

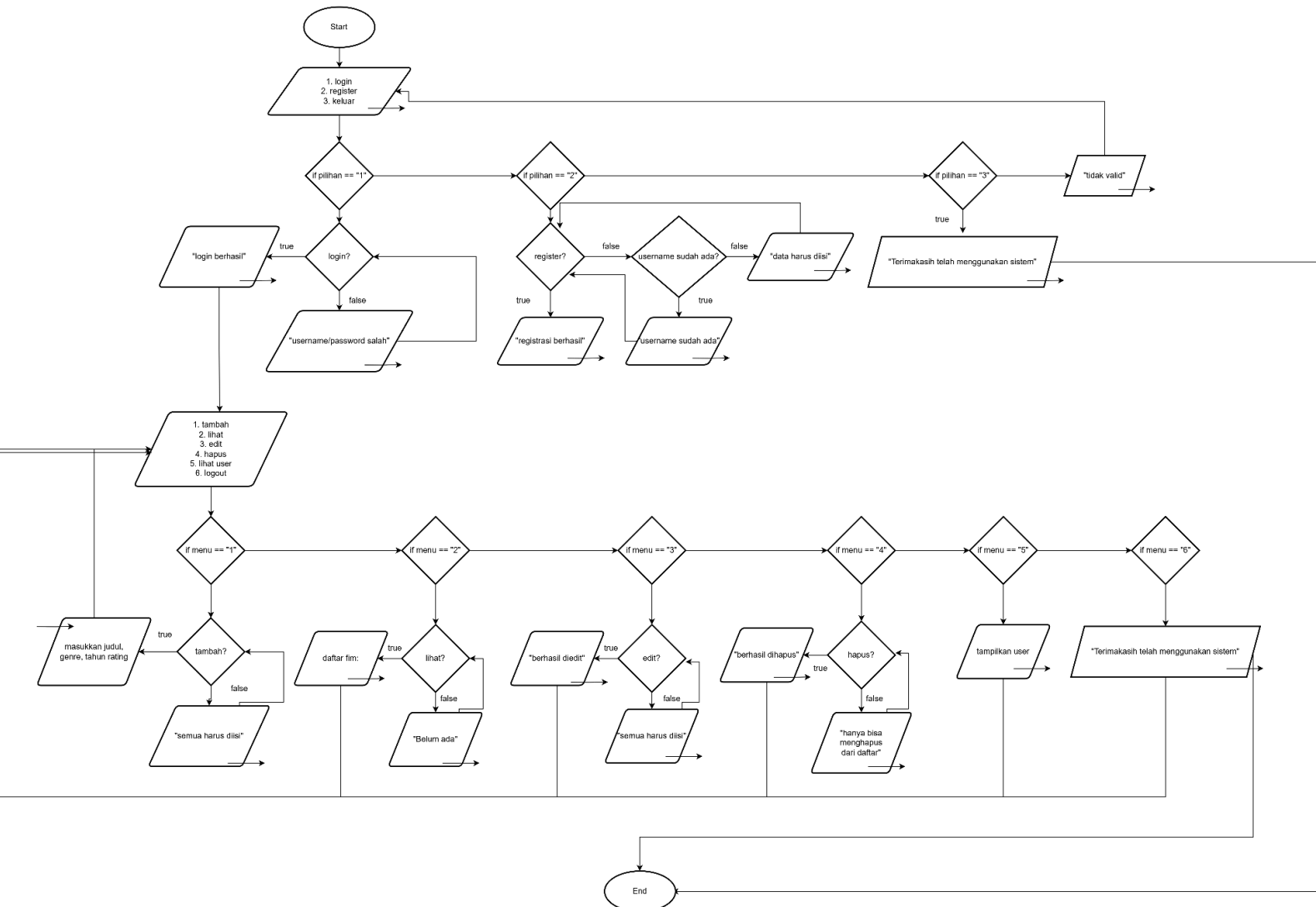
**LAPORAN PRAKTIKUM**  
**POSTTEST 6**  
**ALGORITMA PEMROGRAMAN DASAR**



**Disusun oleh:**  
**Ammara Fadia Fanysarahman (2509106035)**  
**Kelas (A'25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

## 1. Flowchart



## 2. Deskripsi Singkat Program

Program ini merupakan sistem manajemen koleksi film/series sederhana yang memungkinkan pengguna mencatat film/series yang sudah mereka tonton (create), menaampilkan daftar film/series yang sudah mereka tonton (read), mengedit data film/series yang sudah mereka tonton (update), menghapus daftar film/series yang sudah ada (delete).

### 3. Source Code

#### A. Menu utama

Ini digunakan untuk memilih login, register, keluar dari program.

##### Source Code:

```
print("=== SISTEM MANAJEMEN FILM/SERIES ===")
print("1. Login")
print("2. Register")
print("3. Keluar")

pilihan = input("Pilih menu (1-3): ")
```

#### B. Login (pilihan 1)

Ini digunakan untuk melakukan login dan mengecek status login.

##### Source Code:

```
print("=== LOGIN ===")
username = input("Username: ")
password = input("Password: ")

# Cek Login
login_sukses = False
for user in users:
    if user["username"] == username and user["password"] == password:
        current_user = user
        login_sukses = True
        break

if login_sukses:
    print(f"Login berhasil! Selamat datang {username}")
    input("Tekan Enter untuk melanjutkan...")
else:
    print("Error: Username atau password salah!")
    input("Tekan Enter untuk melanjutkan...")
```

### C. Menu (setelah login)

Menampilkan 6 pilihan berupa tambah film/series, lihat film/series, edit film/series, hapus film/series, lihat user (khusus admin), logout.

#### Source Code:

```
while True:
    clear_screen()
    print(f"=== MENU UTAMA ({current_user['username']}) ===")
    print("1. Tambah Film/Series")
    print("2. Lihat Semua Film/Series")
    print("3. Edit Film/Series")
    print("4. Hapus Film/Series")
    if current_user["role"] == "admin":
        print("5. Lihat Semua User")
    print("6. Logout")

    menu = input("Pilih menu (1-6): ")
```

### D. Tambah (user/admin)

Untuk menambahkan daftar film/series baru.

#### Source Code:

```
if menu == "1": # Tambah film/series
    clear_screen()
    print("=== TAMBAH FILM/SERIES ===")
    judul = input("Judul film/series: ")
    genre = input("Genre: ")
    tahun = input("Tahun rilis: ")
    rating = input("Rating (1-10): ")

    # Validasi input
    if not judul or not genre or not tahun or not rating:
        print("Error: Semuanya harus diisi!")
    else:
        # Cek apakah judul sudah ada
        judul_ada = False
        for film in movies:
            if film["judul"].lower() == judul.lower():
```

```

        judul_ada = True
        break

    if judul_ada:
        print("Error: Film/series dengan judul tersebut
sudah ada!")

    else:
        # Validasi rating
        if rating.isdigit():
            rating_num = int(rating)
            if 1 <= rating_num <= 10:
                film_baru = {
                    "judul": judul,
                    "genre": genre,
                    "tahun": tahun,
                    "rating": rating,
                    "added_by": current_user["username"]
                }
                movies.append(film_baru)
                print("Film/Series berhasil
ditambahkan!")

            else:
                print("Error: Rating harus antara
1-10!")

        else:
            print("Error: Rating harus angka antara
1-10!")

    input("Tekan Enter untuk melanjutkan...")

elif menu == "2": # Lihat semua film/series
    clear_screen()
    print("=== DAFTAR FILM/SERIES ===")
    if not movies:
        print("Belum ada film/series dalam koleksi.")
    else:
        for i, film in enumerate(movies, 1):
            print(f"{i}. {film[0]} - {film[1]} ({film[2]}) -
Rating: {film[3]} - Ditambahkan oleh: {film[4]}")
        input("Tekan Enter untuk melanjutkan...")

```

## E. Edit (admin)

Untuk mengedit daftar yang sudah ada.

**Source Code:**

```

elif menu == "3": # Edit film/series
    clear_screen()
    print("=== EDIT FILM/SERIES ===")
    if not movies:
        print("Belum ada film/series dalam koleksi.")
    else:
        for i, film in enumerate(movies, 1):
            print(f"{i}. {film['judul']} - {film['genre']} ({film['tahun']}) - Rating: {film['rating']}")

        nomor = input("Pilih nomor film/series yang akan
diedit: ")

        if nomor.isdigit():
            index = int(nomor) - 1
            if 0 <= index < len(movies):
                film = movies[index]
                # Cek akses - hanya admin yang bisa edit
                if current_user["role"] == "admin":
                    print(f"Edit film: {film['judul']}")
                    judul_baru = input(f"Judul baru
({film['judul']}: ") or film['judul']
                    genre_baru = input(f"Genre baru
({film['genre']}: ") or film['genre']
                    tahun_baru = input(f"Tahun baru
({film['tahun']}: ") or film['tahun']
                    rating_baru = input(f"Rating baru
({film['rating']}: ") or film['rating']

                    # Validasi rating baru
                    if rating_baru.isdigit():
                        rating_num = int(rating_baru)
                        if 1 <= rating_num <= 10:
                            movies[index] = {
                                "judul": judul_baru,
                                "genre": genre_baru,
                                "tahun": tahun_baru,
                                "rating": rating_baru,
                                "added_by": film['added_by']
                            }
                            print("Film/Series berhasil
diupdate!")

                        else:
                            print("Error: Rating harus
antara 1-10!")

                    else:
                        print("Error: Rating harus angka
antara 1-10!")

                else:

```

```

                                print("Error: Hanya admin yang dapat
mengedit film/series!")
                                else:
                                    print("Error: Nomor film/series tidak
valid!")
                                else:
                                    print("Error: Input harus angka!")
                                input("Tekan Enter untuk melanjutkan...")

```

## F. Hapus (admin)

Untuk menghapus daftar yang sudah ada.

### Source Code:

```

elif menu == "4": # Hapus film/series
    clear_screen()
    print("=== HAPUS FILM/SERIES ===")
    if not movies:
        print("Belum ada film/series dalam koleksi.")
    else:
        for i, film in enumerate(movies, 1):
            print(f"{i}. {film['judul']} - {film['genre']}
({film['tahun']} - Rating: {film['rating']}")

        nomor = input("Pilih nomor film/series yang akan
dihapus: ")

        if nomor.isdigit():
            index = int(nomor) - 1
            if 0 <= index < len(movies):
                film = movies[index]
                # Cek hak akses - hanya admin yang bisa
hapus

                if current_user["role"] == "admin":
                    konfirmasi = input(f"Yakin hapus
film/series '{film['judul']}'? (yes/no): ")
                    if konfirmasi.lower() == 'yes':
                        movies.pop(index)
                        print("Film/Series berhasil
dihapus!")
                    else:
                        print("Error: Hanya admin yang dapat
menghapus film/series!")
                else:

```

```

        print("Error: Nomor film/series tidak
valid!")
    else:
        print("Error: Input harus angka!")
        input("Tekan Enter untuk melanjutkan...")

```

### G. Lihat user (admin)

Untuk melihat daftar user yang ada.

#### Source Code:

```

elif menu == "5" and current_user[2] == "admin": # Lihat semua user (admin
only)
    clear_screen()
    print("=== DAFTAR USER ===")
    for i, user in enumerate(users, 1):
        print(f"{i}. {user[0]} - Role: {user[2]}")
    input("Tekan Enter untuk melanjutkan...")

    elif menu == "6": # Logout
        current_user = None
        print("Logout berhasil!")
        input("Tekan Enter untuk melanjutkan...")
        break

    else:
        print("Pilihan tidak valid!")
        input("Tekan Enter untuk melanjutkan...")

else:
    print("Error: Username atau password salah!")
    input("Tekan Enter untuk melanjutkan...")

```

### H. Lihat Film/Series (user/admin)

Ini digunakan untuk melihat film/series yang sudah terdaftar.

#### Source Code:

```

elif menu == "2": # Lihat semua film/series

```



```

clear_screen()
print("=== DAFTAR FILM/SERIES ===")
if not movies:
    print("Belum ada film/series dalam koleksi.")
else:
    for i, film in enumerate(movies, 1):
        print(f"{i}. {film['judul']} - {film['genre']}")
    ({film['tahun']}) - Rating: {film['rating']} - Ditambahkan oleh:
    {film['added_by']}")
input("Tekan Enter untuk melanjutkan...")

```

## I. Register (pilihan 2)

Ini digunakan untuk melakukan daftar pengguna baru.

### Source Code:

```

elif menu == "5" and current_user["role"] == "admin": # Lihat semua user
(admin only)

clear_screen()
print("=== DAFTAR USER ===")
for i, user in enumerate(users, 1):
    print(f"{i}. {user['username']} - Role:
{user['role']}")
input("Tekan Enter untuk melanjutkan...")

```

## J. Logout (pilihan 3)

Ini digunakan untuk logout.

### Source Code:

```

elif menu == "6": # Logout
    current_user = None
    print("Logout berhasil!")
    input("Tekan Enter untuk melanjutkan...")
    break

```

## 4. Hasil Output

```
=== SISTEM MANAJEMEN FILM/SERIES ===  
1. Login  
2. Register  
3. Keluar  
Pilih menu (1-3): █
```

Gambar 4.1 Output pertama  
Menu utama.

```
=== LOGIN ===  
Username: admin  
Password: akuadminkausiapa  
Login berhasil! Selamat datang admin  
Tekan Enter untuk melanjutkan...█
```

Gambar 4.2 Output kedua  
Login.

```
=== MENU UTAMA (admin) ===  
1. Tambah Film/Series  
2. Lihat Semua Film/Series  
3. Edit Film/Series  
4. Hapus Film/Series  
5. Lihat Semua User  
6. Logout  
Pilih menu (1-6): 1█
```

Gambar 4.3 Output ketiga  
Menu setelah login.

```
=== TAMBAH FILM/SERIES ===  
Judul film/series: Fast X  
Genre: Action, Adventure, Crime  
Tahun rilis: 2023  
Rating (1-10): 9  
Film/Series berhasil ditambahkan!  
Tekan Enter untuk melanjutkan...█
```

Gambar 4.4 Output keempat  
Menambah.

```
=== DAFTAR FILM/SERIES ===  
1. Fast X - Action, Adventure, Crime (2023) - Rating: 9 - Ditambahkan oleh: admin  
Tekan Enter untuk melanjutkan...
```

Gambar 4.5 Output kelima  
Cek daftar.

```
=== EDIT FILM/SERIES ===  
1. Fast X - Action (2020) - Rating: 9  
Pilih nomor film/series yang akan diedit: 1  
Edit film: Fast X  
Judul baru (Fast X): Fast X  
Genre baru (Action): Action  
Tahun baru (2020): 2023  
Rating baru (9): 10  
Film/Series berhasil diupdate!  
Tekan Enter untuk melanjutkan...
```

Gambar 4.6 Output keenam  
Update.

```
=== DAFTAR FILM/SERIES ===  
1. Fast X - Action (2023) - Rating: 10 - Ditambahkan oleh: admin  
Tekan Enter untuk melanjutkan...
```

Gambar 4.7 Output ketujuh  
Cek setelah update.

```
=== HAPUS FILM/SERIES ===  
1. Fast X - Action, Adventure, Crime (2023) - Rating: 9.5  
Pilih nomor film/series yang akan dihapus: 1  
Yakin hapus film/series 'Fast X'? (yes/no): yes  
Film/Series berhasil dihapus!  
Tekan Enter untuk melanjutkan...
```

Gambar 4.8 Output kedelapan  
Menghapus.

```
=== DAFTAR FILM/SERIES ===  
Belum ada film/series dalam koleksi.  
Tekan Enter untuk melanjutkan...
```

Gambar 4.9 Output kesembilan  
Cek setelah dihapus.

```
=== DAFTAR USER ===  
1. admin - Role: admin  
Tekan Enter untuk melanjutkan...
```

Gambar 4.10 Output kesepuluh  
Daftar user.

```
=== MENU UTAMA (admin) ===  
1. Tambah Film/Series  
2. Lihat Semua Film/Series  
3. Edit Film/Series  
4. Hapus Film/Series  
5. Lihat Semua User  
6. Logout  
Pilih menu (1-6): 6  
Logout berhasil!  
Tekan Enter untuk melanjutkan...
```

Gambar 4.11 Output kesebelas  
Logout.

## 5. Langkah-langkah GIT

```
Microsoft Windows [Version 10.0.26100.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\Documents\praktikum-apd>git add .

C:\Users\ASUS\Documents\praktikum-apd>git commit -m "pt6"
[main 2df8373] pt6
 3 files changed, 315 insertions(+)
 rename kelas/{pertrmuan-5 => pertemuan-5}/main.py (100%)
 create mode 100644 kelas/pertemuan-6/main.py
 create mode 100644 post-test/post-test-apd-6/2509106035-Ammara Fadia-PT-6.py

C:\Users\ASUS\Documents\praktikum-apd>git push -u origin main
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 12 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (8/8), 3.24 KiB | 1.62 MiB/s, done.
Total 8 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/ammara fadia/praktikum-apd.git
   3a57977..2df8373  main -> main
branch 'main' set up to track 'origin/main'.

C:\Users\ASUS\Documents\praktikum-apd>
```

Gambar 5.1  
GIT.

### 5.1 GIT Add

Add adalah perintah yang berguna untuk staging atau untuk menambahkan perubahan atau registrasi perubahan pada git proyek yang ada.

### 5.2 GIT Commit

Commit adalah perintah yang digunakan untuk menyimpan perubahan pada repository git. Kita juga bisa mengembalikan file yang kita commit ke revisi sebelumnya.

### 5.3 GIT Push

Push adalah perintah yang digunakan untuk mengunggah konten pada repositori lokal.