



PRESENTASI TUGAS BESAR EL2008-PEMECAHAN MASALAH DENGAN PEMROGRAMAN

DEVEN ZIE MANUELA SITANGGANG (13223091)

MUHAMMAD LUTHFII ALGHAZALI (13223097)

SARAH BEATRIX SITORUS (13223100)

NAUFAL RIDHO WICAKSONO (13223102)

ABDUL HAKIM YAFI (13223107)





- Rumah Sakit Semoga Sehat Selalu memiliki sejumlah dokter yang harus dijadwalkan untuk bertugas selama 30 hari ke depan
- Setiap hari terbagi menjadi tiga shift—pagi, siang, dan malam—yang masing-masing wajib diisi minimal oleh satu dokter
- Proses penjadwalan ini perlu memperhatikan batas maksimal shift mingguan untuk tiap dokter serta preferensi waktu dan jenis shift yang diinginkan
- Penyusunan jadwal secara manual rawan kesalahan dan berisiko menimbulkan beban kerja berlebih maupun kekosongan jadwal
- Untuk itu, dikembangkanlah sebuah aplikasi yang mampu menyusun jadwal dokter secara otomatis, efisien, dan dengan tingkat kesalahan yang rendah

FITUR UTAMA

APLIKASI



- Manajemen Data Dokter
 - Tambah, hapus, dan tampilkan data dokter.
 - Data mencakup nama, bidang, tingkat, maksimal shift per minggu, preferensi shift, dan waktu (awal/akhir bulan).
 - Memudahkan pencarian dokter berdasarkan nama, bidang, dan tingkat.
 - Menampilkan daftar semua dokter beserta detail lengkapnya.
 - Dapat melakukan pembatalan aktivitas terakhir (penambahan atau penghapusan dokter).
 - Mencatat dan menampilkan riwayat aktivitas manajemen dokter.
 - Data disimpan dalam file data/daftar_dokter.csv.
- Penjadwalan Otomatis
 - Jadwal dibuat untuk 30 hari ke depan, dengan total 90 shift (3 shift/hari).
 - Setiap shift diisi secara otomatis sesuai kebutuhan jumlah dokter, batasan beban kerja, dan preferensi yang ditentukan dengan menggunakan algoritma greedy
 - Menyimpan jadwal kedalam format CSV data/jadwal_dokter.csv agar mudah diintegrasikan.
- Laporan dan Visualisasi Jadwal
 - Laporan mencakup:
 - Jadwal harian.
 - Distribusi shift per dokter.
 - Jumlah pelanggaran terhadap preferensi.
 - Laporan disimpan dalam format CSV data/laporan_dokter.csv
 - Memungkinkan pengguna untuk melihat total shift yang telah ditugaskan kepada dokter tertentu.
 - Menyediakan opsi untuk melihat jadwal dalam berbagai tampilan:
 - Jadwal Harian: Menampilkan detail shift untuk tanggal tertentu.
 - Jadwal Mingguan: Menampilkan detail shift untuk minggu tertentu.
 - Jadwal Bulanan: Menampilkan keseluruhan jadwal untuk 30 hari.
- Antarmuka Pengguna (GUI)
 - Menggunakan GTK+3 untuk tampilan grafis.
 - Navigasi antar menu utama, manajemen data dokter, penjadwalan, dan laporan.
 - Input dan output dilakukan melalui GUI sepenuhnya (tidak perlu terminal).
 - Menyajikan data dokter dan jadwal dalam format tabel (Tree View) yang rapi dan mudah dianalisis.

Masalah utama di dekomposisi menjadi tiga modul fungsional :

1. Pengelolaan Data Dokter

- Menambah data dokter
 - Menghapus data dokter
 - Mencari data dokter
 - Menampilkan data dokter
- Data dokter disimpan ke data_dokter.csv
- Data dokter terdiri dari

- Nama
- Bidang
- Tingkat
- Maksimal shift perminggu
- Preferensi shift
- Preferensi waktu

2. Penentuan Jadwal Otomatis

- Menyusun jadwal jaga untuk 30 hari (90 shift) dengan
 - Memastikan dua dokter per shift
 - Mematuhi batasan shift mingguan
 - Meminimalkan pelanggaran preferensi
- Jadwal disimpan ke jadwal_dokter.csv.

3. Penampilan Informasi Jadwal dan

- Menampilkan jadwal (harian, mingguan dan bulanan)
 - Menghitung jumlah shift per dokter
 - Mendeteksi pelanggaran preferensi (shift, waktu dan batasan shift)
- Laporan disimpan di laporan_dokter.csv

Struktur Data Fitur 1 : Pengelolaan Data Dokter

Struktur Data

- Struktur Dokter (Dokter) :
 - nama[LEN] : Nama dokter (string)
 - bidang[LEN] : Bidang spesialisasi dokter (string)
 - tingkat[LEN] : Tingkat dokter
 - preferensi_shift[LEN] : preferensi shift (Pagi/Siang/malam)
 - preferensi_waktu[LEN] : preferensi waktu (campur/awal bulan/akhir bulan)
 - next: pointer ke dokter berikutnya dalam linked list.
- Struktur Aktivitas
 - tipe: Jenis aksi (AKSI_TAMBAH/AKSI_HAPUS)
 - dokterData : pointer ke data dokter yang terkait dengan aktivitas
 - next : pointer ke aktivitas berikutnya dalam linked list

- Enum
 - AktiType : Menyimpan jenis aksi yang dilakukan
- Konstanta
 - LEN : Panjang maksimum string
 - OUTPUT_FILE_DOKTER : Nama file untuk menyimpan data dokter
 - AKSI_TAMBAH dan AKSI_HAPUS : Untuk menandai juenis aktivitas (penambahan atau penghapusan dokter).
- Variabel global

Struktur Data Fitur 1 : Pengelolaan Data Dokter

Fungsi Utama

- Menejemen Dokter
 - tambah_dokter_manual() : Menambahkan dokter baru
 - hapus_dokter() : Menghapus dokter berdasarkan nama
 - cari_dokter_nama(), cari_dokter_bidang(), cari_dokter_tingkat() : Fungsi pencarian dengan kriteria berbeda.
 - tampilan_semua() : Menampilkan semua data dokter
 - cari_dokter_menu() : Menampilkan submenu untuk memilih metode pencarian dokter:
 - tampilan_hasil(Dokter* hasil) : Menampilkan hasil pencarian dokter berdasarkan kriteria tertentu.
-

- Sistem Undo :
 - tambah_aktivitas() : mencatat aktivitas ke dalam log
 - hapus_aktivitas_terakhir() : Membatalkan operasi terakhir (undo)
 - tampilan_log : menampilkan log aktivitas
- Menejemen file
 - load_data_dari_csv(): Memuat data dari file CSV
 - save_data_to_csv(): Menyimpan data ke file CSV
 - free_memory(): Membersihkan semua alokasi memori
- Fungsi Pembantu
 - salin_dokter(): Membuat salinan data dokter
 - clean_newline(): Membersihkan karakter newline dari string
 - contains_keyword(): Pencarian string case-insensitive

Struktur Data Fitur 2 : Penentuan Jadwal Otomatis

Struktur Data

- Struktur Dokter (Doctor)
 - name[MAX_NAME_LEN] : nama dokter (string)
 - bidang[MAX_NAME_LEN] : bidang spesialisasi dokter.
 - tingkat : tingkat dokter(Enum TingkatDokter)
 - max_shift_per_weeks : batas maksimal shift per minggu (integer)
 - preffered_shift : Shift yang diingikan (Enum ShiftType)
 - preffered_time: preferensi waktu yang diinginkan (campur/awal bulan/akhir bulan) (Enum PreferensiWaktu)
 - total_shifts_assigned : Jumlah total shift yang sudah diambil dokter (integer)
 - weekly_shifts : jumlah shift perminggu (array integer)
 - weekly_shifts : Array global yntuk menyimpan semua dokter
- Struktur Jadwal (Schedule)
 - day : hari dalam bulan (0 hingga 29) (integer)
 - shift : jenis shift (integer)
 - num_doctors : jumlah dokter yang dijadwalkan pada shift tersebut (integer)
 - doctor_ids : ID dokter yang dijadwalkan (array of integer)
 - penyimpanan : Array global untuk menyimpan semua jadwal (array schedule)
- Enum
 - ShiftType : SHIFT_PAGI, SHIFT_SIANG, SHIFT_MALAM
 - TingkatDokter : TINGKAT_KOASS, TINGKAT_RESIDEN, TINGKAT_SPESIALIS, TINGKAT_KONSULEN

Struktur Data Fitur 2 : Penentuan Jadwal Otomatis

- PreferensiWaktu : WAKTU_AWAL_BULAN,
WAKTU_AKHIR_BULAN,
WAKTU_CAMPUR
- Konstanta
 - MAX_DOCTOR : -
 - DAYS_IN_MONTH : 30
 - SHIFTS_PER_DAY : 3
 - DOCTORS_PER_SHIFT : 2
 - MAX_DOCTOR_PER_SHIFT : 2
 - LINE_BUFFER_SIZE : -
 - OUTPUT_FILE_PENJADWALAN : -
- Array Pendukung
 - hari : nama hari (array string)
 - shift_names : nama shift (array string)
 - tingkat_str : nama tingkat dokter (array string)

Struktur Data Fitur 3 : Penampilan Informasi Jadwal dan Laporan

Struktur Data

- Struktur ShiftEntry

- tanggal : informasi tanggal (integer)
- Hari[20] : nama hari (array of character)
- Shift[10] : nama shift (array of character)
- Nama[MAX_NAME] : nama dokter (array of character)
- Bidang[100] : Bidang dokter (array of character)
- Tingkat[20] : Informasi Tingkatan dokter (array of character)

- Struktur DoctorPref

- nama[MAX_NAME] : nama dokter (array of character)
- Bidang[100] : informasi bidang dokter (array of character)
- Tingkat[20] : informasi tingkatan dokter (array of character)

- max_shift_per_minggu : batas maksimal shift per minggu (integer)
- preferensi_shift[10] : preferensi shift tiap dokter (pagi, siang, malam) (array of character)
- preferensi_waktu[20] : Informasi preferensi waktu tiap dokter (Campur/Awal bulan/Ahir Bulan)

Konstanta

- MAX_ENTRIES : 1000
- MAX_NAME : 100
- MAX_LINE : 1024
- INPUT_FILE_JADWAL : “data/jadwal_dokter.csv”
- INPUT_FILE_DOKTER : “data/daftar_dokter.csv”
- INPUT_FILE_LAPORAN: “data/laporan_dokter.csv”
- OUTPUT_FILE_LAPORAN: “data/laporan_dokter.csv”

Logika Program Utama

Program utama menggunakan tampilan dengan konsep struktur Tree dengan submenu sebagai cabang.

- Fungsi main dari program utama diawali dengan **membaca file data_dokter.csv**
- Program **memberikan menu** berisi pilihan **subprogram** yang bisa digunakan pengguna. Pengguna memilih dengan klik menu yang ada di window.
- Setelah pengguna memilih, program akan **masuk ke submenu** berikutnya.
- Fungsi **menu_manajemen_dokter()**, **menu_lihat_jadwal()**, dan **menu_laporan_kinerja()** masing-masing **memberikan menu baru** untuk subprogram dengan nama yang sama. Ketika fungsi dalam tiap subprogram dipilih oleh pengguna, program akan langsung **mengeksekusi dan menampilkan** output dari fungsi yang dipilih.
- **menu_penjadwalan_otomatis()** berfungsi untuk menjalankan algoritma untuk **membuat jadwal** berdasarkan data dokter yang tersimpan di data_dokter.csv
- Ketika pengguna memilih menu untuk **keluar dari program**, sistem akan **menyimpan** semua perubahan pada file data_dokter.csv dan **menutupnya**. Lalu semua **memori** akan dibebaskan dan program pun selesai.

Logika Fitur 1 : Pengolahan Data Dokter

Alur Utama Pengolahan Data Dokter

1. Inisialisasi awal :
 - Variabel global head dan aktivitas_head diinisialisasi sebagai NULL.
2. Muat data dari CSV (load_data_dari_csv) saat program mulai.
3. Menampilkan menu utama :
 - Tambah dokter
 - Hapus dokter
 - Cari dokter
 - Tampilkan semua
 - Log aktivitas
 - Undo aktivitas terakhir
 - Simpan & keluar
4. Loop menu hingga user memilih keluar .
5. Simpan data ke CSV sebelum keluar.

Fungsi Utama

1. Tambah Dokter (tambah_dokter_manual)
 - Algoritma :
 - Alokasi memori untuk dokter baru.
 - Validasi input tiap field (tidak boleh kosong).
 - Masukkan data satu per satu melalui fgets/scanf.

- Sambungkan dokter baru ke awal linked list ($d->next = head$, $head = d$).
- Buat salinan dokter untuk dimasukkan ke log aktivitas ($salin_dokter \rightarrow tambah_aktivitas$).
- Error Handling :
 - Memastikan tidak ada input kosong.
 - Menghindari alokasi memori gagal.
- 1. Hapus Dokter (hapus_dokter)
 - Algoritma
 - Minta nama dokter yang ingin dihapus.
 - Cari dokter berdasarkan nama menggunakan traversal linked list.
 - Konfirmasi penghapusan dengan y/n.
 - Jika ditemukan, hapus dari linked list:
 - Perbarui pointer $prev->next = curr->next$.
 - Salin data dokter sebelum dihapus untuk disimpan di aktivitas ($salin_dokter \rightarrow tambah_aktivitas$).
 - Membebaskan memori.
 - Error Handling :
 - Nama dokter tidak ditemukan.
 - Input konfirmasi tidak valid.

Logika Fitur 1 : Pengolahan Data Dokter

3. Cari Dokter

- Ada 3 Jenis Pencarian :
 - Berdasarkan nama (cari_dokter_nama)
 - Berdasarkan bidang (cari_dokter_bidang)
 - Berdasarkan tingkat (cari_dokter_tingkat)
- Algoritma
 - Masukkan keyword pencarian.
 - Lakukan traversal pada linked list.
 - Gunakan fungsi contains_keyword untuk pencocokan case-insensitive.
 - Setiap cocok, dokter dicopy dan ditambahkan ke list hasil.
 - Tampilkan hasil menggunakan tampilkan_hasil.
 - Membebaskan semua hasil dari memori setelah ditampilkan
- Fungsi Pendukung
 - contains_keyword: Bandingkan dua string secara case-insensitive.
 - tampilkan_hasil: Menampilkan hasil pencarian.

4. Tampilkan semua dokter (tampilkan_semua)

- Algoritma
 - Jika aktivitas_head == NULL, tampilkan bahwa tidak ada aktivitas.
 - Ambil node aktivitas teratas (aktivitas_head).
 - Lihat tipe aktivitas:
 - Jika AKSI_TAMBAH: cari dokter berdasarkan nama + bidang, lalu hapus dari linked list.
 - Jika AKSI_HAPUS: kembalikan dokter ke linked list ($d \rightarrow next = head$, $head = d$). Bebaskan node aktivitas tersebut.
 - Fitur ini bersifat stack (LIFO), hanya bisa undo aktivitas terakhir.

5. Log Aktivitas (tampilkan_log)

- Algoritma
 - traversal linked list aktivitas.
 - Tampilkan setiap aksi (tambah/hapus) beserta nama dokter.
 - Urutan menurun (terbaru di atas).

Logika Fitur 1 : Pengolahan Data Dokter

6. Load & Save Data ke CSV

- Load Data (load_data_dari_csv)

- Algoritma
 - Buka file CSV.
 - Jika file tidak ditemukan, buat direktori dan file baru dengan header.
 - Baca baris demi baris:
 - Abaikan baris kosong.
 - Parsing tiap field dengan strtok.
 - Simpan ke dalam Dokter*.
 - Tambahkan ke linked list (head).
 - Tampilkan jumlah dokter berhasil dimuat.
 - Error Handling
 - File tidak dapat dibuka
 - Format CSV tidak sesuai

- Save Data (save_data_to_csv)

- Algoritma
 - Buka file CSV dalam mode tulis ("w").
 - Jika file/direktori tidak ada, coba buat direktori dan file baru.
 - Tulis header CSV.
 - Traversal linked list head, tulis setiap dokter ke file.

7. Pembebasan Memori (free_memory)

- Algoritma

- Hapus seluruh node di head (daftar dokter).
- Hapus seluruh node di aktivitas_head:
 - Bebaskan dokterData terlebih dahulu.
 - Lalu bebaskan node aktivitas itu sendiri

8. Fungsi Pendukung Tambahan

- salin_dokter(Dokter*) : Membuat salinan dokter untuk penyimpanan di aktivitas
- clean_newline(char*) : Membersihkan karakter newline dari input fgets
- contains_keyword(const char*, const char*) : Mencari apakah sebuah string mengandung keyword (case-insensitive)

Logika Fitur 2: Penentuan Jadwal Jaga Otomatis

Pendekatan Algoritma

- Algoritma Greedy: Memilih dokter terbaik untuk setiap shift berdasarkan skor preferensi dan batasan.
- Tujuan: Mengisi 90 shift ($30 \text{ hari} \times 3 \text{ shift/hari}$) dengan 2 dokter per shift.
- Mempertimbangkan:
 - Preferensi shift (Pagi, Siang, Malam).
 - Preferensi waktu (Awal/Akhir Bulan).
 - Batas shift mingguan per dokter.
 - Tingkat dokter (Koass, Residen, Spesialis, Konsulen).

Alur Utama Penjadwalan

1. Memuat Data Dokter (load_doctors_from_csv):
 - Membaca file CSV: nama, bidang, tingkat, max shift/minggu, preferensi shift, waktu.
 - Parsing ke array doctors menggunakan strtok.
 - Konversi string ke enum (ShiftType, TingkatDokter, PreferensiWaktu).

2. Menghasilkan Jadwal (generate_schedule):
 - Looping $30 \text{ hari} \times 3 \text{ shift}$ (Pagi, Siang, Malam).
 - Per shift:
 - Hitung skor dokter (calculate_score):
 - Preferensi shift: +15 (sesuai), -5 (tidak sesuai).
 - Preferensi waktu: +10 (sesuai), -5 (tidak sesuai).
 - Sisa shift mingguan: +10 per shift tersisa, -20 jika over.
 - Total shift: Prioritas dokter dengan shift sedikit.
 - Tingkat dokter: +3 per tingkat.
 - Pilih dokter dengan skor tertinggi, belum dijadwalkan.
 - Jika batas shift habis, pilih dokter terbaik (izinkan over shift).
 - Perbarui statistik dokter (total_shifts_assigned, weekly_shifts) via assign_shift.
3. Menyimpan Jadwal (save_schedule):
 - Tulis jadwal ke CSV (OUTPUT_FILE_PENJADWALAN).
 - Format: Tanggal, Hari, Shift, Nama Dokter, Bidang, Tingkat.
 - Konversi indeks ke string (misal, 0 → "Senin", 0 → "Pagi").

Logika Fitur 2: Penentuan Jadwal Jaga Otomatis

Fungsi Pendukung

- `get_week_number`: Hitung minggu ($\text{hari} \div 7$).
- `is_early_month`: Cek $\text{hari} \leq 15$ (awal bulan).
- `is_doctor_already_assigned`: Pastikan dokter belum dijadwalkan di shift sama.
- `parse_shift`, `parse_tingkat`, `parse_waktu`: Konversi string CSV ke enum.

Kelebihan Pendekatan

- Efisiensi: Algoritma greedy cepat menghasilkan jadwal.
- Pemerataan Beban: Skor memprioritaskan dokter dengan shift lebih sedikit.
- Fleksibilitas: Menangani pelanggaran batas shift jika dokter terbatas.
- Otomatisasi: Mengurangi kesalahan manual dan hemat waktu.
- Struktur Modular: Fungsi terpisah untuk pengelolaan data, penjadwalan, dan output.

Kekurangan Pendekatan

- Greedy Tidak Optimal: Mungkin tidak menghasilkan jadwal terbaik secara global.
- Pelanggaran Batas Shift: Mengizinkan over shift dapat menyebabkan ketidakadilan.
- Validasi Terbatas: Tidak ada pengecekan jumlah dokter cukup untuk 90 shift.
- Ketergantungan CSV: Format file input harus konsisten, rentan error jika salah.
- Kapasitas Statis: Array `doctors` dan `schedule` memiliki batas maksimum.

Logika Fitur 3: Menampilkan Informasi Jadwal

Pendekatan Algoritma

- Algoritma BruteForce: Program membaca data dari file dan memprosesnya dengan cara melakukan pencarian linear (linear search), perulangan (for dan while), serta pemrosesan baris per baris untuk menghitung jumlah shift dan pelanggaran. Semua pengecekan seperti pencocokan nama, preferensi shift, dan waktu dilakukan secara langsung dan eksplisit dalam loop.
- Tujuan: Menulis laporan_dokter.csv dan menampilkan jumlah shift per dokter, menampilkan pelanggaran, menampilkan total pelanggaran dokter tertentu, dan menampilkan semua total pelanggaran dari tiap dokter.
- Mempertimbangkan:
 - Preferensi shift (Pagi, Siang, Malam).
 - Preferensi waktu (Awal/Ahir Bulan).
 - Batas shift mingguan per dokter.

Alur Utama Penjadwalan

- Memuat Data jadwal(load_shift_data()):
 - Membaca file CSV: jadwal_dokter.csv
 - Parsing ke array shift_data
- Memuat Data dokter(load_doctor_data()):
 - Membaca file CSV : daftar_dokter.csv
- Mengolah data (proses_data()):
- Menampilkan informasi berikut berdasarkan input user :
 1. Menampilkan Jumlah shift dokter(lihat_jumlah_shift())
 2. Menampilkan pelanggaran (lihat_pelanggaran)
 3. Menampilkan jumlah pelanggaran dokter tertentu(lihat_pelanggaran_dokter())
 4. Menampilkan jumlah pelanggaran dari semua dokter (lihat_jumlah_pelanggaran_per_dokter())

Logika Fitur 3: Menampilkan Informasi Jadwal

Pendekatan Algoritma

- Algoritma BruteForce: Program melakukan perulangan while dan membaca baris per baris dari jadwal_dokter.csv yang kemudian akan dibandingkan dengan tanggal atau minggu sesuai dengan input user. Setiap tanggal atau minggu yang sesuai dengan input user akan ditambahkan ke array baru yang akan ditampilkan ke output ke user
- Tujuan : Menampilkan jadwal shift dokter harian, mingguan dan bulanan sesuai input user
- Mempertimbangkan :
 - Tanggal Shift Dokter (1-30)

Alur Utama Penjadwalan

- Memuat data jadwal
 - Membaca file CSV : jadwal_dokter.csv
 - Parsing ke array nama_dokter, bidang, tingkat, hari dan sift
- Menampilkan jadwal shift dokter berdasarkan input user
 - Menampilkan jadwal harian(jalankan_Opsi1)
 - Menampilkan jadwal mingguan(jalankan_Opsi2)
 - Menampilkan jadwal bulanan(jalankan_Opsi3)

CONTOH STUDI KASUS



Kondisi awal

- Pengguna adalah seorang pegawai tata usaha Rumah Sakit Semoga Sehat Selalu. Ia bertugas sebagai manajer penjadwalan shift jaga dokter di rumah sakit tersebut.
- Program sudah di-compile dan sudah siap dipakai.
- Sudah ada data_dokter.csv berisikan daftar dokter yang diisi sebelumnya.
- Belum ada file jadwal_dokter.csv dan file laporan_dokter.csv.

Alur studi kasus

1. Pengguna ingin membuka menu manajemen dokter untuk menambahkan dokter baru:
 - Nama : Mazaya Bari Nafiah
 - Bidang : Bedah Toraks
 - Tingkat : Konsulen
 - Maks. shift per minggu : 6
 - Preferensi shift : Siang
 - Preferensi waktu : Awal bulan
1. Pengguna kemudian ingin melihat semua data dokter.
2. Lalu, untuk memastikan, pengguna ingin melihat apakah Mazaya sudah berhasil ditambahkan atau belum.
3. Ketika melihat data semua dokter, pengguna ingin menghapus dokter bernama Rafif Farras.
4. Setelah itu, pengguna lupa perubahan apa saja yang telah Ia lakukan. Sehingga pengguna ingin melihat log aktivitas.

CONTOH STUDI KASUS

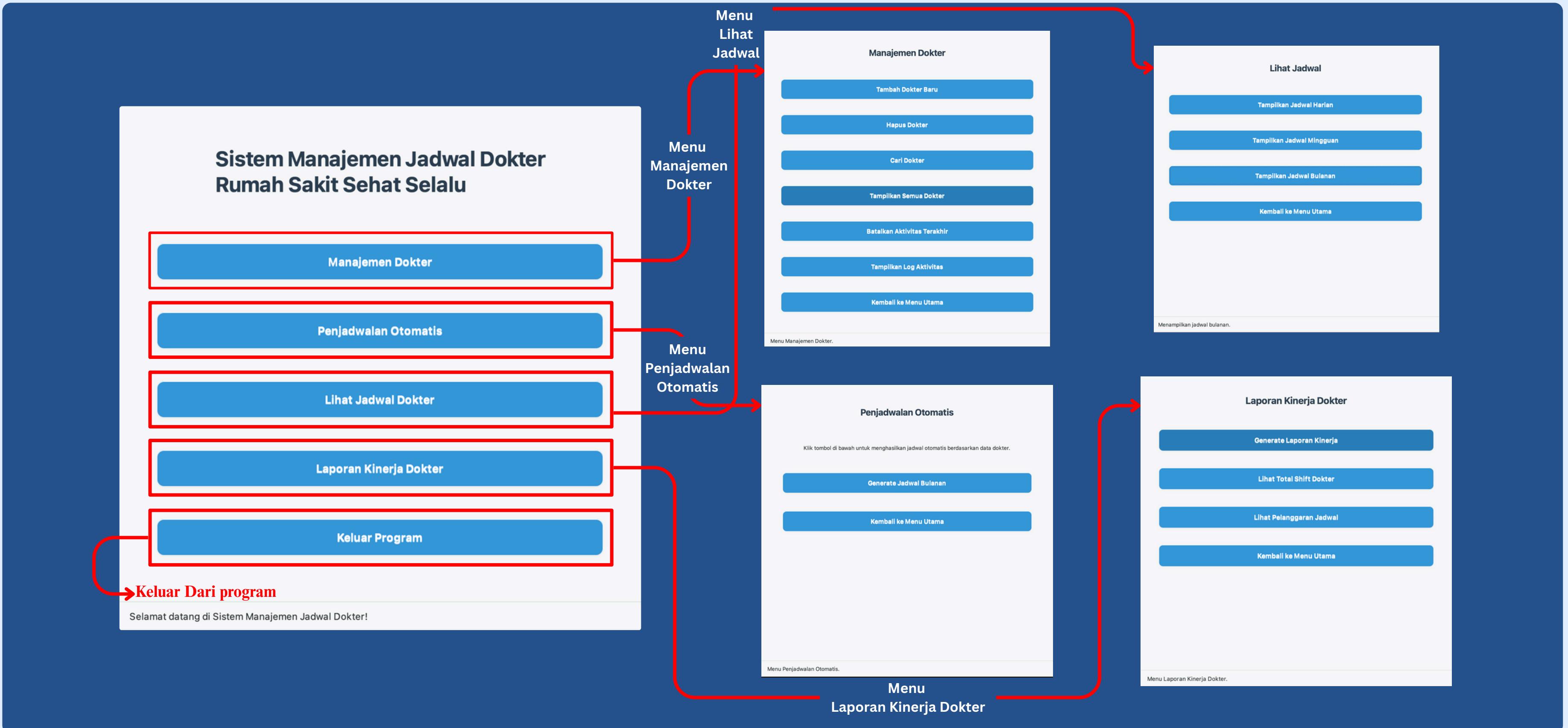


Alur Studi kasus

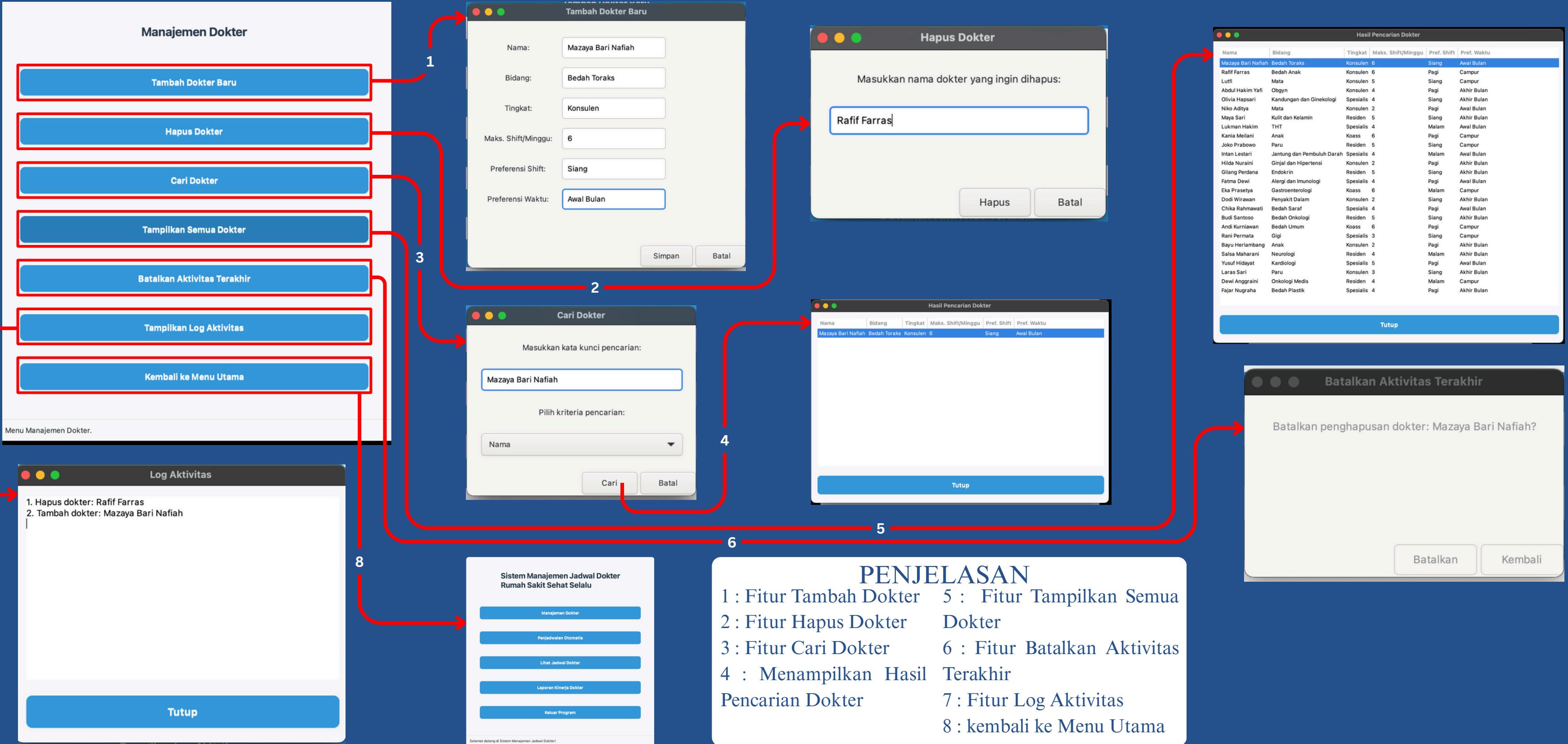
5. Setelah itu, pengguna lupa perubahan apa saja yang telah Ia lakukan. Sehingga pengguna ingin melihat log aktivitas.
6. Pengguna teringat bahwa Rafif Farras belum seharusnya dikeluarkan dari daftar dokter di RS tersebut di bulan ini. Pengguna kemudian ingin membatalkan aksi sebelumnya.
7. Setelah mengatur data dokter di RS tersebut, Pengguna ingin membuat jadwal secara otomatis dengan mempertimbangkan preferensi para dokter.
8. Setelah jadwal berhasil dibuat, Pengguna ingin melihat jadwal yang telah dibuat. Ia ingin melihat jadwal bulanan, minggu ke-2, dan hari ke-4.
9. Setelah berhasil membuat jadwal, Pengguna ingin memeriksa seberapa sesuai jadwal yang dibuat dengan preferensi para dokter.

10. Setelah laporan berhasil dibuat, Pengguna ingin melihat total shift jaga yang didapatkan oleh dokter bernama Abdul Hakim Yafi.
11. Kemudian, pengguna ingin melihat jumlah pelanggaran preferensi yang dialami semua dokter dan juga yang dialami dokter Abdul Hakim Yafi.
12. Terakhir, Pengguna ingin melihat detail pelanggaran apa saja yang terjadi terhadap preferensi para dokter.
13. Setelah itu, Pengguna ingin mengetahui apa yang terjadi jika memasukkan input yang salah ke dalam program. Pengguna pun mencoba memasukkan nama dokter, tanggal, dan minggu yang tidak ada pada data.
14. Setelah melakukan semua hal di atas, Pengguna telah menyelesaikan tugasnya untuk hari itu. Studi kasus pun selesai.

CONTOH STUDI KASUS



CONTOH STUDI KASUS



CONTOH STUDI KASUS



Penjadwalan Otomatis

Klik tombol di bawah untuk menghasilkan jadwal otomatis berdasarkan data dokter.

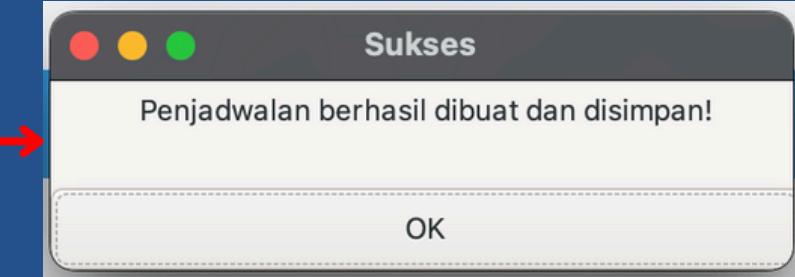
Generate Jadwal Bulanan

Kembali ke Menu Utama

Menu Penjadwalan Otomatis.

Fitur Penjadwalan Otomatis

Kembali Ke
Menu Utama



**Sistem Manajemen Jadwal Dokter
Rumah Sakit Sehat Selalu**

Manajemen Dokter

Penjadwalan Otomatis

Lihat Jadwal Dokter

Laporan Kinerja Dokter

Keluar Program

Selamat datang di Sistem Manajemen Jadwal Dokter!

CONTOH STUDI KASUS



Lihat Jadwal

Tampilkan Jadwal Harian

Tampilkan Jadwal Mingguan

Tampilkan Jadwal Bulanan

Kembali ke Menu Utama

Menampilkan jadwal bulanan.

Fitur Mencari Jadwal Harian

Fitur Mencari Jadwal Mingguan

Fitur Mencari Jadwal Bulanan

Pilih Tanggal

Masukkan tanggal (1-30):

Tampilkan Batal

Pilih Minggu

Masukkan minggu (1-5):

Tampilkan Batal

Tanggal	Hari	Nama	Spesialisasi	Tingkat	Shift
1	Senin	Rafif Farris	Bedah Anak	Konsulen	Pagi
1	Senin	Yusuf Hidayat	Kardiolog	Spesialis	Pagi
1	Senin	Mazaya Bari Nafiah	Bedah Toraks	Konsulen	Siang
1	Senin	Lutfi	Mata	Konsulen	Siang
1	Senin	Eka Prasetya	Gastroenterologi	Koass	Malam
1	Senin	Lukman Hakim	THT	Spesialis	Malam
2	Selasa	Kania Mellani	Anak	Koass	Pagi
2	Selasa	Andi Kurniawan	Bedah Umum	Koass	Pagi
2	Selasa	Mazaya Bari Nafiah	Bedah Toraks	Konsulen	Siang
2	Selasa	Joko Prabowo	Paru	Residen	Siang
2	Selasa	Intan Lestari	Jantung dan Pembuluh Darah	Spesialis	Malam
2	Selasa	Eka Prasetya	Gastroenterologi	Koass	Malam
3	Rabu	Rafif Farris	Bedah Anak	Konsulen	Pagi
3	Rabu	Fatma Dewi	Alergi dan Imunologi	Spesialis	Pagi
3	Rabu	Mazaya Bari Nafiah	Bedah Toraks	Konsulen	Siang
3	Rabu	Lutfi	Mata	Konsulen	Siang
3	Rabu	Dewi Anggraini	Onkologi Medis	Residen	Malam
3	Rabu	Lukman Hakim	THT	Spesialis	Malam
4	Kamis	Chika Rahmawati	Bedah Saraf	Spesialis	Pagi
4	Kamis	Yusuf Hidayat	Kardiolog	Spesialis	Pagi
4	Kamis	Maya Sari	Kulit dan Kelamin	Residen	Siang
4	Kamis	Gilang Perdana	Endokrin	Residen	Siang
4	Kamis	Intan Lestari	Jantung dan Pembuluh Darah	Spesialis	Malam
4	Kamis	Eka Prasetya	Gastroenterologi	Koass	Malam
5	Jumat	Kania Mellani	Anak	Koass	Pagi

Kembali ke
Menu Utama

Menampilkan
Jadwal
Harian

Menampilkan
Jadwal
Mingguan

Tanggal	Hari	Nama	Spesialisasi	Tingkat	Shift
4	Kamis	Chika Rahmawati	Bedah Saraf	Spesialis	Pagi
4	Kamis	Yusuf Hidayat	Kardiolog	Spesialis	Pagi
4	Kamis	Maya Sari	Kulit dan Kelamin	Residen	Siang
4	Kamis	Gilang Perdana	Endokrin	Residen	Siang
4	Kamis	Intan Lestari	Jantung dan Pembuluh Darah	Spesialis	Malam
4	Kamis	Eka Prasetya	Gastroenterologi	Koass	Malam

Tutup

Tanggal	Hari	Nama	Spesialisasi	Tingkat	Shift
8	Senin	Rafif Farris	Bedah Anak	Konsulen	Pagi
8	Senin	Yusuf Hidayat	Kardiolog	Spesialis	Pagi
8	Senin	Mazaya Bari Nafiah	Bedah Toraks	Konsulen	Siang
8	Senin	Lutfi	Mata	Konsulen	Siang
8	Senin	Eka Prasetya	Gastroenterologi	Koass	Malam
8	Senin	Lukman Hakim	THT	Spesialis	Malam
9	Selasa	Kania Mellani	Anak	Koass	Pagi
9	Selasa	Andi Kurniawan	Bedah Umum	Koass	Pagi
9	Selasa	Mazaya Bari Nafiah	Bedah Toraks	Konsulen	Siang
9	Selasa	Joko Prabowo	Paru	Residen	Siang
9	Selasa	Intan Lestari	Jantung dan Pembuluh Darah	Spesialis	Malam
9	Selasa	Eka Prasetya	Gastroenterologi	Koass	Malam
10	Rabu	Rafif Farris	Bedah Anak	Konsulen	Pagi
10	Rabu	Fatma Dewi	Alergi dan Imunologi	Spesialis	Pagi
10	Rabu	Mazaya Bari Nafiah	Bedah Toraks	Konsulen	Siang
10	Rabu	Lutfi	Mata	Konsulen	Siang
10	Rabu	Dewi Anggraini	Onkologi Medis	Residen	Malam
10	Rabu	Lukman Hakim	THT	Spesialis	Malam
11	Kamis	Chika Rahmawati	Bedah Saraf	Spesialis	Pagi
11	Kamis	Yusuf Hidayat	Kardiolog	Spesialis	Pagi
11	Kamis	Maya Sari	Kulit dan Kelamin	Residen	Siang
11	Kamis	Gilang Perdana	Endokrin	Residen	Siang
11	Kamis	Intan Lestari	Jantung dan Pembuluh Darah	Spesialis	Malam

Tutup

Sistem Manajemen Jadwal Dokter
Rumah Sakit Sehat Selalu

Manajemen Dokter

Penjadwalan Otomatis

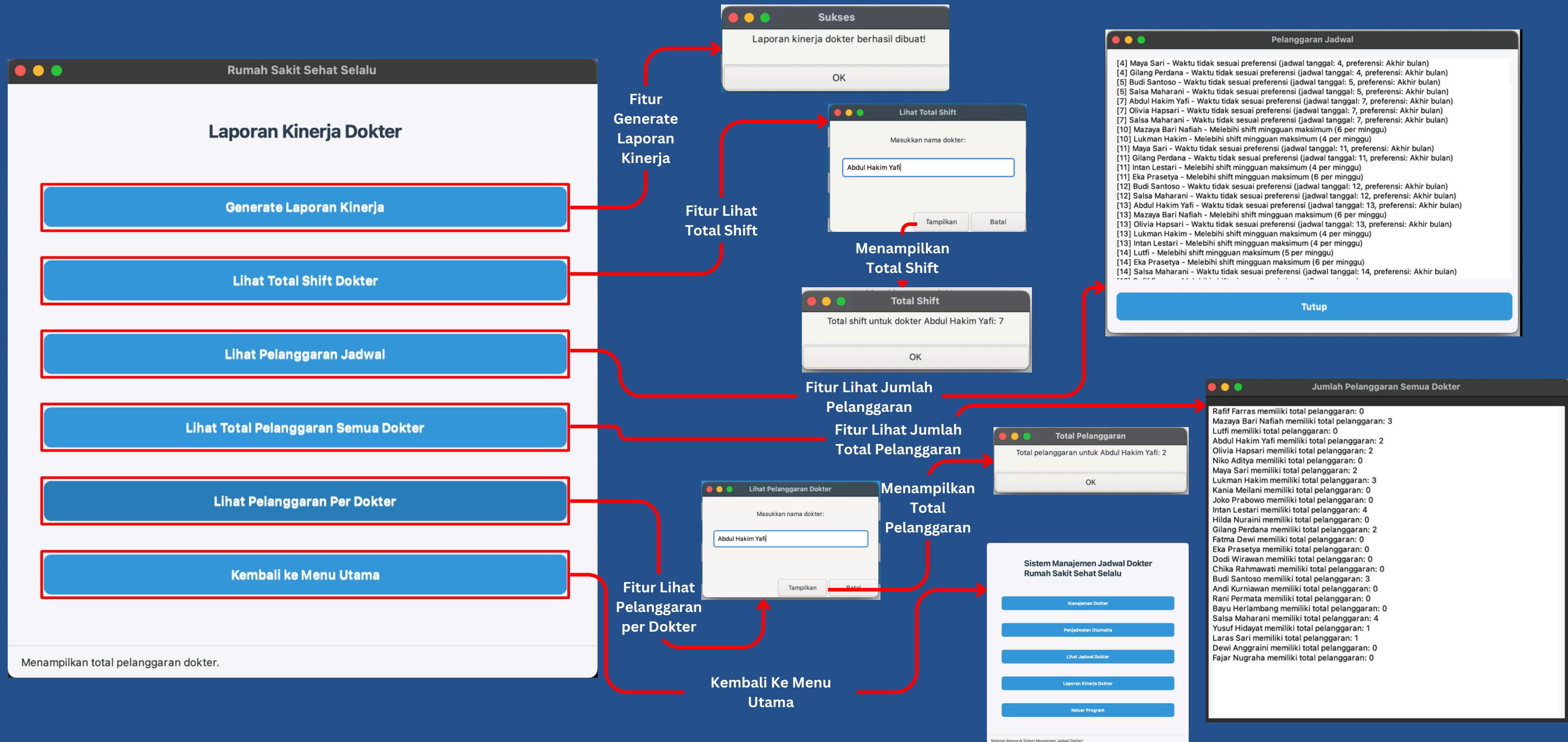
Lihat Jadwal Dokter

Laporan Kinerja Dokter

Keluar Program

Selamat datang di Sistem Manajemen Jadwal Dokter!

CONTOH STUDI KASUS





Analisis Hasil

- Menambahkan dokter
- Menampilkan data dokter
- Mencari data dokter
- Menghapus data dokter
- log aktivitas
- Fitur Undo
- Generate jadwal
- Melihat jadwal harian, mingguan, bulanan
- Generate laporan kinerja jadwal
- Melihat total shift dokter
- Melihat jumlah pelanggaran dokter
- Melihat detail pelanggaran dokter

BERHASIL DILAKUKAN!

Justifikasi pendekatan

• Antarmuka

Antarmuka **modular** dengan konsep **Tree** yang bercabang ketika menu dari subprogram dipilih membuat penggunaan yang **intuitif dan clean**.

• Format CSV

Format CSV yang digunakan sudah memenuhi dan lebih lengkap dari pada spesifikasi yang diminta.

Kolom **tambahan** pada **data diri** dokter menambah informasi **kompetensi** dokter sehingga pengguna dapat lebih mengenal dokter yang berjaga. Parameter **preferensi tambahan** (preferensi tanggal) menambah akurasi program dalam memberikan jadwal yang memastikan **performa** dokter juga **optimal**.

• Performa jadwal dan algoritma

Performa jadwal dokter yang di-generate algoritma sudah **cukup baik**. Algoritma **greedy** yang digunakan sudah membuat program lebih **optimal** dan tetap **cepat**. Meski dalam kasus tertentu tidak memberikan **global best**, namun, program ini sudah memberikan performa **optimal** untuk local best.

ANALISIS HASIL



Output yang Dihasilkan

```
D: > Luthfii > School 2.0 > Sem 4 > PMP > transit > TUBES-PMP > data > daftar_dokter.csv > data
1  nama,bidang,tingkat,max_shift_per_minggu,preferensi_shift,preferensi_waktu
2  Mazaya Bari Nafiah,Bedah Toraks,Konsulen,6,Siang,Awal Bulan
3  Fajar Nugraha,Bedah Plastik,Spesialis,4,Pagi,Akhir Bulan
4  Dewi Anggraini,Onkologi Medis,Residen,4,Malam,Campur
5  Laras Sari,Paru,Konsulen,3,Siang,Akhir Bulan
6  Yusuf Hidayat,Kardiologi,Spesialis,5,Pagi,Awal Bulan
7  Salsa Maharani,Neurologi,Residen,4,Malam,Akhir Bulan
8  Bayu Herlambang,Anak,Konsulen,2,Pagi,Akhir Bulan
9  Rani Permata,Gigi,Spesialis,3,Siang,Campur
10 Andi Kurniawan,Bedah Umum,Koass,6,Pagi,Campur
11 Budi Santoso,Bedah Onkologi,Residen,5,Siang,Akhir Bulan
12 Chika Rahmawati,Bedah Saraf,Spesialis,4,Pagi,Awal Bulan
13 Dodi Wirawan,Penyakit Dalam,Konsulen,2,Siang,Akhir Bulan
14 Eka Prasetya,Gastroenterologi,Koass,6,Malam,Campur
15 Fatma Dewi,Alergi dan Imunologi,Spesialis,4,Pagi,Awal Bulan
16 Gilang Perdana,Endokrin,Residen,5,Siang,Akhir Bulan
17 Hilda Nuraini,Ginjal dan Hipertensi,Konsulen,2,Pagi,Akhir Bulan
18 Intan Lestari,Jantung dan Pembuluh Darah,Spesialis,4,Malam,Awal Bulan
19 Joko Prabowo,Paru,Residen,5,Siang,Campur
20 Kania Meilani,Anak,Koass,6,Pagi,Campur
21 Lukman Hakim,THT,Spesialis,4,Malam,Awal Bulan
22 Maya Sari,Kulit dan Kelamin,Residen,5,Siang,Akhir Bulan
23 Niko Aditya,Mata,Konsulen,2,Pagi,Awal Bulan
24 Olivia Hapsari,Kandungan dan Ginekologi,Spesialis,4,Siang,Akhir Bulan
25 Abdul Hakim Yafi,Obgyn,Konsulen,4,Pagi,Akhir Bulan
26 Lutfi,Mata,Konsulen,5,Siang,Campur
27 Rafif Farras,Bedah Anak,Konsulen,6,Pagi,Campur
28
```

```
1  Tanggal,Hari,Shift,Nama_Dokter,Bidang,Tingkat
2  1,Senin,Pagi,Rafif Farras,Bedah Anak,Konsulen
3  1,Senin,Pagi,Yusuf Hidayat,Kardiologi,Spesialis
4  1,Senin,Siang,Mazaya Bari Nafiah,Bedah Toraks,Konsulen
5  1,Senin,Siang,Lutfi,Mata,Konsulen
6  1,Senin,Malam,Eka Prasetya,Gastroenterologi,Koass
7  1,Senin,Malam,Lukman Hakim,THT,Spesialis
8  2,Selasa,Pagi,Kania Meilani,Anak,Koass
9  2,Selasa,Pagi,Andi Kurniawan,Bedah Umum,Koass
10 2,Selasa,Siang,Mazaya Bari Nafiah,Bedah Toraks,Konsulen
11 2,Selasa,Siang,Joko Prabowo,Paru,Residen
12 2,Selasa,Malam,Intan Lestari,Jantung dan Pembuluh Darah,Spesialis
13 2,Selasa,Malam,Eka Prasetya,Gastroenterologi,Koass
14 3,Rabu,Pagi,Rafif Farras,Bedah Anak,Konsulen
15 3,Rabu,Pagi,Fatma Dewi,Alergi dan Imunologi,Spesialis
16 3,Rabu,Siang,Mazaya Bari Nafiah,Bedah Toraks,Konsulen
17 3,Rabu,Siang,Lutfi,Mata,Konsulen
18 3,Rabu,Malam,Dewi Anggraini,Onkologi Medis,Residen
19 3,Rabu,Malam,Lukman Hakim,THT,Spesialis
20 4,Kamis,Pagi,Chika Rahmawati,Bedah Saraf,Spesialis
21 4,Kamis,Pagi,Yusuf Hidayat,Kardiologi,Spesialis
22 4,Kamis,Siang,Maya Sari,Kulit dan Kelamin,Residen
23 4,Kamis,Siang,Gilang Perdana,Endokrin,Residen
24 4,Kamis,Malam,Intan Lestari,Jantung dan Pembuluh Darah,Spesialis
25 4,Kamis,Malam,Eka Prasetya,Gastroenterologi,Koass
26 5,Jumat,Pagi,Kania Meilani,Anak,Koass
27 5,Jumat,Pagi,Andi Kurniawan,Bedah Umum,Koass
28 5,Jumat,Siang,Budi Santoso,Bedah Onkologi,Residen
29 5,Jumat,Siang,Joko Prabowo,Paru,Residen
30 5,Jumat,Malam,Salsa Maharani,Neurologi,Residen
31 5,Jumat,Malam,Dewi Anggraini,Onkologi Medis,Residen
32 6,Sabtu,Pagi,Rafif Farras,Bedah Anak,Konsulen
33 6,Sabtu,Pagi,Fatma Dewi,Alergi dan Imunologi,Spesialis
```

Output yang Dihasilkan

Kesimpulan

1. Perancangan sistem pengelolaan data dokter dengan fitur tambah, hapus, tampil, dan baca data dari file eksternal (CSV) telah **berhasil dilakukan**.
2. Pengembangan algoritma penjadwalan otomatis dengan mempertimbangkan batasan kerja maksimal dan preferensi dokter telah **berhasil dibuat** dengan optimal menggunakan **algoritma greedy**.
3. Fitur visualisasi jadwal dan pelaporan jadwal shift dokter telah **berhasil dibuat** dengan visualisasi **berbagai format** dan **pelaporan informasi** pelanggaran preferensi dan distribusi beban kerja.
4. Sistem modular **berhasil dibuat** dengan antarmuka yang bisa digunakan melalui terminal powershell maupun dengan **GUI berbasis GTK**.

Saran pengembangan

1. **Penyesuaian format** file CSV dengan kebutuhan
2. **Eksplorasi pendekatan** lainnya dalam algoritma penjadwalan otomatis. Salah satu pendekatan yang dapat dicoba adalah backtracking, pendekatan matematis untuk kasus spesifik, dan pendekatan prioritas.
3. **Tuning pembobotan skor** algoritma penugasan dokter pada shift. Tuning ini harus disesuaikan dengan kebutuhan rumah sakit saat itu.
4. **Tampilan jadwal** dokter menggunakan GUI dalam **bentuk kalender**. Bentuk tampilan seperti ini lebih intuitif dan mudah dicerna lebih banyak kalangan.



TERIMA KASIH