

**LAPORAN PRAKTIKUM  
POSTTEST 8**

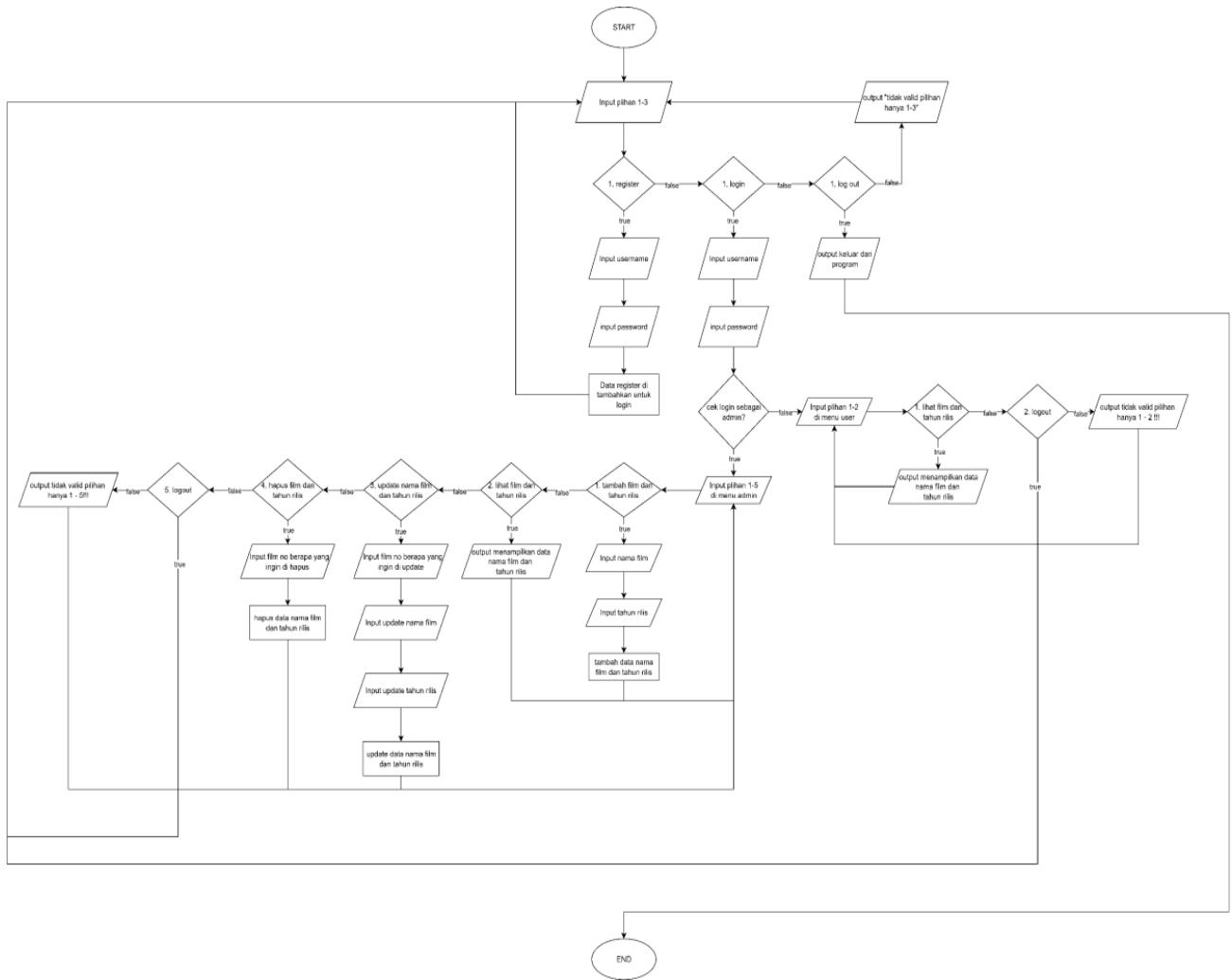
**ALGORITMA PEMROGRAMAN DASAR**



**Disusun oleh:**  
**Muhammad Hajar Aswad (2509106118)**  
**Kelas (C-25 ‘C2)**

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

## 1. Flowchart



### Penjelasan Flowchart

1. Mulai
2. Input pilihan di menu utama 1-3.
3. Kondisi Pilihan 1 register *true* di mana pengguna di minta untuk memasukkan username dan password yang kemudian data nya akan di simpan untuk menu login, kemudian setelah itu program akan kembali lagi ke menu utama untuk input pilihan 1-
3. Namun jika kondisi pilihan 1 maka ke kondisi pilihan 2.
4. Kondisi pilihan 2 *login* pengguna di minta untuk memasukkan username dan passowrd untuk login masuk ke menu admin atau user sesuai data login yang telah

di simpan. Namun jika false kondisi 2 maka ke kondisi pilihan 3.

5. Kondisi Pilihan 3 *true* logout di mana program akan berhenti. Namun jika Kondisi 3 false maka dia akan menampilkan output tidak valid pilihan hanya 1-3.
6. Kemudian akan mengecek apakah pengguna login sebagai admin atau bukan.
7. Jika *true* login sebagai admin akan masuk ke menu admin.
8. Input pilihan di menu admin 1-5.
9. Kondisi Pilihan 1 *true* menambahkan data, di mana admin dapat menambahkan data nama film dan tahun rilis film, kemudian setelah menambahkan program akan kembali lagi ke menu admin untuk input pilihan 1-5. Namun jika kondisi pilihan 1 false akan lanjut ke kondisi pilihan 2.
10. Kondisi Pilihan 2 *true* Menampilkan data, di mana admin dapat melihat data nama film serta tahun rilis yang telah ia tambahkan, kemudian setelah itu program akan kembali ke menu admin untuk input pilihan 1-5. Namun jika kondisi pilihan 2 false akan lanjut ke kondisi pilihan 3.
11. Kondisi Pilihan 3 *true* Mengupdate data, di mana admin dapat mengubah data nama film serta tahun rilis film yang telah di tambahkan sebelumnya, kemudian setelah itu program akan kembali ke menu admin untuk input pilihan 1-5. Namun jika kondisi pilihan 3 false akan lanjut ke kondisi pilihan 4.
12. Kondisi Pilihan 4 *true* hapus data, di mana admin dapat menghapus data nama film serta tahun rilis yang ada pada data yang telah di tambahkan sebelumnya, kemudian setelah itu program akan kembali ke menu admin untuk input pilihan 1-5. Namun jika kondisi pilihan 4 false akan lanjut ke kondisi pilihan 5.
13. Kondisi Pilihan 5 logout, jika true program akan kembali ke menu utama input pilihan 1-3 jika false program akan tetap di menu admin untuk input pilihan 1-5.
14. False login sebagai user maka akan masuk ke menu user.
15. Input pilihan di menu user 1-2.
16. Kondisi pilihan 1 *true* melihat data, maka user akan di tampilkan data nama film dan tahun rilis yang telah di tambahkan oleh admin. Namun jika kondisi pilihan 1 false akan ke kondisi pilihan 2.
17. Kondisi pilihan 2 *true* logout user akan keluar dari menu user dan kembali ke menu utama untuk input pilihan 1-3. Namun jika kondisi pilihan 2 false maka akan menampilkan output tidak valid pilihan hanya 1-2 dan kembali ke menu user input pilihan 1-2.
18. Selesai.

## **2. Deskripsi Singkat Program**

### **Tujuan**

Program ini di buat bertujuan untuk mengelola data nama film dan tahun rilisnya dengan sistem multi user ada admin dan user dan dua pengguna ini bisa login sesuai dengan tipe akunnya dan fungsi-fungsi pada program nya telah di pecah menjadi beberapa file yang bertujuan untuk lebih mempermudah mengelola program.

### **Fungsi**

1. Sebagai sistem manajemen film yang di mana ada admin yang dapat menambahkan,mengubah,melihat, dan menghapus data film beserta tahun rilisnya.
2. Bisa membedakan hak akses antara user dan admin yang di mana user hanya bisa melihat data nama film dan tahun rilis saja.
3. Melatih logika dan pemahaman dalam penggunaan fungsi untuk memudahkan dalam pembuatan program dan tidak perlu membuat fitur yang sama karena tinggal memanggil fungsi yang di perlukan saja.
4. Menerapkan konsep modular yaitu memecah beberapa fungsi menjadi beberapa file sesuai fungsinya masing-masing agar program lebih terstruktur, rapi, dan efisien.

## **3. Source Code**

### **A. File main**

#### **1. Fungsi Liblary (File main)**

Modul *auth* menyediakan fungsi register dan login untuk proses pendaftaran serta autentikasi pengguna. Modul *admin* berisi fungsi *admin\_menu* yang menampilkan menu khusus untuk admin, Modul *user* berisi fungsi *menu\_user* untuk menampilkan menu pengguna biasa. Modul *termcolor* digunakan agar teks di terminal dapat berwarna sehingga tampilan lebih menarik. Modul *pyfiglet* digunakan untuk menampilkan teks bergaya ASCII art.

### Source Code:

```
from auth import register, login
from admin import admin_menu
from user import menu_user
from termcolor import colored
import pyfiglet
```

## 2. Fitur di File main

Isi file main ini adalah tampilan utama yang akan muncul saat program di running di terminal yang berisi pilihan register, login, dan out(keluar program).

### Source Code:

```
while True:
    judul = pyfiglet.figlet_format("WELCOME", font="slant")
    print(colored(judul, "yellow"))
    print(colored(
    """
=====
| 1. REGISTER      |
| 2. LOGIN         |
| 3. OUT           |
=====
    """, "cyan"))
    menu_utama = input(colored("Pilih menu 1-3 : ", "green"))
    if menu_utama == "1":
        os.system('cls' if os.name == 'nt' else 'clear')
        print(colored("SILAHKAN REGISTER", "green"))
        register()
    elif menu_utama == "2":
        os.system('cls' if os.name == 'nt' else 'clear')
        user_type = login()
        if user_type == "admin":
            admin_menu()
        elif user_type == "user":
            menu_user()
    elif menu_utama == "3":
        os.system('cls' if os.name == 'nt' else 'clear')
        print(colored("Anda telah keluar dari program", "green"))
        break
```

```
        else:
            os.system('cls' if os.name == 'nt' else 'clear')
            print(colored("Pilihan tidak valid, silahkan coba lagi.", "red"))
```

## B. File auth

### 1. Fungsi Liblary(File auth)

Modul `login_data` berisi variabel `data_login` yang menyimpan informasi akun pengguna seperti username, password, dan tipe pengguna (admin atau user). Sedangkan modul `termcolor` menyediakan fungsi `colored()` yang digunakan untuk memberi warna pada teks di terminal.

**Source Code:**

```
from login_data import data_login
from termcolor import colored
```

### 2. Fitur File auth

Isi utama dari file auth ini adalah di mana fungsi untuk fitur login, register, dan logout yang akan di panggil nantinya di file [main.py](#) atau menu utama untuk melakukan proses register,login, maupun logout.

**Source Code:**

```
def register():
    regis_username = input("Masukkan username :")
    regis_password = input("Masukkan password :")
    new_user = len(data_login) + 1
    data_login[new_user] = {"username": regis_username, "password": regis_password, "tipe": "user"}
    print("Anda berhasil register")

def login():
    from login_data import status_login
    os.system('cls' if os.name == 'nt' else 'clear')
```

```

print("LOGIN")
input_username = input("Masukkan username :")
input_password = input("Masukkan password :")

for value in data_login.values():
    if input_username == value["username"] and input_password == value["password"]:
        print(colored(f" Login berhasil sebagai {value['tipe']}", "green"))
        status_login = True
        return value["tipe"]

def logout():
    from login_data import status_login
    status_login = False
    print(colored("Anda berhasil logout", "green"))

```

## C. File data

Fungsi file ini berguna untuk menyimpan data login admin dan user dan juga untuk menyimpan daftar film yang telah di tambahkan beserta tahun rilisnya.

Source Code:

```

def tambah_film(nama, tahun):
    global daftar_film
    daftar_film.update({nama: tahun})
    return f"Film {nama} dengan tahun rilis {tahun} berhasil ditambahkan"

```

## D. File crud

### 1. Fungsi Liblary (File crud)

Modul prettytable digunakan untuk membuat tabel yang rapi di terminal melalui kelas PrettyTable, sehingga data film dapat ditampilkan dalam format yang terstruktur dan mudah dibaca. Sedangkan modul data berisi variabel daftar\_film, yaitu tempat penyimpanan utama data film beserta tahun rilisnya.

### Source Code :

```
from prettytable import PrettyTable
from data import daftar_film
```

## 2. Fitur File crud

Fungsi file crud ini adalah membuat fitur proses CRUD dari program yang di buat di mana ada fungsi tampilkan\_film yang untuk menampilkan keseluruhan daftar film dan tahun rilis, fungsi tambah\_film untuk menambahkan film, fungsi mengubah\_film untuk mengedit daftar film dan tahun rilis yang telah ditambahkan, fungsi hapus\_film untuk menghapus film yang telah di tambahkan ke daftar film, dan ada fungsi logout untuk keluar dari program.

### Source Code :

```
def tampilkan_film():
    global daftar_film
    table = PrettyTable()
    table.field_names = ["No", "Nama Film", "Tahun Rilis"]
    for i, (NamaFilm, TahunRilis) in enumerate(daftar_film.items(), start=1):
        table.add_row([i, NamaFilm, TahunRilis])
    print(table)

def tambah_film(nama, tahun):
    global daftar_film
    daftar_film.update({nama: tahun})
    return f"Film {nama} dengan tahun rilis {tahun} berhasil ditambahkan"

def mengubah_film(nama_lama, nama_baru, tahun_baru):
    global daftar_film
    tampilkan_film
    del daftar_film[nama_lama]
    daftar_film.update({nama_baru: tahun_baru})
    return f"Film {nama_lama} berhasil di ubah menjadi {nama_baru} dengan tahun rilis {tahun_baru}"

def hapus_film():
    global daftar_film
    i = 0
```

```

tampilkan_film()
try:
    film_ke_nomor = {}
    for NamaFilm, TahunRilis in daftar_film.items():
        film_ke_nomor[i] = NamaFilm
        i +=1
    pilih_film = int(input(f"Pilih film yang ingin dihapus (1-{len(daftar_film)}): "))
    nama_lama = film_ke_nomor[pilih_film]
    del daftar_film[nama_lama]
    print(f"film {nama_lama.title()} berhasil di hapus")
except KeyError:
    print("Pilihan tidak valid")

```

## E. File admin

### 1. Fungsi Liblary (File admin)

Modul crud berisi fungsi tampilkan\_film, tambah\_film, mengubah\_film, dan hapus\_film yang masing-masing digunakan untuk menampilkan, menambah, mengedit, dan menghapus data film. Modul auth menyediakan fungsi logout untuk keluar dari sesi pengguna. Modul termcolor digunakan untuk memberi warna pada teks di terminal agar tampilan lebih menarik. Terakhir, modul data berisi variabel daftar\_film yang menyimpan data utama berupa nama film dan tahun rilisnya.

**Source Code :**

```

from crud import tampilkan_film, tambah_film, mengubah_film,
hapus_film
from auth import logout
from termcolor import colored
from data import daftar_film

```

### 2. Fitur File Admin

Fitur dari file admin ini adalah untuk menampilkan menu admin di saat pengguna login sebagai admin dan menampilkan menu admin yang dapat menambah,melihat, mengedit, dan menghapus data daftar film.

## Source Code :

```
def admin_menu():
    while True:
        print(colored("""
=====
| 1. TAMBAH FILM & TAHUN RILIS |
| 2. LIHAT FILM & TAHUN RILIS |
| 3. MENGUBAH NAMA FILM / TAHUN RILIS |
| 4. HAPUS FILM / TAHUN RILIS |
| 5. LOGOUT |
=====
""", "cyan"))
        menu_admin = input(colored("Pilih menu 1-5 : ", "green"))
        if menu_admin == "1":
            os.system('cls' if os.name == 'nt' else 'clear')
            try:
                print("TAMBAH FILM & TAHUN RILIS")
                nama_film = input("Masukkan nama film :").strip().title()
                tahun_rilis = input("Masukkan tahun rilis :")
                print(tambah_film(nama_film, tahun_rilis))
            except Exception as e:
                print(f"Terjadi kesalahan: {e}")

        elif menu_admin == "2":
            os.system('cls' if os.name == 'nt' else 'clear')
            print("LIHAT FILM & TAHUN RILIS")
            tampilan_film()

        elif menu_admin == "3":
            os.system('cls' if os.name == 'nt' else 'clear')
            try:
                i = 0
                tampilan_film()
                film_ke_nomor = {}
                for NamaFilm, TahunRilis in daftar_film.items():
                    film_ke_nomor[i] = NamaFilm
                    i +=1
                pilih_film = int(input(f"Pilih film yang ingin di
ubah (1-{len(daftar_film)}): ")) - 1
                nama_lama = film_ke_nomor[pilih_film]
                nama_baru = input("Masukkan nama film baru
:").strip().title()
                tahun_baru = int(input("Masukkan tahun rilis baru
:"))
                print(mengubah_film(nama_lama, nama_baru,
                tahun_baru))
            except ValueError:
                print("Masukkan angka yang benar")
```

```
tahun_baru))
        except KeyError:
            print("Pilihan tidak valid")
        except ValueError:
            print("Tahun rilis harus berupa angka!")

    elif menu_admin == "4":
        os.system('cls' if os.name == 'nt' else 'clear')
        print("HAPUS FILM / TAHUN RILIS")
        hapus_film()

    elif menu_admin == "5":
        os.system('cls' if os.name == 'nt' else 'clear')
        logout()
        break

    else:
        os.system('cls' if os.name == 'nt' else 'clear')
        print("Pilihan tidak valid, silahkan coba lagi.")
```

## F. File user

### 1. Fungsi Liblary (File user)

Modul crud berisi fungsi tampilan\_film yang digunakan untuk menampilkan daftar film yang tersimpan. Modul auth menyediakan fungsi logout untuk keluar dari sesi pengguna. Sedangkan modul termcolor digunakan untuk memberi warna pada teks di terminal agar tampilan program lebih menarik dan mudah dibaca.

**Source Code :**

```
from crud import tampilan_film
from auth import logout
from termcolor import colored
```

## 2. Fitur File User

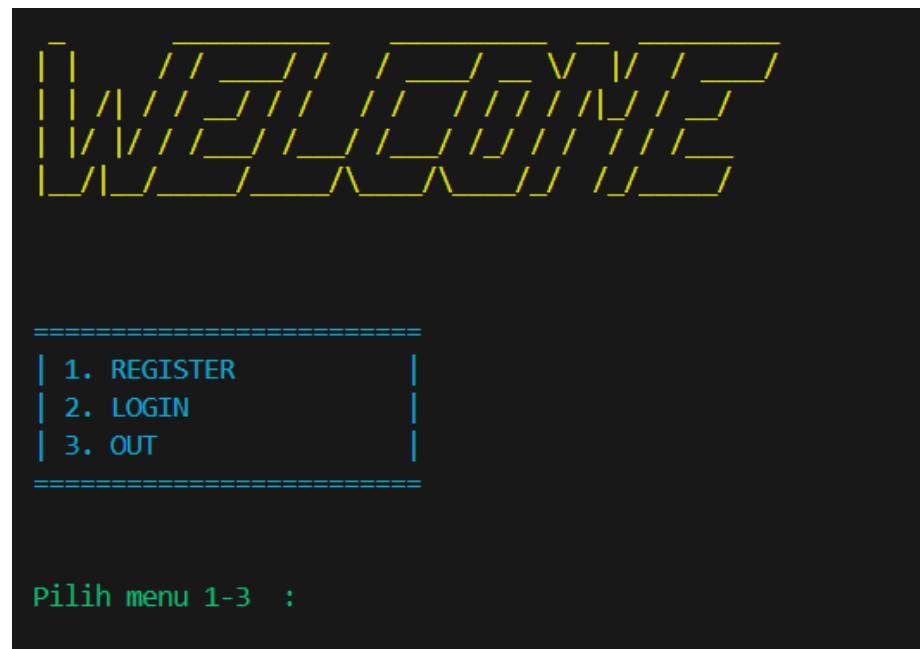
Fungsi dari file user ini adalah untuk membuat fitur menu user jika pengguna login sebagai user dan akan menampilkan menu user yang hanya bisa mengakses melihat daftar nama film dan tahun rilis saja.

**Source Code :**

```
def menu_user():
    while True:
        print(colored("""
=====
| 1. LIHAT FILM & TAHUN RILIS      |
| 2. LOGOUT                         |
=====
""", "cyan"))
        pilihan = input(colored("Pilih menu 1-2 : ", "green"))
        os.system('cls' if os.name == 'nt' else 'clear')

        if pilihan == "1":
            tampilkan_film()
        elif pilihan == "2":
            logout()
            break
        else:
            print(colored("Pilihan hanya 1-2!", "red"))
```

#### 4. Hasil Output



Gambar 4.1 tampilan menu utama program

```
LOGIN
Masukkan username :admin
Masukkan password :adminfilm22
Login berhasil sebagai admin

=====
| 1. TAMBAH FILM & TAHUN RILIS |
| 2. LIHAT FILM & TAHUN RILIS  |
| 3. MENGUBAH NAMA FILM / TAHUN RILIS |
| 4. HAPUS FILM / TAHUN RILIS    |
| 5. LOGOUT                      |
=====

Pilih menu 1-5 : |
```

Gambar 4.2 tampilan menu admin

```
TAMBAH FILM & TAHUN RILIS
Masukkan nama film :gampang cuan
Masukkan tahun rilis :2023
Film Gampang Cuan dengan tahun rilis 2023 berhasil ditambahkan

=====
| 1. TAMBAH FILM & TAHUN RILIS      |
| 2. LIHAT FILM & TAHUN RILIS      |
| 3. MENGUBAH NAMA FILM / TAHUN RILIS |
| 4. HAPUS FILM / TAHUN RILIS       |
| 5. LOGOUT                         |
=====

Pilih menu 1-5 : |
```

Gambar 4.3 tampilan menambah film

```
LIHAT FILM & TAHUN RILIS
+---+-----+-----+
| No | Nama Film | Tahun Rilis |
+---+-----+-----+
| 1 | Gampang Cuan | 2023 |
| 2 | Laura | 2024 |
| 3 | Samsara | 2024 |
| 4 | Jumbo | 2025 |
| 5 | Pabrik Gula | 2025 |
+---+-----+-----+

=====
| 1. TAMBAH FILM & TAHUN RILIS      |
| 2. LIHAT FILM & TAHUN RILIS      |
| 3. MENGUBAH NAMA FILM / TAHUN RILIS |
| 4. HAPUS FILM / TAHUN RILIS       |
| 5. LOGOUT                         |
=====

Pilih menu 1-5 : |
```

Gambar 4.4 tampilan melihat daftar film

```
+---+-----+-----+
| No | Nama Film | Tahun Rilis |
+---+-----+-----+
| 1 | Gampang Cuan | 2023 |
| 2 | Laura | 2024 |
| 3 | Samsara | 2024 |
| 4 | Jumbo | 2025 |
| 5 | Pabrik Gula | 2025 |
+---+-----+
Pilih film yang ingin di ubah (1-5): 4
Masukkan nama film baru :sugar factory
Masukkan tahun rilis baru :2025
Film Jumbo berhasil di ubah menjadi Sugar Factory dengan tahun rilis 2025

=====
| 1. TAMBAH FILM & TAHUN RILIS |
| 2. LIHAT FILM & TAHUN RILIS |
| 3. MENGUBAH NAMA FILM / TAHUN RILIS |
| 4. HAPUS FILM / TAHUN RILIS |
| 5. LOGOUT |
=====

Pilih menu 1-5 :|
```

Gambar 4.5 tampilan mengedit daftar film

```
HAPUS FILM / TAHUN RILIS
+---+-----+-----+
| No | Nama Film | Tahun Rilis |
+---+-----+-----+
| 1 | Gampang Cuan | 2023 |
| 2 | Laura | 2024 |
| 3 | Samsara | 2024 |
| 4 | Pabrik Gula | 2025 |
| 5 | Sugar Factory | 2025 |
+---+-----+
Pilih film yang ingin dihapus (1-5): 3
film Samsara berhasil di hapus

=====
| 1. TAMBAH FILM & TAHUN RILIS |
| 2. LIHAT FILM & TAHUN RILIS |
| 3. MENGUBAH NAMA FILM / TAHUN RILIS |
| 4. HAPUS FILM / TAHUN RILIS |
| 5. LOGOUT |
=====

Pilih menu 1-5 :|
```

Gambar 4.6 tampilan menghapus daftar film

```
LOGIN
Masukkan username :user
Masukkan password :user123
Login berhasil sebagai user

=====
| 1. LIHAT FILM & TAHUN RILIS |
| 2. LOGOUT
=====

Pilih menu 1-2 : |
```

Gambar 4.7 tampilan menu user

No	Nama Film	Tahun Rilis
1	Gampang Cuan	2023
2	Laura	2024
3	Pabrik Gula	2025
4	Sugar Factory	2025

```
=====
| 1. LIHAT FILM & TAHUN RILIS |
| 2. LOGOUT
=====

Pilih menu 1-2 : |
```

Gambar 4.8 tampilan menu user lihat daftar film

```
SILAHKAN REGISTER
Masukkan username :new
Masukkan password :gweh1234
Anda berhasil register
```



---

- | 1. REGISTER
- | 2. LOGIN
- | 3. OUT

---

Pilih menu 1-3 : |

*Gambar 4.9 tampilan register*

## **5. Langkah-langkah GIT**

## 5.1 GIT Add

Membuat file kita menjadi ke staging area sebelum benar-benar di simpan ke repository kita.

```
PS D:\praktikum-apd> git add .  
PS D:\praktikum-apd>
```

## 5.2 GIT Commit

Git Commit berfungsi untuk menyimpan perubahan yang telah kita lakukan pada file kita yang dari staging area dan biasanya berisi pesan singkat sebagai catatan perubahan apa yang kita lakukan.

```
PS D:\praktikum-apd> git commit -m "membuat post-test-8"
[main 98ad8f6] membuat post-test-8
 7 files changed, 275 insertions(+)
create mode 100644 kelas/pertemuan-8/pertemuan-8.py
create mode 100644 post-test/post-test-apd-8/2509106118-MuhammadHajarAswad-PT-8/admin.py
create mode 100644 post-test/post-test-apd-8/2509106118-MuhammadHajarAswad-PT-8/auth.py
create mode 100644 post-test/post-test-apd-8/2509106118-MuhammadHajarAswad-PT-8/crud.py
create mode 100644 post-test/post-test-apd-8/2509106118-MuhammadHajarAswad-PT-8/data.py
create mode 100644 post-test/post-test-apd-8/2509106118-MuhammadHajarAswad-PT-8/main.py
create mode 100644 post-test/post-test-apd-8/2509106118-MuhammadHajarAswad-PT-8/user.py
```

### 5.3 GIT Push

Git Push berfungsi untuk mengupload perubahan kita dari repository lokal kita agar tersimpan online di akun github kita.

```
PS D:\praktikum-apd> git push
Enumerating objects: 23, done.
Counting objects: 100% (23/23), done.
Delta compression using up to 12 threads
Compressing objects: 100% (19/19), done.
Writing objects: 100% (20/20), 4.44 KiB | 649.00 KiB/s, done.
Total 20 (delta 5), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (5/5), completed with 2 local objects.
To https://github.com/Aswad129/praktikum-apd.git
  bc09ef4..e2ced99  main -> main
PS D:\praktikum-apd>
```