

LAMPIRAN

Lampiran 1 Surat Balasan Observasi



Pimpinan Daerah Aisyiyah Kabupaten Tegal Majelis Kesehatan Klinik Pratama Aisyiyah Hj Mafroh

Jl. Pancasila No 35 Grogol Dukuhturi Kabupaten Tegal Telp : 0283 321920

Nomor : 30/VI-2/E/KAHM/2025
Lamp : ---
Perihal : Persetujuan Tempat Penelitian

Tegal, 30 April 2025

Kepada Yth
Kepala Prodi Teknik Informatika
Politeknik Harapan Bersama
Di Tempat

Assalamu'alaikum Wr Wb

Dengan hormat,

Berdasarkan surat dari Politeknik Harapan Bersama nomor 45.03/TL.PHB/TV/2025

tanggal 25 April 2025 perihal permohonan tempat penelitian atas nama :

Nama : M Roy Samsul Akbar
Nim : 21090123
Prodi : Sarjana Terapan Teknik Informatika

Dengan ini kami beritahukan bahwa yang bersangkutan dapat kami terima untuk melakukan penelitian di Klinik Pratama Aisyiyah Hj Mafroh

Demikian surat persetujuan ini kami buat. Atas kerjasama Bapak/Ibu kami mengucapkan terimakasih.

Wassalamu'alaikum Wr Wb

Pimpinan Daerah Aisyiyah Kabupaten Tegal
Majelis Kesehatan
Klinik Pratama Aisyiyah "Hj Mafroh"

Direktur

(*dr. Moh. Abdurrohman, MARS*)

NIP/NRP : 200519852023

Lampiran 2 Surat Ijin Observasi



POLITEKNIK HARAPAN BERSAMA
The True Educational Campus

Sarjana Terapan Teknik Informatika

Nomor : 45.03/TI.PHB/IV/2025
Lampiran : -
Hal : Permohonan Peminjaman
Kepada :
Yth. : **Pemilik Klinik Pratama Aisyah Hj. Mafroh**
Di Tempat

Dengan hormat, mahasiswa dengan identitas berikut ini:

nama : M Roy Samsul Akbar
NIM : 21090123
prodi : Sarjana Terapan Teknik Informatika

Bermaksud melakukan penelitian untuk keperluan Skripsi dengan judul **"Rancang Bangun Aplikasi Pendaftaran Pasien Online Berbasis Website"**. Kami memohon Bapak/Ibu memberikan izin kepada mahasiswa yang bersangkutan agar dapat memperoleh data, keterangan, dan bahan yang diperlukan.

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

Tegal, 25 April 2025
Ka. Prodi S.Tr. Teknik Informatika,

Dyan Aprilliadi, S.T., M.Kom
NIPY : 09.015.225

Lampiran 3 Surat Kesepakatan Bimbingan Skripsi

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini:

Pihak Pertama

Nama : M. Roy Samsul Akbar
NIM : 21090123
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Dwi Intan Af'idah, S.T., M.Kom.
Status : Dosen Tetap
NIDN : 0620089203
Jabatan Fungsional : Lektor
Pangkat/Golongan : Penata III-C

Pada hari ini Jum'at tanggal 16 Mei 2025 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing I Skripsi Pihak Pertama dengan syarat Pihak Pertama wajib melakukan bimbingan Skripsi minimal 2 minggu sekali kepada Pihak kedua. Adapun waktu dan tempat pelaksanaan disepakati antar pihak. Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Tugas Akhir

Tegal, 16 Mei 2025

Pihak Pertama,



M. Roy Samsul Akbar

Pihak Kedua,



Dwi Intan Af'idah, S.T., M.Kom.

Mengetahui,
Ka. Prodi Sarjana Terapan Teknik Informatika



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

SURAT KESEPAKATAN BIMBINGAN SKRIPSI

Kami yang bertanda tangan di bawah ini:

Pihak Pertama

Nama : M. Roy Samsul Akbar
NIM : 21090123
Program Studi : Sarjana Terapan Teknik Informatika

Pihak Kedua

Nama : Dairoh, S.Si., M.Sc.
Status : Dosen Tetap
NIDN : 0612108701
Jabatan Fungsional : Lektor
Pangkat/Golongan : Penata III-D

Pada hari ini Jum'at tanggal 16 Mei 2025 telah terjadi sebuah kesepakatan bahwa Pihak Kedua bersedia menjadi Pembimbing II Skripsi Pihak Pertama dengan syarat adanya peningkatan progres dari proyek skripsi yang dilaporkan setiap satu pekan. Jika Pihak 1 tidak memenuhi syarat tersebut, maka pihak 2 berhak memutuskan kesepakatan ini. Demikian kesepakatan ini dibuat dengan penuh kesadaran guna kelancaran penyelesaian Tugas Akhir

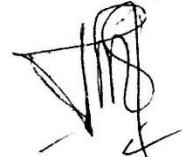
Tegal, 16 Mei 2025

Pihak Pertama,



M. Roy Samsul Akbar

Pihak Kedua,



: Dairoh, S.Si., M.Sc.

Mengesahui,
Ka. Prodi Sarjana Terapan Teknik Informatika



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

Lampiran 4 Surat Pernyataan HKI

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini :

1. N a m a : M. Roy Samsul Akbar
Kewarganegaraan : Indonesia
Alamat : Desa Grogol RT/RW 007/003, Kecamatan Dukuhturi,
Kabupaten Tegal, Jawa Tengah 52192
2. N a m a : Dwi Intan Afidah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Jalan Grinting RT 003/RW 002, Kec. Bulakamba, Kab. Brebes 52253
3. N a m a : Dairoh, S.Si., M.Sc.
Kewarganegaraan : Indonesia
Alamat : Jalan Garuda No 03 RT 002/RW 009 Randugunting, Kec. Tegal
Selatan, Kota Tegal 52131

Dengan ini menyatakan bahwa:

1. Karya Cipta yang saya mohonkan:
Berupa : Program Komputer
Berjudul : Aplikasi Pendaftaran Pasien Online Berbasis Web Di Klinik Pratama Aisyiyah Haji Mafroh
 - 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 berperkara 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, 10 Juli 2025



(M. Roy Samsul Akbar)

Handwritten signature of Dwi Intan Af'idah in blue ink.

(Dwi Intan Af'idah, S.T., M.Kom.)

Handwritten signature of Dairoh in black ink.

(Dairoh, S.Si., M.Sc.)

Lampiran 5 Surat Pengalihan HKI

SURAT PENGALIHAN HAK CIPTA

Yang bertanda tangan di bawah ini :

1. N a m a : M. Roy Samsul Akbar
Kewarganegaraan : Indonesia
Alamat : Desa Grogol RT/RW 007/003, Kecamatan Dukuhturi,
Kabupaten Tegal, Jawa Tengah 52192
2. N a m a : Dwi Intan Af'idah, S.T., M.Kom.
Kewarganegaraan : Indonesia
Alamat : Jalan Grinting RT 003/RW 002, Kec. Bulakamba, Kab. Brebes 52253
3. N a m a : Dairoh, S.Si., M.Sc.
Kewarganegaraan : Indonesia
Alamat : Jalan Garuda No 03 RT 002/RW 009 Randugunting, Kec. Tegal Selatan, Kota Tegal
52131

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

Nama : Pusat Penelitian dan Pengabdian Masyarakat (P3M)
Alamat : Jl. Mataram No. 9 Pesurungan Lor Kota Tegal

Adalah **Pihak II** selaku Pemegang Hak Cipta berupa Program Komputer dengan judul "**Aplikasi Pendaftaran Pasien Online Berbasis Website Di Klinik Pratama Aisyiyah Haji. Mafroh** ". 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.

Pemegang Hak Cipta
Kepala P3M



(Muhammad Fikri Hidayattullah, S.T., M.Kom.)

Tegal, 10 Juli 2025
Pencipta



(M. Roy Samsul Akbar)



(Dwi Intan Af'idah, S.T., M.Kom.)



(Dairoh, S.Si., M.Sc.)

Manual Book

APLIKASI PENDAFTARAN PASIEN ONLINE BERBASIS WEBSITE

oleh :

**M Roy Samsul Akbar
Dwi Intan Af'idah, S.T., M.Kom.
Dairoh, S.Si., M.Sc.**

1. PENDAHULUAN

1.1. Tujuan Pembuatan Dokumen

Tujuan dibuatnya dokumen petunjuk penggunaan *website* Aplikasi Pendaftaran Pasien *Online* berbasis *Website* yang dirancang untuk Klinik Pratama Aisyiyah Hj.. Mafroh ini dibuat untuk tujuan sebagai berikut :

1. Petunjuk Penggunaan ini digunakan untuk menjelaskan kepada pengguna (*User/admin*) mengenai penggunaan Aplikasi Pendaftaran Pasien *Online* berbasis *Website*.
2. Untuk menggambarkan penggunaan Aplikasi Pendaftaran Pasien *Online* berbasis *Website*.
3. Menyediakan informasi teknis tentang kebutuhan aplikasi dan alur penggunaannya.

1.2. Deskripsi Umum Sistem

1.2.1. Deskripsi Umum Aplikasi

Aplikasi pendaftaran pasien *online* Klinik Pratama Aisyiyah Hj.. Mafroh merupakan sistem berbasis *website* yang bertujuan mengurangi antrian dan meningkatkan efisiensi proses pendaftaran di fasilitas kesehatan. Aplikasi ini memungkinkan pasien untuk :

1. Melakukan *Registrasi* secara mandiri melalui perangkat elektronik.
2. Melihat estimasi waktu tunggu atau nomor antrian.

1.2.2. Deskripsi Umum Kebutuhan Aplikasi

Aplikasi ini dikembangkan dengan mempertimbangkan kebutuhan utama berikut:

➤ Pengguna Utama:

- a. Pasien : Dapat mendaftar secara *online*, mengisi data diri, jadwal dokter dan memilih poli, , dan mendapatkan nomor antrian.

- b. Admin : Memverifikasi data pasien dan mengelola jadwal dokter dan antrian.
- c. Dokter : Melihat daftar pasien yang telah terdaftar sesuai poli masing-masing dan rekam medis

1.2.3. Deskripsi Dokumen

Dokumen ini dibuat untuk memberikan panduan penggunaan Aplikasi Pendaftaran Pasien *Online* berbasis *Website* di Klinik Pratama Aisyiyah Hj.. Mafroh. Dokumen ini berisi informasi sebagai berikut :

a. BAB I

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

b. BAB II

Berisi perangkat yang dibutuhkan untuk pembuatan Aplikasi Pendaftaran Pasien Online berbasis *Website* di Klinik Pratama Aisyiyah Hj. Mafroh meliputi perangkat lunak (*Software*) dan perangkat keras (*Hardware*).

c. BAB III

Berisi panduan penggunaan (*Manual Book*) Aplikasi Pendaftaran Online berbasis *Website* di Klinik Pratama Aisyiyah Hj.. Mafroh.

2. PERANGKAT YANG DIBUTUHKAN

2.1. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan :

- | | |
|---|------------------------------|
| 1. <i>Windows</i> sebagai <i>Operating System</i> . | 4. <i>Visual Studio Code</i> |
| 2. <i>Google Chrome</i> | 5. <i>Laravel 11</i> |
| 3. <i>Laragon</i> | 6. <i>Selenium</i> |

2.2. Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan meliputi :

1. Laptop Asus
2. Ram 8 GB
3. Storage 512 GB

2.1 Penggunaan Aplikasi

Sasaran pengguna aplikasi yang dapat menggunakan Aplikasi Pendaftaran Pasien *Online* Berbasis *Website* di Klinik Pratama Aisyiyah Hj.. Mafroh ini sebagai berikut:

1. Masyarakat kota Tegal.
2. Owner / pemilik klinik pratama asiyiyah Hj.. mafroh.

3. FITUR DAN CARA PENGGUNAAN

3.1. Struktur Menu

Adapun struktur halaman pada aplikasi penggunaan *website* Aplikasi Pendaftaran *Online* berbasis *Website* berdasarkan peran pengguna sebagai berikut :

1. Menu User (Pengguna)

Adapun struktur menu pada Aplikasi Pendaftaran Pasien *Online* Berbasis *Website* di Klinik Pratama Aisyiyah Hj.. Mafroh adalah sebagai berikut :

- *Landing page*
- *Tentang Kami*
- *Pengisian form*
- *Kontak Kami*
- *Cek antrean*

2. Menu Admin (Perawat)

- *Register*
- *Registrasi pasien*
- *Login*
- *Jadwal dokter*
- *Dashboard*
- *Kajian awal*
- *Rekam medis*
- *Antrian pasien*
- *Data Pasien*
- *Laporan*
- *Data Dokter*
- *Logout*

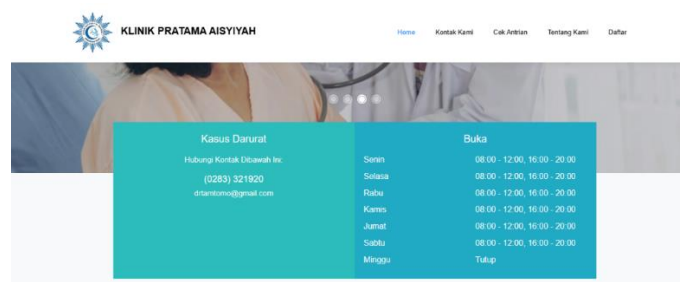
3.2. Pengguna

Bagian ini merupakan panduan penggunaan aplikasi dengan menggunakan gambar dan penjelasan mengenai fitur – fitur yang tersedia pada aplikasi dan panduan pengimputan data ke dalam aplikasi tersebut.

1. Cara Membuka Aplikasi

Berikut cara penggunaan Aplikasi Pendaftaran Pasien *Online* Berbasis *Website* Di Klinik Pratama Aisyiyah Hj. Mafroh :

1. Pilih *browser* untuk membuka aplikasi, masukan alamat *url* (..).
Kemudian tekan *enter* agar dapat masuk ke *website* Klinik Pratama Aisyiyah Hj. Mafroh.
2. Setelah berhasil masuk ke alamat url, maka akan muncul tampilan *landing pages* yang berisi *navbar* tentang kami, kontak kami, cek antrian. Berikut merupakan tampilan *landing pages* dapat dilihat pada gambar 3.1 dibawah ini:



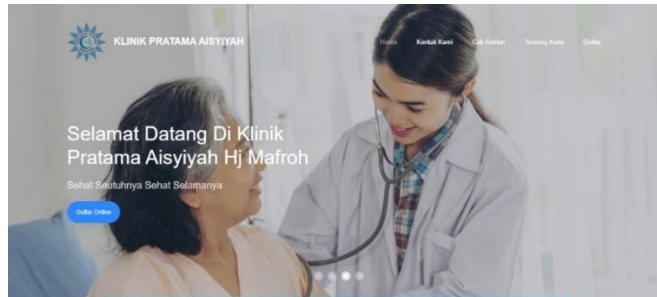
2. Halaman Pengguna

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

a. Landing Pages

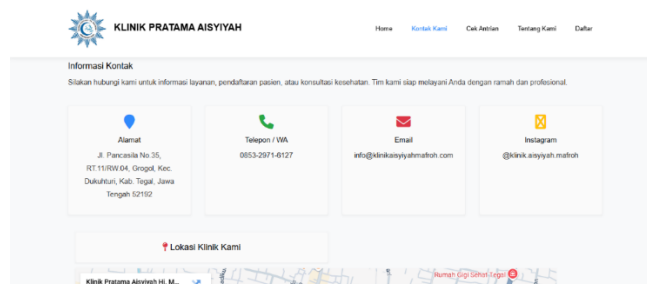
Halaman utama pengguna saat pertama kali mengakses *website* klinik. Halaman ini menampilkan informasi penting seperti

profil klinik, jadwal operasional, kontak darurat, layanan poli yang tersedia, daftar dokter, serta fasilitas klinik



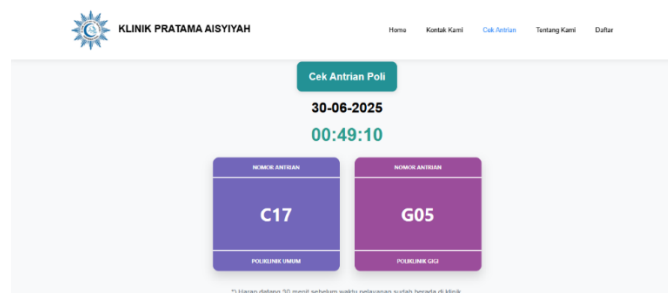
b. Fitur Kontak Kami

Fitur Kontak Kami menampilkan informasi lengkap untuk menghubungi klinik, termasuk alamat, nomor telepon/*WhatsApp*, *email*, dan akun media sosial. Halaman ini juga dilengkapi dengan peta lokasi interaktif dari *Google Maps* untuk memudahkan pasien menemukan rute menuju klinik.



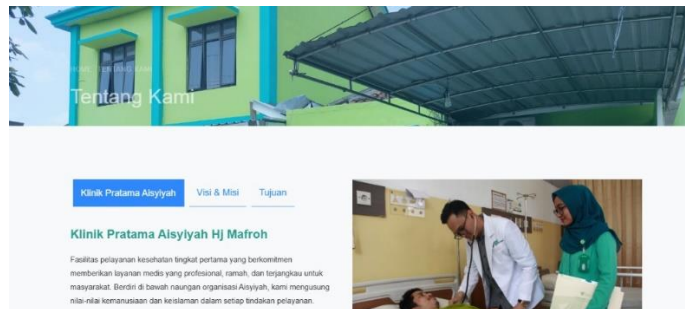
c. Fitur Cetak Antrian

Fitur cek antrian menampilkan untuk melihat status antrian pengguna atau *user* secara *real-time*. Informasi yang ditampilkan mencakup posisi antrian, estimasi waktu layanan.



d. Fitur Tentang Kami

Fitur tentang kami berfungsi untuk mengenalkan Klinik Pratama Aisyiyah Hj Mafroh secara umum kepada pengunjung *website*. Di dalamnya terdapat informasi mengenai latar belakang klinik, visi misi, tujuan, hingga sambutan dari direktur utama.



e. Fitur Registrasi

Fitur *Registrasi* digunakan oleh pasien untuk mengisi data diri saat melakukan pendaftaran online. Form ini mencakup identitas pasien, seperti nama, NIK, tanggal lahir, dan status pasien (baru atau lama).

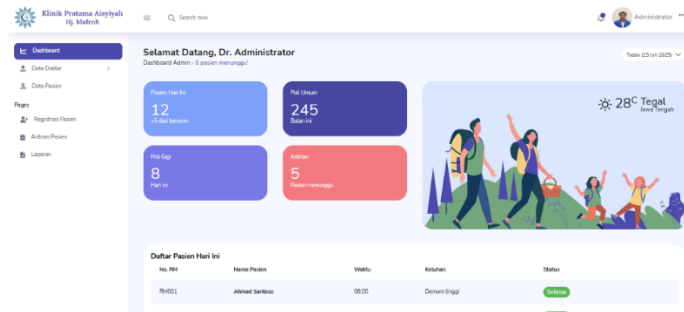
3. Halaman Admin

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

a. *Dashboard Admin*

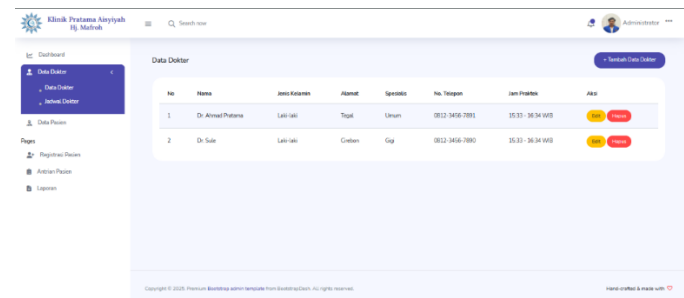
Fitur *Dashboard* admin menampilkan ringkasan data harian klinik secara *real-time*. Seperti jumlah pasien hari ini, total

kunjungan poli, jumlah antrean, serta daftar pasien yang sudah dan sedang dilayani, lengkap dengan keluhan dan statusnya, dapat dilihat gambar dibawah ini



b. Data Dokter

Fitur data dokter menampilkan dan mengelola informasi seluruh dokter yang bertugas di klinik, yang mencakup nama, jenis kelamin, alamat, spesialis, nomor telepon, dan jam praktik, dapat dilihat gambar dibawah ini.



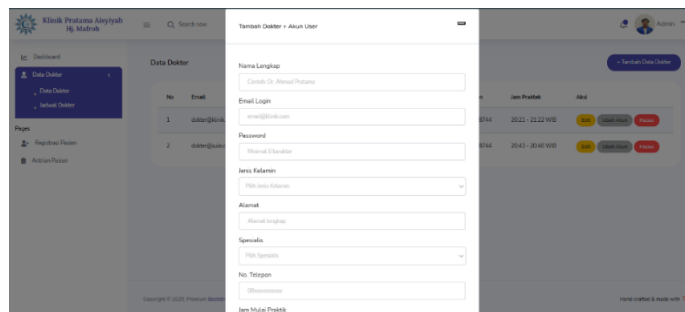
Pada halaman Data Dokter, admin diberikan kontrol penuh terhadap data dokter yang terdaftar. Setiap baris data dokter memiliki empat tombol aksi yaitu :

1. Button Tambah Dokter

Langkah Menambahkan Data Dokter :

1. Klik tombol “+ **Tambah Data Dokter**” di kanan atas.
2. Isi seluruh kolom sesuai form :
 - a. Nama lengkap dokter
 - b. Email *Login* untuk akun dokter
 - c. Password *Login*

- d. Jenis kelamin
 - e. Alamat lengkap
 - f. Spesialisasi yang dipilih
 - g. Nomor telepon aktif
 - h. Jam mulai & jam selesai praktik
3. Klik **“Simpan”** Akun otomatis dibuat dan data tersimpan ke sistem. Dapat dilihat pada gambar dibawah ini



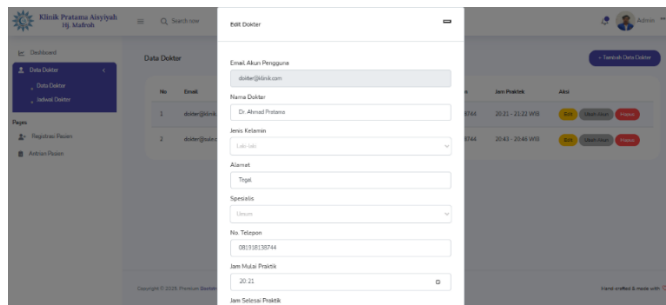
2. Button *Edit*

Fungsi: Untuk mengubah data umum dokter yang sudah terdaftar.

Langkah Mengedit Data Dokter :

1. Klik tombol **“Edit”** (warna kuning).
2. Form akan muncul dengan data dokter yang bisa diedit seperti:
 - a. Nama lengkap
 - b. Jenis kelamin
 - c. Alamat
 - d. Spesialis
 - e. Nomor telepon
 - f. Jam praktik
3. Setelah selesai mengedit, klik tombol **“Simpan”** untuk menyimpan perubahan.

Catatan: *Email Login* dan *password* tidak bisa *diedit* di sini, gunakan tombol Ubah Akun. Dapat dilihat pada gambar dibawah ini.



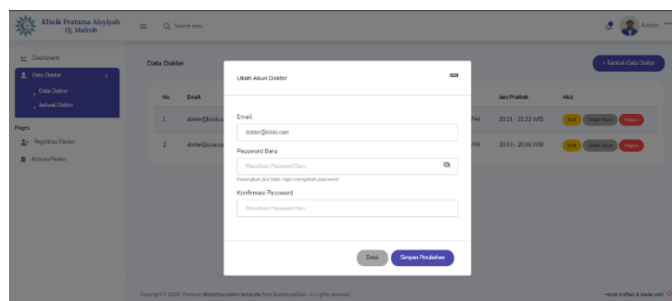
3. Button Ubah Akun

Fungsi: Untuk mengubah data *Login* dokter, seperti email dan password.

Langkah Mengubah Akun Dokter :

1. Klik tombol **“Ubah Akun”** (warna abu-abu).
2. Form akan tampil untuk mengubah:
 - *Email Login* baru
 - *Password* baru
3. Setelah selesai, klik tombol **“Simpan”**.

Dapat dilihat pada gambar dibawah ini.



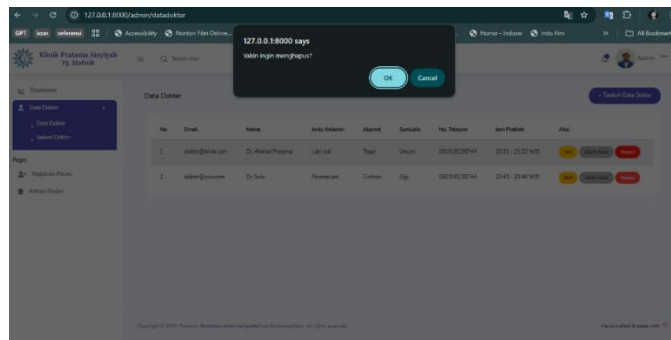
4. Button Hapus

Fungsi: Untuk menghapus data dokter beserta akun *Login* dari sistem.

Langkah Hapus Data Dokter :

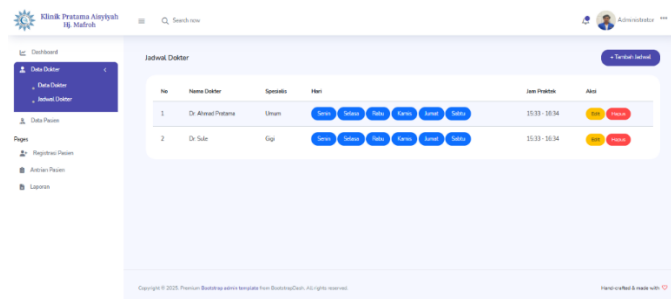
1. Klik tombol **“Hapus”** (warna merah).
2. Akan muncul konfirmasi penghapusan.
3. Klik **“Ya, Hapus”** untuk menyetujui penghapusan.

Dapat dilihat pada gambar dibawah ini :



c. Data Jadwal Dokter

Fitur jadwal dokter menampilkan dan mengelola jadwal praktik dokter berdasarkan hari dan jam kerja. Fitur ini membantu memastikan ketersediaan dokter selalu ter-update sehingga layanan pasien berjalan lancar dan sesuai waktu, dapat dilihat gambar dibawah ini.



Pada halaman Jadwal Dokter, admin diberikan kontrol penuh terhadap jadwal dokter yang terdaftar. Setiap baris jadwal dokter memiliki tiga tombol aksi yaitu :

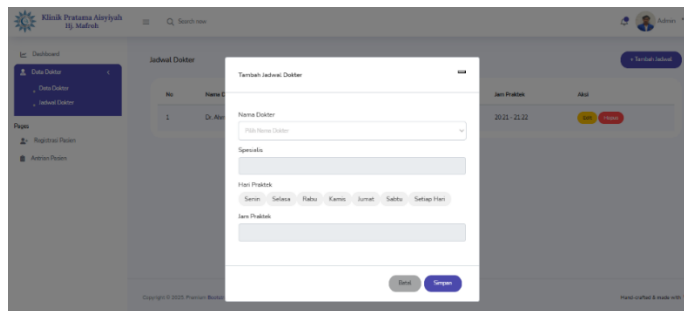
1. *Button* Tambah Jadwal

Fungsi : Menampilkan dan mengelola jadwal praktik dokter.

Langkah Menambahkan Jadwal Dokter :

1. Klik “+ **Tambah Jadwal**” di kanan atas, spesialis otomatis terisi.
2. Pilih hari praktik (Senin–Minggu atau Setiap Hari)
3. Isi Jam Praktik, kemudia klik “**Simpan**”.

Dapat dilihat pada gambar dibawah ini.



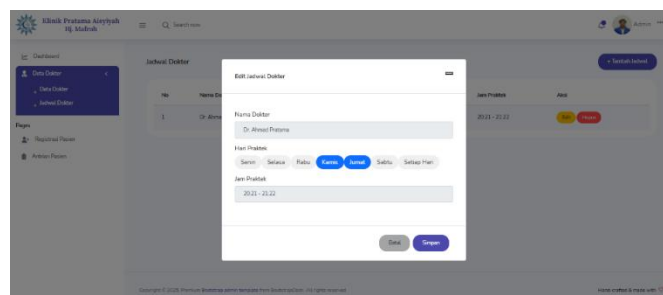
2. *Button Edit*

Button *Edit* digunakan untuk mengubah informasi jadwal praktik dokter yang sudah ada.

Langkah Mengedit Jadwal Dokter :

1. Klik tombol “**Edit**” di baris jadwal dokter yang ingin diubah.
2. *Form "Edit Jadwal Dokter"* akan muncul dalam modal *popup*.
3. Admin dapat mengubah:
 - a. Hari praktik (bisa pilih lebih dari satu)
 - b. Jam praktik (mulai dan selesai)
4. Klik “**Simpan**” untuk menyimpan perubahan.

Dapat dilihat pada gambar dibawah ini.



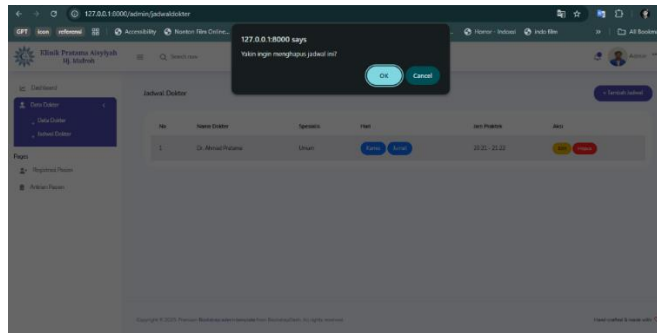
3. *Button Hapus*

Button Hapus berfungsi untuk menghapus jadwal praktik yang tidak berlaku lagi atau sudah tidak digunakan.

Langkah Menghapus Jadwal Dokter:

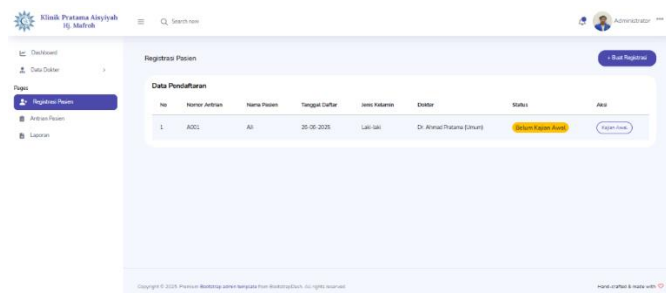
1. Klik *button* “**Hapus**” di baris jadwal dokter yang ingin dihapus.

2. Muncul konfirmasi (opsional: "Apakah Anda yakin ingin menghapus jadwal ini?").
 3. Klik **“Ya/Hapus”** untuk mengonfirmasi penghapusan.
- Dapat dilihat pada gambar dibawah ini.



d. *Registrasi Pasien*

Fitur *Registrasi* pasien menampilkan untuk mencatat data pendaftaran pasien baru, termasuk nomor antrian, nama, tanggal daftar, jenis kelamin, dan dokter tujuan. Setelah *Registrasi*, langsung melanjutkan ke pengisian kajian awal pasien, dapat dilihat gambar dibawah ini.



Pada halaman *Registrasi* Pasien, admin diberikan kontrol penuh terhadap *Registrasi* pasien yang terdaftar. Setiap baris *Registrasi* pasien memiliki dua tombol aksi yaitu :

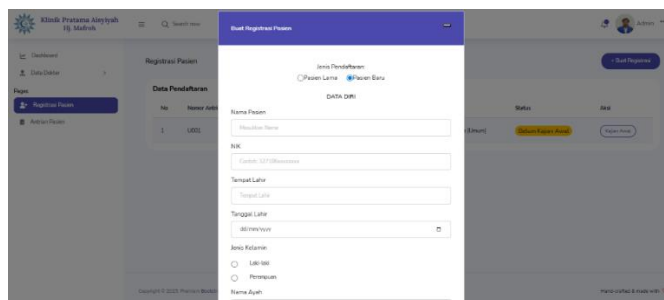
1. *Button Registrasi Pasien*

Fungsi : Untuk mencatat dan mengelola pendaftaran pasien, baik pasien baru maupun pasien lama, serta mencatat data kajian awal oleh admin sebelum pasien ditangani dokter.

Langkah *Registrasi* Pasien :

1. Klik *button* “+ **Buat Registrasi**” di kanan atas.
2. Pilih Jenis Pendaftaran:
 - Pasien Baru → isi seluruh data diri.
 - Pasien Lama → data otomatis terisi berdasarkan *database* pasien sebelumnya.
3. Isi form Data Diri, meliputi:
 - Nama Pasien
 - NIK
 - Tempat & Tanggal Lahir
 - Jenis Kelamin
 - Nama Ayah/Ibu
 - Alamat
 - Poli & Dokter Tujuan
4. Klik “**Simpan**” untuk mendaftarkan pasien.

Dapat dilihat pada gambar dibawah ini.



2. *Button* Kajian Awal

Setelah pasien terdaftar, akan muncul status Belum Kajian Awal. Dan klik *button* kajian awal di kolom aksi dan mengisi data pemeriksaan awal yang berisi :

- Suhu Tubuh
- Tekanan Darah

- Tinggi & Berat Badan
- Keluhan Pasien

Lalu klik **“Simpan Kajian”** untuk menyimpan hasil pemeriksaan. Dapat dilihat gamabar dibawah ini.

e. Antrian Pasien

Fitur antrian pasien menampilkan daftar pasien berdasarkan poli yang dituju, lengkap dengan nomor antrian, nama pasien, dokter, tanggal *Registrasi*, dan status pelayanan. Dan dapat memantau siapa saja yang masih menunggu atau sudah selesai, serta memanggil pasien secara langsung dari sistem, dapat dilihat gambar dibawah ini

Data Antrian						
Antrian Poli Umum						
No	Nomor Antrian	Nama Pasien	Nama Dokter	Tanggal Registrasi	Status	Aksi
1	A002	Budi Santoso	dr. Sari Pratama	2025-08-01	Menunggu	Detail
Antrian Poli Gigi						
No	Nomor Antrian	Nama Pasien	Nama Dokter	Tanggal	Status	Aksi
1	A001	Ayu Lestari	dr. Andi Wijaya	2025-08-01	Menunggu	Detail

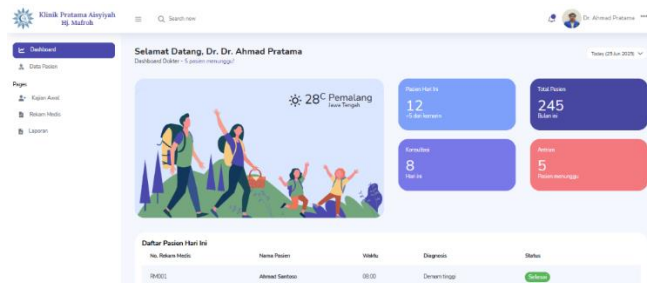
4. Halaman Dokter

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

a. Dashboard Dokter

Fitur *Dashboard* dokter menampilkan ringkasan aktivitas harian dokter seperti jumlah pasien, data konsultasi, dan antrian yang sedang

menunggu. Tersedia pula daftar pasien hari ini lengkap dengan jadwal, diagnosis, dan status layanan, dapat dilihat gambar dibawah ini.



b. Data Pasien

Fitur data pasien menampilkan daftar pasien yang sudah memiliki rekam medis lengkap dengan nama, jenis kelamin, alamat, dan nomor rekam medis, dapat dilihat gambar dibawah ini.

No	No. Rekam Medis	Nama Pasien	Jenis Kelamin	Alamat	Aksi
1	RM0010000001	Kendek	Laki-laki	Tegal	<button>Detail</button> <button>Edit</button>

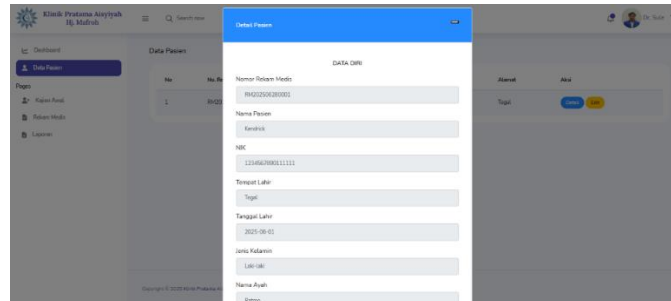
Pada halaman Data Pasien, dokter diberikan kontrol penuh terhadap data pasien yang terdaftar. Setiap baris data pasien memiliki dua tombol aksi yaitu :

1. *Button Detail*

Langkah menampilkan *detail* data pasien :

1. Klik tombol **“Detail”** pada baris pasien yang ingin dilihat.
2. Lalu akan muncul jendela *pop-up* berisi informasi lengkap pasien, seperti:
 - Nomor Rekam Medis
 - Nama, NIK, Tempat & Tanggal Lahir
 - Jenis Kelamin, Alamat
 - Nama Ayah & Ibu
 - Agama, Pendidikan, Pekerjaan, dan Status Perkawinan

3. Data ini hanya bisa dilihat, tidak dapat diedit. Dapat dilihat pada gambar dibawah ini.



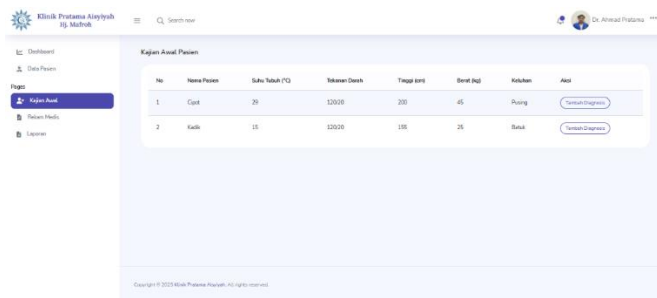
2. *Button Edit*

Langkah mengedit data pasien :

1. Klik *button Edit* pada pasien yang ingin diperbarui datanya.
2. Lalu akan muncul jendela form *edit*.
3. Silakan ubah data seperti:
 - Nama pasien
 - NIK
 - Tempat/Tanggal lahir
 - Agama, Pendidikan, Pekerjaan, Status, dll.
4. Pastikan semua kolom bertanda * diisi dengan benar.
5. Klik *button "Simpan"* untuk menyimpan perubahan.
6. Dan klik *button "Batal"* jika tidak ingin menyimpan.

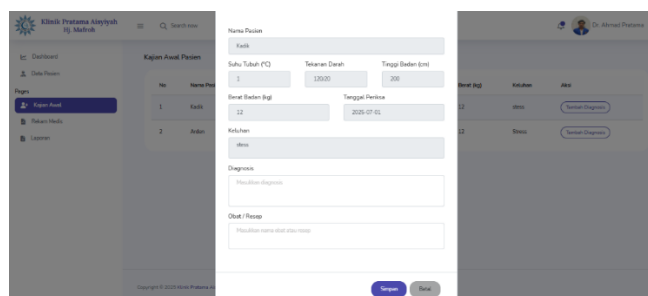
c. Kajian Awal

Fitur kajian awal pada halaman dokter digunakan untuk melihat hasil pemeriksaan awal pasien yang telah diinput oleh admin saat proses *Registrasi*. Di halaman ini, dokter dapat melakukan pemeriksaan lanjutan dengan menambahkan informasi diagnosis dan obat atau resep yang diberikan kepada pasien. Saat dokter masuk ke dalam sistem menggunakan akun yang telah terdaftar, dokter dapat mengakses menu Kajian Awal yang terletak pada bagian kiri layar (*sidebar*). Setelah memilih menu tersebut, akan tampil daftar pasien yang telah mengisi data awal melalui admin, termasuk data seperti suhu tubuh, tekanan darah, tinggi badan, berat badan, dan keluhan pasien.



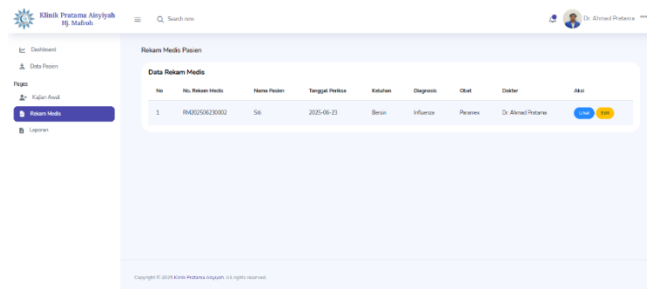
Langkah tambah diagnosis :

1. Klik *button* "**Tambah Diagnosis**" pada pasien yang akan diperiksa.
2. Form *pop-up* akan muncul dengan data pasien.
3. Isi Diagnosis dan Obat / Resep sesuai hasil pemeriksaan.
4. Klik "**Simpan**" untuk menyimpan data, atau klik "**Batal**" untuk membatalkan. Dapat dilihat pada gambar dibawah ini.



d. Rekam Medis

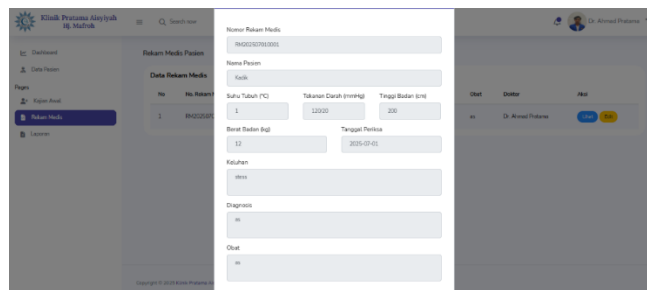
Fitur rekam medis pada dokter digunakan untuk menampilkan catatan medis setiap pasien yang telah ditangani. Untuk mengakses fitur ini, dokter hanya perlu memilih menu Rekam Medis pada *sidebar* kiri. Di halaman ini akan ditampilkan tabel yang berisi daftar rekam medis pasien, termasuk nomor rekam medis, nama pasien, tanggal periksa, keluhan, diagnosis, resep obat, dan nama dokter yang menangani. Setiap entri rekam medis memiliki dua tombol aksi, yaitu tombol Lihat dan Edit. Tombol **Lihat** digunakan untuk membuka *detail* lengkap rekam medis dalam tampilan modal tanpa mengubah data. Sementara tombol **Edit** memungkinkan dokter untuk memperbarui informasi diagnosis atau obat yang sebelumnya telah dicatat. Setelah perubahan dilakukan, klik tombol **Simpan** pada modal agar sistem menyimpan pembaruan secara permanen.



Langkah menampilkan data rekam medis di *button* Lihat :

1. Pada halaman Rekam Medis, cari baris data pasien yang ingin dilihat.
2. Klik tombol "**Lihat**" berwarna biru yang terletak pada kolom Aksi.
3. Setelah diklik, akan muncul modal berisi data lengkap seperti:
 - Nomor Rekam Medis
 - Nama Pasien
 - Tanggal Periksa
 - Keluhan
 - Diagnosis
 - Obat/Resep
 - Nama Dokter

Data yang tampil bersifat hanya baca (*read-only*), sehingga aman untuk ditinjau tanpa mengubah informasi apa pun. Modal ini dapat ditutup dengan klik tombol tutup (X) di kanan atas atau klik di luar area modal. Dapat dilihat pada gambar dibawah ini.



Langkah mengedit data rekam medis pada *button* Edit :

1. Buka halaman Rekam Medis pada menu *sidebar* kiri.
2. Cari data pasien yang ingin diperbarui.

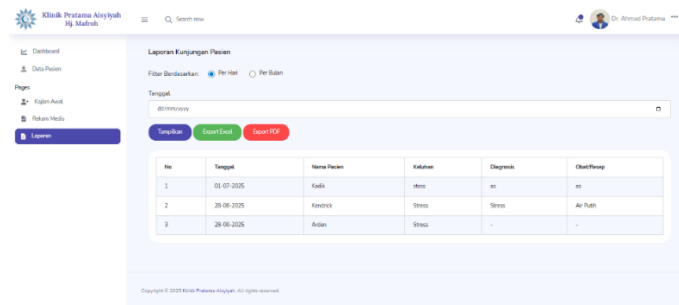
3. Klik tombol "*Edit*" berwarna kuning yang berada di kolom Aksi.
4. Setelah itu, akan muncul form dalam bentuk modal berisi data yang dapat diedit, seperti:
 - Diagnosis
 - Obat/Resep
5. Lakukan perubahan yang diperlukan, lalu klik tombol Simpan untuk memperbarui data rekam medis pasien.
6. Untuk membatalkan proses *edit*, klik tombol Batal. Dapat dilihat pada gambar dibawah ini.

e. Laporan

Fitur laporan kunjungan pasien pada role dokter digunakan untuk menampilkan data kunjungan pasien ke klinik berdasarkan data rekam medis yang sudah diisi sebelumnya. Dokter dapat memantau laporan kunjungan pasien sesuai periode tertentu untuk kebutuhan evaluasi atau dokumentasi. Pada halaman ini, dokter dapat memfilter laporan berdasarkan dua opsi yaitu Per Hari atau Per Bulan. Untuk memulai, dokter cukup memilih salah satu opsi filter, lalu memasukkan tanggal pada kolom yang tersedia. Setelah tanggal dipilih, klik tombol **Tampilkan** untuk memunculkan data pasien sesuai dengan periode yang dipilih. Setelah data tampil, dokter akan melihat tabel yang berisi informasi lengkap mengenai:

- Tanggal kunjungan
- Nama pasien
- Keluhan yang dikeluhkan pasien
- Diagnosis yang telah diberikan dokter

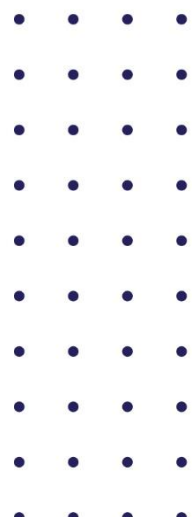
➤ Obat atau resep yang diberikan



The screenshot shows a web application for 'Klinik Pratama Alayyah Hj. Madroh'. The left sidebar contains navigation links: Dashboard, Data Pasien, Pages, Kelainan Awal, Riwayat Medis, and Laporan. The main area is titled 'Laporan Kunjungan Pasien' and includes a filter for 'Per Hari' (selected) and 'Per Bulan'. A date input field shows '01/07/2025'. Below the filters are three buttons: 'Tampilkan', 'Export Excel', and 'Export PDF'. A table displays patient visit data with columns: No, Tanggal, Nama Pasien, Keluhan, Diagnosis, and Obat/Resep.

No	Tanggal	Nama Pasien	Keluhan	Diagnosis	Obat/Resep
1	01-07-2025	Kadli	stres	ss	ss
2	28-06-2025	Kandisa	Stress	Stress	Air Putih
3	28-06-2025	Andan	Stress	-	-

Copyright © 2025 Klinik Pratama Alayyah. All rights reserved.

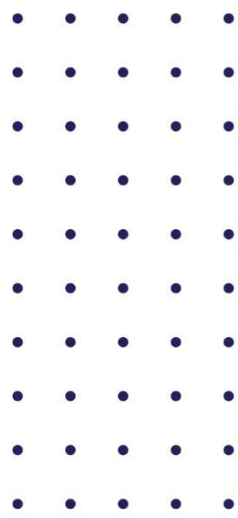


DOKUMEN TEKNIKAL

APLIKASI PENDAFTARAN PASIEN ONLINE BERBASIS WEBSITE

Oleh :

M. Roy Samsul Akbar
Dwi Intan Afidah, S.T., M.Kom.
Dairoh, S.Si, M.sc.



Profil

Aplikasi Pendaftaran Pasien *Online* Berbasis *Website* merupakan solusi digital yang dikembangkan untuk mempermudah masyarakat dalam melakukan proses pendaftaran layanan kesehatan di Klinik Pratama Aisyiyah Hj. mafroh. Aplikasi ini dirancang agar pasien dapat melakukan *Registrasi* secara mandiri kapan saja dan di mana saja tanpa harus datang langsung ke klinik hanya untuk mengantri. Melalui aplikasi ini, pasien dapat mengisi data pribadi, memilih dokter, serta menjadwalkan waktu konsultasi sesuai kebutuhan. Selain mengurangi kerumunan di area pendaftaran, sistem ini juga membantu mempercepat alur pelayanan, meningkatkan efisiensi administrasi, serta memberikan pengalaman yang lebih nyaman dan tertib bagi pasien.

Proses Pendaftaran dengan menggunakan aplikasi ini diharapkan pasien dapat melakukan pemeriksaan ke dokter tanpa harus datang lebih awal ke Klinik Pratama Aisyiyah Hj.. Mafroh. Selain itu pasien juga dapat mengetahui waktu kapan pasien yang bersangkutan dilakukan pemeriksaan oleh dokter karena dapat memantau proses antrian pasien langsung di *Smartphone* pasien masing-masing.

Latar Belakang

Penerapan Teknologi Informasi sudah masuk ke berbagai bidang dan mulai dirasakan manfaatnya. Hal ini mendorong beberapa pengembang teknologi Informasi untuk mengembangkan proses yang ada di perusahaan untuk memanfaatkan teknologi informasi. Salah satu bidang yang memanfaatkan teknologi ini adalah perusahaan di bidang Rumah Sakit/Klinik. Teknologi informasi digunakan dari berbagai sektor diantaranya administrasi rumah sakit, antrian pasien rumah sakit, manajemen pengelolaan rawat inap dan lain-lain. Dari beberapa sektor tersebut proses yang digunakan sebagai bahan penelitian mengenai pendaftaran pasien dan antrian pasien di klinik atau rumah sakit. Pemilihan ini berdasarkan beberapa kasus bahwa pasien harus datang ke klinik untuk melakukan pendaftaran dan menunggu dipanggil antrian oleh dokter. Pada saat pendaftaran, pasien tidak mengetahui berapa jumlah pendaftar yang telah daftar sehingga apabila sudah banyak maka pasien harus mencari klinik lain. Selain itu ketika menunggu antrian, pasien harus menunggu panggilan dari dokter dalam jangka waktu yang

lama misalkan harus menunggu dari jam 11.00 sedangkan dipanggil jam 16.00. Waktu yang digunakan untuk menunggu antrian tergolong lama.

Selama ini, pasien diharuskan datang langsung ke klinik untuk mendaftar dan menunggu antrian tanpa mengetahui jumlah pendaftar sebelumnya, sehingga sering kali terjadi penumpukan dan waktu tunggu yang lama. Dengan kondisi seperti ini maka diperlukan suatu penerapan teknologi informasi untuk menangani proses pendaftaran dan antrian di klinik. Masalah ini sangat penting untuk dijadikan sebagai penelitian karena pasien ketika mau datang ke dokter belum mengetahui berapa banyak pasien yang sudah terdaftar sebelumnya di hari tersebut dan memerlukan waktu yang lama untuk menunggu dipanggil oleh dokter karena tidak mengetahui status antrian sebelumnya. Dengan diterapkannya aplikasi untuk menangani masalah tersebut diharapkan dapat menyelesaikan masalah tersebut.

Manfaat

Penerapan aplikasi Pendaftaran Pasien *Online* Berbasis *Website* di Klinik Pratama Aisyiyah Hj. mafroh memberikan berbagai manfaat dalam meningkatkan efisiensi layanan kesehatan. Aplikasi ini mempermudah pasien dalam melakukan proses pendaftaran secara mandiri tanpa harus datang langsung ke klinik hanya untuk mengantri. Pasien dapat mengetahui jumlah pendaftar yang sudah terdaftar pada hari tersebut dan dapat memantau status antrian secara langsung melalui perangkat *smartphone* masing-masing.

Melalui sistem ini, pasien tidak perlu lagi datang lebih awal hanya untuk mendapatkan antrean, karena waktu pemeriksaan dapat diketahui lebih pasti. Hal ini secara langsung mengurangi kerumunan di area pendaftaran dan membuat proses pelayanan menjadi lebih tertib, nyaman, dan cepat. Bagi pihak klinik, aplikasi ini juga bermanfaat dalam mempercepat proses administrasi, meningkatkan akurasi data, serta membantu mengatur alur kunjungan pasien dengan lebih baik. Secara keseluruhan, aplikasi ini memberikan dampak positif dalam menciptakan sistem pelayanan kesehatan yang lebih efektif dan responsif terhadap kebutuhan pasien.

Spesifikasi Teknik

Spesifikasi teknik seperti :

1. *Manual Book*
2. *Source Code*

Berikut merupakan uraian spesifikasi untuk pembuatan aplikasi berbasis website :

1. *Framework Laravel*
2. *PHP*
3. *Laragon*
4. *Web Browser*
5. *Visual Studio Code*
6. *Mysql*

Berikut spesifikasi modul aplikasi :

1. Halaman Pengguna

- a. Halaman *Landing Pages*

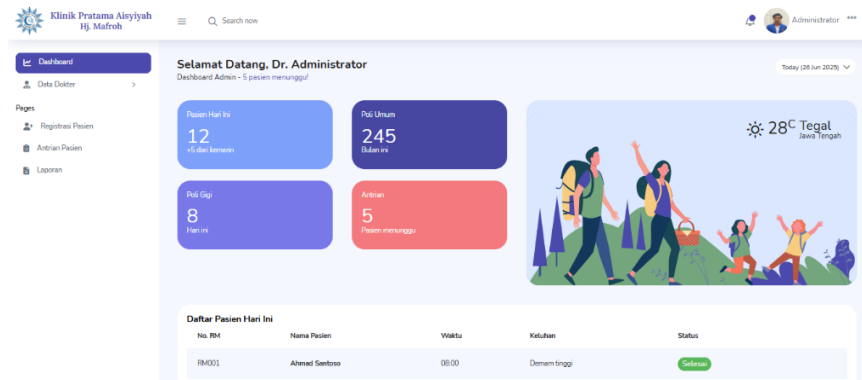
Bagian ini adalah aplikasi *website* klinik pratama aisiyah Hj.. mafroh bagian yang akan ditampilkan untuk pengguna beserta beragam fitur yang disediakan oleh aplikasi berupa *home*, tentang kami, cek antrean, kontak kami. Dapat dilihat gambar dibawah ini.



Gambar 1. 1 Halaman Landing Pages

- b. Halaman *Dashboard Admin*

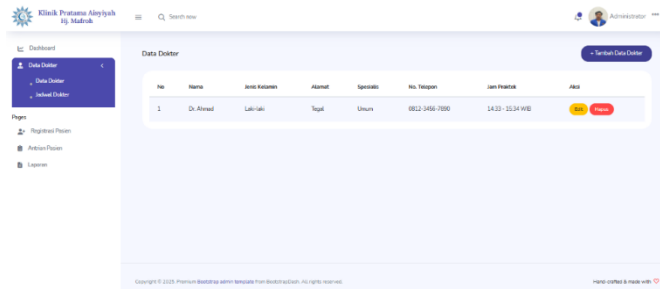
Bagian ini hanya dapat diakses oleh admin, bertujuan untuk mengelola dan memantau seluruh aktivitas dalam sistem, termasuk pengelolaan data dokter, jadwal dokter, *Registrasi* pasien, antrian pasien, laporan. Dapat dilihat gambar dibawah ini.



Gambar 1. 33 Halaman Dashboard Admin

c. Halaman Data Dokter

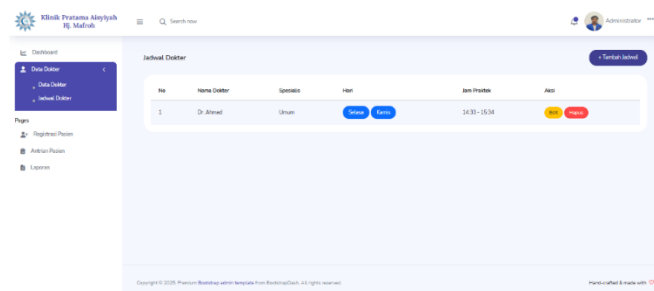
1. Menampilkan daftar data dokter dan *button edit* dan hapus.
2. Menampilkan *button* untuk tambah data dokter yang berisi form untuk tambah dokter seperti, akun dokter, nama dokter, jenis kelamin, Alamat, spesialis, nomer telepon, jam mulai, jam selesai.



Gambar 1. 65 Halaman Data Dokter

d. Halaman Jadwal Dokter

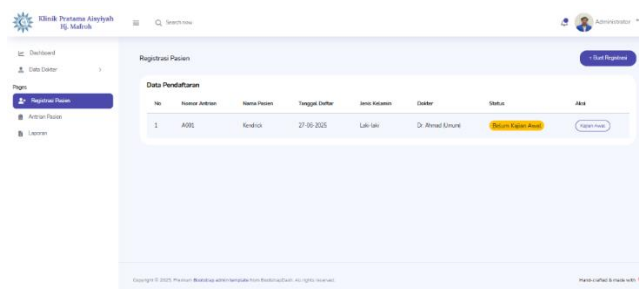
1. Menampilkan daftar data jadwal dokter dan *button edit* dan *hapus*.
2. Menampilkan *button* untuk tambah data jadwal dokter yang berisi form untuk tambah data jadwal dokter seperti, nama dokter, spesialis, hari praktek, dan jam praktek.



Gambar 1. 97 Jadwal Dokter

e. Halaman Registrasi Pasien

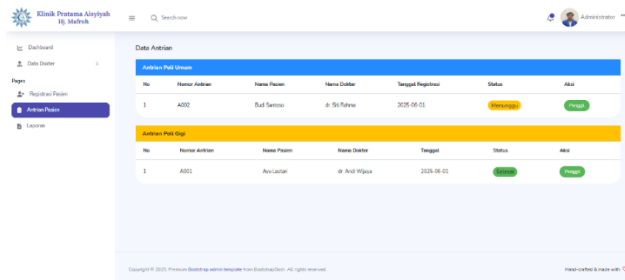
1. Menampilkan daftar data pasien dan *button* kajian awal
2. Menampilkan *button* untuk buat Registrasi yang berisi form untuk tambah data pasien seperti, isi form data diri, penanggung jawab, dan jadwal dokter



Gambar 1. 129 Halaman Registrasi Pasien

f. Halaman Antrian Pasien

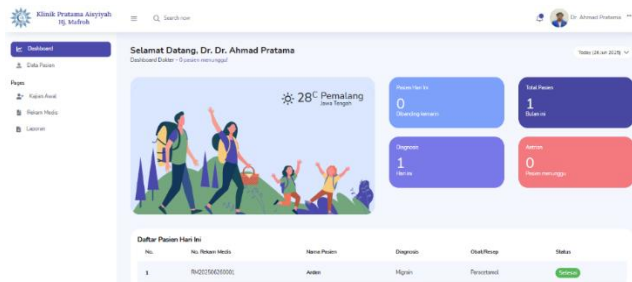
1. Menampilkan data antrian pasien dengan 2 poli yaitu, poli umum dan poli gigi
2. Menampilkan *button* panggil untuk memanggil nomor antrian selanjutnya.



Gambar 1. 161 Halaman Antrian Pasien

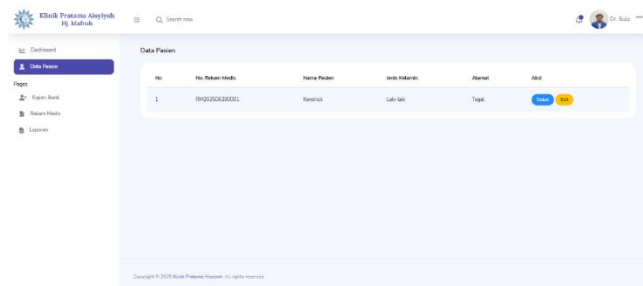
g. Halaman *Dashboard* Dokter

Bagian ini hanya dapat diakses oleh dokter, menampilkan ringkasan statistik penting untuk mendukung aktivitas harian dokter, seperti jumlah pasien hari ini, total pasien bulanan, jumlah diagnosis, dan antrian yang menunggu.



Gambar 1. 193 Halaman Dashboard Dokter

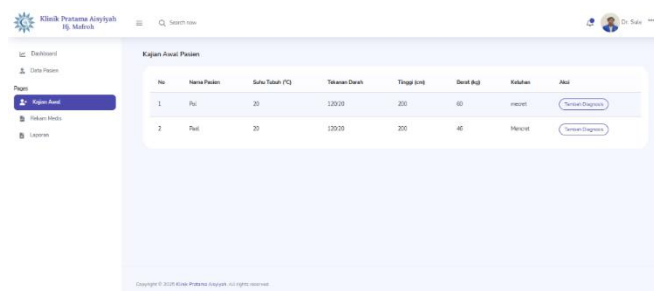
1. Menampilkan daftar data pasien dalam tabel yang berisi, Nomor urut, Nomor Rekam Medis, Nama Pasien, Jenis Kelamin, Alamat
2. Menyediakan *button Detail* untuk melihat informasi pasien secara lengkap
3. Menyediakan *button Edit* untuk memperbarui data pasien



Gambar 1. 225 Halaman Data Pasien

h. Halaman Kajian Awal

1. Menampilkan daftar data kajian awal pasien dalam bentuk tabel, yang meliputi, Nomor urut, Nama Pasien, Suhu Tubuh ($^{\circ}\text{C}$), Tekanan Darah, Tinggi Badan (cm), Berat Badan (kg), Keluhan Pasien
2. Menyediakan *button* Tambah Diagnosis pada setiap baris data untuk melanjutkan proses diagnosa pasien.

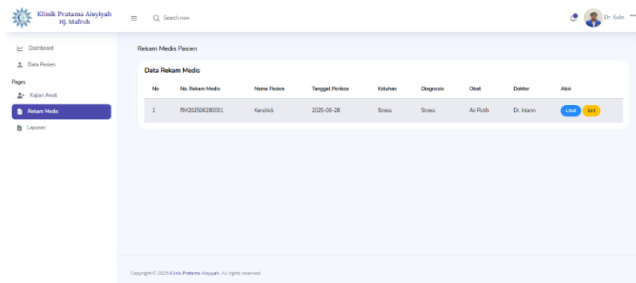


Gambar 1. 257 Halaman Kajian Awal

i. Halaman Rekam Medis

1. Menampilkan data rekam medis pasien dalam bentuk tabel, yang mencakup, Nomor urut, Nomor Rekam Medis, Nama Pasien, Tanggal Periksa, Keluhan Pasien, Diagnosis, Obat yang diberikan, Nama Dokter pemeriksa

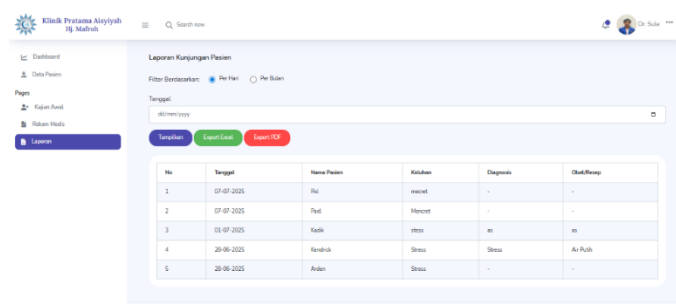
2. Menyediakan *button* Lihat untuk melihat *detail* rekam medis
3. Menyediakan *button* Edit untuk mengubah data rekam medis



Gambar 1. 289 Halaman Rekam Medis

j. Halaman Laporan

1. Menampilkan laporan kunjungan pasien dalam bentuk tabel yang mencakup, Nomor urut, Tanggal kunjungan, Nama pasien, Keluhan, Diagnosis, Obat / Resep
2. Menyediakan fitur filter berdasarkan, Per Hari, Per Bulan
3. Menyediakan input tanggal untuk pencarian laporan
4. Menyediakan tombol, Tampilkan untuk melihat laporan sesuai filter, *Export Excel* untuk mengunduh data dalam *format Excel*, *Export PDF* untuk mengunduh data dalam *format PDF*.



Gambar 1. 321 Halaman Laporan

2. Source Code

a. Source Code UI (User Interface)

1. Register.blade.php

1. Merupakan *source code* untuk menampilkan halaman *Registrasi* Admin pada aplikasi.

2. *Source code* ini berada di dalam *folder resources > views* > **Auth** dan digunakan untuk menambahkan akun Admin baru.
3. Menampilkan form yang terdiri dari :
 - a. Nama Lengkap
 - b. *Email*
 - c. Nomor Telepon (opsional)
 - d. *Password*
4. Terdapat validasi *error* untuk setiap *field* yang ditampilkan secara dinamis.
5. Data akan dikirim menggunakan *method POST* ke *route Register*.
6. Dilengkapi dengan input tersembunyi *role_id* untuk menetapkan peran Admin.
7. Setelah form dikirim, Admin akan terdaftar dalam sistem.

Gambar 1. 353 Source Code Register

1. Merupakan *source code* untuk menampilkan halaman *Login* Admin pada aplikasi klinik.
2. *Source code* ini berada di dalam folder **resources** > **views** > **Auth** dan digunakan untuk autentikasi Admin.

3. Menampilkan form *Login* dengan *field* :
 - a. *Email*
 - b. *Password*
4. Menampilkan pesan *error* atau sukses secara dinamis jika terdapat validasi atau notifikasi dari *session*.
5. Dilengkapi fitur :
 - a. *Toggle* tampilan *password* (show/hide)
 - b. *Link* untuk Lupa *Password*
 - c. *Link* untuk Buat Akun jika belum memiliki akun.
6. Menggunakan *method POST* untuk mengirim data ke *route Login*.

```

1 <@csrf@>
2 <div class="form">
3   <div class="form-group">
4     <input type="text" value="" />
5   </div>
6   <div class="form-group">
7     <input type="password" value="" />
8   </div>
9   <div class="form-group">
10    <input type="checkbox" />
11  </div>
12  <div class="form-group">
13    <input type="button" value="Login" />
14  </div>
15  <div class="form-group">
16    <input type="button" value="Lupa Password" />
17  </div>
18  <div class="form-group">
19    <input type="button" value="Buat Akun" />
20  </div>
21 </div>
22 </div>
23 </div>
24 </div>
25 </div>
26 </div>
27 </div>
28 </div>
29 </div>
30 </div>
31 </div>
32 </div>
33 </div>
34 </div>
35 </div>
36 </div>
37 </div>
38 </div>
39 </div>
40 </div>
41 </div>
42 </div>
43 </div>
44 </div>
45 </div>
46 </div>
47 </div>
48 </div>
49 </div>
50 </div>
51 </div>
52 </div>
53 </div>
54 </div>
55 </div>
56 </div>
57 </div>
58 </div>
59 </div>
60 </div>
61 </div>
62 </div>
63 </div>
64 </div>
65 </div>
66 </div>
67 </div>
68 </div>
69 </div>
70 </div>
71 </div>
72 </div>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
78 </div>
79 </div>
80 </div>
81 </div>
82 </div>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </div>
92 </div>
93 </div>
94 </div>
95 </div>
96 </div>
97 </div>
98 </div>
99 </div>
100 </div>

```

Gambar 1. 385 Source Code Login

3. *Dashboard.blade.php*

1. Merupakan *source code* untuk menampilkan halaman *Dashboard Admin* pada aplikasi klinik.
2. Menampilkan sambutan kepada pengguna yang *Login* (menggunakan *Auth::User()->name*).
3. *Source code* ini berada di dalam *folder views* > *admin* dan digunakan untuk autentikasi Admin.

[illegible]

Gambar 1. 417 Source Code Dashboard Admin

4. *datadokter.blade.php*

1. Merupakan *source code* untuk menampilkan halaman data dokter dalam sistem admin klinik.
2. Halaman ini menggunakan template *layouts.admin*.
3. *Source code* ini berada di dalam **folder views > admin** dan digunakan untuk autentikasi Admin.

```

1  #!/usr/bin/perl -w
2
3  # Perl 5.10.1.55 on Linux
4
5  # Perl 5.10.1.55
6
7  # Perl 5.10.1.55
8
9  # Perl 5.10.1.55
10
11 # Perl 5.10.1.55
12
13 # Perl 5.10.1.55
14
15 # Perl 5.10.1.55
16
17 # Perl 5.10.1.55
18
19 # Perl 5.10.1.55
20
21 # Perl 5.10.1.55
22
23 # Perl 5.10.1.55
24
25 # Perl 5.10.1.55
26
27 # Perl 5.10.1.55
28
29 # Perl 5.10.1.55
30
31 # Perl 5.10.1.55
32
33 # Perl 5.10.1.55
34
35 # Perl 5.10.1.55
36
37 # Perl 5.10.1.55
38
39 # Perl 5.10.1.55
40
41 # Perl 5.10.1.55
42
43 # Perl 5.10.1.55
44
45 # Perl 5.10.1.55
46
47 # Perl 5.10.1.55
48
49 # Perl 5.10.1.55
50
51 # Perl 5.10.1.55
52
53 # Perl 5.10.1.55
54
55 # Perl 5.10.1.55
56
57 # Perl 5.10.1.55
58
59 # Perl 5.10.1.55
60
61 # Perl 5.10.1.55
62
63 # Perl 5.10.1.55
64
65 # Perl 5.10.1.55
66
67 # Perl 5.10.1.55
68
69 # Perl 5.10.1.55
70
71 # Perl 5.10.1.55
72
73 # Perl 5.10.1.55
74
75 # Perl 5.10.1.55
76
77 # Perl 5.10.1.55
78
79 # Perl 5.10.1.55
80
81 # Perl 5.10.1.55
82
83 # Perl 5.10.1.55
84
85 # Perl 5.10.1.55
86
87 # Perl 5.10.1.55
88
89 # Perl 5.10.1.55
90
91 # Perl 5.10.1.55
92
93 # Perl 5.10.1.55
94
95 # Perl 5.10.1.55
96
97 # Perl 5.10.1.55
98
99 # Perl 5.10.1.55
100
101 # Perl 5.10.1.55
102
103 # Perl 5.10.1.55
104
105 # Perl 5.10.1.55
106
107 # Perl 5.10.1.55
108
109 # Perl 5.10.1.55
110
111 # Perl 5.10.1.55
112
113 # Perl 5.10.1.55
114
115 # Perl 5.10.1.55
116
117 # Perl 5.10.1.55
118
119 # Perl 5.10.1.55
120
121 # Perl 5.10.1.55
122
123 # Perl 5.10.1.55
124
125 # Perl 5.10.1.55
126
127 # Perl 5.10.1.55
128
129 # Perl 5.10.1.55
130
131 # Perl 5.10.1.55
132
133 # Perl 5.10.1.55
134
135 # Perl 5.10.1.55
136
137 # Perl 5.10.1.55
138
139 # Perl 5.10.1.55
140
141 # Perl 5.10.1.55
142
143 # Perl 5.10.1.55
144
145 # Perl 5.10.1.55
146
147 # Perl 5.10.1.55
148
149 # Perl 5.10.1.55
150
151 # Perl 5.10.1.55
152
153 # Perl 5.10.1.55
154
155 # Perl 5.10.1.55
156
157 # Perl 5.10.1.55
158
159 # Perl 5.10.1.55
160
161 # Perl 5.10.1.55
162
163 # Perl 5.10.1.55
164
165 # Perl 5.10.1.55
166
167 # Perl 5.10.1.55
168
169 # Perl 5.10.1.55
170
171 # Perl 5.10.1.55
172
173 # Perl 5.10.1.55
174
175 # Perl 5.10.1.55
176
177 # Perl 5.10.1.55
178
179 # Perl 5.10.1.55
180
181 # Perl 5.10.1.55
182
183 # Perl 5.10.1.55
184
185 # Perl 5.10.1.55
186
187 # Perl 5.10.1.55
188
189 # Perl 5.10.1.55
190
191 # Perl 5.10.1.55
192
193 # Perl 5.10.1.55
194
195 # Perl 5.10.1.55
196
197 # Perl 5.10.1.55
198
199 # Perl 5.10.1.55
200
201 # Perl 5.10.1.55
202
203 # Perl 5.10.1.55
204
205 # Perl 5.10.1.55
206
207 # Perl 5.10.1.55
208
209 # Perl 5.10.1.55
210
211 # Perl 5.10.1.55
212
213 # Perl 5.10.1.55
214
215 # Perl 5.10.1.55
216
217 # Perl 5.10.1.55
218
219 # Perl 5.10.1.55
220
221 # Perl 5.10.1.55
222
223 # Perl 5.10.1.55
224
225 # Perl 5.10.1.55
226
227 # Perl 5.10.1.55
228
229 # Perl 5.10.1.55
230
231 # Perl 5.10.1.55
232
233 # Perl 5.10.1.55
234
235 # Perl 5.10.1.55
236
237 # Perl 5.10.1.55
238
239 # Perl 5.10.1.55
240
241 # Perl 5.10.1.55
242
243 # Perl 5.10.1.55
244
245 # Perl 5.10.1.55
246
247 # Perl 5.10.1.55
248
249 # Perl 5.10.1.55
250
251 # Perl 5.10.1.55
252
253 # Perl 5.10.1.55
254
255 # Perl 5.10.1.55
256
257 # Perl 5.10.1.55
258
259 # Perl 5.10.1.55
260
261 # Perl 5.10.1.55
262
263 # Perl 5.10.1.55
264
265 # Perl 5.10.1.55
266
267 # Perl 5.10.1.55
268
269 # Perl 5.10.1.55
270
271 # Perl 5.10.1.55
272
273 # Perl 5.10.1.55
274
275 # Perl 5.10.1.55
276
277 # Perl 5.10.1.55
278
279 # Perl 5.10.1.55
280
281 # Perl 5.10.1.55
282
283 # Perl 5.10.1.55
284
285 # Perl 5.10.1.55
286
287 # Perl 5.10.1.55
288
289 # Perl 5.10.1.55
290
291 # Perl 5.10.1.55
292
293 # Perl 5.10.1.55
294
295 # Perl 5.10.1.55
296
297 # Perl 5.10.1.55
298
299 # Perl 5.10.1.55
300
301 # Perl 5.10.1.55
302
303 # Perl 5.10.1.55
304
305 # Perl 5.10.1.55
306
307 # Perl 5.10.1.55
308
309 # Perl 5.10.1.55
310
311 # Perl 5.10.1.55
312
313 # Perl 5.10.1.55
314
315 # Perl 5.10.1.55
316
317 # Perl 5.10.1.55
318
319 # Perl 5.10.1.55
320
321 # Perl 5.10.1.55
322
323 # Perl 5.10.1.55
324
325 # Perl 5.10.1.55
326
327 # Perl 5.10.1.55
328
329 # Perl 5.10.1.55
330
331 # Perl 5.10.1.55
332
333 # Perl 5.10.1.55
334
335 # Perl 5.10.1.55
336
337 # Perl 5.10.1.55
338
339 # Perl 5.10.1.55
340
341 # Perl 5.10.1.55
342
343 # Perl 5.10.1.55
344
345 # Perl 5.10.1.55
346
347 # Perl 5.10.1.55
348
349 # Perl 5.10.1.55
350
351 # Perl 5.10.1.55
352
353 # Perl 5.10.1.55
354
355 # Perl 5.10.1.55
356
357 # Perl 5.10.1.55
358
359 # Perl 5.10.1.55
360
361 # Perl 5.10.1.55
362
363 # Perl 5.10.1.55
364
365 # Perl 5.10.1.55
366
367 # Perl 5.10.1.55
368
369 # Perl 5.10.1.55
370
371 # Perl 5.10.1.55
372
373 # Perl 5.10.1.55
374
375 # Perl 5.10.1.55
376
377 # Perl 5.10.1.55
378
379 # Perl 5.10.1.55
380
381 # Perl 5.10.1.55
382
383 # Perl 5.10.1.55
384
385 # Perl 5.10.1.55
386
387 # Perl 5.10.1.55
388
389 # Perl 5.10.1.55
390
391 # Perl 5.10.1.55
392
393 # Perl 5.10.1.55
394
395 # Perl 5.10.1.55
396
397 # Perl 5.10.1.55
398
399 # Perl 5.10.1.55
400
401 # Perl 5.10.1.55
402
403 # Perl 5.10.1.55
404
405 # Perl 5.10.1.55
406
407 # Perl 5.10.1.55
408
409 # Perl 5.10.1.55
410
411 # Perl 5.10.1.55
412
413 # Perl 5.10.1.55
414
415 # Perl 5.10.1.55
416
417 # Perl 5.10.1.55
418
419 # Perl 5.10.1.55
420
```

Gambar 1. 449 Source Code

5. *jadwaldokter.blade.php*

1. Digunakan oleh admin untuk melihat, menambahkan, dan *mengedit* jadwal dokter.
2. Menggunakan template *layouts.admin*.
3. *Source code* ini berada di dalam folder pada *folder resources > views > admin > jadwaldokter.blade.php*. dan digunakan untuk autentikasi Admin.

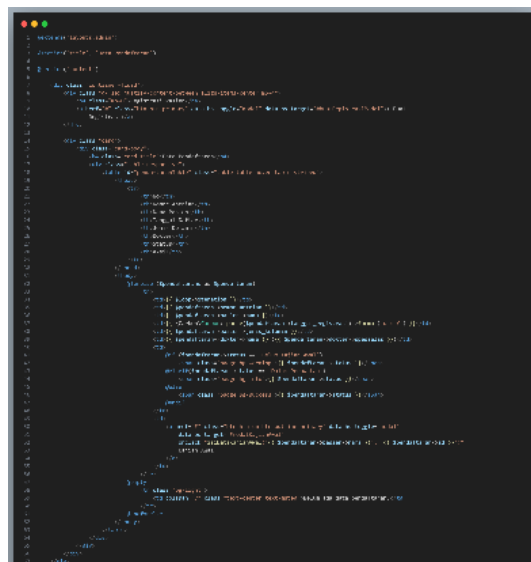


Gambar 1. 481 Source Code Jadwal Dokter

6. *datapendaftaran.blade.php*

1. *Source code* ini merupakan halaman tabel data pendaftaran pasien yang berada pada *folder resources > views > admin > datapendaftaran.blade.php*.
2. *Source code* ini menampilkan tabel data pendaftaran yang berisi informasi :
 - a. Nomor antrian
 - b. Nama pasien
 - c. Nama dokter
 - d. Tanggal *Registrasi*
 - e. Status
 - f. Aksi (lihat *detail* atau hapus data)
3. *Source code* ini menggunakan *Blade* templating dari *Laravel* dan memanfaatkan :

- a. *@extends* dan *@section* untuk integrasi ke dalam *layout* utama admin.
 - b. *@foreach* untuk menampilkan data pendaftaran secara dinamis dalam tabel.
 - c. *@csrf* dan *@method('DELETE')* untuk mengamankan form hapus data.
4. *Source code* ini juga menyediakan :
- a. *Button* untuk menambahkan data baru.
 - b. Aksi *edit* dan *hapus* menggunakan ikon *fa-edit* dan *fa-trash*.
 - c. Konfirmasi *JavaScript* saat akan menghapus data (*onclick="return confirm(...)"*).



7. *dataantrian.blade.php*

2. *Source code* ini menampilkan dua tabel antrian secara terpisah, yaitu :
 - a. Antrian Poli Umum
 - b. Antrian Poli Gigi

3. Setiap tabel menampilkan data antrian pasien secara dinamis menggunakan *@foreach*, meliputi :
 - a. Nomor urut
 - b. Nomor antrian
 - c. Nama pasien
 - d. Nama dokter
 - e. Tanggal *Registrasi*
 - f. Status (Belum Kajian Awal, Selesai, Dalam Perawatan)
 - g. Aksi (*button* “Panggil” atau “Selesai”)
4. *Source code* ini menggunakan *badge* warna untuk memperjelas status:
 - a. *bg-warning* untuk “Belum Kajian Awal”
 - b. *bg-success* untuk “Selesai”
 - c. *bg-info* untuk “Dalam Perawatan”
5. *Source code* ini juga menambahkan :
 - a. *Button* "Panggil" dengan *method POST* dan *CSRF* untuk status yang belum dikaji.
 - b. *Button disabled* "Selesai" jika antrian sudah ditindaklanjuti.
 - c. Pesan *fallback* “Tidak ada antrian hari ini” jika data kosong, dengan *@empty*.

```

1 @section('content')
2
3 @section('title', 'Data Antrian')
4
5 @section('content')
6     <div class="container-fluid">
7         <div class="d-flex justify-content-between align-items-center mb-4">
8             <div class="h2">Data Antrian Pasien</div>
9         </div>
10
11         <table border="1">
12             <thead>
13                 <tr>
14                     <th>No</th>
15                     <th>No. Antrian</th>
16                     <th>Nama Pasien</th>
17                     <th>Nama Dokter</th>
18                     <th>Tanggal Registrasi</th>
19                     <th>Status</th>
20                     <th>Aksi</th>
21                 </tr>
22             </thead>
23             <tbody>
24                 @foreach ($antrian as $index => $antrian)
25                     <tr>
26                         <td>{{ $index + 1 }}</td>
27                         <td>{{ $antrian->nomor_antrian }}</td>
28                         <td>{{ $antrian->nama_pasien }}</td>
29                         <td>{{ $antrian->nama_dokter }}</td>
30                         <td>{{ $antrian->tanggal_registrasi }}</td>
31                         <td>{{ $antrian->status }}</td>
32                         <td>
33                             @if ($antrian->status == 'Belum Kajian Awal')
34                                 <div class="badge bg-warning text-dark">Memanggil</div>
35                             @elseif ($antrian->status == 'Selesai')
36                                 <div class="badge bg-success text-dark">Selesai</div>
37                             @elseif ($antrian->status == 'Dalam Perawatan')
38                                 <div class="badge bg-info text-dark">Dalam Perawatan</div>
39                             @endif
40                         </td>
41                     </tr>
42                 @endforeach
43             </tbody>
44         </table>
45
46         @if ($antrian->isEmpty())
47             <div class="text-center">
48                 <div class="text-warning">Tidak ada antrian hari ini</div>
49             </div>
50         @endif
51     </div>
52 @endsection

```

Gambar 1. 545 Source Code Data Antrian

8. *Dashboard.blade.php*

1. *Source code* ini merupakan halaman *Dashboard* Dokter yang berada pada folder **resources** > **views** > **dokter** > **Dashboard.blade.php**.
2. *Source code* ini menampilkan :
 - a. Sambutan kepada dokter yang sedang *Login* (`{{ Auth::User()->name }}`).
 - b. Informasi jumlah pasien yang sedang menunggu.
 - c. Tanggal hari ini dengan tombol *dropdown* untuk memilih periode bulan (Januari – November).
3. *Source code* ini juga menyajikan ringkasan statistik dalam bentuk card, seperti :
 - a. Jumlah pasien hari ini
 - b. Total pasien bulan ini
 - c. Jumlah konsultasi hari ini
 - d. Antrian dan pasien yang sedang menunggu
4. *Source code* ini menampilkan informasi Lokasi :
 - a. Kota: Tegal
 - b. Provinsi: Jawa Tengah
5. *Source code* ini menampilkan tabel daftar pasien hari ini yang memuat data :
 - a. Nomor RM
 - b. Nama Pasien
 - c. Diagnosis
 - d. Obat/Resep
6. Status (Selesai, Sedang dilayani, Menunggu) dengan *badge* warna yang sesuai :
 - a. *badge-success* untuk Selesai
 - b. *badge-warning* untuk Sedang dilayani
 - c. *badge-info* untuk Menunggu

```

1 <divclass="layouts-header">
2
3 <divclass="content">
4
5 <div class="col-md-12 grid-margin">
6
7 <div class="row">
8
9 <div class="col-12 col-sm-6 col-md-6">
10
11 <div class="font-weight-bold">Selamat Pagi, Dr. {{ Auth::user()->name }}</div>
12 <div class="font-weight-normal">Good</div>
13 <div class="font-weight-normal">Dashboard Dokter</div>
14
15 </div>
16 <div class="col-12 col-sm-6 col-md-6">
17
18 <div class="font-weight-normal">Kardio</div>
19 <div class="font-weight-normal">Kardio</div>
20 <div class="font-weight-normal">Kardio</div>
21 <div class="font-weight-normal">Kardio</div>
22 <div class="font-weight-normal">Kardio</div>
23 <div class="font-weight-normal">Kardio</div>
24 <div class="font-weight-normal">Kardio</div>
25 <div class="font-weight-normal">Kardio</div>
26 <div class="font-weight-normal">Kardio</div>
27 <div class="font-weight-normal">Kardio</div>
28 <div class="font-weight-normal">Kardio</div>
29 <div class="font-weight-normal">Kardio</div>
30 <div class="font-weight-normal">Kardio</div>
31 <div class="font-weight-normal">Kardio</div>
32 <div class="font-weight-normal">Kardio</div>
33 <div class="font-weight-normal">Kardio</div>
34 <div class="font-weight-normal">Kardio</div>
35 <div class="font-weight-normal">Kardio</div>
36 <div class="font-weight-normal">Kardio</div>
37 <div class="font-weight-normal">Kardio</div>
38 <div class="font-weight-normal">Kardio</div>
39 <div class="font-weight-normal">Kardio</div>
40 <div class="font-weight-normal">Kardio</div>
41 <div class="font-weight-normal">Kardio</div>
42 <div class="font-weight-normal">Kardio</div>
43 <div class="font-weight-normal">Kardio</div>
44 <div class="font-weight-normal">Kardio</div>
45 <div class="font-weight-normal">Kardio</div>
46 <div class="font-weight-normal">Kardio</div>
47 <div class="font-weight-normal">Kardio</div>
48 <div class="font-weight-normal">Kardio</div>
49 <div class="font-weight-normal">Kardio</div>
50 <div class="font-weight-normal">Kardio</div>
51 <div class="font-weight-normal">Kardio</div>
52 <div class="font-weight-normal">Kardio</div>
53 <div class="font-weight-normal">Kardio</div>
54 <div class="font-weight-normal">Kardio</div>
55 <div class="font-weight-normal">Kardio</div>
56 <div class="font-weight-normal">Kardio</div>
57 <div class="font-weight-normal">Kardio</div>
58 <div class="font-weight-normal">Kardio</div>
59 <div class="font-weight-normal">Kardio</div>
60 <div class="font-weight-normal">Kardio</div>
61 <div class="font-weight-normal">Kardio</div>
62 <div class="font-weight-normal">Kardio</div>
63 <div class="font-weight-normal">Kardio</div>
64 <div class="font-weight-normal">Kardio</div>
65 <div class="font-weight-normal">Kardio</div>
66 <div class="font-weight-normal">Kardio</div>
67 <div class="font-weight-normal">Kardio</div>
68 <div class="font-weight-normal">Kardio</div>
69 <div class="font-weight-normal">Kardio</div>
70 <div class="font-weight-normal">Kardio</div>
71 <div class="font-weight-normal">Kardio</div>
72 <div class="font-weight-normal">Kardio</div>
73 <div class="font-weight-normal">Kardio</div>
74 <div class="font-weight-normal">Kardio</div>
75 <div class="font-weight-normal">Kardio</div>
76 <div class="font-weight-normal">Kardio</div>
77 <div class="font-weight-normal">Kardio</div>
78 <div class="font-weight-normal">Kardio</div>
79 <div class="font-weight-normal">Kardio</div>
80 <div class="font-weight-normal">Kardio</div>
81 <div class="font-weight-normal">Kardio</div>
82 <div class="font-weight-normal">Kardio</div>
83 <div class="font-weight-normal">Kardio</div>
84 <div class="font-weight-normal">Kardio</div>
85 <div class="font-weight-normal">Kardio</div>
86 <div class="font-weight-normal">Kardio</div>
87 <div class="font-weight-normal">Kardio</div>
88 <div class="font-weight-normal">Kardio</div>
89 <div class="font-weight-normal">Kardio</div>
90 <div class="font-weight-normal">Kardio</div>
91 <div class="font-weight-normal">Kardio</div>
92 <div class="font-weight-normal">Kardio</div>
93 <div class="font-weight-normal">Kardio</div>
94 <div class="font-weight-normal">Kardio</div>
95 <div class="font-weight-normal">Kardio</div>
96 <div class="font-weight-normal">Kardio</div>
97 <div class="font-weight-normal">Kardio</div>
98 <div class="font-weight-normal">Kardio</div>
99 <div class="font-weight-normal">Kardio</div>
100 <div class="font-weight-normal">Kardio</div>

```

Gambar 1. 577 Source Code Dashboad Dokter

9. *datap pasien.blade.php*

1. *Source code* ini merupakan halaman data pasien dalam sistem dokter, dan berada pada *folder resources > views > dokter > datap pasien.blade.php*.
2. *Source code* ini menampilkan tabel data pasien dalam bentuk *table-responsive* dan menggunakan *@foreach* untuk menampilkan isi data.
3. Informasi yang ditampilkan pada tabel meliputi:
 - a. No. RM
 - b. Nama Pasien
 - c. Jenis Kelamin
 - d. Tempat, Tanggal Lahir
 - e. Alamat
 - f. No. Telepon
 - g. Aksi: *Edit*, *Hapus*, dan *Lihat Rekam Medis*
4. *Source code* ini menggunakan fitur :
 - a. *@csrf* dan *@method('DELETE')* untuk mengamankan aksi hapus pasien.

- b. Form untuk melakukan *update* data pasien dan mengarah ke *updateDataPasien controller*.
 - c. Modal atau konfirmasi aksi penghapusan dilakukan langsung dalam baris tabel menggunakan form.
5. *Source code* ini juga menggunakan:
 - a. Struktur *Blade @extends*, *@section*, dan *@foreach*.
 - b. Format tanggal menggunakan *Carbon* untuk Tanggal Lahir pasien.
6. Akses tombol aksi :
 - a. *Edit* → menuju form *update*.
 - b. *Hapus* → kirim *request DELETE*.
 - c. Lihat Rekam Medis → link ke *rekammedis.index*.

- Struktur *Blade @extends*, *@section*, dan *@foreach*.
 - Format tanggal menggunakan *Carbon* untuk Tanggal Lahir pasien.
- Akses tombol aksi :
- Edit* → menuju form *update*.
 - Hapus* → kirim *request DELETE*.
 - Lihat Rekam Medis → link ke *rekammedis.index*.

- b. *Hapus* $\rightarrow$ kirim request *DELETE*.

- c. Lihat Rekam Medis → link ke *rekammedis.index*.

```

1  #!/usr/bin/perl -w
2
3  use strict;
4  use warnings;
5  use Data::Dumper;
6
7  # Constants
8  my $VERSION = "1.0.0";
9  my $AUTHOR = "John Doe";
10
11 # Main function
12 sub main {
13     my $input = "data.txt";
14     my $output = "result.txt";
15     my $format = "json";
16
17     # Process input
18     my $data = load_data($input);
19     my $result = process_data($data);
20
21     # Save result
22     save_data($result, $output, $format);
23 }
24
25 # Load data from file
26 sub load_data {
27     my $file = shift;
28     my $data = {};
29
30     if (-f $file) {
31         open(my $fh, "<$file") or die "Cannot open file: $!";
32         while (my $line = <$fh) {
33             my $json = $line;
34             my $obj = decode_json($json);
35             $data->{$obj->id} = $obj;
36         }
37     } else {
38         die "File not found: $file";
39     }
40
41     return $data;
42 }
43
44 # Process data
45 sub process_data {
46     my $data = shift;
47     my $result = {};
48
49     foreach my $id (keys %$data) {
50         my $obj = $data->{$id};
51         my $new_obj = {
52             id => $obj->id,
53             name => $obj->name,
54             value => $obj->value * 2,
55             status => $obj->status,
56         };
57         $result->{$id} = $new_obj;
58     }
59
60     return $result;
61 }
62
63 # Save data to file
64 sub save_data {
65     my $data = shift;
66     my $file = shift;
67     my $format = shift;
68
69     if ($format eq "json") {
70         my $json = encode_json($data);
71         write_file($file, $json);
72     } else {
73         die "Unsupported format: $format";
74     }
75 }
76
77 # Write to file
78 sub write_file {
79     my $file = shift;
80     my $content = shift;
81
82     open(my $fh, ">$file") or die "Cannot open file: $!";
83     print $fh $content;
84     close($fh);
85 }
86
87 # Main execution
88 main();
89
90 # End of script
91 __END__
92
93 # Test data
94 my %test_data = (
95     1 => { id => 1, name => "Alice", value => 10, status => "Active" },
96     2 => { id => 2, name => "Bob", value => 20, status => "Inactive" },
97     3 => { id => 3, name => "Charlie", value => 30, status => "Active" },
98 );
99
100 # Expected result
101 my %expected_result = (
102     1 => { id => 1, name => "Alice", value => 20, status => "Active" },
103     2 => { id => 2, name => "Bob", value => 40, status => "Inactive" },
104     3 => { id => 3, name => "Charlie", value => 60, status => "Active" },
105 );
106
107 # Run tests
108 my $data = load_data("data.txt");
109 my $result = process_data($data);
110 my $json = encode_json($result);
111 my $file = "result.txt";
112 write_file($file, $json);
113
114 # Verify result
115 my $content = read_file($file);
116 my $obj = decode_json($content);
117
118 if (Dumper($obj) eq Dumper(%expected_result)) {
119     print "Tests passed!\n";
120 } else {
121     print "Tests failed!\n";
122 }
123
124 # Cleanup
125 unlink($file);
126
127 # End of script
128

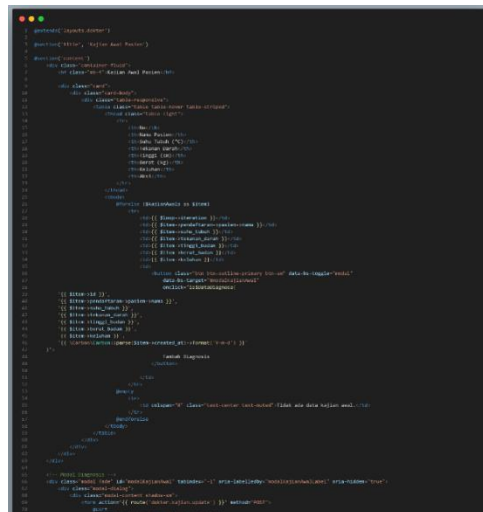
```

Gambar 1. 609 Source Code Data Pasien

10. *kajianawal.blade.php*

1. *Source code* ini merupakan halaman Kajian Awal Pasien yang terletak pada *folder resources > views > dokter > kajianawal.blade.php*.

2. *Source code* ini menampilkan tabel kajian awal pasien secara dinamis menggunakan *@foreach*, yang mencakup data:
 - a. Nama Pasien
 - b. Suhu Tubuh
 - c. Tekanan Darah
 - d. Tinggi Badan
 - e. Berat Badan
 - f. Keluhan
 - g. Aksi tombol untuk membuka form diagnosis
3. *Source code* ini dilengkapi button “Tambah Diagnosis” pada setiap baris untuk membuka modal form dengan *detail* kajian awal pasien.
4. Modal tersebut menampilkan informasi kajian awal dalam form *input* dengan atribut *readonly*, dan memiliki :
 - a. *Input* untuk Diagnosis (wajib diisi)
 - b. Input untuk Obat / Resep
 - c. Tombol “Simpan” dan “Batal”
5. *Source code* ini juga menggunakan :
 - a. *Carbon* untuk format tanggal pemeriksaan.
 - b. *@empty* untuk menampilkan *fallback* jika tidak ada data kajian awal : “Tidak ada data kajian awal.”
 - c. *@csrf* dan *method="POST"* untuk *submit* form dengan aman ke *route dokter.kajian.update*.



Gambar 1. 641 Source Code Kajian Awal

11. *rekammedis.blade.php*

1. *Source code* ini merupakan halaman Rekam Medis Pasien yang berada pada *folder resources > views > dokter > rekammedis.blade.php*.
2. *Source code* ini menampilkan tabel data rekam medis pasien yang berisi:
 - a. Nomor RM
 - b. Nama Pasien
 - c. Jenis Pemeriksaan
 - d. Tanggal Periksa
 - e. Diagnosa
 - f. Obat
 - g. Aksi (Lihat & Edit)
3. Setiap baris memiliki dua *button* aksi:
 - a. Lihat → untuk menampilkan modal *detail* kajian awal pasien (*read-only*).
 - b. *Edit* → untuk membuka modal *edit* rekam medis.
4. *Source code* ini menggunakan dua modal :
 1. Modal *Detail* Kajian Awal :

1. Menampilkan informasi pasien dan hasil pemeriksaan secara lengkap :
 2. Suhu, Tekanan Darah, Tinggi Badan, Berat Badan, Keluhan, Diagnosa, Obat.
 3. Semua input dalam keadaan *readonly*.
2. Modal *Edit Rekam Medis* :
1. Menampilkan data yang sama namun dapat diubah: Diagnosa dan Obat dapat diedit.
 2. Tombol "Simpan Perubahan" mengirim data via *POST* ke *route* `dokter.kajian.update`.
 3. Menggunakan `@csrf` dan `@method('POST')` untuk keamanan.
5. *Source code* juga menggunakan:
1. Struktur `@extends`, `@section`, `@foreach`, dan `@empty` untuk tampilan dinamis.
 2. Atribut `data-*` untuk mengisi modal menggunakan *JavaScript*.
 3. Validasi pada *textarea* diagnosis (*required*) agar tidak kosong saat penyimpanan.

```

1 <script({ layout: 'doctor' })
2 <script({ title: 'Rekam Medis Pasien' })
3 <script({ content: '' })
4 <div class="container-fluid"
5 <div class="card-body"
6 <div class="card-body"
7 <div class="card-body"
8 <div class="card-body"
9 <div class="card-body"
10 <div class="card-body"
11 <div class="card-body"
12 <div class="card-body"
13 <div class="card-body"
14 <div class="card-body"
15 <div class="card-body"
16 <div class="card-body"
17 <div class="card-body"
18 <div class="card-body"
19 <div class="card-body"
20 <div class="card-body"
21 <div class="card-body"
22 <div class="card-body"
23 <div class="card-body"
24 <div class="card-body"
25 <div class="card-body"
26 <div class="card-body"
27 <div class="card-body"
28 <div class="card-body"
29 <div class="card-body"
30 <div class="card-body"
31 <div class="card-body"
32 <div class="card-body"
33 <div class="card-body"
34 <div class="card-body"
35 <div class="card-body"
36 <div class="card-body"
37 <div class="card-body"
38 <div class="card-body"
39 <div class="card-body"
40 <div class="card-body"
41 <div class="card-body"
42 <div class="card-body"
43 <div class="card-body"
44 <div class="card-body"
45 <div class="card-body"
46 <div class="card-body"
47 <div class="card-body"
48 <div class="card-body"
49 <div class="card-body"
50 <div class="card-body"
51 <div class="card-body"
52 <div class="card-body"
53 <div class="card-body"
54 <div class="card-body"
55 <div class="card-body"
56 <div class="card-body"
57 <div class="card-body"
58 <div class="card-body"
59 <div class="card-body"
60 <div class="card-body"
61 <div class="card-body"
62 <div class="card-body"
63 <div class="card-body"
64 <div class="card-body"
65 <div class="card-body"
66 <div class="card-body"
67 <div class="card-body"
68 <div class="card-body"
69 <div class="card-body"
70 <div class="card-body"
71 <div class="card-body"
72 <div class="card-body"
73 <div class="card-body"
74 <div class="card-body"
75 <div class="card-body"
76 <div class="card-body"
77 <div class="card-body"
78 <div class="card-body"
79 <div class="card-body"
80 <div class="card-body"
81 <div class="card-body"
82 <div class="card-body"
83 <div class="card-body"
84 <div class="card-body"
85 <div class="card-body"
86 <div class="card-body"
87 <div class="card-body"
88 <div class="card-body"
89 <div class="card-body"
90 <div class="card-body"
91 <div class="card-body"
92 <div class="card-body"
93 <div class="card-body"
94 <div class="card-body"
95 <div class="card-body"
96 <div class="card-body"
97 <div class="card-body"
98 <div class="card-body"
99 <div class="card-body"
100 <div class="card-body"

```

Gambar 1. 673 Source Code Halaman Rekam Medis

12. *laporan.blade.php*

1. *Source code* ini merupakan halaman laporan kunjungan pasien oleh dokter yang berada di ***folder resources > views > dokter > laporan.blade.php***.
2. *Source code* ini menyediakan fitur filter laporan berdasarkan:
 - a. Per Hari: menggunakan input bertipe *date*.
 - b. Per Bulan: menggunakan input bertipe *month*.
 - c. *Radio button* digunakan untuk memilih jenis filter, dan tampilan filter akan ditampilkan atau dihilangkan berdasarkan pilihan tersebut.
3. *Source code* ini juga menyediakan fitur ekspor data ke dalam dua format:
 - a. *Export Excel* (button hijau)
 - b. *Export PDF* (button merah)
4. Bagian laporan ditampilkan dalam bentuk tabel responsif dengan kolom:
 - a. No
 - b. Tanggal
 - c. Nama Pasien
 - d. Keluhan
 - e. Diagnosis
 - f. Obat/Resep
5. Data ditampilkan menggunakan *@forelse*, dan jika data kosong, akan menampilkan pesan *fallback*: “Tidak ada data” dalam satu baris tabel.

```

1 <@extends('layouts.default')
2
3 @section('title', 'Laporan')
4
5 @section('content')
6 <div class="container-fluid">
7 <div class="row">
8 <div class="col-md-12">
9 <div class="card">
10 <div class="card-header">
11 <h3>Laporan Kumpulan Pasien</h3>
12 </div>
13 <div class="card-body">
14 <div class="form-check">
15 <input type="checkbox" value="hari" /> Hari
16 </div>
17 <div class="form-check">
18 <input type="checkbox" value="bulan" /> Bulan
19 </div>
20 <div class="form-check">
21 <input type="checkbox" value="tahun" /> Tahun
22 </div>
23 </div>
24 </div>
25 </div>
26 <div class="col-md-12">
27 <div class="card">
28 <div class="card-header">
29 <h3>Laporan Kumpulan Pasien</h3>
30 </div>
31 <div class="card-body">
32 <div class="form-check">
33 <input type="checkbox" value="hari" /> Hari
34 </div>
35 <div class="form-check">
36 <input type="checkbox" value="bulan" /> Bulan
37 </div>
38 <div class="form-check">
39 <input type="checkbox" value="tahun" /> Tahun
40 </div>
41 </div>
42 </div>
43 </div>
44 </div>
45 </div>
46 </div>
47 </div>
48 </div>
49 </div>
50 </div>
51 </div>
52 </div>
53 </div>
54 </div>
55 </div>
56 </div>
57 </div>
58 </div>
59 </div>
60 </div>
61 </div>
62 </div>
63 </div>
64 </div>
65 </div>
66 </div>
67 </div>
68 </div>
69 </div>
70 </div>
71 </div>
72 </div>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
78 </div>
79 </div>
80 </div>
81 </div>
82 </div>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </div>
92 </div>
93 </div>
94 </div>
95 </div>
96 </div>
97 </div>
98 </div>
99 </div>
100 </div>

```

Gambar 1. 705 Source Code Halaman Laporan

2. Configure Database

1. .env

1. Source code ini merupakan *file* konfigurasi lingkungan (.env) dalam *framework* *Laravel*. Berfungsi untuk menyimpan informasi penting terkait konfigurasi sistem aplikasi.
2. Source code ini berisi pengaturan utama *Laravel*, seperti:
 1. *APP_NAME*: Nama aplikasi *Laravel*.
 2. *APP_ENV*: Lingkungan aplikasi (*local*, *production*).
 3. *APP_KEY*: Kunci enkripsi aplikasi.
 4. *APP_DEBUG*: Aktifkan/Nonaktifkan mode *debug*.
 5. *APP_TIMEZONE*: Zona waktu (*Asia/Jakarta*).
3. Source code ini juga memuat konfigurasi:
 - a. *LOCALE*: Bahasa lokal (*id*) dan *fallback*.
 - b. *Database* (*MySQL*):
 1. *DB_CONNECTION=mysql*
 2. *DB_HOST=127.0.0.1*, *DB_PORT=3306*
 3. Nama *database*: *klinik-pratama*, *Username*: *root*, *password* kosong.
 - c. *Session*:
 1. *Driver*: *database*

2. Enkripsi aktif
 3. *Lifetime*: 120 menit
- d. *Log*: *Channel* menggunakan *stack*, *level debug*.
- e. *Source code* ini juga mencakup:
1. *Queue & Cache*:
QUEUE_CONNECTION=database,
CACHE_STORE=database
 2. *Mail*: *MAIL_MAILER=log*, *MAIL_PORT=2525*
 3. [*MAIL_FROM_ADDRESS=hello@example.com*](mailto:hello@example.com)
 4. *Redis & AWS* konfigurasi juga tersedia (belum diisi lengkap).

```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:+G8EZAu/OEMPDv3b1Wk36KwStmE03MNC44ssyqkDCnQ=
4 APP_DEBUG=true
5 APP_TIMEZONE=Asia/Jakarta
6 APP_URL=http://127.0.0.1:8000
7
8 APP_LOCALE=id
9 APP_FALLBACK_LOCALE=id
10 APP_FAKER_LOCALE=id_ID
11
12 APP_MAINTENANCE_DRIVER=file
13 # APP_MAINTENANCE_STORE=database
14
15 PHP_CLI_SERVER_WORKERS=4
16
17 BCRYPT_ROUNDS=12
18
19 LOG_CHANNEL=stack
20 LOG_STACK=single
21 LOG_DEPRECATIONS_CHANNEL=null
22 LOG_LEVEL=debug
23
24 DB_CONNECTION=mysql
25 DB_HOST=127.0.0.1
26 DB_PORT=3306
27 DB_DATABASE=klinik-pratama
28 DB_USERNAME=root
29 DB_PASSWORD=
30
31 SESSION_DRIVER=database
32 SESSION_LIFETIME=120
33 SESSION_ENCRYPT=true
34 SESSION_PATH=/
35 SESSION_DOMAIN=null
36
37 BROADCAST_CONNECTION=log
38 FILESYSTEM_DISK=local
39 QUEUE_CONNECTION=database
40
41 CACHE_STORE=database
42 CACHE_PREFIX=
43
44 MEMCACHED_HOST=127.0.0.1
```

Gambar 1. 737 Source Code Configure Database

3. *Class Models*

1. *Pendaftaran.php*

1. *Source code* ini merupakan model *Eloquent Laravel* untuk data pendaftaran pasien dan terletak pada ***folder app > Models > Pendaftaran.php***.
2. *Source code* ini juga menjelaskan relasi antara model Pendaftaran dengan model :
 - a. Pasien → menggunakan relasi *belongsTo*,
 - b. Dokter → menggunakan relasi *belongsTo*,
 - c. KajianAwal → menggunakan relasi *hasOne*.
3. *Source code* ini menyediakan fungsi untuk:
 1. Menghasilkan nomor antrian berdasarkan dokter dan tanggal,
 2. Mengecek apakah pasien sudah terdaftar,
 3. Mendapatkan total pendaftaran per dokter per tanggal,
 4. Mengatur *scope query* berdasarkan tanggal, dokter, dan status.
4. *Source code* ini mendefinisikan atribut seperti:
 - a. *nomor_antrian*,
 - b. *pasien_id*,
 - c. *dokter_id*,
 - d. *tanggal_Registrasi*
 - e. *status*.

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class Pendaftaran extends Model
9 {
10     use HasFactory;
11
12     protected $fillable = [
13         'nomor_antrian',
14         'pasien_id',
15         'dokter_id',
16         'tanggal_registrasi',
17         'status',
18     ];
19
20     protected $dates = ['tanggal_registrasi'];
21
22     // Relasi dengan pasien
23     public function pasien()
24     {
25         return $this->belongsTo(Pasien::class);
26     }
27
28     // Relasi dengan dokter
29     public function dokter()
30     {
31         return $this->belongsTo(Dokter::class);
32     }
33
34     public function kajianAwal()
35     {
36         return $this->hasOne(KajianAwal::class);
37     }
38
39     /**
40      * Generate nomor antrian berdasarkan dokter dan tanggal
41      * Method yang benar - sesuai dengan yang ada di controller
42      */
43     public static function generateNomorAntrian($dokterId, $tanggal)
44     {
45         // Hitung jumlah pendaftaran pada dokter dan tanggal yang sama
46         $count = self::where('dokter_id', $dokterId)
47             ->whereDate('tanggal_registrasi', $tanggal)
48             ->count();
49
50         // Format: ABB1, ABB2, dst
51         return 'A' . str_pad($count + 1, 3, '0', STR_PAD_LEFT);
52     }
53 }

```

Gambar 1. 769 Source Code Model Pendaftaran

2. Dokter.php

1. Source code ini merupakan model *Eloquent Laravel* untuk entitas Dokter dan berada pada **folder app > Models > Dokter.php**.
2. Source code ini mendefinisikan atribut *\$fillable* yang dapat diisi massal: *User_id*, nama, jenis_kelamin, alamat, spesialis, telepon, jam_mulai, jam_selesai.
3. Source code ini juga menjelaskan relasi antar model:
 - a. *One-to-Many*:
 1. *jadwals()* → relasi ke *JadwalDokter*
 2. *kajianAwals()* → relasi ke *KajianAwal*
 - b. *Many-to-One*:
 1. *User()* → relasi ke *User* (dokter dimiliki oleh *User*)
4. Source code ini memiliki *accessor* khusus:
 - a. *getJamPraktekAttribute()*

1. Mengubah format jam_mulai dan jam_selesai dari 'H:i:s' menjadi 'H:i'
2. Mengembalikan *string* seperti: "08:00 - 11:00 WIB" menggunakan *library Carbon*.

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Carbon\Carbon;
8
9 class Dokter extends Model
10 {
11     use HasFactory;
12
13     protected $fillable = [
14         'user_id',
15         'nama',
16         'jenis_kelamin',
17         'alamat',
18         'spesialis',
19         'telepon',
20         'jam_mulai',
21         'jam_selesai',
22     ];
23
24     // Relasi One-to-Many: Dokter punya banyak jadwal
25     public function jadwal()
26     {
27         return $this->hasMany(JadwalDokter::class);
28     }
29
30     public function user()
31     {
32         return $this->belongsTo(User::class);
33     }
34
35     public function kajianAwais()
36     {
37         return $this->hasMany(KajianAwais::class);
38     }
39
40     public function getJamPraktekAttribute()
41     {
42         {
43             $mulai = Carbon::createFromFormat('H:i:s', $this->jam_mulai)->format('H:i');
44             $selesai = Carbon::createFromFormat('H:i:s', $this->jam_selesai)->format('H:i');
45             return $mulai - $selesai WIB";
46         }
47     }
48 }

```

Gambar 1. 801 Source Code Model Dokter

3. *Jadwaldokter.php*

1. *Source code* ini merupakan model *Eloquent Laravel* untuk jadwal praktik dokter dan berada di **folder app > Models > JadwalDokter.php**.
2. *Source code* ini mengatur :
 - a. Nama tabel khusus: *protected \$table* = 'jadwal_dokters'.
 - b. *Field* yang dapat diisi massal (*\$fillable*): *dokter_id*, hari, jam_mulai, jam_selesai. Konversi otomatis *field* hari dari *JSON* ke *array* menggunakan *\$casts*.
3. *Source code* ini menjelaskan relasi *belongsTo* ke model Dokter.
4. Terdapat dua *accessor*:
 - a. *getJamPraktekAttribute()* :

1. Mengubah waktu `jam_mulai` dan `jam_selesai` menjadi format H:i,
 2. Contoh *output*: 08:00 - 11:00.
- b. `getHariDisplayAttribute()` : Menyusun *array* hari menjadi string dengan koma, contoh : Senin, Rabu, Jumat.

```

1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7
8 class JadwalDokter extends Model
9 {
10     use HasFactory;
11
12     protected $table = 'jadwal_dokters';
13
14     protected $fillable = [
15         'dokter_id',
16         'hari',
17         'jam_mulai',
18         'jam_selesai',
19     ];
20
21     protected $casts = [
22         'hari' => 'array', // Cast JSON ke array
23     ];
24
25     // Relasi ke model Dokter
26     public function dokter()
27     {
28         return $this->belongsTo(Dokter::class);
29     }
30
31     // Accessor untuk menampilkan jam praktek
32     public function getJamPraktekAttribute()
33     {
34         $jamMulai = date('H:i', strtotime($this->jam_mulai));
35         $jamSelesai = date('H:i', strtotime($this->jam_selesai));
36         return $jamMulai . ' - ' . $jamSelesai;
37     }
38

```

Gambar 1. 833 Source Code Model Jadwaldokter

4. *Pasien.php*

1. *Source code* ini merupakan model *Eloquent Laravel* untuk data pasien dan berada di **folder app > Models > Pasien.php**.
2. *Source code* ini mendefinisikan *field-field* yang dapat diisi massal melalui *\$fillable*, meliputi:
 - a. Data pribadi pasien: nama, nik, tempat_lahir, tanggal_lahir, jenis_kelamin, nama_ayah, nama_ibu, agama, alamat, pendidikan, pekerjaan, status.
3. *Source code* ini juga mengatur agar *field* tanggal_lahir dikenali sebagai tipe tanggal melalui properti *\$dates*.
4. *Source code* ini menjelaskan relasi *One-to-Many*

```

1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Pasien extends Model
9  {
10     use HasFactory;
11
12     protected $fillable = [
13         'nama',
14         'nik',
15         'tempat_lahir',
16         'tanggal_lahir',
17         'jenis_kelamin',
18         'nama_ayah',
19         'nama_ibu',
20         'agama',
21         'alamat',
22         'pendidikan',
23         'pekerjaan',
24         'status',
25         'penanggung_hubungan',
26         'penanggung_nama',
27         'penanggung_alamat',
28         'penanggung_pekerjaan',
29         'penanggung_gender',
30         'penanggung_agama',
31         'penanggung_status',
32         'no_whatsapp',
33     ];
34
35     protected $dates = ['tanggal_lahir'];
36

```

Gambar 1. 865 Source Code Model

5. *Kajianawal.php*

1. *Source code* ini merupakan model *Eloquent Laravel* untuk entitas kajian awal pasien, dan berada pada **folder app > Models > KajianAwal.php**.
2. *Source code* ini mendefinisikan *field \$fillable* yang dapat diisi massal, meliputi : *pendaftaran_id*, *dokter_id*, *suhu_tubuh*, *tekanan_darah*, *tinggi_badan*, *berat_badan*, *keluhan*, *diagnosis*, *obat*, *nomor_rekam_medis*
3. *Source code* ini juga menjelaskan relasi antar model :
 - a. *pendaftaran()* → *belongsTo* Pendaftaran
 - b. *dokter()* → *belongsTo* Dokter
 - c. *pasien()* → *belongsTo* Pasien

```

1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Model;
6
7  class KajianAwal extends Model
8  {
9      protected $fillable = [
10         'pendaftaran_id',
11         'dokter_id',
12         'suhu_tubuh',
13         'tekanan_darah',
14         'tinggi_badan',
15         'berat_badan',
16         'keluhan',
17         'diagnosis',
18         'obat',
19         'nomor_rekam_medis'
20     ];
21
22     public function pendaftaran()
23     {
24         return $this->belongsTo(Pendaftaran::class);
25     }
26
27     public function dokter()
28     {
29         return $this->belongsTo(Dokter::class);
30     }
31
32     public function pasien()
33     {
34         return $this->belongsTo(Pasien::class);
35     }
36

```

Gambar 1. 897 Source Code Model Kajianawal

6. User.php

1. Source code ini merupakan model *Eloquent Laravel* untuk entitas *User* (pengguna) dan berada di **folder app > Models > User.php**.
2. Source code ini mewarisi *Authenticatable*, yang digunakan untuk sistem *Login Laravel*.
3. Field yang dapat diisi secara massal (*\$fillable*) meliputi : *name, email, password, role_id, phone, is_active*.
4. Field yang disembunyikan dari serialisasi (*\$hidden*): *password, remember_token*.
5. Field yang dikonversi (*\$casts*) ke tipe data tertentu :
 - a. *email_verified_at* → *datetime*
 - b. *password* → *hashed*
 - c. *is_active* → *boolean*
6. Source code ini menjelaskan relasi :
 - a. *role()* → *belongsTo* ke model *Role*

- b. *dokter()* → *hasOne* ke model Dokter berdasarkan *User_id*
7. Tersedia *method* untuk pengecekan peran :
- a. *hasRole(\$role)* → cek apakah *User* memiliki peran tertentu
 - b. *isAdmin()* → cek apakah *User* adalah admin
 - c. *isDokter()* → cek apakah *User* adalah dokter



```

1 <?php
2
3 namespace App\Models;
4
5 // use Illuminate\Contracts\Auth\MustVerifyEmail;
6 use Illuminate\Database\Eloquent\Factories\HasFactory;
7 use Illuminate\Foundation\Auth\User as Authenticatable;
8 use Illuminate\Notifications\Notifiable;
9
10 class User extends Authenticatable
11 {
12     use HasFactory, Notifiable;
13
14     /**
15      * The attributes that are mass assignable.
16      *
17      * @var array<int, string>
18      */
19     protected $fillable = [
20         'name',
21         'email',
22         'password',
23         'role_id',
24         'phone',
25         'is_active',
26     ];
27
28     /**
29      * The attributes that should be hidden for serialization.
30      *
31      * @var array<int, string>
32      */
33     protected $hidden = [
34         'password',
35         'remember_token',
36     ];
37
38     /**
39      * Get the attributes that should be cast.
40      *
41      * @return array<string, string>
42      */
43     protected function casts(): array
44     {
45         return [
46             'email_verified_at' => 'datetime',
47             'password' => 'hashed',
48             'is_active' => 'boolean',
49         ];
50     }
51 }

```

Gambar 1. 929 Source Code Model User

4. Class Controller

1. AuthController.php

1. Source code ini merupakan *controller* autentikasi pengguna yang berada di **folder app > Http > Controllers > AuthController.php**.
2. Source code ini menangani berbagai proses autentikasi, yaitu:
 - a. Login : Method: Login(Request \$request)
 1. Fitur : Validasi *input email* dan *password*, Autentikasi pengguna menggunakan *Auth::attempt*, Cek apakah akun aktif (*is_active*) dan memiliki *role*.

- b. *Redirect* berdasarkan *role*:
 - 1. admin → ke admin.*Dashboard*
 - 2. dokter → ke dokter.*Dashboard*
 - 3. lainnya → ke /
 - 4. Jika gagal *Login*, dikembalikan ke halaman *Login* dengan pesan *error*.
- c. *Form Login* : *Method: showLoginForm()*
 - 1. Fitur : Menampilkan view *auth.Login*.
- d. *Register* (khusus Admin) : *Method: Register(Request \$request)*.
 - 1. Fitur : Validasi input: nama, *email*, *password*, *role*, dan nomor telepon.
 - 2. Buat *User* baru dengan *password* yang di-hash.
 - 3. *Auto Login* setelah *Registrasi*.
 - 4. *Redirect* sesuai *role* (admin atau dokter atau *default*).
- e. *Form Registrasi* : *Method: showRegisterAdminForm()*
 - 1. Fitur : Mengambil data *role* Admin dari *database*.
 - 2. Menampilkan view *auth.Register* dengan data *role* admin.
- f. *Logout* : *Method: Logout(Request \$request)*
 - 1. Fitur : *Logout* pengguna dengan *Auth::Logout*.
 - 2. Menghapus *session* dan token lalu *redirect* ke halaman *Login (/Login)*.

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\Auth;
7 use Illuminate\Support\Facades\Hash;
8 use App\Models\User;
9 use App\Models\Role;
10
11 class AuthController extends Controller
12 {
13     /**
14      * Show the login form
15      */
16     public function showLoginForm()
17     {
18         return view('auth.login');
19     }
20
21     /**
22      * Handle login request
23      */
24     public function login(Request $request)
25     {
26         $request->validate([
27             'email' => 'required|email',
28             'password' => 'required',
29         ]);
30
31         $credentials = $request->only('email', 'password');
32
33         if (Auth::attempt($credentials, $request->filled('remember'))) {
34             $request->session()->regenerate();
35
36             // eager load the role relationship
37             $user = Auth::user()->load('role');
38
39             // Check if user is active
40             if (!$user->is_active) {
41                 Auth::logout();
42                 return back()->withErrors([
43                     'email' => 'Akan anda tidak aktif. Silakan hubungi administrator.',
44                 ]);
45             }
46
47             // Check if user has role
48             if (!$user->role) {
49                 Auth::logout();
50                 return back()->withErrors([
51                     'email' => 'Role pengguna tidak ditemukan. Silakan hubungi administrator.',
52                 ]);
53             }
54         }
55     }
56 }

```

Gambar 1. 961 Source Code AuthController

2. DashboardController.php

1. Source code ini adalah *controller* untuk halaman *Dashboard* admin, berada di **folder app > Http > Controllers > DashboardController.php**.

2. Terdapat dua fungsi utama: *Dashboard()* dan *chartData()*.

a. *Dashboard()* : Menampilkan statistik kunjungan pasien ke dalam *Dashboard*.

1. Fitur : totalHariIni: menghitung total pasien berdasarkan tanggal hari ini.
2. totalUmum: total pasien yang ditangani oleh dokter dengan spesialisasi Umum.
3. totalGigi: total pasien yang ditangani oleh dokter dengan spesialisasi Gigi.
4. totalPasien: total seluruh pendaftaran pasien tanpa filter.
5. menunggu: jumlah pasien dengan status Belum Kajian Awal.

- b. *Return: View admin.Dashboard* dengan data statistik di-compact.
- c. *chartData()* : Digunakan untuk menampilkan data grafik jumlah pasien berdasarkan tanggal (7 hari terakhir).
 1. Fitur : *Query* data dari tabel pendaftaran, *group* berdasarkan tanggal *Registrasi*.
 2. Ambil maksimal 7 data terakhir dan urutkan secara naik.
 3. Ubah tanggal ke format lokal d M menggunakan *Carbon*.
 4. Hasil *return* dalam bentuk *JSON* untuk digunakan di grafik *frontend* : *labels: array* tanggal, *data: array* jumlah pasien per tanggal

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Pendaftaran;
6 use Illuminate\Http\Request;
7 use Carbon\Carbon;
8 use Illuminate\Support\Facades\DB;
9
10 class DashboardController extends Controller
11 {
12
13     public function dashboard()
14     {
15         $today = Carbon::today();
16
17         // Total pasien hari ini
18         $totalHariIni = Pendaftaran::whereDate('tanggal_registrasi', $today)->count();
19
20         // Total pasien berdasarkan poli
21         $totalPoli = Pendaftaran::whereNotNull('dokter', function ($query) {
22             $query->where('specialis', 'User');
23         }->count());
24
25         $totalSpesialis = Pendaftaran::whereNotNull('dokter', function ($query) {
26             $query->where('specialis', 'Digi');
27         }->count());
28
29         // Total seluruh pasien
30         $totalPasien = Pendaftaran::count();
31
32         // Pasien minggu (belum kajian awal)
33         $seminggu = Pendaftaran::where('status', 'Belum Kajian Awal')->count();
34
35         return view('admin.dashboard', compact(
36             'totalHariIni',
37             'totalPoli',
38             'totalSpesialis',
39             'totalPasien',
40             'seminggu'
41         ));
42     }
43
44     public function chartData()
45     {
46         $data = Pendaftaran::selectRaw('DATE(tanggal_registrasi) as tanggal, COUNT(*) as jumlah')
47             ->groupBy('tanggal')
48             ->orderBy('tanggal', 'asc')
49             ->take(7) // ambil 7 hari terakhir
50             ->get();
51
52         $labels = [];
53         $jumlah = [];
54
55         foreach ($data as $d) {
56             $labels[] = Carbon::parse($d->tanggal)->translatedFormat('d M');
57             $jumlah[] = $d->jumlah;
58         }
59
60         return response()->json([
61             'labels' => $labels,
62             'data' => $jumlah
63         ]);
64     }
65 }

```

Gambar 1. 993 Source Code DashboardController

3. DokterController.php

1. Berfungsi sebagai *controller CRUD* untuk manajemen data dokter dan akun *User* yang terkait.

2. *Source code* ini adalah *controller* untuk halaman dokter, berada di ***folder app > Http > Controllers > DokterController.php***.
3. *index()* : Menampilkan daftar semua data dokter (*Dokter::all()*).
4. Menampilkan *User* yang memiliki *role* dokter melalui *query* relasi *whereHas('role')*.
5. *Return: view('admin.datadokter')* dengan *variabel* dokters dan *Users*.
6. *store(Request \$request)* :
 - a. Menyimpan data dokter baru sekaligus membuat akun *User* baru.
 - b. Validasi *input* : *name, email, password, jenis_kelamin, alamat, spesialis, telepon, jam_mulai, jam_selesai*.
 - c. Proses : Membuat akun *User* baru dengan *role* dokter (*role_id = 2*), Menyimpan data dokter yang di-link ke *User* tersebut via *User_id*.
7. *Return: redirect back* dengan pesan sukses.
8. *update(Request \$request, \$id)* :
 - a. *Mengupdate* data dokter berdasarkan *ID*.
 - b. Validasi: *nama, jenis_kelamin, alamat, spesialis, telepon, jam_mulai, jam_selesai*., Menggunakan *update(\$request->all())* untuk menyimpan perubahan.
9. *Return: redirect ke route dokter.index* dengan pesan sukses.
10. *destroy(\$id)* : Menghapus data dokter berdasarkan *ID*, Menggunakan *findOrFail()* lalu *delete()*.
11. *Return: redirect back* dengan notifikasi sukses.
12. *updateAkun(Request \$request, \$id)* : *Mengupdate* akun *User* dokter : Validasi *email* harus unik (kecuali milik sendiri), dan *password* opsional (*nullable, min:6, confirmed*), Jika

password diisi, akan di-encrypt dengan *bcrypt()*, Simpan perubahan dengan *\$User->save()*.

13. *Return: redirect back* dengan notifikasi bahwa akun berhasil diperbarui.

```

1 <?php
2 namespace App\Http\Controllers;
3
4 use App\Models\User;
5 use App\Models\Dokter;
6 use Illuminate\Http\Request;
7
8 class DokterController extends Controller
9 {
10     public function index()
11     {
12         $dokters = Dokter::all();
13         $users = User::where('role', function ($q) {
14             $q where('name', 'Dokter');
15         }->get());
16
17         return view('admin.data.dokter', compact('dokters', 'users'));
18     }
19
20     public function store(Request $request)
21     {
22         $request->validate([
23             'name' => 'required|string|max:255',
24             'email' => 'required|email|max:191|unique:users,email',
25             'password' => 'required|string|min:6',
26             'jenis_kelamin' => 'required|in:laki-laki,perempuan',
27             'alamat' => 'required|string',
28             'spesialis' => 'required|string',
29             'telepon' => 'required|string|max:11',
30             'jam_mulai' => 'required|date_format:H:i',
31             'jam_selesai' => 'required|date_format:H:i|after:jam_mulai',
32         ]);
33
34         // Buat akun user dokter
35         $user = User::create([
36             'name' => $request->name,
37             'email' => $request->email,
38             'password' => bcrypt($request->password),
39             'role_id' => 2, // 2 - dokter
40             'phone' => $request->telepon,
41         ]);
42
43         // Buat data dokter
44         Dokter::create([
45             'user_id' => $user->id,
46             'name' => $request->name,
47             'jenis_kelamin' => $request->jenis_kelamin,
48             'alamat' => $request->alamat,
49             'spesialis' => $request->spesialis,
50             'telepon' => $request->telepon,
51             'jam_mulai' => $request->jam_mulai,
52             'jam_selesai' => $request->jam_selesai,
53         ]);
54
55         return redirect()->back()->with('success', 'Akun dan data dokter berhasil dibuat!');
56     }
57 }

```

Gambar 1. 1025 Source Code DokterController

4. *Jadwaldoktercontroller.php*

1. Berfungsi sebagai *controller* pengelolaan jadwal praktik dokter.
2. Lokasi *file* di **folder app > Http > Controllers > JadwalDokterController.php**.
3. *urutkanHari(array \$hariArray): array*
 - a. *Method privat* untuk mengurutkan *array* hari sesuai format standar : ['Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat', 'Sabtu', 'Minggu'].
 - b. Digunakan untuk memastikan urutan hari yang konsisten.
4. *index()* : Mengambil seluruh data jadwal dokter (relasi dokter) dan semua dokter.

5. *Return: view('admin.jadwaldokter')* dengan *variabel* jadwals dan dokters.
6. *store(Request \$request)* :
 - a. Menyimpan atau meng-*update* jadwal dokter.
 - b. Validasi: *dokter_id*, *hari*, *jam_mulai*, *jam_selesai* (*jam_selesai* harus setelah *jam_mulai*).
 - c. Proses : Konversi *input* hari dari *string* menjadi *array*, lalu validasi dan urutkan, Jika jadwal sudah ada: *update* hari dan jam, Jika belum ada: buat data baru.
7. *Return: redirect* dengan notifikasi sukses.
8. *update(Request \$request, \$id)* :
 - a. Mengubah data jadwal berdasarkan *ID*.
 - b. Validasi: *dokter_id* dan *hari*, Memastikan hari adalah valid dan mengurutkan menggunakan *urutkanHari()*,
 - c. Menyimpan perubahan *dokter_id* dan *hari*.
9. *Return: redirect back* dengan pesan berhasil diperbarui.
10. *destroy(\$id)* :
 - a. Menghapus data jadwal berdasarkan *ID*.
 - b. Menggunakan *findOrFail()* dan *delete()*.
11. *Return: redirect back* dengan pesan berhasil dihapus.

```

1  /ign
2
3  namespace App\Http\Controllers;
4
5  use App\Models\JadwalDokter;
6  use App\Models\Dokter;
7  use Illuminate\Http\Request;
8
9  class JadwalDokterController extends Controller
10 {
11     /**
12      * Tampilkan semua jadwal dokter
13      */
14     private function urutkanHari(array $hariArray): array
15     {
16         $urutkanHari = ['Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat', 'Sabtu', 'Minggu'];
17         return array_values(array_intersect($urutkanHari, $hariArray));
18     }
19
20     public function index()
21     {
22         $jadwals = JadwalDokter::with('dokter')->get();
23         $dokter = Dokter::all();
24         return view('admin.jadwaldokter', compact('jadwals', 'dokter'));
25     }
26
27     public function store(Request $request)
28     {
29         $request->validate([
30             'dokter_id' => 'required|exists:dokter,id',
31             'hari' => 'required|string',
32             'jam_mulai' => 'required|date_format:H:i',
33             'jam_selesai' => 'required|date_format:H:i|after:jam_mulai',
34         ]);
35
36         $hariArray = explode('-', $request->hari);
37         $hariArray = array_map('trim', $hariArray);
38
39         if (in_array('Senin', $hariArray)) {
40             $hariArray = ['Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat', 'Sabtu'];
41         }
42
43         $urutkanHari = ['Senin', 'Selasa', 'Rabu', 'Kamis', 'Jumat', 'Sabtu', 'Minggu'];
44         $hariArray = array_intersect($uriArray, $urutkanHari);
45         $hariArray = $uriArray-surutkanHari($uriArray);
46
47         if (empty($uriArray)) {
48             return redirect()-back()-with('error', 'Pilih minimal satu hari yang valid.');
```

Gambar 1. 1057 Source Code JadwaldokterController

5. Kajiancontroller.php

1. *Controller* untuk kajian awal medis oleh dokter.
2. Lokasi *file* di **folder app > Http > Controllers > KajianController.php**.
3. *kajianAwalPasien()* :
 - a. Menampilkan daftar pasien yang belum memiliki kajian awal.
 - b. Filter: pasien yang terdaftar di hari yang sama dengan *Login* dokter (*Carbon::today()*), belum punya entri di *kajian_awals*.
4. *Return: view('dokter.kajianawals.index')* dengan data *\$kajianAwal*.
5. *store(Request \$request)* :
 - a. Menyimpan data kajian awal.
 - b. Validasi : *suhu_tubuh*, *tekanan_darah*, *tinggi_badan*, *berat_badan*, *keluhan*, *diagnosis*, *obat*, *nomor_rekam_medis*.
 - c. Pemeriksaan tambahan : Pastikan dokter *Login* hanya *menginput* pasien yang ditugaskan padanya. Menyimpan data ke tabel *kajian_awals* via relasi ->*create()*.
6. *Return: redirect back* dengan pesan sukses atau *error* jika terjadi *exception*.
7. *rekamMedis()* :
 - a. Menampilkan semua data pasien yang sudah memiliki kajian awal.
 - b. Filter: hanya milik dokter yang sedang *Login*.
 - c. Mengambil data dari tabel *pendaftarans*, *pasiens*, *kajian_awals*.
8. *Return: view('dokter.rekamedis.index')* dengan data *rekamMedis*.
9. *daftarPasien()* :

- a. Menampilkan daftar seluruh pasien yang pernah ditangani dokter *Login*, Termasuk pasien yang belum ada kajian awal.
10. *Return: view ('dokter.datapasien', compact ('dataPasien'))*.
11. *updatePasien(Request \$request) :*
- a. Melakukan *update* terhadap data pasien dari form dokter.
 - b. Validasi data lengkap: biodata pasien dan penanggung jawab.
 - c. Proses : Cari data pendaftaran berdasarkan *ID*, Ambil relasi pasien, lalu *update* seluruh atributnya.
 - d. *Update* juga tanggal *Registrasi* dari pendaftaran.
12. *Return: redirect* ke daftar pasien dengan pesan berhasil atau *error handling*



```

1  <?php
2  namespace App\Http\Controllers;
3
4  use Illuminate\Http\Request;
5  use App\Models\Pasien;
6  use App\Models\Dokter;
7  use App\Models\Pendaftaran;
8  use App\Models\PasienDokter;
9  use App\Models\PasienDokter;
10 use App\Models\PasienDokter;
11 use App\Models\PasienDokter;
12
13 class KajianController extends Controller
14 {
15     // Menampilkan daftar seluruh pasien yang pernah ditangani dokter yang sedang login.
16     public function index()
17     {
18         $dokter = auth()->user();
19
20         // Validasi data dokter
21         if ($dokter instanceof Dokter) {
22             $dokter = $dokter;
23         } else {
24             return redirect()->back();
25         }
26
27         // Cari data pendaftaran berdasarkan ID, Ambil relasi pasien, lalu update seluruh atributnya.
28         $daftar = Pendaftaran::where('dokter_id', $dokter->id)
29             >> with(['pasien', 'dokter'])
30             >> get();
31
32         // Return view
33         return view('dokter.datapasien', compact('daftar'));
34     }
35
36     // Update data pasien
37     public function updatePasien(Request $request)
38     {
39         // Validasi data
40         $request->validate([
41             'nama' => 'required|string|max:255',
42             'alamat' => 'required|string|max:255',
43             'telepon' => 'required|string|max:255',
44             'email' => 'required|string|max:255',
45             'password' => 'required|string|max:255',
46             'confirm_password' => 'required|string|max:255',
47             'tanggal_registrasi' => 'required|date',
48         ]);
49
50         // Cari data pendaftaran berdasarkan ID, Ambil relasi pasien, lalu update seluruh atributnya.
51         $daftar = Pendaftaran::where('dokter_id', $request->id_dokter)
52             >> with(['pasien', 'dokter'])
53             >> get();
54
55         // Return view
56         return view('dokter.datapasien', compact('daftar'));
57     }
58
59     // Update data pendaftaran
60     public function updateRegistrasi(Request $request)
61     {
62         // Validasi data
63         $request->validate([
64             'id_pasien' => 'required|integer',
65             'id_dokter' => 'required|integer',
66             'tanggal_registrasi' => 'required|date',
67         ]);
68
69         // Cari data pendaftaran berdasarkan ID, Ambil relasi pasien, lalu update seluruh atributnya.
70         $daftar = Pendaftaran::where('id_pasien', $request->id_pasien)
71             >> with(['pasien', 'dokter'])
72             >> get();
73
74         // Return view
75         return view('dokter.datapasien', compact('daftar'));
76     }
77 }

```

Gambar 1. 1089 Source Code KajianController

6. Laporancontroller.php

1. *Controller* untuk menampilkan dan mengekspor laporan kajian awal pasien.
2. Lokasi file di **folder app > Http > Controllers > LaporanController.php**.
3. *index(Request \$request) :*
 - a. Menampilkan laporan kajian awal pada halaman view.

- b. Data diambil dari model *KajianAwal* beserta relasi *pendaftaran.pasien*.
 - c. Mendukung filter laporan berdasarkan waktu : Per hari: jika *filter_type == 'hari'* dan tanggal tersedia. Per bulan: jika *filter_type == 'bulan'* dan bulan tersedia.
 - d. Format bulan: 'YYYY-MM' (misal: '2025-07').
4. *Return view* dokter.laporan dengan data hasil filter.
5. *exportPdf(Request \$request)* :
- a. Mengekspor data laporan ke dalam bentuk *file PDF* .
 - b. Data diambil dari *KajianAwal* beserta relasi *pendaftaran.pasien* dan dokter, kemudian di-map ke struktur ringkas : tanggal, nomor_rm, nama, keluhan, diagnosis, obat.
 - c. Data yang kosong akan diganti dengan '-'. *PDF* dibuat menggunakan *library dompdf* melalui *Pdf::loadView(...)*
 - d. *View PDF*: laporan.pdf.
 - e. Ukuran kertas: A4, orientasi: *portrait*.
 - f. Nama *file* unduhan: laporan_kunjungan_pasien.pdf.
6. *exportExcel()* :
- a. Mengekspor data laporan dalam format *file Excel (.xlsx)*
 - b. Menggunakan *package Maatwebsite\Excel*
 - c. Menggunakan *Class export LaporanKajianAwalExport*.
 - d. Nama *file* unduhan: laporan_kunjungan_pasien.xlsx.

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\KajianAwal;
6 use Barryvdh\DomPDF\Facade\Pdf;
7 use Illuminate\Http\Request;
8 use PhpOffice\PhpSpreadsheet\Writer\Xlsx;
9 use App\Exports\LaporanKunjunganAwalExport;
10
11 class LaporanController extends Controller
12 {
13     public function index(Request $request)
14     {
15         $query = KajianAwal::with('pendaftaran.pasien')->orderBy('created_at', 'desc');
16
17         // Filter per hari
18         if ($request->filter_type == 'hari' && $request->filled('tanggal')) {
19             $query->whereDate('created_at', $request->tanggal);
20         }
21
22         // Filter per bulan
23         if ($request->filter_type == 'bulan' && $request->filled('bulan')) {
24             $query->whereMonth('created_at', substr($request->bulan, 0, 4));
25             $query->whereMonth('created_at', substr($request->bulan, 5, 2));
26         }
27
28         $data = $query->get();
29
30         return view('dokter.laporan', compact('data'));
31     }
32
33     public function exportPDF(Request $request)
34     {
35         // Ambil data dari database, sesuaikan dengan struktur kamu
36         $data = KajianAwal::with('pendaftaran.pasien', 'dokter')
37             ->orderBy('created_at', 'desc')
38             ->get();
39         $pdf = Pdf::loadView('dokter.laporan', compact('data'));
40         return $pdf->download('laporan_kunjungan_pasien.pdf');
41     }
42
43     public function exportExcel()
44     {
45         return Excel::download(new LaporanKunjunganAwalExport, 'laporan_kunjungan_pasien.xlsx');
46     }
47 }

```

Gambar 1. 1121 Source Code LaporanController

7. Pendaftarancontroller.php

1. Mengatur alur pendaftaran pasien dalam sistem klinik.
2. Lokasi file di **folder app > Http > Controllers > PendaftaranController.php**
3. *index()* :
 - a. Menampilkan seluruh data pendaftaran.
 - b. Ambil daftar dokter untuk ditampilkan bersama pendaftaran.
 - c. Menampilkan view: *admin.pendaftaran.index*.
4. *store(Request \$request)* :
 - a. Menyimpan data pendaftaran baru.
 - b. Validasi data *input*: *pasien_id*, *dokter_id*, status, tanggal_Registrasi, waktu_Registrasi, keluhan, spesialis, nomor_rekam_medis, nama pasien, tanggal_lahir, tempat_lahir, alamat, jenis_kelamin, telepon, dll.

- c. Membuat data pasien jika belum ada (berdasarkan NIK).
 - d. Membuat data pendaftaran dengan status *default*: Belum Kajian Awal.
 - e. *Redirect* kembali dengan pesan sukses.
5. *edit(\$id)* :
 - a. Menampilkan form *edit* untuk data pendaftaran dengan *ID* tertentu.
 - b. Mengirimkan data pendaftaran dan dokters ke *view*.
 6. *update(Request \$request, \$id)* :
 - a. Mengupdate data pendaftaran sesuai *ID*.
 - b. Validasi data serupa dengan *method store*.
 - c. *Update* data pendaftaran dan relasi pasien.
 7. *destroy(\$id)* :
 - a. Menghapus data pendaftaran berdasarkan *ID*.
 - b. *Redirect* kembali dengan pesan sukses.
 8. *cari(Request \$request)* :
 - a. Melakukan pencarian pasien berdasarkan nomor rekam medis (*nomor_rm*).
 - b. Digunakan dalam form dinamis untuk mengisi data pasien jika sudah terdaftar sebelumnya.

- Menampilkan form *edit* untuk data pendaftaran dengan *ID* tertentu.
- Mengirimkan data pendaftaran dan dokters ke *view*.

- 6.
- update(Request \$request, \$id) :*

- Mengupdate data pendaftaran sesuai ID.
- Validasi data serupa dengan *method store*.
- Update* data pendaftaran dan relasi pasien.

- b. Validasi data serupa dengan *method store*.
- c. *Update* data pendaftaran dan relasi pasien.

- c. *Update* data pendaftaran dan relasi pasien.

7. *destroy(\$id)* :

- Menghapus data pendaftaran berdasarkan *ID*.
- Redirect* kembali dengan pesan sukses.

- b. *Redirect* kembali dengan pesan sukses.

8. *cari(Request \$request) :*

- a. Melakukan pencarian pasien berdasarkan nomor rekam medis (nomor_rm).

- medis (nomor_rm).

- b. Digunakan dalam form dinamis untuk mengisi data pasien jika sudah terdaftar sebelumnya.

- pasien jika sudah terdaftar sebelumnya.

[illegible]

Gambar 1. 1153 Source Code PendaftaranController

5. *web.php*

1. *Source code* ini merupakan *file routing* utama dalam aplikasi *Laravel* yang mengatur semua jalur (*route*) akses halaman *frontend* dan *backend*. File ini berada di **folder routes > web.php**.
2. *Source code* ini juga menangani *route* untuk *AJAX* seperti :
 - a. *getAntrianTerkini*,
 - b. *getDokterBySpesialis*,
 - c. dan *cekPasienLama*.
5. *Source code* ini mengelompokkan *route* berdasarkan *role* pengguna, yaitu:
 - a. Admin :
 1. Akses *Dashboard* admin dan data *chart*.
 2. Manajemen data dokter (*DokterController*) dan jadwal dokter (*JadwalDokterController*).
 3. *CRUD* data pendaftaran pasien (*PendaftaranController*).
 4. Statistik dan ekspor data pendaftaran.
 5. Data antrian dan pengaturan umum.
 - b. Dokter :
 1. Akses *Dashboard* dokter dan laporan.
 2. Manajemen data pasien dan rekam medis via *KajianController*.
 3. Akses laporan serta ekspor laporan ke *Excel* dan *PDF* via *LaporanController*.

[illegible]

Gambar 1. 1185 Source Code Route Web

Lampiran 8 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	: EC002025089958, 15 Juli 2025
Pencipta	
Nama	: M. Roy Samsul Akbar, Dwi Intan Af'idah, S.T., M.Kom. dkk
Alamat	: Desa Grogol, Jalan Candi Moncol, No 02 RT 07/RW 03., Dukuhturi, Kab. Tegal, Jawa Tengah, 52192
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	: Aplikasi Pendaftaran Pasien Online Berbasis Website Di Klinik Pratama Aisyiyah Haji. Mafroh
Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia	: 15 Juli 2025, di Kota Tegal
Jangka waktu perlindungan	: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.
Nomor Pencatatan	: 000930219
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 keabsahannya dengan memindai kode QR pada dokumen ini dan informasi akan ditampilkan dalam browser.

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	M. Roy Samsul Akbar	Desa Grogol, Jalan Candi Moncol, No 02 RT 07/RW 03, Dukuhturi, Kab. Tegal
2	Dwi Intan Af'idah, S.T., M.Kom.	Jalan Grinting RT 003/RW 002, Kec. Bulakamba, Kab. Brebes 52253 Bulakamba, Kab. Brebes
3	Dairoh, S.Si., M.Sc.	Jalan Garuda No 03 RT 002/RW 009 Randugunting, Kec. Tegal Selatan, Kota Tegal 52131 Tegal Selatan, Kota Tegal

