

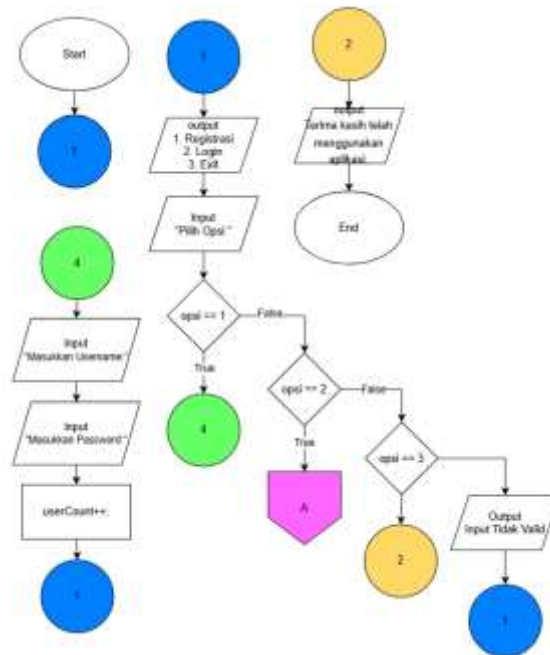
LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT



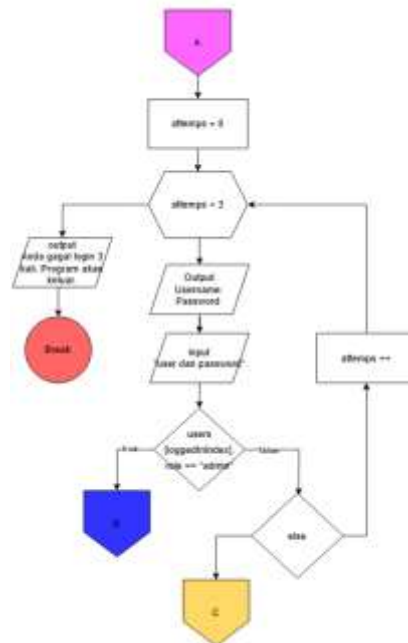
Disusun oleh:
Ikhsan (2409106118)
Kelas (C2 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

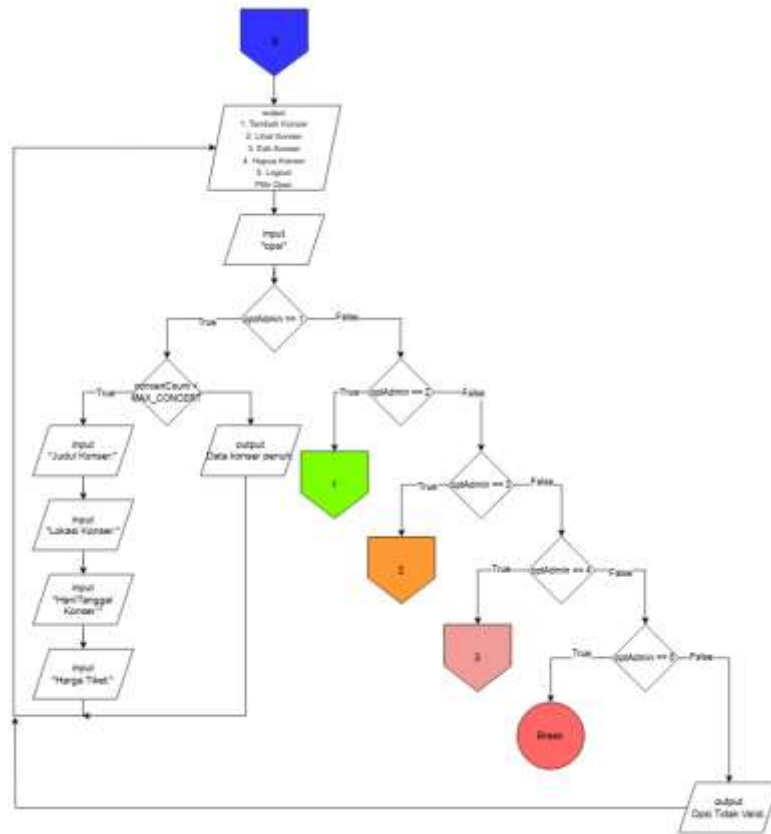
1. Flowchart



Gambar 1.1 Flowchart



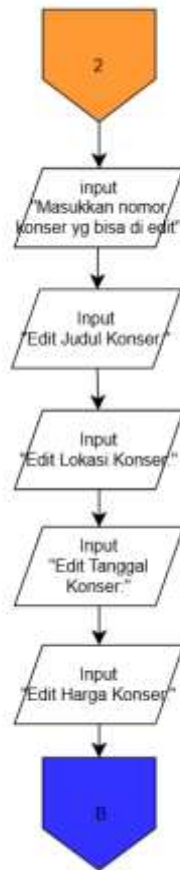
Gambar 1.2 Flowchart



Gambar 1.3 Flowchart



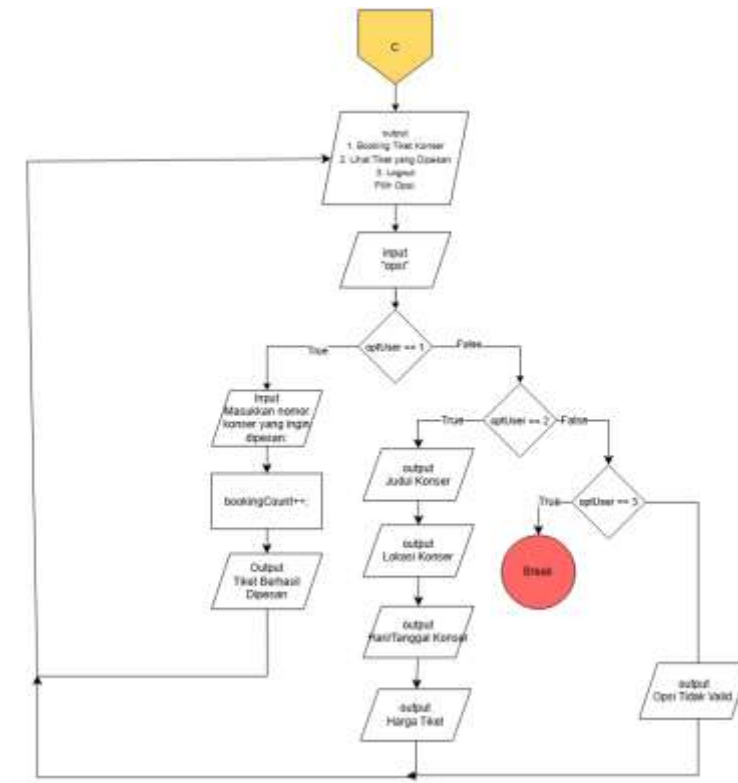
Gambar 1.4 Flowchart



Gambar 1.5 Flowchart



Gambar 1.6 Flowchart



Gambar 1.7 Flowchart

2. Analisis Program

2.1 Tujuan dan Fungsi Program

Program ini dibuat untuk **mengelola pemesanan tiket konser** secara sederhana dengan fitur:

1. **Registrasi & Login Pengguna**
 - Pengguna bisa mendaftar dan login ke sistem.
 - Terdapat dua jenis akun: **admin** dan **user**.
2. **Fitur Admin**
 - Menambahkan konser baru.
 - Melihat daftar konser yang tersedia.
 - Mengedit data konser.
 - Menghapus konser.
3. **Fitur Pengguna**
 - Melihat daftar konser yang tersedia.
 - Memesan tiket konser.
 - Melihat daftar tiket yang telah dipesan.

2.2 Manfaat Utama

- **Mudah Digunakan:** Antarmuka berbasis teks yang sederhana dan mudah dimengerti.
- **Manajemen Tiket yang Efektif:** Admin dapat mengelola konser, dan pengguna dapat melakukan pemesanan dengan mudah.
- **Keamanan:** Sistem login dengan pembatasan tiga kali percobaan untuk mencegah akses tidak sah.
- **Penyimpanan Data Sementara:** Menggunakan array untuk menyimpan data selama program berjalan.

3. Source Code

```
#include <iostream>
using namespace std;

#define MAX_CONCERT 100 // Ukuran maksimum array konser
#define INFO 4 // Kolom: {judul, lokasi, tanggal, harga}
#define MAX_BOOKING 100 // Ukuran maksimum array booking per user
#define MAX_USER 100 // Ukuran maksimum array pengguna

struct Concert {
    string title;
    string location;
    string date;
    string price;
};

struct Booking {
    Concert concert;
};

struct User {
    string username;
    string password;
    string role; // "admin" atau "user"
    int bookingCount;
    Booking bookings[MAX_BOOKING];
};

int main() {
    Concert concerts[MAX_CONCERT] = {
        {"Midnighth Serenade", "Samarinda", "11-Oktober-2024", "200000"},
        {"Arctic Monkey", "Balikpapan", "12-Oktober-2024", "500000"}
    };
    int concertCount = 2;

    User users[MAX_USER];
    int userCount = 2;
    // Pre-registered akun
    users[0].username = "admin";
    users[0].password = "admin";
    users[0].role = "admin";
    users[0].bookingCount = 0;

    users[1].username = "Ikhsan";
    users[1].password = "2409106118";
    users[1].role = "user";
    users[1].bookingCount = 0;
}
```

```

int bookingCount = 0;

while (true) {
    cout << "\n\n=====\n";
    cout << "          Pemesanan Tiket Konser\n";
    cout << "  1. Register\n";
    cout << "  2. Login   \n";
    cout << "  3. Exit    \n";
    cout << "=====\n";
    cout << "Pilih opsi: ";
    int opsi;
    cin >> opsi;
    cin.ignore();
    cout << "\n";

    if (opsi == 1) {
        cout << "\n=== Registrasi Pengguna Baru ===\n";
        if (userCount < MAX_USER) {
            cout << "Masukkan username: ";
            cin >> users[userCount].username;
            cout << "Masukkan password: ";
            cin >> users[userCount].password;
            users[userCount].role = "user";
            users[userCount].bookingCount = 0;
            userCount++;
            cout << "\nRegistrasi berhasil! Silakan login.\n";
        } else {
            cout << "\nMaksimum pengguna telah tercapai.\n";
        }
    }
    else if (opsi == 2) {
        int attempts = 0;
        bool loggedIn = false;
        string user, pass;
        int loggedInIndex = -1;
        while (attempts < 3 && !loggedIn) {
            cout << "Username: ";
            cin >> user;
            cout << "Password: ";
            cin >> pass;
            cin.ignore();
            for (int i = 0; i < userCount; i++) {
                if (users[i].username == user && users[i].password ==
pass) {
                    loggedIn = true;
                    loggedInIndex = i;

```



```

        cout << "\nLogin berhasil!\n";
        break;
    }
}
if (!loggedIn) {
    attempts++;
    if (attempts < 3)
        cout << "\nLogin gagal. Coba lagi (" << 3 - attempts
<< " kali tersisa).\n";
    }
}
if (!loggedIn) {
    cout << "\nAnda gagal login 3 kali. Program akan keluar.\n";
    break;
}

// Tampilkan menu berdasarkan role tanpa memeriksa username
secara eksplisit
if (users[loggedInIndex].role == "admin") {
    while (true) {
        cout << "\n\n===== Menu Admin =====\n";
        cout << "1. Tambah Konser\n";
        cout << "2. Lihat Konser\n";
        cout << "3. Edit Konser\n";
        cout << "4. Hapus Konser\n";
        cout << "5. Logout\n";
        cout << "=====\n";
        cout << "Pilih opsi: ";
        int optAdmin;
        cin >> optAdmin;
        cin.ignore();

        if (optAdmin == 1) {
            if (concertCount < MAX_CONCERT) {
                cout << "\n=== Tambah Konser ===\n";
                cout << "Judul Konser: ";
                getline(cin, concerts[concertCount].title);
                cout << "Lokasi Konser: ";
                getline(cin, concerts[concertCount].location);
                cout << "Hari/Tanggal Konser: ";
                getline(cin, concerts[concertCount].date);
                cout << "Harga Tiket: ";
                getline(cin, concerts[concertCount].price);
                concertCount++;
                cout << "\nKonser berhasil ditambahkan.\n";
            } else {
                cout << "\nData konser penuh.\n";
            }
        }
    }
}

```

[illegible]

```

        getline(cin, concerts[idx].title);
        cout << "Edit Lokasi Konser (" <<
concerts[idx].location << "): ";
        getline(cin, concerts[idx].location);
        cout << "Edit Hari/Tanggal (" <<
concerts[idx].date << "): ";
        getline(cin, concerts[idx].date);
        cout << "Edit Harga Tiket (" <<
concerts[idx].price << "): ";
        getline(cin, concerts[idx].price);
        cout << "\nKonser berhasil diubah.\n";
    }
}
}
else if (optAdmin == 4) {
    if (concertCount == 0)
        cout << "\nTidak ada konser yang bisa
dihapus.\n";
    else {
        cout << "\nDaftar Konser:\n";
        for (int i = 0; i < concertCount; i++) {
            cout << i + 1 << ". " << concerts[i].title
<< "\n";
        }
        cout << "Masukkan nomor konser yang ingin
dihapus: ";

        int num;
        cin >> num;
        cin.ignore();
        if (num < 1 || num > concertCount)
            cout << "Nomor tidak valid.\n";
        else {
            int idx = num - 1;
            for (int i = idx; i < concertCount - 1; i++)
            {
                concerts[i] = concerts[i + 1];
            }
            concertCount--;
            cout << "\nKonser berhasil dihapus.\n";
        }
    }
}
else if (optAdmin == 5) {
    break;
}
else {
    cout << "\nOpsi tidak valid.\n";
}
}

```



```

        cout << "| Judul Konser          | Lokasi\n";
Konser    | Hari/Tanggal Konser   | Harga Tiket|\n";
        cout << "+-----+-----+\n";
        +-----+-----+\n";
            for (int i = 0; i <
users[loggedInIndex].bookingCount; i++) {
                cout << "| " <<
users[loggedInIndex].bookings[i].concert.title;
                    int sp = 20 -
users[loggedInIndex].bookings[i].concert.title.length();
                        for (int j = 0; j < sp; j++) cout << " ";
                            cout << " | " <<
users[loggedInIndex].bookings[i].concert.location;
                                sp = 14 -
users[loggedInIndex].bookings[i].concert.location.length();
                                    for (int j = 0; j < sp; j++) cout << " ";
                                        cout << " | " <<
users[loggedInIndex].bookings[i].concert.date;
                                            sp = 20 -
users[loggedInIndex].bookings[i].concert.date.length();
                                                for (int j = 0; j < sp; j++) cout << " ";
                                                    cout << " | " <<
users[loggedInIndex].bookings[i].concert.price;
                                                        sp = 10 -
users[loggedInIndex].bookings[i].concert.price.length();
                                                            for (int j = 0; j < sp; j++) cout << " ";
                                                                cout << " |\n";
                                                                    }
                                                                        cout << "+-----+-----+\n";
                                                                            +-----+-----+\n";
                                                                                }
                                                                                    }
                                                                                        else if (optUser == 3) {
                                                                                            break;
                                                                                        }
                                                                                            else {
                                                                                                cout << "\nOpsi tidak valid.\n";
                                                                                              }
                                                                                                  cout << "\nTekan Enter untuk melanjutkan...";
                                                                                                      cin.ignore();
                                                                                                          cin.get();
                                                                                                              }
                                                                                                                  }
                                                                                                                      }
else if (opsi == 3) {
    cout << "\nTerima kasih telah menggunakan aplikasi.\n";
    break;
}

```

```
    else {  
        cout << "\nInput tidak valid.\n";  
        cout << "\nTekan Enter untuk melanjutkan...";  
        cin.ignore();  
        cin.get();  
    }  
}  
return 0;  
}
```

4. Uji Coba dan Hasil Output

```
=====
                Pemesanan Tiket Konser
            1. Register
            2. Login
            3. Exit
=====
Pilih opsi: 1

=== Registrasi Pengguna Baru ===
Masukkan username: ican
Masukkan password: 2409106001

Registrasi berhasil! Silakan login.

=====
                Pemesanan Tiket Konser
            1. Register
            2. Login
            3. Exit
=====
Pilih opsi: 2

Username: ican
Password: 2409106001

Login berhasil!

===== Menu Pengguna =====
1. Booking Tiket Konser
2. Lihat Tiket yang Dipesan
3. Logout
=====
Pilih opsi: 1

----- Daftar Konser -----
1. Midnigth Serenade | Samarinda | 11-Oktober-2024 | 200000
2. Arctic Monkey | Balikpapan | 12-Oktober-2024 | 500000

Masukkan nomor konser yang ingin dipesan: 1

Tiket berhasil dipesan.

Tekan Enter untuk melanjutkan...
```

Gambar 4.1 Output Regis, Login dan Beli Tiket

```

===== Menu Pengguna =====
1. Booking Tiket Konser
2. Lihat Tiket yang Dipesan
3. Logout
=====
Pilih opsi: 2

===== Tiket yang Dipesan =====
+-----+-----+-----+-----+
| Judul Konser | Lokasi Konser | Hari/Tanggal Konser | Harga Tiket |
+-----+-----+-----+-----+
| Midnigh Serenade | Samarinda | 11-Oktober-2024 | 200000 |
+-----+-----+-----+-----+

Tekan Enter untuk melanjutkan...

===== Menu Pengguna =====
1. Booking Tiket Konser
2. Lihat Tiket yang Dipesan
3. Logout
=====
Pilih opsi: 3

=====
Pemesanan Tiket Konser
1. Register
2. Login
3. Exit
=====
Pilih opsi: 3

Terima kasih telah menggunakan aplikasi.

```

Gambar 4.2 Output Lihat Tiket Konser Acc Baru

```

=====
Pemesanan Tiket Konser
1. Register
2. Login
3. Exit
=====
Pilih opsi: 2

Username: Admin
Password: admin

Login gagal. Coba lagi (2 kali tersisa).
Username: admin
Password: admin

Login berhasil!

===== Menu Admin =====
1. Tambah Konser
2. Lihat Konser
3. Edit Konser
4. Hapus Konser
5. Logout
=====
Pilih opsi: 1

=== Tambah Konser ===
Judul Konser: Denny Caknan
Lokasi Konser: Samarinda
Hari/Tanggal Konser: 18-Januari-2025
Harga Tiket: 100000

Konser berhasil ditambahkan.

Tekan Enter untuk melanjutkan...

```

Gambar 4.3 Output Admin Tambah Konser


```

----- Menu Admin -----
1. Tambah Konser
2. Lihat Konser
3. Edit Konser
4. Hapus Konser
5. Logout
-----
Pilih opsi: 2

----- Daftar Konser -----
+-----+-----+-----+-----+
| Judul Konser | Lokasi Konser | Hari/Tanggal Konser | Harga Tiket |
+-----+-----+-----+-----+
| Midnigth Serenade | Samarinda | 11-Oktober-2024 | 200000 |
| Arctic Monkey | Balikpapan | 12-Oktober-2024 | 500000 |
| Denny Caknan | Samarinda | 18-Januari-2025 | 100000 |
+-----+-----+-----+-----+

Tekan Enter untuk melanjutkan...

----- Menu Admin -----
1. Tambah Konser
2. Lihat Konser
3. Edit Konser
4. Hapus Konser
5. Logout
-----
Pilih opsi: 3

Daftar Konser:
1. Midnigth Serenade
2. Arctic Monkey
3. Denny Caknan
Masukkan nomor konser yang ingin diedit: 2
Edit Judul Konser (Arctic Monkey): Ikhsan Band
Edit Lokasi Konser (Balikpapan): Samarinda
Edit Hari/Tanggal (12-Oktober-2024): 4-Januari-2025
Edit Harga Tiket (500000): 1000000

Konser berhasil diubah.

Tekan Enter untuk melanjutkan...

```

Gambar 4.4 Output Lihat Konser dan Edit

```

----- Menu Admin -----
1. Tambah Konser
2. Lihat Konser
3. Edit Konser
4. Hapus Konser
5. Logout
-----
Pilih opsi: 4

Daftar Konser:
1. Midnigth Serenade
2. Ikhsan Band
3. Denny Caknan
Masukkan nomor konser yang ingin dihapus: 2

Konser berhasil dihapus.

Tekan Enter untuk melanjutkan...

----- Menu Admin -----
1. Tambah Konser
2. Lihat Konser
3. Edit Konser
4. Hapus Konser
5. Logout
-----
Pilih opsi: 5

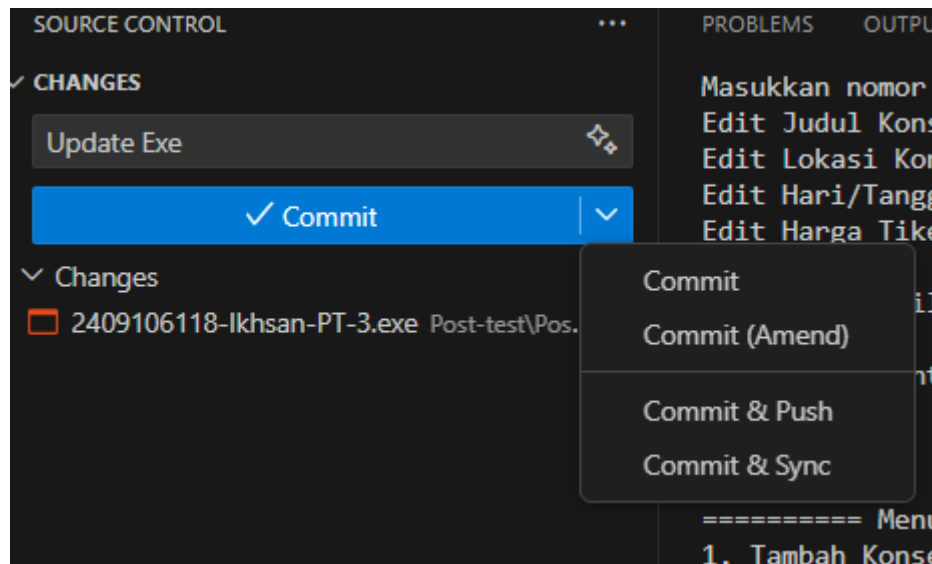
-----
Pemesanan Tiket Konser
1. Register
2. Login
3. Exit
-----
Pilih opsi: 3

Terima kasih telah menggunakan aplikasi.
PS D:\GIT\B\Praktikum-Apl\Post-test\Post-test-3>

```

Gambar 4.5 Output Hapus hingga Exit

5. Langkah-Langkah Git pada VSCode



Gambar 5.1 Source Control Git Visual Code Studio

Pertama buka source control pada bagian sidebar di aplikasi Visual Code Studio, lalu input pesan yang ingin dipasang di message commit cintihnya seperti punya saya “Update Exe”, lalu pencet tanda panah kebawah untuk mencari opsi “Commit & Push” agar programnya terupdate dan tersimpan di Github Cloud.



Gambar 5.2 Hasil apabila berhasil masuk Cloud