



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODE DAN PERANCANGAN APLIKASI

3.1 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Studi Literatur

Pada tahap ini, dicari referensi mengenai teori-teori pendukung penelitian, seperti *Hybrid Filtering*, API, sistem rekomendasi, dan KNN. Referensi yang dibutuhkan dapat ditemukan dari berbagai sumber, seperti buku, jurnal, hingga situs *website*.

2. Analisis Sistem

Analisis sistem dilakukan dengan tujuan untuk mencari tahu kebutuhan dari sistem yang akan dibangun. Pada tahap ini, akan ditentukan bagaimana sistem akan dibangun, menyesuaikan dengan hasil analisis sistem yang telah dilakukan.

3. Perancangan dan Pembuatan Program

Pada tahap ini, dibuat rancangan dari aplikasi yang akan dibuat, menyesuaikan dengan hasil analisis sistem. Rancangan yang dibuat terdiri dari beberapa macam model, seperti *sitemap*, *flowchart*, Data Flow Diagram (DFD), Entity Relationship Diagram (ERD), struktur tabel, dan rancangan antarmuka. Setelah rancangan aplikasi selesai dibuat, maka proses berlanjut ke tahapan pembuatan aplikasi. Proses pembuatan aplikasi meliputi pembuatan *database*, tampilan antarmuka, proses koneksi dengan TMDb, dan sistem aplikasi secara keseluruhan.

4. Uji Coba dan Perbaikan Program

Pada tahap ini, dilakukan proses uji coba untuk mengetahui kekurangan dari aplikasi yang telah dibuat dan memperbaiki setiap kekurangan yang ditemukan.

5. Evaluasi

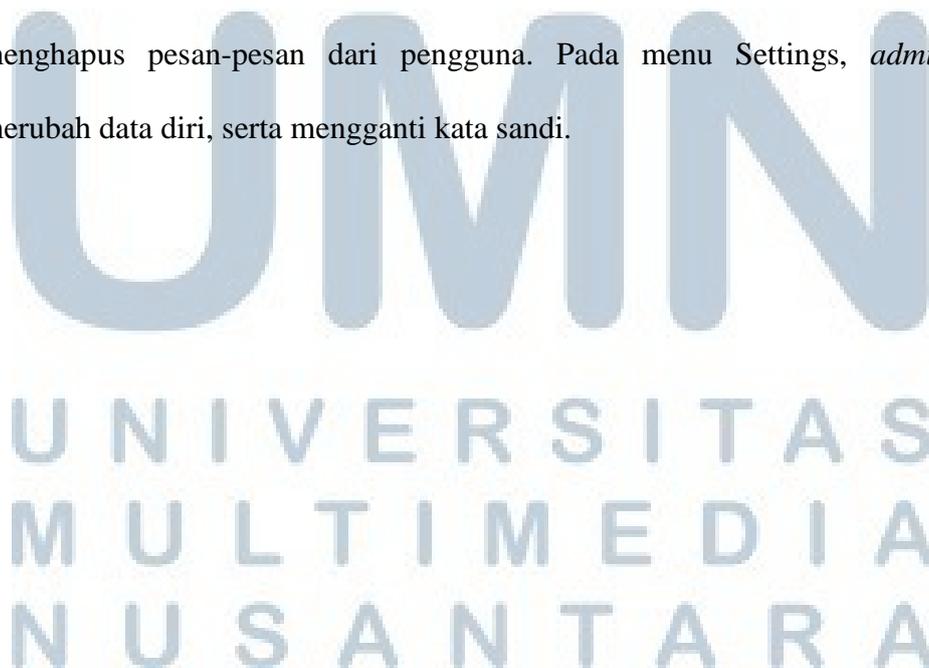
Proses evaluasi aplikasi dilakukan dengan menggunakan kuesioner untuk mengetahui tingkat kepuasan pengguna, dan juga untuk mengetahui tingkat akurasi dari rekomendasi yang telah dihasilkan oleh aplikasi. Singarimbun dan Efendi (1995) mengatakan bahwa jumlah minimal uji coba kuesioner adalah minimal 30 responden. Karena objek penelitian yang digunakan sebagai sampel tidak memiliki kriteria tertentu, digunakanlah teknik *Simple Random Sampling* untuk mengambil sampel. *Simple Random Sampling* adalah suatu teknik penentuan sampel di mana setiap anggota populasi memiliki peluang yang sama untuk dijadikan sampel (Kriyantono, 2009).

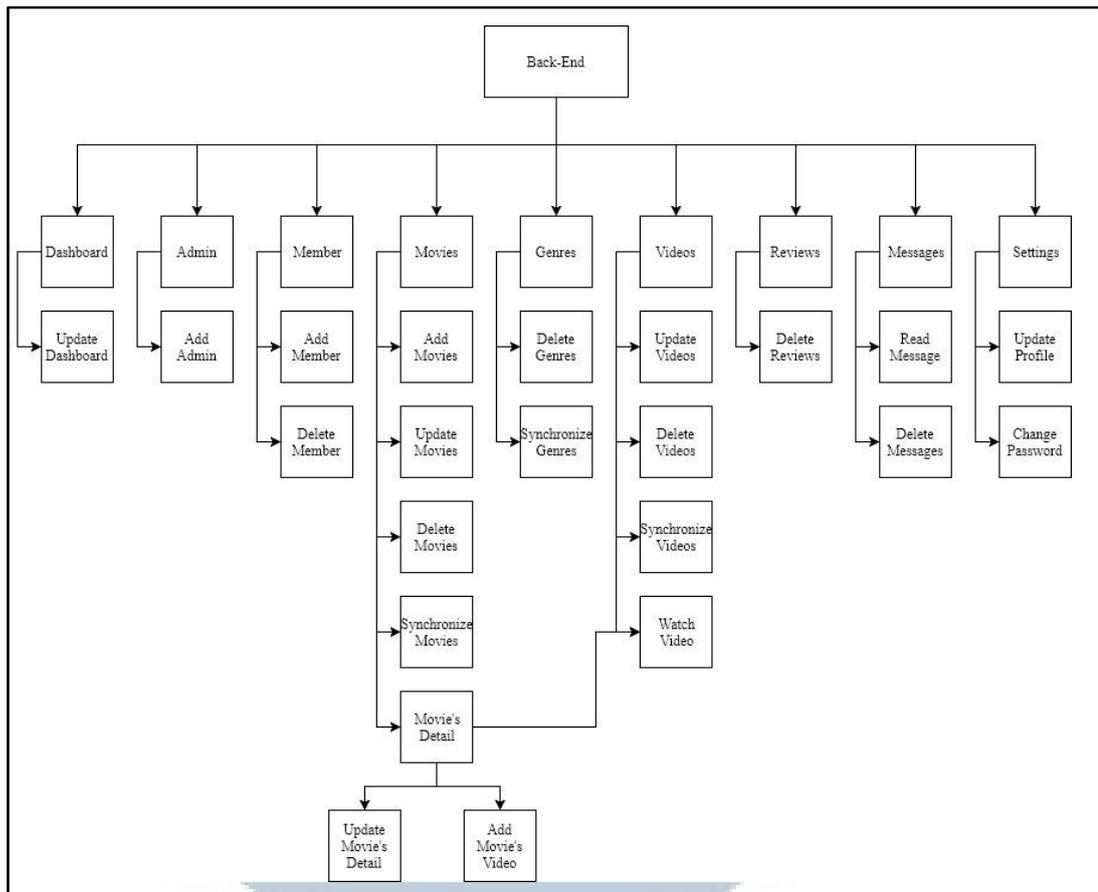
3.2 Perancangan Aplikasi

Proses perancangan aplikasi dilakukan dengan menggunakan beberapa macam model perancangan sistem, seperti *sitemap* untuk menggambarkan hirarki sistem, *Data Flow Diagram* (DFD) yang dibuat untuk menggambarkan aliran data antar proses dalam aplikasi. *Flowchart* yang berfungsi untuk menggambarkan alur kerja dari aplikasi yang telah dibuat, *Entity Relationship Diagram* (ERD) yang berfungsi untuk menggambarkan relasi antara tabel-tabel yang digunakan dalam aplikasi, rancangan antarmuka yang menggambarkan tampilan aplikasi, dan terakhir struktur tabel, yang berisi penjelasan mengenai setiap tabel yang digunakan dalam aplikasi.

3.2.1 Sitemap

Gambar 3.1 menunjukkan *sitemap* dari sistem pada sisi *admin*. Terdapat 9 menu utama, yaitu menu Dashboard, Admin, Member, Movies, Genres, Videos, Reviews, Messages dan Settings. Pada menu Dashboard, *admin* dapat melakukan proses *update* data dashboard. Pada menu Admin, *admin* dapat menambahkan *admin* baru pada sistem, sedangkan pada menu Member, *admin* dapat menambahkan *member* serta menghapus *member* yang ada. Pada menu Movies, *admin* dapat menambahkan film, merubah data film, menghapus film, hingga melakukan proses sinkronisasi film dengan TMDb API. Pada menu Genres, *admin* hanya dapat menghapus *genre* yang ada atau melakukan proses sinkronisasi *genre* dengan TMDb API. Pada menu Videos, *admin* dapat mengubah data video, menghapus video yang ada, hingga melakukan sinkronisasi video dengan TMDb API. Pada menu Reviews, *admin* dapat menghapus ulasan-ulasan pada film yang diberikan oleh pengguna. Pada menu Messages, *admin* dapat melihat pesan yang telah dikirimkan oleh pengguna dan juga dapat menghapus pesan-pesan dari pengguna. Pada menu Settings, *admin* dapat merubah data diri, serta mengganti kata sandi.

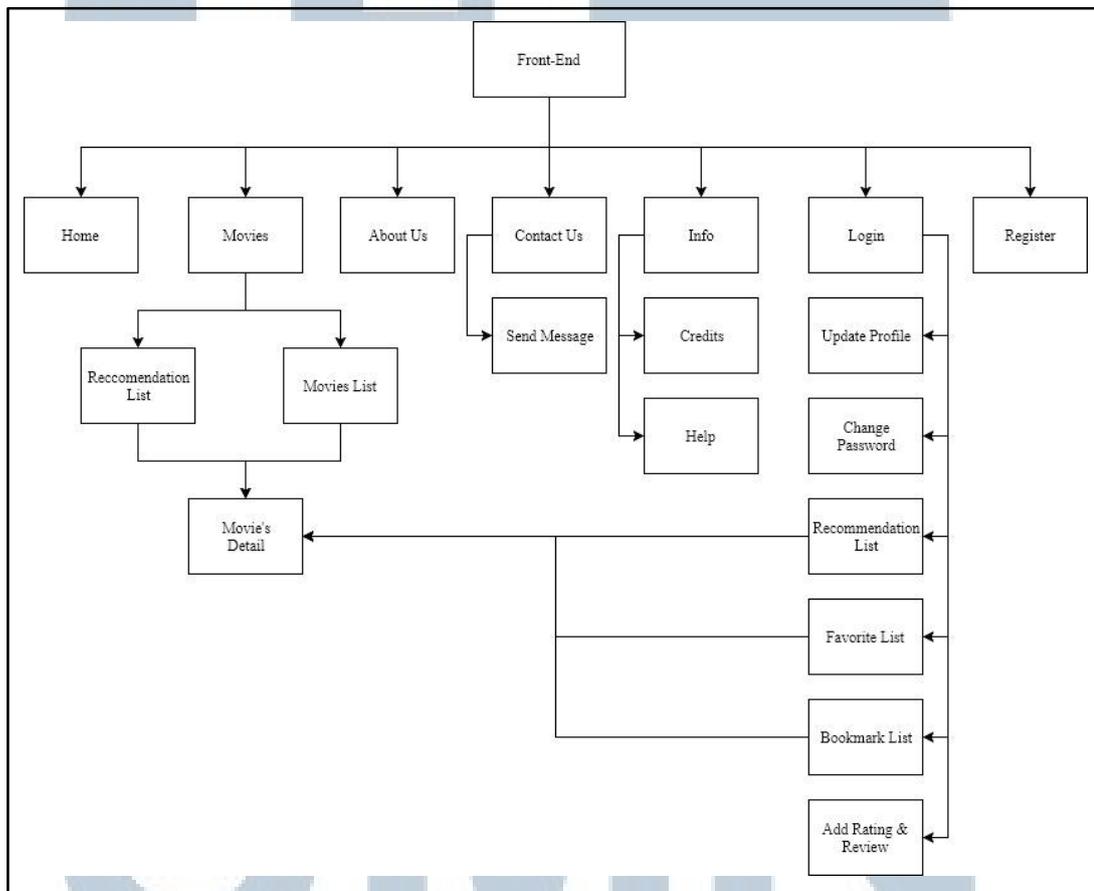




Gambar 3.1 Sitemap Sistem Sisi Back-End

Gambar 3.2 menunjukkan sitemap dari sistem pada sisi *user*. Terdapat 7 menu utama yang dapat diakses oleh *user*, yaitu menu Home, Movies, About Us, Contact Us, Info, Login dan Register. Pada menu Movies, *user* dapat melihat daftar film yang ada pada sistem, serta melihat daftar rekomendasi film yang diberikan oleh sistem. Pada menu About Us, *user* dapat melihat informasi singkat mengenai aplikasi yang telah dibangun. Pada menu Contact Us, *user* dapat mengirimkan pesan baik berupa saran maupun kritik kepada sistem. Pada menu Info, terdapat dua submenu, yaitu halaman Credits dan halaman Help. Pada halaman Credits, *user* dapat melihat daftar *framework* yang digunakan dalam membangun aplikasi, sedangkan pada halaman Help, *user* dapat melihat daftar panduan untuk membantu *user* dalam menggunakan aplikasi. Menu Login

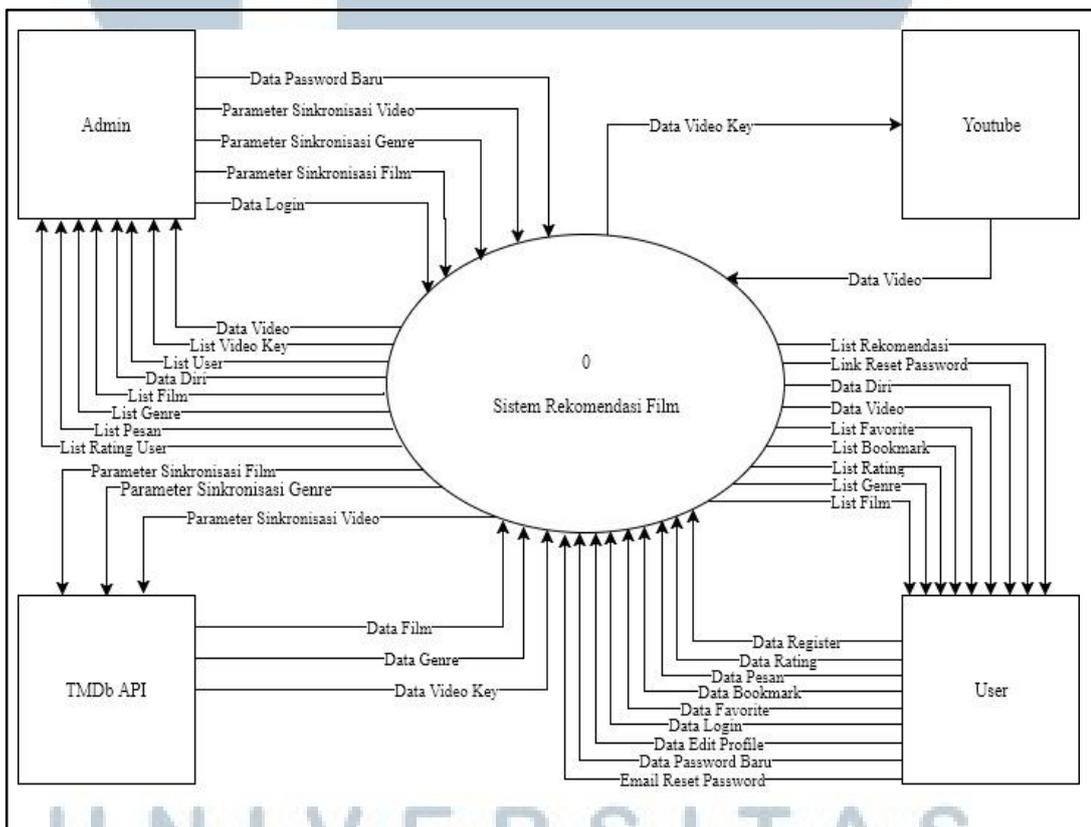
digunakan oleh *user* untuk mengakses sistem sebagai *member* yang terdaftar. Dengan melakukan login, *user* dapat menambahkan film-film yang ada ke dalam daftar *bookmark* atau daftar *favorite*, serta menambahkan *rating* maupun *review* terhadap film-film di dalam sistem. Setelah melakukan proses login, *user* dapat mengubah profil dan mengubah kata sandi jika diperlukan. Menu Register digunakan saat *user* ingin mendaftar menjadi *member* pada aplikasi.



Gambar 3.2 Sitemap Sistem Sisi Front-End

3.2.2 Data Flow Diagram

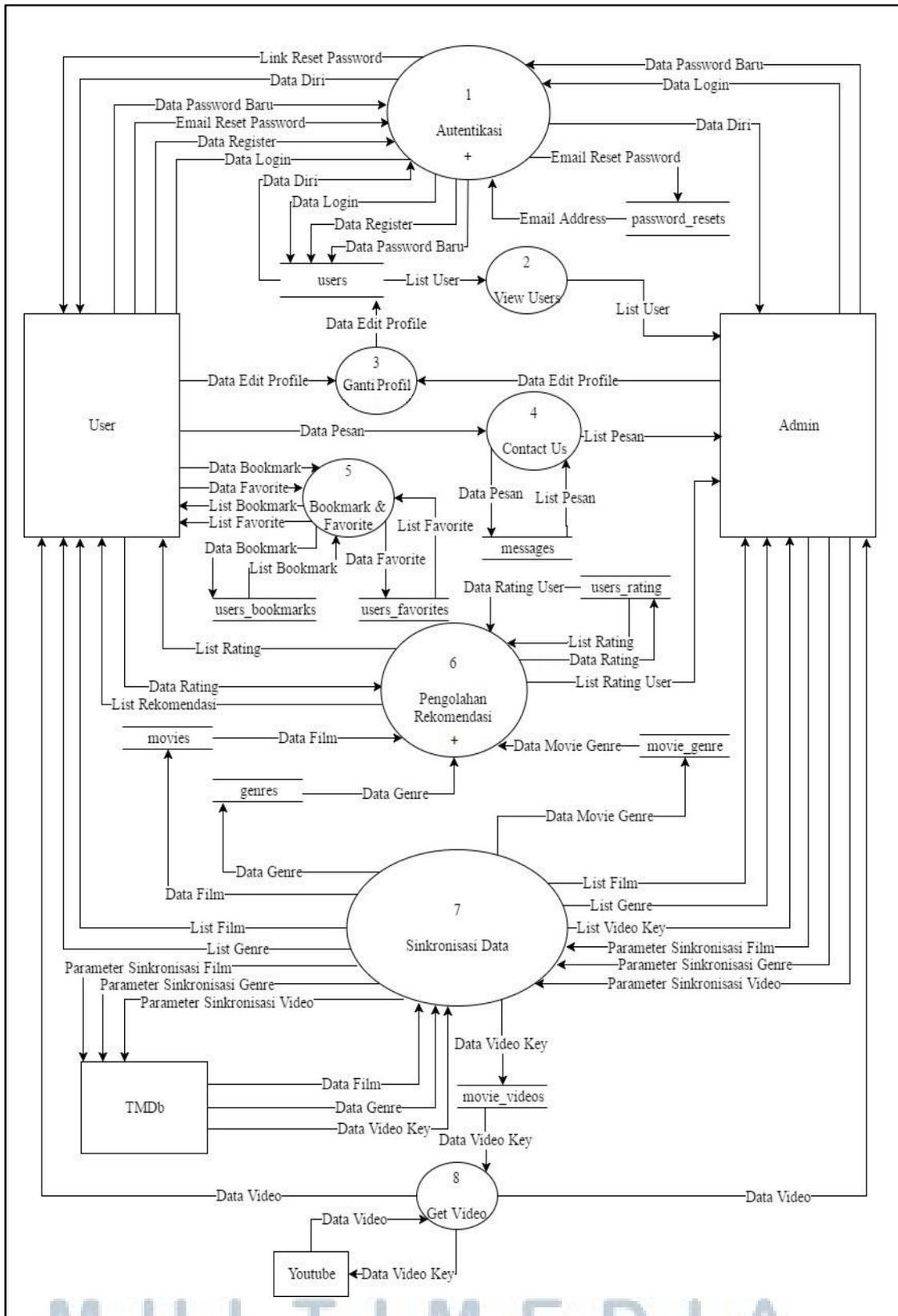
Gambar 3.3 menunjukkan konteks diagram dari aplikasi yang dibangun. Diagram ini menggambarkan alur data pada aplikasi yang dibuat. Terdapat empat entitas dalam aplikasi yang dibangun, yaitu *admin*, *user*, Youtube, dan TMDb API. *User* adalah pengguna yang menggunakan sistem rekomendasi film yang dibangun. *Admin* adalah pengelola aplikasi yang bertugas untuk mengelola data-data yang dibutuhkan dalam aplikasi. TMDb API memiliki peran sebagai penyedia data-data berkaitan dengan film, sedangkan Youtube berperan sebagai penyedia video yang dibutuhkan di dalam aplikasi.



Gambar 3.3 Konteks Diagram

Gambar 3.4 merupakan *Data Flow Diagram* (DFD) level 1 dari aplikasi rekomendasi film yang dibangun. Terdapat delapan proses, yaitu sebagai berikut.

1. Proses Autentikasi, yaitu proses yang berhubungan dengan akun pengguna, seperti proses *login*, proses *register*, dan proses lainnya yang akan dijelaskan lebih lanjut pada DFD level 2.
2. Proses *View Users*, yaitu proses di mana *admin* melihat daftar *user* yang terdaftar pada aplikasi.
3. Proses Ganti Profil, yaitu proses pergantian profil pada akun *user* maupun *admin*.
4. Proses Contact Us, yaitu proses di mana *user* dapat mengirimkan pesan kepada *admin*, baik berupa saran maupun kritik. Pesan yang dikirimkan akan disimpan pada tabel *messages*.
5. Proses *Bookmark & Favorite*, yaitu proses penambahan atau penghapusan sebuah film oleh *user* dari daftar *bookmark* atau daftar *favorite user* tersebut.
6. Proses Pengolahan Rekomendasi, yaitu proses di mana rekomendasi film dihasilkan, yang akan dijelaskan lebih lanjut pada DFD level 2.
7. Proses Sinkronisasi Data, yaitu proses pengumpulan data dari TMDb API untuk selanjutnya digunakan di dalam aplikasi. Pada proses ini, didapatkan tiga data yaitu data *video key*, data film dan data genre yang selanjutnya disimpan pada tabel masing-masing. Untuk menjalankan proses sinkronisasi data, diperlukan parameter sinkronisasi yang didapatkan dari entitas *admin*.
8. Proses *Get Video*, yaitu proses mendapatkan video dari Youtube berdasarkan *video key* yang didapatkan dari proses Sinkronisasi Data.



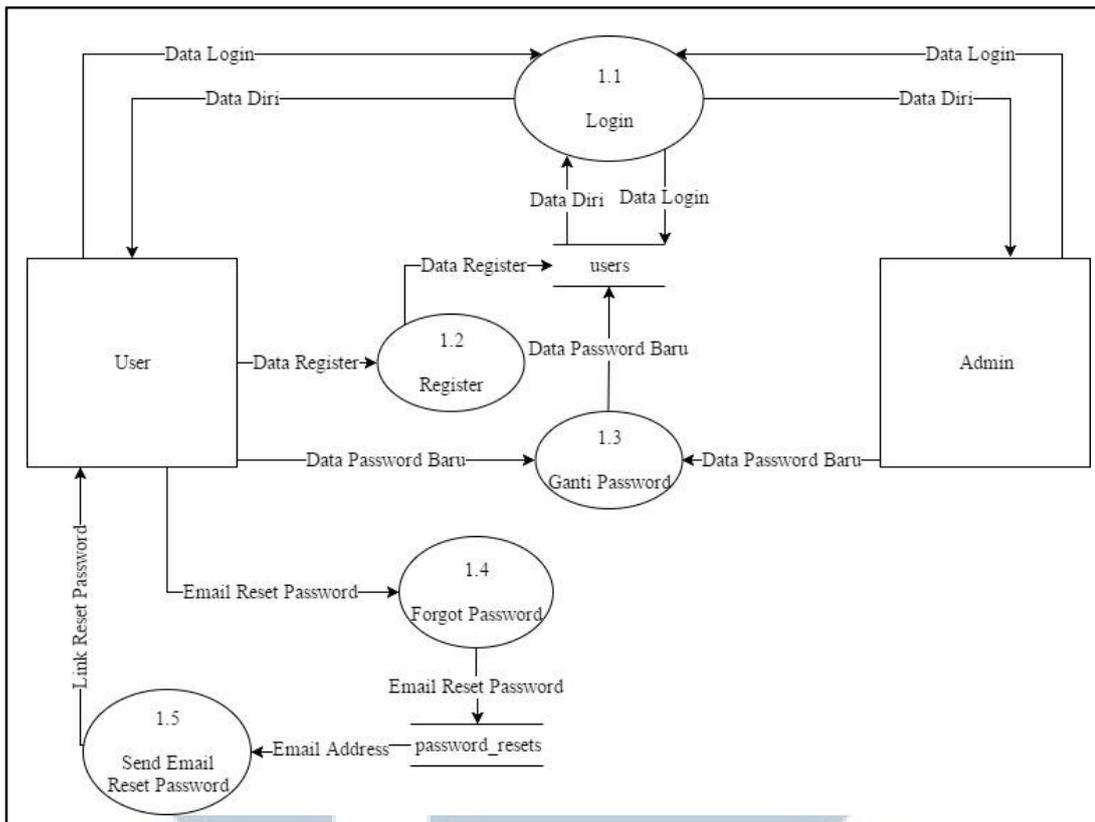
Gambar 3.4 Data Flow Diagram (DFD) Level 1

MULTIMEDIA
NUSANTARA

Gambar 3.5 merupakan DFD level 2 sub-proses Autentikasi. Proses Autentikasi memiliki 5 proses, yaitu.

1. Proses *Login*, yaitu proses yang membutuhkan data *login* berupa alamat *email* dan kata sandi dari *user* atau *admin*, yang kemudian akan dicek pada tabel *users*. Ketika *login* berhasil, maka *user* atau *admin* akan mendapatkan data diri masing-masing.
2. Proses *Register*, yaitu proses di mana *user* mendaftarkan diri pada aplikasi dengan mengirimkan data *register*, yang selanjutnya akan disimpan pada tabel *users*.
3. Proses Ganti Password, yaitu proses di mana *user* maupun *admin* dapat mengganti *password* masing-masing dengan yang baru.
4. Proses *Forgot Password*, yaitu proses yang dijalankan ketika *user* tidak mengingat kata sandi dari akun miliknya, sehingga *user* diminta untuk menginput alamat *email* dari akun tersebut, yang selanjutnya akan disimpan pada tabel *password_resets*.
5. Proses *Send Email Reset Password*, yaitu proses di mana sistem akan mengirimkan *email* berisi *link* untuk proses pengaturan ulang kata sandi. *Email* akan dikirimkan ke alamat *email* yang didapatkan dari tabel *password_resets*.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

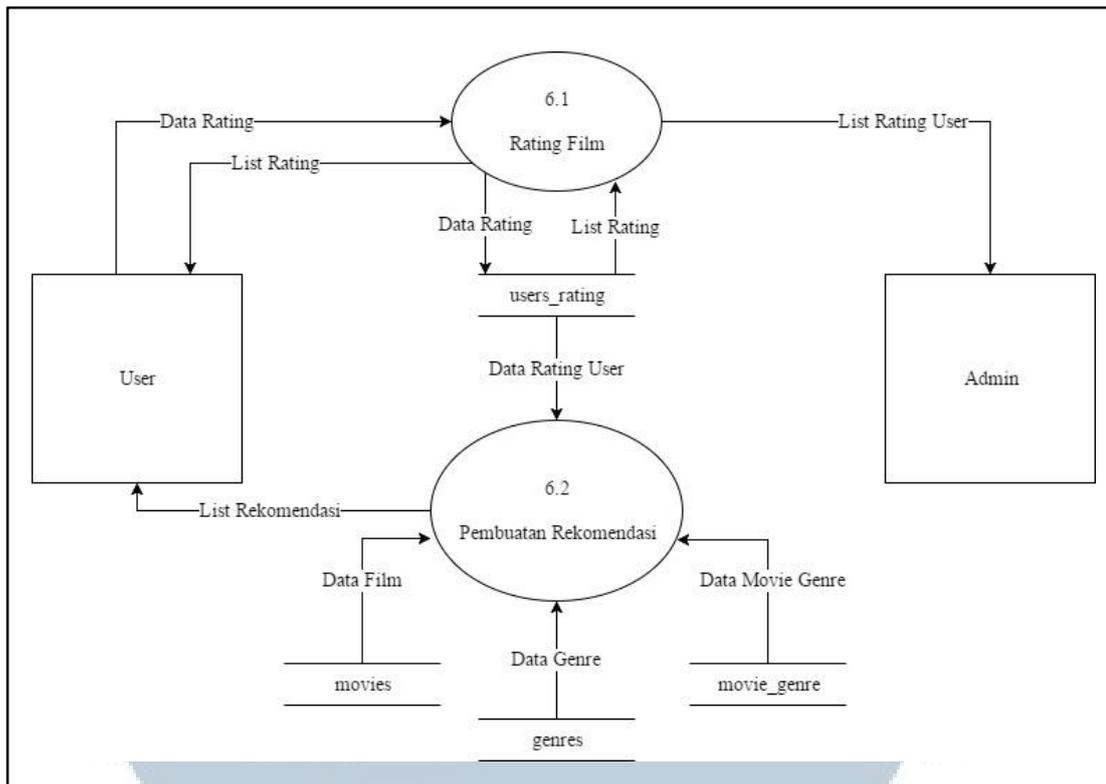


Gambar 3.5 Data Flow Diagram Level 2 Sub-Proses Autentikasi

Gambar 3.6 merupakan DFD level 2 sub-proses Pengolahan Rekomendasi.

Pada proses ini, terdapat dua proses detail, yaitu.

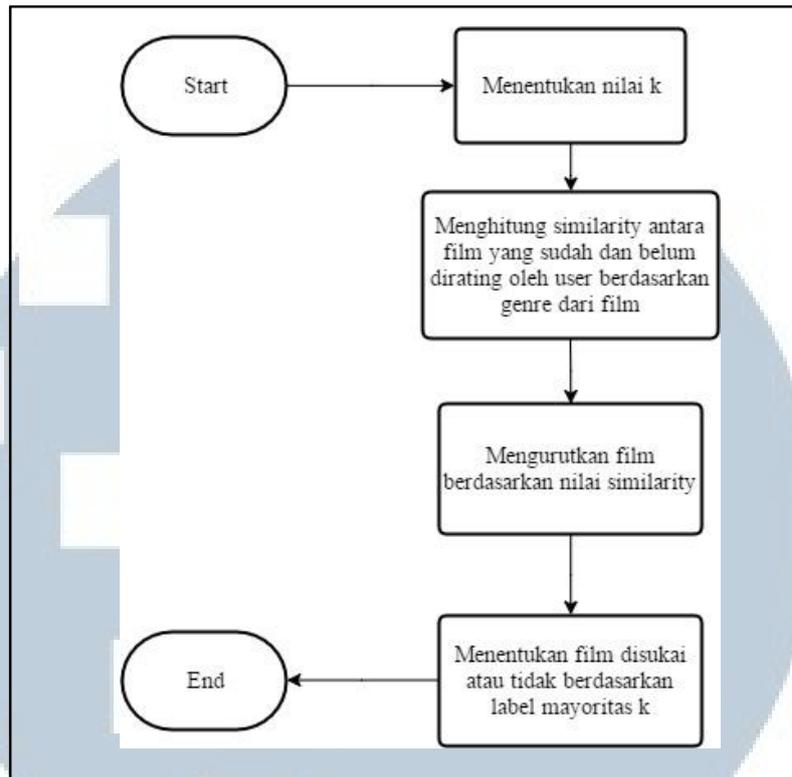
1. Proses *Rating* Film, yaitu proses dimana *user* memberikan *rating* (penilaian) terhadap suatu film, yang selanjutnya data *rating* tersebut akan disimpan pada tabel *users_rating*.
2. Proses Pembuatan Rekomendasi, yaitu proses di mana daftar film yang direkomendasikan akan dibuat untuk ditampilkan kepada *user*. Proses ini membutuhkan data film dari tabel *movies*, data *genre* dari tabel *genres*, data *movie_genre* dari tabel *movie_genre*, dan data *rating user* dari tabel *users_rating*.



Gambar 3.6 Data Flow Diagram Level 2 Sub-Proses Pengolahan Rekomendasi

3.2.3 Flowchart

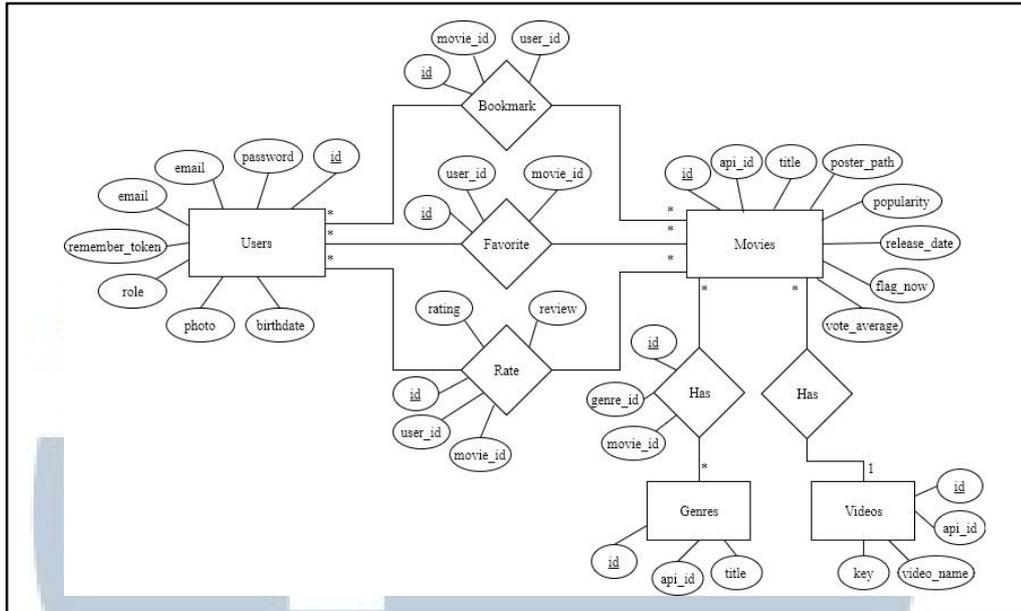
Gambar 3.7 merupakan *flowchart* yang menggambarkan alur proses dalam algoritma K-Nearest Neighbor yang digunakan dalam aplikasi rekomendasi film yang dibangun. Proses dimulai dengan menentukan nilai k , yaitu jumlah tetangga yang akan diperhitungkan nantinya. Selanjutnya, proses dilanjutkan dengan menghitung *similarity* antara film yang merupakan *training* data dengan film yang merupakan *testing* data. Proses perhitungan *similarity* menggunakan rumus *cosine similarity*. Setelah menghitung *similarity*, proses dilanjutkan dengan mengurutkan film berdasarkan nilai *similarity*-nya, kemudian menentukan apakah film tersebut termasuk ke dalam golongan yang disukai oleh *user* atau tidak. Proses dilakukan secara berulang, sehingga ditemukan beberapa film yang dapat direkomendasikan kepada *user*.



Gambar 3.7 Flowchart Algoritma K-Nearest Neighbor

3.2.4 Entity Relationship Diagram

Entity Relationship Diagram (ERD) dibuat dengan tujuan untuk memudahkan proses pemodelan *database*. Gambar 3.8 menunjukkan ERD dari aplikasi rekomendasi film yang dibuat. Seorang *user* dapat menambahkan lebih dari satu film ke dalam daftar *bookmark* ataupun daftar *favorite*, dan sebuah film dapat ditambahkan ke banyak daftar *bookmark* maupun daftar *favorite*. Selain itu, *user* juga dapat memberikan *rating* (penilaian) dan *review* terhadap banyak film, dan sebuah film juga dapat di-*rating* atau di-*review* oleh banyak *user*. Sebuah film dapat memiliki lebih dari satu *genre*, dan sebuah *genre* dapat ditemukan pada banyak film. Sebuah film juga dapat memiliki lebih dari satu video, namun sebuah video hanya dapat dimiliki oleh sebuah film.



Gambar 3.8 Entity Relationship Diagram

3.2.5 Struktur Tabel

Berikut struktur tabel yang digunakan dalam sistem rekomendasi film yang dibangun.

1. Nama Tabel : users

Fungsi : menampung data personal milik *user*

Primary Key : id

Foreign Key : -

Tabel 3.1 Struktur Tabel users

No	Nama Kolom	Tipe Data	Keterangan
1	id	int (10)	Identifier pengguna
2	name	varchar (255)	Nama pengguna
3	email	varchar (255)	Email pengguna
4	birthdate	Date	Tanggal lahir pengguna
5	password	varchar (255)	Kata sandi pengguna
6	role	varchar (100)	Peran pengguna
7	photo	varchar (255)	Foto pengguna

Tabel 3.1 Struktur Tabel users (lanjutan)

No	Nama Kolom	Tipe Data	Keterangan
8	remember_token	varchar	Token pengguna untuk login
9	created_at	timestamp	Field penampung waktu proses <i>insert</i>
10	updated_at	timestamp	Field penampung waktu proses <i>update</i>

2. Nama Tabel : users_bookmarks

Fungsi : menampung daftar *bookmark* dari semua *user*

Primary Key : id

Foreign Key : user_id, movie_id

Tabel 3.2 Struktur Tabel users_bookmarks

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier <i>bookmark</i>
2	user_id	int(11)	Identifier pengguna
3	movie_id	int(11)	Identifier film
4	created_at	timestamp	Field penampung waktu proses <i>insert</i>
5	updated_at	timestamp	Field penampung waktu proses <i>update</i>

3. Nama Tabel : users_favorites

Fungsi : menampung daftar *favorite* dari semua *user*

Primary Key : id

Foreign Key : user_id, movie_id

Tabel 3.3 Struktur Tabel users_favorites

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier <i>favorite</i>
2	user_id	int(11)	Identifier pengguna
3	movie_id	int(11)	Identifier film
4	created_at	timestamp	Field penampung waktu proses <i>insert</i>
5	updated_at	timestamp	Field penampung waktu proses <i>update</i>

4. Nama Tabel : users_rating
 Fungsi Tabel : menampung *rating* dan *review user* terhadap sebuah film
 Primary Key : id
 Foreign Key : user_id, movie_id

Tabel 3.4 Struktur Tabel users_rating

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier <i>rating</i>
2	user_id	int(11)	Identifier pengguna
3	movie_id	int(11)	Identifier film
4	rating	float	<i>Rating</i> dari user
5	review	text	<i>Review</i> dari user
6	created_at	timestamp	Field penampung waktu proses <i>insert</i>
7	updated_at	timestamp	Field penampung waktu proses <i>update</i>

5. Nama Tabel : movies
 Fungsi Tabel : menampung data film
 Primary Key : id
 Foreign Key : -

Tabel 3.5 Struktur Tabel movies

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier film
2	api_id	int(11)	Identifier film dari TMDb API
3	title	varchar(255)	Judul film
4	release_date	date	Tanggal rilis dari film
5	poster_path	varchar(255)	<i>Path</i> poster dari film
6	synopsis	varchar(1000)	Deskripsi singkat mengenai film
7	flag_now	tinyint(1)	<i>Flag</i> film sedang tayang atau tidak
8	vote_average	float	Rata-rata vote film (<i>rating</i>)
9	popularity	float	Tingkat popularitas film
10	created_at	timestamp	Field penampung waktu proses <i>insert</i>
11	updated_at	timestamp	Field penampung waktu proses <i>update</i>

6. Nama Tabel : movie_genre
- Fungsi : Menampung daftar *genre* dari setiap film
- Primary Key : id
- Foreign Key : movie_id, genre_id

Tabel 3.6 Struktur Tabel movie_genre

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier <i>genre</i> dari film
2	movie_id	int(11)	Identifier film
3	genre_id	int(11)	Identifier <i>genre</i>
4	created_at	timestamp	Field penampung waktu proses <i>insert</i>
5	updated_at	timestamp	Field penampung waktu proses <i>update</i>

7. Nama Tabel : movie_videos
- Fungsi : Menampung daftar video dari setiap film
- Primary Key : id
- Foreign Key : movie_id

Tabel 3.7 Struktur Tabel movie_videos

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier video
2	api_id	int(11)	Identifier video dari TMDb API
3	movie_id	int(11)	Identifier film
4	video_name	varchar(255)	Judul dari video
5	key	varchar(255)	Youtube <i>key</i> dari video
6	created_at	timestamp	Field penampung waktu proses <i>insert</i>
7	updated_at	timestamp	Field penampung waktu proses <i>update</i>

8. Nama Tabel : *genres*

Fungsi : menampung daftar *genre* yang ada pada aplikasi

Primary Key : id

Foreign Key : -

Tabel 3.8 Struktur Tabel *genres*

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier <i>genre</i>
2	api_id	int(11)	Identifier <i>genre</i> dari TMDb API
3	name	varchar(255)	Nama <i>genre</i>
4	created_at	timestamp	Field penampung waktu proses <i>insert</i>
5	updated_at	timestamp	Field penampung waktu proses <i>update</i>

9. Nama Tabel : *messages*

Fungsi : menampung pesan dari *user*

Primary Key : id

Foreign Key : -

Tabel 3.9 Struktur Tabel *messages*

No	Nama Kolom	Tipe Data	Keterangan
1	id	int(11)	Identifier pesan
2	name	varchar(255)	Nama pengirim pesan
3	email	varchar(255)	Alamat <i>email</i> pengirim pesan
4	subject	varchar(255)	Judul dari pesan yang dikirimkan
5	message	text	Isi dari pesan yang dikirimkan
6	flagread	tinyint(1)	<i>Flag</i> pesan sudah dibaca atau belum
7	created_at	timestamp	Field penampung waktu proses <i>insert</i>
8	updated_at	timestamp	Field penampung waktu proses <i>update</i>

10. Nama Tabel : password_resets
- Fungsi : menampung daftar *user* yang melakukan proses pengaturan ulang kata sandi.
- Primary Key : -
- Foreign Key : -

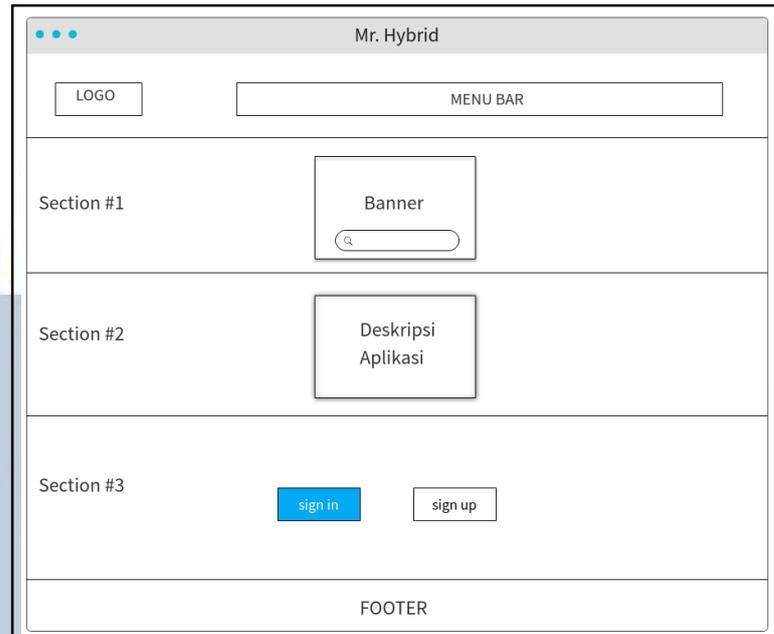
Tabel 3.10 Struktur Tabel password_resets

No	Nama Kolom	Tipe Data	Keterangan
1	email	varchar(255)	Email pengguna yang melakukan <i>reset password</i>
2	token	varchar(255)	Token unik untuk proses <i>reset password</i>
3	created_at	timestamp	Field penampung waktu proses <i>insert</i>

3.2.6 Rancangan Antarmuka

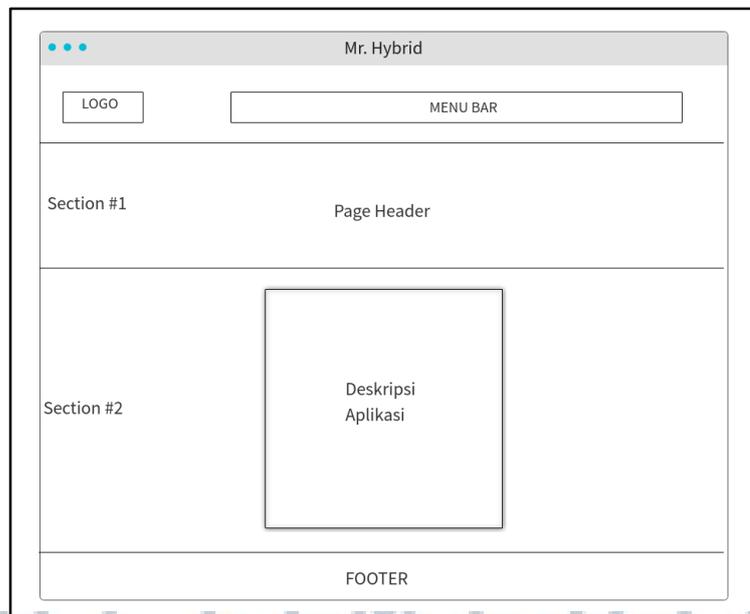
Setiap halaman dalam aplikasi tersusun atas *header*, konten dan *footer*. *Header* dan *footer* pada setiap halaman sama, di mana *header* berisi logo aplikasi dan *menubar*, sedangkan *footer* berisi kalimat *copyright*.

Gambar 3.9 menunjukkan rancangan antarmuka dari halaman *Home* (beranda). Pada halaman ini, tepat di bawah *header*, terdapat sebuah *banner* yang berisi nama aplikasi serta deskripsi singkat, dilengkapi dengan sebuah *input box* untuk melakukan proses pencarian film. Di bawah *banner* tersebut, terdapat sebuah *section* yang berisi ulasan singkat mengenai sistem yang dibangun, dilengkapi dengan sebuah *button* yang jika diklik akan mengarah ke halaman *About Us*. Di bawah *section* tersebut, terdapat sebuah *section* yang berisi *button* untuk *login* dan *button* untuk *sign-up*, kemudian halaman diakhiri dengan *footer*.



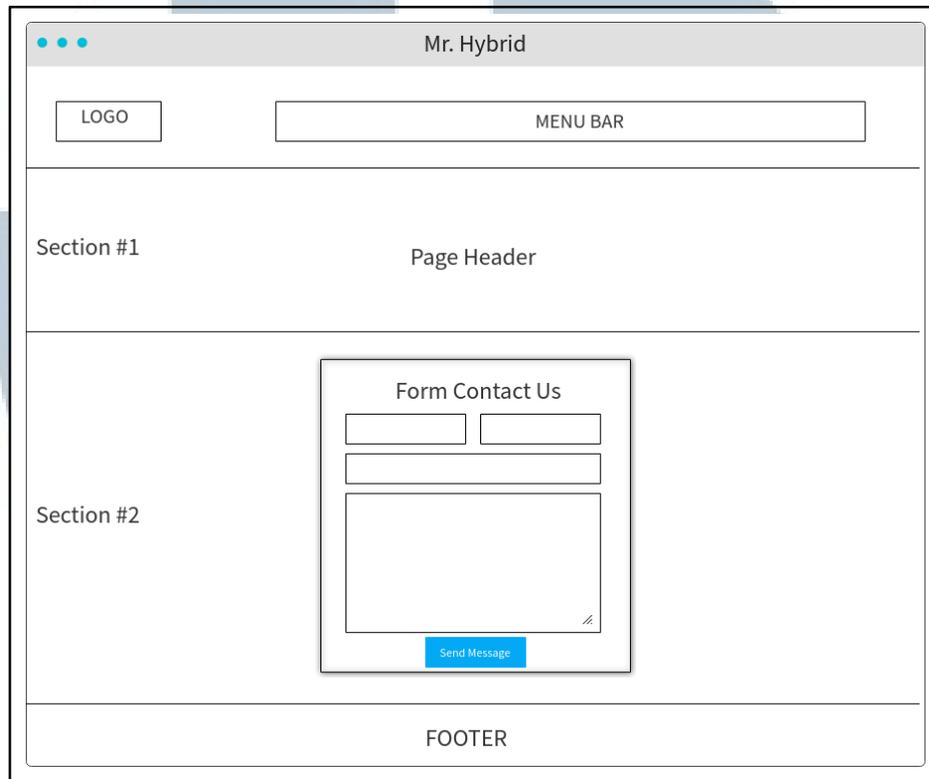
Gambar 3.9 Rancangan Antarmuka Halaman Home

Gambar 3.10 merupakan rancangan antarmuka pada halaman *About Us*. Pada halaman ini, terdapat informasi mengenai sistem rekomendasi film yang dibangun, baik informasi tentang aplikasinya, maupun informasi tentang pihak di balik proses pembuatannya.



Gambar 3.10 Rancangan Antarmuka Halaman About Us

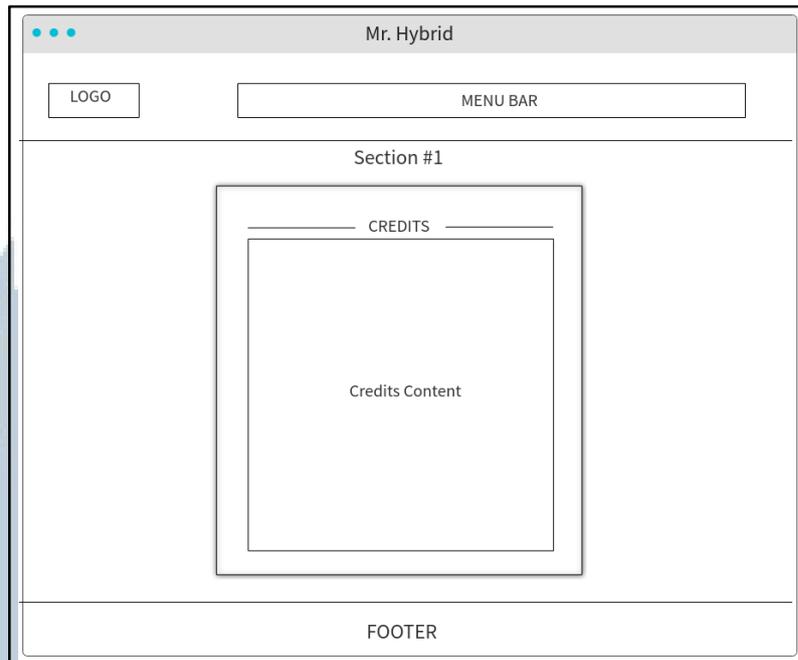
Gambar 3.11 merupakan rancangan antarmuka halaman *Contact Us*. Pada halaman ini, *user* dapat mengirimkan pesan kepada *admin* dengan mengisi formulir yang tersedia, lalu menekan tombol *submit* ketika sudah selesai.



Gambar 3.11 Rancangan Antarmuka Halaman Contact Us

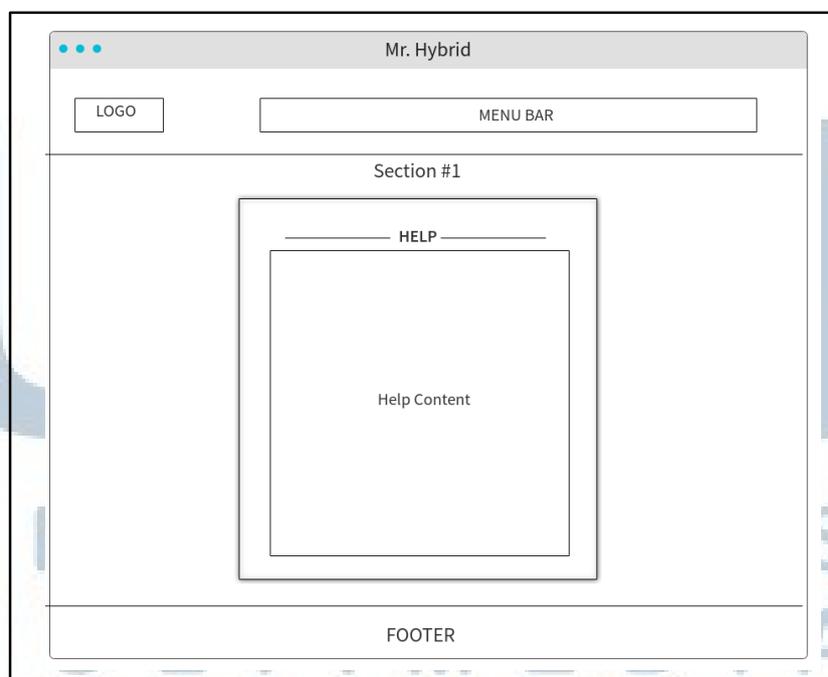
Gambar 3.12 menunjukkan rancangan antarmuka pada halaman *Credits*. Halaman ini merupakan halaman yang berisi daftar pihak-pihak yang membantu proses pembuatan sistem rekomendasi ini, meskipun secara tidak langsung, contohnya adalah TMDb API sebagai penyedia data film, PHP sebagai bahasa pemrograman, dan lainnya.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



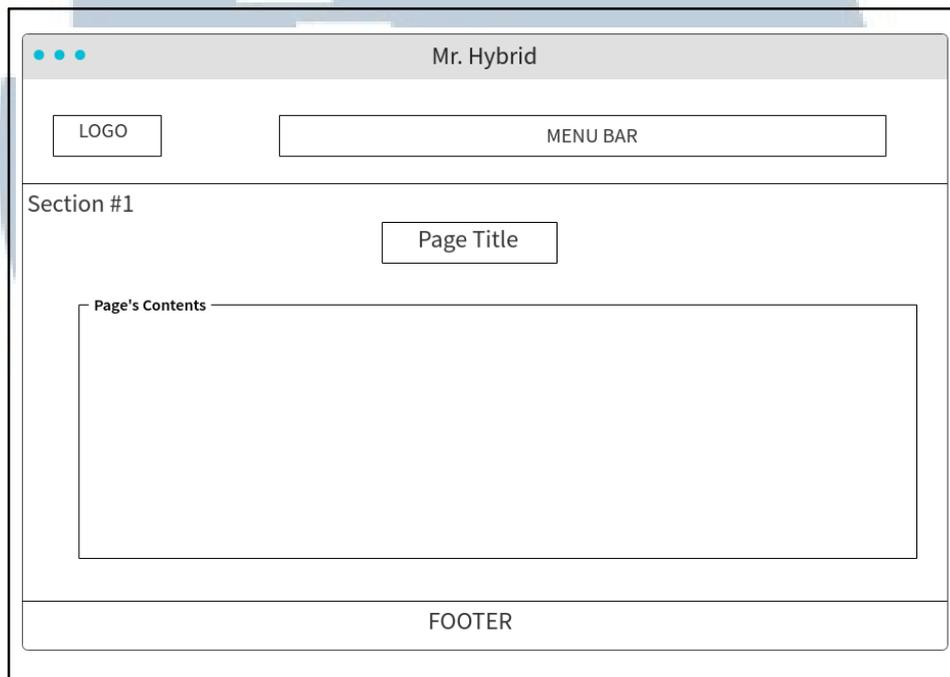
Gambar 3.12 Rancangan Antarmuka Halaman Credits

Gambar 3.13 menunjukkan rancang antarmuka dari halaman *Help*. Pada halaman ini, terdapat beberapa panduan bagi *user* untuk membantu *user* dalam memahami cara penggunaan sistem rekomendasi yang telah dibangun.



Gambar 3.13 Rancangan Antarmuka Halaman Help

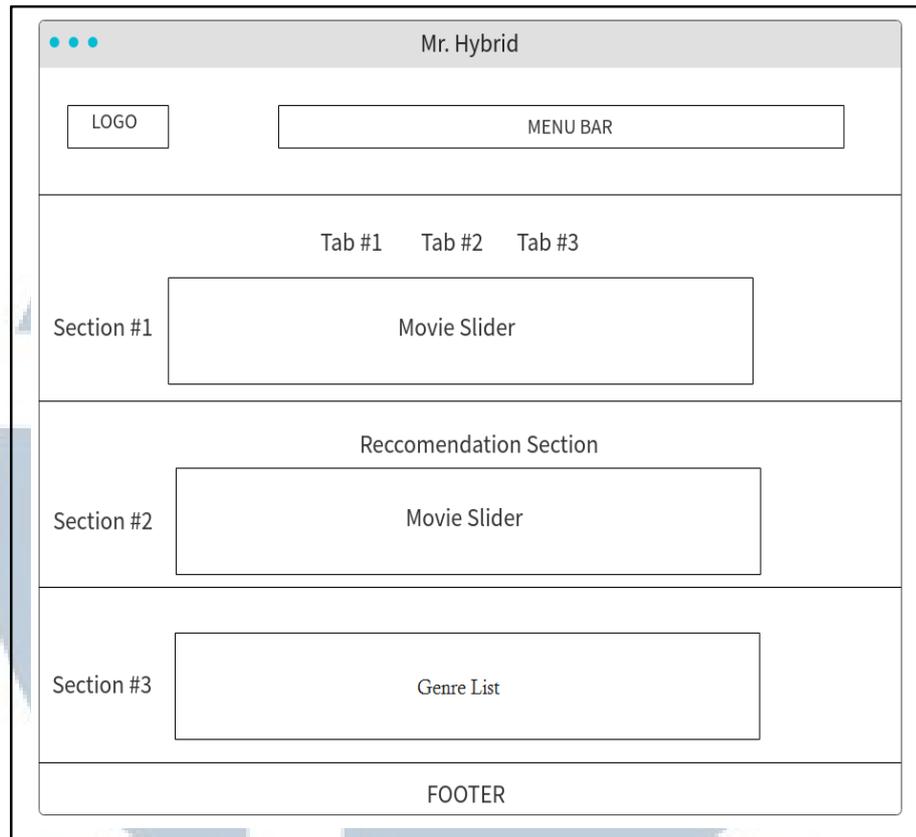
Gambar 3.14 merupakan rancangan antarmuka halaman umum (*general*) yang digunakan pada beberapa halaman, seperti halaman hasil pencarian dan halaman daftar film berdasarkan genre. Pada halaman ini, film yang ditampilkan sebanyak delapan film setiap halaman, dan untuk melihat film lainnya, *user* dapat mengganti halaman dengan menekan *button* yang tersedia di bawah daftar film.



Gambar 3.14 Rancangan Antarmuka Halaman Umum (*General*)

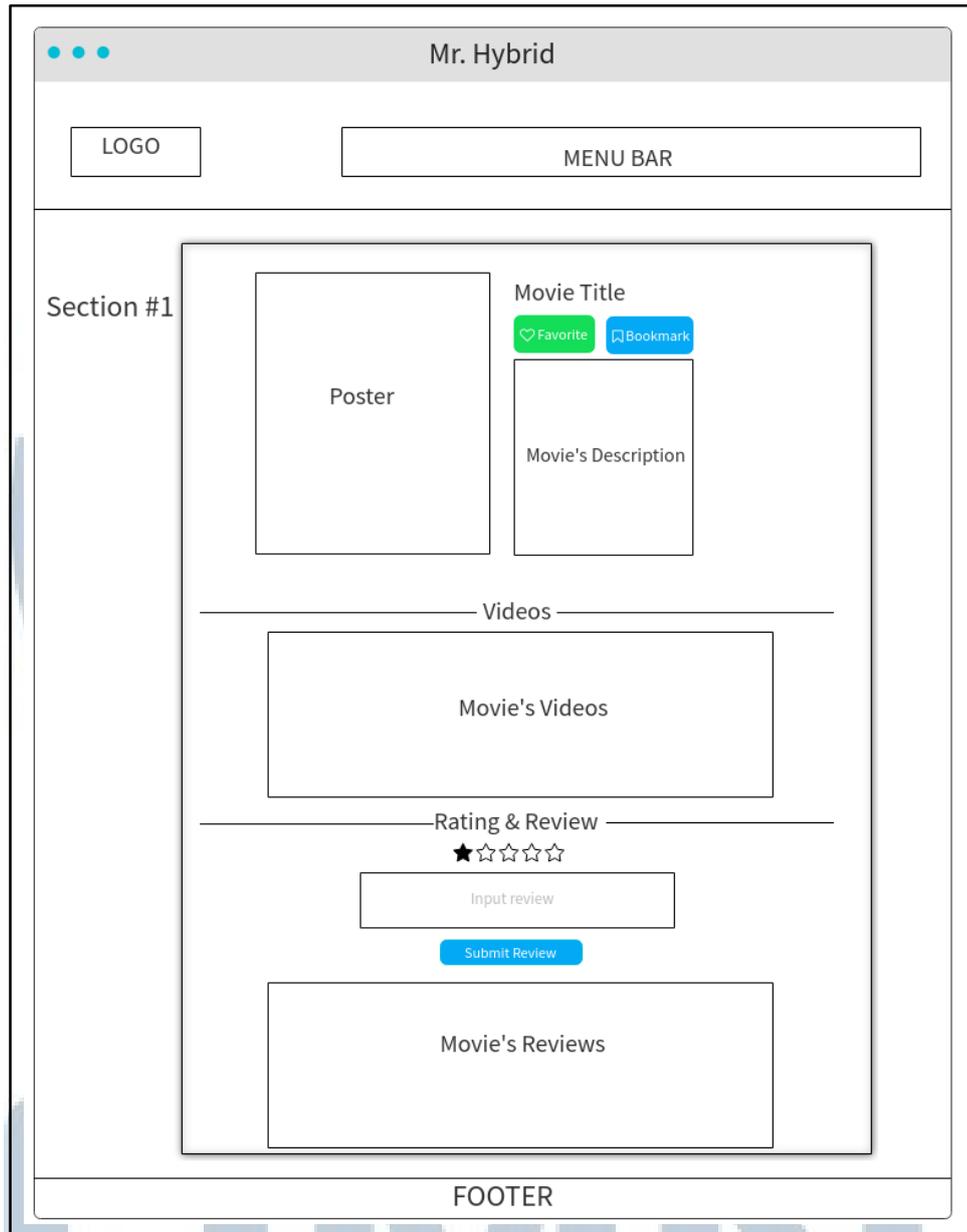
Gambar 3.15 merupakan rancangan antarmuka halaman *Movies*. Pada bagian atas halaman ini, terdapat *banner* yang berupa *slider*. Pada halaman ini, *user* dapat mencari film berdasarkan beberapa aspek, seperti tingkat popularitas, tanggal rilis terbaru, *genre* dari film, dan film yang sedang tayang saat ini. Selain itu, pada halaman ini juga terdapat *list* film yang direkomendasikan kepada *user*.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



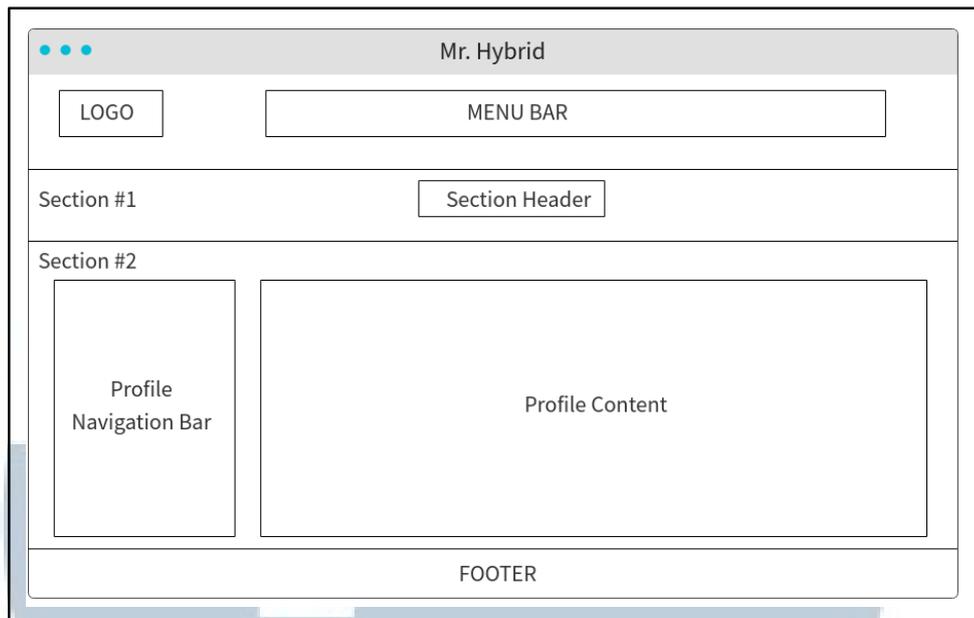
Gambar 3.15 Rancangan Antarmuka Halaman Movies

Gambar 3.16 merupakan rancangan antarmuka halaman *Movie Detail*. Pada halaman ini, *user* dapat menemukan informasi detail dari sebuah film, seperti judul film, *genre* dari film, tanggal rilis, *rating*, serta sinopsis. Selain informasi, *user* juga dapat melihat satu atau lebih video yang berkaitan dengan film, seperti video *trailer*. Pada halaman ini juga *user* dapat memberikan *rating* serta ulasan terhadap suatu film, dengan mengisi kolom yang ada dan menekan *button* yang tersedia ketika sudah selesai. Ulasan maupun *rating* yang telah diberikan dapat diubah dengan menekan tombol *edit* yang tersedia. Ulasan dari *user* lain juga dapat dilihat pada halaman ini. Selain memberikan *rating* ataupun ulasan, *user* juga dapat menambahkan film ke dalam daftar *bookmark* atau daftar *favorite*, dengan menekan *button* yang tersedia di bawah judul film.



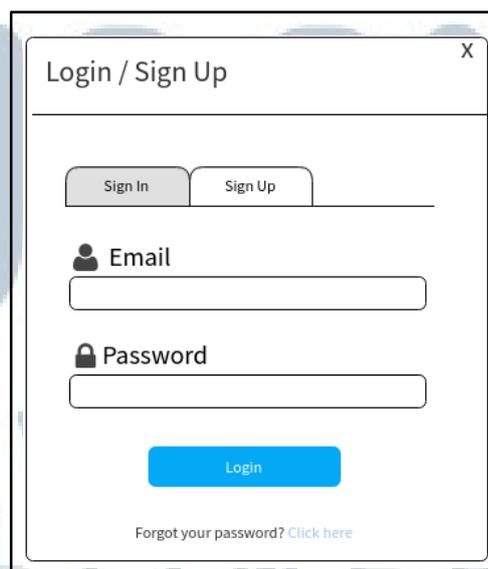
Gambar 3.16 Rancangan Antarmuka Halaman Movie Detail

Gambar 3.17 menunjukkan rancangan antarmuka halaman *Profile*. Halaman ini hanya dapat diakses setelah *user* mendaftar dan *login* ke dalam sistem. Pada halaman ini, *user* dapat mengubah profil dan mengubah kata sandi dengan menekan *button* yang tersedia. Selain itu, pada halaman ini *user* dapat melihat daftar *bookmark* dan daftar *favorite user* tersebut, serta melihat daftar rekomendasi yang telah disediakan oleh sistem.



Gambar 3.17 Rancangan Antarmuka Halaman Profile

Gambar 3.18 menunjukkan rancangan antarmuka pada modal (*pop up window*) yang muncul saat *user* mengklik *button login*. Modal memiliki dua *tab*, *tab* pertama untuk proses *login* ke dalam sistem, dan *tab* ke 2 yang ditunjukkan oleh Gambar 3.19, digunakan untuk proses mendaftar. Pada *tab login*, *user* juga dapat memilih pilihan *forgot password* jika *user* lupa dengan kata sandinya.



Gambar 3.18 Rancangan Antarmuka Modal Login

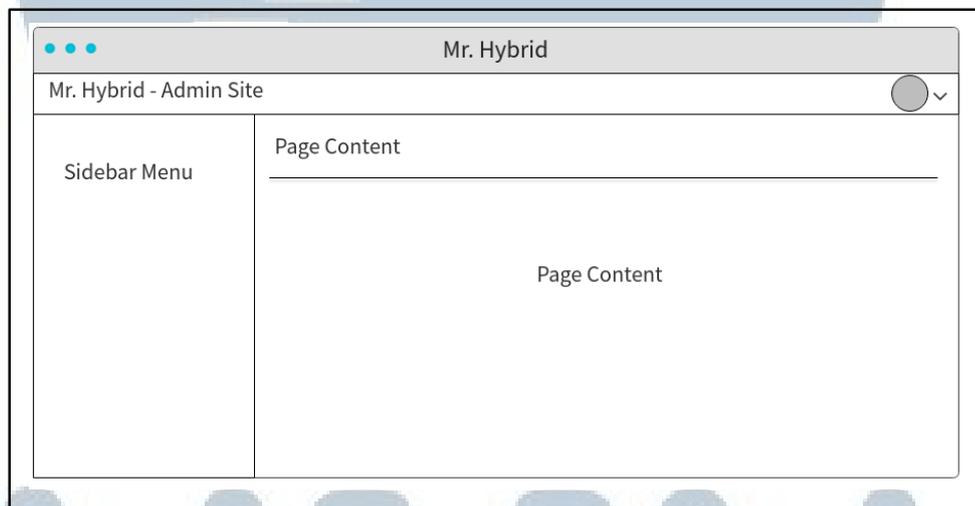
Gambar 3.19 Rancangan Antarmuka Popup Sign Up

Gambar 3.20 merupakan rancangan antarmuka halaman *Login Admin*.

Halaman ini berisi sebuah *form* yang perlu diisi jika ingin melakukan proses *login* sebagai *admin*.

Gambar 3.20 Rancangan Antarmuka Halaman Login Admin

Gambar 3.21 merupakan rancang antarmuka yang digunakan pada setiap halaman yang ada di sisi *admin*. Terdapat beberapa menu yang tersedia di bagian *admin*, seperti menu *Dashboard*, menu *Users*, menu *Movies* dan menu *Messages*. Menu *Member* memiliki dua *submenu*, yaitu *submenu Admin* dan *submenu Member*. Menu *Movies* memiliki empat *submenu*, yaitu *submenu Movies*, *submenu Genres*, *submenu Videos* dan *submenu Reviews*. Semua menu dan *submenu* memiliki format tampilan yang sama, namun dengan isi konten yang berbeda.



Gambar 3.21 Rancangan Antarmuka Halaman Admin

U M M N
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A