembalikan `true`; jika belum, mengembalikan `false`.
- Berguna di controller untuk mencegah pasien membuat data kajian awal lebih dari sekali dalam satu hari.

6. Relasi Eloquent ke Model Registrasi

- Menyatakan bahwa setiap `KajianAwal` dimiliki oleh satu entri Registrasi melalui kolom `registrasi_id`.
- Dengan memanggil `$kajianAwal->registrasi`, kita dapat mengambil data registrasi terkait yang berisi informasi antrian, tanggal konsultasi, dsb.

7. Atribut Default di `$attributes`

- Menetapkan nilai awal default untuk kolom tanda vital (sistole, diastole, nadi, dan `frekui_napas`) menjadi 0.
- Jika saat pembuatan data tidak ada input untuk kolom-kolom tersebut, nilai defaultnya otomatis 0 daripada null.

8. Kolom-kolom yang Diizinkan untuk Mass-Assignment (`$fillable`)

- Hanya kolom-kolom ini yang boleh diisi via `create()` atau `fill()` secara massal.
- `registrasi_id`: ID pendaftaran yang memicu kajian awal.

- suhu_tubuh, nadi, sistole, diastole, frekui_napas: Tanda-tanda vital pasien.
- berat_badan, tinggi_badan: Data antropometri pasien.
- keluhan_utama: Keluhan utama yang diungkap pasien.
- riwayat_penyakit, riwayat_alergi, riwayat_pengobatan, riwayat_alergi_obat, obat_dikonsumsi: Riwayat medis pasien.
- status_perokok: Menandakan apakah pasien merokok atau tidak.
- nama_lengkap: Diisi otomatis dari relasi Pasien.
- pasien_id: ID pasien yang menjalani kajian awal.

Ringkasan kegunaan model **KajianAwal**:

- Menyimpan Data Pemeriksaan Awal Pasien
Model ini menampung semua informasi awal yang dikumpulkan saat pasien baru saja terdaftar, seperti tanda vital dan riwayat penyakit.
- Menyalin Nama Lengkap Otomatis
Saat membuat data baru, kolom nama_lengkap otomatis diisi dengan nama pasien dari tabel pasiensi, tanpa perlu input ulang.
- Mengubah Status Registrasi
Begitu data kajian awal tersimpan, model otomatis memperbarui kolom status di entri Registrasi terkait menjadi "kajian_awal", sehingga petugas tidak perlu mengubah status secara manual di controller.
- Validasi Duplikasi per Hari
Dengan method existsForToday(), aplikasi dapat mencegah pasien membuat kajian awal lebih dari sekali dalam satu hari.
- Relasi Eloquent
 - pasien(): Mengambil data pasien terkait.
 - registrasi(): Mengambil data registrasi yang memicu kajian awal.

Dengan struktur ini, controller dan view cukup memanggil model KajianAwal untuk menyimpan dan memeriksa data awal pasien, sementara logika penyalinan nama dan perubahan status registrasi sudah ditangani di dalam model.

4. Model Rekam Medis

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Model;
6 use Illuminate\Database\Eloquent\Factories\HasFactory;
7
8 class RekamMedis extends Model
9 {
10     protected static function boot()
11     {
12         parent::boot();
13
14         static::saved(function ($rekamMedis) {
15             if ($rekamMedis->registrasi_id) {
16                 $registrasi = \App\Models\Registrasi::find($rekamMedis->registrasi_id);
17                 if ($registrasi) {
18                     $registrasi->status = 'selesai';
19                     $registrasi->save();
20                 }
21             }
22         });
23     }
24     //
25     protected $casts = [
26         'obat' => 'array',
27         'layanan' => 'array',
28         'total_harga' => 'float'
29     ];
30     use HasFactory;
31     protected $guarded = [];
32     protected $fillable = [
33         'registrasi_id',
34         'pasien_id',
35         'kajian_awal_id',
36         'diagnosa',
37         'layanan',
38         'obat',
39         'status_pulang',
40         'suhu_tubuh',
41         'nadi',
42         'sistol',
43         'diastole',
44         'frekuensi_napas',
45         'berat_badan',
46         'tinggi_badan',
47         'keluhan_utama',
48         'status_perokok',
49         'riwayat_penyakit',
50         'riwayat_alergi',
51         'riwayat_alergi_obat',
52         'obat_dikonsumsi',
53         'total_harga',
54         'obat_id',
55         'rekomendasi_aktivitas'
56     ];
57     public function registrasi()
58     {
59         return $this->belongsTo(Registrasi::class);
60     }
61
62     public function pasien()
63     {
64         return $this->belongsTo(Pasien::class);
65     }
66
67     public function kajianAwal()
68     {
69         return $this->belongsTo(KajianAwal::class, 'registrasi_id', 'registrasi_id');
70     }
71     public function obat()
72     {
73         return $this->belongsTo(Obat::class);
74     }
75
76     public function layanan()
77     {
78         return $this->hasMany(Layanan::class);
79     }
80 }
81
82
```

Model RekamMedis ini bertugas mencatat data rekam medis pasien setelah proses kajian awal, sekaligus mengubah status registrasi menjadi “selesai” secara otomatis. Kode di atas menunjukkan bagaimana setiap bagian berfungsi:

1. Representasi Tabel `rekam_medises` di Database

Model RekamMedis mewarisi kemampuan Eloquent, sehingga setiap instansinya merepresentasikan baris pada tabel `rekam_medises` (konvensi Laravel: nama tabel bentuk jamak). Dengan properti `$fillable`, model ini hanya mengizinkan kolom yang tercantum di dalam `$fillable` untuk diisi secara massal saat membuat atau memperbarui data, sehingga kolom lain di luar daftar tersebut tidak akan terisi tanpa izin eksplisit.

2. Method `boot()` untuk Mengubah Status Registrasi

- Pada event `saved`, setelah data RekamMedis selesai disimpan (baik saat pembuatan baru maupun pembaruan), `callback` dijalankan:
 1. Jika properti `registrasi_id` ada, cari entri Registrasi yang sesuai.
 2. Jika entri Registrasi ditemukan, ubah kolom status menjadi 'selesai' dan simpan kembali.
- Dengan ini, setiap kali dokter atau petugas mengisi data rekam medis untuk suatu registrasi, status registrasi otomatis berubah menjadi “selesai” tanpa perlu logika tambahan di controller.

3. `protected $casts` untuk Kolom JSON/Float

- `'obat' => 'array'` dan `'layanan' => 'array'`
 1. Mengartikan bahwa kolom obat dan layanan di database disimpan sebagai JSON (atau teks yang berupa JSON). Saat diambil, otomatis diubah menjadi array PHP sehingga memudahkan iterasi dan manipulasi data obat/layanan.
- `'total_harga' => 'float'`
 1. Memastikan kolom `total_harga` dikonversi menjadi tipe float saat diakses, memudahkan perhitungan atau pembulatan harga.

4. Trait HasFactory dan Properti \$guarded

- use HasFactory; memungkinkan pembuatan data dummy untuk testing atau seeding (RekamMedis::factory()->create()).
- \$guarded = [] menyatakan tidak ada kolom yang dilarang untuk mass-assignment, namun karena kita mendefinisikan \$fillable, hanya kolom di dalam \$fillable yang benar-benar boleh diisi secara massal.

5. Kolom-kolom di \$fillable

- registrasi_id: ID entri Registrasi terkait.
- pasien_id: ID pasien yang memiliki rekam medis ini.
- kajian_awal_id: ID entri KajianAwal yang terkait.
- diagnosa: Hasil diagnosa dokter.
- layanan: Array layanan yang diberikan (contoh: pemeriksaan lab, tindakan).
- obat: Array obat yang diresepkan.
- status_pulang: Status pasien saat pulang (misalnya rawat_jalan, rawat_inap, rujuk).
- suhu_tubuh, nadi, sistole, diastole, frekui_napas, berat_badan, tinggi_badan: Tanda vital dan data antropometri pasien.
- keluhan_utama, status_perokok, riwayat_penyakit, riwayat_alergi, riwayat_alergi_obat, obat_dikonsumsi: Informasi keluhan dan riwayat medis pasien.
- total_harga: Total biaya layanan dan obat (float).
- obat_id: ID obat utama (jika perlu relasi).
- rekomendasi_aktivitas: Teks rekomendasi aktivitas pasca konsultasi.

6. Relasi Eloquent ke Model Lain

1. Setiap entri RekamMedis dimiliki oleh satu entri Registrasi melalui kolom registrasi_id. Memanggil \$rekamMedis->registrasi akan mengambil data pendaftaran terkait.

2. Setiap entri RekamMedis dimiliki oleh satu entri Pasien melalui kolom pasien_id. Memanggil \$rekamMedis->pasien akan mengambil data pasien terkait.
3. Menunjukkan relasi ke KajianAwal yang menggunakan kolom registrasi_id sebagai kunci asing dan kunci lokal. Saat memanggil \$rekamMedis->kajianAwal, data kajian awal terkait diambil berdasarkan registrasi_id.
4. Relasi belongsTo ke model Obat. Biasanya digunakan jika ada satu obat utama yang disimpan di kolom obat_id.
5. Relasi hasMany ke model Layanan, menandakan satu RekamMedis dapat memiliki banyak entri layanan terpisah. Dengan \$rekamMedis->layanan, Anda mendapatkan koleksi semua layanan yang terkait.

Ringkasan kegunaan model RekamMedis:

- Menyimpan Data Rekam Medis Lengkap Pasien
Menampung semua informasi hasil konsultasi dokter, termasuk diagnosa, layanan, obat, tanda vital, riwayat medis, dan total biaya.
- Menangani Logika Otomatisasi Status Registrasi
Setelah data rekam medis tersimpan, status entri Registrasi terkait otomatis diubah menjadi "selesai", sehingga alur kerja lebih efisien tanpa update manual di controller.
- Konversi Kolom JSON ke Array
Kolom obat dan layanan otomatis dikonversi menjadi array saat diakses, memudahkan iterasi dan manipulasi data.
- Relasi Eloquent
 - registrasi(): Mengambil data registrasi yang memicu rekam medis.
 - pasien(): Mengambil data pasien terkait.
 - kajianAwal(): Mengambil data kajian awal berdasarkan registrasi_id.
 - obat(): Mengambil data satu obat utama (jika ada).

- layanan(): Mengambil koleksi layanan yang diberikan selama konsultasi.

Dengan struktur ini, controller dan view cukup memanggil properti \$rekamMedis->pasien, \$rekamMedis->diagnosa, atau \$rekamMedis->layanan untuk menampilkan informasi rekam medis lengkap, sementara logika perubahan status registrasi sudah ditangani di dalam model.

5. Chart Penyakit Terbanyak

```

1  <?php
2
3  namespace App\Filament\Widgets;
4
5  use Filament\Widgets\ChartWidget;
6  use Illuminate\Support\Facades\DB;
7
8  class WPenyakitChart extends ChartWidget
9  {
10     protected static ?string $heading = 'Jumlah Diagnosa Penyakit';
11     protected int|string|array $columnSpan = 3;
12     protected static ?string $pollingInterval = null;
13
14     protected int $labelCount = 0;
15     protected static ?string $maxHeight = '400px';
16
17     protected function getData(): array
18     {
19         $stop = DB::table('rekam_medis')
20             ->select('diagnosa', DB::raw('COUNT(*) as total'))
21             ->groupBy('diagnosa')
22             ->orderByDesc('total')
23             ->get();
24
25         $labels = $stop->pluck('diagnosa')->toArray();
26         $data = $stop->pluck('total')->map(fn($v) => (int) $v)->toArray();
27
28         $this->labelCount = count($labels);
29         static::$maxHeight = min(600, 100 + count($labels) * 40) . 'px';
30
31         $backgroundColors = [];
32         $borderColors = [];
33
34         foreach ($labels as $i) {
35             $r = rand(0, 255);
36             $g = rand(0, 255);
37             $b = rand(0, 255);
38
39             $bg = "rgba($r, $g, $b, 0.7)";
40             $border = "rgba($r, $g, $b, 1)";
41
42             $backgroundColors[] = $bg;
43             $borderColors[] = $border;
44         }
45
46         return [
47             'labels' => $labels,
48             'datasets' => [
49                 [
50                     'label' => 'Jumlah Diagnosa',
51                     'data' => $data,
52                     'backgroundColor' => $backgroundColors,
53                     'borderColor' => $borderColors,
54                     'borderWidth' => 2,
55                     'borderRadius' => 8,
56                     'barThickness' => count($labels) > 10 ? 20 : 30,
57                     'hoverBackgroundColor' => $borderColors,
58                 ],
59             ],
60         ];
61     }
62
63     protected function getType(): string
64     {
65         return 'bar';
66     }
67 }
68

```

```

1  protected function getOptions(): array
2  {
3      return [
4          'maintainAspectRatio' => false,
5          'aspectRatio' => 1.5,
6          'indexAxis' => 'x',
7          'scales' => [
8              'x' => [
9                  'grid' => [
10                     'display' => false,
11                 ],
12                 'ticks' => [
13                     'color' => '#6B7280',
14                     'font' => [
15                         'weight' => 600,
16                     ],
17                 ],
18             ],
19             'y' => [
20                 'grid' => [
21                     'display' => false,
22                 ],
23                 'ticks' => [
24                     'color' => '#6B7280',
25                     'font' => [
26                         'size' => $this->labelCount > 10 ? 12 : 14,
27                         'weight' => 600,
28                     ],
29                 ],
30             ],
31         ],
32         'plugins' => [
33             'legend' => [
34                 'display' => false,
35             ],
36             'tooltip' => [
37                 'enabled' => true,
38                 'backgroundColor' => '#1F2937',
39                 'titleColor' => '#E5E7EB',
40                 'bodyColor' => '#E5E7EB',
41                 'titleFont' => [
42                     'size' => 16,
43                     'weight' => '600',
44                 ],
45                 'bodyFont' => [
46                     'size' => 14,
47                 ],
48                 'padding' => 12,
49                 'cornerRadius' => 8,
50             ],
51             'datalabels' => [
52                 'anchor' => 'end',
53                 'align' => 'end',
54                 'color' => '#1F2937',
55                 'font' => [
56                     'size' => 12,
57                     'weight' => 'bold',
58                 ],
59                 'formatter' => function ($value) {
60                     return number_format($value);
61                 },
62             ],
63         ],
64     ];
65 }
66
67 protected function getPlugins(): array
68 {
69     return [
70         'datalabels' => true,
71     ];
72 }
73 }
74

```

Penjelasan Kode WPenyakitChart

1. Widget Chart Filament untuk Diagnosa Penyakit

Kelas WPenyakitChart merupakan turunan dari ChartWidget pada Filament, digunakan untuk menampilkan grafik (chart) statistik jumlah diagnosa penyakit dari tabel rekam_medis. Widget ini dirancang untuk tampil di *Dashboard* aplikasi berbasis Filament Laravel.

2. Properti Utama Widget

- `protected static ?string $heading`: Judul grafik, diatur sebagai "Jumlah Diagnosa Penyakit".
- `protected int|string|array $columnSpan`: Mengatur lebar kolom widget pada grid *Dashboard* (diset 3 kolom).
- `protected static ?string $pollingInterval`: Tidak diaktifkan (null) karena data tidak di-refresh otomatis.
- `protected int $labelCount`: Menyimpan jumlah label (diagnosa) untuk mengatur tampilan chart secara dinamis.
- `protected static ?string $maxHeight`: Membatasi tinggi maksimal chart secara dinamis berdasarkan banyaknya data (default 400px, dapat berubah).

3. Method `getData()`: Pengolahan dan Persiapan Data Chart

- Query Data Diagnosa:
 - Mengambil data dari tabel `rekam_medis` dengan mengelompokkan berdasarkan kolom `diagnosa` dan menghitung jumlah kemunculannya (`COUNT(*) as total`).
 - Hasil diurutkan dari yang paling banyak (`orderByDesc`).
- Label & Data:
 - Label: Nama-nama diagnosa (`pluck('diagnosa')`).
 - Data: Total kasus per diagnosa, dikonversi ke integer.
- Pengaturan Tinggi Chart:
 - Jumlah label dihitung untuk menentukan tinggi maksimal chart (semakin banyak data, semakin tinggi chart, tapi dibatasi max 600px).
- Pengacakan Warna Bar Chart:
 - Untuk setiap label diagnosa, sistem membuat warna acak (R, G, B) untuk background dan border bar chart.
- Return Format Data:
 - labels: Nama-nama diagnosa.

- datasets: Data jumlah per diagnosa beserta konfigurasi tampilan seperti warna, ketebalan border, radius, dan ketebalan bar. Opsi `hoverBackgroundColor` membuat efek saat bar disentuh.

4. Method `getType()`: Jenis Chart

- Mengembalikan 'bar', sehingga chart yang ditampilkan berbentuk diagram batang (bar chart).

5. Method `getOptions()`: Opsi Tampilan Chart

Konfigurasi detail chart, di antaranya:

- Tampilan Axis dan Skala:
 - Grid pada sumbu x dan y disembunyikan untuk tampilan bersih.
 - Teks label pada axis diatur warna, ukuran, dan ketebalannya.
- Plugins:
 - Legend: Tidak ditampilkan.
 - Tooltip: Dihidupkan, dengan custom warna dan style agar konsisten dengan tema aplikasi.
 - Datalabels: Plugin tambahan untuk menampilkan label data pada ujung bar, diatur warna, ukuran, dan font-nya.
 - Formatter: Angka pada datalabel otomatis diformat ribuan (`number_format`).
- Ukuran dan Aspect Ratio:
 - Chart tidak dipaksa aspect ratio tertentu, namun aspect ratio default 1.5.

6. Method `getPlugins()`: Aktivasi Plugin Tambahan

- Mengaktifkan plugin datalabels agar label angka jumlah muncul di atas/bagian akhir bar.

Ringkasan Kegunaan `WPenyakitChart`

- Menampilkan Statistik Diagnosa Penyakit:
Memvisualisasikan jumlah kasus berbagai diagnosa penyakit dalam bentuk diagram batang, sangat bermanfaat bagi admin/tenaga kesehatan untuk monitoring tren penyakit.
- Otomatis dan Dinamis:
Data diambil langsung dari database, tanpa input manual. Chart secara otomatis menyesuaikan jumlah data, warna, dan tampilan sesuai jumlah diagnosa yang berbeda.
- Interaktif dan Informatif:
Pengguna bisa melihat jumlah kasus tiap diagnosa dengan mudah lewat tooltip, bar, dan label angka pada chart.

6. Chart Pasien Terdaftar

```

1  <?php
2
3  namespace App\Filament\Widgets;
4
5  use Filament\Widgets\ChartWidget;
6  use Illuminate\Support\Facades\DB;
7  use Carbon\Carbon;
8
9  class WPasienChart extends ChartWidget
10 {
11     protected static ?string $heading = 'Jumlah Pasien Terdaftar';
12     protected int|string|array $columnSpan = 3;
13     protected static ?string $maxHeight = '250px';
14
15     protected function getFilters(): array
16     {
17         return [
18             'weekly' => 'Mingguan',
19             'monthly' => 'Bulanan',
20             'yearly' => 'Tahunan',
21         ];
22     }
23
24     protected function getData(): array
25     {
26         $filter = $this->filter ?? 'weekly'; // Ubah default ke mingguan
27         $labels = [];
28         $data = [];
29
30         if ($filter === 'weekly') {
31             $startOfWeek = Carbon::now()->startOfWeek(Carbon::MONDAY);
32
33             for ($i = 0; $i < 7; $i++) {
34                 $date = $startOfWeek->copy()->addDays($i);
35                 $labels[] = $date->translatedFormat('D, d M');
36                 $count = DB::table('registrasis')
37                     ->whereDate('tanggal_konsultasi', $date)
38                     ->count();
39                 $data[] = (int)$count; // Pastikan bilangan bulat
40             }
41         } elseif ($filter === 'monthly') {
42             $startOfMonth = Carbon::now()->startOfMonth();
43             for ($i = 0; $i < 12; $i++) {
44                 $date = $startOfMonth->copy()->addMonths($i);
45                 $labels[] = $date->translatedFormat('M Y');
46                 $count = DB::table('registrasis')
47                     ->whereYear('tanggal_konsultasi', $date->year)
48                     ->whereMonth('tanggal_konsultasi', $date->month)
49                     ->count();
50                 $data[] = (int)$count;
51             }
52         } elseif ($filter === 'yearly') {
53             $currentYear = Carbon::now()->year;
54             for ($i = 4; $i >= 0; $i--) {
55                 $year = $currentYear - $i;
56                 $labels[] = $year;
57                 $count = DB::table('registrasis')
58                     ->whereYear('tanggal_konsultasi', $year)
59                     ->count();
60                 $data[] = (int)$count;
61             }
62         }
63     }
64 }

```

```

1      return [
2          'datasets' => [
3              [
4                  'label' => 'Jumlah Pasien Konsultasi',
5                  'data' => $data,
6                  'borderColor' => '#007bff',
7                  'backgroundColor' => 'rgba(0, 123, 255, 0.5)',
8                  'tension' => 0.4,
9              ],
10         ],
11         'labels' => $labels,
12     ];
13 }
14
15 protected function getType(): string
16 {
17     return 'line';
18 }
19
20 protected function getOptions(): array
21 {
22     return [
23         'scales' => [
24             'y' => [
25                 'beginAtZero' => true,
26                 'ticks' => [
27                     'precision' => 0
28                 ]
29             ],
30         ],
31         'plugins' => [
32             'legend' => [
33                 'display' => true,
34                 'position' => 'top',
35             ]
36         ]
37     ];
38 }
39 }

```

Penjelasan Kode WPasienChart

1. Widget Chart Filament untuk Statistik Pasien

Kelas WPasienChart adalah widget statistik pada Filament yang menampilkan jumlah pasien terdaftar pada sistem, dengan tampilan grafik linier. Data diambil dari tabel registrasis, dan pengguna bisa melihat statistik pasien secara mingguan, bulanan, atau tahunan.

2. Properti Utama Widget

- `protected static ?string $heading`: Judul widget, yaitu “Jumlah Pasien Terdaftar”.
- `protected int|string|array $columnSpan`: Mengatur lebar widget pada *Dashboard* (default 3 kolom).
- `protected static ?string $maxHeight`: Tinggi maksimal chart (default 250px).

3. Filter Data (getFilters())

- Widget menyediakan tiga filter tampilan data:
 - Mingguan (weekly): Data harian selama seminggu terakhir.
 - Bulanan (monthly): Data per bulan selama setahun terakhir.
 - Tahunan (yearly): Data per tahun selama lima tahun terakhir.
- Filter ini memudahkan admin untuk menganalisa tren pendaftaran pasien pada berbagai rentang waktu.

4. Pengolahan dan Pengambilan Data Chart (getData())

- Menentukan Filter:
Filter diambil dari pilihan pengguna; jika tidak ada, default ke "mingguan".
- Logika Query Berdasarkan Filter:
 - Mingguan:
 - Mulai dari hari Senin minggu ini.
 - Untuk 7 hari berturut-turut, label diisi nama hari dan tanggal (D, d M), data diisi jumlah pasien per tanggal (tanggal_konsultasi).
 - Bulanan:
 - Mulai dari awal tahun berjalan.
 - Untuk 12 bulan, label diisi nama bulan dan tahun (M Y), data diisi jumlah pasien setiap bulan di tahun berjalan.
 - Tahunan:
 - Untuk 5 tahun terakhir, label diisi tahun, data diisi jumlah pasien per tahun (berdasarkan tanggal_konsultasi).
- Data Format:
 - labels: List label waktu (hari, bulan, tahun) sesuai filter.
 - data: List jumlah pasien sesuai label (semua dikonversi ke integer).

- Return Data:

Disusun dalam format yang dapat dipakai langsung oleh Chart.js pada Filament:

- datasets: Label, data, warna garis, warna area, dan tingkat lengkungan garis (tension).
- labels: Label waktu.

5. Jenis Grafik (getType())

- Mengembalikan 'line', sehingga widget menampilkan grafik garis (line chart), cocok untuk visualisasi tren dari waktu ke waktu.

6. Konfigurasi Opsi Tampilan (getOptions())

- Axis Y:
 - Grafik selalu mulai dari 0 (beginAtZero).
 - Ticks diatur tanpa desimal (precision: 0) supaya angka bulat.
- Legend:
 - Ditampilkan di atas chart (position: 'top').
- Scalability:

Opsi ini memastikan chart tetap mudah dibaca, terutama saat data banyak (bulanan/tahunan).

7. Ringkasan Fungsi WPasienChart

- Monitoring Dinamis:

Memantau jumlah pasien yang terdaftar dan konsultasi dalam berbagai rentang waktu (mingguan, bulanan, tahunan).
- Visualisasi Tren:

Memudahkan admin untuk melihat pola naik-turun jumlah pasien, mendukung pengambilan keputusan berbasis data.

- Interaktif & Responsif:

Opsi filter membuat chart lebih interaktif. Tampilan otomatis menyesuaikan label dan data.

- Otomatis & Real-time:

Data diambil langsung dari database, selalu up-to-date sesuai data terbaru.

7. Api Rekomendasi Aktivitas

```

1 from flask import Flask, request, jsonify
2 from flask_cors import CORS
3 import joblib
4 import pandas as pd
5 import logging
6
7
8 logging.basicConfig(level=logging.INFO)
9 logger = logging.getLogger(__name__)
10
11
12 app = Flask(__name__)
13 CORS(app)
14
15
16 try:
17     preprocessor = joblib.load('preprocessor_knn.pkl')
18     knn_model = joblib.load('model_knn.pkl')
19
20
21     df = pd.read_excel('dataset_rekomendasi_medis_final.xlsx')
22     df.columns = ['penyakit', 'umur', 'sistolik', 'diastolik', 'rekomendasi']
23
24
25     from sklearn.model_selection import train_test_split
26     df_train, _ = train_test_split(df, test_size=0.2, random_state=42)
27
28     logger.info("Models and data loaded successfully")
29 except Exception as e:
30     logger.error(f"Error loading models: {e}")
31     raise
32
33
34 def recommend(pasien_dict, k=5):
35
36     df_filtered = df_train[df_train['penyakit'].str.lower() == pasien_dict['penyakit'].lower()]
37
38     if df_filtered.empty:
39         return [{
40             'penyakit': pasien_dict['penyakit'],
41             'rekomendasi': '',
42             'similarity': 0
43         }]
44
45
46     df_in = pd.DataFrame([pasien_dict])
47     X_in = preprocessor.transform(df_in)
48
49
50     X_filtered = preprocessor.transform(df_filtered[['penyakit', 'umur', 'sistolik', 'diastolik']])
51
52
53     from sklearn.metrics.pairwise import cosine_similarity
54     similarities = cosine_similarity(X_in, X_filtered)[0]
55
56
57     top_k_idx = similarities.argsort()[::-1][:k]
58
59     results = []
60     for idx in top_k_idx:
61         rec_row = df_filtered.iloc[idx]
62         results.append({
63             'penyakit': rec_row['penyakit'],
64             'umur': int(rec_row['umur']),
65             'sistolik': int(rec_row['sistolik']),
66             'diastolik': int(rec_row['diastolik']),
67             'rekomendasi': rec_row['rekomendasi'],
68             'similarity': float(similarities[idx])
69         })
70
71     return results
72

```

```

1 def validate_input(data):
2     required = ['penyakit', 'umur', 'sistolik', 'diastolik']
3     missing = [f for f in required if f not in data]
4     if missing:
5         return None, f"Missing fields: {', '.join(missing)}"
6     try:
7         return {
8             'penyakit': str(data['penyakit']).strip(),
9             'umur': int(data['umur']),
10            'sistolik': int(data['sistolik']),
11            'diastolik': int(data['diastolik'])
12        }, None
13     except Exception:
14         return None, "Invalid data types"
15
16
17 @app.route('/recommend', methods=['POST'])
18 def recommend_endpoint():
19     if not request.is_json:
20         return jsonify({'error': 'Request must be JSON'}), 400
21     data = request.get_json()
22     logger.info(f"Incoming payload: {data}")
23
24     payload, error = validate_input(data)
25     if error:
26         logger.warning(f"Validation failed: {error}")
27         return jsonify({'error': error}), 400
28
29     recs = recommend(payload, k=5)
30     logger.info(f"Recommendations for {payload}: {recs}")
31
32     return jsonify({'recommendations': recs})
33
34 if __name__ == '__main__':
35     app.run(host='0.0.0.0', port=5000)
36

```

Penjelasan Kode: Sistem Rekomendasi Medis Berbasis Flask

1. Deskripsi Umum

Kode ini adalah aplikasi web API berbasis Flask yang menyediakan endpoint `/recommend` untuk memberikan rekomendasi medis berdasarkan data input pasien. Sistem ini memanfaatkan model KNN yang sudah dilatih (file pickle) untuk menghitung kemiripan kasus pasien baru dengan data historis, lalu mengembalikan rekomendasi medis serupa.

2. Import dan Inisialisasi

- Flask: Framework web Python untuk membangun API.
- CORS: Mengizinkan akses API dari domain lain (untuk kebutuhan front-end yang berbeda origin).

- `joblib`: Untuk memuat model machine learning yang sudah disimpan (`preprocessor_knn.pkl` dan `model_knn.pkl`).
- `pandas`: Pengolahan data tabular, terutama data pasien dan rekomendasi.
- `logging`: Untuk logging aktivitas sistem dan debugging.

3. Load Model, Preprocessor, dan Data

- Sistem mencoba memuat:
 - Preprocessor: Pipeline untuk preprocessing fitur (encoding, scaling, dsb).
 - Model KNN: Model K-Nearest Neighbors (meskipun di kode hanya digunakan fitur similaritas, bukan predict).
 - Data: File Excel dataset medis berisi kolom penyakit, umur, sistolik, diastolik, dan rekomendasi.
- Data di-split: 80% untuk training, 20% sisanya tidak digunakan di API ini.
- Jika ada error pada saat load, sistem akan logging error dan menghentikan proses.

4. Fungsi Utama: `recommend(pasien_dict, k=5)`

- Filter Kasus Relevan: Data latih (`df_train`) difilter agar hanya mengambil kasus dengan penyakit yang sama dengan pasien input.
- Jika Tidak Ada Kasus Relevan: Mengembalikan hasil kosong, dengan similarity 0.
- Transformasi Data:
 - Data pasien baru di-preprocess sesuai pipeline yang digunakan saat training.
 - Data filter kasus juga dipreprocess agar bisa dibandingkan secara numerik.
- Perhitungan Similaritas:
 - Menggunakan cosine similarity antar pasien input dan seluruh kasus pada penyakit yang sama.

- Mengambil top-k kasus dengan nilai similarity tertinggi.
- Hasil Rekomendasi:
 - Dikembalikan list berisi penyakit, umur, sistolik, diastolik, rekomendasi medis, dan skor similarity.

5. Validasi Input: `validate_input(data)`

- Mengecek field wajib: penyakit, umur, sistolik, diastolik.
- Jika ada field yang kurang, error.
- Jika tipe data salah (misal umur bukan angka), juga error.
- Jika lolos validasi, input dirapikan (string strip, numerik jadi int).

6. Endpoint API: `/recommend`

- HTTP Method: POST, hanya menerima input JSON.
- Proses:
 1. Cek apakah payload JSON valid.
 2. Validasi input (cek field dan tipe data).
 3. Jika valid, proses rekomendasi dengan fungsi `recommend`.
 4. Hasil dikembalikan dalam format JSON (field `recommendations`).
 5. Logging aktivitas untuk setiap request masuk dan hasil keluar.

Ringkasan Kegunaan

- Sistem ini dapat diintegrasikan dengan aplikasi front-end (misal React, mobile, dsb) untuk memberikan rekomendasi medis berbasis pengetahuan kasus terdahulu.
- Sangat cocok untuk Clinical Decision Support berbasis kemiripan kasus dan histori data pasien.
- Input dan output sudah distandarisasi dalam format JSON, memudahkan integrasi dengan sistem lain.

Lampiran 6 Sertifikat Hki

 REPUBLIK INDONESIA KEMENTERIAN HUKUM	
SURAT PENCATATAN CIPTAAN	
Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:	
Nomor dan tanggal permohonan	: EC002025075598, 25 Juni 2025
Pencipta	
Nama	: Rama Oktabara, Ginanjar Wiro Sasmito dkk
Alamat	: Jl. Jati Raya No. 157, Griya Pelutan Indah, RT/RW(03/03) 52311, Kel. Pelutan, Kec. Pemalang, Kab. Pemalang, Prov. Jawa Tengah, Pemalang, Kab. Pemalang, Jawa Tengah, 52311
Kewarganegaraan	: Indonesia
Pemegang Hak Cipta	
Nama	: Pusat Penelitian dan Pengabdian Masyarakat (P3M) Politeknik Harapan Bersama
Alamat	: Jalan Mataram No. 9, Pesurungan Lor, Kecamatan Margadana, Margadana, Kota Tegal, Jawa Tengah, 52142
Kewarganegaraan	: Indonesia
Jenis Ciptaan	: Program Komputer
Judul Ciptaan	: Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Menggunakan Algoritma K-Nearest Neighbor
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	: 24 Juni 2025, di Kota Tegal
Jangka waktu perlindungan	: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor Pencatatan	: 000915859
adalah benar berdasarkan keterangan yang diberikan oleh Pemohon. Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.	
	a.n. MENTERI HUKUM DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL u.b Direktur Hak Cipta dan Desain Industri
	Agung Damarsasongko, SH., MH. NIP. 196912261994031001
	Disclaimer: 1. Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan. 2. Surat Pencatatan ini telah disegel secara elektronik menggunakan segel elektronik yang diterbitkan oleh Balai Besar Sertifikasi Elektronik, Badan Siber dan Sandi Negara. 3. Surat Pencatatan ini dapat dibuktikan keasliannya dengan memindai kode QR pada dokumen ini dan informasi akan ditampilkan dalam browser.

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Rama Oktabara	Jl. Jati Raya No.157, Griya Pelutan Indah,RT/RW(03/03) 52311, Kel.Pelutan,Kec.Pemalang. Kab.Pemalang, Prov.Jawa Tengah Pemalang, Kab. Pemalang
2	Ginanjari Wiro Sasmito	Desa Kluwut RT.003/002, Kecamatan Bulakamba, Kabupaten Brebes, Provinsi Jawa Tengah Bulakamba, Kab. Brebes
3	Dega Surono Wibowo	Perumahan Sapphire Regency Blok II, No.1, RT.004, RW.001, Kelurahan Pulosari, Kecamatan Brebes Brebes, Kab. Brebes



Lampiran 7 Lembar Bimbingan



**D IV TEKNIK INFOTMATIKA
POLITEKNIK HARAPAN BERSAMA**

LEMBAR BIMBINGAN TUGAS AKHIR

Nama : Rama Oktabara
NIM : 21090008
No. Ponsel : 0895401277019
Judul Ta : Sistem Rekam Medis Elektronik(RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN

Dosen Pembimbing II : Dega Surono Wibowo, S.T., M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1.	19/3 2025	a. Data primer & sekunder b. Usecase diperbaiki	Logatung dengan data sekunder . .	f
2	24/4 2025	a. Perbaiki akses data cet b. Langsung develop UI/UX dan aplikasi sambil cek form (selenium / -).		f
3.	23/5 2025	a. Lanjut aplikasi user.		f
4.	28/5 2025	a. test aplikasi pakai selenium / katalon b. bikin manual book dan dokumen teknis		f

5.	5 Juni 2015	* silalah lura dptar Hui.	
6.	30/6 2015	Cikri & laporan bab 1 dan 2 - format penulisan - kata-kata cetak miring - sitasi IEEE - tabel.	
7.	7/7 2015	Bab 1 dan Bab 2. con-	
8.	8/7 2015	Class Diagram Jarga Hui Hitam	

Tegal, 9/7 2015
Dosen Pembimbing II

Dega Suvono Wibowo, S.T., M.Kom.
NIPY.06.014.183



D IV TEKNIK INFOTMATIKA
POLITEKNIK HARAPAN BERSAMA

LEMBAR BIMBINGAN TUGAS AKHIR

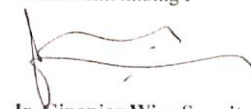
Nama : Rama Oktabara
NIM : 21090008
No. Ponsel : 0895401277019
Judul Ta : Sistem Rekam Medis Elektronik (RME) Berbasis Web Dengan Fitur Rekomendasi Aktivitas Pasien Dengan Menggunakan Algoritma KNN

Dosen Pembimbing I : Ir.Ginanjari Wiro Sasmito,S.Kom.,M.Kom.

No	Tanggal	Pemeriksaan	Perbaikan Yang Perlu Dilakukan	Paraf Pembimbing
1	17/25 3	benar.	Bussiness process	f.
2	21/25 4	produk.	refor x 2 RHR	f.
3	23/25 5	produk	Langutan data register/ pasien - laporan	f.
4	28/25 65	produk	- DUE. - fi/king Pelayanan - lanjut draft HCI - draft jurnal	f.

26/25 /6	Bul 1	Revisi	f
30/25 /6	Laporan	Revisi	f
1/25 /7	Laporan	Revisi	f
8/25 /7	Laporan	Revisi	f
.	Laporan	Revisi Siap ujian	f

Tegal, 9/7 2025
Dosen Pembimbing I



Ir. Ginanjar Wiro Sasmito, S.Kom., M.Kom.
NIPY 10 007 032