BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1. Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

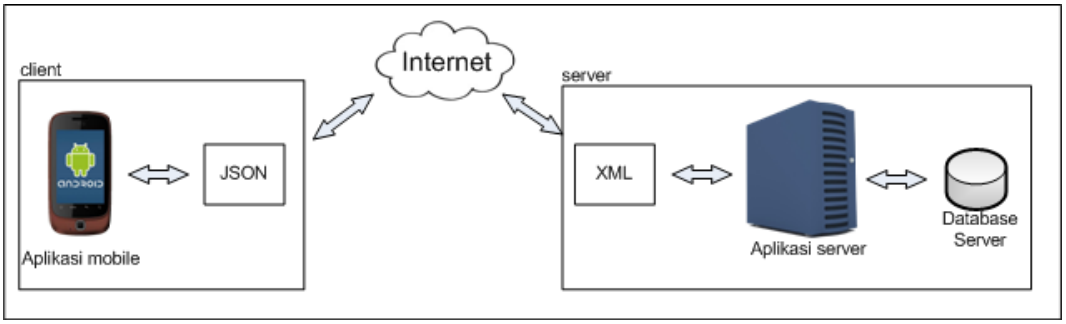
3.1.1. Analisis Masalah

Adapun aplikasi *mobile* yang beredar di masyarakat, diketahui bahwa jarang sekali sistem aplikasi pada suatu perangkat *mobile* yang menyediakan kumpulan informasi dalam satu aplikasi *mobile*.Biasanya kebanyakan dari kumpulan informasi tersebut di sajikan dalam bentuk web.

Oleh karena itu, untuk mengatasi permasalahan tersebut dalam penelitian ini mencoba membangun aplikasi *mobile* yang menyediakan beberapa fitur dan informasi yang berkaitan dengan bioskop. Dimana pengguna tidak harus membuka *browser* pada ponselnya, selain itu proses dalam pencarian informasi lebih cepat dikarenakan minimnya penggunaan bandwidth seperti untuk gambar dll. Penggunaan aplikasi *mobile* ini pun terbilang lebih praktis dan bisa digunakan kapan dan dimana saja. Sehingga dapat mempermudah pengguna dalam mendapatkan informasi-informasi tersebut.

3.1.2. Analisis Arsitektur Sistem

Pada penelitian ini, sistem menerapkan arsitektur sistem yang terdiri dari client dan server.Arsitektur sistem yang digunakan dalam merancang sistem ditunjukan pada gambar 3.1.



Gambar 3.1. Arsitektur Sistem

Aplikasi *client* merupakan antarmuka antara pengguna dengan sistem.Pada bagian *client* terdapat dua bagian yaitu JSON dan aplikasi *mobile*.JSON merupakan bagian dari sistem yang terdapat pada sisi *client*, dimana berfungsi untuk mengumpulkan data yang terdapat di database server dengan cara mengalokasikan data yang didapat dari database server ke dalam masing-masing variabel. Aplikasi *mobile* merupakan sistem yang berfungsi untuk menampilkan semua informasi yang terdapat di *database server*dengan cara menampilkan informasi yang sudah dialokasikan ke dalam variabel-variabel yang terdapat di dalam JSON.

Server merupakan lapisan yang berfungsi sebagai pengelola data yang terdapat pada sistem serta sebagai penghubung antara aplikasi *client* dengan *database server*.Pada perancangan sistem, lapisan ini dibagi menjadi tiga bagian yaitu aplikasi XML, aplikasi *server* dan *database server*.XML merupakan pemformatan data dengan mengalokasikan data yang di-*parsing* ke dalam variabel sesuai dengan variabel yang dibutuhkan, pemformatan data ini berguna dalam pengambilan data oleh JSON.Aplikasi *server* bertanggung jawab dalam mengelola data serta sebagai penghubung aplikasi *client* dan *database server*.*Database server* merupakan aplikasi yang berfungsi menyimpan data-data yang dibutuhkan sistem.

3.1.3. Analisis Kebutuhan Data

Sistem yang akan dibangun merupakan sistem untuk membantu pengguna dalam mendapatkan informasi bioskop dan film. Untuk proses pengolahan data dapat dilakukan melalui aplikasi *backend*.

Http Connection digunakan sebagai penghubung antara aplikasi android ke *database* pada *web server*, sehingga *client* pada android dapat melakukan proses pemanggilan *method* atau fungsi untuk mendapatkan data berupa informasi bioskop beserta film yang dikelola melalui aplikasi *backend*.

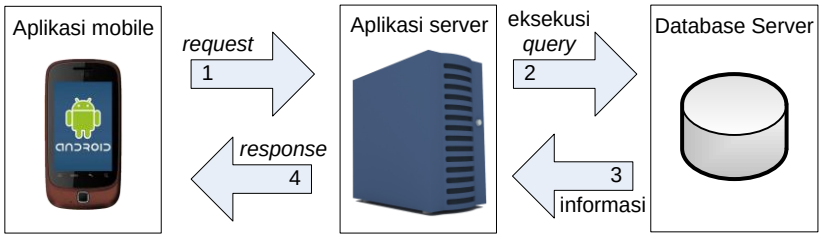
Adapun data yang digunakan dalam merancang sistem Aplikasi Informasi Bioskop Kota-Kota Besar Indonesia dapat dilihat pada Tabel 3.1.

Tabel 3.1. Kebutuhan Data Aplikasi

No	Nama Data	Keterangan
1	Data kota	Data ini berisi informasi kota yang terdiri dari nama kota, jumlah bioskop yang terdapat di masing-masing kota, dan gambar dari kota.
2	Data cinema	Data ini berisi informasi cinema yang terdiri dari id cinema, nama cinema, alamat cinema, telp cinema, kode mtix cinema, htm, nomat, 3d, id kota, kota, logo, latitude dan longitude.
3	Data film	Data ini berisi informasi film yang terdiri dari id film, judul, kode mtix, jenis film, produser, produksi, sutradara, sinopsis, info, likes, link, sampul, dan status film.
4	Data board	Data ini berisi informasi dari jadwal film di masing-masing cinema yang terdiri dari id board, id film, id cinema, studio, showtime1, showtime2, showtime3, showtime4, showtime5, showtime6, dan status.

3.1.4. Analisis Komunikasi Data

Analisis komunikasi data menjelaskan cara pertukaran data antara *client* dan *server*. Dalam penelitian ini pertukaran data menggunakan teknik *HTTP request*, dimana aplikasi akan mengirim sebuah *request* pada suatu aplikasi *server* melalui jaringan *internet* lalu akan menerima *response* dari aplikasi *server* tersebut. Mekanisme komunikasi data dalam perancangan Aplikasi Informasi Bioskop Kota-Kota Besar Indonesia dapat dilihat pada gambar 3.2.



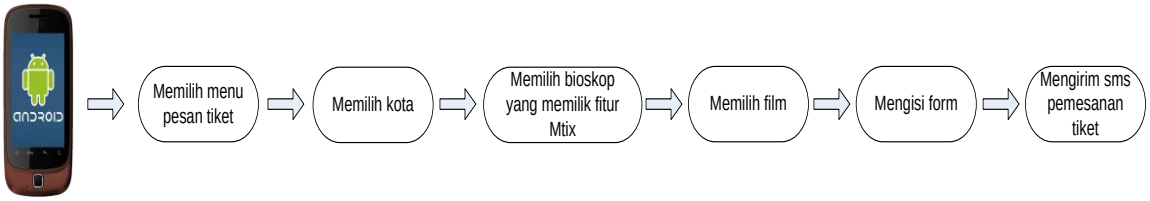
Gambar 3.2. Mekanisme Komunikasi Data Aplikasi

Proses mekanisme komunikasi data dalam perancangan aplikasi dimulai dari aplikasi *client* yang akan mengirim *request* ke aplikasi *server*, kemudian aplikasi *server* pun menerima *request* tersebut, lalu melakukan eksekusi *query* terhadap *database server* dan menerima informasi yang diminta dari *databaseserver* sebagai hasil eksekusi *query* sebelumnya. Setelah itu aplikasi server akan menampilkan informasi dengan format yang sudah ditentukan sebagai *response* yang akan diterima aplikasi *client*. Sehingga aplikasi *client* dapat menampilkan informasi yang didapat dari *database server*.

3.1.5. Analisis Pemesanan Tiket Mtix

Pada sistem yang dibangun, terdapat salah satu fitur dari aplikasi yaitu pemesanan tiket via sms. Fitur ini bernama Mtix, dimana untuk melakukan pemesanan tiket via Mtix ini pengguna diharuskan sudah terdaftar sebagai member Mtix, dengan arti lain pengguna non-*member* layanan Mtix tidak bisa menggunakan fitur ini untuk melakuka pemesanan tiket.

Apabila pengguna sekaligus member layanan Mtix ingin menggunakan fitur pemesanan tiket ini, pengguna tidak harus mengetahui kode-kode seperti kode mtix untuk bioskop, serta kode mtix untuk film. Karena pada aplikasi ini, sistem hanya akan menampilkan bioskop-bioskop yang memiliki fitur mtix dan setelah itu sistem juga akan menampilkan daftar film sekaligus kode mtix dari film yang sedang tayang pada bioskop tertentu.Mekanisme fitur pemesanan tiket Mtix dapat dilihat pada gambar 3.3.



Gambar 3.3 Mekanisme Pemesanan Tiket Mtix

3.1.6. Analisis Dan Kebutuhan Non Fungsional

Analisis dan kebutuhan non fungsional meliputi analisis dan kebutuhan perangkat keras serta analisis dan kebutuhan perangkat lunak. Adapun kebutuhan non-fungsional untuk menjalankan aplikasi ini meliputi kebutuhan perangkat keras, kebutuhan perangkat lunak dan pengguna sistem yang akan memakai aplikasi. Analisis kebutuhan non-fungsional bertujuan agar aplikasi yang dibangun dapat digunakan sesuai dengan kebutuhan pengguna aplikasi dalam mencari informasi yang dibutuhkan.

3.1.6.1 Analisis Pengguna

Pengguna dari aplikasi ini adalah Admin dan masyarakat umum yang terbiasa dengan jaringan internet dan aplikasi *mobile* serta terbiasa dengan perangkat *mobile* dengan *display* layar sentuh, karena aplikasi ini menggunakan beberapa objek sesuai dengan proses-proses yang dapat dilakukan pengguna.

Admin adalah bagian pengguna aplikasi yang dapat menggunakan akses lebih banyak daripada para pengguna aplikasi yang akan dibangun. Admin pada aplikasi yang akan dibangun ini merupakan pengelola program *backend* aplikasi yang bertugas melakukan *insert*, *update* dan *delete* data di *database*.

3.1.6.2 Analisis dan Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembuatan aplikasi informasi bioskop ini adalah :

1. Sistem Operasi Microsoft Windows 7
2. Eclipse Helios
3. XAMPP 1.7.4
4. Android OS v.2.2 (Froyo)
5. JDK 1.6.0
6. AVD Emulator
7. DroiDraw

3.1.6.3 Analisis dan Kebutuhan Perangkat Keras

Untuk menjalankan suatu aplikasi maka diperlukan perangkat keras yang dapat mendukung proses kerja dari sistem itu sendiri. Spesifikasi perangkat keras yang digunakan dalam pembuatan aplikasi informasi bioskop ini adalah :

1. Processor : Intel Core 2 Duo E7500 2.93 GHz
2. RAM : 2 Giga
3. VGA : 256 MB Onboard
4. HDD : 250 GB
5. Monitor : 14”

Sedangkan untuk menjalankan aplikasi ini dibutuhkan seperangkat *handphone*, dengan spesifikasi yang direkomendasikan sebagai berikut :

1. CPU 600 MHz
2. RAM 256 Mb.
3. Memori penyimpanan 128 Mb.
4. Display, TFT capacitive touchscreen 3.14”

3.1.7. Analisis Kebutuhan Fungsional Perangkat Lunak

Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik.

Analisis yang dilakukan dimodelkan dengan menggunakan UML (*Unified Modeling Language*).Tahap-tahap pemodelan dalam analisis tersebut antara lain identifikasi aktor, *usecase diagram, skenario, activity diagram, sequence diagram, class diagram*.

3.1.8. Identifikasi Aktor

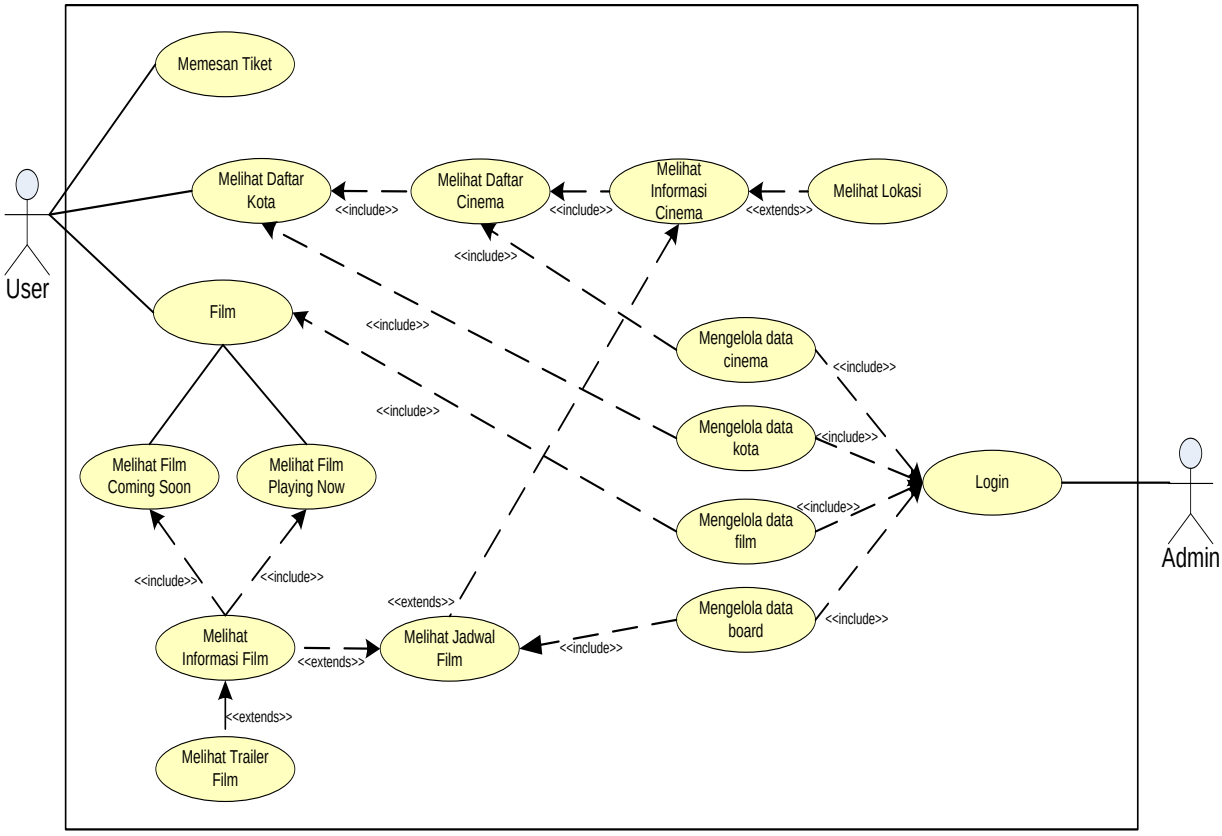
Identifikasi aktor dimaksudkan untuk mengetahui siapa saja aktor yang terlibat di dalam sistem aplikasi ini.Deskripsi dari tiap aktor yang terlibat dapat dilihat pada tabel 3.2.

Tabel 3.2 Identifikasi Aktor

No	Aktor	Deskripsi
1	*User*	*User* ini merupakan pengguna aplikasi pada platform android. Dimana dalam menggunakan aplikasi *user* harus memiliki koneksi internet yang berguna dalam *parsing* data ke *database server*.
2	Admin	Pengelola konten secara keseluruhan, dimana admin memiliki hak akses penuh untuk dapat menambah, mengubah dan menghapus data yang terdapat di *database server*.

3.1.9. Use Case Diagram

Use case Diagram merupakan konstruksi untuk mendeskripsikan hubungan-hubungan yang terjadi antar aktor dengan aktivitas yang terdapat pada sistem. Sasaran pemodelan *use case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan skenario penggunaan sistem yang akan dibangun. Dari hasil analisis aplikasi yang ada maka *use case diagram* pada aplikasi ini dapat dilihat pada gambar 3.4.



Gambar 3.4 *Use Case Diagram* Aplikasi

3.1.9.1 Definisi *Use Case*

Use case diagram merupakan konstruksi untuk mendeskripsikan hubungan-hubungan yang terjadi antar aktor dengan aktivitas yang terdapat pada sistem. Sasaran pemodelan *use case* diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem dengan mendefinisikan skenario penggunaan yang disepakati antara pemakai dan pengembang. Definisi dari *use case* aplikasi dapat dilihat pada tabel 3.3.

Tabel 3.3 Definisi *Use Case*

No	*Use Case*	Deskripsi
1	Pesan tiket	Proses untuk fitur pemesanan tiket via sms.
2	Daftar kota	Proses melihat daftar kota yang terdaftar pada *database*.
3	Daftar cinema	Proses melihat daftar dari bioskop-bioskop yang terdapat pada tiap kota.
4	Film *coming soon*	Proses melihat daftar film yang akan tayang.
5	Film *playing now*	Proses melihat daftar film yang sedang tayang
6	Informasi cinema	Proses melihat informasi dari cinema atau bioskop yang dipilih
7	Lokasi	Proses melihat tampilan map lokasi dari bioskop yang dipilih.
8	Informasi film	Proses melihat informasi film yang dipilih.
9	Jadwal film	Proses melihat informasi film berupa jadwal film di sebuah cinema.
10	*Trailer* film	Proses melihat tentang cuplikan dari salah satu film.
11	Login	Prose untuk login masuk ke halaman admin.
12	Mengelola data kota	Proses untuk melakukan edit, dan hapus data kota.
13	Mengelola data cinema	Proses untuk melakukan edit dan hapus data cinema.
14	Mengelola data film	Proses untuk melakukan edit dan hapus data film.
15	Mengelola data board	Proses untuk melakukan edit dan hapus data board.
16	Tambah data cinema	Proses untuk melakukan penambahan data cinema.
17	Tambah data kota	Proses untuk melakukan penambahan data kota.
18	Tambah data film	Proses untuk melakukan penambahan data film.
19	Tambah data board	Proses untuk melakukan penambahan data board.

3.1.10. Skenario *Use Case*

Skenario *Use Case* mendeskripsikan urutan langkah-langkah dalam proses bisnis, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor.

3.1.10.1 *Use Case* Skenario Pesan Tiket

Interaksi antara aktor *user* dengan *use case* pesan tiket dijelaskan dalam tabel 3.4.

Tabel 3.4 *Use Case* Skenario Pesan Tiket

Identifikasi		
Nomor	1	
Nama	Pesan tiket	
Tujuan	*User* dapat melakukan pemesanan tiket via sms.	
Deskripsi	Menu ini berisi form yang terdiri dari beberapa textfield yang harus diisi, antara lain jumlah tiket, jam tayang, tanggal dan pin.	
Aktor	*User*	
Skenario Utama		
Kondisi Awal	*User* berada di menu utama aplikasi bioskop	
Aksi Aktor		**Reaksi Sistem**
1. *User* memilih menu Pesan tiket.		2. Aplikasi menampilkan daftar kota yang terdapat fitur Mtix.
3. Memilih kota.		4. Menampilkan daftar bioskop yang terdapat fitur Mtix.
5. Memilih bioskop.		6. Menampilkan daftar film
8. Mengisi form pemesanan tiket.		7. Menampilkan form pemesanan tiket
9. Menekan tombol kirim		10. Mengirim pesan.
11. *User* mendapat sms balasan untuk konfirmasi		
Kondisi Akhir	*User* telah melakukan pemesanan tiket via sms.	

3.1.10.2 *Use Case* Skenario Daftar Kota

Interaksi antara aktor *user* dengan *use case* daftar kota dijelaskan dalam tabel 3.5.

Tabel 3.5 *Use Case* Skenario Daftar Kota

Identifikasi		
Nomor	2	
Nama	Daftar kota.	
Tujuan	Pengguna memilih salah satu kota dari daftar kota.	
Deskripsi	Menu ini menampilkan daftar kota-kota yang terdaftar pada *database* sehingga dengan memilih kota pengguna dapat mencari bioskop dari kota yang dipilih	
Aktor	*User*.	
Skenario Utama		
Kondisi Awal	*User* berada di menu utama aplikasi bioskop.	
Aksi Aktor		**Reaksi Sistem**
1. *User* memilih menu kota.		2. Aplikasi melakukan *parsing* data dari tabel kota yang terdapat di *databaseserver*.
		3. Aplikasi menampilkan daftar kota sesuai dengan data yang terdapat di *database server*.
Kondisi Akhir	Aplikasi menampilkan daftar kota sesuai dengan data kota yang terdapat di *database server*.	

3.1.10.3 *Use Case* Skenario Daftar Bioskop

Interaksi antara aktor *user* dengan *use case* daftar bioskop dijelaskan dalam tabel 3.6.

Tabel 3.6 *Use Case* Skenario Daftar Bioskop

Identifikasi		
Nomor	3	
Nama	Daftar bioskop.	
Tujuan	Pengguna memilih salah satu bioskop sessuai dengan kota yang dipilih.	
Deskripsi	Menu ini menampilkan daftar dari bioskop dimana bioskop yang ditampilkan sesuai dengan kota yang dipilih. Data bioskop dan kota terdapat di *database server*.	
Aktor	*User.*	
Skenario Utama		
Kondisi Awal	*User* berada di menu kota.	
Aksi Aktor		**Reaksi Sistem**
1. *User* memilih menu kota.		2. Aplikasi melakukan *parsing* data dari tabel kota yang terdapat di *database server*.
		3. Aplikasi menampilkan daftar kota sesuai dengan data yang terdapat di *database server*.
4. *User* memilih salah satu dari daftar kota yang ditampilkan		5. Melakukan *parsing* data dari tabel cinema dari *database server* sesuai dengan kota yang dipilih.
		6. Menampilkan daftar bioskop sesuai dengan data kota yang dipilih.
Kondisi Akhir	Sistem menampilkan daftar bioskop sesuai dengan kota yang dipilih *user*.	

3.1.10.4 *Use Case* Skenario Film *Coming Soon*

Interaksi antara aktor *user* dengan *use case* film *coming soon* dijelaskan dalam tabel 3.7.

Tabel 3.7 *Use Case* Skenario Film *Coming Soon*

Identifikasi		
Nomor	4	
Nama	Film *coming soon*.	
Tujuan	*User* dapat melihat daftar film yang akan tayang di bioskop-bioskop.	
Deskripsi	Menu ini menampilkan daftar dari film-film, dimana film yang ditampilkan memiliki status *coming soon*. Daftar film-film ini didapat dari data film yang terdapat di *database server*.	
Aktor	*User*.	
Skenario Utama		
Kondisi Awal	*User* berada di menu utama aplikasi bioskop.	
Aksi Aktor		**Reaksi Sistem**
1. *User* memilih menu film.		2. Aplikasi melakukan *parsing* data film yang memiliki status *playing now* di *database server*.
		3. Menampilkan tab halaman film *playing now* dan menampilkan daftar film *playing now*.
4. *User* memilih tab halaman film *coming soon*.		5. Aplikasi melakukan *parsing* data film yang memiliki status *coming soon* di *database server*.
		6. Menampilkan tab halaman film *coming soon* dan menampilkan daftar film *coming soon*.
Kondisi Akhir	Aplikasi menampilkan tab film *coming soon* yang berisi daftar film *coming soon*	

3.1.10.5 *Use Case* Skenario Film *Playing Now*

Interaksi antara aktor *user* dengan *use case* film *playing now* dijelaskan dalamtabel 3.8.

Tabel 3.8 *Use Case* Skenario Film *Playing Now*

Identifikasi		
Nomor	5	
Nama	Film *playing now*.	
Tujuan	*User* dapat melihat dan memilih salah satu dari daftar film yang sedang tayang.	
Deskripsi	Pada menu ini, aplikasi akan mengambil data film yang terdapat di *database server*. Dimana data film yang diambil terdapat pada tabel film dan memiliki status *playing now*.	
Aktor	*User*.	
Skenario Utama		
Kondisi Awal	*User* berada di menu utama aplikasi bioskop.	
Aksi Aktor		**Reaksi Sistem**
1. *User* memilih menu film.		2. Aplikasi melakukan *parsing* data film yang memiliki status *playing now* di *database server*.
		4. Menampilkan tab halaman film *playing now* dan menampilkan daftar film *playing now*.
Kondisi Akhir	Aplikasi menampilkan tab halaman film *playing now* yang berisi daftar film *playing now*.	

3.1.10.6 *Use Case* Skenario Informasi Bioskop

Interaksi antara aktor *user* dengan *use case* informasi bioskop dijelaskan dalam tabel 3.9.

Tabel 3.9 *Use Case* Skenario Informasi Bioskop

Identifikasi		
Nomor	6	
Nama	Informasi bioskop.	
Tujuan	*User* dapat melihat informasi dari bioskop yang dipilih, dianatarnya nama bioskop, alamat, nomor telepon, harga tiket.	
Deskripsi	Pada menu ini, aplikasi akan mengambil data di *database server*, yaitu data dari semua field yang ada pada tabel bioskop sesuai dengan bioskop yang dipilih oleh *user*.	
Aktor	*User*.	
Skenario Utama		
Kondisi Awal	*User* berada di menu utama aplikasi bioskop.	
Aksi Aktor		**Reaksi Sistem**
1. *User* memilih menu kota.		2. Aplikasi melakukan *parsing* data dari tabel kota yang terdapat di *database server*.
		3. Menampilkan daftar kota sesuai dengan data yang terdapat di *database server*.
4. *User* memilih salah satu dari daftar kota yang ditampilkan		5. Melakukan *parsing* data dari tabel cinema dari *database server* sesuai dengan kota yang dipilih.
		6. Menampilkan daftar bioskop sesuai dengan data kota yang dipilih.
7. *User* memilih salah satu bioskop dari daftar bioskop.		8. Aplikasi mengambil nama kota dan bioskop yang dipilih *user*, kemudian menggunakannya untuk melakukan *parsing* data informasi bioskop yang di pilih oleh *user*.
		9. Menampilkan halaman yang berisi informasi dari bioskop yang dipilih oleh *user*.
Kondisi Akhir	Aplikasi menampilkan halaman informasi bioskop yang dipilih.	

3.1.10.7 *Use Case* Skenario Lokasi

Interaksi antara aktor *user* dengan *use case* lokasi dijelaskan dalam tabel 3.10.

Tabel 3.10 *Use Case* Skenario Lokasi

Identifikasi		
Nomor	7	
Nama	Lokasi.	
Tujuan	*User* dapat melihat lokasi dari bioskop yang dipilih.	
Deskripsi	Menu ini menampilkan lokasi bioskop pada peta, dimana untuk lokasi bioskop menggunakan titik longitude dan latitude. Titik-titik tersebut telah terdaftar pada *database server*.	
Aktor	*User*.	
Skenario Utama		
Kondisi Awal	*User* berada pada halaman informasi bioskop.	
Aksi Aktor		**Reaksi Sistem**
1. *User* menekan tombol lihat lokasi.		2. Aplikasi mengambil nama bioskop dan kota untuk mencari data latitude dan longitude bioskop tersebut.
		3. Aplikasi menggunakan titik latitude dan longitude yang didapat dari *database server*.
		4. Menampilkan posisi bioskop yang dipilih pada peta.
Kondisi Akhir	Aplikasi menampilkan posisi bioskop yang dipilih *user* dalam peta.	

3.1.10.8 *Use Case* Skenario Informasi Film

Interaksi antara aktor *user* dengan *use case* informasi film dijelaskan dalam tabel 3.11.

Tabel 3.11 *Use Case* Skenario Informasi Film

<table>
<tr><th colspan="3">Identifikasi</th></tr>
<tr><td>Nomor</td><td colspan="2">8</td></tr>
<tr><td>Nama</td><td colspan="2">Informasi film.</td></tr>
<tr><td>Tujuan</td><td colspan="2">User dapat melihat informasi dari film yang dipilih, seperti judul, jenis film, sinopsis dll.</td></tr>
<tr><td>Deskripsi</td><td colspan="2">Pada menu ini, aplikasi mengambil data informasi film yang dipilih dari database server, lalu menampilkan data yang didapat tersebut sehingga informasi dapat dilihat user.</td></tr>
<tr><td>Aktor</td><td colspan="2">User.</td></tr>
<tr><th colspan="3">Skenario Utama</th></tr>
<tr><td>Kondisi Awal</td><td colspan="2">User berada di menu film.</td></tr>
<tr><td colspan="2">Aksi Aktor</td><td>Reaksi Sistem</td></tr>
<tr><td colspan="2">1. User memilih salah satu film dari daftar film.</td><td>2. Aplikasi mengambil nama film yang dipilih lalu menggunakannya untuk melakukan parsing data informasi film di database server.</td></tr>
<tr><td colspan="2"></td><td>3. Menampilkan halaman informasi film.</td></tr>
<tr><td>Kondisi Akhir</td><td colspan="2">Aplikasi menampilkan halaman informasi dari film yang dipilih user.</td></tr>
</table>

3.1.10.9 *Use Case* Skenario Jadwal Film

Interaksi antara aktor *user* dengan *use case* jadwal film dijelaskan dalam tabel 3.12.

Tabel 3.12 *Use Case* Skenario Jadwal Film

Identifikasi		
Nomor	9	
Nama	Jadwal film.	
Tujuan	*User* dapat melihat jadwal film yang sedang tayang di bioskop yang dipilih.	
Deskripsi	Pada menu ini, aplikasi akan mengambil nama dari bioskop dan kota untuk mencari jadwal film yang sedang tayang, dimana jadwal film yang sedang tayang terdapat pada tabel board di *database server*.	
Aktor	*User*.	
Skenario Utama		
Kondisi Awal	*User* berada di halaman informasi bioskop.	
Aksi Aktor		**Reaksi Sistem**
1. *User* memilih menu lihat jadwal film.		2. Aplikasi melakukan *parsing* data jadwal film di dalam tabel board dengan menggunakan nama bioskop yang dipilih.
		3. Menampilkan data jadwal film yang didapat dari *database server*.
Kondisi Akhir	Aplikasi menampilkan halaman jadwal dari film yang dipilih oleh *user*.	

3.1.10.10 *Use Case* Skenario *Trailer* Film

Interaksi antara aktor *user* dengan *use case trailer* film dijelaskan dalam tabel 3.13.

Tabel 3.13 *Use Case* Skenario *Trailer* Film

Identifikasi		
Nomor	10	
Nama	*Trailer* film.	
Tujuan	*User* dapat melihat cuplikan dari film yang sedang tayang dan akan tayang.	
Deskripsi	Pada menu ini, aplikasi akan mengambil data berupa alamat link dari cuplikan film yang dipilih dimana alamat link tersebut telah disimpan di *database server*, sehingga aplikasi dapat membuka dan menjalankan *browser* langsung ke alamat link.	
Aktor	*User*.	
Skenario Utama		
Kondisi Awal	*User* berada di halaman informasi film.	
Aksi Aktor		**Reaksi Sistem**
1. *User* menekan tombol lihat *trailer* pada halaman informasi film.		2. Aplikasi mengambil data berupa link alamat dari cuplikan film.
		3. Aplikasi mengaktifkan *browser* pada handphone dan membuka link alamat cuplikan film yang baru saja diambil.
Kondisi Akhir	Aplikasi membuka *browser* dan langsung memanggil alamat dari link yang didapat.	

3.1.10.11 *Use Case* Skenario Tambah Data Board

Interaksi antara aktor admin dengan *use case* tambah data board dijelaskan dalam tabel 3.14.

Tabel 3.14 *Use Case* Skenario Tambah Data Board

Identifikasi		
Nomor	11	
Nama	Tambah data board.	
Tujuan	Admin dapat menambah data board.	
Deskripsi	Pada halaman ini, admin mengisi form untuk menambah data board, seperti id film, id cinema, studio, *showtime*1sampai dengan *showtime*6.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu tambah data board.		4. Menampilkan form pengisian data board.
5. Admin mengisi form pengisian data board dan menekan tombol simpan.		6. Memeriksa apakah data sudah sesuai, jika : - Ya, simpan data. - Tidak, kembali ke form pengisian data board.
Kondisi Akhir	Pengisian data board berhasil dan data tersimpan ke *database server*.	

3.1.10.12 *Use Case* Skenario *Update* Data Board

Interaksi antara aktor admin dengan *use case update* data board dijelaskan dalam tabel 3.15.

Tabel 3.15 *Use Case* Skenario *Update* Data Board

Identifikasi		
Nomor	12	
Nama	*Update* data board.	
Tujuan	Admin dapat meng-*update* data board yang sudah tersimpan sebelumnya di *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu data board, dan mengisi form peng-*update*-an data board.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data board.		4. Menampilkan daftar data board yang tersimpan di *database server* beserta.
5. Admin memilih data board yang akan diubah lalu menekan tombol update *edit*.		6. Menampilkan form *update* data board.
7. Mengisi form *update* data board, lalu menekan tombol *update*.		8. Memeriksa apakah data sudah sesuai, jika : - Ya, *update* data. - Tidak, kembali ke form *update* data board.
Kondisi Akhir	Peng-*update*-an data board berhasil dan data board yang telah tersimpan sebelumnya telah berubah.	

3.1.10.13 *Use Case* Skenario Hapus Data Board

Interaksi antara aktor admin dengan *use case* hapus data board dijelaskan dalam tabel 3.16.

Tabel 3.16 *Use Case* Skenario Hapus Data Board

Identifikasi		
Nomor	13	
Nama	Hapus data board.	
Tujuan	Admin dapat menghapus data board yang sudah tidak diperlukan lagi dari *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu dari data board yang tersimpan, dan memilih data tersebut untuk di hapus dari *database server*.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data board.		4. Menampilkan daftar data board yang tersimpan di *database server*.
5. Admin memilih data board dan menekan tombol hapus.		6. Menampilkan jendela konfirmasi apakah data akan dihapus atau tidak.
7. Menjawab jendela konfirmasi dengan menekan tombol ya atau tidak.		8. Memeriksa jawaban konfirmasi, jika : - Ya, hapus data. - Tidak, kembali ke halaman data board
Kondisi Akhir	Data board yang dipilih telah terhapus dari *database server*.	

3.1.10.14 *Use Case* Skenario Tambah Data Kota

Interaksi antara aktor admin dengan *use case* tambah data kota dijelaskan dalam tabel 3.17.

Tabel 3.17 *Use Case* Skenario Tambah Data Kota

Identifikasi		
Nomor	14	
Nama	Tambah data kota.	
Tujuan	Admin dapat menambah data kota.	
Deskripsi	Pada halaman ini, admin mengisi form untuk menambah data kota, seperti id, nama dan jumlah bioskop yang terdapat pada masing-masing kota.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu tambah data kota.		4. Menampilkan form pengisian data kota.
5. Admin mengisi form pengisian data kota dan menekan tombol simpan.		6. Memeriksa apakah data sudah sesuai, jika : - Ya, simpan data. - Tidak, kembali ke form pengisian data kota.
Kondisi Akhir	Pengisian data kota berhasil dan data tersimpan ke *database server*.	

3.1.10.15 *Use Case* Skenario *Update* Data Kota

Interaksi antara aktor admin dengan *use case update* data kota dijelaskan dalam tabel 3.18.

Tabel 3.18 *Use Case* Skenario *Update* Data Kota

Identifikasi		
Nomor	15	
Nama	*Update* data kota.	
Tujuan	Admin dapat meng-*update* data kota yang sudah tersimpan sebelumnya di *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu data kota, dan mengisi form peng-*update*-an data kota.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data kota.		4. Menampilkan daftar data kota yang tersimpan di *database server*.
5. Admin memilih data kota dan menekan tombol *edit*.		6. Menampilkan form *update* data kota.
7. Mengisi form *update* data kota dan menekan tombol *update*.		8. Memeriksa apakah data sudah sesuai, jika : - Ya, *update* data. - Tidak, kembali ke form *update* data kota.
Kondisi Akhir	Peng-*update*-an data kota berhasil dan data kota yang telah tersimpan sebelumnya telah berubah.	

3.1.10.16 *Use Case* Skenario Hapus Data Kota

Interaksi antara aktor admin dengan *use case* hapus data kota dijelaskan dalam tabel 3.19.

Tabel 3.19 *Use Case* Skenario Hapus Data Kota

Identifikasi		
Nomor	16	
Nama	Hapus data kota.	
Tujuan	Admin dapat menghapus data kota yang sudah tidak diperlukan lagi dari *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu dari data kota yang tersimpan, dan memilih data tersebut untuk di hapus dari *database server*.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data kota.		4. Menampilkan daftar data kota yang tersimpan di *database server*.
5. Admin memilih data kota dan menekan tombol hapus.		6. Menampilkan jendela konfirmasi apakah data akan dihapus atau tidak.
7. Menjawab jendela konfirmasi dengan menekan tombol ya atau tidak.		8. Memeriksa jawaban konfirmasi, jika : - Ya, hapus data kota. - Tidak, kembali ke halaman data kota.
Kondisi Akhir	Data kota yang dipilih telah terhapus dari *database server*.	

3.1.10.17 *Use Case* Skenario Tambah Data Bioskop

Interaksi antara aktor admin dengan *use case* tambah data bioskop dijelaskan dalam tabel 3.20.

Tabel 3.20 *Use Case* Skenario Tambah Data Bioskop

Identifikasi		
Nomor	17	
Nama	Tambah data bioskop.	
Tujuan	Admin dapat menambah data bioskop.	
Deskripsi	Pada halaman ini, admin mengisi form untuk menambah data bioskop, seperti id, nama, alamat, telp, harga tiket, kota, logo dan titik posisi.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu tambah data bioskop.		4. Menampilkan form pengisian data bioskop.
5. Admin mengisi form pengisian data bioskop dan menekan tombol simpan.		6. Memeriksa apakah data sudah sesuai, jika : - Ya, simpan data. - Tidak, kembali ke form pengisian data bioskop.
Kondisi Akhir	Pengisian data bioskop berhasil dan data tersimpan ke *database server*.	

3.1.10.18 *Use Case* Skenario *Update* Data Bioskop

Interaksi antara aktor admin dengan *use case update* data bioskop dijelaskan dalam tabel 3.21.

Tabel 3.21 *Use Case* Skenario *Update* Data Bioskop

Identifikasi		
Nomor	18	
Nama	*Update* data bioskop.	
Tujuan	Admin dapat meng-*update* data bioskop yang sudah tersimpan sebelumnya di *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu data bioskop, dan mengisi form peng-*update*-an data bioskop.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data bioskop.		4. Menampilkan daftar data bioskop yang tersimpan di *database server*.
5. Admin memilih data bioskop dan menekan tombol *edit*.		6. Menampilkan form *update* data bioskop.
7. Mengisi form *update* data bioskop dan menekan tombol *update*.		8. Memeriksa apakah data sudah sesuai, jika : - Ya, *update* data. - Tidak, kembali ke form *update* data bioskop.
Kondisi Akhir	Peng-*update*-an data bioskop berhasil dan data bioskop yang telah tersimpan sebelumnya telah berubah.	

3.1.10.19 *Use Case* Skenario Hapus Data Bioskop

Interaksi antara aktor admin dengan *use case* hapus data bioskop dijelaskan dalam tabel 3.22.

Tabel 3.22 *Use Case* Skenario Hapus Data Bioskop

Identifikasi		
Nomor	19	
Nama	Hapus data bioskop.	
Tujuan	Admin dapat menghapus data bioskop yang sudah tidak diperlukan lagi dari *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu dari data bioskop yang tersimpan, dan memilih hapus data tersebut dari *database server*.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data bioskop.		4. Menampilkan daftar data bioskop yang tersimpan di *database server*.
5. Admin memilih data bioskop dan menekan tombol hapus.		6. Menampilkan jendela konfirmasi apakah data akan dihapus atau tidak.
7. Menjawab jendela konfirmasi dengan menekan tombol ya atau tidak.		8. Menerima jawaban konfirmasi, jika : - Ya, hapus data bioskop. - Tidak, kembali ke halaman data bioskop.
Kondisi Akhir	Data bioskop yang dipilih telah terhapus dari *database server*.	

3.1.10.20 *Use Case* Skenario Tambah Data Film

Interaksi antara aktor admin dengan *use case* tambah data film dijelaskan dalam tabel 3.23.

Tabel 3.23 *Use Case* Skenario Tambah Data Film

Identifikasi		
Nomor	20	
Nama	Tambah data film.	
Tujuan	Admin dapat menambah data film.	
Deskripsi	Pada halaman ini, admin mengisi form untuk menambah data film, seperti id, judul, jenis, sinopsis, info, link, sampul dan status.	
Aktor	Admin	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu tambah data film.		4. Menampilkan form pengisian data film.
5. Admin mengisi form pengisian data film dan menekan tombol simpan.		6. Memeriksa apa data sudah sesuai, jika : - Ya, simpan data. - Tidak, kembali ke form pengisian data film.
Kondisi Akhir	Pengisian data film berhasil dan data tersimpan ke *database server*.	

3.1.10.21 *Use Case* Skenario *Update* Data Film

Interaksi antara aktor admin dengan *use case update* data film dijelaskan dalam tabel 3.24.

Tabel 3.24 *Use Case* Skenario *Update* Data Film

Identifikasi		
Nomor	21	
Nama	*Update* data film.	
Tujuan	Admin dapat meng-*update* datafilm yang sudah tersimpan sebelumnya di *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu data film, dan mengisi form peng-*update*-an data film.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data film.		4. Menampilkan daftar data film yang tersimpan di *database server*.
5. Admin memilih data film dan menekan tombol *edit*.		6. Menampilkan form *update* data film.
7. Mengisi form *update* data film dan menekan tombol *update*.		8. Memeriksa apa data sudah sesuai, jika : - Ya, *update* data. - Tidak, kembali ke form *update* data film..
Kondisi Akhir	Peng-*update*-an data film berhasil dan data film yang telah tersimpan sebelumnya telah berubah.	

3.1.10.22 *Use Case* Skenario Hapus Data Film

Interaksi antara aktor admin dengan *use case* hapus data film dijelaskan dalam tabel 3.25.

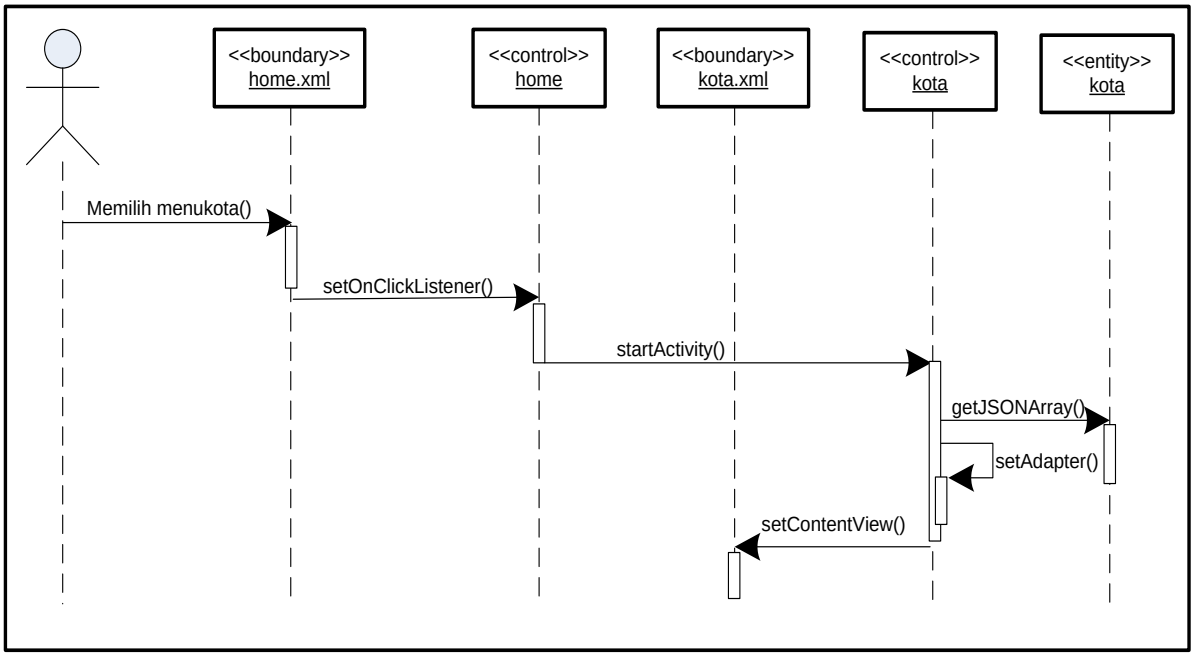
Tabel 3.25 *Use Case* Skenario Hapus Data Film

Identifikasi		
Nomor	22	
Nama	Hapus data film.	
Tujuan	Admin dapat menghapus data film yang sudah tidak diperlukan lagi dari *database server*.	
Deskripsi	Pada halaman ini, admin akan memilih salah satu dari data film yang tersimpan, dan memilih hapus data tersebut dari *database server*.	
Aktor	Admin.	
Skenario Utama		
Kondisi Awal	Admin berada di halaman login admin.	
Aksi Aktor		**Reaksi Sistem**
1. Admin login.		2. Menampilkan halaman utama *backend* aplikasi.
3. Admin memilih menu data film.		4. Menampilkan daftar data film yang tersimpan di *database server*.
5. Admin memilih data film dan menekan tombol hapus.		6. Menampilkan jendela konfirmasi apakah data film akan dihapus atau tidak.
7. Menjawab jendela konfirmasi dengan menekan tombol ya atau tidak		8. Memeriksa jawab konfirmasi, jika : - Ya, hapus data film. - Tidak, kembali ke halaman data film.
Kondisi Akhir	Data film yang dipilih telah terhapus dari *database server*.	

3.1.11. *Sequence Diagram*

3.1.11.1 *Sequence Diagram* Daftar Kota

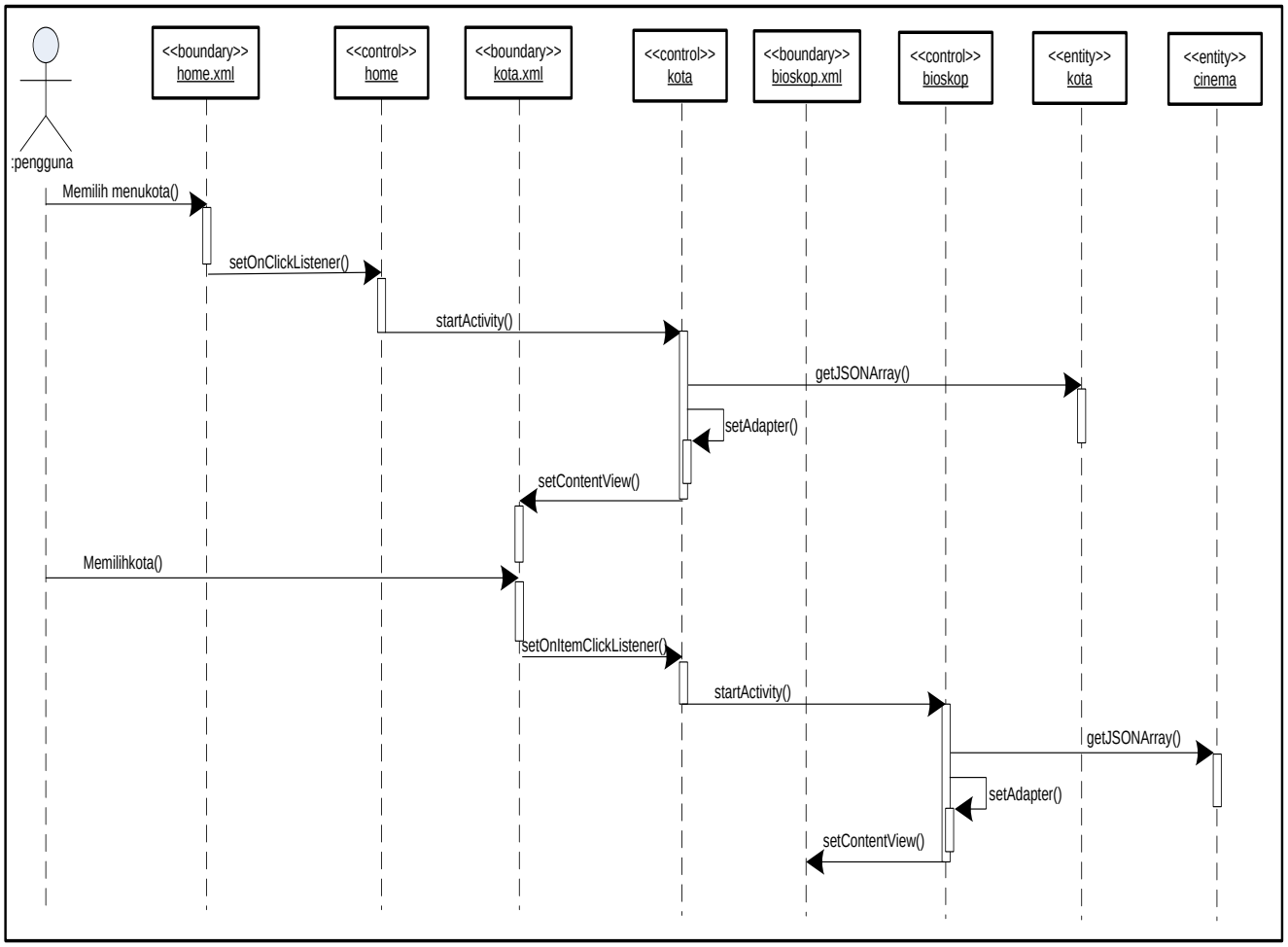
Sequence diagram daftar kota menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat daftar kota. *Sequence diagram* daftar kota dapat dilihat pada gambar 3.5.



Gambar 3.5 *Sequence Diagram* Daftar Kota

3.1.11.2 *Sequence Diagram* Daftar Bioskop

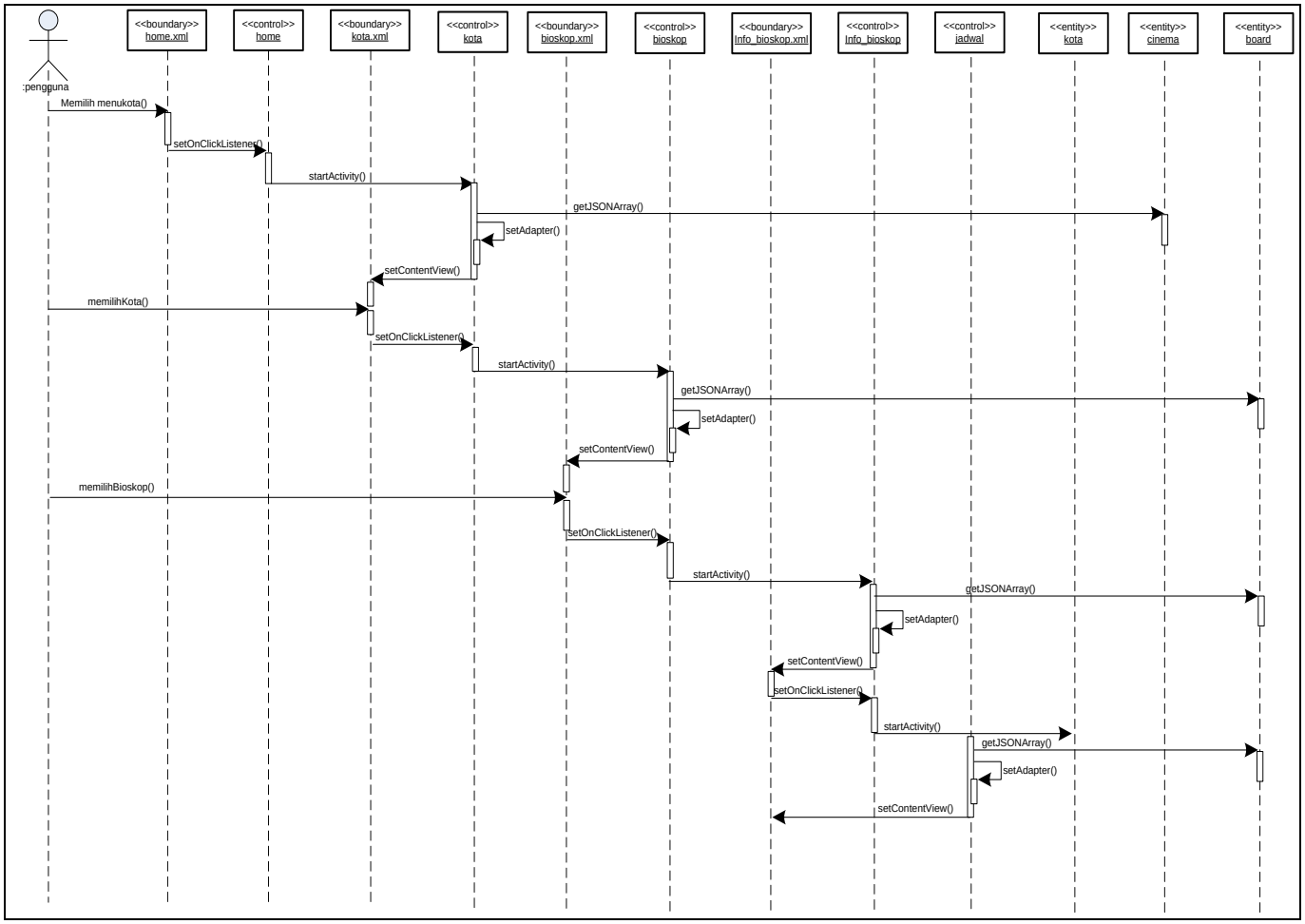
Sequence diagram daftar bioskop menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat daftar bioskop. *Sequence diagram* daftar bioskop dapat dilihat pada gambar 3.6.



Gambar 3.6 *Sequence Diagram* Daftar Bioskop

3.1.11.3 *Sequence Diagram* Informasi Bioskop

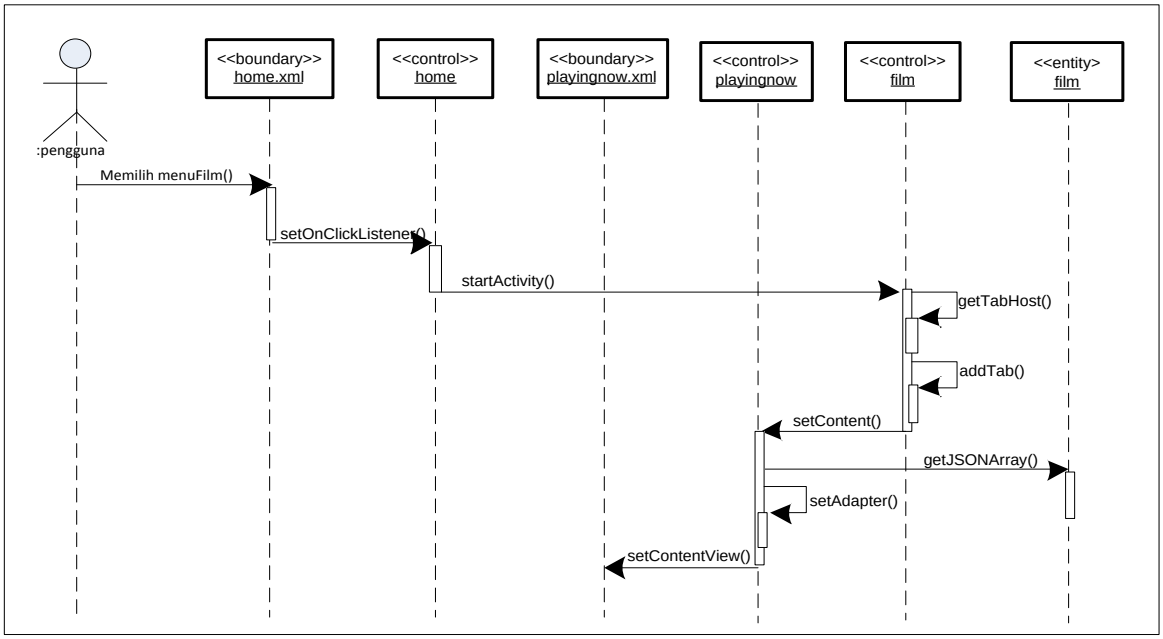
Sequence diagram informasi bioskop menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat informasi dari bioskop yang dipilih. *Sequence diagram* informasi bioskop dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Sequence Diagram* Informasi Bioskop

3.1.11.4 *Sequence Diagram* Film Playing Now

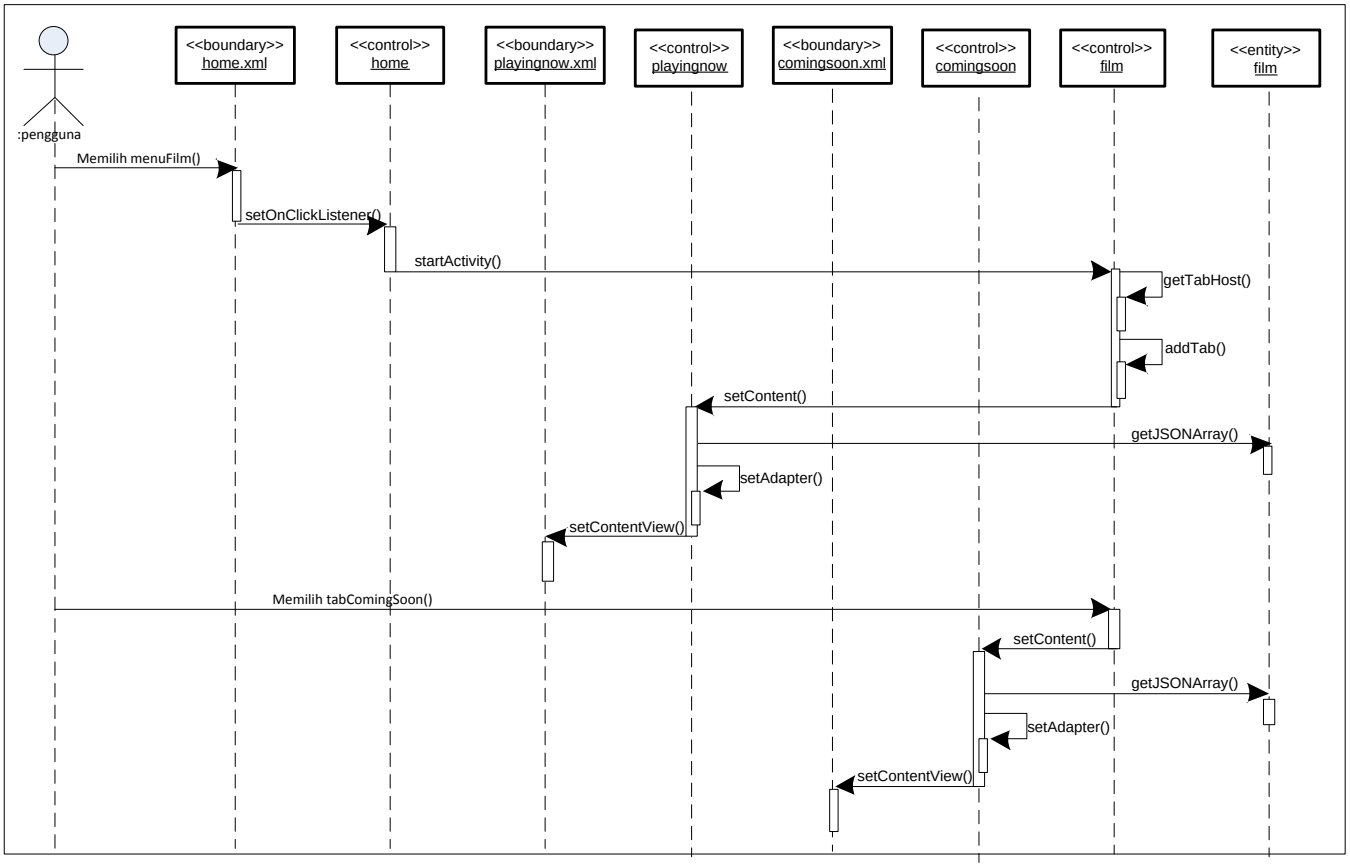
Sequence diagram film *playing now* menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat daftar film yang sedang tayang. *Sequence diagram* film *playing now* dapat dilihat pada gambar 3.8.



Gambar 3.8 *Sequence Diagram* Film Playing Now

3.1.11.5 *Sequence Diagram* Film Coming Soon

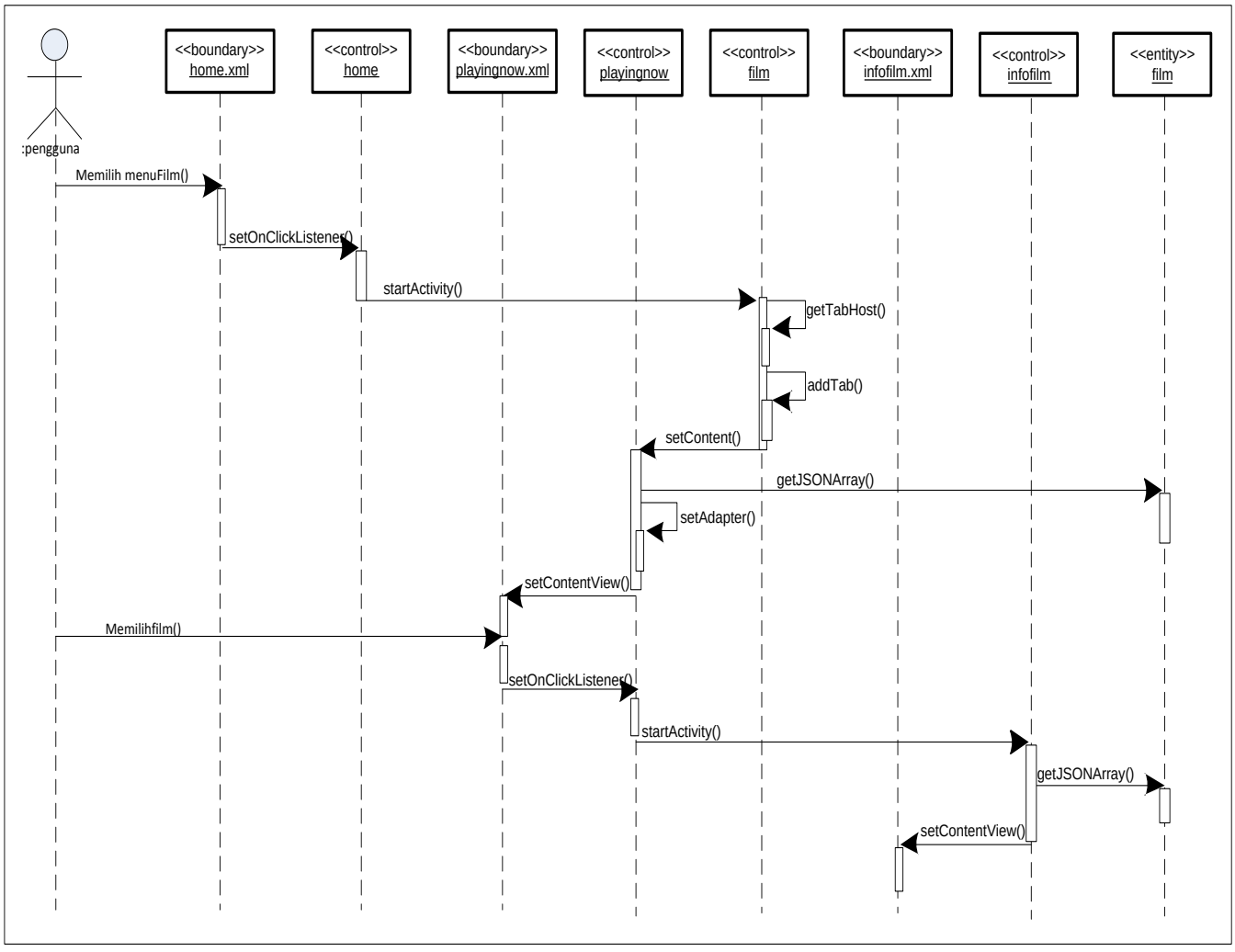
Sequence diagram film *coming soon* menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat daftar film yang akan tayang. *Sequence diagram* film *coming soon* dapat dilihat pada gambar 3.9.



Gambar 3.9 *Sequence Diagram* Film Coming Soon

3.1.11.6 *Sequence Diagram* Melihat Informasi Film

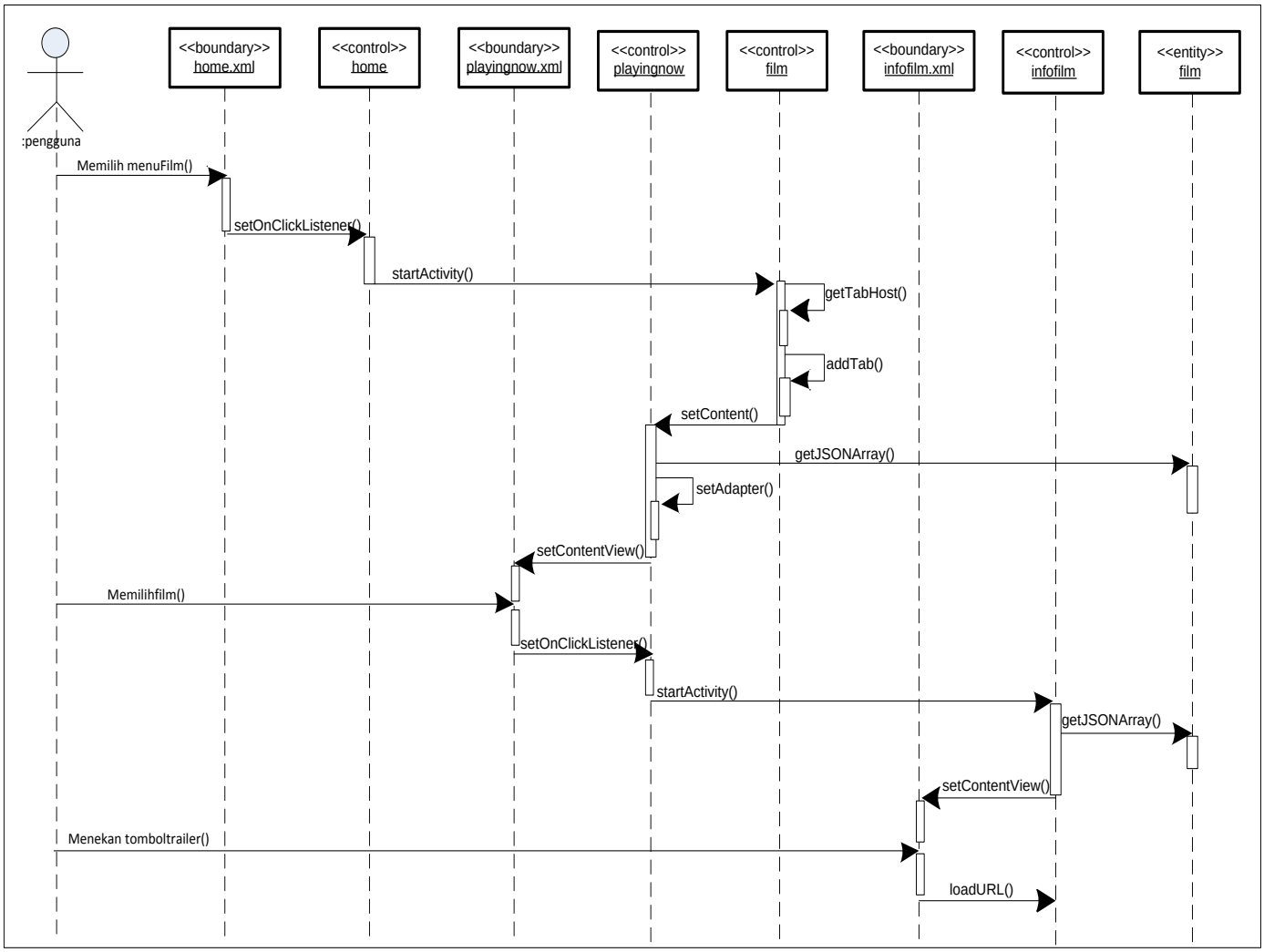
Sequence diagram melihat informasi film menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat informasi dari film yang dipilih. *Sequence diagram* melihat informasi film dapat dilihat pada gambar 3.10.



Gambar 3.10 *Sequence Diagram* Melihat Informasi Film

3.1.11.7 *Sequence Diagram* Trailer Film

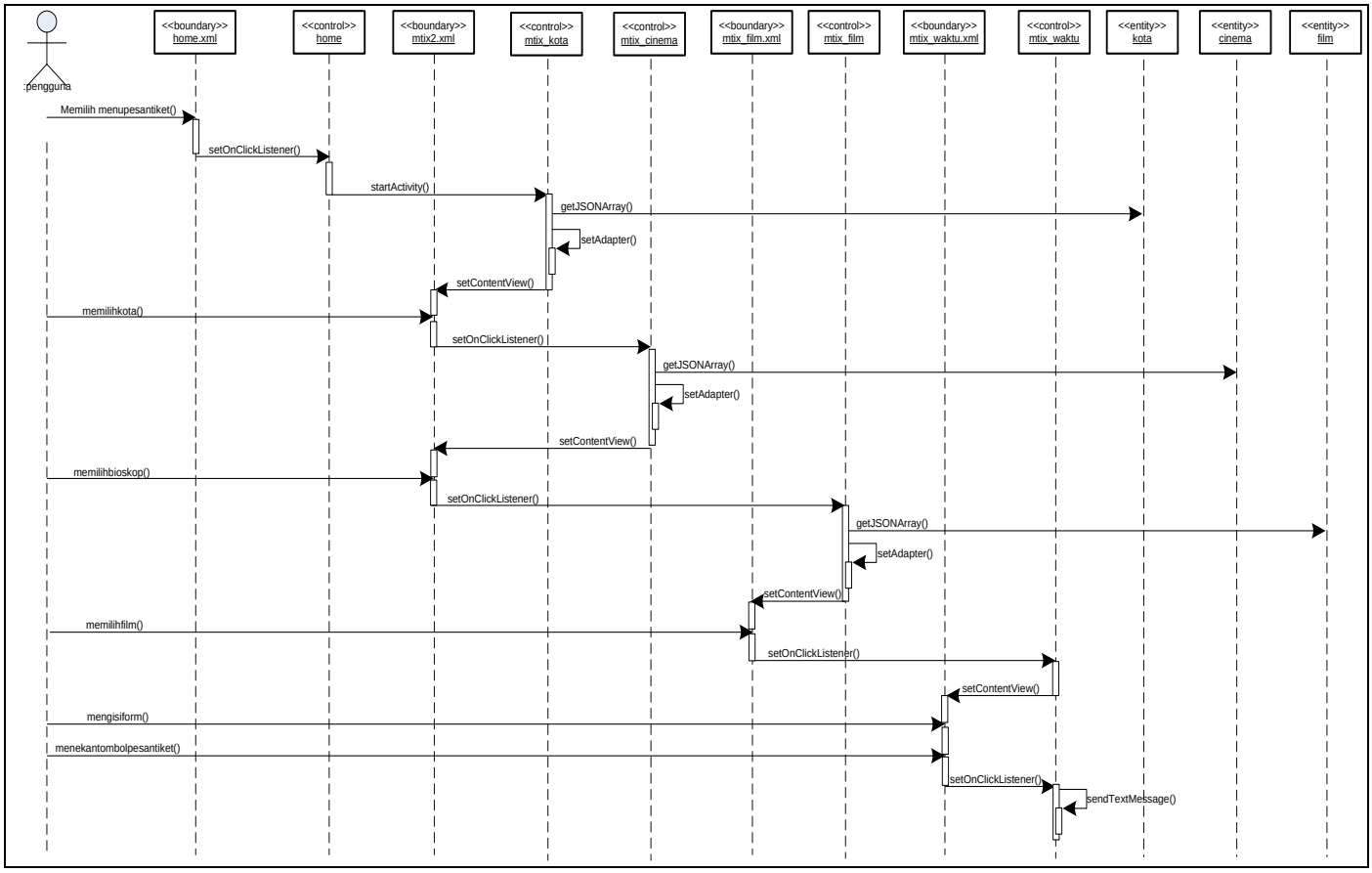
Sequence diagram trailer film menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat cuplikan dari sebuah film yang dipilih. *Sequence diagram trailer* film dapat dilihat pada gambar 3.11.



Gambar 3.11 *Sequence Diagram* Trailer Film

3.1.11.8 *Sequence Diagram* Pesan Tiket

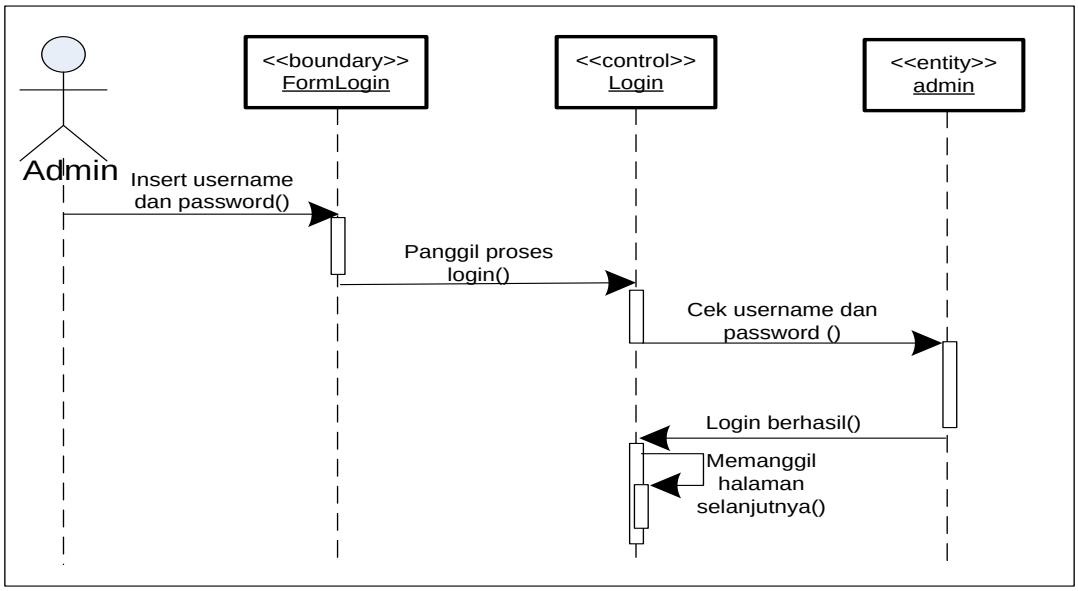
Sequence diagram Pesan Tiket menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses pemesanan tiket via sms. *Sequence diagram* pesan tiket dapat dilihat pada gambar 3.12.



Gambar 3.12 *Sequence Diagram* Pesan Tiket

3.1.11.9 *Sequence Diagram* Login Admin

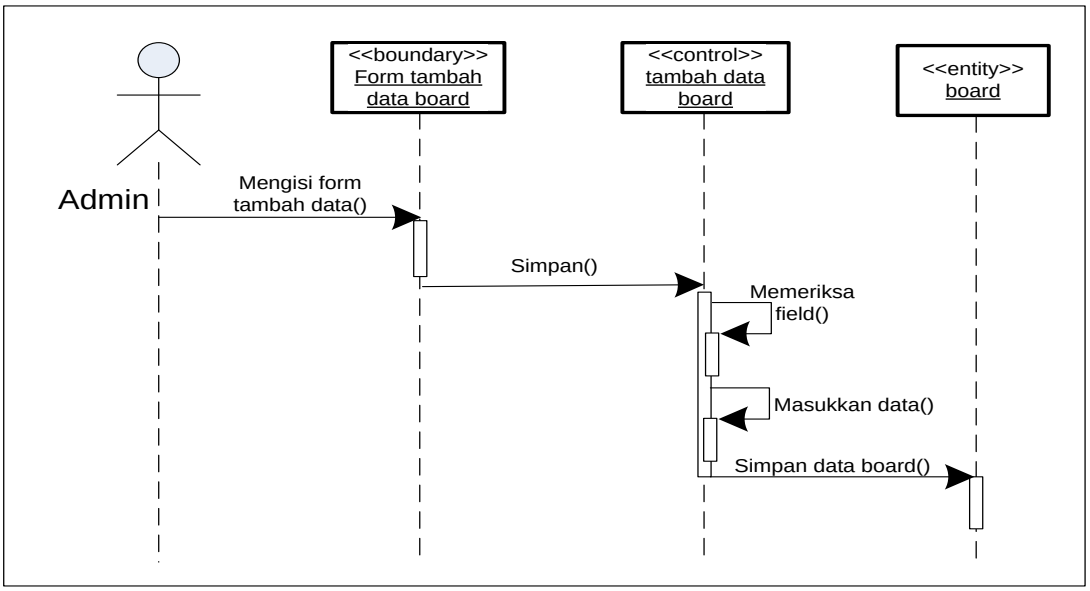
Sequence diagram Login Admin menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses login. *Sequence diagram* login admin dapat dilihat pada gambar 3.13.



Gambar 3.13 *Sequence Diagram* Login Admin

3.1.11.10 *Sequence Diagram* Tambah Data Board

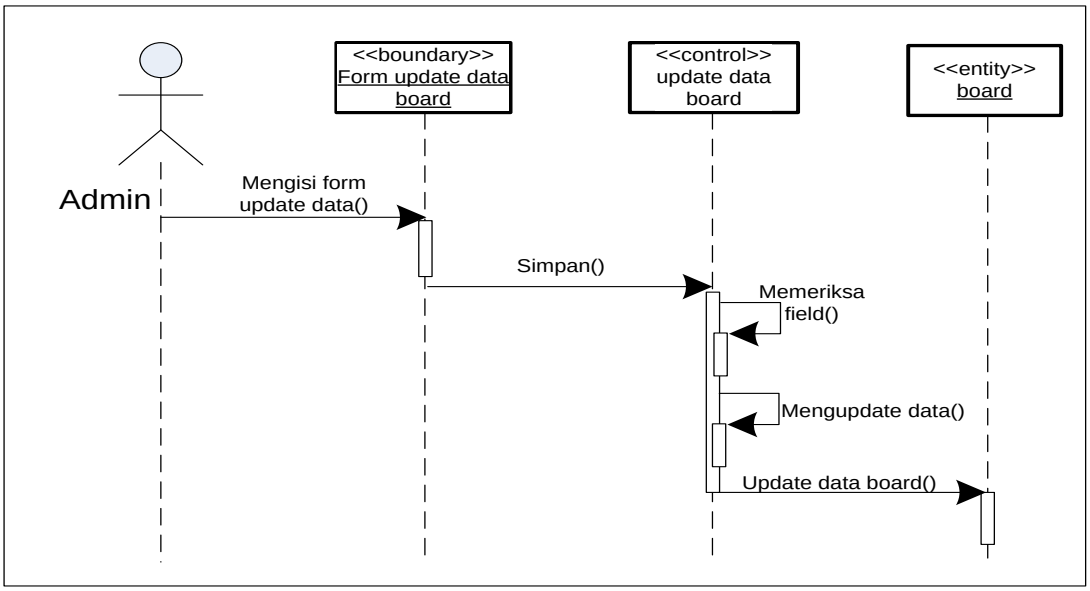
Sequence diagram tambah data board menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menambah data board. *Sequence diagram* tambah data board dapat dilihat pada gambar 3.14.



Gambar 3.14 *SequenceDiagram* Tambah Data Board

3.1.11.11 *Sequence Diagram Update* Data Board

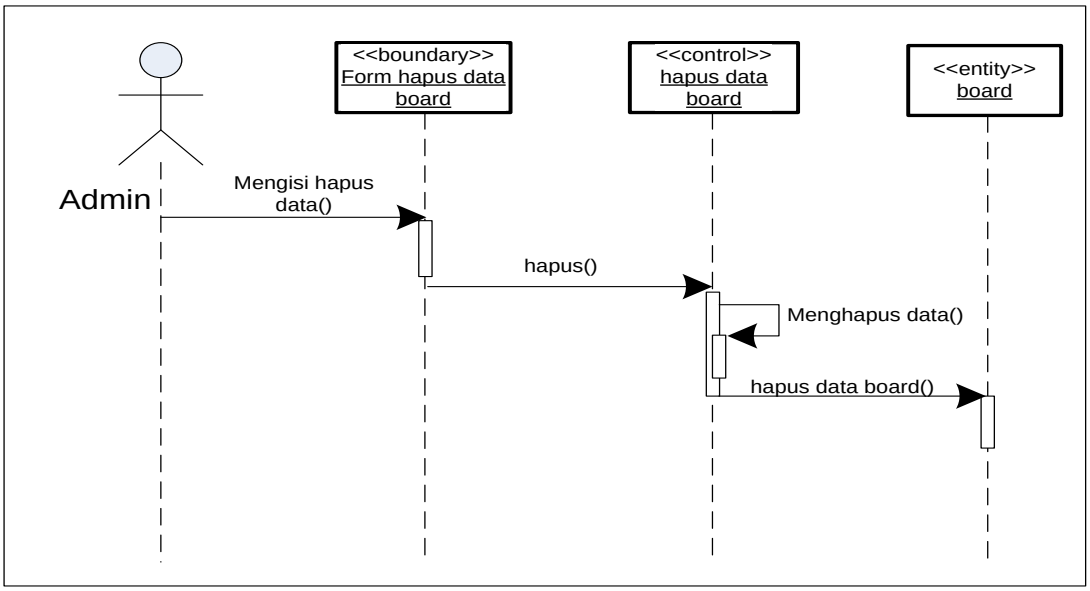
Sequence diagram update data board menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses meng-*update* data board. *Sequence diagram update* data board dapat dilihat pada gambar 3.15.



Gambar 3.15 *Sequence Diagram* Update Data Board

3.1.11.12 *Sequence Diagram* Hapus Data Board

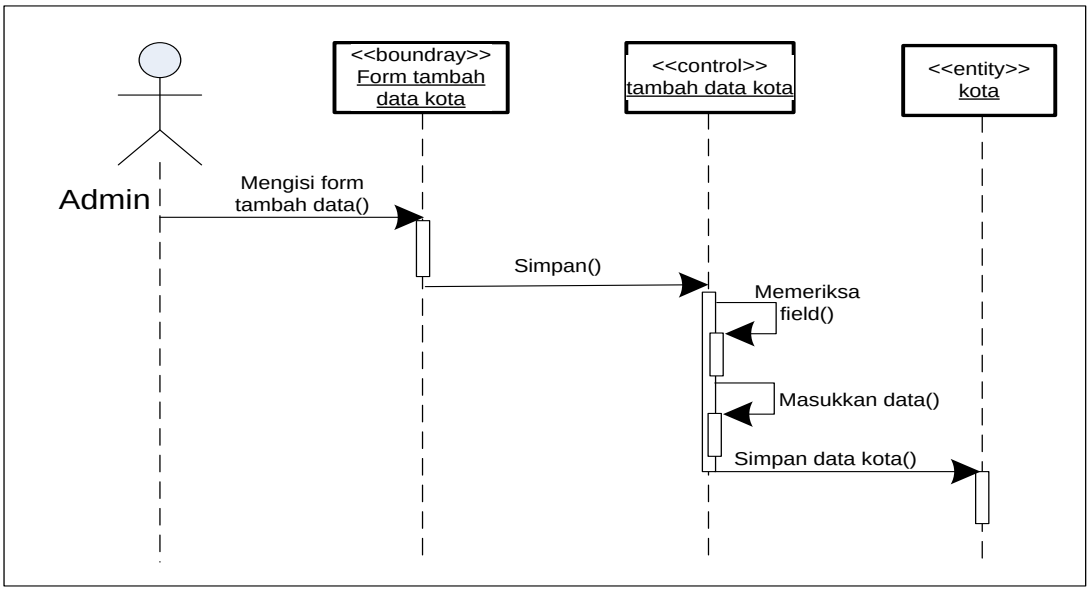
Sequence diagram hapus data board menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menghapus data board. *Sequence diagram* hapus data board dapat dilihat pada gambar 3.16.



Gambar 3.16 *Sequence Diagram* Hapus Data Board

3.1.11.13 *Sequence Diagram* Tambah Data Kota

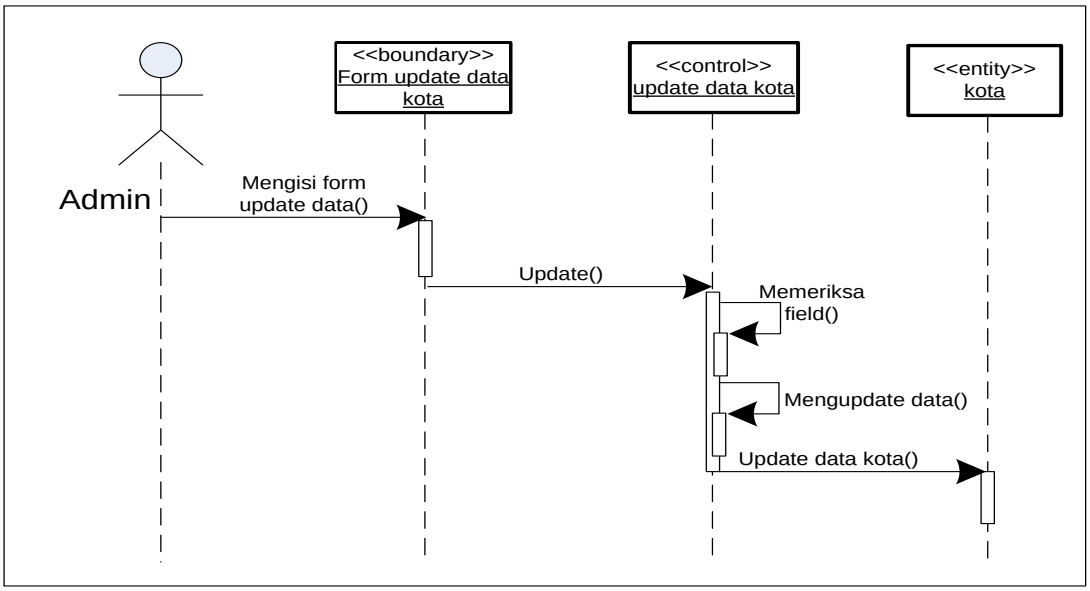
Sequence diagram Tambah Data Kota menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menambah data kota. *Sequence diagram* tambah data kota dapat dilihat pada gambar 3.17.



Gambar 3.17 *Sequence Diagram* Tambah Data Kota

3.1.11.14 *Sequence Diagram Update* Data Kota

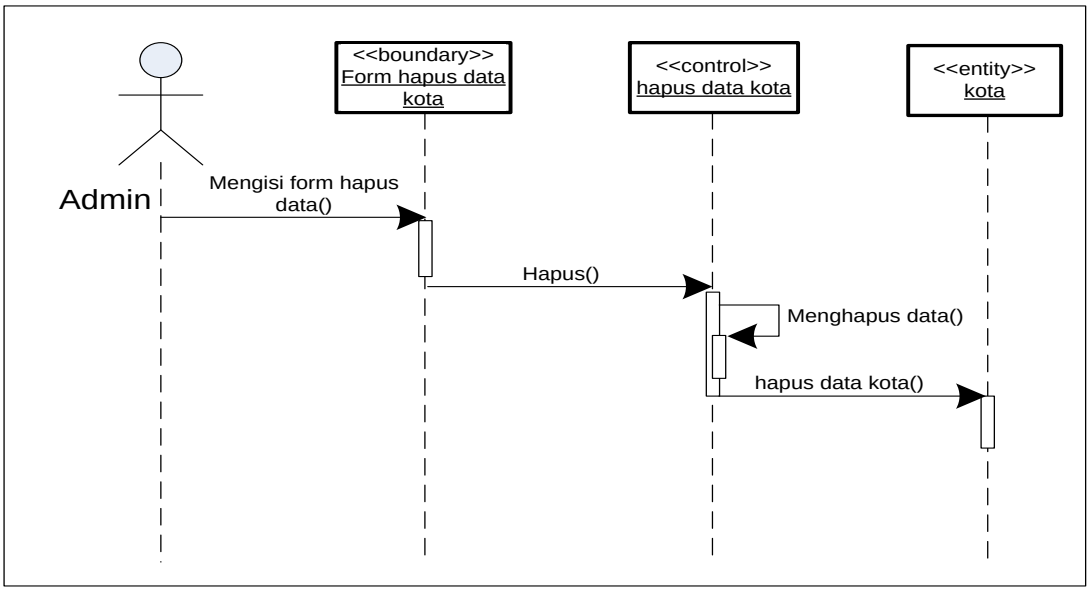
Sequence diagram Update Data Kota menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses meng-*update* data kota. *Sequence diagram update* data kota dapat dilihat pada gambar 3.18.



Gambar 3.18 *Sequence Diagram* Update Data Kota

3.1.11.15 *Sequence Diagram* Hapus Data Kota

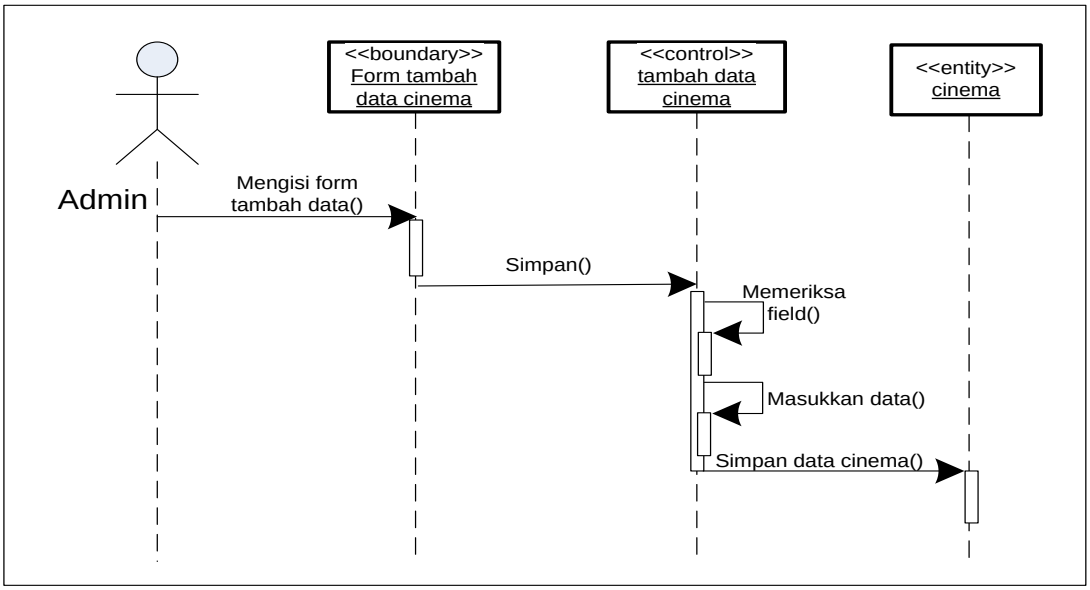
Sequence diagram Hapus Data Kota menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menghapus data kota. *Sequence diagram* hapus data kota dapat dilihat pada gambar 3.19.



Gambar 3.19 *Sequence Diagram* Hapus Data Kota

3.1.11.16 *Sequence Diagram* Tambah Data Cinema

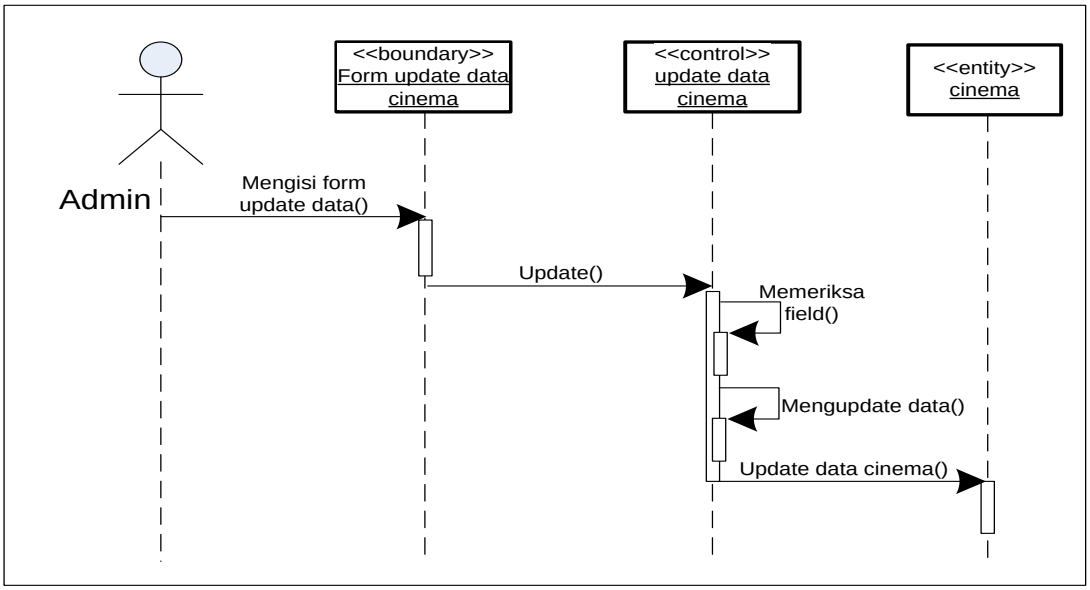
Sequence diagram Tambah Data Cinema menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menambah data cinema. *Sequence diagram* tambah data cinema dapat dilihat pada gambar 3.20.



Gambar 3.20 *Sequence Diagram* Tambah Data Cinema

3.1.11.17 *Sequence Diagram Update* Data Cinema

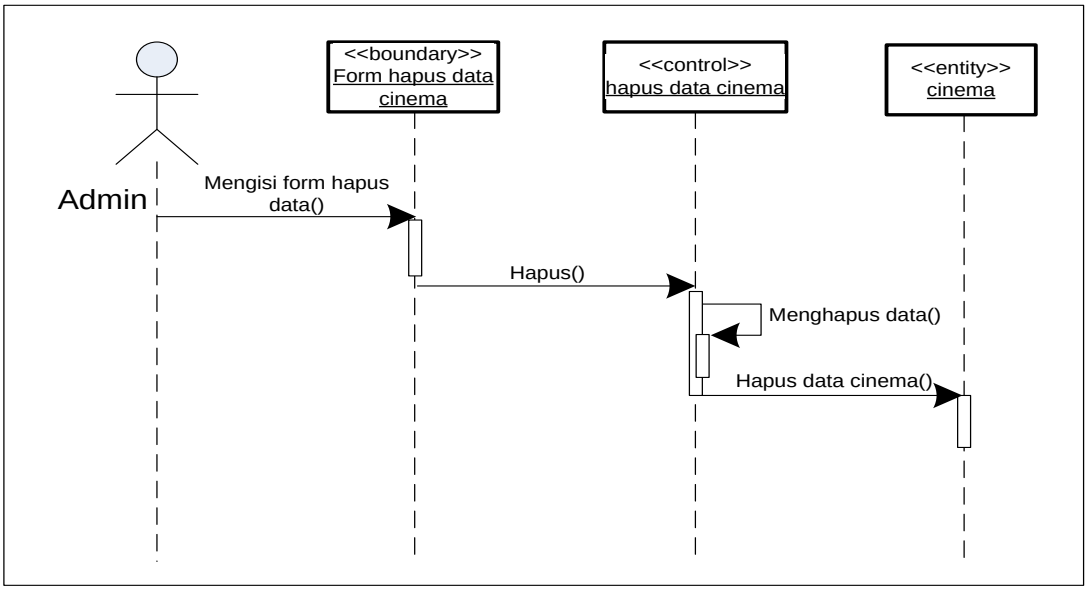
Sequence diagram Update Data Cinema menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses meng-*update* data cinema. *Sequence diagram update* data cinema dapat dilihat pada gambar 3.21.



Gambar 3.21 *Sequence Diagram Update* Data Cinema

3.1.11.18 *Sequence Diagram* Hapus Data Cinema

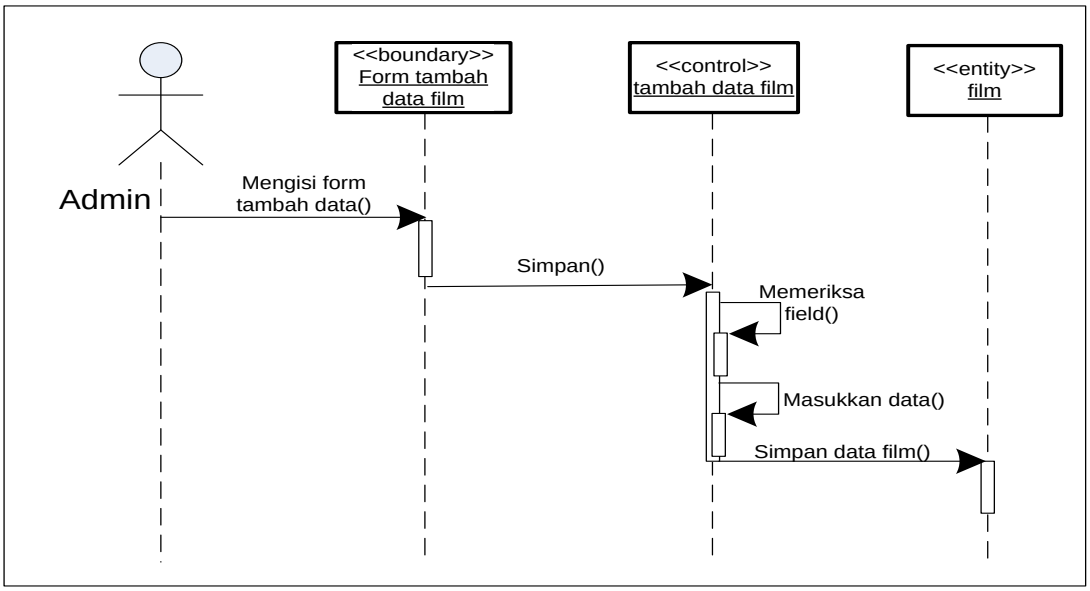
Sequence diagram Hapus Data Cinema menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menghapus data cinema. *Sequence diagram* hapus data cinema dapat dilihat pada gambar 3.22.



Gambar 3.22 *Sequence Diagram* Hapus Data Cinema

3.1.11.19 *Sequence Diagram* Tambah Data Film

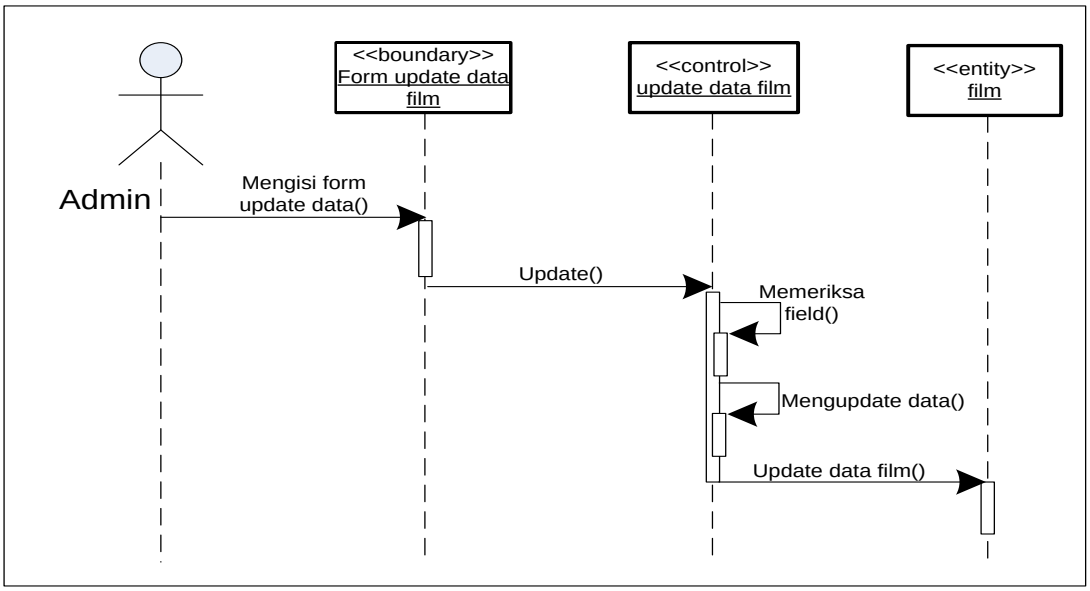
Sequence diagram Tambah Data Film menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menambah data film. *Sequence diagram* tambah data film dapat dilihat pada gambar 3.23.



Gambar 3.23 *Sequence Diagram* Tambah Data Film

3.1.11.20 *Sequence Diagram Update* Data Film

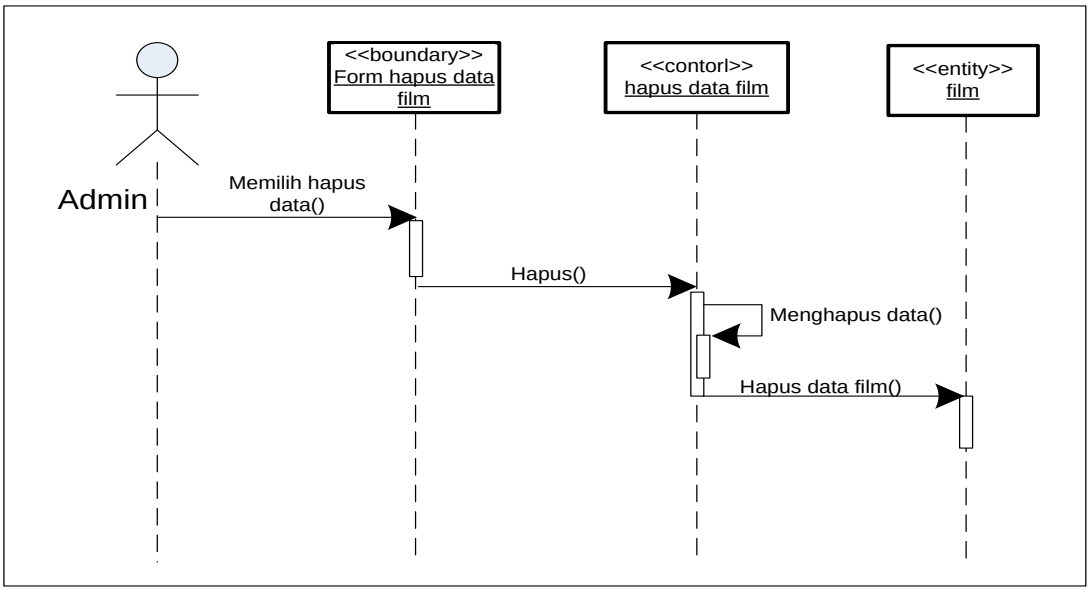
Sequence diagram Update Data Film menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses meng-*update* data film. *Sequence diagram update* data film dapat dilihat pada gambar 3.24.



Gambar 3.24 *Sequence Diagram Update* Data Film

3.1.11.21 *Sequence Diagram* Hapus Data Film

Sequence diagram hapus data film menunjukkan rangkaian pesan yang dikirim antara objek admin dengan objek lainnya dalam proses menghapus data film. *Sequence diagram* hapus data film dapat dilihat pada gambar 3.25.

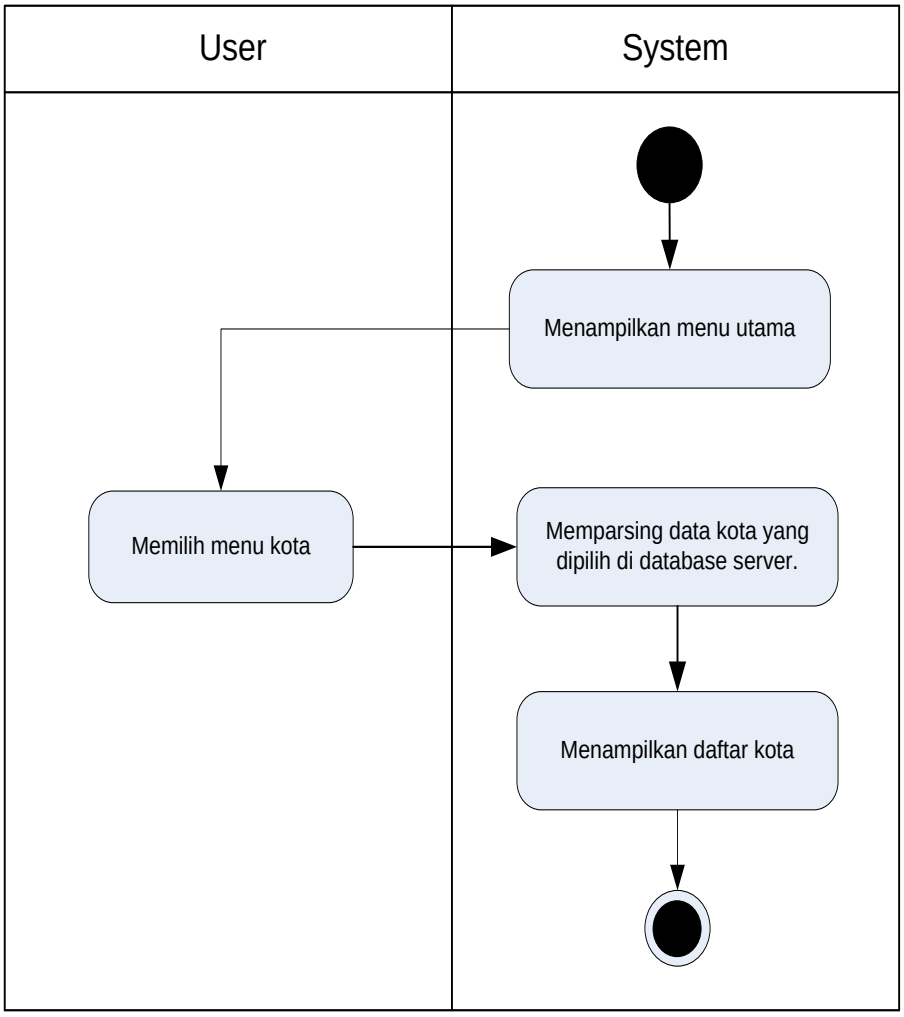


Gambar 3.25 *Sequence Diagram* Hapus Data Film

3.1.12. *Activity Diagram*

3.1.12.1 *Activity Diagram* Daftar Kota

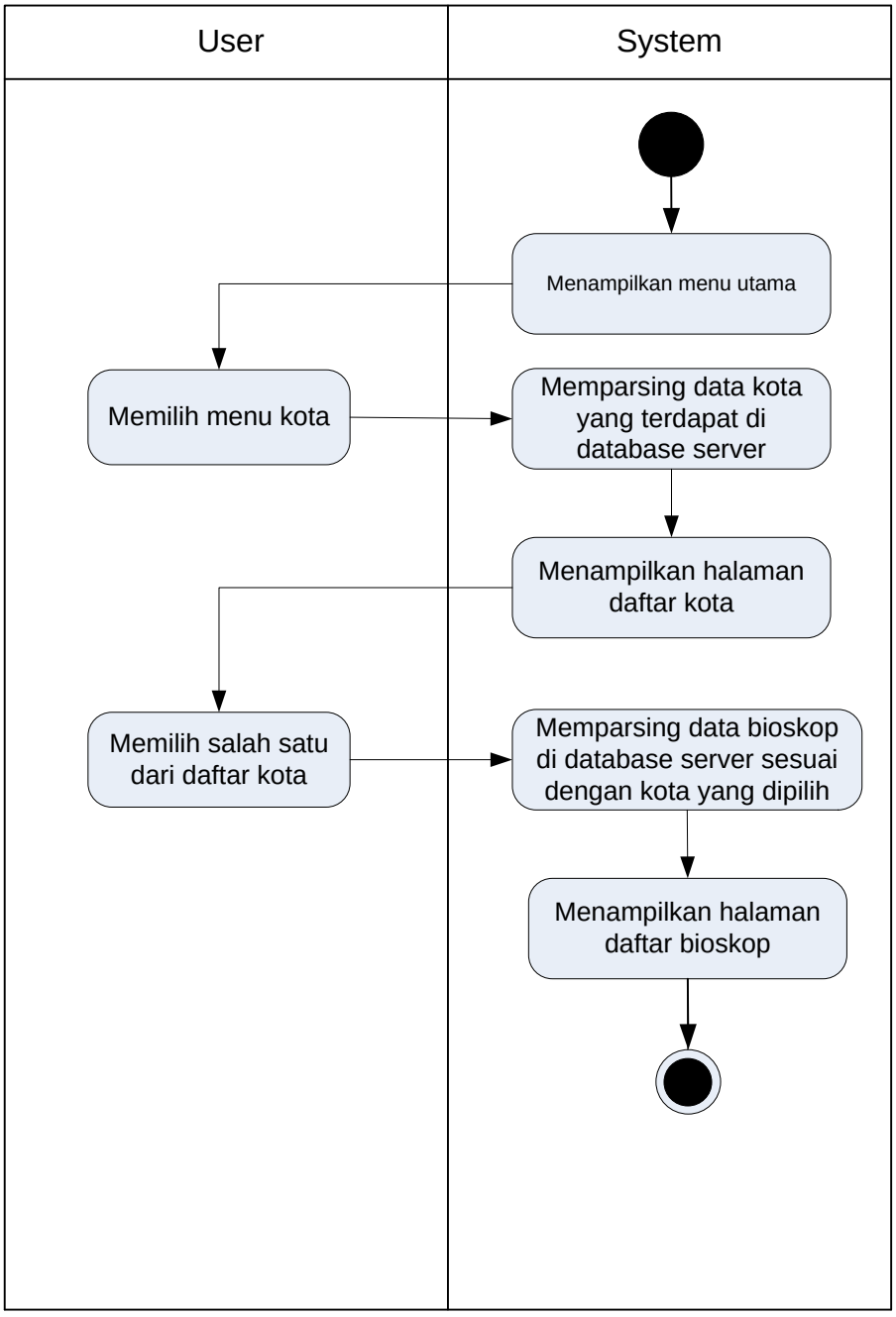
Activity diagram daftar kota menjelaskan aliran kerja aktor pengguna pada saat memilih menu kota pada halaman menu utama. *Activity diagram* daftar kota dapat dilihat pada gambar 3.26.



Gambar 3.26 *Activity Diagram* Daftar Kota

3.1.12.2 *Activity Diagram* Daftar Bioskop

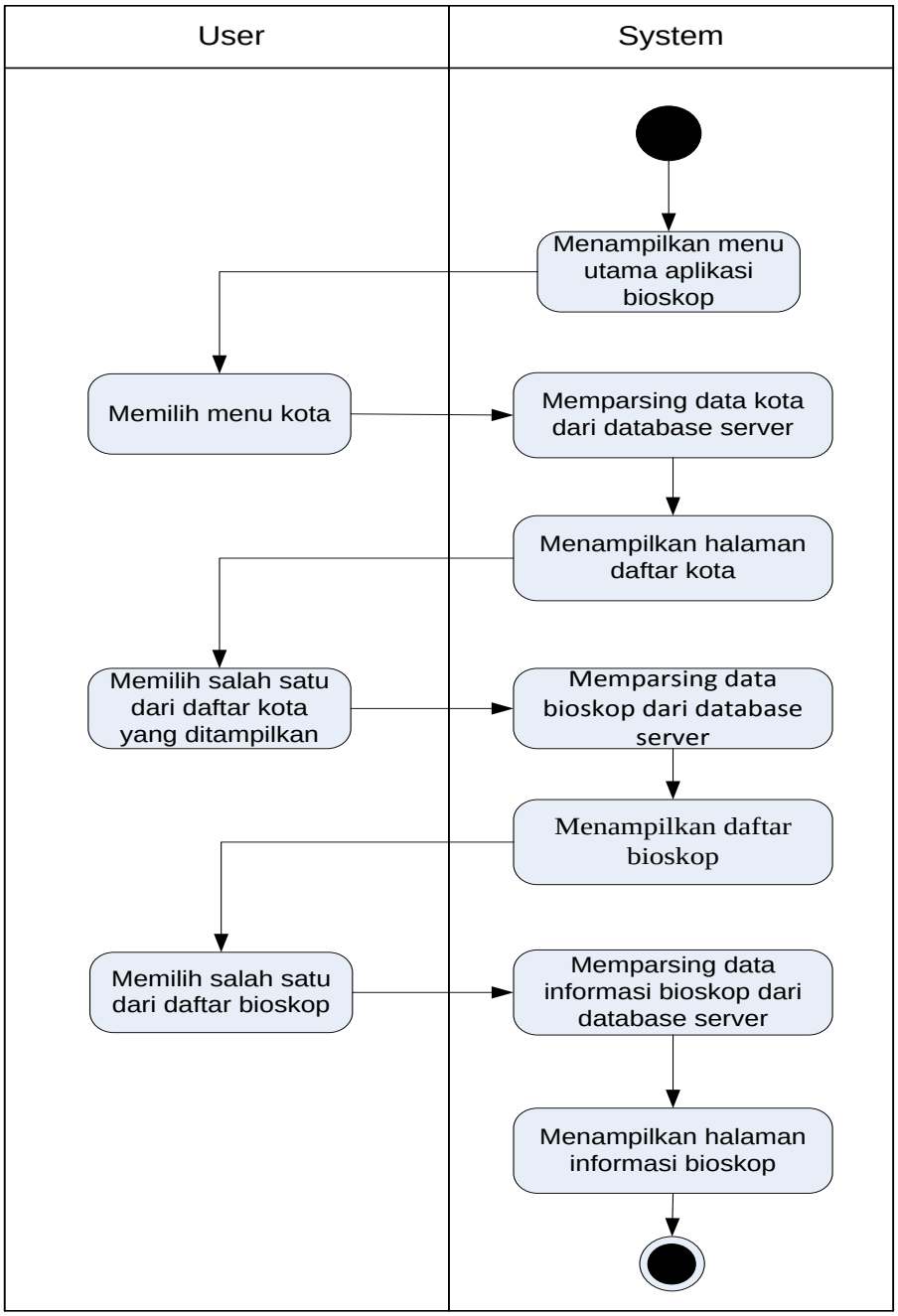
Activity diagram daftar bioskop menjelaskan aliran kerja aktor pengguna pada saat memilih cinema dari daftar kota yang ditampilkan. *Activity diagram* melihat daftar bioskop dapat dilihat pada gambar 3.27.



Gambar 3.27 *Activity Diagram* Daftar Bioskop

3.1.12.3 *Activity Diagram* Informasi Dioskop

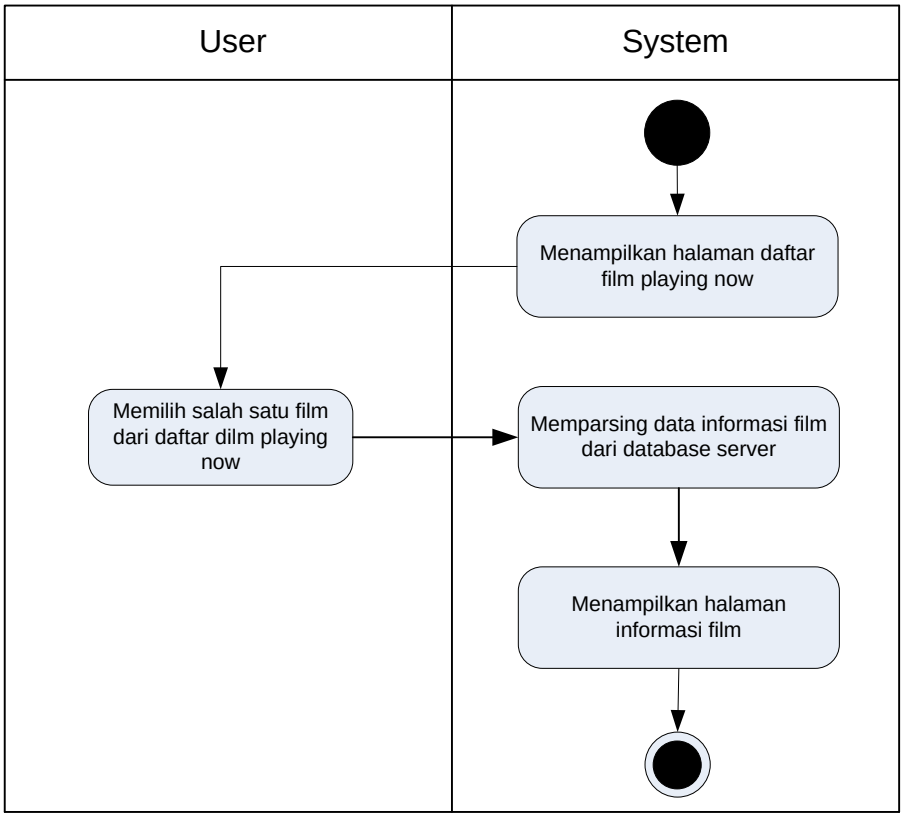
Activity diagram informasi bioskop menjelaskan aliran kerja aktor pengguna pada saat memilih melihat informasi bioskop yang dipilih dari daftar bioskop. *Activity diagram* informasi bioskop dapat dilihat pada gambar 3.28.



Gambar 3.28 *Activity Diagram* Informasi Bioskop

3.1.12.4 *Activity Diagram* Informasi Film

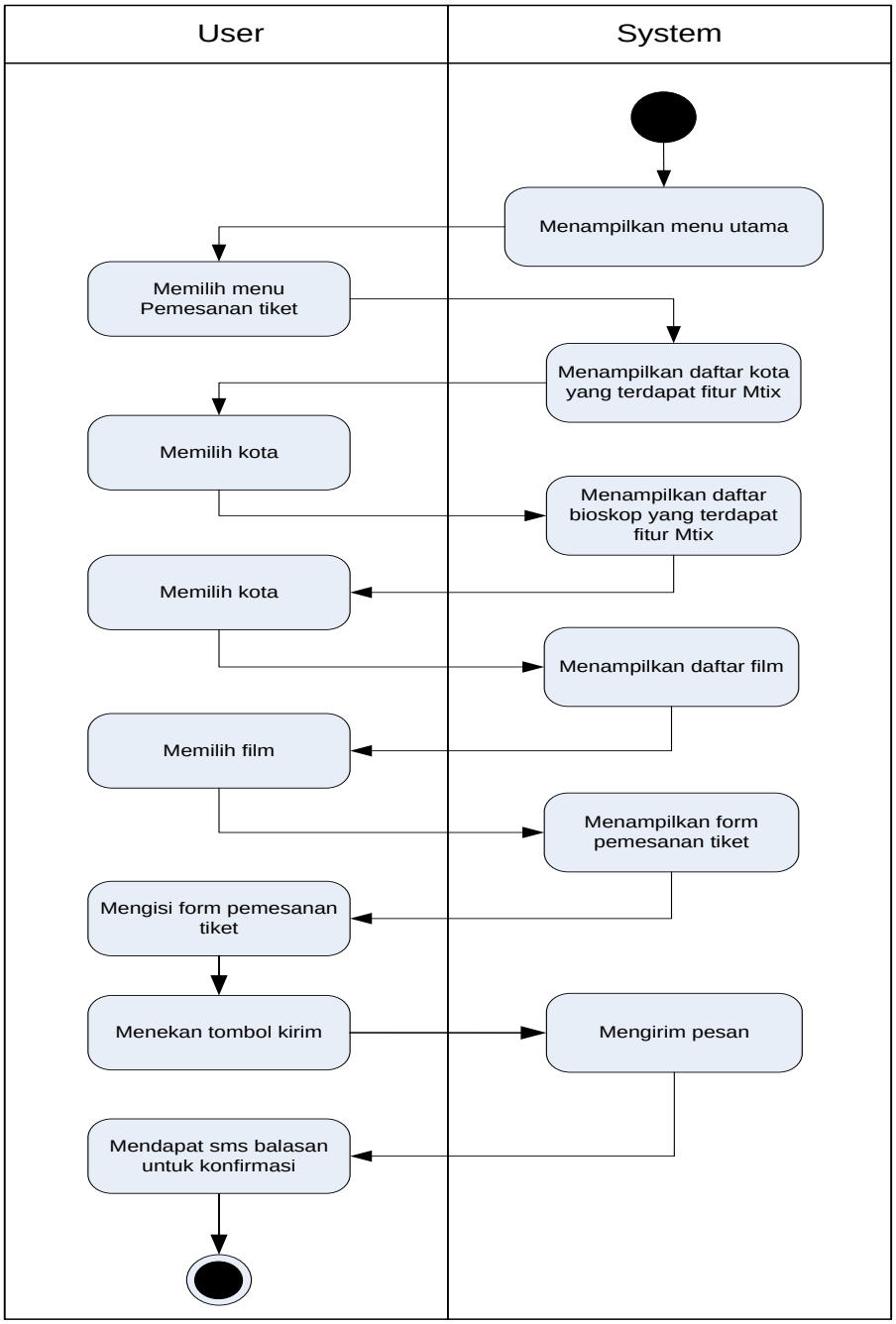
Activity diagram informasi film menjelaskan aliran kerja aktor pengguna pada saat melihat informasi dari film yang dipilih. *Activity diagram* informasi film dapat dilihat pada gambar 3.29.



Gambar 3.29 *Activity Diagram* Informasi Film

3.1.12.5 *Activity Diagram* Pesan Tiket

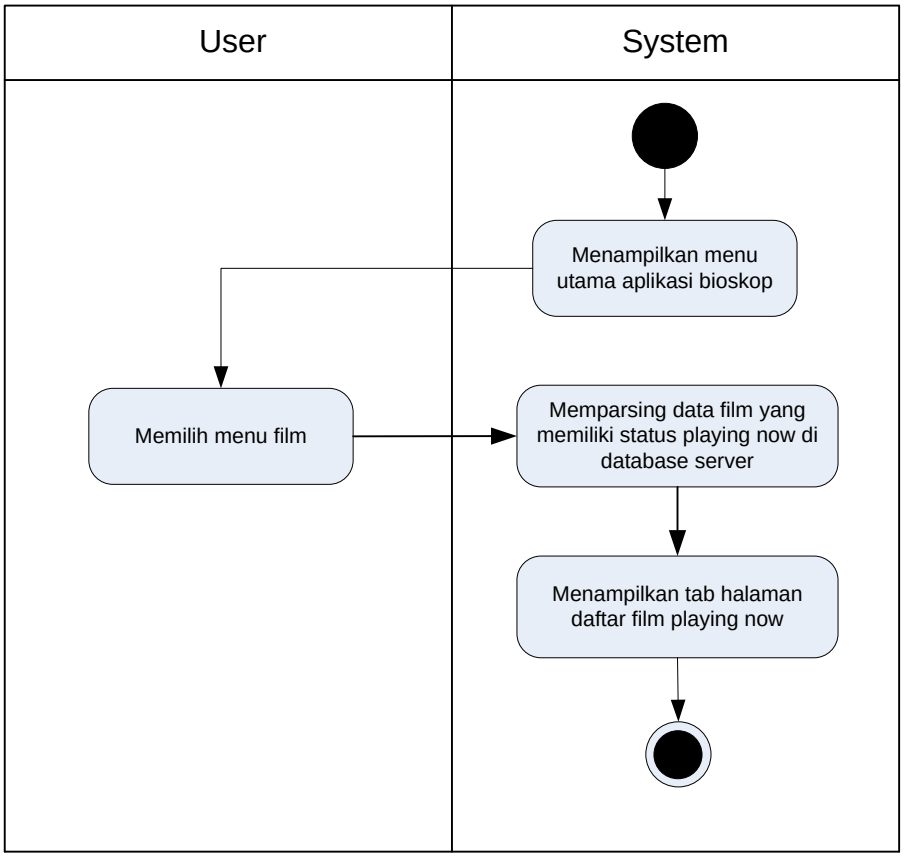
Activity diagram pesan tiket menjelaskan aliran kerja aktor pengguna pada saat melakukan pemesanan tiket via sms. *Activity diagram* pesan tiket dapat dilihat pada gambar 3.30.



Gambar 3.30 *Activity Diagram* Pesan Tiket

3.1.12.6 *Activity Diagram* Film *Playing Now*

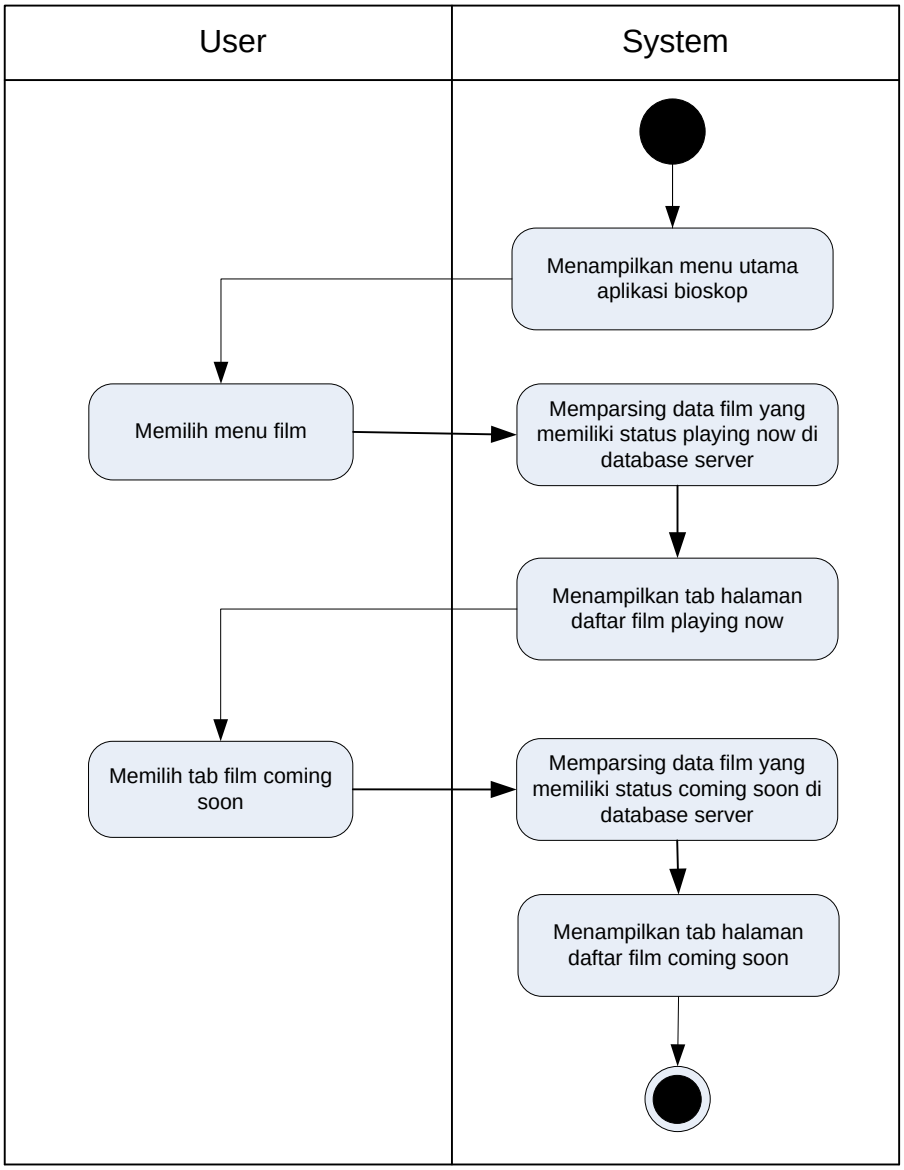
Activity diagram film *playing now* menjelaskan aliran kerja aktor pengguna pada saat melihat daftar film yang sedang tayang. *Activity diagram* film *playing now* dapat dilihat pada gambar 3.31.



Gambar 3.31 *Activity Diagram* Film *Playing Now*

3.1.12.7 *Activity Diagram* Film *Coming Soon*

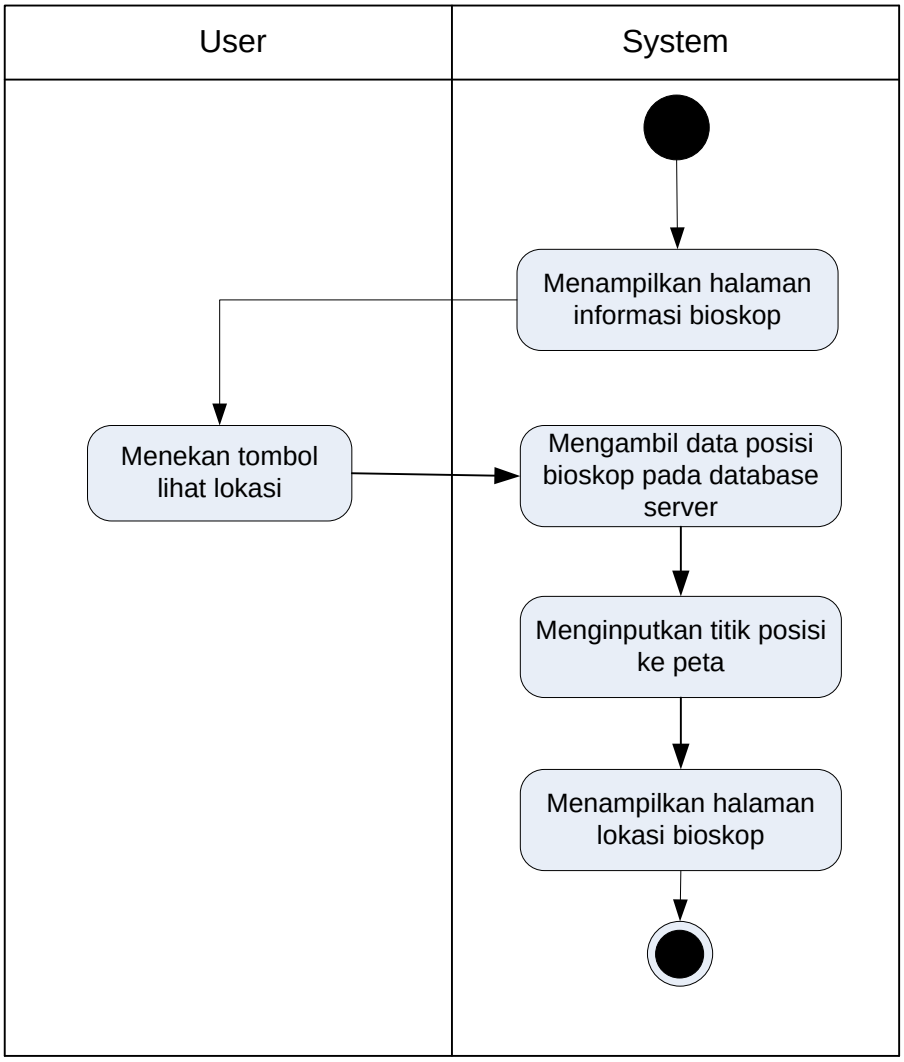
Activity diagram film *coming soon* menjelaskan aliran kerja aktor pengguna pada saat melihat daftar film yang akan tayang. *Activity diagram* film *coming soon* dapat dilihat pada gambar 3.32.



Gambar 3.32 *Activity Diagram* Film *Coming Soon*

3.1.12.8 *Activity Diagram* Lokasi

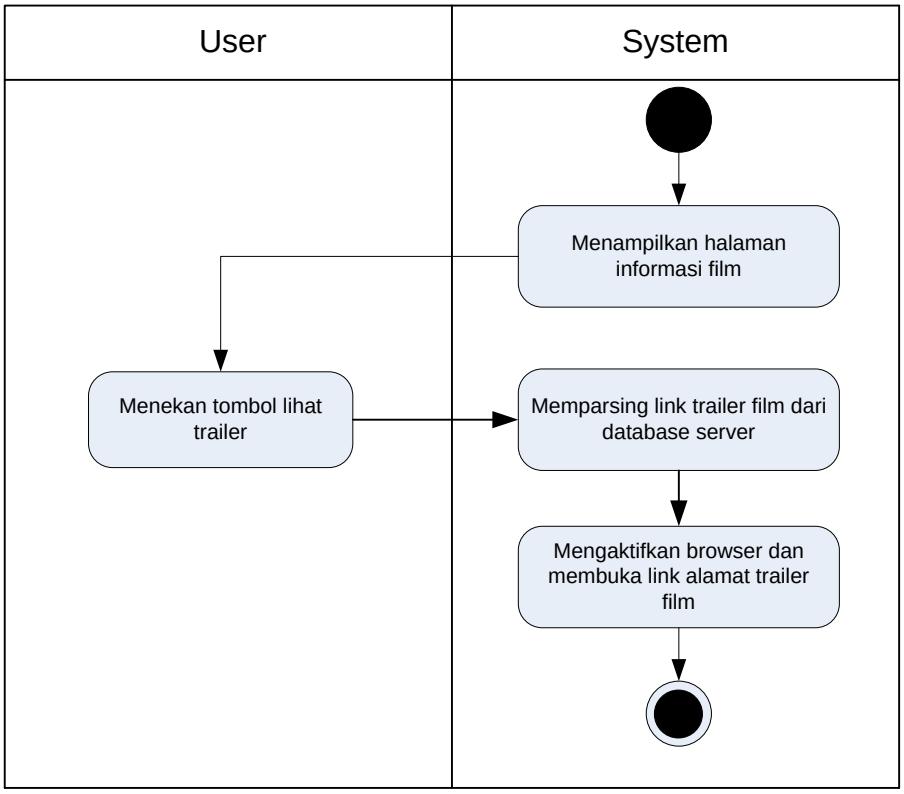
Activity diagram lokasi menjelaskan aliran kerja aktor pengguna pada saat melihat lokasi dari bioskop yang dipilih. *Activity diagram* lokasi dapat dilihat pada gambar 3.33.



Gambar 3.33 *Activity Diagram* Lokasi

3.1.12.9 *Activity DiagramTrailer* Film

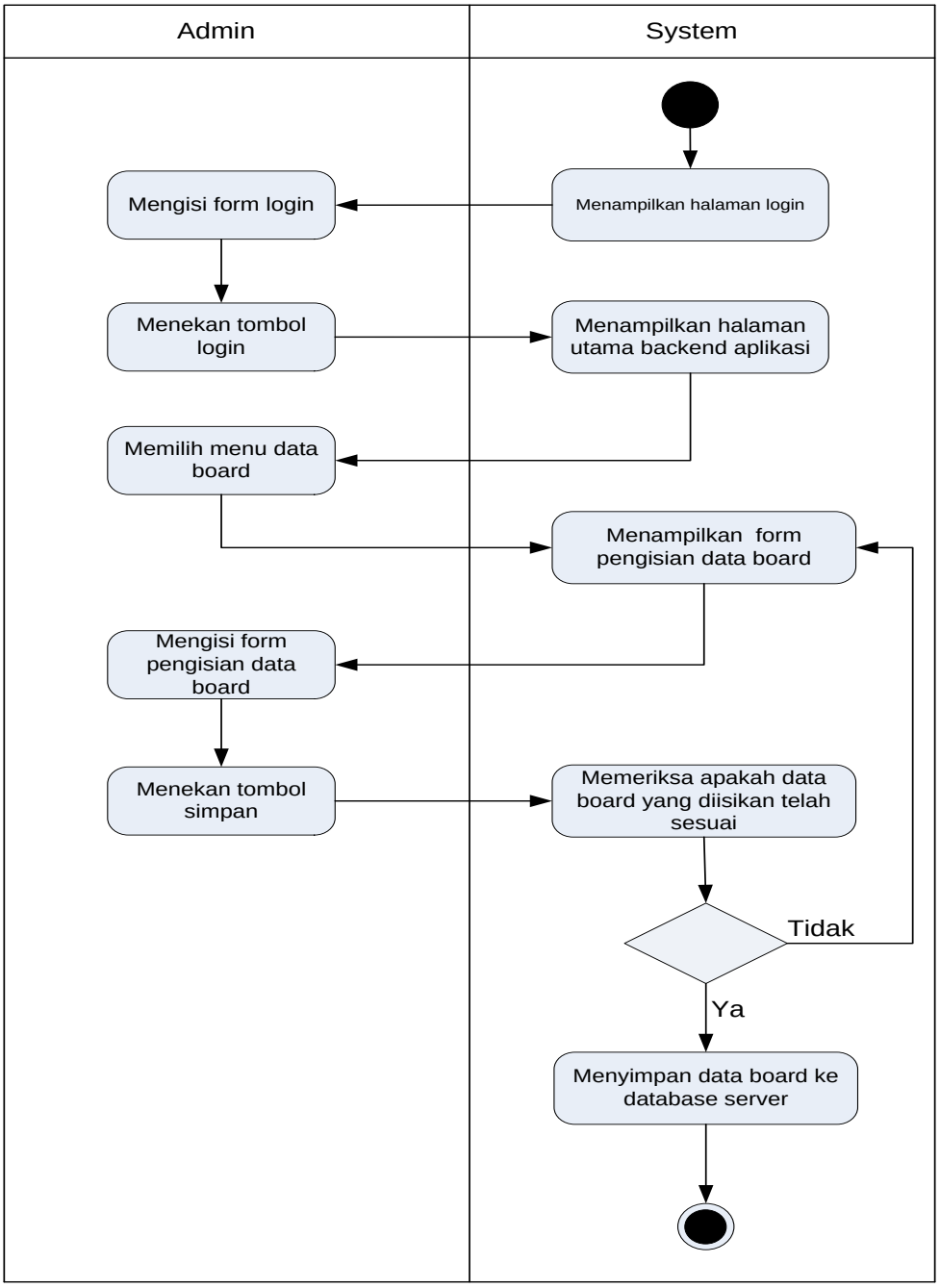
Activity diagram trailer film menjelaskan aliran kerja aktor pengguna pada saat melihat cuplikan dari film yang dipilih. *Activity diagram trailer* film dapat dilihat pada gambar 3.34.



Gambar 3.34 *Activity Diagram Trailer* Film

3.1.12.10 *Activity Diagram* Tambah Data Board

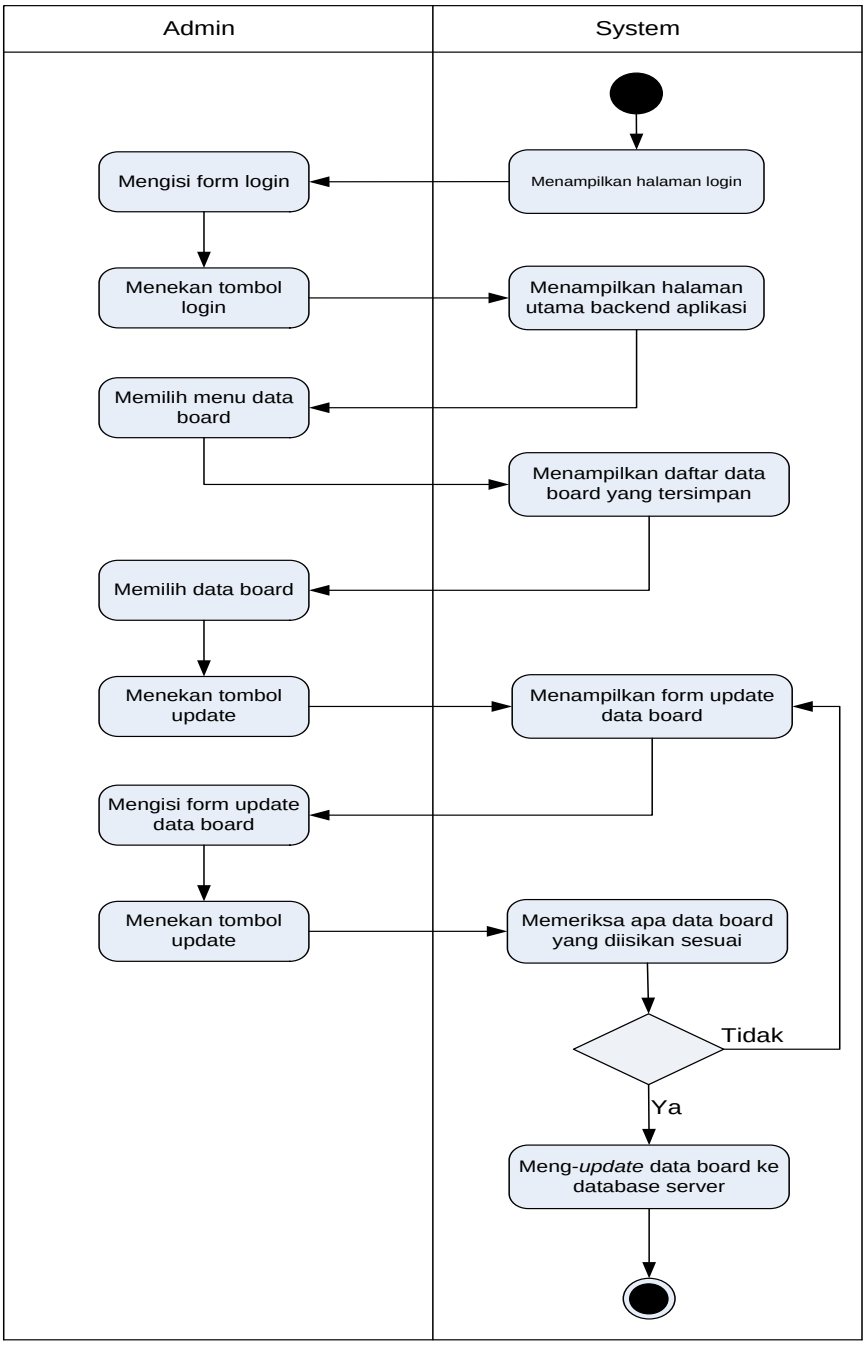
Activity diagram tambah data board menjelaskan aliran kerja aktor admin pada saat melakukan penambahan data board ke *database server*. *Activity diagram* tambah data board dapat dilihat pada gambar 3.35.



Gambar 3.35 *Activity Diagram* Tambah Data Board

3.1.12.11 *Activity Diagram Update* Data Board

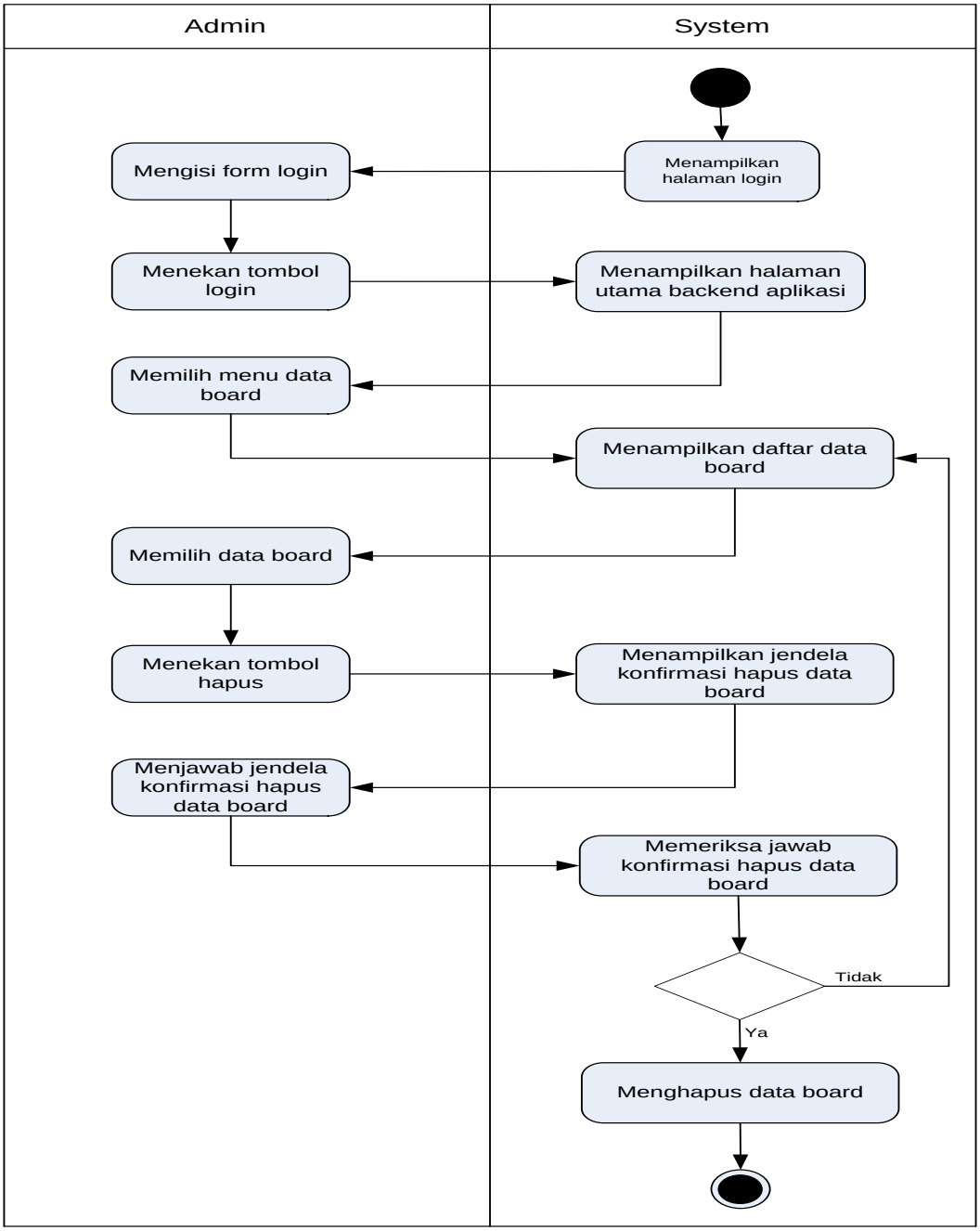
Activity diagram update data board menjelaskan aliran kerja aktor admin pada saat melakukan peng-*update*-an data board yang telah tersimpan di *database server*. *Activity diagram update* data board dapat dilihat pada gambar 3.36.



Gambar 3.36 *Activty Diagram Update* Data Board

3.1.12.12 *Activity Diagram* Hapus Data Board

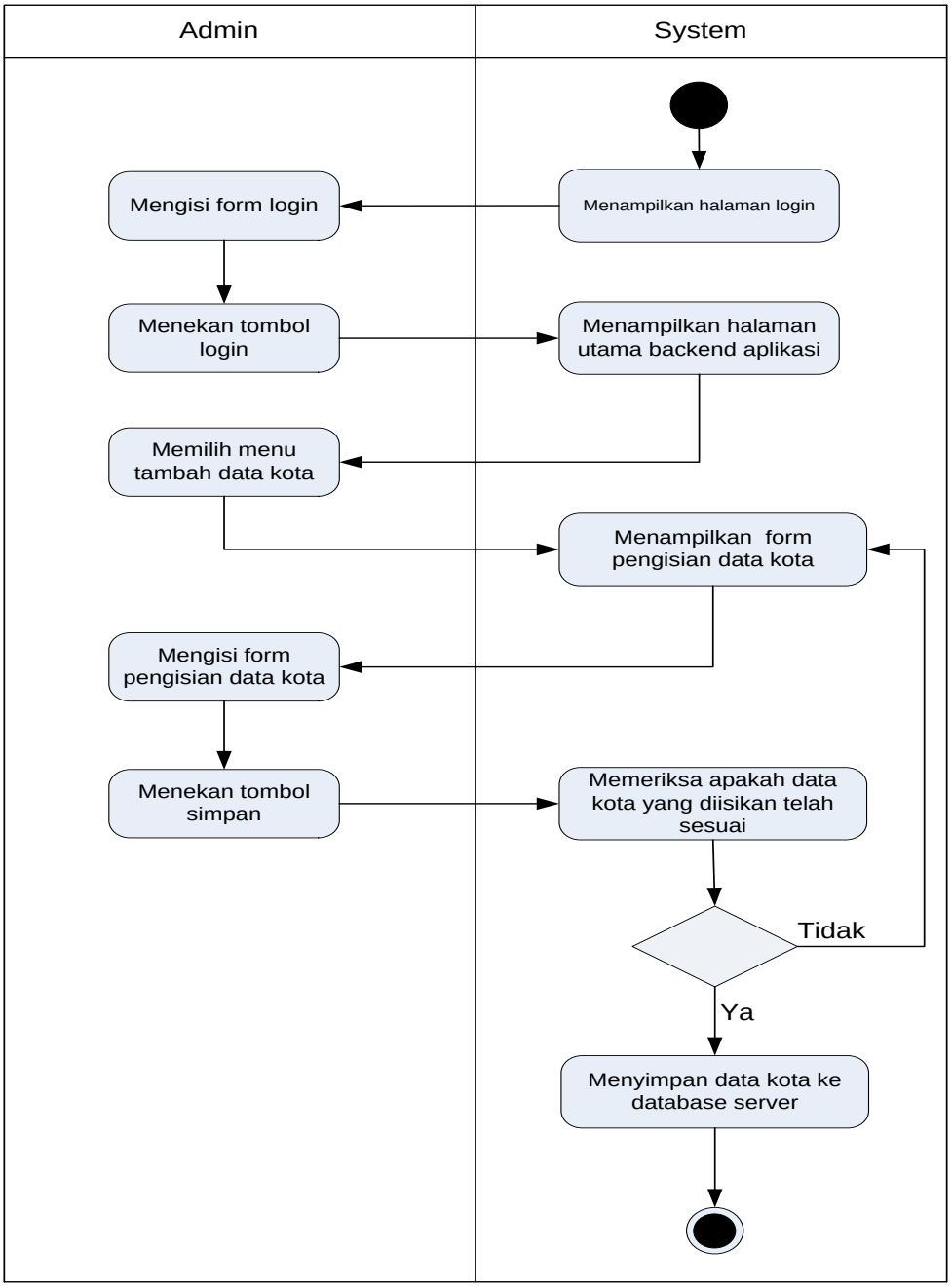
Activity diagram hapus data board menjelaskan aliran kerja aktor admin pada saat melakukan penghapusan data board dari *database server*. *Activity diagram* hapus data board dapat dilihat pada gambar 3.37.



Gambar 3.37 *Activty Diagram* Hapus Data Board

3.1.12.13 *Activity Diagram* Tambah Data Kota

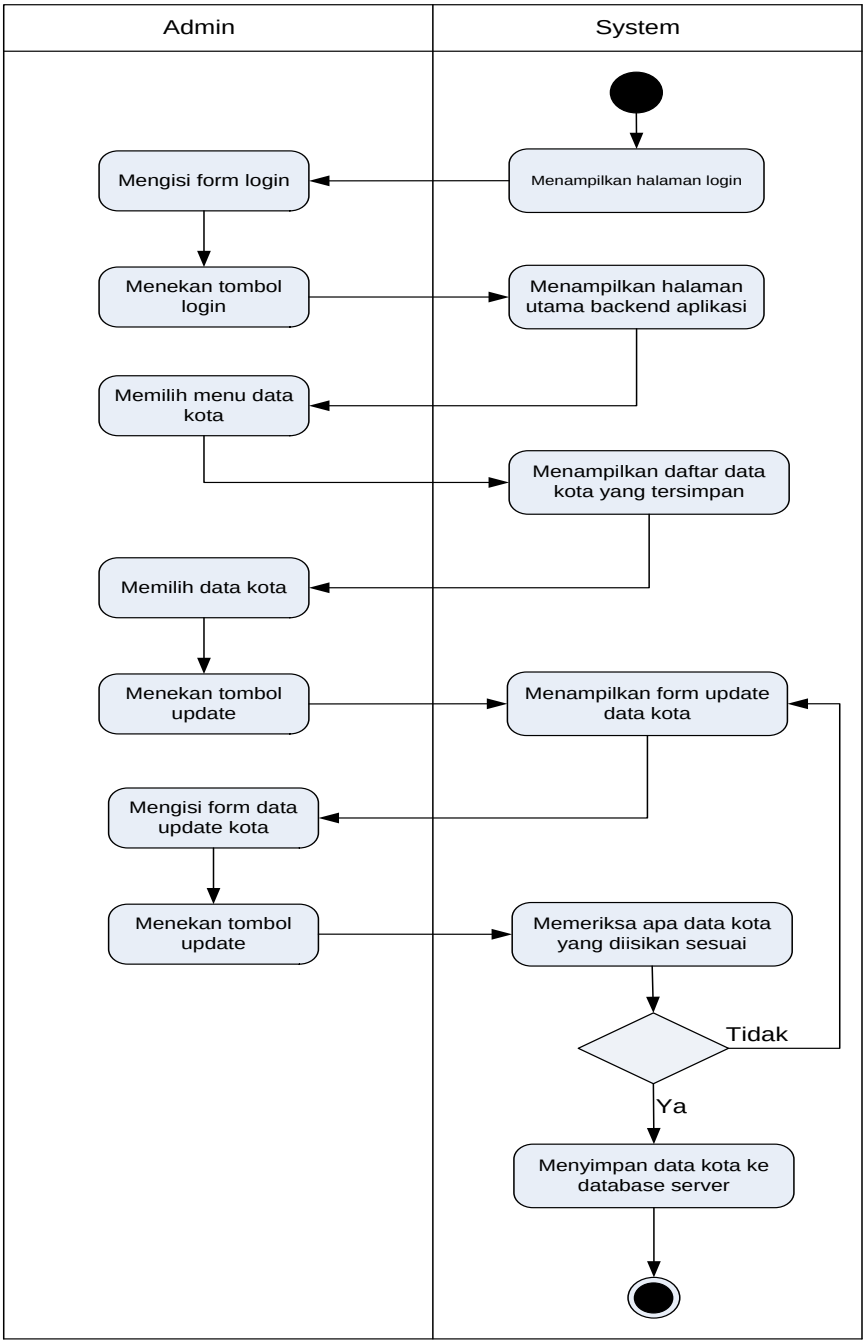
Activity diagram tambah data kota menjelaskan aliran kerja aktor admin pada saat melakukan penambahan data kota ke *database server*. *Activity diagram* tambah data kota dapat dilihat pada gambar 3.38.



Gambar 3.38 *Activty Diagram* Tambah Data Kota

3.1.12.14 *Activity Diagram Update* Data Kota

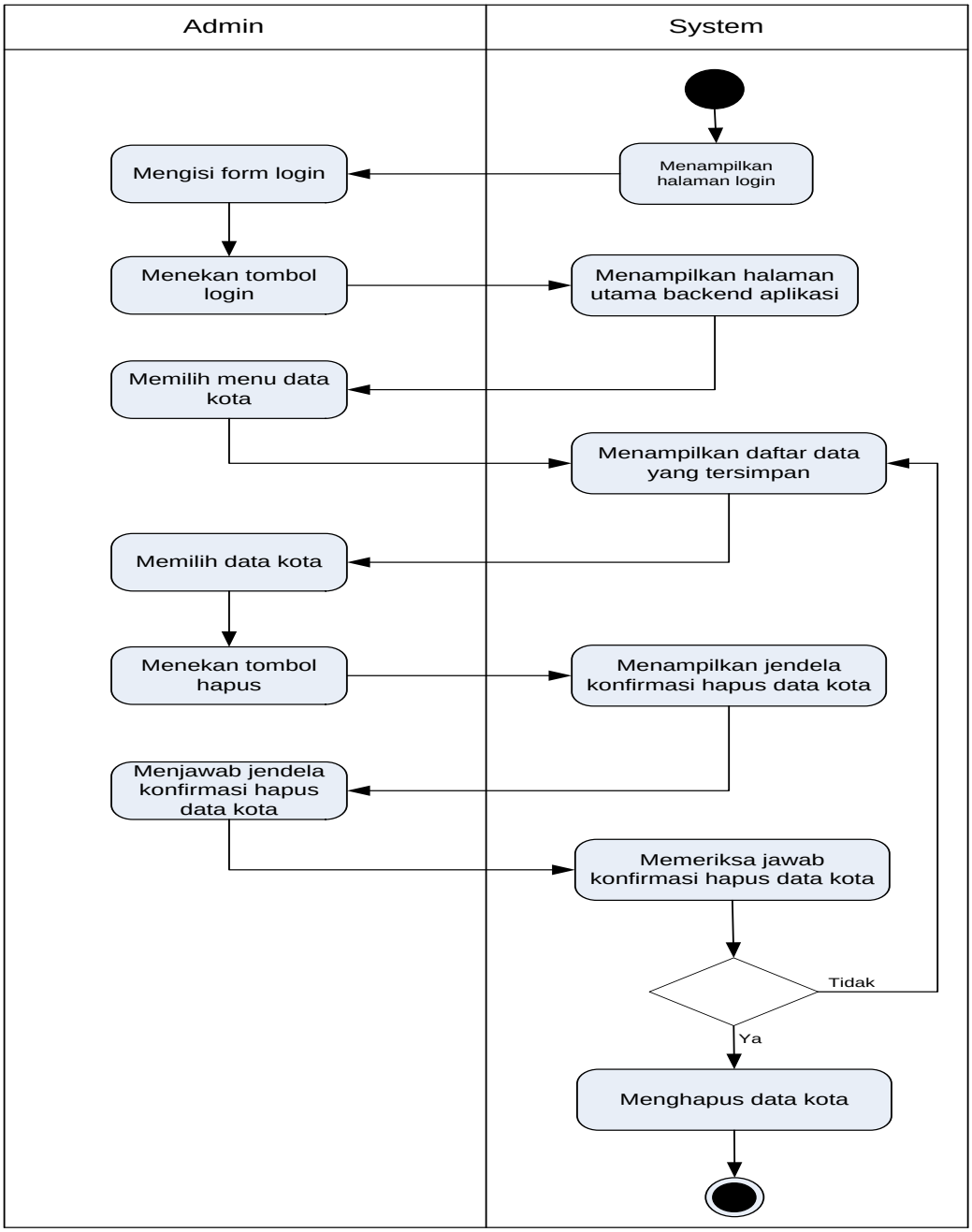
Activity diagram update data kota menjelaskan aliran kerja aktor admin pada saat melakukan peng-*update*-an data kota yang telah tersimpan di *database server*. *Activity diagram update* data kota dapat dilihat pada gambar 3.39.



Gambar 3.39 *Activty Diagram Update* Data Kota

3.1.12.15 *Activity Diagram* Hapus Data Kota

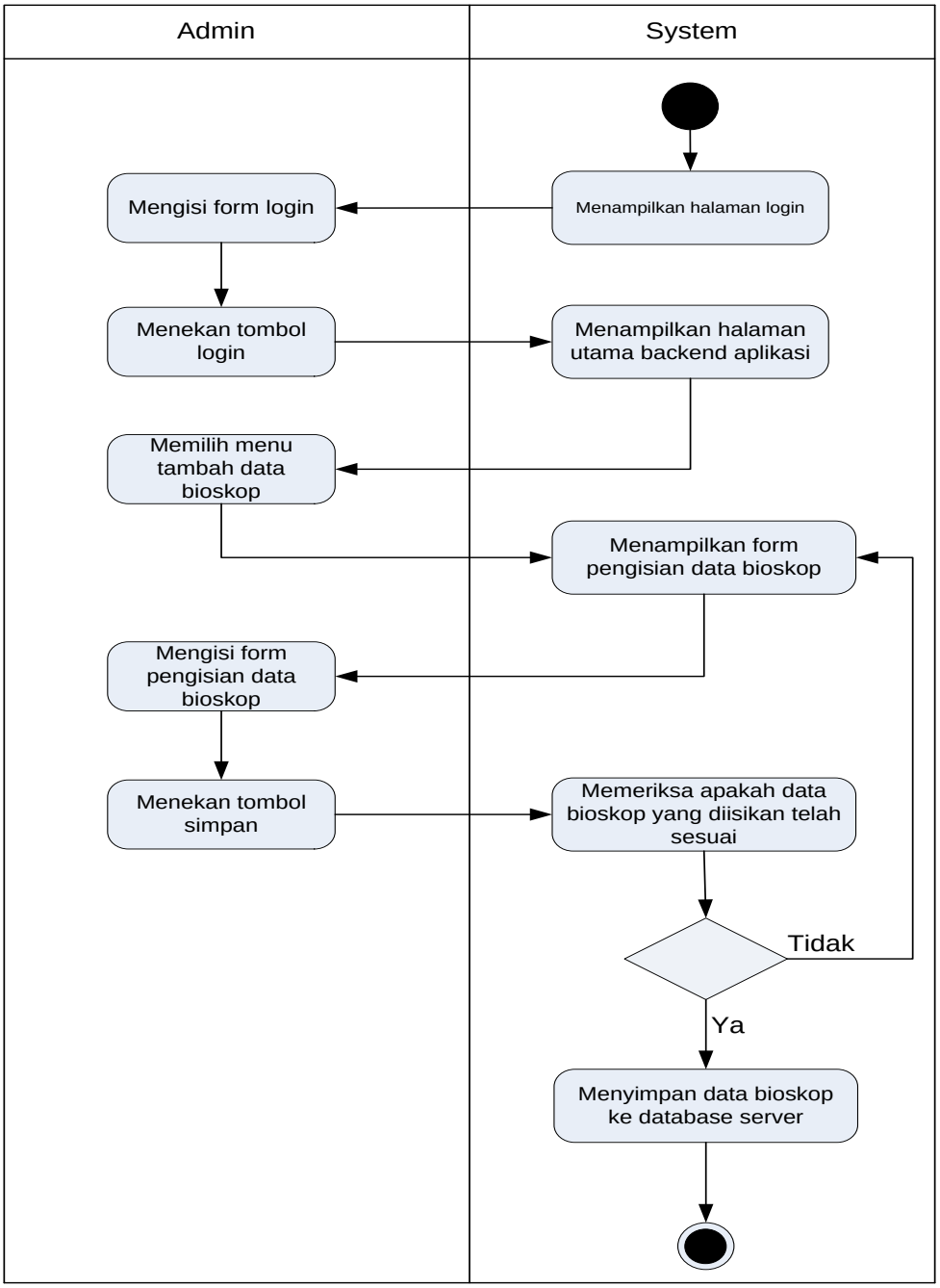
Activity diagram hapus data kota menjelaskan aliran kerja aktor admin pada saat melakukan penghapusan data kota dari *database server*. *Activity diagram* hapus data kota dapat dilihat pada gambar 3.40.



Gambar 3.40 *Activty Diagram* Hapus Data Kota

3.1.12.16 *Activity Diagram* Tambah Data Bioskop

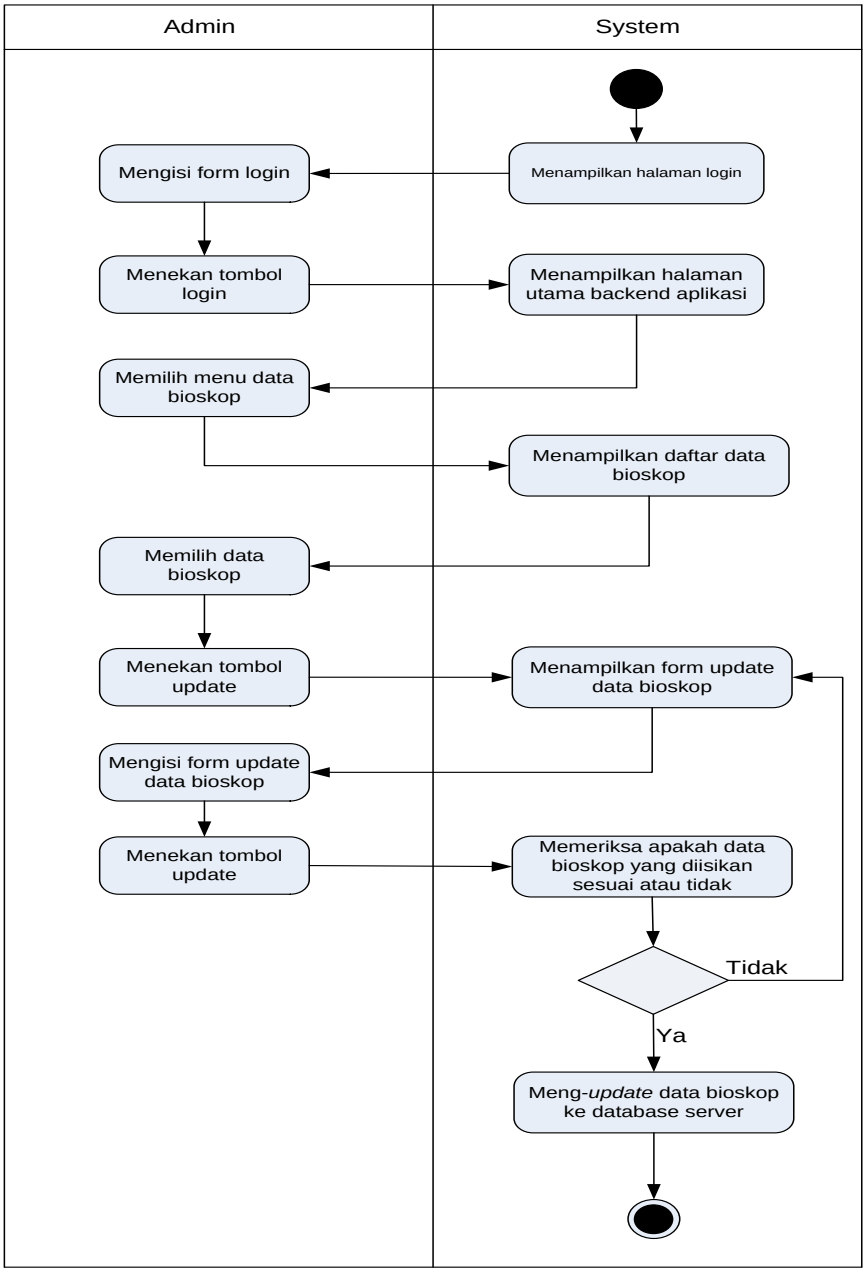
Activity diagram tambah data bioskop menjelaskan aliran kerja aktor admin pada saat melakukan penambahan data bioskop ke *database server*. *Activity diagram* tambah data bioskop dapat dilihat pada gambar 3.41.



Gambar 3.41 *Activty Diagram* Tambah Data Bioskop

3.1.12.17 *Activity Diagram Update* Data Bioskop

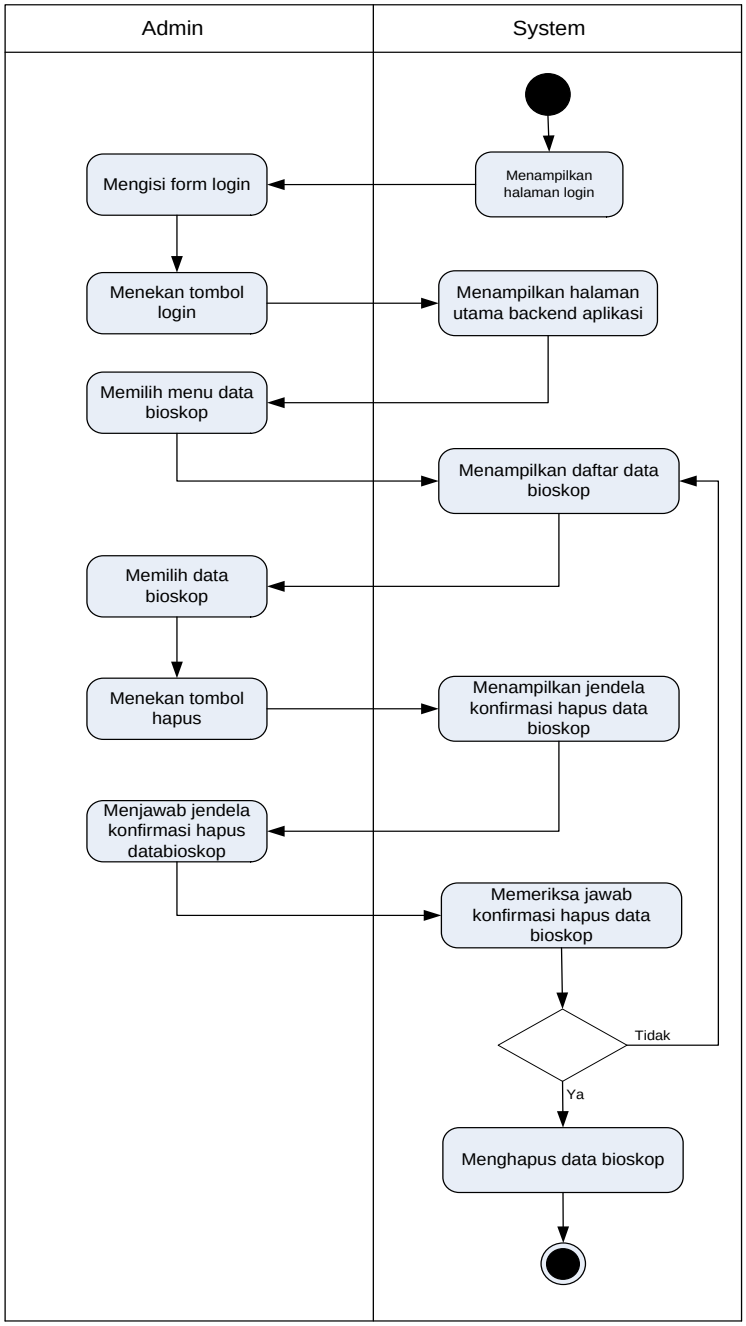
Activity diagram update data bioskop menjelaskan aliran kerja aktor admin pada saat melakukan peng-*update*-an data bioskop yang telah tersimpan di *database server*. *Activity diagram update* data bioskop dapat dilihat pada gambar 3.42.



Gambar 3.42 *Activty Diagram Update* Data Bioskop

3.1.12.18 *Activity Diagram* Hapus Data Bioskop

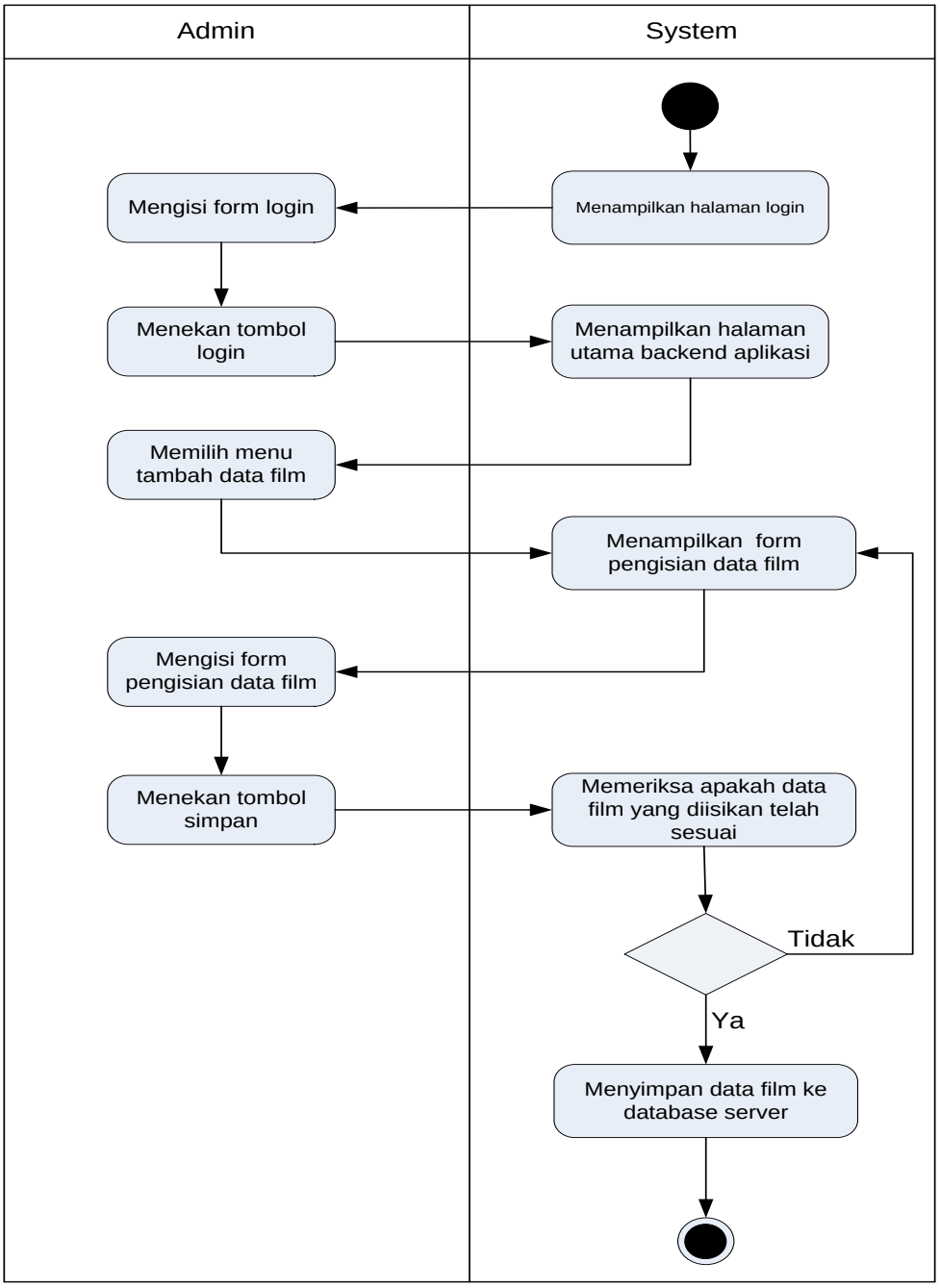
Activity diagram hapus data bioskop menjelaskan aliran kerja aktor admin pada saat melakukan penghapusan data bioskop dari *database server*. *Activity diagram* hapus data bioskop dapat dilihat pada gambar 3.43.



Gambar 3.43 *Activty Diagram* Hapus Data Bioskop

3.1.12.19 *Activity Diagram* Tambah Data Film

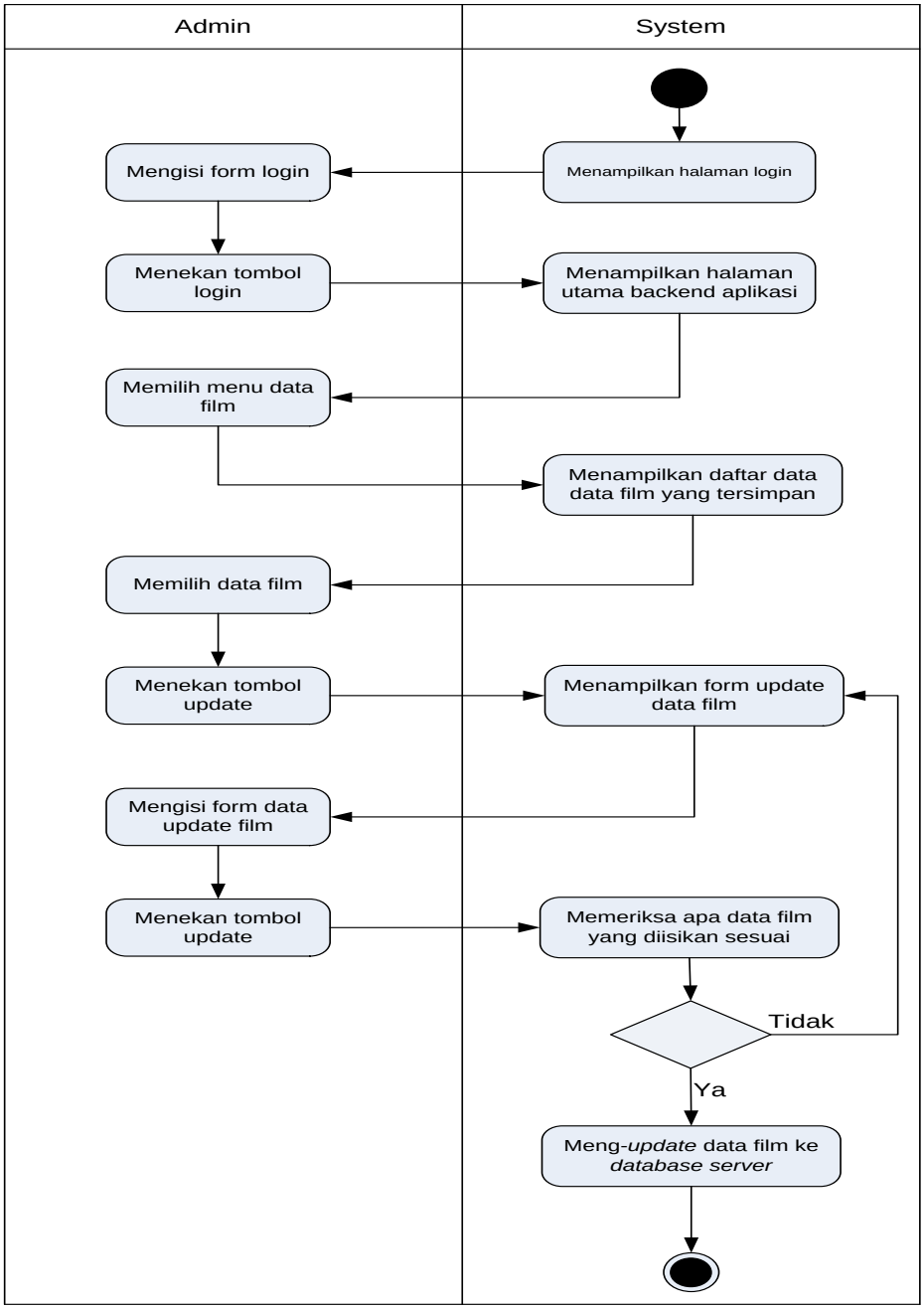
Activity diagram tambah data film menjelaskan aliran kerja aktor admin pada saat melakukan penambahan data film ke *database server*. *Activity diagram* tambah data film dapat dilihat pada gambar 3.44.



Gambar 3.44 *Activty Diagram* Tambah Data Film

3.1.12.20 *Activity Diagram Update* Data Film

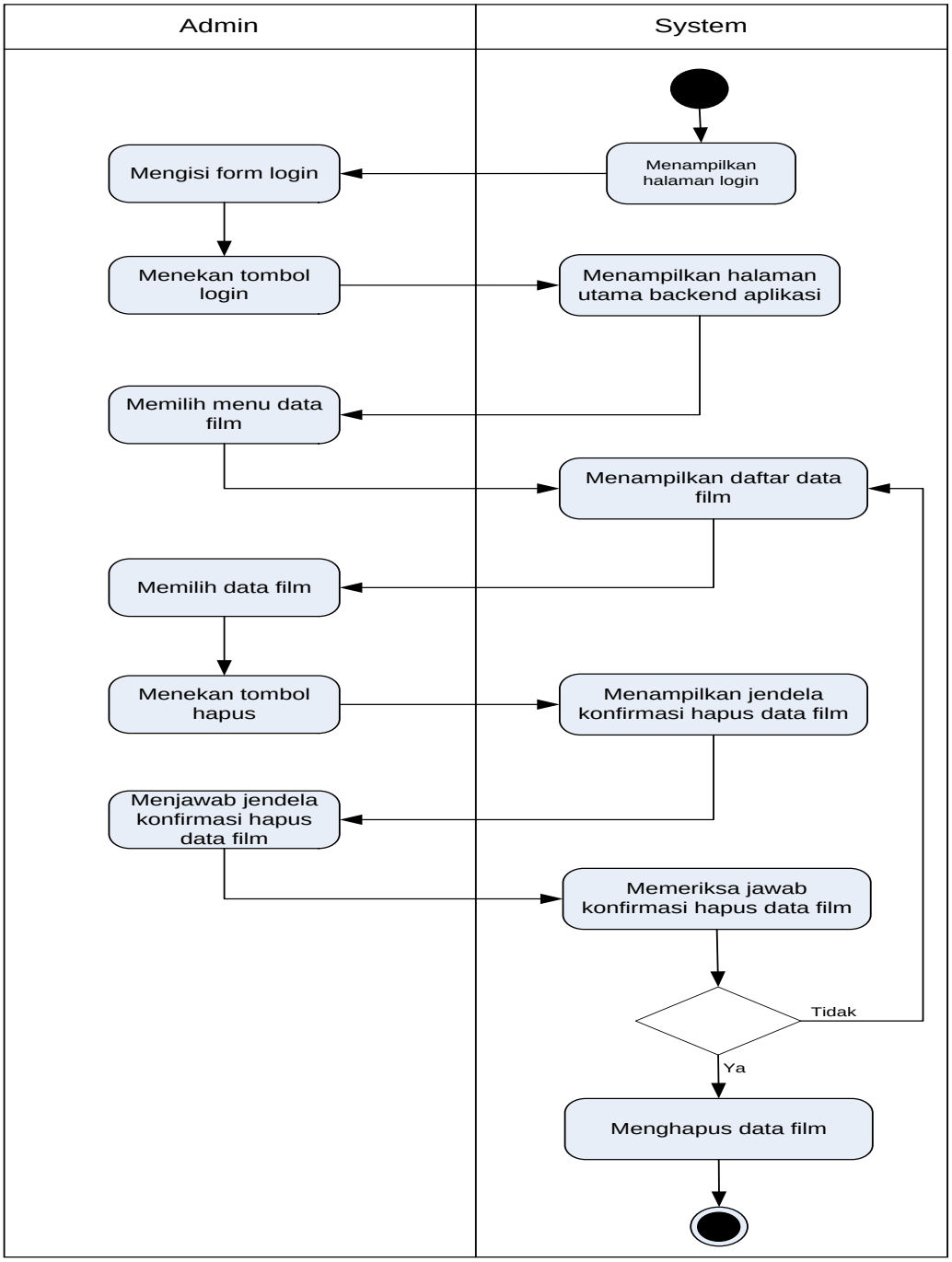
Activity diagram update data film menjelaskan aliran kerja aktor admin pada saat melakukan peng-*update*-an data film yang telah tersimpan di *databaseserver*. *Activity diagram update* data film dapat dilihat pada gambar 3.45.



Gambar 3.45 *Activty Diagram Update* Data Film

3.1.12.21 *Activity Diagram* Hapus Data Film

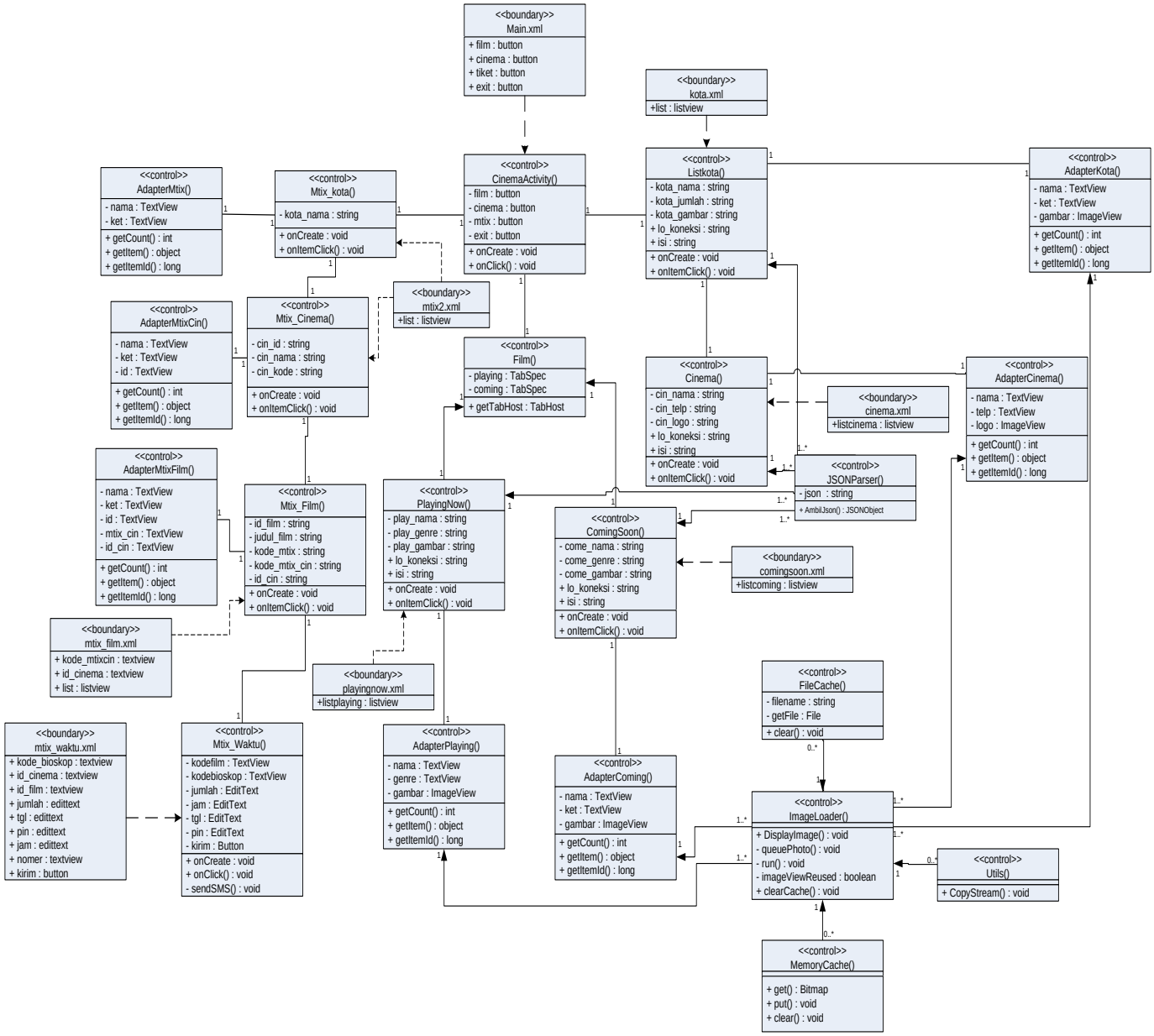
Activity diagram hapus data film menjelaskan aliran kerja aktor admin pada saat melakukan penghapusan data film dari *database server*. *Activity diagram* hapus data film dapat dilihat pada gambar 3.46.



Gambar 3.46 *Activty Diagram* Hapus Data Film

3.1.13. *Class Diagram*

Class Diagram menggambarkan struktur dan hubungan antar objek-objek yang ada pada sistem.Struktur itu meliputi atribut-atribut dan metode-metode yang ada pada masing-masing *class*. Hubungnnya dari masing-masing *class* yang ada tersebut digambarkan seperti pada gambar 3.47.



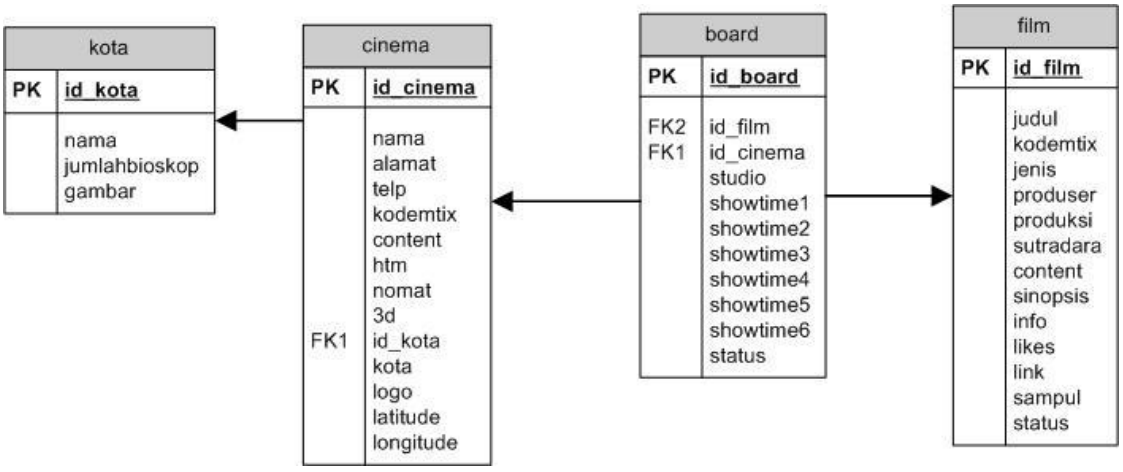
Gambar 3.47 *Class Diagram*

3.2. Perancangan Sistem

Perancangan akan dimulai setelah tahap analisis terhadap sistem selesai dilakukan. Perancangan dapat didefinisikan sebagai proses aplikasi berbagai teknik dan prinsip bagi tujuan pendefinisian suatu perangkat, suatu proses atau sistem dalam detail yang memadai untuk memungkinkan realisasi fisiknya. Perancangan digambarkan sebagai proses multi-langkah dimana representasi struktur data, struktur program, karakteristik *interface*, dan detail prosedur, disintesis dari persyaratan informasi.

3.2.1. Skema Relasi

Skema relasi menggambarkan suatu hubungan antar tabel yang sudah ada dalam keadaan normal. Perancangan tabel relasi dalam membangun sebuah perangkat lunak aplikasi ini dapat dilihat pada gambar3.48.



Gambar 3.48 Skema Relasi

3.2.2. Perancangan Data

Perancangan data merupakan tahapan untuk memetakan model konseptual ke model basis data yang akan digunakan. Perancangan data dari aplikasi *mobile* Informasi Bioskop merupakan perancangan struktur tabel dari tabel kategori. Struktur tabel kategori menggambarkan detail tabel yang berisi *field*, tipe data,

panjang data, dan keterangan lainnya. Berikut ini gambaran struktur tabel pengaturan yang terlihat pada tabel 3.26 sampai dengan tabel 3.29.

Tabel 3.26 Perancangan Data Kota

Nama Field	Tipe Data	Ukuran	Keterangan
id_kota	integer	2	Primary key
nama	varchar	30	
jumlahbioskop	integer	2	
gambar	varchar	50	

Tabel 3.27 Perancangan Data Cinema

Nama Field	Tipe Data	Ukuran	Keterangan
id_cinema	varchar	30	Primary key
nama	varchar	100	
alamat	varchar	200	
telp	varchar	20	
kodemtix	varchar	20	
content	text		
htm	varchar	100	
nomat	varchar	100	
3d	varchar	100	
id_kota	varchar	20	Foreign Key
kota	varchar	200	
logo	varchar	20	
longitude	double		
latitude	double		

Tabel 3.28 Perancangan Data Film

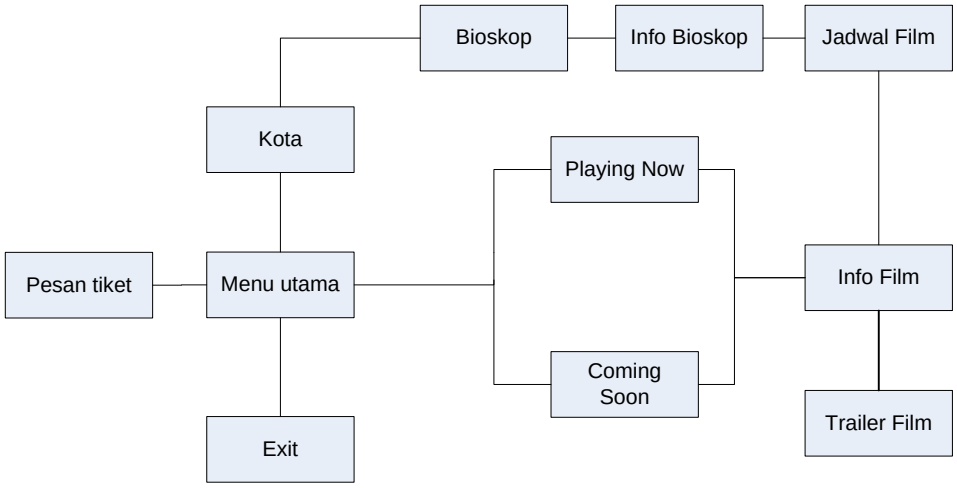
Nama Field	Tipe Data	Ukuran	Keterangan
id_film	integer	8	Primary key
judul	varchar	75	
kodemtix	varchar	20	
jenis	varchar	20	
produser	varchar	200	
produksi	varchar	200	
sutradara	varchar	200	
content	text		
sinopsis	text		
info	varchar	20	
likes	integer	3	
link	varchar	100	
sampul	varchar	20	
status	varchar	20	

Tabel 3.29 Perancangan Data Board

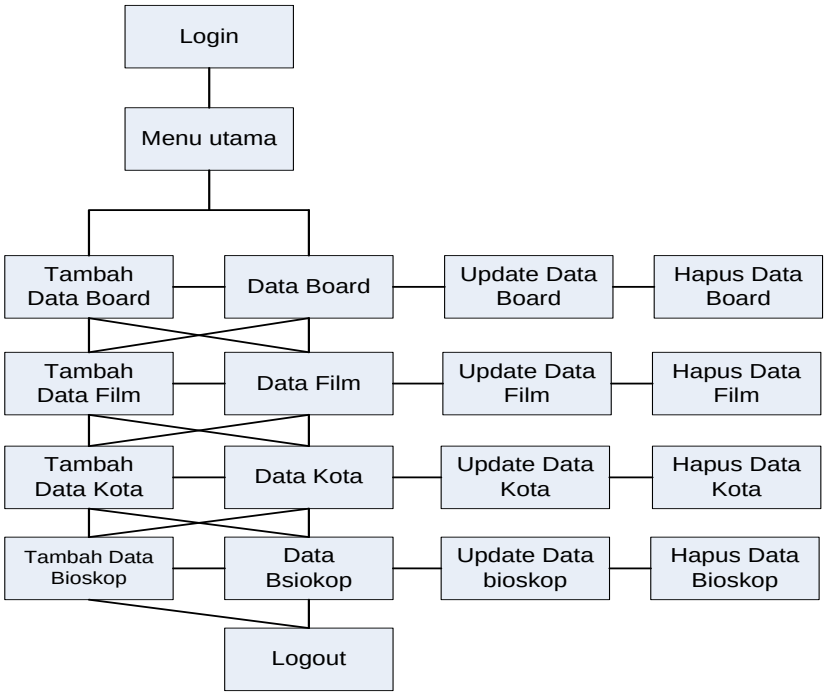
Nama Field	Tipe Data	Ukuran	Keterangan
Id_board	integer	11	Primary key
id_film	integer	20	Foreign Key 1
id_cinema	varchar	20	Foreign Key 2
studio	integer	11	
showtime1	varchar	20	
showtime2	varchar	20	
showtime3	varchar	20	
showtime4	varchar	20	
showtime5	varchar	20	
showtime6	varchar	20	
status	varchar	20	

3.2.3. Struktur Menu

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi. Perancangan struktur menu dari aplikasi *mobile* ini dapat dilihat pada gambar 3.49 dan struktur menu untuk *backend* aplikasi *mobile* ini dapat dilihat pada gambar 3.50.



Gambar 3.49 Struktur Menu *Frontend* Aplikasi



Gambar 3.50Struktur Menu *Backend* Aplikasi

3.2.4. Perancangan Antarmuka

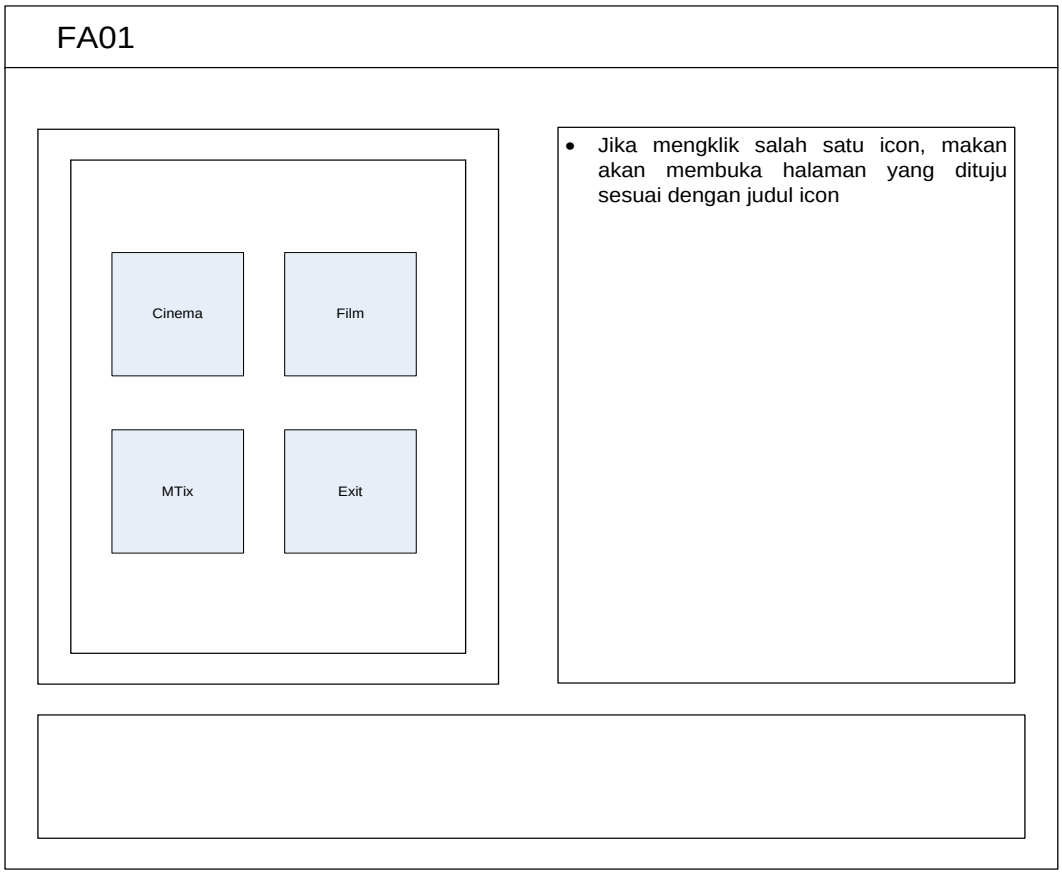
Perancangan antarmuka (*interface*) menggambarkan bagaimana perangkat lunak berkomunikasi dengan dirinya sendiri, dengan sistem yang berinteroperasi dengannnya, dan dengan manusia yang menggunakannya.

3.2.4.1 Perancangan Tampilan

Perancangan *form* mendeskripsikan rencana tampilan dari setiap *form* yang akan digunakan pada tampilan aplikasi *mobile* Informasi Bioskop. Adapun gambaran dari perancangan antarmuka aplikasi *mobile* Informasi Bioskop adalah sebagai berikut:

1. Perancangan Menu Utama

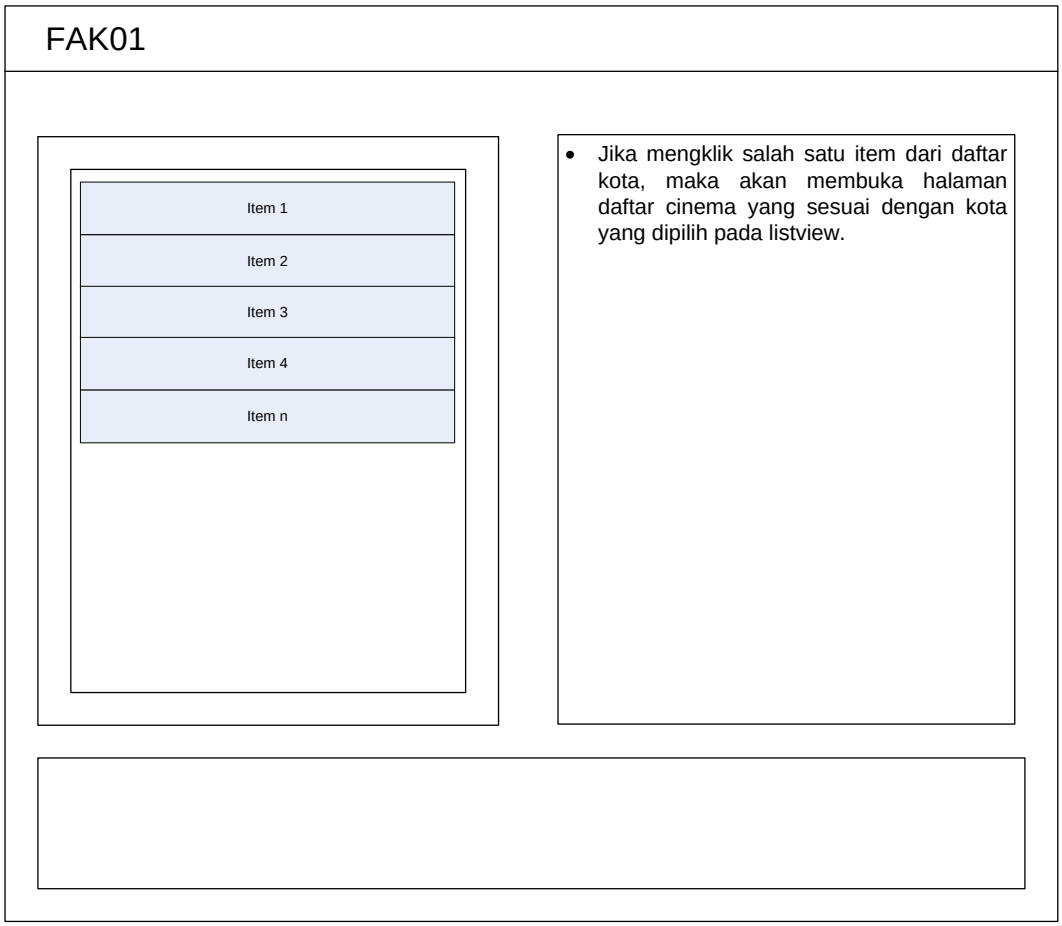
Gambaran umum dari perancangan antarmuka menu utama pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.51.



Gambar 3.51 Tampilan Menu Utama

2. Perancangan Menu Kota

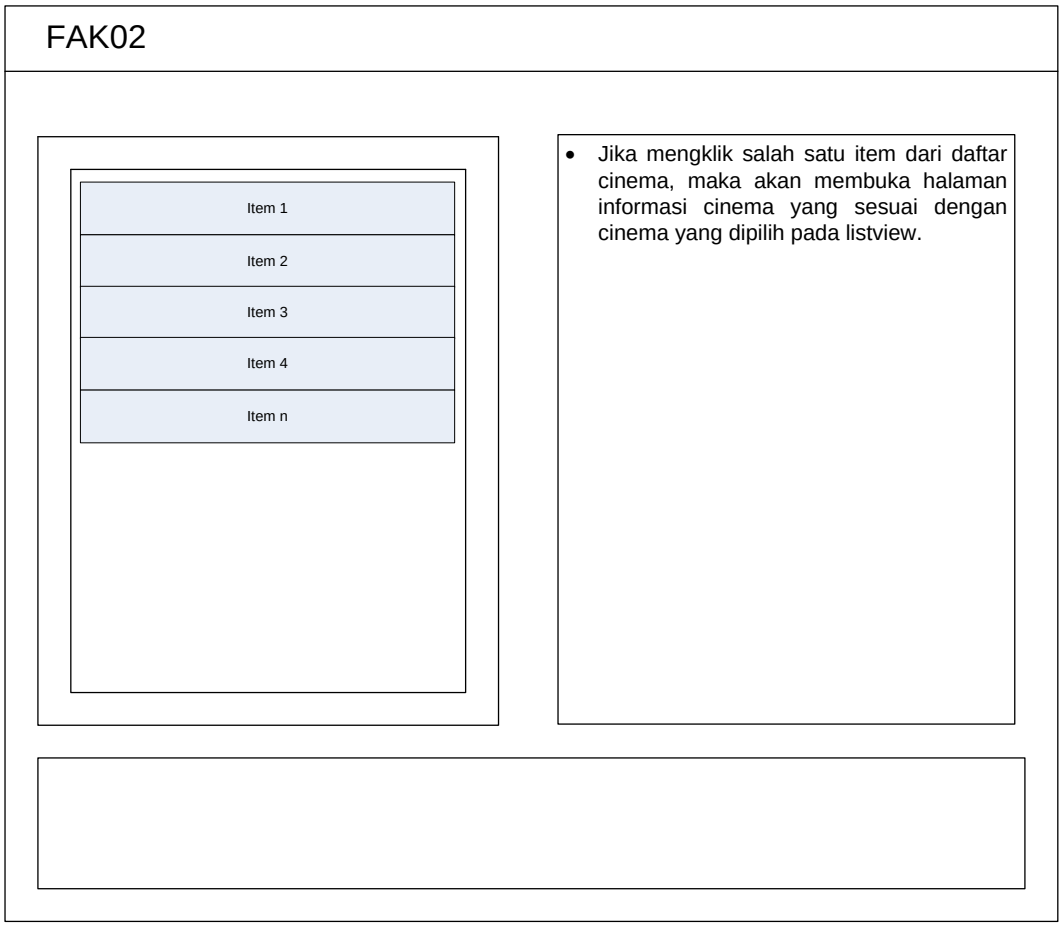
Gambaran umum dari perancangan antarmuka menu kota pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.52.



Gambar 3.52 Tampilan Menu Kota

3. Perancangan Menu Bioskop

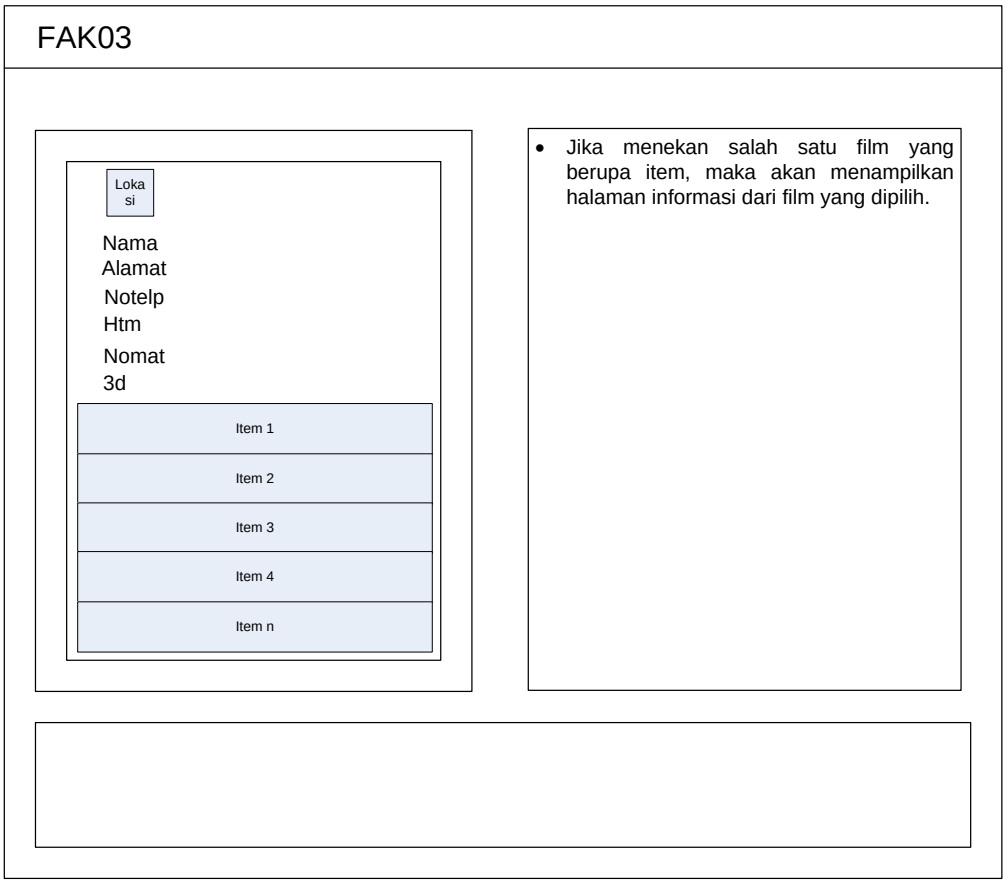
Gambaran umum dari perancangan antarmuka menu bioskop pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.53.



Gambar 3.53 Tampilan Menu Bioskop

4. Perancangan Menu Informasi Bioskop

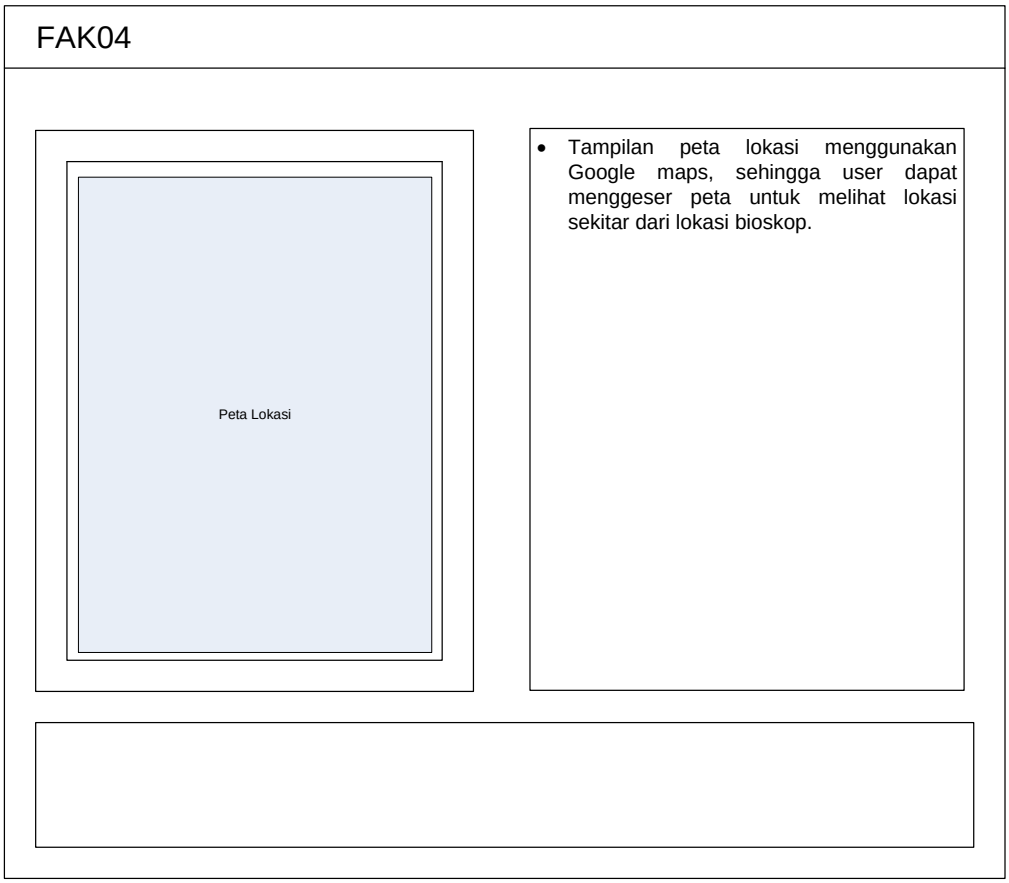
Gambaran umum dari perancangan antarmuka menu informasi cinema pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.54.



Gambar 3.54 Tampilan Menu Informasi Bioskop

5. Perancangan Menu Lokasi Bioskop

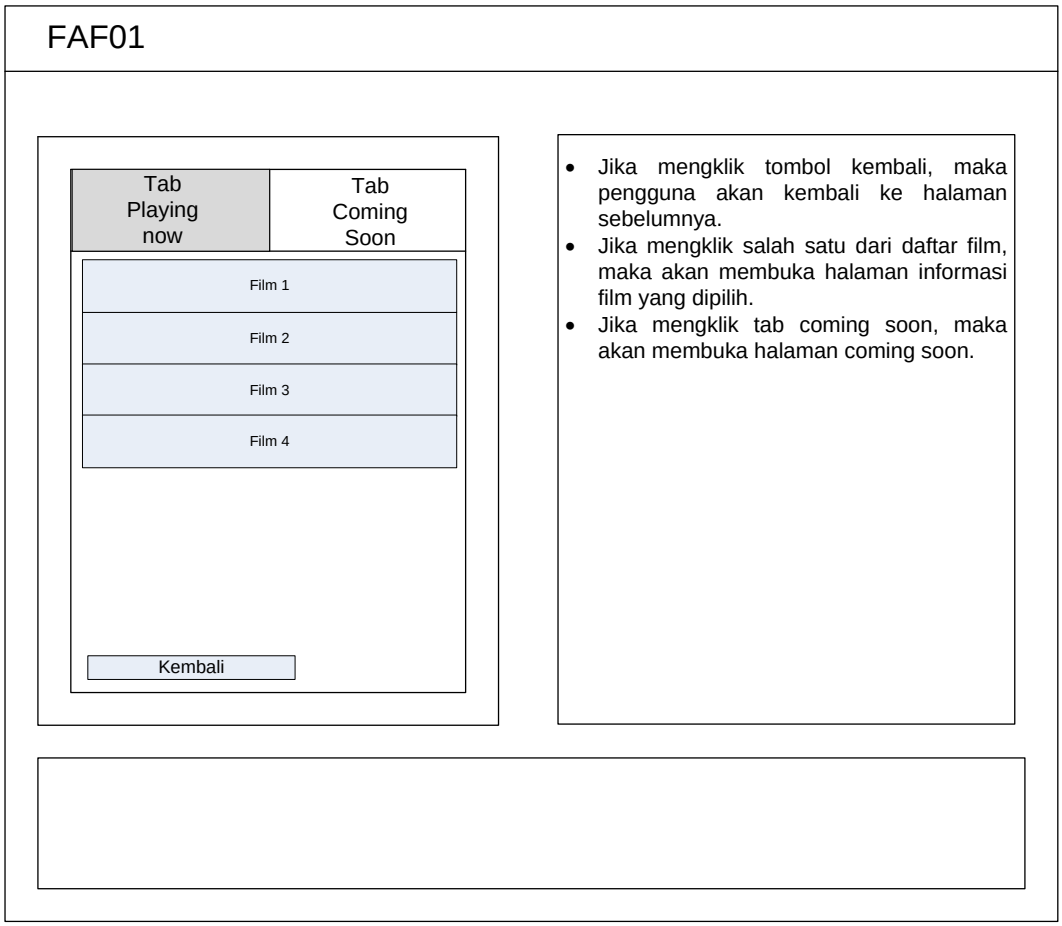
Gambaran umum dari perancangan antarmuka menu lokasi bioskop pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.55.



Gambar 3.55 Tampilan Menu Lokasi Bioskop

6. Perancangan Menu *Playing Now*

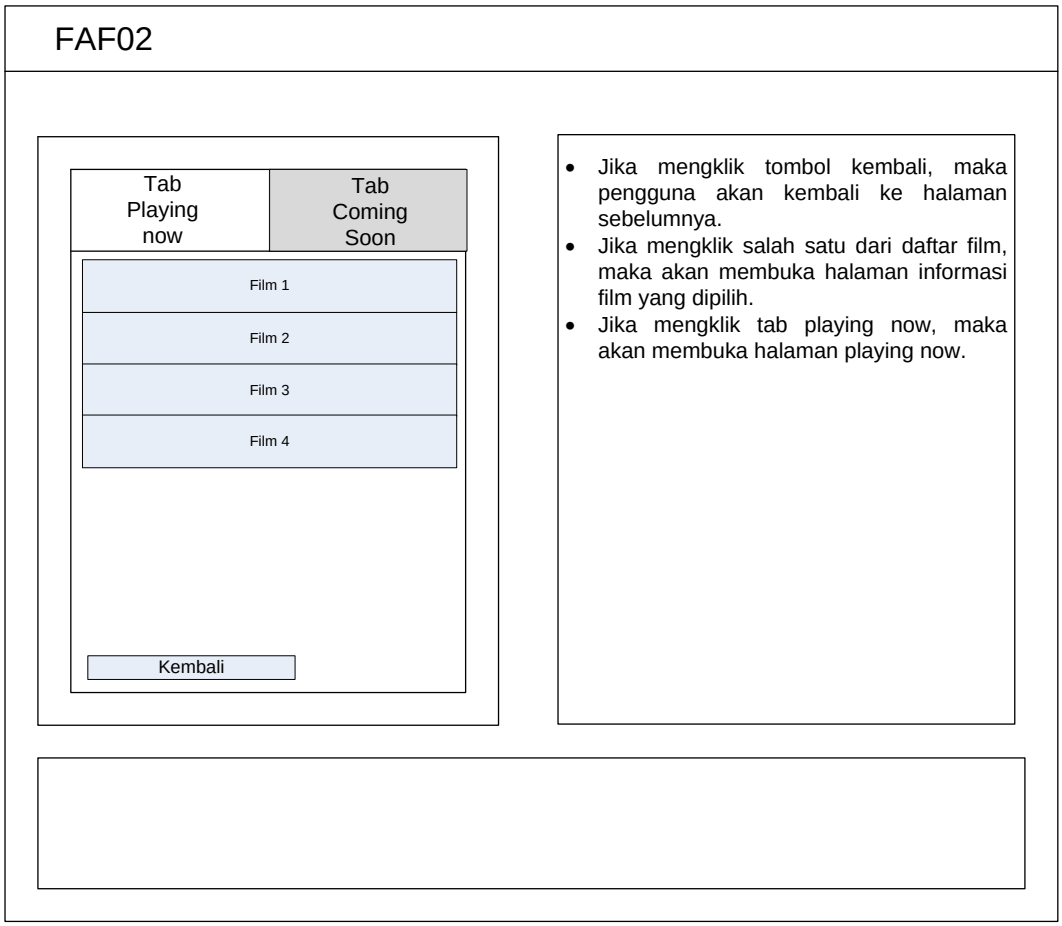
Gambaran umum dari perancangan antarmuka menu *playing now* pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.56.



Gambar 3.56 Tampilan Menu *Playing Now*

7. Perancangan Menu *Coming Soon*

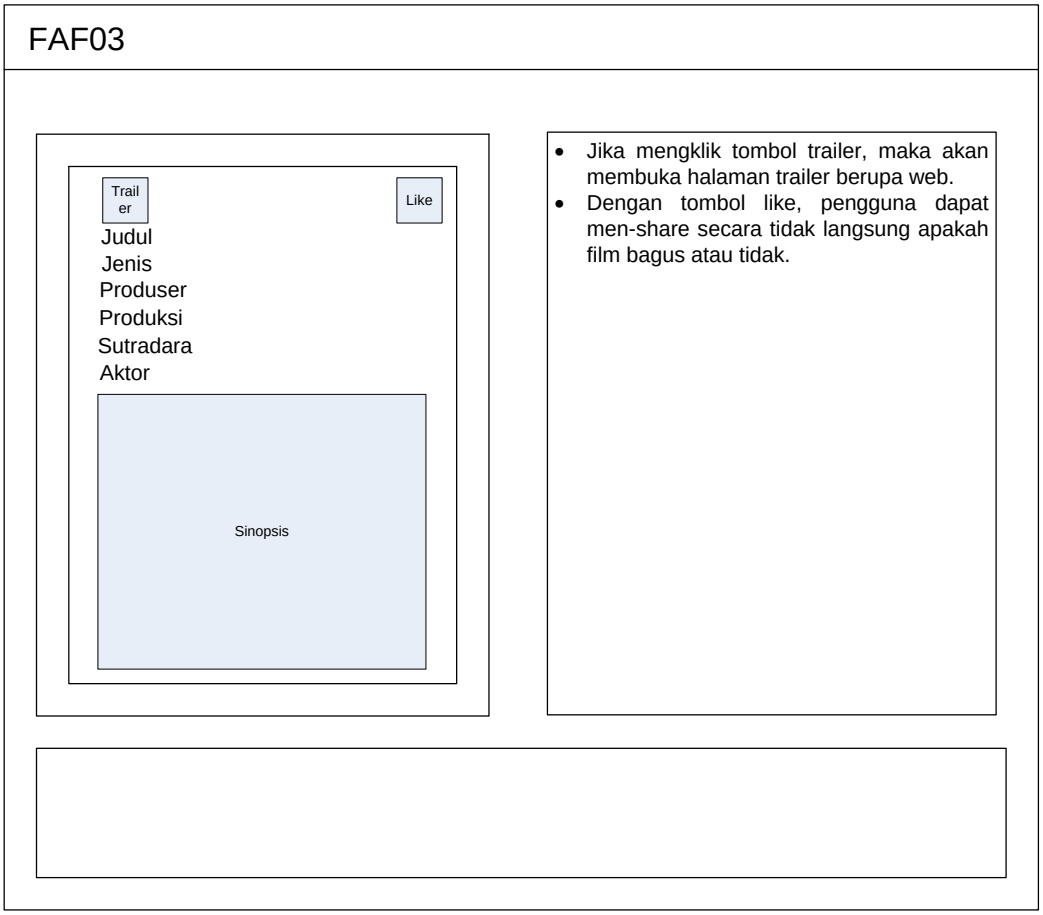
Gambaran umum dari perancangan antarmuka menu *coming soon* pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.57.



Gambar 3.57 Tampilan Menu *Coming Soon*

8. Perancangan Menu Informasi Film

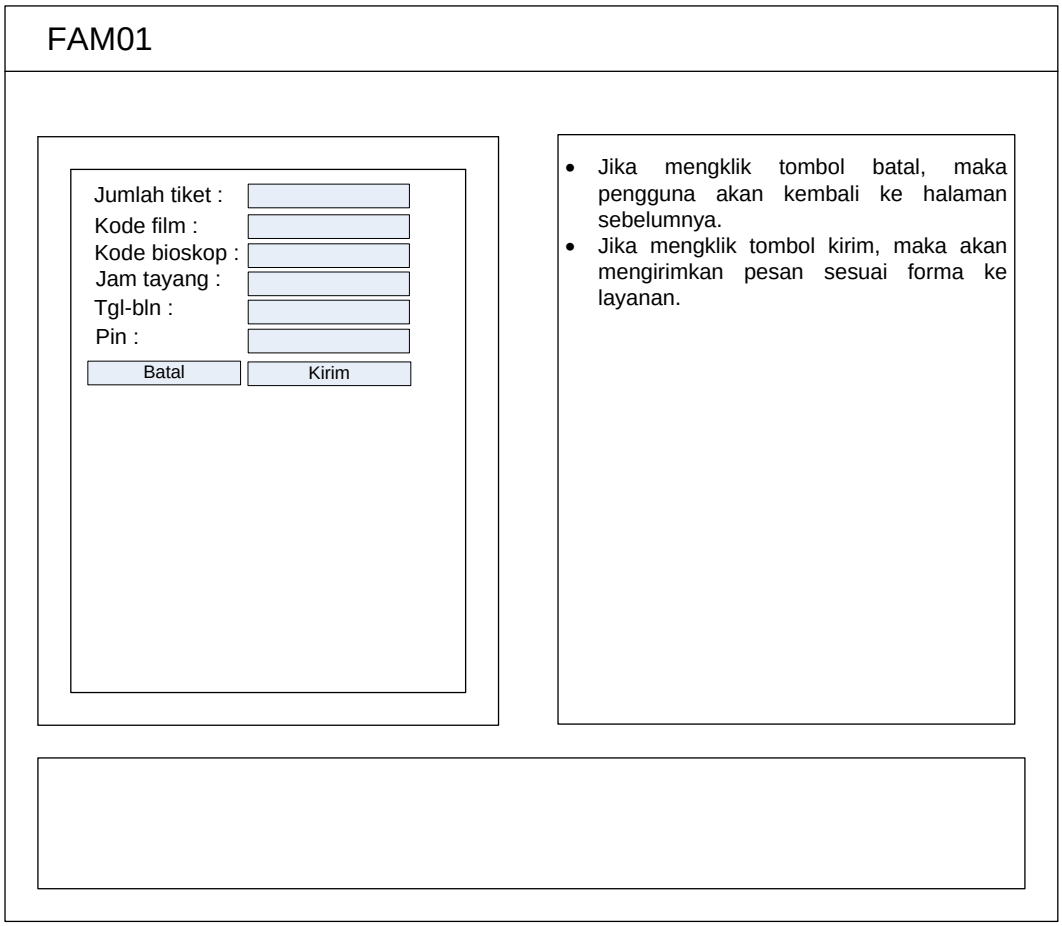
Gambaran umum dari perancangan antarmuka menu informasi film pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.58.



Gambar 3.58 Tampilan Menu Informasi Film

9. Perancangan Menu Mtix

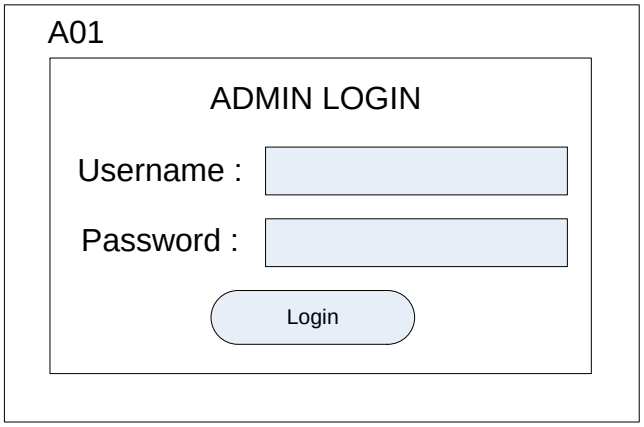
Gambaran umum dari perancangan antarmuka menu mtix film pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.59.



Gambar 3.59 Tampilan Menu Mtix

10. Perancangan Halaman Login Admin

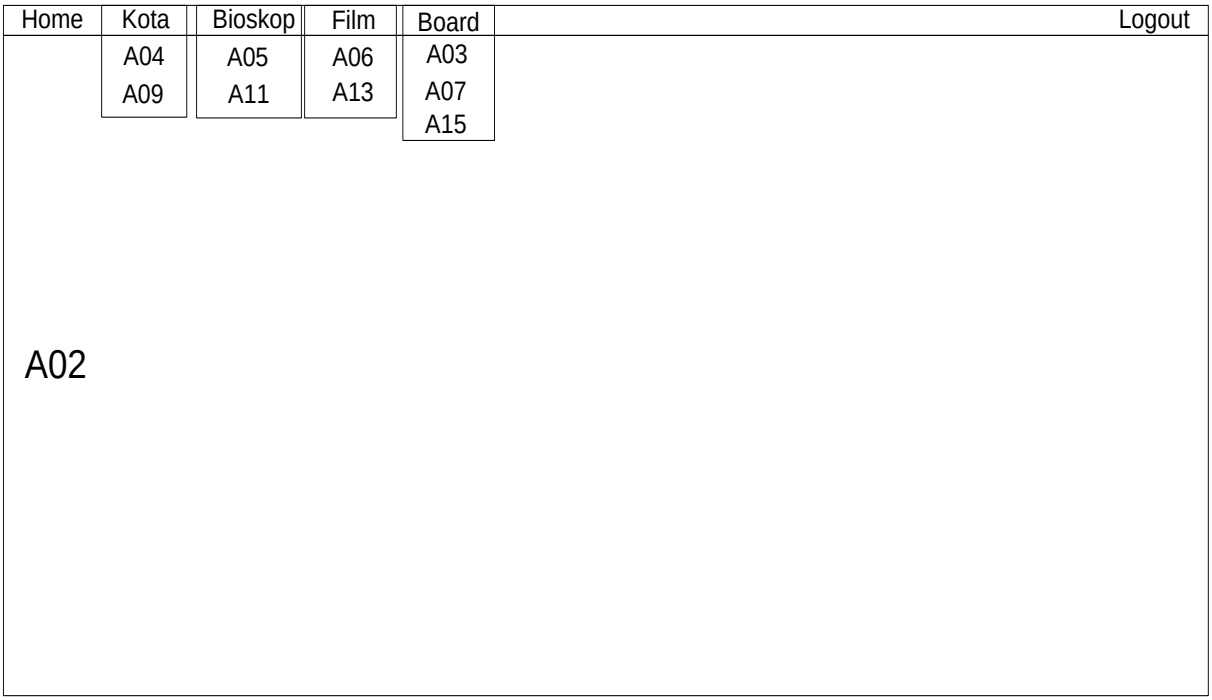
Gambaran umum dari perancangan antarmuka halaman login admin pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.60.



Gambar 3.60 Tampilan Halaman Login Admin

11. Perancangan Halaman Utama Admin

Gambaran umum dari perancangan antarmuka halaman utama admin pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.61.



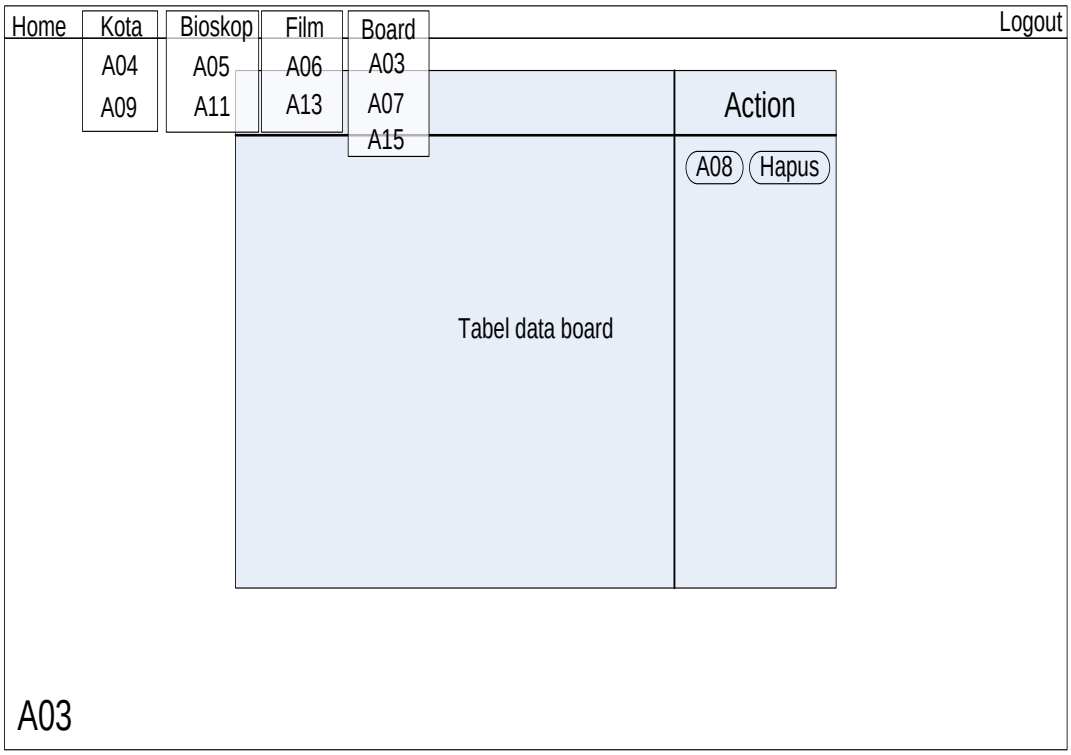
Gambar 3.61 Tampilan Halaman Utama Admin

Tabel 3.30 Keterangan Perancangan Halaman Utama Admin

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman Playing At

12. Perancangan Halaman Data Board

Gambaran umum dari perancangan antarmuka halaman data board pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.62.



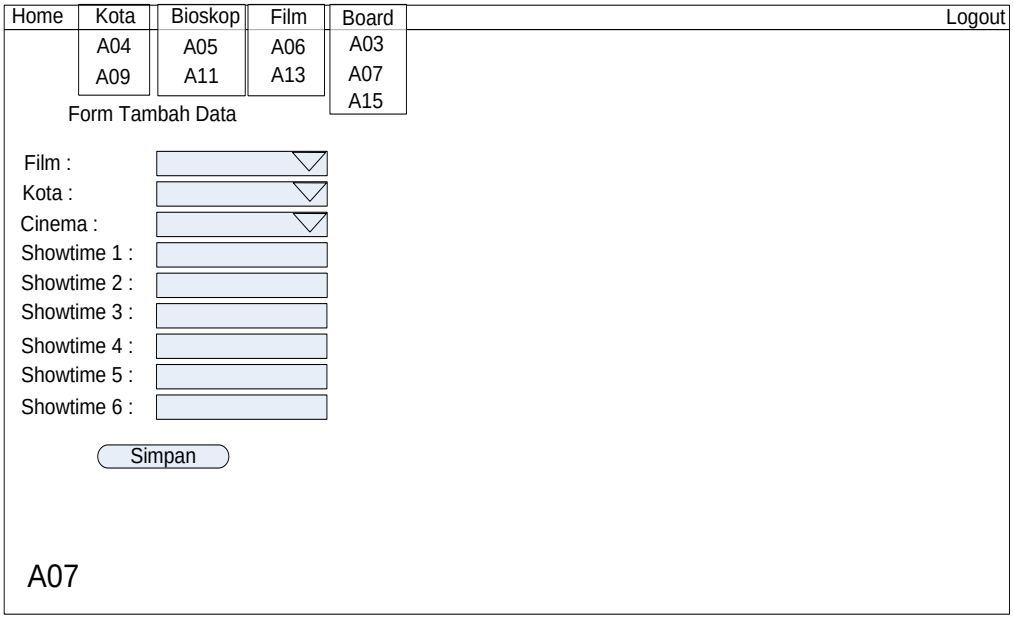
Gambar 3.62 Tampilan Halaman Data Board

Tabel 3.31 Keterangan Perancangan Halaman Data Board

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A08	Halaman *update* board
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

13. Perancangan Halaman Tambah Data Board

Gambaran umum dari perancangan antarmuka halaman tambah data board pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.63.



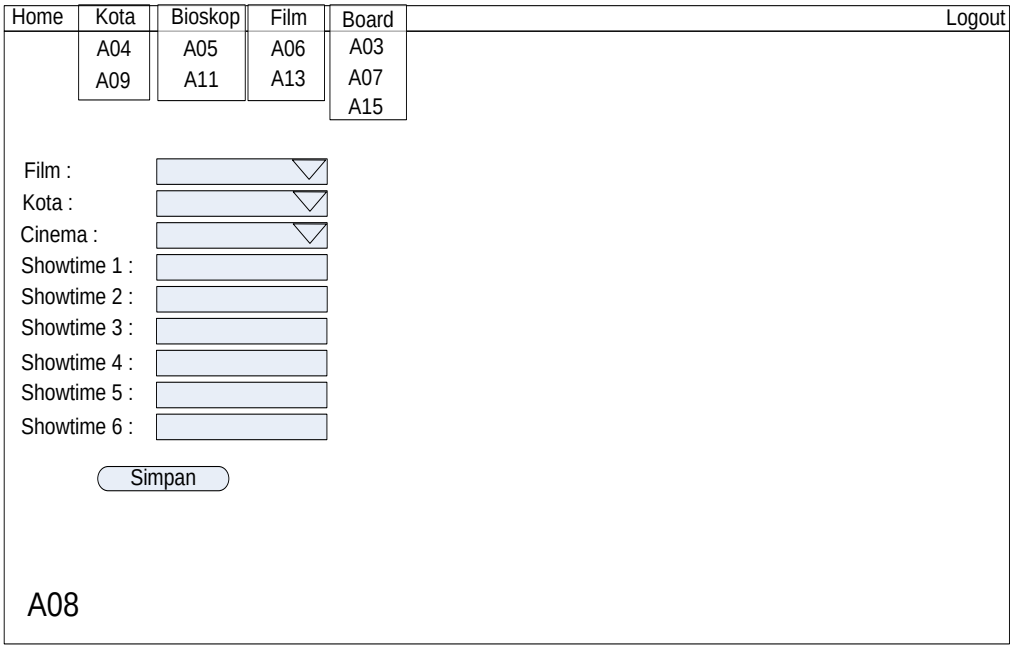
Gambar 3.63 Tampilan Halaman Tambah Data Board

Tabel 3.32 Keterangan Perancangan Halaman Tambah Data Board

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film

14. Perancangan Halaman *Update* Data Board

Gambaran umum dari perancangan antarmuka halaman *update* data board pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.64.



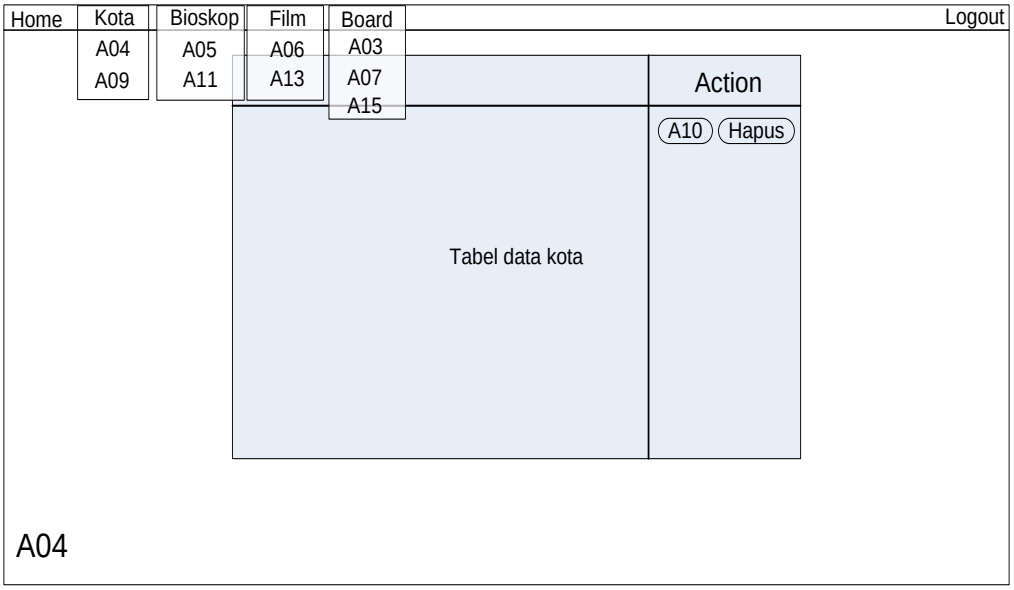
Gambar 3.64 Tampilan Halaman *Update* Data Board

Tabel 3.33 Keterangan Perancangan Halaman *Update* Data Board

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

15. Perancangan Halaman Data Kota

Gambaran umum dari perancangan antarmuka halaman data kota pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.65.



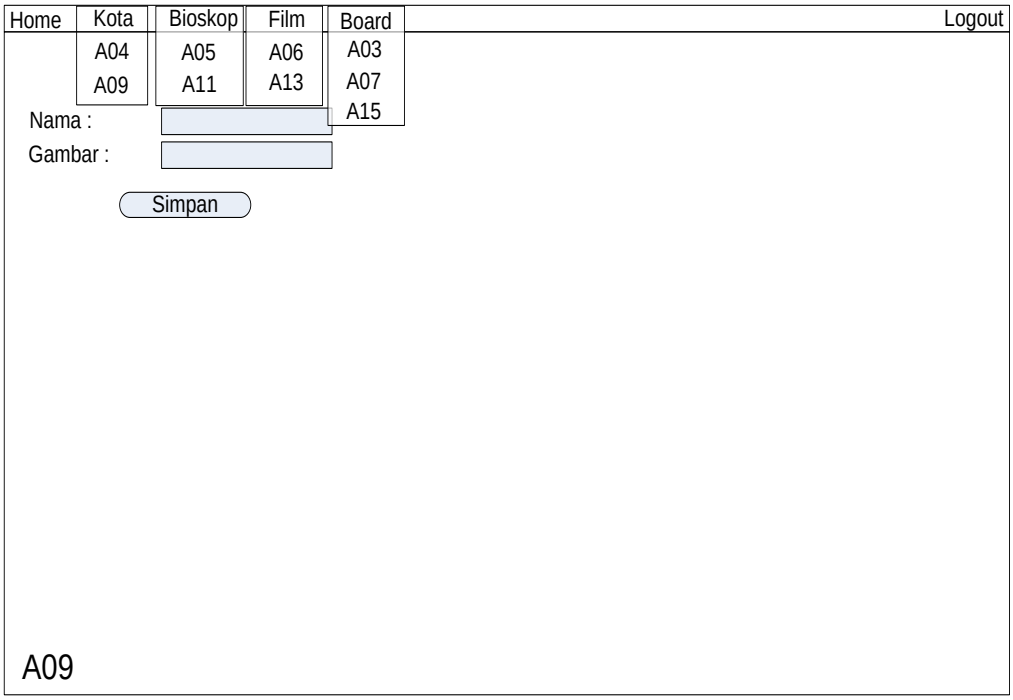
Gambar 3.65 Tampilan Halaman Data Kota

Tabel 3.34 Keterangan Perancangan Halaman Data Kota

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A10	Halaman *update* data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

16. Perancangan Halaman Tambah Data Kota

Gambaran umum dari perancangan antarmuka halaman tambah data kota pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.66.



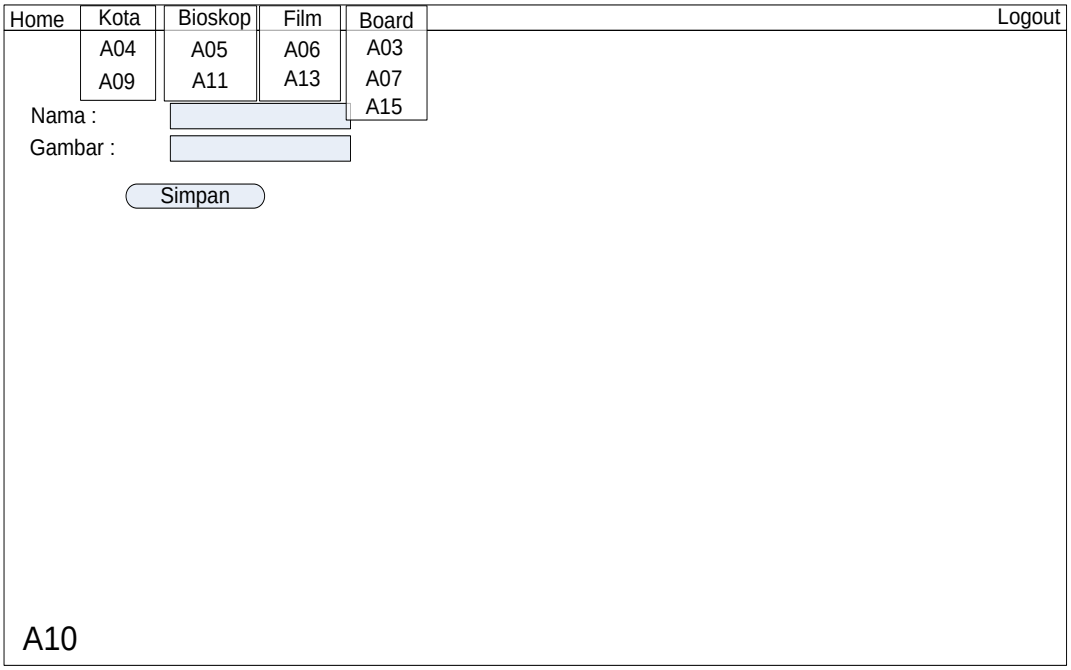
Gambar 3.66 Tampilan Halaman Tambah Data Kota

Tabel 3.35 Keterangan Perancangan Halaman Tambah Data Kota

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

17. Perancangan Halaman *Update* Data Kota

Gambaran umum dari perancangan antarmuka halaman *update* data kota pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.67.



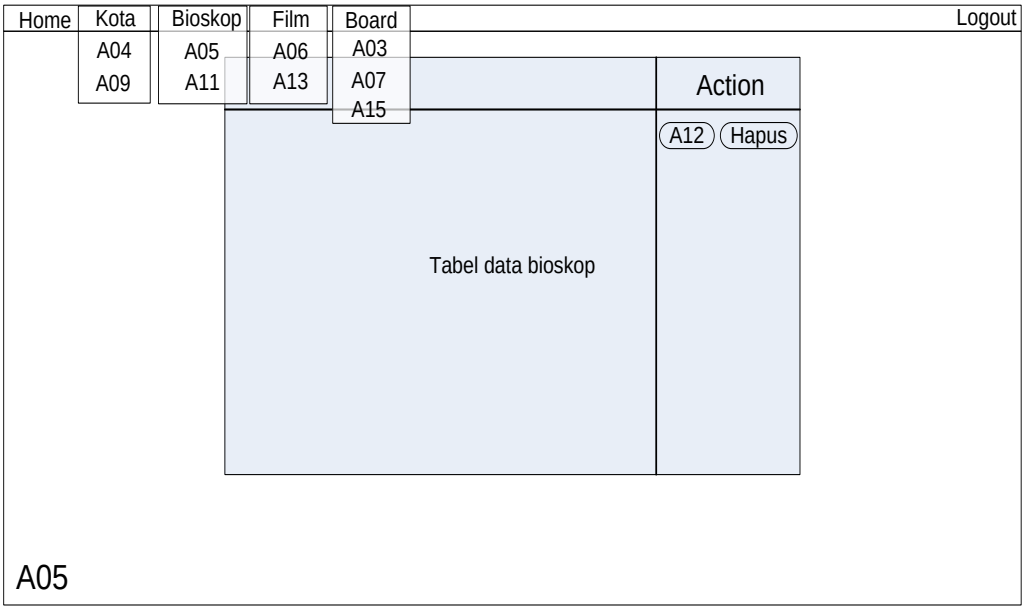
Gambar 3.67 Tampilan Halaman *Update* Data Kota

Tabel 3.36 Keterangan Perancangan Halaman *Update* Data Kota

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

18. Perancangan Halaman Data Bioskop

Gambaran umum dari perancangan antarmuka halaman data bioskop pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.68.



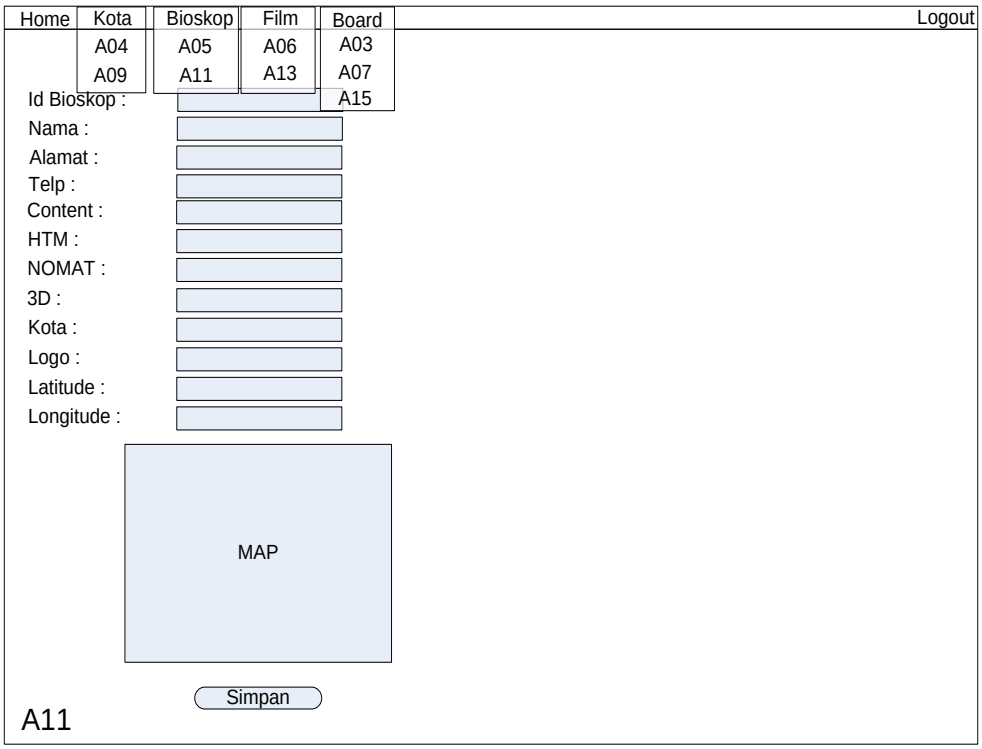
Gambar 3.68 Tampilan Halaman Data Bioskop

Tabel 3.37 Keterangan Perancangan Halaman Data Bioskop

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A12	Halaman *update* data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

19. Perancangan Halaman Tambah Data Bioskop

Gambaran umum dari perancangan antarmuka halaman tambah data bioskop pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.69.



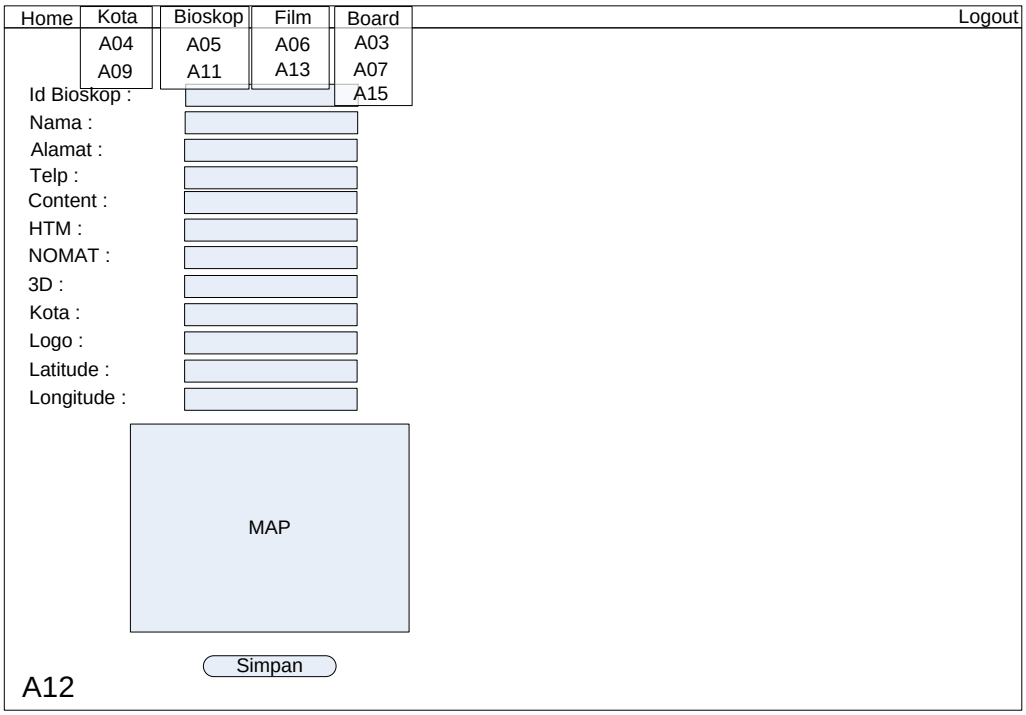
Gambar 3.69 Tampilan Halaman Tambah Data Bioskop

Tabel 3.38 Keterangan Perancangan Halaman Tambah Data Bioskop

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

20. Perancangan Halaman *Update* Data Bioskop

Gambaran umum dari perancangan antarmuka halaman *update* data bioskop pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.70.



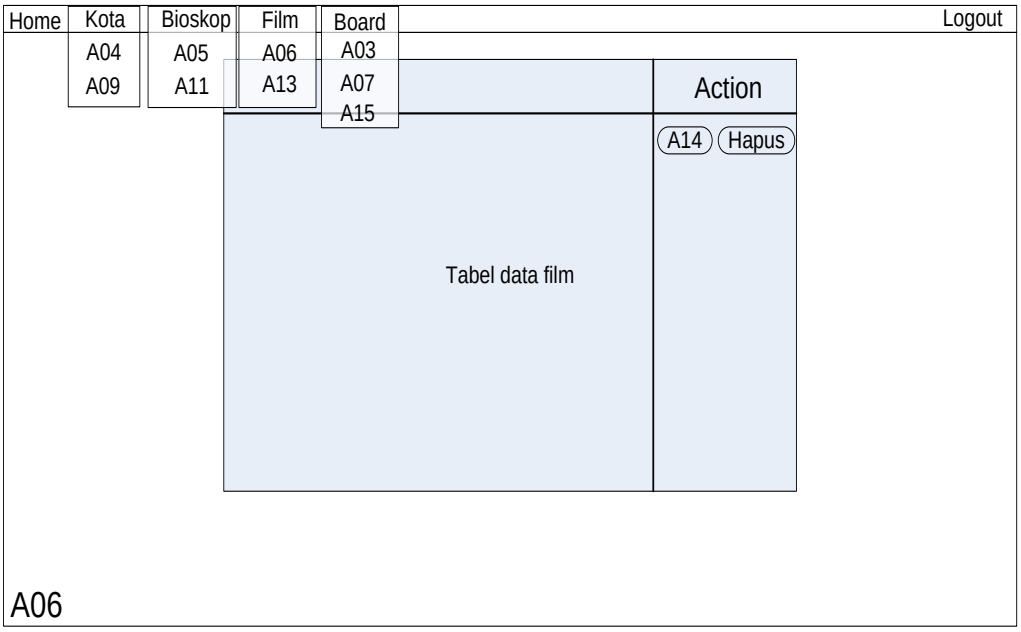
Gambar 3.70 Tampilan Halaman *Update* Data Bioskop

Tabel 3.39 Keterangan Perancangan Halaman *Update* Data Bioskop

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

21. Perancangan Halaman Data Film

Gambaran umum dari perancangan antarmuka halaman data film pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.71.



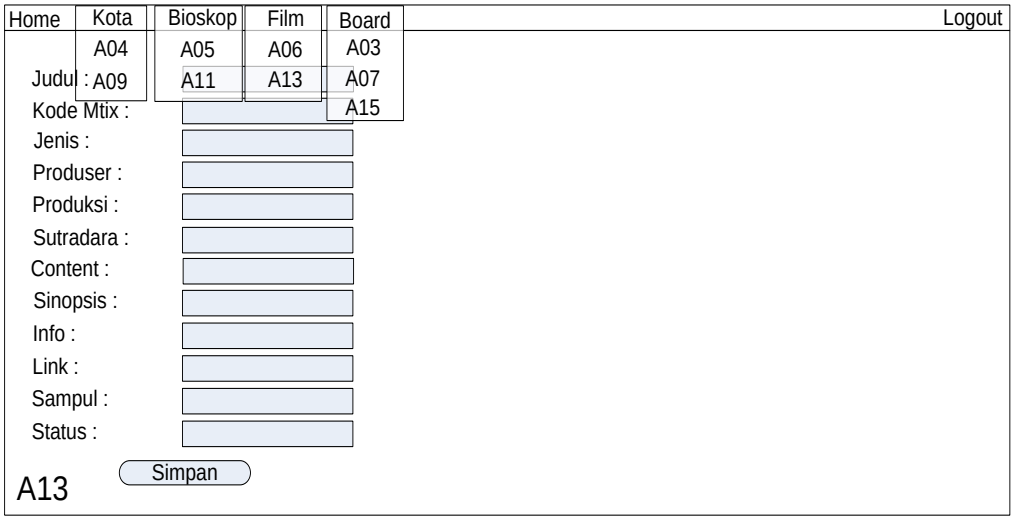
Gambar 3.71 Tampilan Halaman Data Film

Tabel 3.40 Keterangan Perancangan Halaman Data Film

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A14	Halaman *update* data film
A15	Halaman playing at

22. Perancangan Halaman Tambah Data Film

Gambaran umum dari perancangan antarmuka halaman tambah data film pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.72.



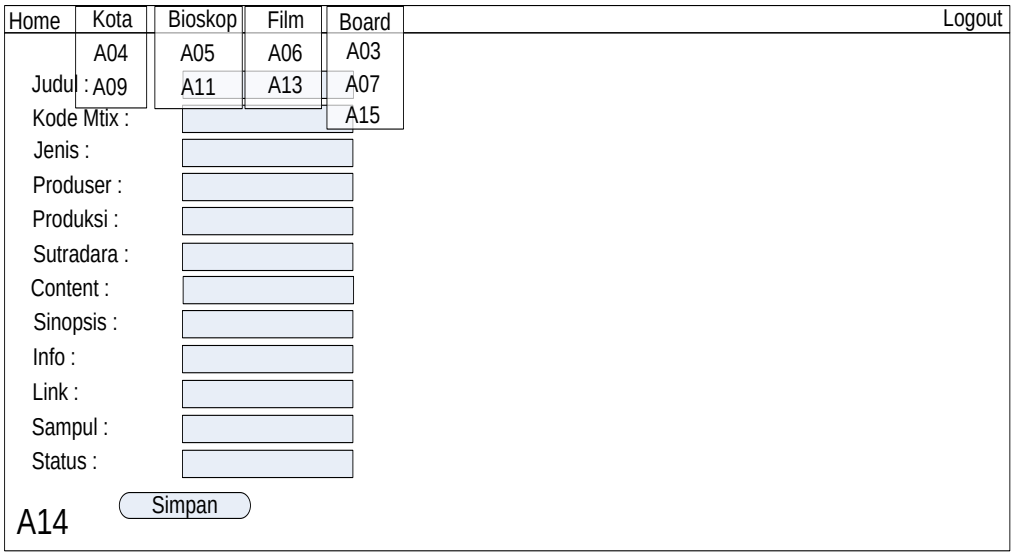
Gambar 3.72 Tampilan Halaman Tambah Data Film

Tabel 3.41 Keterangan Perancangan Halaman Tambah Data Film

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

23. Perancangan Halaman *Update* Data Film

Gambaran umum dari perancangan antarmuka halaman *update* data film pada aplikasi *mobile* Informasi Bioskop dapat dilihat pada gambar 3.73.



Gambar 3.73 Tampilan Halaman *Update* Data Film

Tabel 3.42 Keterangan Perancangan Halaman *Update* Data Film

Kode	Keterangan
A03	Halaman data board
A04	Halaman data kota
A05	Halaman data bioskop
A06	Halaman data film
A07	Halaman tambah data board
A09	Halaman tambah data kota
A11	Halaman tambah data bioskop
A13	Halaman tambah data film
A15	Halaman playing at

3.2.4.2 Perancangan Pesan

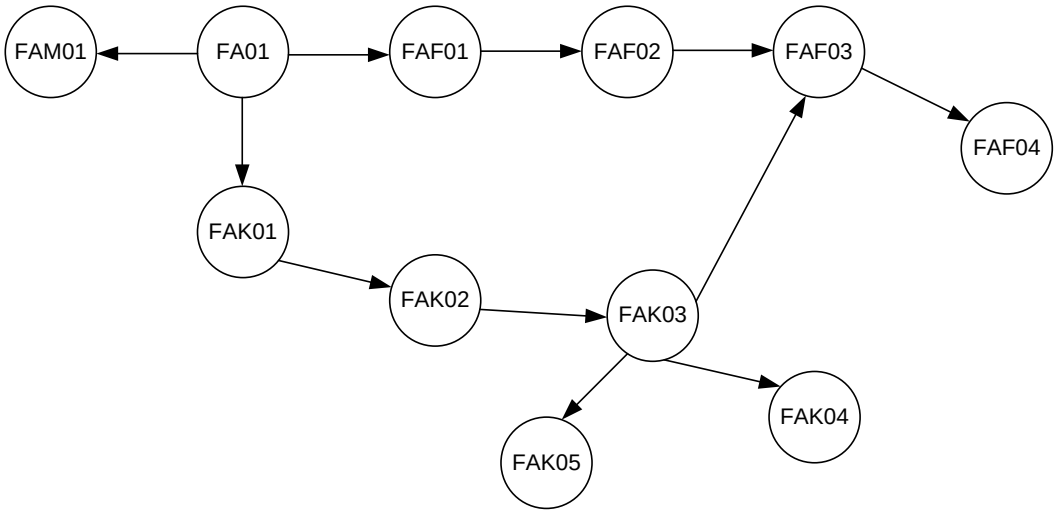
Perancangan pesan mendeskripsikan rencana tampilan dari setiap pesan yang digunakan pada aplikasi Sistem Informasi Bioskop ini. Berikut ini gambaran dari perancangan pesan pada aplikasi ini dapat dilihat pada gambar 3.74.



Gambar 3.74 Perancangan Pesan

3.2.5. Jaringan Semantik

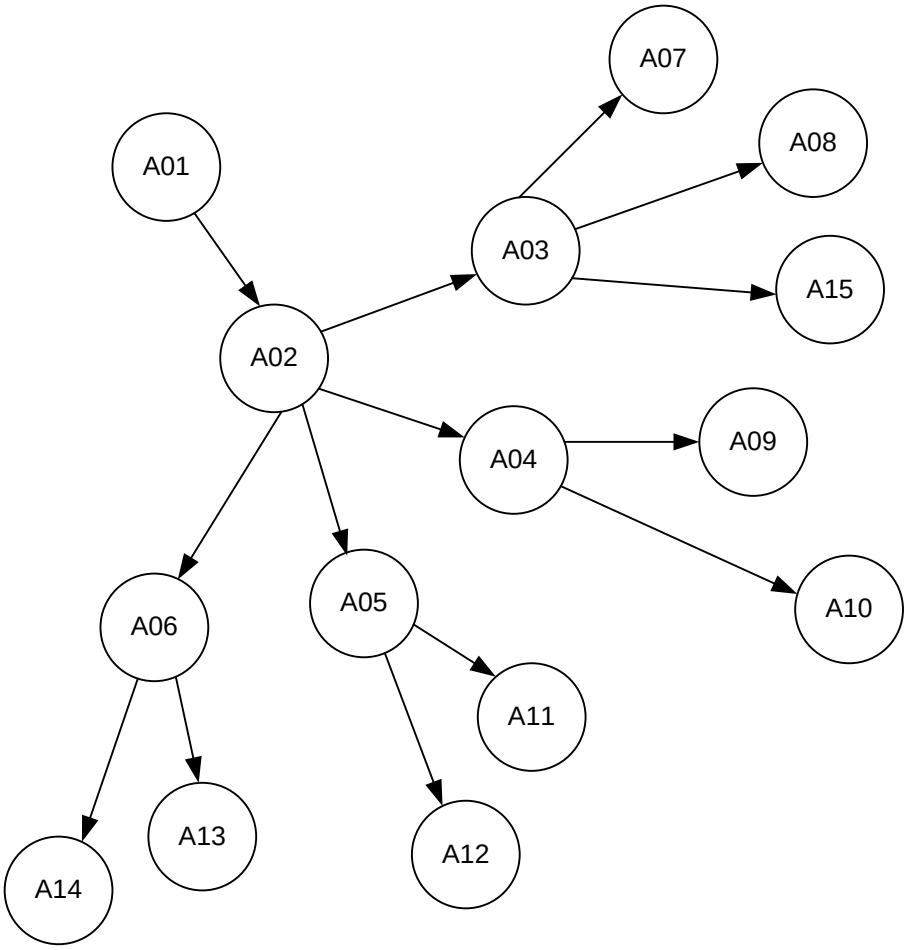
Jaringan semantik merupakan gambaran pengetahuan grafis yang mana menunjukkan hubungan antar objek (lingkaran) dan grafis yang menggambarkan formasi antar objek tersebut.Adapun jaringan semantik untuk *mobile* pada aplikasi ini dapat dilihat pada gambar 3.75 dan untuk *backend* pada aplikasi ini dapat dilihat pada gambar 3.76.



Gambar 3.75 Jaringan Semantik *Frontend*

Keterangan :

FA01	Halaman utama aplikasi
FAM01	Form pesan tiket
FAK01	Halaman daftar kota
FAK02	Halaman daftar bioskop
FAK03	Halaman informasi bioskop
FAK04	Halaman jadwal film
FAK05	Halaman playing at
FAK05	Halaman lokasi bioskop
FAF01	Halaman film *playing now*
FAF02	Halaman film *coming soon*
FAF03	Halaman informasi film
FAF04	Halaman *trailer* film



Gambar 3.76 Jaringan Semantik untuk *Backend*

Keterangan :

A01 Halaman Login

A02 Halaman utama *backend*

A03 Halaman data board

A04 Halaman data kota

A05 Halaman data bioskop

A06 Halaman data film

A07 Halaman tambah data board

A08 Halaman *update* data board

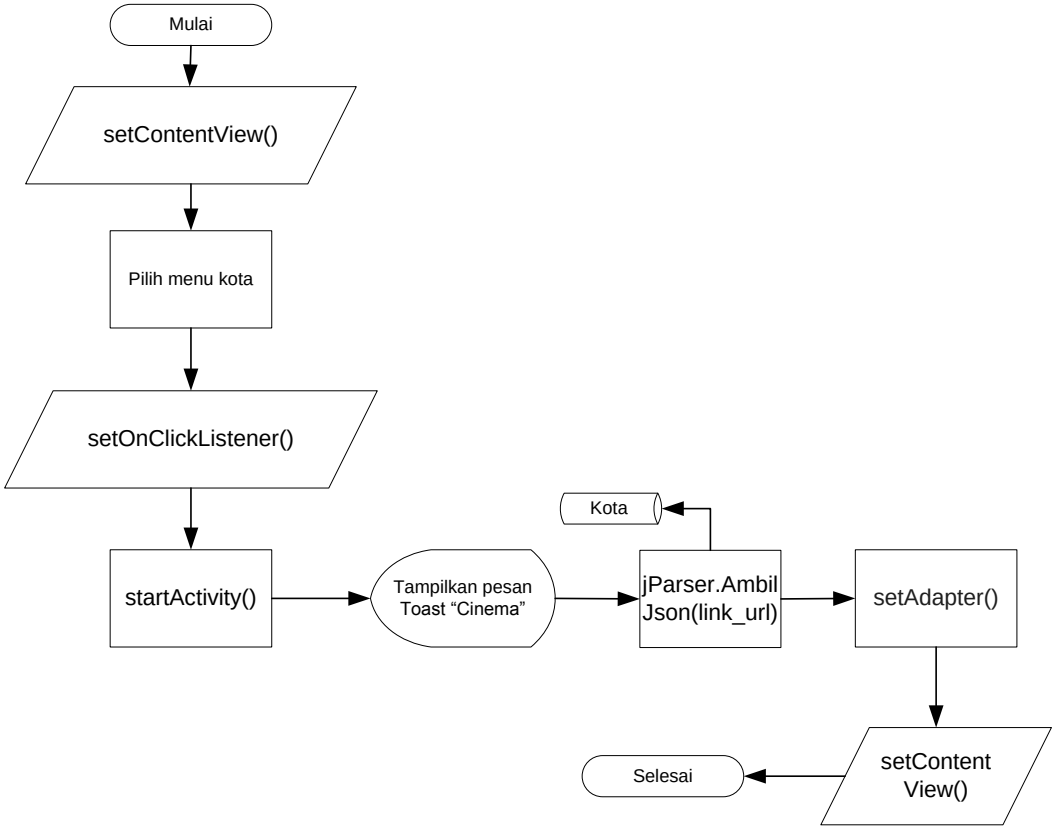
A09 Halaman tambah data kota

A10	Halaman *update* data kota
A11	Halaman tambah data bioskop
A12	Halaman *update* data bioskop
A13	Halaman tambah data film
A14	Halaman *update* data film
A15	Halaman playing at

3.2.6 Perancangan *Method*

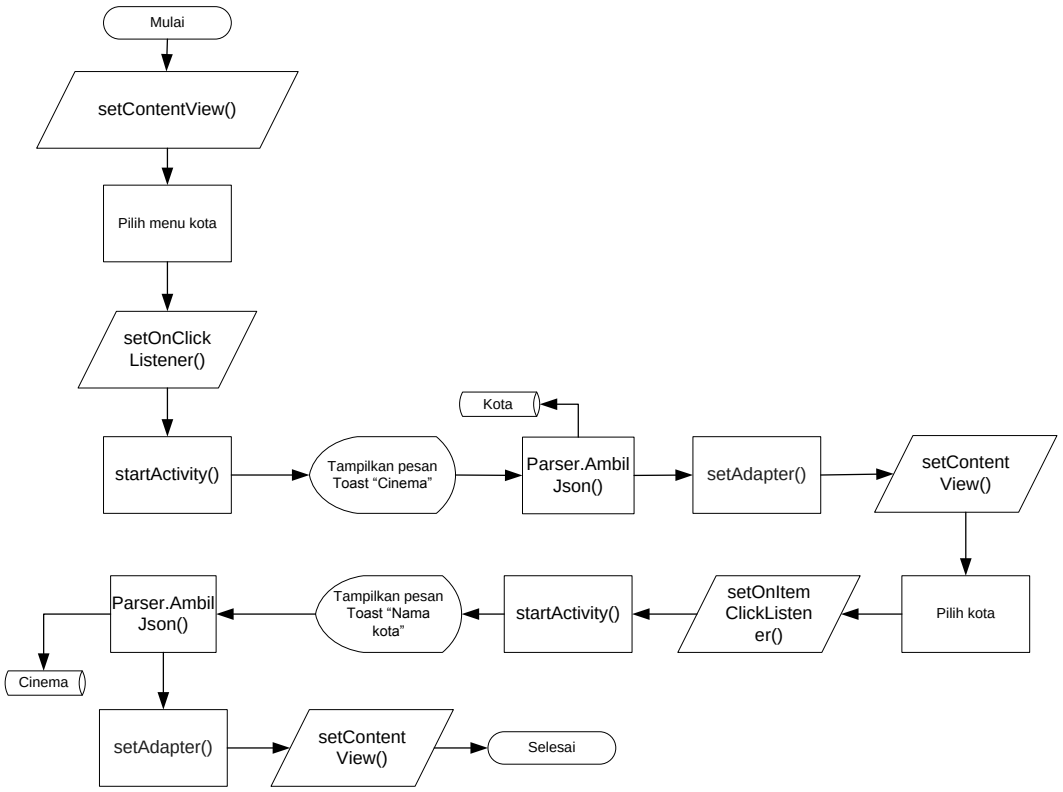
Sebagai langkah terakhir dalam proses perancangan yaitu merancang *method* yang akan diimplementasikan ke dalam sistem. *Method* ini akan digunakan sebagai algoritma dasar dalam mengkodekan prosedur yang ada. Adapun perancangan *method* untuk membangun aplikasi *mobile* yang akan dibangun adalah sebagai berikut :

1. *Flowchart* menu kota, *user* dapat melihat daftar kota. Alurnya dapat dilihat pada gambar 3.77.



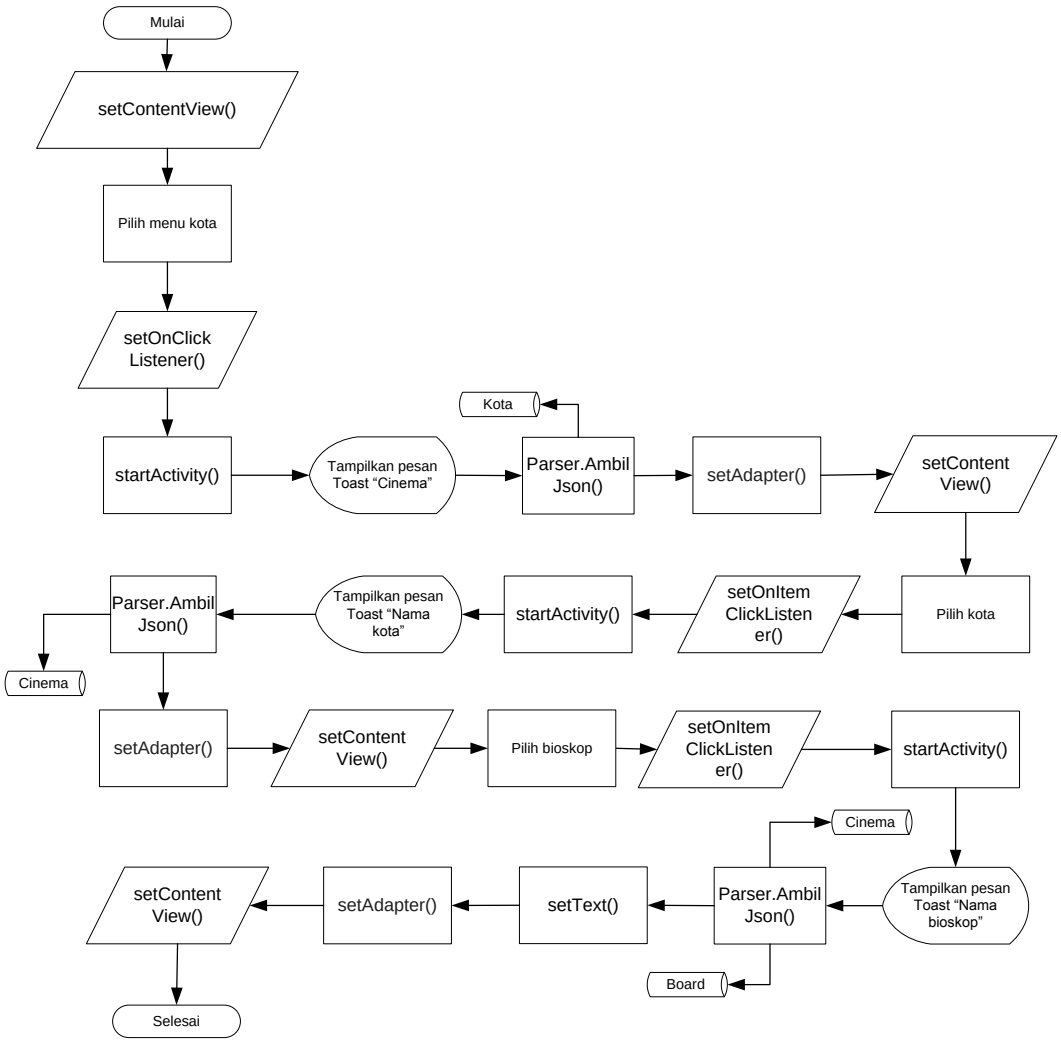
Gambar 3.77 *Flowchart* Menu Kota

2. *Flowchart* menu bioskop, *user* dapat melihat daftar bioskop sesuai dengan kota yang dipilih. Alurnya dapat dilihat pada gambar 3.78.



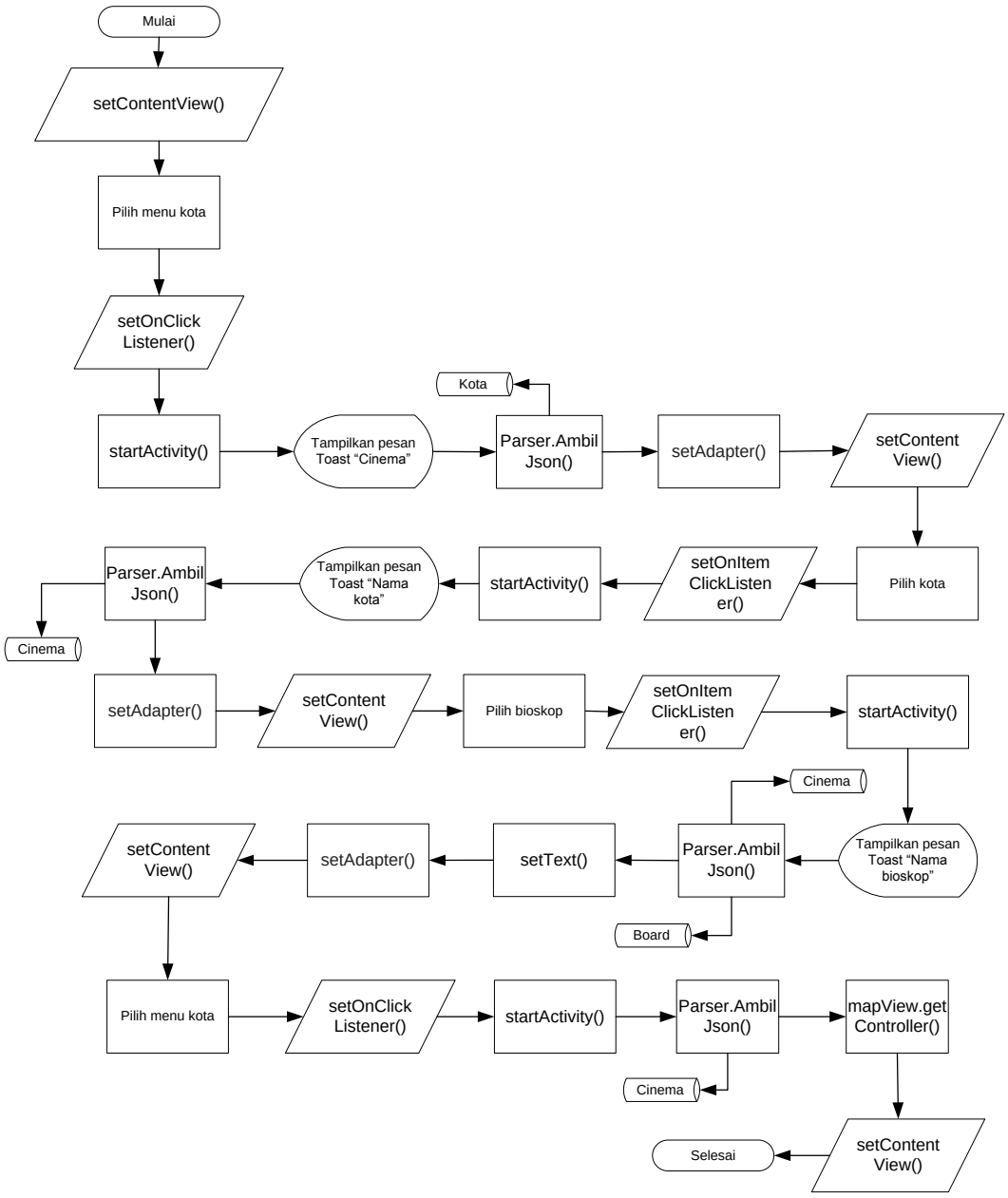
Gambar 3.78 *Flowchart* Menu Bioskop

3. *Flowchart* menu informasi bioskop, *user* dapat melihat informasi dari bioskop yang dipilih. Alurnya dapat dilihat pada gambar 3.79.



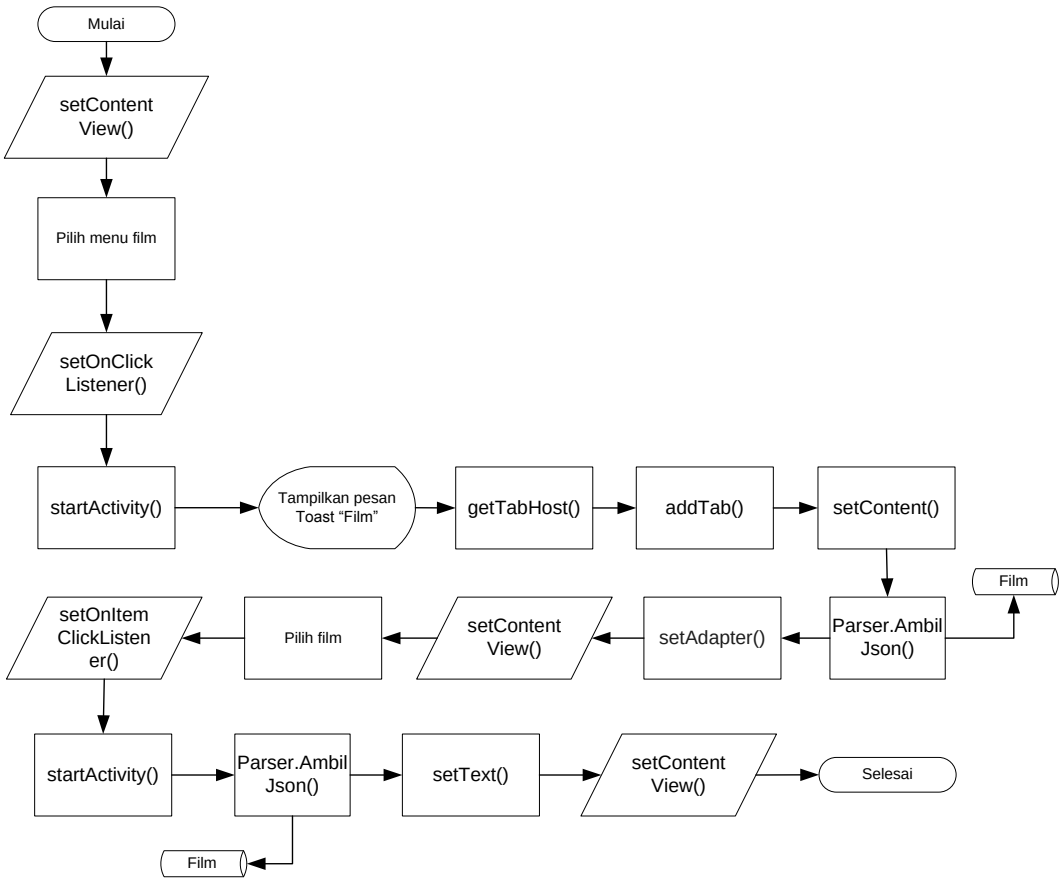
Gambar 3.79 *Flowchart* Menu Informasi Bioskop

4. *Flowchart* menu lokasi bioskop, *user* dapat melihat lokasi dari bioskop yang dipilih. Alurnya dapat dilihat pada gambar 3.80.



Gambar 3.80 *Flowchart* Menu Lokasi

5. *Flowchart* menu informasi film, *user* dapat melihat informasi dari film yang dipilih. Alurnya dapat dilihat pada gambar 3.81.



Gambar 3.81 *Flowchart* Menu Informasi Film