

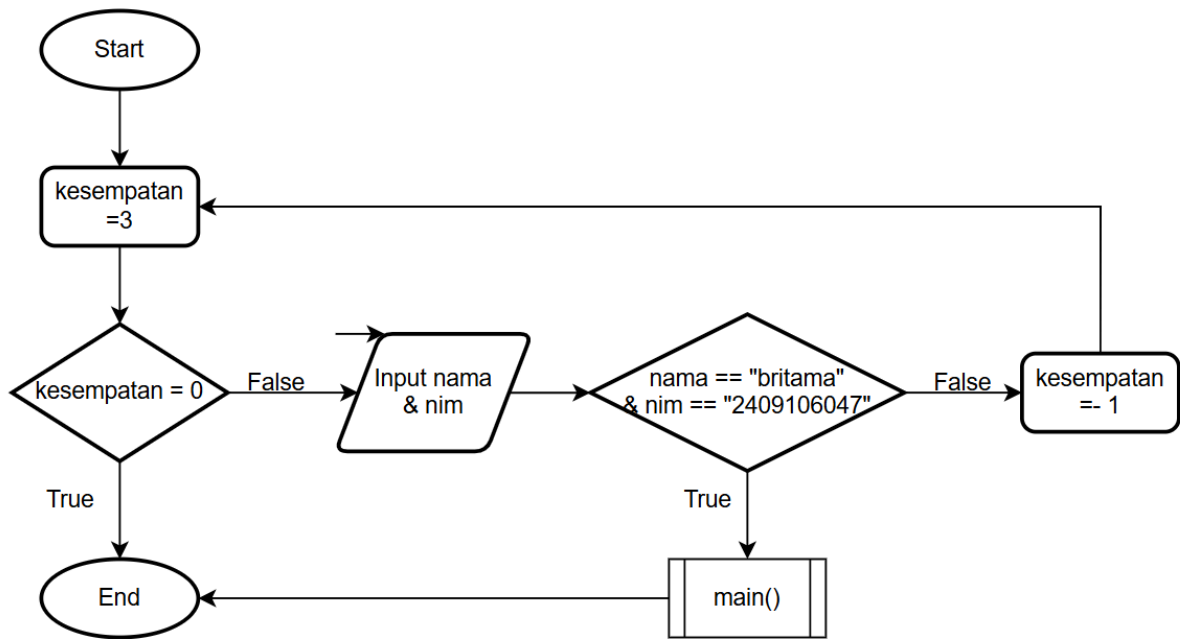
LAPORAN PRAKTIKUM
POSTTEST 2
ALGORITMA PEMROGRAMAN LANJUT



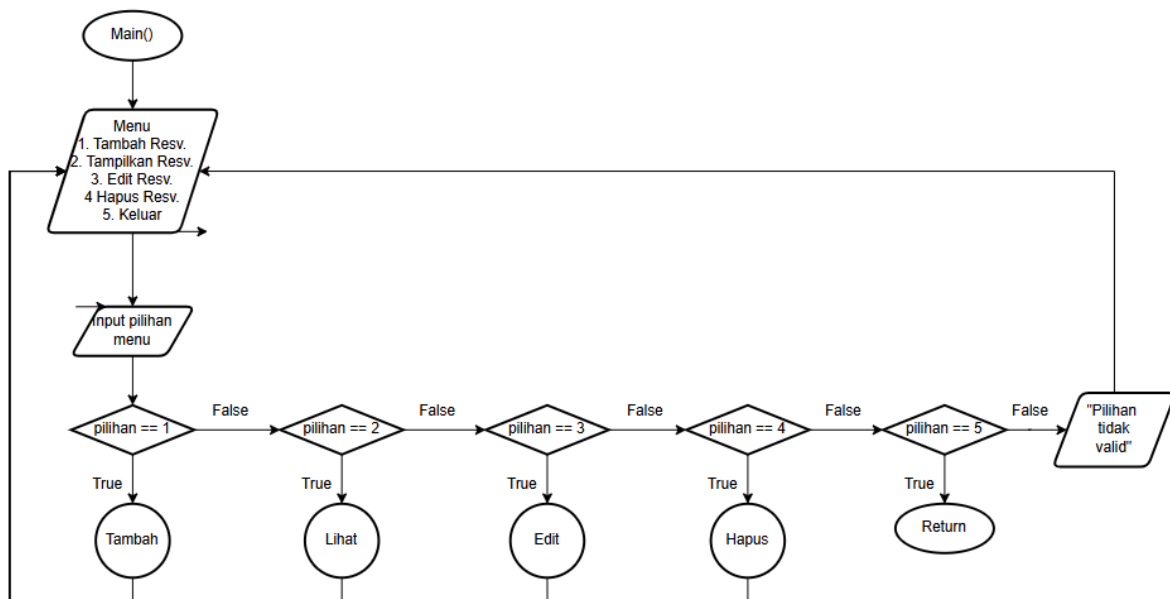
Disusun oleh:
Muhammad Britama Putra Jaya (2409106047)
Kelas (B1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

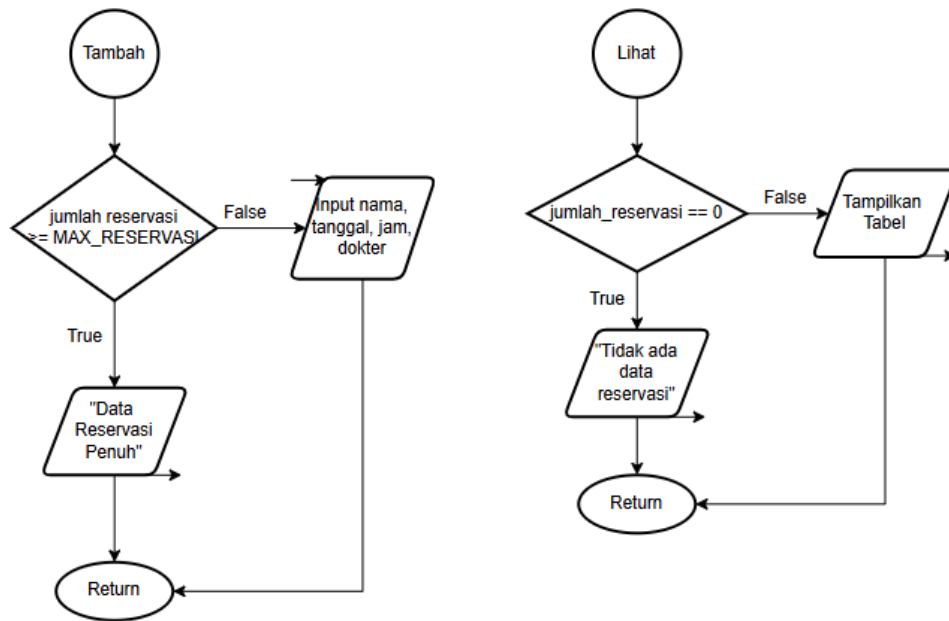
1. Flowchart



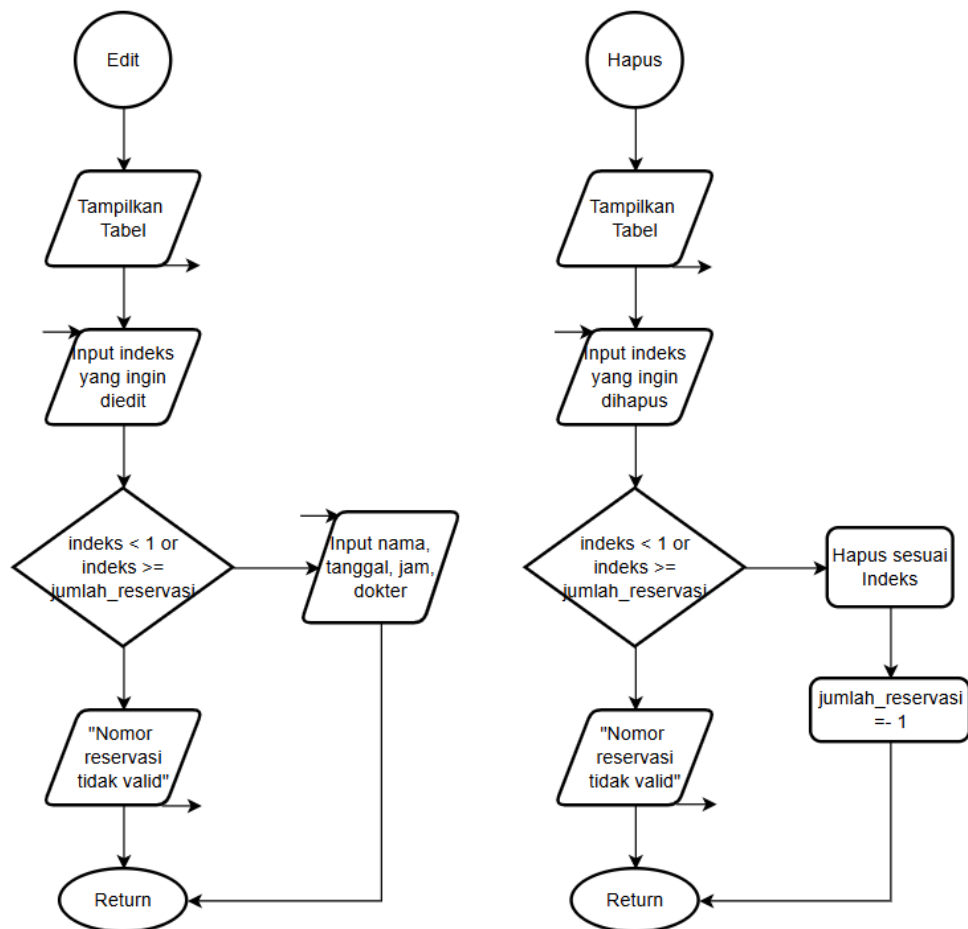
Gambar 1.1 Flowchart Login



Gambar 1.2 Flowchart Menu



Gambar 1.3 Flowchart Tambah Reservasi dan Lihat Reservasi



Gambar 1.4 Flowchart Edit Reservasi dan Hapus Reservasi

2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini merupakan sistem manajemen reservasi untuk klinik gigi yang memungkinkan pengguna untuk menambahkan, melihat, mengedit, dan menghapus data reservasi pasien. Sistem ini menggunakan login sederhana untuk mengakses menu utama. Data reservasi disimpan dalam array dua dimensi, mencakup nama pasien, tanggal, jam, dan nama dokter. Program juga menampilkan daftar reservasi dalam format tabel yang rapi, memastikan keterbacaan dan kemudahan penggunaan.

2.2 Penjelasan Alur & Algoritma

Pertama kali program dimulai, user akan diminta untuk login menggunakan nama dan NIM. Apabila user salah memasukkan nama atau NIM sebanyak 3 kali, maka program akan secara otomatis memblokir user. Setelah berhasil login maka user akan dibawa ke menu utama program. Dimana user dapat melakukan CRUD, mulai dari “Tambah Reservasi”, “Tampilkan Reservasi”, “Edit Reservasi”, “Hapus Reservasi” dan “Keluar”. Apabila user memilih menu “Tambah Reservasi”, program akan mengecek apakah jumlah reservasi yang ada tidak melewati batas maksimal reservasi yaitu 20. Kalau sudah 20, maka user tidak dapat melakukan reservasi lagi. Saat menambahkan reservasi user diminta untuk menginput nama, jam, tanggal, dan dokter, setelah selesai inputan dari user akan masuk ke array. Apabila user memilih menu “Tampilkan reservasi”, program akan menampilkan semua reservasi yang ada di array dalam bentuk tabel. Jika tidak ada reservasi sama sekali, maka program akan mengeluarkan output “Tidak ada reservasi”. Apabila user memilih menu “Edit Reservasi”, maka program akan menampilkan tabel reservasi dan meminta user untuk menginput nomor reservasi yang ingin diedit, jika nomor reservasi yang diinput tidak ada dalam tabel maka program akan mengeluarkan output “Nomor reservasi tidak valid”. Program akan meminta user untuk kembali menginput ulang nama, jam, tanggal, dan dokter. Apabila user memilih menu “Hapus Reservasi”, maka program akan menampilkan tabel reservasi dan meminta user untuk menginput nomor reservasi yang ingin dihapus, jika nomor reservasi yang diinput tidak ada dalam tabel maka program akan mengeluarkan output “Nomor reservasi tidak valid”. Setelah itu program akan secara otomatis menghapus nomor reservasi yang telah di input oleh user. Apabila user memilih menu “Keluar”, maka program akan berhenti.

3. Source Code

A. Menyimpan Data Reservasi dalam Array 2D

Fitur ini digunakan untuk menyimpan data reservasi dalam array dua dimensi, dengan tiap baris mewakili satu reservasi dan kolomnya berisi nama pasien, tanggal, jam, dan dokter. Disini juga terdapat MAX_RESERVASI untuk membatasi array hingga hanya 20 reservasi.

Source Code:

```
const int MAX_RESERVASI = 20;
int jumlah_reservasi = 0;

string reservasi[MAX_RESERVASI][4]; // [0] = Nama Pasien, [1] = Tanggal, [2] =
Jam, [3] = Dokter
```

B. Tambah Reservasi

Fitur ini digunakan menambah reservasi klinik gigi, dimana program akan mengecek apakah reservasi sudah penuh, meminta user untuk menginput nama pasien, tanggal, jam, dan dokter, serta menambah jumlah_reservasi setelah data berhasil disimpan.

Source Code:

```
if (pilihan == 1) {
    if (jumlah_reservasi >= MAX_RESERVASI) {
        cout << "Data reservasi penuh!" << endl;
    } else {
        cout << "\nMasukkan Nama Pasien : ";
        getline(cin, reservasi[jumlah_reservasi][0]);
        cout << "Masukkan Tanggal (DD-MM-YYYY) : ";
        getline(cin, reservasi[jumlah_reservasi][1]);
        cout << "Masukkan Jam (HH:MM) : ";
        getline(cin, reservasi[jumlah_reservasi][2]);
        cout << "Masukkan Nama Dokter : ";
        string nama_dokter;
        getline(cin, nama_dokter);
        reservasi[jumlah_reservasi][3] = "Drg. " + nama_dokter;
        jumlah_reservasi++;
        cout << "Reservasi berhasil ditambahkan!" << endl;
    }
}
```

C. Tampilkan Reservasi

Fitur ini digunakan untuk menampilkan data reservasi secara rapi ke dalam bentuk tabel menggunakan `setw()`. Sebelum itu program akan mengecek apakah ada reservasi yang tersedia.

Source Code:

```
else if (pilihan == 2 || pilihan == 3 || pilihan == 4) {
    if (jumlah_reservasi == 0) {
        cout << "Tidak ada data reservasi!" << endl;
    } else {
        cout <<
        "\n===== " <<
        endl;
        cout << "| No | Nama Pasien | Tanggal | Jam |" << endl;
        cout <<
        "===== " << endl;
        for (int i = 0; i < jumlah_reservasi; i++) {
            cout << "| " << setw(3) << i+1 << " | "
                << setw(20) << left << reservasi[i][0] << " | "
                << setw(10) << reservasi[i][1] << " | "
                << setw(5) << reservasi[i][2] << " | "
                << setw(15) << left << reservasi[i][3] << " | " << endl;
        }
        cout <<
        "===== " << endl;
    }
}
```

D. Edit Reservasi

Fitur ini digunakan mengedit reservasi yang ada sesuai dengan nomor reservasi yang ingin diedit.

Source Code:

```
if (pilihan == 3) {
    int indeks;
    cout << "Masukkan nomor reservasi yang ingin diedit : ";
    cin >> indeks;
    cin.ignore();
    if (indeks < 1 || indeks > jumlah_reservasi) {
        cout << "Nomor reservasi tidak valid!" << endl;
    } else {
```

```

        cout << "\nMasukkan Nama Pasien : ";
        getline(cin, reservasi[indeks-1][0]);
        cout << "Masukkan Tanggal (DD-MM-YYYY) : ";
        getline(cin, reservasi[indeks-1][1]);
        cout << "Masukkan Jam (HH:MM) : ";
        getline(cin, reservasi[indeks-1][2]);
        cout << "Masukkan Nama Dokter : ";
        string nama_dokter;
        getline(cin, nama_dokter);
        reservasi[indeks-1][3] = "Drg. " + nama_dokter;
        cout << "Reservasi berhasil diperbarui!" << endl;
    }
}

```

E. Hapus Reservasi

Fitur ini digunakan untuk menghapus reservasi yang ada sesuai dengan nomor reservasi yang ingin dihapus.

Source Code:

```

if (pilihan == 4) {
    int indeks;
    cout << "Masukkan nomor reservasi yang ingin dihapus : ";
    cin >> indeks;
    cin.ignore();
    if (indeks < 1 || indeks > jumlah_reservasi) {
        cout << "Nomor reservasi tidak valid!" << endl;
    } else {
        for (int i = indeks - 1; i < jumlah_reservasi - 1; i++) {
            for (int j = 0; j < 4; j++) {
                reservasi[i][j] = reservasi[i + 1][j];
            }
        }
        jumlah_reservasi--;
        cout << "Reservasi berhasil dihapus!" << endl;
    }
}

```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

Skenario 1 : Login Berhasil dan Menambahkan Reservasi

1. Pengguna memasukkan nama dan NIM yang benar.
2. Memilih menu 1 (Tambah Reservasi).
3. Memasukkan Nama Pasien, Tanggal, Jam, dan Nama Dokter.
4. Sistem menampilkan pesan bahwa reservasi berhasil ditambahkan.

Skenario 2 : Menampilkan Reservasi yang Sudah Ada

1. Pengguna memilih menu 2 (Tampilkan Reservasi).
2. Jika ada data reservasi, program menampilkan daftar dalam tabel.
3. Jika tidak ada data, program akan memberi tahu pengguna.

Skenario 3 : Gagal Login dan Terblokir

1. Pengguna salah memasukkan nama atau NIM sebanyak tiga kali.
2. Program menampilkan peringatan setiap kali login gagal.
3. Setelah percobaan ketiga, program berhenti.

4.2 Hasil Output

```
=====
                        LOGIN SISTEM KLINIK
=====
Masukkan Nama : britama
Masukkan NIM : 2409106047

=====
                MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar

=====
Pilih menu : 1

Masukkan Nama Pasien : Yanto
Masukkan Tanggal (DD-MM-YYYY) : 18-03-2025
Masukkan Jam (HH:MM) : 14:00
Masukkan Nama Dokter : Ernesto
Reservasi berhasil ditambahkan!
```

Gambar 4.1 Skenario 1

```
=====
                MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar

=====
Pilih menu : 2

=====
| No | Nama Pasien          | Tanggal    | Jam   | Dokter          |
=====
|  1 | Yanto                | 18-03-2025 | 14:00 | Drg. Ernesto    |
=====
```

Gambar 4.2 Skenario 2

```
=====
LOGIN SISTEM KLINIK
=====
Masukkan Nama : abc
Masukkan NIM : abc
Login gagal! Sisa kesempatan : 2

=====
LOGIN SISTEM KLINIK
=====
Masukkan Nama : abc
Masukkan NIM : abc
Login gagal! Sisa kesempatan : 1

=====
LOGIN SISTEM KLINIK
=====
Masukkan Nama : abc
Masukkan NIM : abc
Login gagal! Sisa kesempatan : 0
Anda diblokir! Silahkan hubungi admin.
```

Gambar 4.3 Skenario 3

5. Git

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENTS

PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> git init
Initialized empty Git repository in C:/Users/my asus/OneDrive/Desktop/praktikum-apl/.git/
```

Gambar 5.1 git init : Membuat repository lokal yang kosong

```
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> git add .
warning: in the working copy of 'post-test/post-test-1/.vscode/tasks.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.drawio', LF will be replaced by CRLF the next time Git touches it
```

Gambar 5.2 git add : Menambah file ke git commit. Dengan menambahkan “.” secara otomatis semua perubahan file yang ada di direktori “praktikum-apl”

```
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> git commit -m "Finish Code Post Test 1"
[main (root-commit) 7310fea] Finish Code Post Test 1
5 files changed, 367 insertions(+)
create mode 100644 pertemuan-1/main.cpp
create mode 100644 post-test/post-test-1/.vscode/tasks.json
create mode 100644 post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.cpp
create mode 100644 post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.drawio
create mode 100644 post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.exe
```

Gambar 5.3 git commit : digunakan untuk menyimpan perubahan dalam repository Git secara permanen.

```
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (11/11), 679.42 KiB | 3.47 MiB/s, done.
Total 11 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Britamaxx/praktikum-apl.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl>
```

Gambar 5.4 git push : digunakan untuk mengirim perubahan (commit) dari repository lokal ke repository jarak jauh, sehingga bisa diakses oleh orang lain atau disinkronkan dengan repository utama.