

Penjelasan Integrasi Front End & Back End

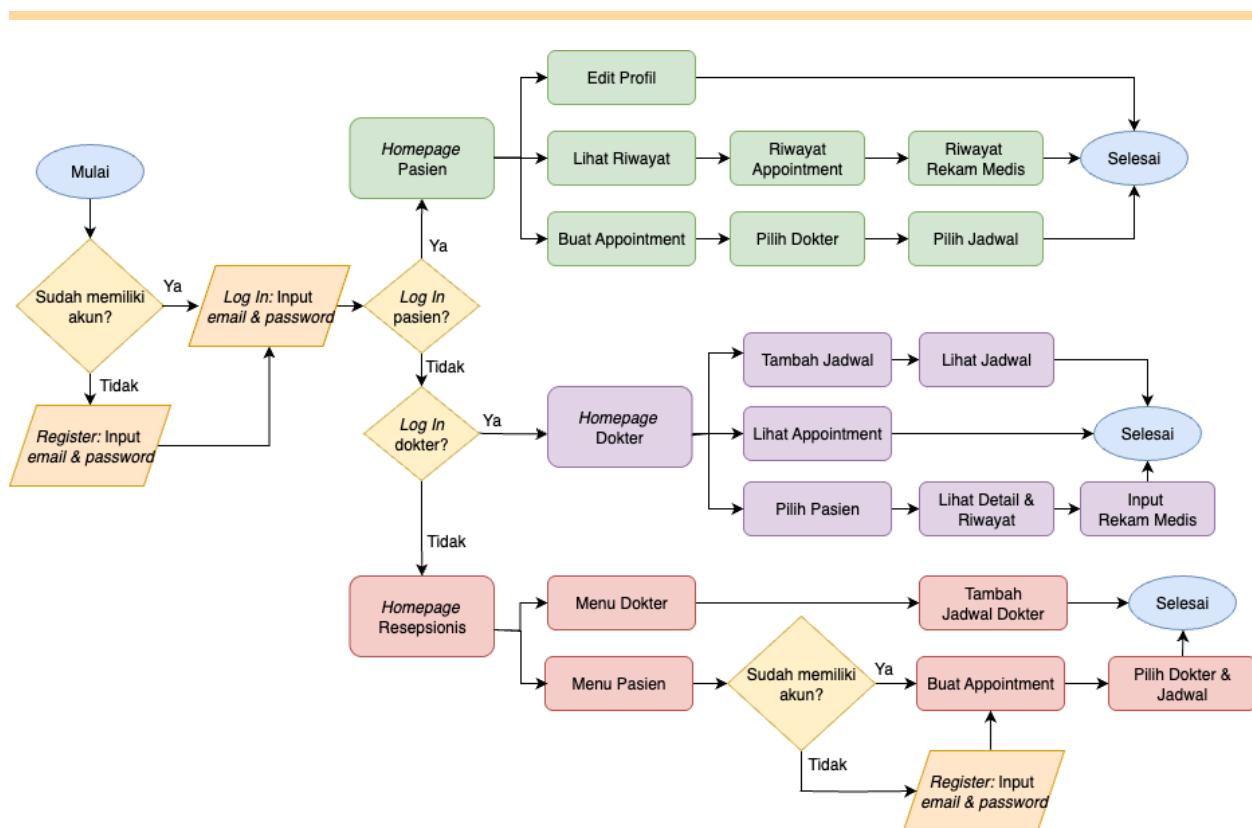
Week 4 - Application Integration

Kelas : KOM-A

Anggota : Aliya Khairun Nisa (24/543832/PA/23111)

Farsya Nabila Tori (24/543855/PA/23113)

Gracella Wiendy Koesnadi (24/541739/PA/22991)



Gambar 1. Diagram Alir (Flowchart) Penggunaan Website

Front End & Back End Connection

Mekanisme koneksi dan interaksi antara database Relasional (MySQL) dengan web frontend (HTML/Jinja) pada proyek ini diimplementasikan melalui framework Python Flask. Proses ini mengikuti arsitektur Model-View-Controller (MVC) atau varian serupa yang memisahkan logika backend dari presentasi frontend. Selain itu, database di-deploy ke

Railway agar semua perangkat yang mengakses aplikasi ini menggunakan database yang sama. Berikut adalah penjelasan rinci mengenai mekanisme koneksi dan interaksi data:

1) Lapisan Koneksi Database (Backend)

Koneksi database diatur sepenuhnya dalam file app.py yang bertindak sebagai server aplikasi (Controller dan Model).

- **Konfigurasi dan Inisialisasi Koneksi:** Aplikasi menggunakan library flask_mysqldb yang berfungsi sebagai adaptor antara Flask dan MySQL, serta pymysql sebagai driver koneksi.
 - Kredensial sensitif database (Host, User, Password, DB Name, Port) tidak ditulis langsung dalam kode, melainkan diambil dari variabel .env (menggunakan os.getenv dan dotenv).
 - Detail koneksi diatur dalam konfigurasi aplikasi Flask (app.config).
 - Objek MySQL(app) diinisialisasi sehingga dapat membuat koneksi database yang siap digunakan oleh seluruh aplikasi.

Python

```
# app.py: Inisialisasi dan Konfigurasi

import pymysql
pymysql.install_as_MySQLdb()
from flask_mysqldb import MySQL
from dotenv import load_dotenv
import os

load_dotenv()

# Mengambil kredensial dari environment variables (keamanan)
app.config['MYSQL_HOST'] = os.getenv('MYSQL_HOST')
app.config['MYSQL_USER'] = os.getenv('MYSQL_USER')
app.config['MYSQL_PASSWORD'] = os.getenv('MYSQL_PASSWORD')
app.config['MYSQL_DB'] = os.getenv('MYSQL_DB')
```

```
# Inisialisasi objek MySQL
mysql = MySQL(app)
```

- **Eksekusi Perintah SQL:** Interaksi data dilakukan menggunakan objek Cursor yang diperoleh dari objek koneksi:
 - Membuka Kursor: Untuk setiap operasi database (SELECT, INSERT, UPDATE, DELETE), kursor dibuka dengan "cur = mysql.connection.cursor()".
 - Eksekusi Query: Kursor digunakan untuk mengeksekusi perintah SQL melalui cur.execute("QUERY", (parameter,)). Semua input dari pengguna (seperti email atau password pada login atau data formulir) selalu diteruskan sebagai parameter terpisah (%s) untuk mencegah serangan SQL Injection.
 - Pengambilan Hasil:
 - cur.fetchone(): Digunakan untuk mengambil satu baris hasil (misalnya saat login atau mengambil detail profil).
 - cur.fetchall(): Digunakan untuk mengambil semua baris hasil (misalnya saat menampilkan daftar dokter atau riwayat janji temu).
 - Commit Perubahan: Untuk operasi modifikasi data (INSERT, UPDATE, DELETE), perubahan permanen dilakukan dengan mysql.connection.commit().
 - Menutup Kursor: Kursor selalu ditutup dengan cur.close() setelah operasi selesai untuk melepaskan sumber daya database.

2) Lapisan Interaksi Data (Frontend)

Interaksi data antara backend (Python) dan frontend (HTML) difasilitasi oleh Flask Routes dan template engine Jinja2.

- **Alur Data (Read Operation)**

- Permintaan Klien (Frontend): Pengguna mengakses URL tertentu (misalnya /profile atau /appointmentHistory).
- Pemrosesan Route (Backend): Fungsi route Flask terkait dieksekusi. Fungsi ini melakukan otentikasi sesi (if 'email' not in session:), menjalankan query SQL kompleks (misalnya join tabel Appointment, Pasien, Dokter, dan Rekam Medis) menggunakan kursor, dan mengambil hasil (history_data = cur.fetchall()).
- Rendering Template: Hasil data (berbentuk list atau dictionary) dilewatkan ke template HTML menggunakan render_template(). Contoh: return render_template('profile.html', user=user_info, phones=phone_list).
- Template Engine (Frontend): Jinja2 menerima variabel data dan mengulanginya (loop) untuk membuat HTML dinamis (misalnya {% for app in appointments %}).

- **Alur Data (Write/Update Operation)**

- Permintaan Klien (Frontend): Pengguna mengisi formulir HTML (misalnya /editProfile) dan menekan tombol submit (metode POST).
- Pengambilan Data Formulir (Backend): Fungsi route Flask menangkap data yang dikirim melalui request.form['nama_kolom'].
- Validasi/Pemrosesan: Data diproses (misalnya, hashing password saat registrasi atau validasi input telepon).
- Eksekusi Query SQL: Perintah INSERT atau UPDATE dieksekusi menggunakan kursor dan data yang diterima dari formulir.
- Pesan Umpam Balik: Aplikasi menggunakan flash() untuk menyimpan pesan status (sukses atau gagal) di sesi, yang kemudian ditampilkan di layout halaman berikutnya.
- Redirect: Pengguna diarahkan ulang ke halaman lain (misalnya return redirect(url_for('profile'))) untuk mencegah pengiriman formulir ganda.

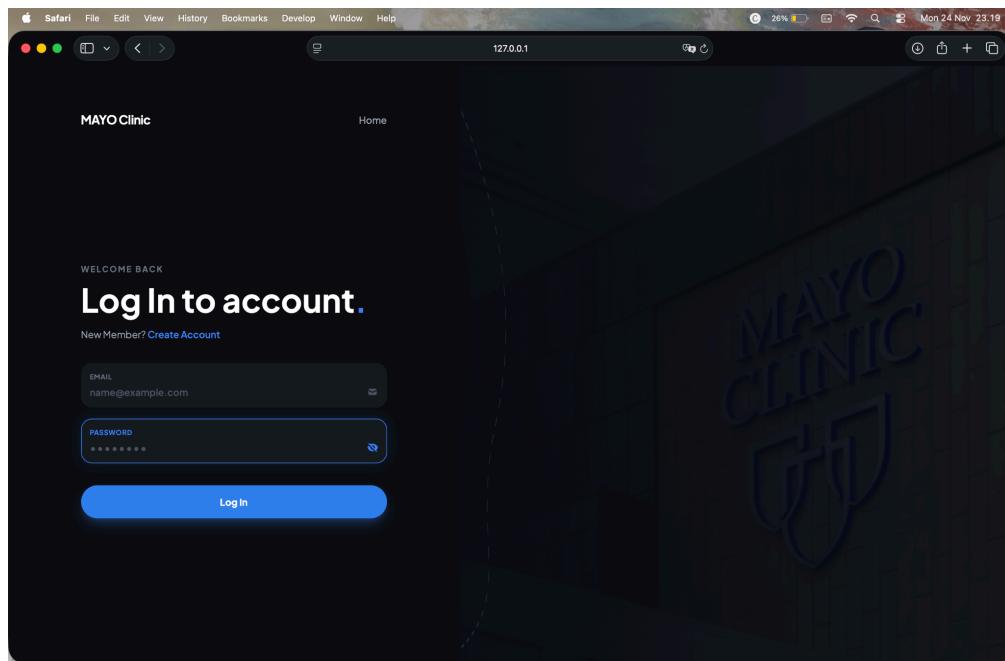
Secara keseluruhan, proyek ini menggunakan pendekatan server-side rendering yang memisahkan tanggung jawab database (MySQL), logika aplikasi (Flask/Python),

dan presentasi data (HTML/Jinja2), memastikan alur data yang aman dan terstruktur.

Snapshot Implementasi

UNIVERSAL

Page login ini akan diakses oleh pasien, dokter, dan resepsionis. Formulir login terdiri dari dua field input utama yaitu Email, di mana pengguna memasukkan alamat email mereka, yang berfungsi sebagai username untuk autentikasi sistem. Field kedua adalah Password, yang memiliki fitur tombol toggle mata di samping field yang memungkinkan pengguna menampilkan atau menyembunyikan karakter password (fungsi `togglePassword()`). Formulir ini mengirimkan data menggunakan metode POST ke route `/login` di backend, di mana server akan memproses autentikasi. Pesan sistem seperti notifikasi login berhasil atau pesan error "Invalid email or password", ditampilkan di atas formulir melalui mekanisme flashed messages Flask. Setelah mengisi kedua field, pengguna menekan tombol Log In berwarna biru untuk memproses permintaan masuk. Background page ini gradien dari warna ke gambar (mohon tingkatkan kecerahan layar agar dapat melihat gambar dengan jelas).

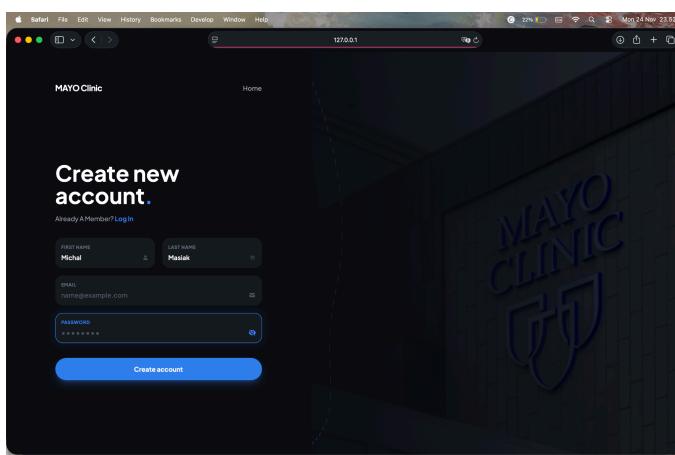


Gambar 1. Log In Dokter, Resepsionis, dan Pasien

PASIEN

- **Register**

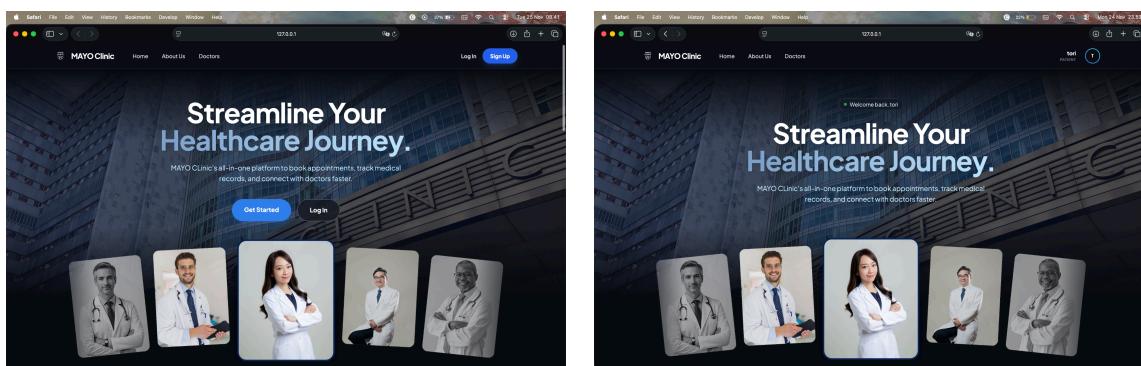
Pasien akan register di halaman ini sebelum membuat janji temu. Form register berisi email dan password yang memiliki fitur tombol toggle mata di samping field yang memungkinkan pengguna menampilkan atau menyembunyikan karakter password (fungsi `togglePassword()`). Password kemudian akan dihash sebelum dimasukkan ke database. Setelah itu, pasien akan dialihkan ke halaman login.



Gambar 2. Register akun Pasien

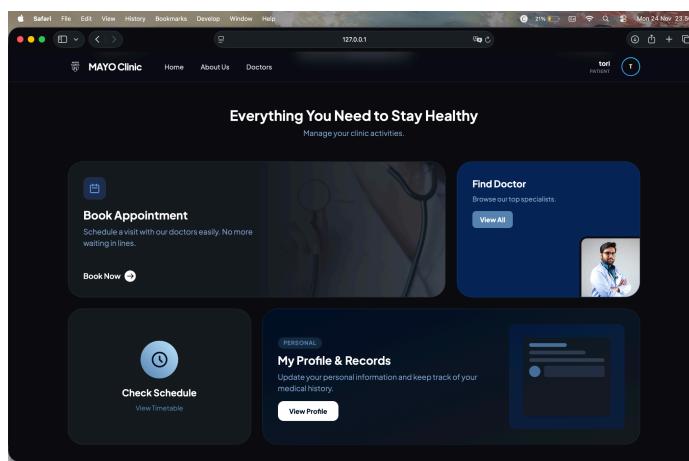
- **Homepage**

Berikut adalah homepage yang mempunyai 5 container, container pertama sebagai pembuka, container kedua berisi utilities/fitur yang ada pada website, container ketiga adalah penjelasan singkat mengenai klinik. Lalu container keempat berisi daftar dokter beserta fotonya dan container terakhir berisi informasi tentang kontak klinik.



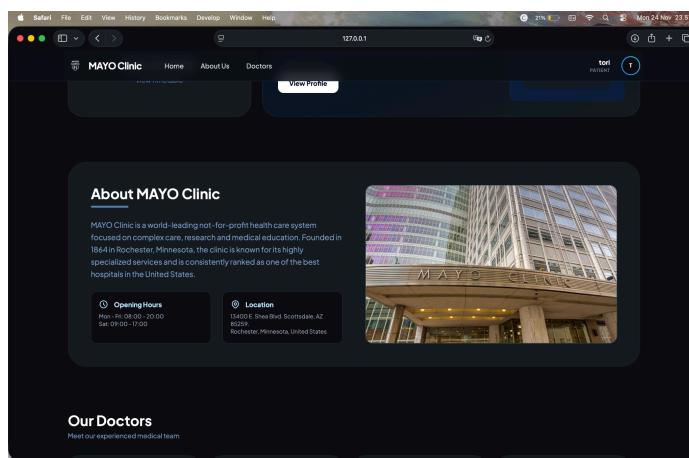
Gambar 3 dan 4. Homepage Pasien Container 1

Container kedua berisi 4 subcontainer. Subcontainer pertama berisi tombol untuk membuat janji temu yang jika diklik akan dialihkan ke bookAppointment.html. Subcontainer kedua berisi tombol untuk melihat semua daftar dokter yang jika diklik akan dialihkan ke ke container keempat. Lalu subcontainer ketiga mempunyai tombol untuk melihat jadwal dokter sedangkan subcontainer terakhir memiliki tombol view profile yang jika diklik, pasien akan dialihkan ke page profile.html



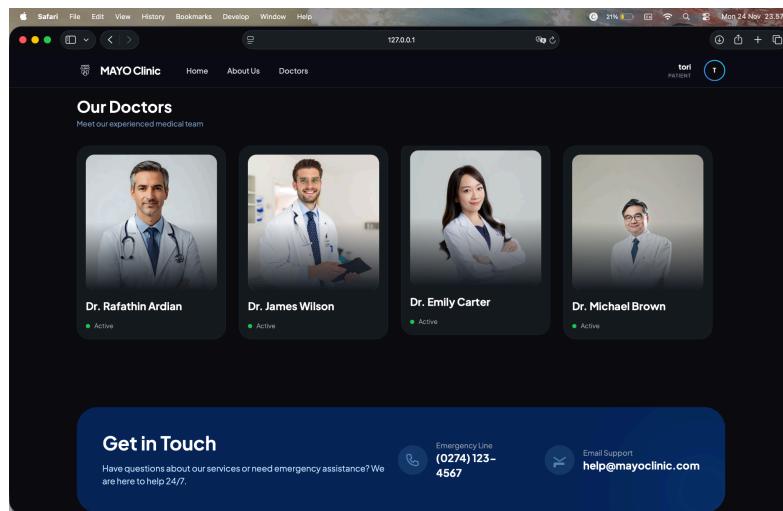
Gambar 5. Homepage Pasien Container 2

Berikut adalah container ketiga dari home.html yang berisi informasi singkat mengenai klinik. Informasi tersebut mencakup sejarah singkat, jam operasional, alamat, dan foto gedung.



Gambar 6. Homepage Pasien Container 3

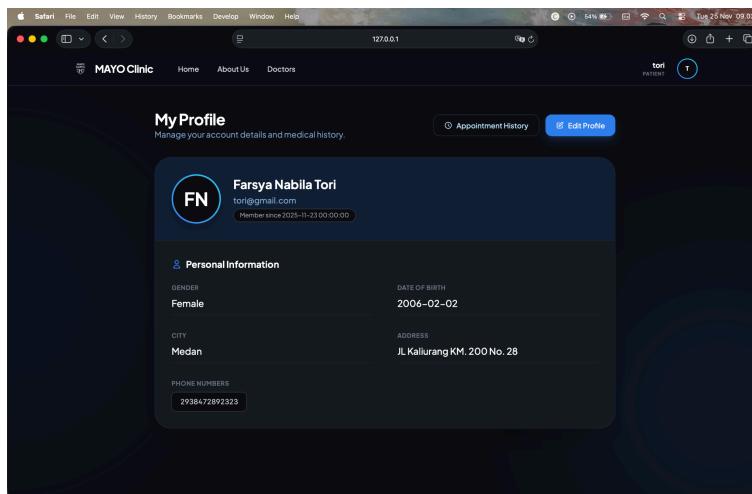
Container keempat berisi list dokter yang bertugas di klinik beserta fotonya. List ini bisa di swipe ke kanan agar dapat melihat semua dokter. Container terakhir yaitu container 5, berisi kontak emergency klinik yang dapat dihubungi serta email klinik jika pasien ingin memberi feedback.



Gambar 7. Homepage Pasien Container 4 dan 5

- **Profile**

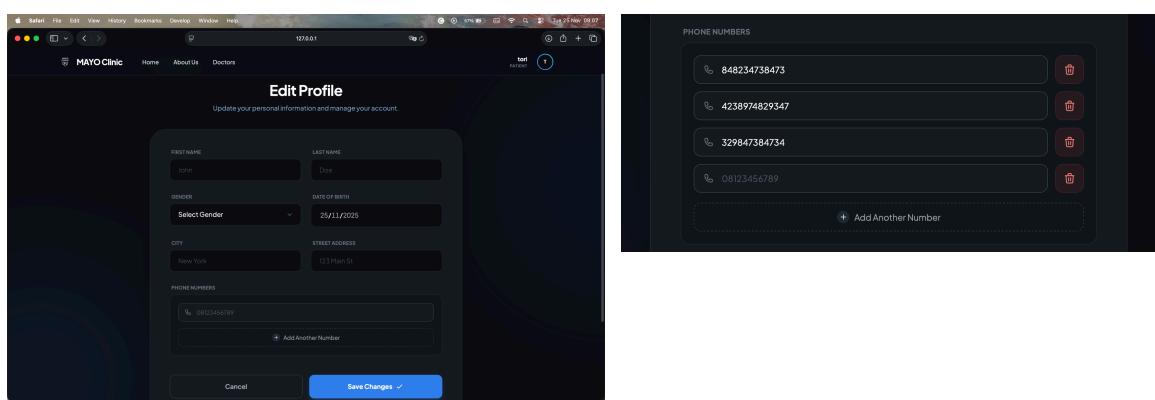
Halaman ini menyajikan data pribadi pasien. Terdapat profile picture dengan inisial yang dapat menyesuaikan dengan inisial 2 kata pertama dari nama pengguna. Dalam page ini terdapat 2 tombol, yaitu tombol appointment history yang jika diklik akan dialihkan ke appointmentHistory.html. Tombol kedua yaitu edit profile yang dapat mengalihkan ke editProfile.html.



Gambar 8. Profile Pasien

- **Edit Profile**

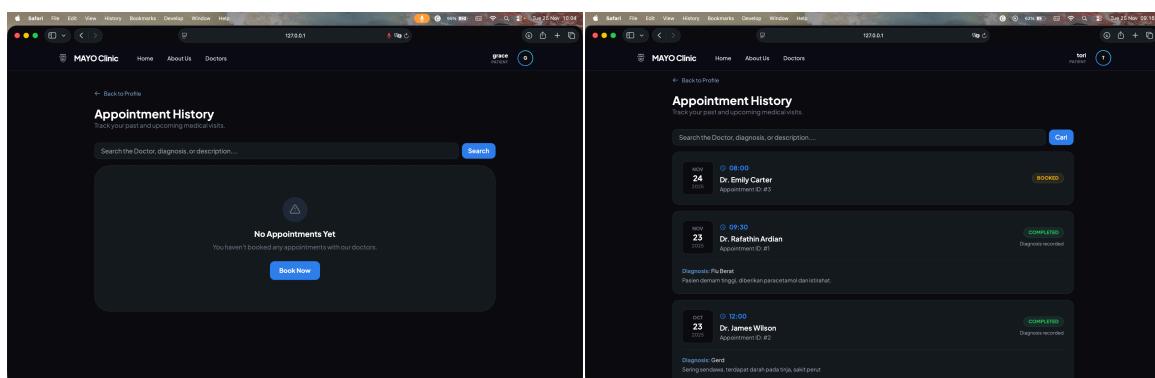
Halaman ini berbentuk form yang mempunyai 7 field yaitu first name, last name, gender, date of birth, city, street address, dan phone numbers. Field phone numbers dapat mempunyai banyak nilai. Selain itu, terdapat 2 tombol yaitu cancel yang berguna untuk kembali ke halaman profile, dan tombol save changes yang akan commit perubahan tersebut ke database yang ada di railway.



Gambar 9 dan 10. Edit Profile Pasien

- **Appointment History**

Halaman ini mempunyai fitur search yang dapat menghilangkan card appointment history jika tidak cocok dengan apa yang dicari. Card berisi informasi tanggal appointment, status, beserta rekam medis yang ada di bawahnya.

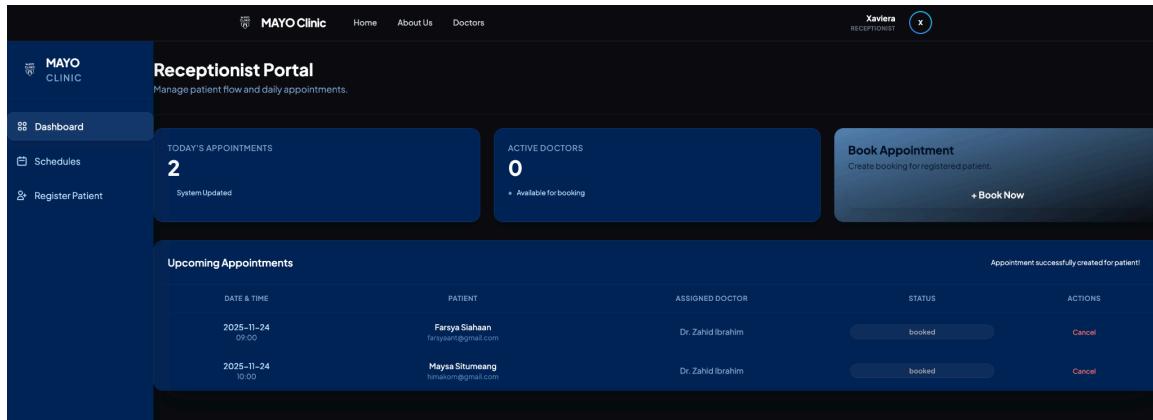


Gambar 11. Appointment History

RESEPSIONIS

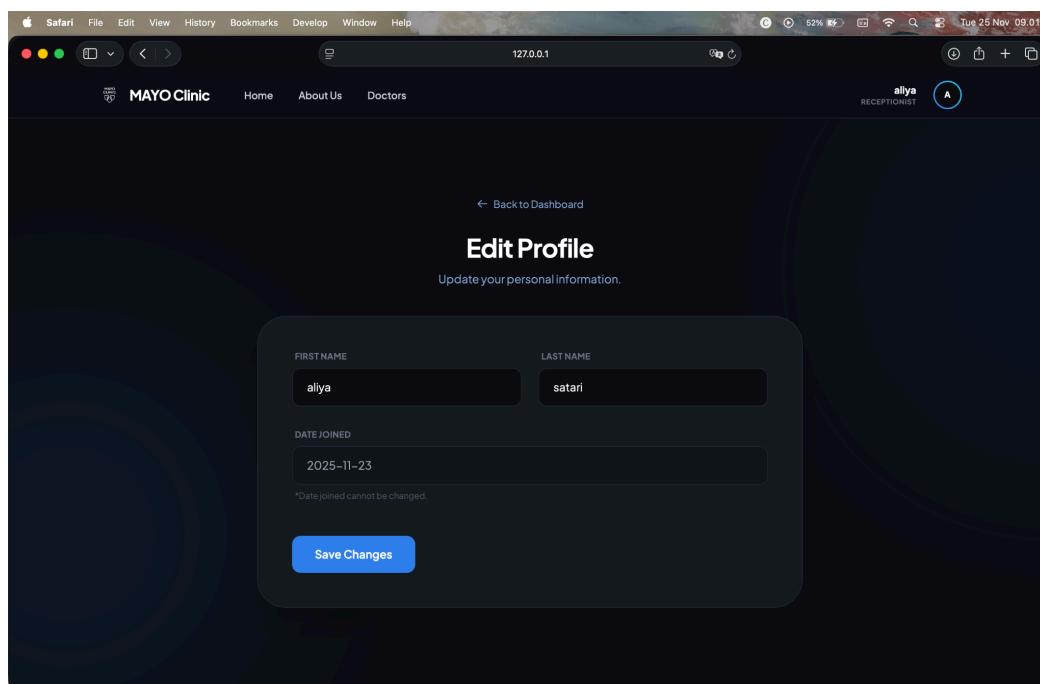
- **Dashboard**

Tampilan *dashboard* resepsionis menyediakan akses menu melalui *sidebar*, yaitu: *Dashboard*, *Schedules* (jadwal dokter), dan *Register Patient* (pendaftaran pasien).



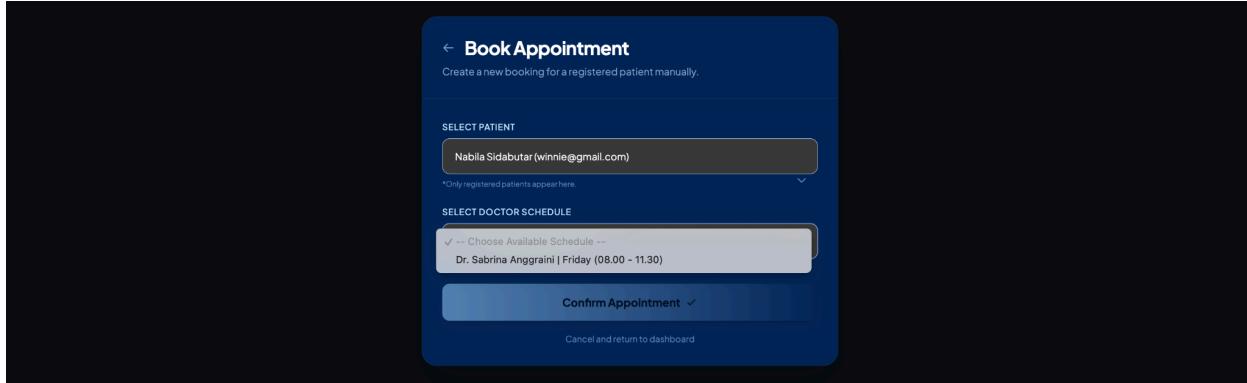
Gambar 12. *Dashboard* Resepsionis

Resepsionis dapat memperbarui nama akun melalui fitur *Edit Profile*. Perubahan data akan otomatis tersinkronisasi pada tampilan utama *dashboard*.



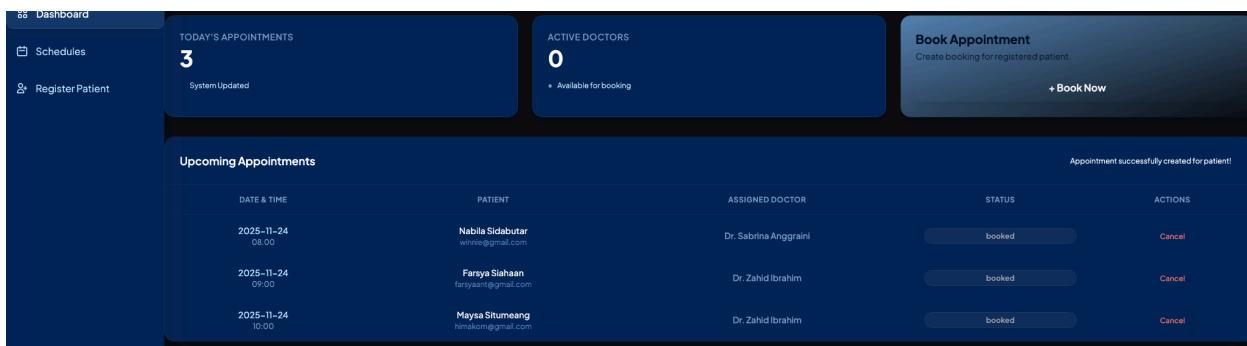
Gambar 13. Edit Profil Resepsionis

Fitur *Book Appointment* digunakan untuk mendaftarkan janji temu baru dengan memilih pasien terdaftar dan jadwal dokter yang tersedia.



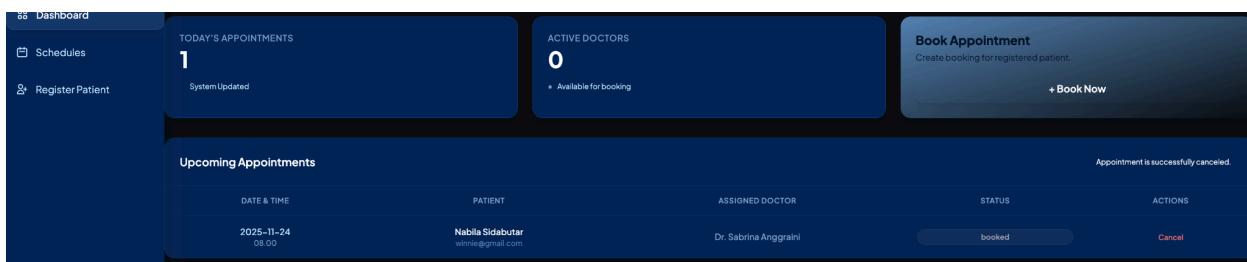
Gambar 14. *Book Appointment* Melalui Resepsionis

Data appointment yang berhasil ditambahkan akan langsung muncul pada tabel *Upcoming Appointments* di dashboard.



Gambar 15. *Appointment* Berhasil Ditambahkan

Resepsionis dapat membatalkan jadwal menggunakan tombol *Cancel*, yang secara otomatis menghapus data tersebut dari sistem. Sementara angka pada *Active Doctors* merepresentasikan jumlah dokter yang jadwalnya masih tersedia pada hari tersebut.



Gambar 16. *Appointment* Berhasil Dibatalkan

Dokter

- Dashboard

The screenshot shows the Dokter dashboard for Dr. Emily. On the left sidebar, there are links for Dashboard, My Schedule, Appointments, Profile, and Log Out. The main area displays a welcome message "Welcome, Dr. Emily." and today's statistics: 1 Patient and 0 Upcoming. A "YOUR SCHEDULE" section lists appointments for Selasa (10:00 - 01:30 WIB), Kamis (08:00 - 17:00 WIB), and Jumat (05:27 - 11:27 WIB). A "Today's Appointments" section shows a single appointment for Farsya Nabila Tori at 08:00 with ID 5, with a "Done" button. To the right is a calendar for November 2025, showing the days from Monday to Sunday. Notifications on the right side mention a meeting at 13:00 and an update on stock levels.

Gambar 17. Tampilan Dashboard Dokter

A modal window titled "Tambah Jadwal Baru" (Add New Schedule) is displayed over the dashboard. It contains fields for "Hari Praktek" (Practicing Day) with a dropdown menu showing "--Choose Day--", "Jam Mulai" (Start Time) with a time picker set to "--:--", and "Jam Selesai" (End Time) with a time picker also set to "--:--". At the bottom are "Cancel" and "Save" buttons. The background of the modal is semi-transparent, showing parts of the dashboard like the schedule and notifications.

Gambar 18. Fitur Tambah Jadwal

- Manage Schedule

Clinic App
General Practitioner

Dashboard

My Schedule

Appointments

Profile

Log Out

Manage Schedule

View and manage your weekly practice hours.

Active Schedule		
DAY	PRACTICE TIME	ACTION
- Selasa	⌚ 10:00 - 01:30	Delete
- Kamis	⌚ 08:00 - 17:00	Delete
- Jumat	⌚ 05:27 - 11:27	Delete
- Jumat	⌚ 10:00 - 02:00	Delete
- Sabtu	⌚ 06:30 - 14:30	Delete

Clinic App
General Practitioner

Dashboard

My Schedule

Appointments

Profile

Log Out

Manage Schedule

127.0.0.1:5000 says

Are you sure you want to delete the schedule for Jumat?

OK Cancel

Active Schedule		
DAY	PRACTICE TIME	ACTION
- Selasa	⌚ 10:00 - 01:30	Delete
- Kamis	⌚ 08:00 - 17:00	Delete
- Jumat	⌚ 05:27 - 11:27	Delete
- Jumat	⌚ 10:00 - 02:00	Delete
- Sabtu	⌚ 06:30 - 14:30	Delete

The screenshot shows the 'Clinic App General Practitioner' interface. On the left is a dark sidebar with icons for Dashboard, My Schedule, Appointments, Profile, and Log Out. The main area is titled 'Manage Schedule' with the sub-instruction 'View and manage your weekly practice hours.' Below this is a table titled 'Active Schedule'.

DAY	PRACTICE TIME	ACTION
- Selasa	⌚ 10:00 - 01:30	Delete
- Kamis	⌚ 08:00 - 17:00	Delete
- Jumat	⌚ 05:27 - 11:27	Delete
- Sabtu	⌚ 06:30 - 14:30	Delete

Gambar 19 - 21. Tampilan My Schedule

- **Appointment**

The screenshot shows the 'Clinic App General Practitioner' interface. On the left is a dark sidebar with icons for Dashboard, My Schedule, Appointments, Profile, and Log Out. The main area is titled 'Patient Queue List' with the sub-instruction 'Manage examination schedules and medical records for today.' It features a search bar for 'SELECT DATE' (27/11/2025), 'SEARCH PATIENT' (Farsya Nabila Tori), and 'FILTER STATUS' (All Status). Below this is a table titled 'Patient Queue List'.

TIME	PATIENT	STATUS	ACTION
08:00	F Farsya Nabila Tori ID-4	Selesai	History

Gambar 22. Tampilan Appointment

[Back to History](#)

MEDICAL RECORD VIEW

Medical Report
 Clinic App Record System

Examination Date
2025-11-27
08:00 WIB

PATIENT NAME Farsya Nabila Tori	GENDER Female	AGE 19 Years	STATUS Selesai
-------------------------------------------	------------------	-----------------	-------------------

DIAGNOSIS

Belum Diperiksa

CLINICAL NOTES & PRESCRIPTION

Silakan input rekam medis baru.

Generated electronically by Clinic App.
Confidential Medical Record.
Attending Physician

Gambar 23. Catatan Rekam Medis

● Profile

Clinic App
 General Practitioner

- [Dashboard](#)
- [My Schedule](#)
- [Appointments](#)
- [Profile](#)

[Log Out](#)

Profile Settings

Manage your account information and appearance.

dr. Emily
 General Practitioner

First Name
Emily
Last Name
Carter

Phone Number
0812345678908
Status
Active

Professional Bio (Visible to patients)
 Contoh: Berpengalaman 5 tahun dalam menangani...

Security

New Password
Leave blank to keep current
Confirm New Password

[Cancel](#) [Save Changes](#)

Gambar 24. Tampilan Untuk Edit Profile