BAB III

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, kesempatankesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

3.1.1. Analisis Masalah

Dari hasil pengamatan diketahui bahwa jarang sekali sistem alikasi 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 masalah tersebut maka Penelitian ini akan mencoba membangun sebuah aplikasi *mobile* yang menyediakan berbagai informasi *lifestyle* menggunakan perangkat *Android* untuk mempermudah mereka dalam mengakses informasi yang sudah di sediakan oleh suatu web yang menyediakan informasi yang di yang berhubungan di bidangnya.

3.1.2. Proses Yang Terjadi Dalam Perangkat Lunak

Perangkat lunak dalam skripsi ini merupakan aplikasi *Content aggregator* dengan menggunakan Model parsing untuk mengambil berita atau informasi yang update di situs-situs web yang telah di tentukan.

Aplikasi ini akan menerima sebuah berita atau informasi dari sumber berita atau informasi. Berita atau informasi yang di tampilkan dalam aplikasi ini akan sama dan update sesuai dengan web yang telah di parsing oleh aplikasi .

3.1.3. Keluaran Perangkat Lunak

Output yang diharapkan dari sistem ini andalah berita atau informasi yang dibutuhkan oleh pengguna setelah pengguna tersebut mencari informasi sesuai dengan kebutuhan yang di inginkanya.

3.1.4. 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.

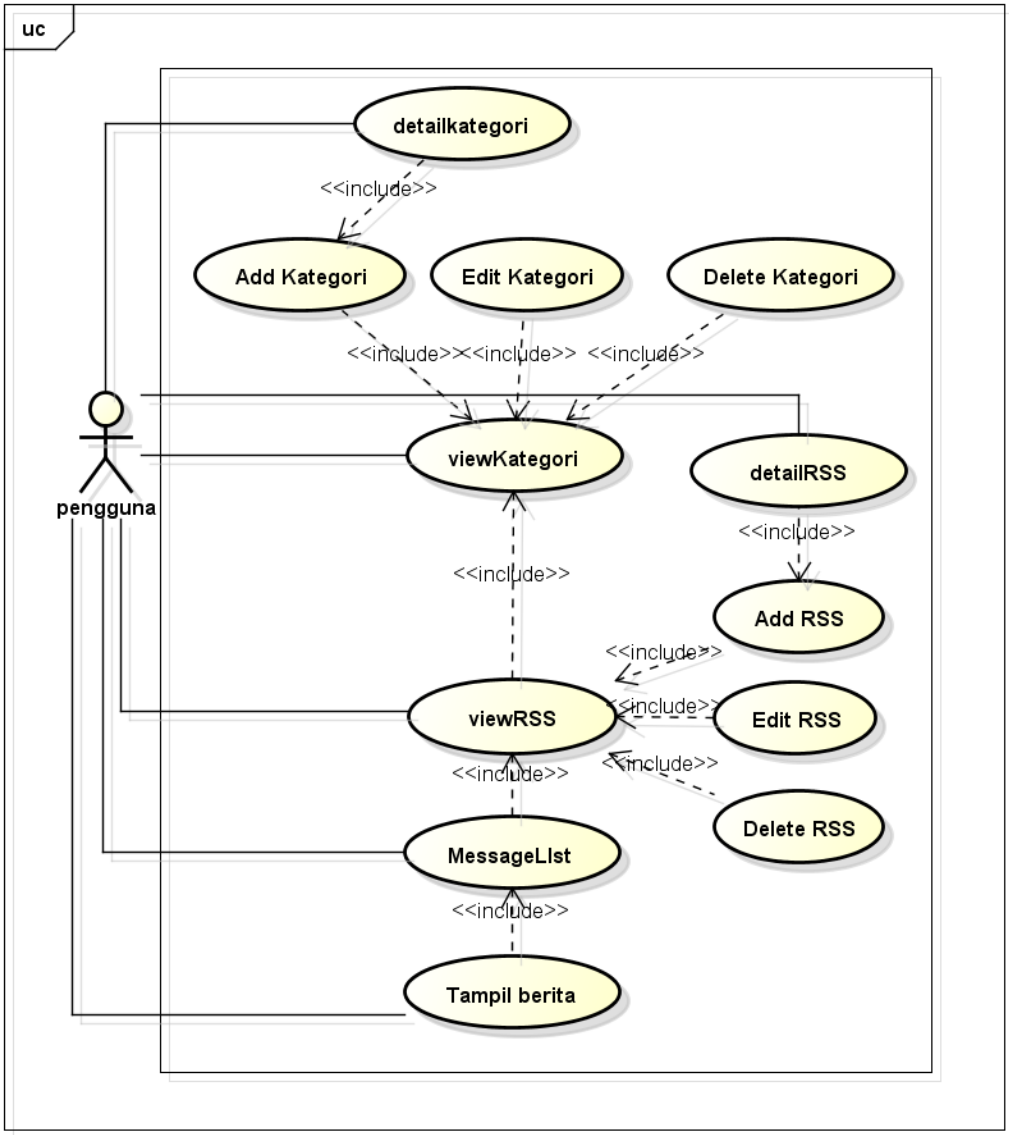
Analisis yang dilakukan dimodelkan dengan menggunakan UML (*Unified Modeling Language*). Tahap-tahap pemodelan dalam analisis tersebut antara lain identifikasi aktor, *usecase diagram, skenario, activity diagram, sequence diagram, class diagram*.

3.1.4.1. Identifikasi Aktor

Terdapat aktor yang berhubungan dengan *Content Aggregator*. Aktor pada sistem ini adalah user, user hanya mempunyai hak akses di halaman user untuk melakukan pencarian dokumen berita.

3.1.5. Usecase Diagram

Usecase Diagram merupakan konstruksi untuk mendeskripsikan hububungan-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* untuk aplikasi Content Aggregator dapatt di lihat pada gambar berikut



Gambar 3. 1 Usecase Diagram Aplikasi Content Aggregator

3.1.5.1. Definisi Usecase

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.

Tabel 3. 1 Definisi Usecase

No	Use Case	Deskripsi
1.	viewKategori	Berisi daftar list dari semua konten yang tersedia, menu pilihan *add, edit, delete*
2.	viewRSS	Berisi daftar dari alamat-alamat rss dari berbagai situs, menu pilihan *add, edit, delete*
3.	MessageList	Berisi daftar dari judul-judul berita yang telah di ambil dari suatu web
4.	detailkategori	Form untuk menambah atau mengedit kategori
5.	detailRSS	Form untuk menambah atau mengedit alamat rss
6.	Tampil berita	Tampilan berita dari sebuah web setelah kita memilih judul berita di MessageList
7.	Add Kategori	Menu untuk menambah kategori
8.	Edit Kategori	Menu untuk mengedit kategori
9.	Delete Kategori	Menu untuk menghapus kategori
10.	Add RSS	Menu untuk menambah alamat RSS
11.	Edit RSS	Menu untuk mengedit alamat RSS
12.	Delete RSS	Menu untuk menghapus alamat RSS

3.1.6. Skenario *Usecase*

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

3.1.6.1. *Usecase Skenario* View Kategori

Interaksi antara aktor pengguna dengan *use case* View Kategori dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 2 *Usecase Skenario* View kategori

Identifikasi	
Nama	viewKategori
Tujuan	Memastikan user agar bisa mesuk ke form utama untuk memilih konten yang di inginkan, menambah, mengedit dan menghapus kategori
Deskripsi	Menu ini akan menampilkan konten-konten yang sudah tersedia kepada user
Aktor	User
Skenario Utama	
Aksi Aktor	**Reaksi Sistem**
User membuka aplikasi	Sistem akan menampilkan form view kategori yang berisi konten-konten untuk dipilih oleh user
Skenario Alternatif – Autentikasi Gagal	
Aksi Aktor	**Reaksi Sistem**
User membuka kembali aplikasi	User dapat melihat kembali konten yang terdapat pada form view kategori, kemudian user dapat memilih kategori yang sesuai keinginan yang berisi alamat rss suatu website
Kondisi akhir	User dapat melihat dan memilih konten yang ada di dalam form view kategori

3.1.6.2. *Usecase Skenario* View RSS

Interaksi antara aktor pengguna dengan *use case view* RSS dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 3 *Usecase Skenario View* RSS

Identifikasi	
Nama	viewRSS
Tujuan	Memastikan user agar bisa mesuk ke form utama untuk memilih konten yang di inginkan, menambah, mengedit dan menghapus RSS yang tersedia
Deskripsi	Menu ini akan menampilkan RSS yang sudah tersedia dari beberapa situs kepada user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih konten mana yang akan dia buka,kemudian tampil alamat rss dari suatu website
Aksi Aktor	**Reaksi Sistem**
User membuka view rss	Sistem akan menampilkan alamat RSS yang sudah tersedia dari beberapa situs kepada user dan akan menampilkan judul berita dari alamat rss yang dipilih
Kondisi akhir	User dapat melihat dan memilih konten yang ada di dalam view RSS, juga dapat menambah, mengedit, dan menghapus RSS

3.1.6.3. *Usecase Skenario* MessageList

Interaksi antara aktor pengguna dengan *use case* MessageList dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 4 *Usecase Skenario* MessageList

Identifikasi	
Nama	MessageList
Tujuan	Agar dapat menampilkan judul berita dari suatu RSS yang telah di pilih oleh user
Deskripsi	Menu ini akan menampilkan judul berita dari suatu RSS yang telah du pilih oleh user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih judul berita yang tersedia di dalam form
Aksi Aktor	**Reaksi Sistem**
User membuka memlih judul berita	Sitem akan menampilkan halaman dari suatu website yang berisi berita sesuai dengan judul berita yang telah dipilih oleh user
Kondisi akhir	Sistem akan menampilkan berita sesuai dengan judul berita yang telah dipilih oleh user

3.1.6.4. *Usecase Skenario* Detail Kategori

Interaksi antara aktor pengguna dengan *use case* Detail Kategori dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 5 *Usecase Skenario Detail* Kategori

Identifikasi	
Nama	detailKategori
Tujuan	Agar dapat menambah dan mengedit kategori
Deskripsi	Form ini akan manambahkan atau mengedit kategori setelah user mengisikan data pada kolom yang tersedia
Aktor	User
Skenario Utama	
Kondisi awal	User mengisikan data yang akan ditambah atau diedit
Aksi Aktor	**Reaksi Sistem**
User menginput data	Sistem akan menyimpan data yang telah dimasukan oleh user dan menampilkannya di form view kategori sesuai dengan inputan dari user
Kondisi akhir	Sistem akan menampilkan data yang telah di isikan oleh user berupa nama kategori

3.1.6.5. *Usecase Skenario* Detail RSS

Interaksi antara aktor pengguna dengan *use case* Tambah RSS dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 6 *Usecase Skenario* Tambah RSS

Identifikasi	
Nama	detailrss
Tujuan	Agar dapat menambah dan mengedit *feed* RSS
Deskripsi	Form ini akan manambahkan atau mengedit *feed* RSS setelah user mengisikan data pada kolom yang tersedia
Aktor	User
Skenario Utama	
Kondisi awal	User mengisikan data yang akan ditambah atau diedit
Aksi Aktor	**Reaksi Sistem**
User menginput data	Sistem akan menyimpan data yang telah dimasukan oleh user dan menampilkannya di form view rss sesuai dengan inputan dari user
Kondisi akhir	Sistem akan menampilkan data yang telah di isikan oleh user berupa alamat rss dari suatu website

3.1.6.6. *Usecase Skenario* Tampil Berita

Interaksi antara aktor pengguna dengan *use* Tampil Berita Otomotif dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 7 *Usecase Skenario* Tampil Berita

Identifikasi	
Nama	Tampil berita
Tujuan	Agar dapat menampilkan berita setelah user memilih judul berita di MessageList
Deskripsi	Halaman ini akan menampilkan berita dari suatu halaman web site setelah user memilih judul berita di MessageLIst
Aktor	User
Skenario Utama	
Kondisi awal	User memilih judul berita di MessageLIst
Aksi Aktor	**Reaksi Sistem**
User Memilih judul berita	Sistem akan menampilkan suatu halaman website dan menampilkan berita sesuai dengan judul berita yang telah di pilih oleh user
Kondisi akhir	Sistem menampilkan suatu halam website dan menampilkan berita sesuai dengan judul yang telah dipilih oleh user

3.1.6.7. *Usecase Skenario Add* Kategori

Interaksi antara aktor pengguna dengan *use case Add* Kategori dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 8 *Usecase Add* Kategori

Identifikasi	
Nama	*Add* Kategori
Tujuan	Agar dapat menampilkan form untuk menambah kategori
Deskripsi	Menu ini akan menampilkan form untuk menambah kategori sesuai keinginan user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih menu option Add untuk menambah kategori yang di inginkan
Aksi Aktor	**Reaksi Sistem**
1. User memilih menu option Add	Sistem akan menampilkan form untuk menambah kategori
2. User menginput data	Sistem akan menyimpan data yang telah dimasukan oleh user
Kondisi akhir	Data yang telah di isi oleh user akan di simpan dan di tampilkan di dalam form view kategori

3.1.6.8. *Usecase Skenario Edit* Kategori

Interaksi antara aktor pengguna dengan *use case Edit* Kategori dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 9 *Usecase* Edit

Identifikasi	
Nama	*Edit* Kategori
Tujuan	Agar dapat menampilkan form untuk mengedit kategori
Deskripsi	Menu ini akan menampilkan form untuk mengedit kategori sesuai keinginan user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih menu option Edit untuk mengedit kategori yang di inginkan
Aksi Aktor	**Reaksi Sistem**
1. User memilih menu option Edit	Sistem akan menampilkan form yang berisi list kategori yang tersedia
2. User memilih kategori	Sistem akan menampilkan form untuk mengedit kategori agar user dapat mengisikan data
Kondisi akhir	Data yang telah di isi oleh user akan di simpan dan di tampilkan di dalam form view rss

3.1.6.9. *Usecase Skenario Delete* Kategori

Interaksi antara aktor pengguna dengan *use case Delete* Kategori dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 10 Usecase Skenario Delete Kategori

Identifikasi	
Nama	*Delete* Kategori
Tujuan	Agar dapat menampilkan form untuk menghapus kategori
Deskripsi	Menu ini akan menampilkan form untuk menghapus kategori sesuai keinginan user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih menu option edit untuk menghapus kategori yang di inginkan
Aksi Aktor	**Reaksi Sistem**
1. User memilih menu option Edit	Sistem akan menampilkan form yang berisi list kategori yang tersedia
2. User memilih kategori	Sistem akan menampilkan konfirmasi berupa message kepada user
Kondisi akhir	Data berhasil di hapus dan sistem akan menampilkan kembali form view kategori

3.1.6.10. *Usecase Skenario Add* RSS

Interaksi antara aktor pengguna dengan *use case Add* RSS dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 11 *use case Skenario Add* RSS

Identifikasi	
Nama	*Add* RSS
Tujuan	Agar dapat menampilkan form untuk menambah alamat RSS
Deskripsi	Menu ini akan menampilkan form untuk menambah alamat RSS sesuai keinginan user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih menu option Add untuk menambah alamat RSS yang di inginkan
Aksi Aktor	**Reaksi Sistem**
1. User memilih menu option Add	Sistem akan menampilkan form untuk menambah alamat rss
2. User menginput data	Sistem akan menyimpan data yang telah dimasukan oleh user
Kondisi akhir	Data berhasi di hapus dan sistem akan menampilkan kembali form view rss

3.1.6.11. *Usecase Skenario Edit* RSS

Interaksi antara aktor pengguna dengan *use case Edit* RSS dijelaskan dalam *use case* skenario sebagai berikut :

Tabel 3. 12 *Usecase Skenario Edit* RSS

Identifikasi	
Nama	*Edit* RSS
Tujuan	Agar dapat menampilkan form untuk mengedit alamat RSS
Deskripsi	Menu ini akan menampilkan form untuk mengedit alamat RSS sesuai keinginan user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih menu option Edit untuk mengedit alamat RSS yang di inginkan
Aksi Aktor	**Reaksi Sistem**
1. User memilih menu option Edit	Sistem akan menampilkan form untuk mengedit alamat rss sesuai keinginan user
2. User memilih alamat rss	Sistem akan menampilkan form untuk mengedit alamat rss agar user dapat mengisikan data
Kondisi akhir	Data yang telah diisi oleh user akan di simpan dan di tampilkan di dalam form view rss

3.1.6.12. *Usecase Skenario Delete* RSS

Interaksi antara aktor pengguna dengan *use case Delete* RSS dijelaskan dalam *use case* skenario sebagai berikut :

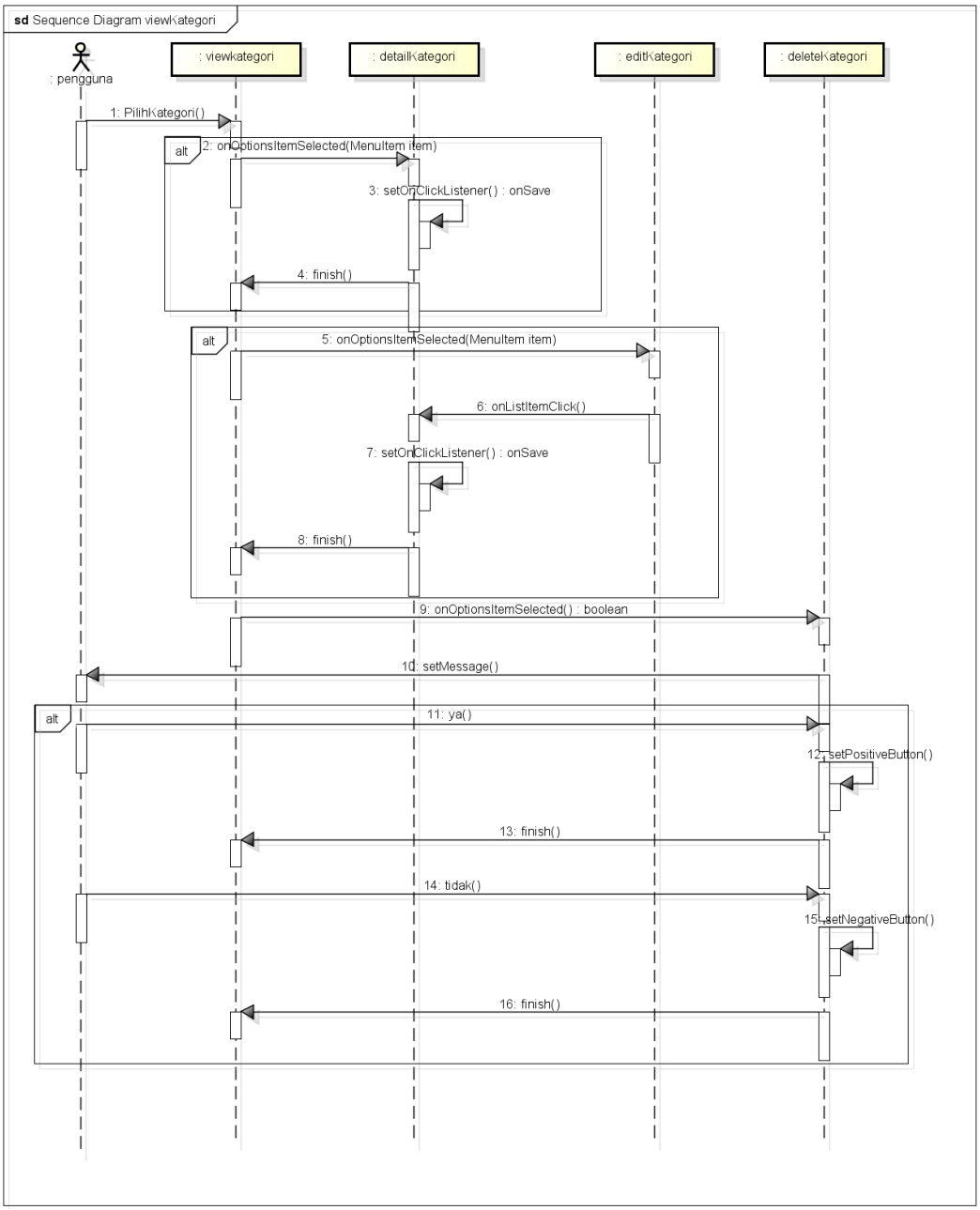
Tabel 3. 13 *Usecase Skenario Delete* RSS

Identifikasi	
Nama	*Delete* RSS
Tujuan	Agar dapat menampilkan form untuk menghapus RSS
Deskripsi	Menu ini akan menampilkan form untuk menghapus RSS sesuai keinginan user
Aktor	User
Skenario Utama	
Kondisi awal	User memilih menu option edit untuk menghapus RSS yang di inginkan
Aksi Aktor	**Reaksi Sistem**
1. User memilih menu option Edit	Sistem akan menampilkan form untuk menghapus alamat rss
2. User memilih alamat rss	Sistem akan menampilkan konfirmasi berupa message kepada user
Kondisi akhir	Data berhasil di hapus dan sistem akan menampilkan kembali form view rss

3.1.7. Sequence Diagram

Sequence diagram menggambarkan interaksi antar masing-masing objek pada setiap *use case* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi.

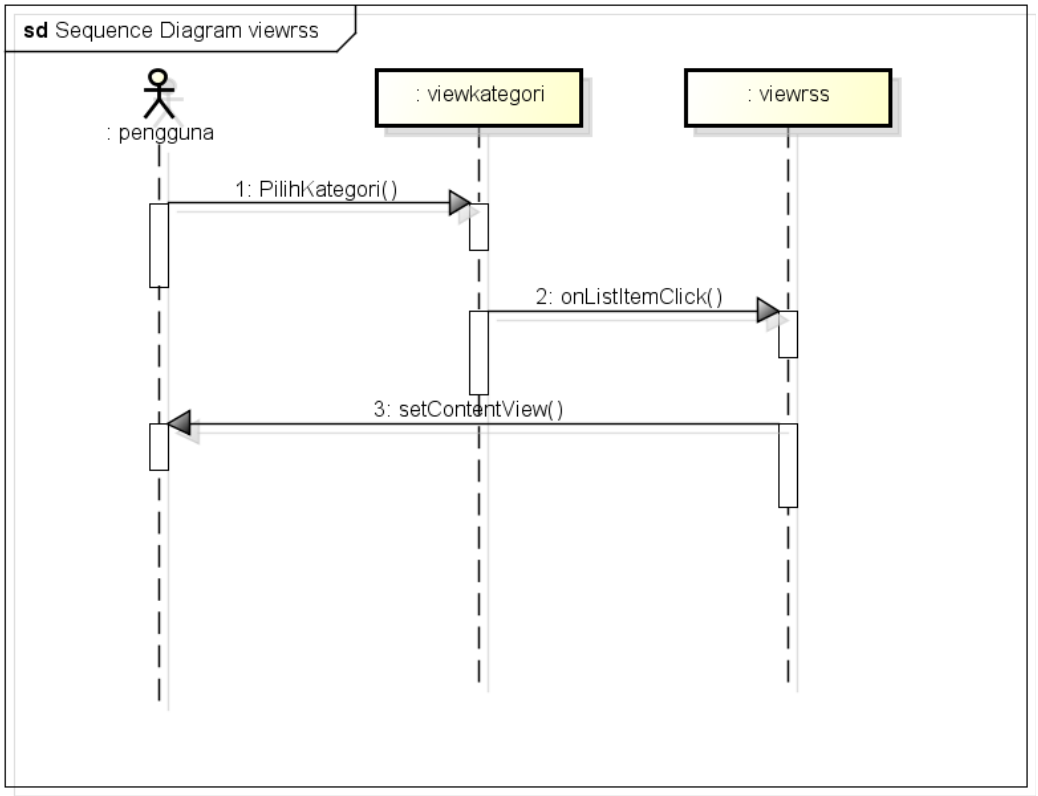
3.1.7.1. *Sequence Diagram* View Kategori



Gambar 3. 2 Sequence Diagram View Kategori

3.1.7.2. *Sequence Diagram* View RSS

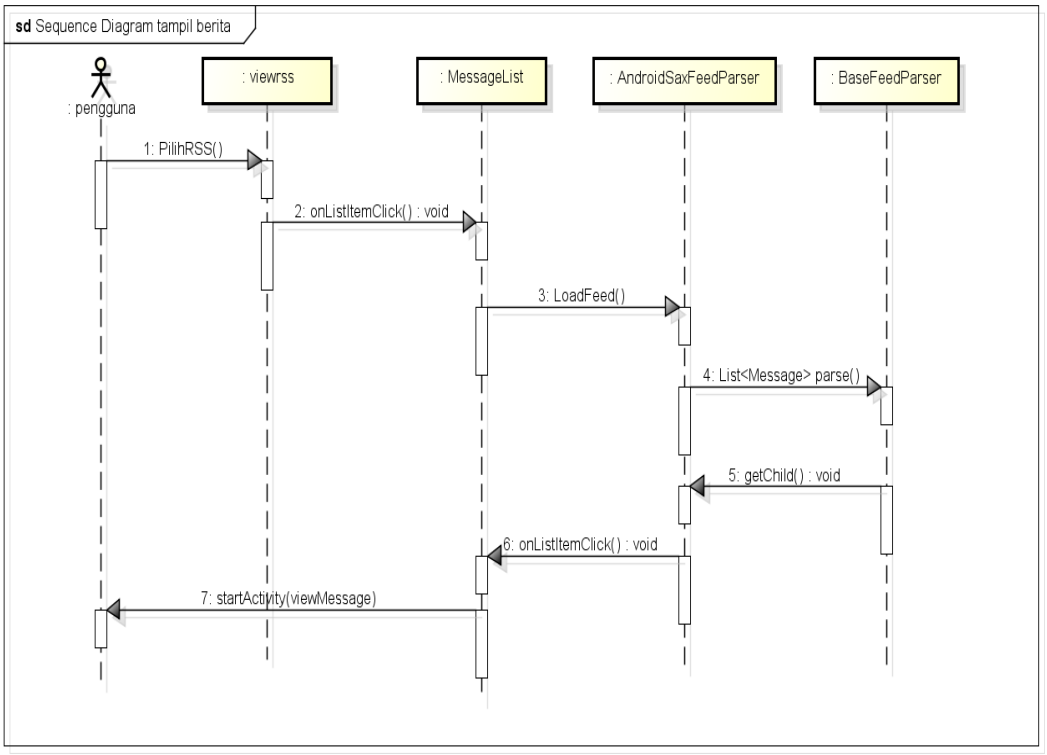
Sequence diagram view RSS menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat alamat RSS yang tersedia dalam aplikasi *mobile Content Aggregator*. *Sequence diagram* view RSS dapat dilihat pada Gambar 3.3



Gambar 3. 3 Sequence Diagram View RSS

3.1.7.3. *Sequence Diagram* Tampil Berita

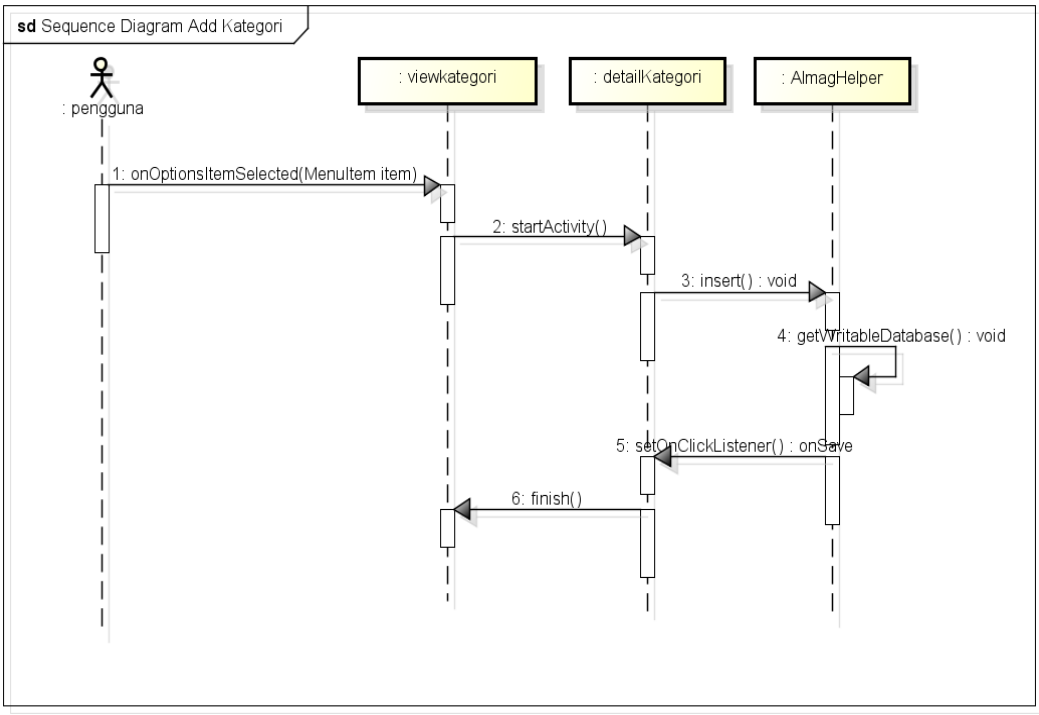
Sequence diagram Tampil berita menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses melihat berita dari suatu web site yang tersedia dalam aplikasi *mobile Content Aggregator*. *Sequence diagram* tampil berita dapat dilihat pada Gambar 3.4



Gambar 3. 4 *Sequence diagram* Tampil berita

3.1.7.4. *Sequence Diagram Add* Kategori

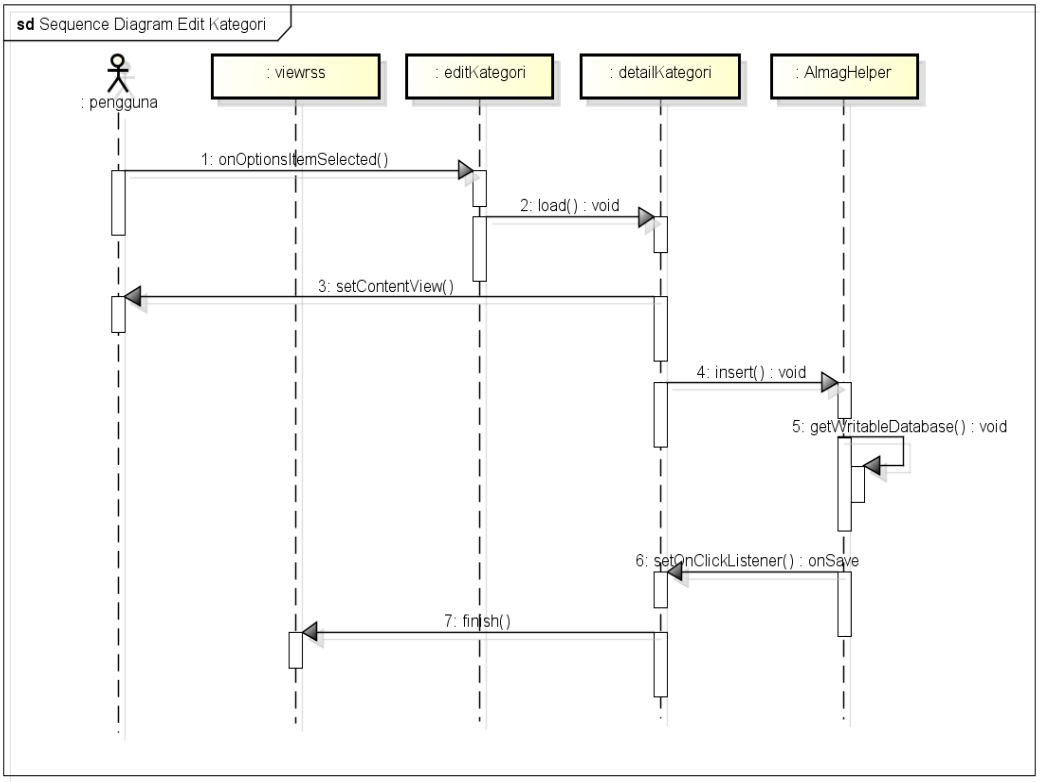
Sequence diagram Add Kategori menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses menambah kategori baru dalam aplikasi *mobile Content Aggregator*. *Sequence diagram* Add Kategori dapat dilihat pada Gambar 3.5



Gambar 3. 5 *Sequence diagram Add* Kategori

3.1.7.5. *Sequence Diagram Edit* Kategori

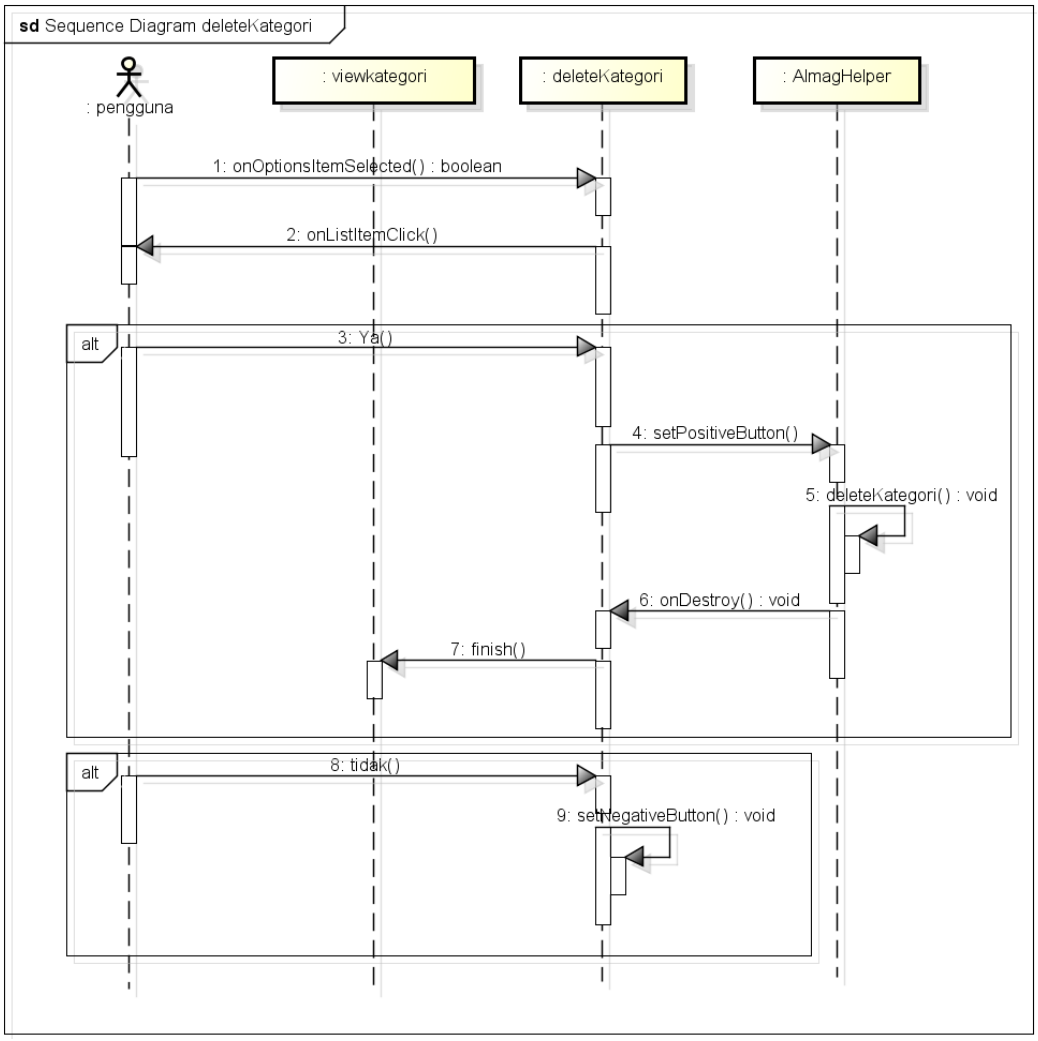
Sequence diagram Edit Kategori menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses merubah kategori dalam aplikasi *mobile Content Aggregator*. *Sequence diagram Edit* Kategori dapat dilihat pada Gambar 3.6



Gambar 3. 6 *Sequence diagram Edit* Kategori

3.1.7.6. *Sequence Diagram Delete* Kategori

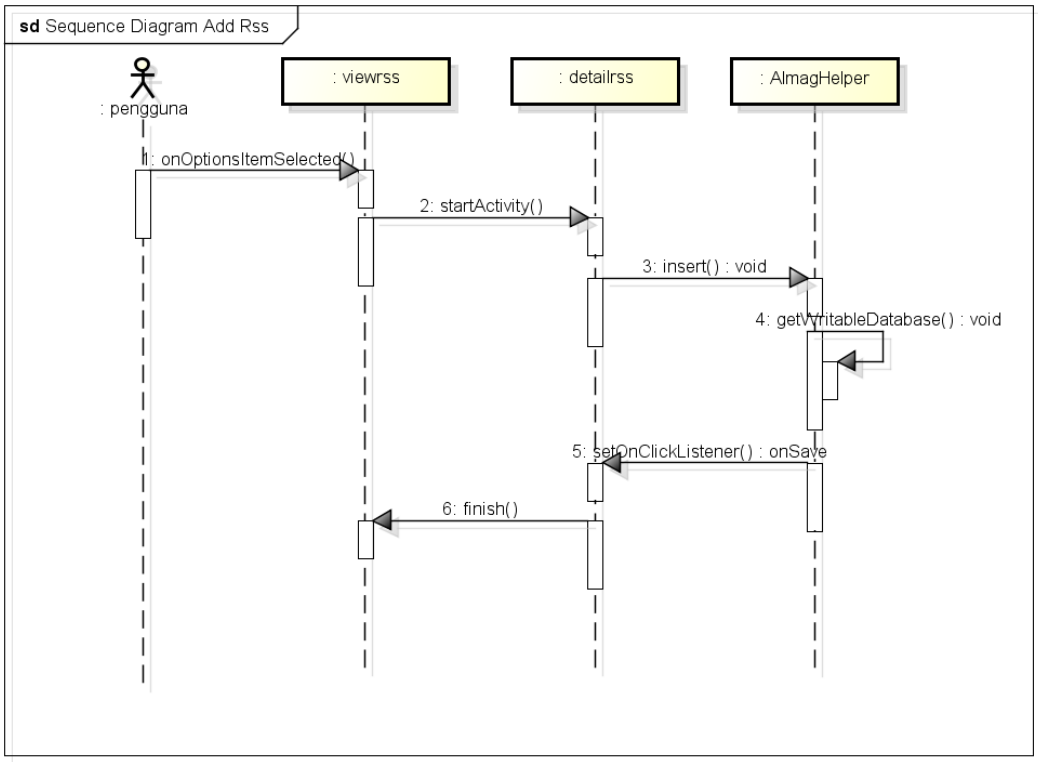
Sequence diagram Delete Kategori menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses menghapus kategori dalam aplikasi *mobile Content Aggregator*. *Sequence diagram Delete* Kategori dapat dilihat pada Gambar 3.7



Gambar 3. 7 *Sequence diagram Delete* Kategori

3.1.7.7. *Sequence Diagram Add* RSS

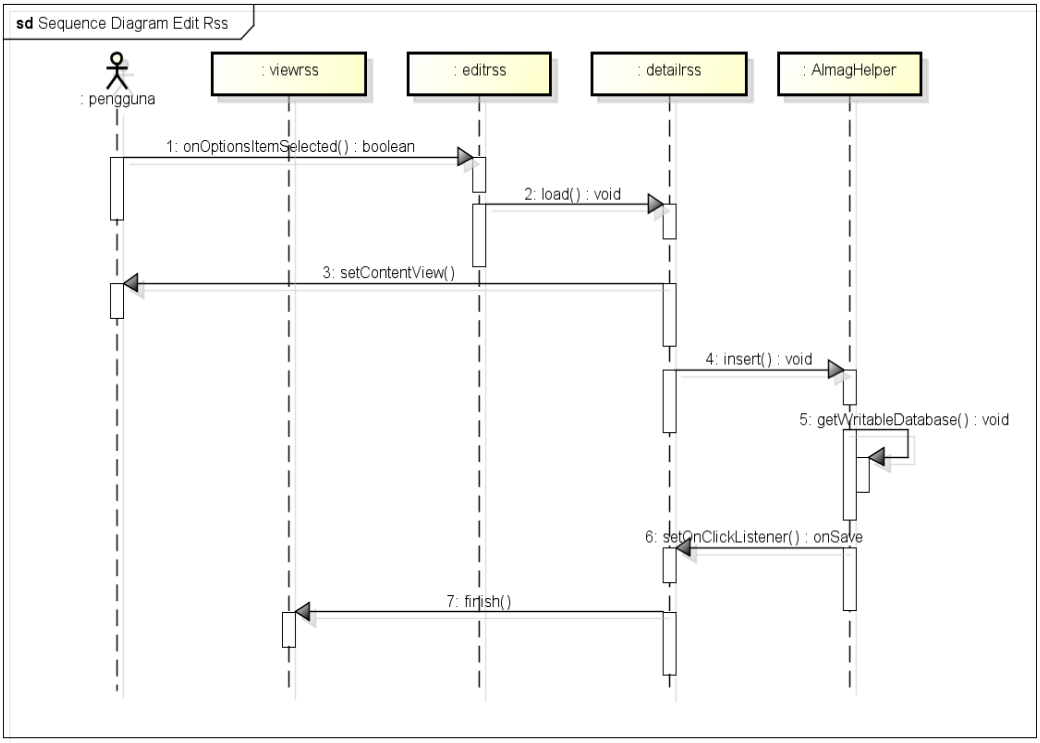
Sequence diagram Add RSS menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses menambah alamat RSS baru dalam aplikasi *mobile Content Aggregator*. *Sequence diagram Add* RSS dapat dilihat pada Gambar 3.8



Gambar 3. 8 *Sequence Diagram Add* RSS

3.1.7.8. *Sequence Diagram Edit* RSS

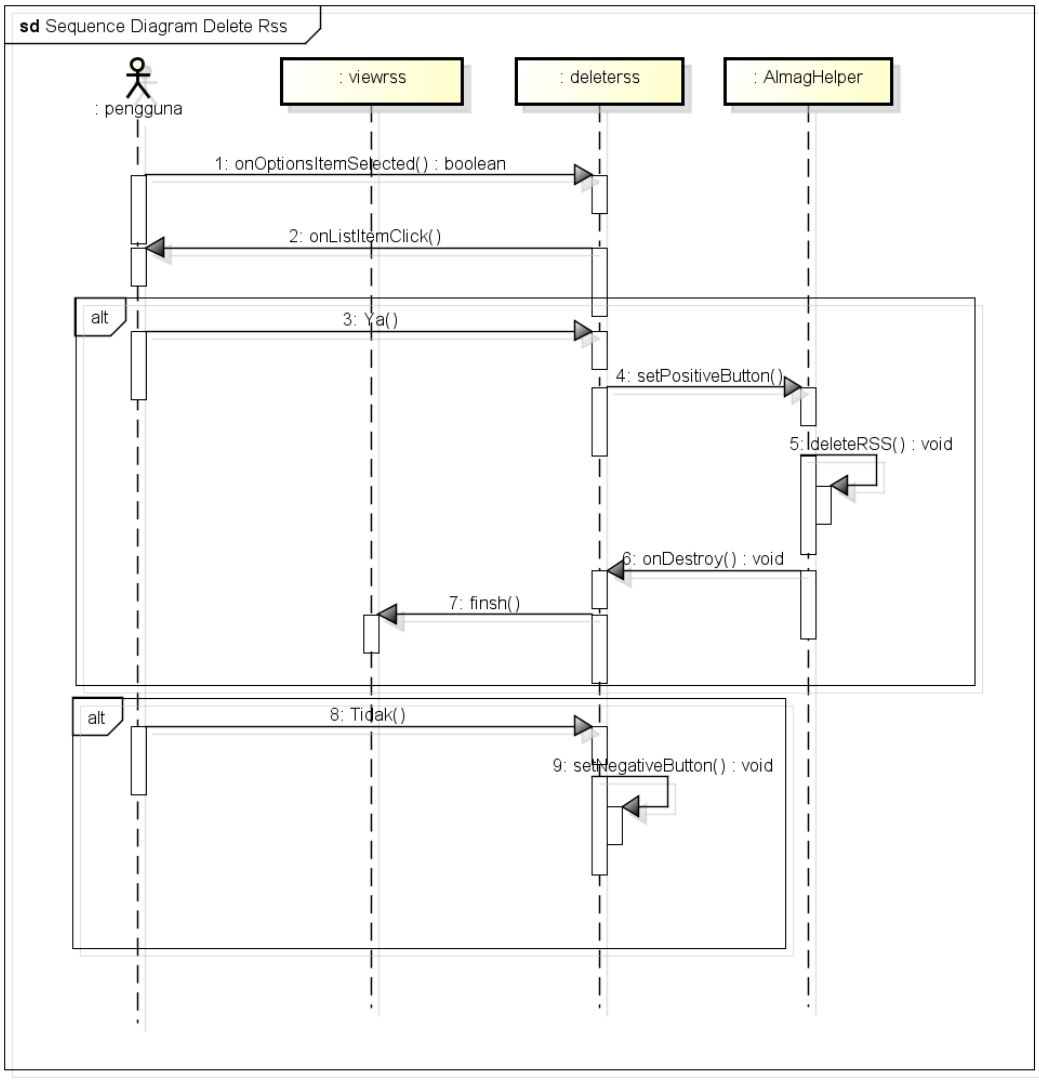
Sequence diagram Edit RSS menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses merubah alamat RSS dalam aplikasi *mobile Content Aggregator*. *Sequence diagram Edit* RSS dapat dilihat pada Gambar 3.9



Gambar 3. 9 *Sequence Diagram Edit* RSS

3.1.7.9. *Sequence Diagram Delete* RSS

Sequence diagram Delete RSS menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses menghapus alamat RSS dalam aplikasi *mobile Content Aggregator*. *Sequence diagram Delete* RSS dapat dilihat pada Gambar 3.10



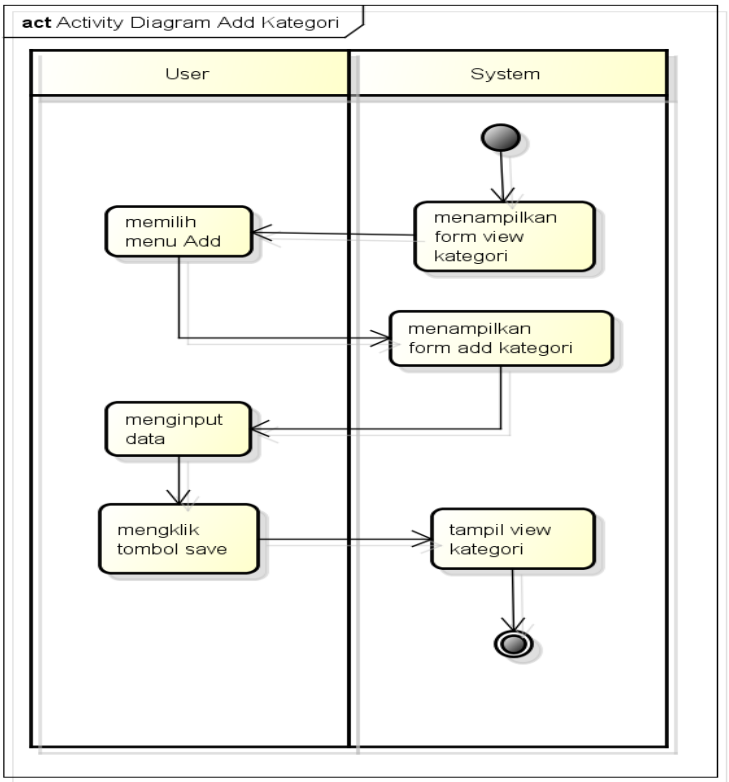
Gambar 3. 10 *Sequence diagram Delete* RSS

3.1.8. *Activity Diagram*

Activity diagram merupakan diagram yang memodelkan aliran kerja atau *workflow* dari urutan aktifitas dalam suatu proses yang mengacu pada *use case diagram* yang ada. Berikut ini penjelasan dari tiap *activity diagram*:

3.1.8.1. *Activity Diagram Add* Kategori

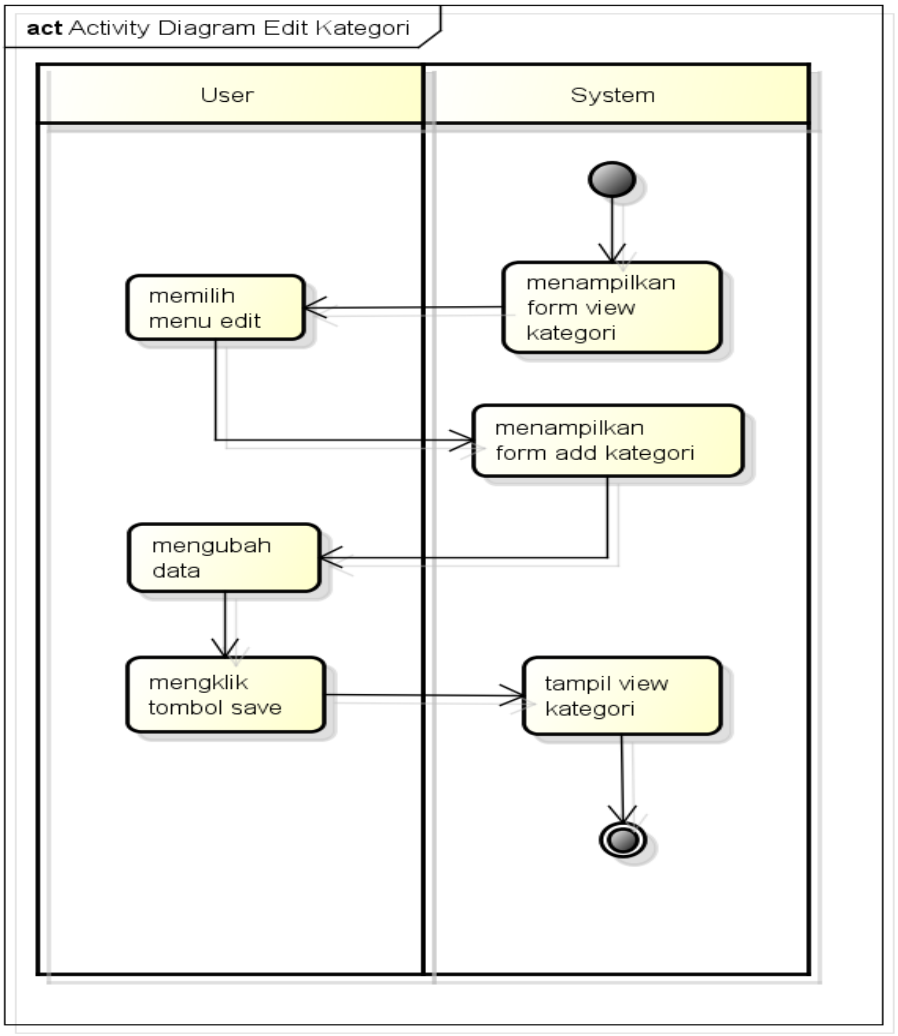
Activity diagram Add Kategori menjelaskan aliran kerja aktor pengguna pada saat menambahkan kategori baru ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram Add* Kategori dapat dilihat pada gambar berikut ini.



Gambar 3. 11 *Activity diagram Add* Kategori

3.1.8.2. *Activity Diagram Edit* Kategori

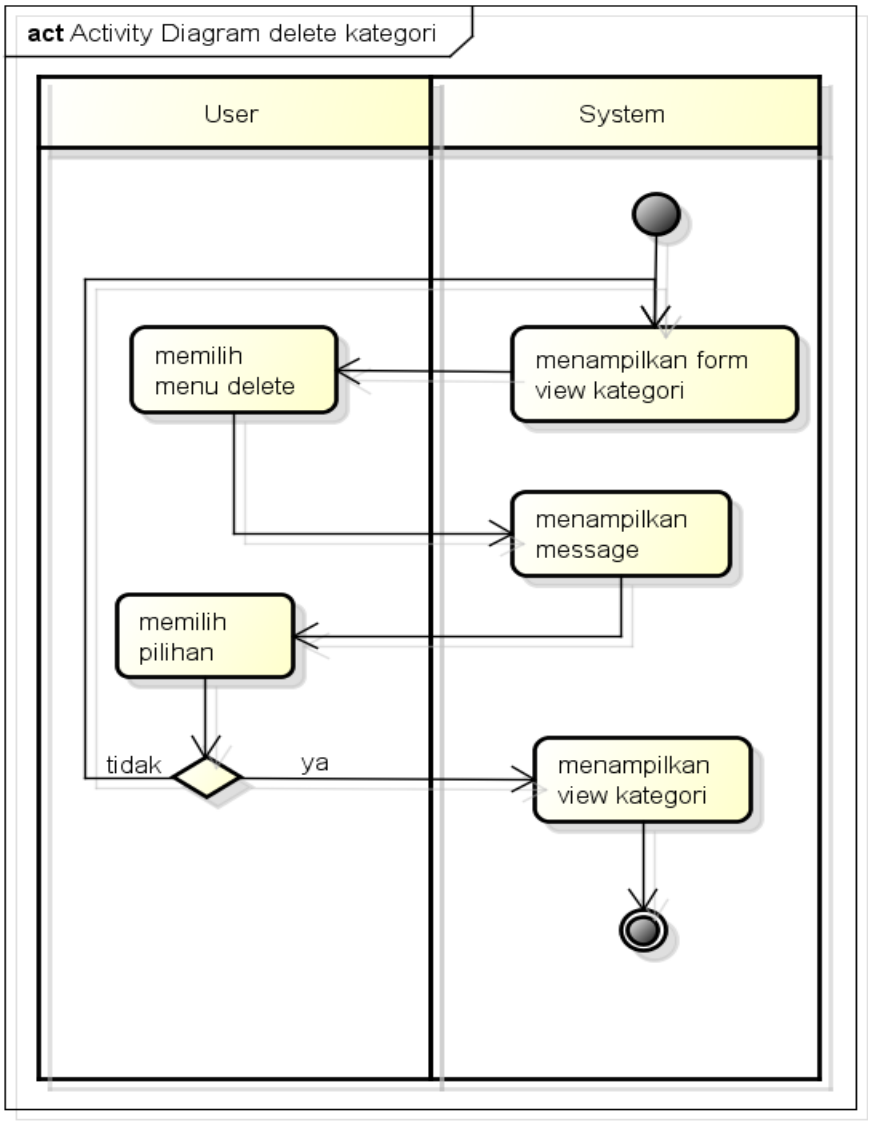
Activity diagram Edit Kategori menjelaskan aliran kerja aktor pengguna pada saat merubah kategori ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram Edit* Kategori dapat dilihat pada gambar berikut ini.



Gambar 3. 12 *Activity diagram Edit* Kategori

3.1.8.3. *Activity Diagram Delete* Kategori

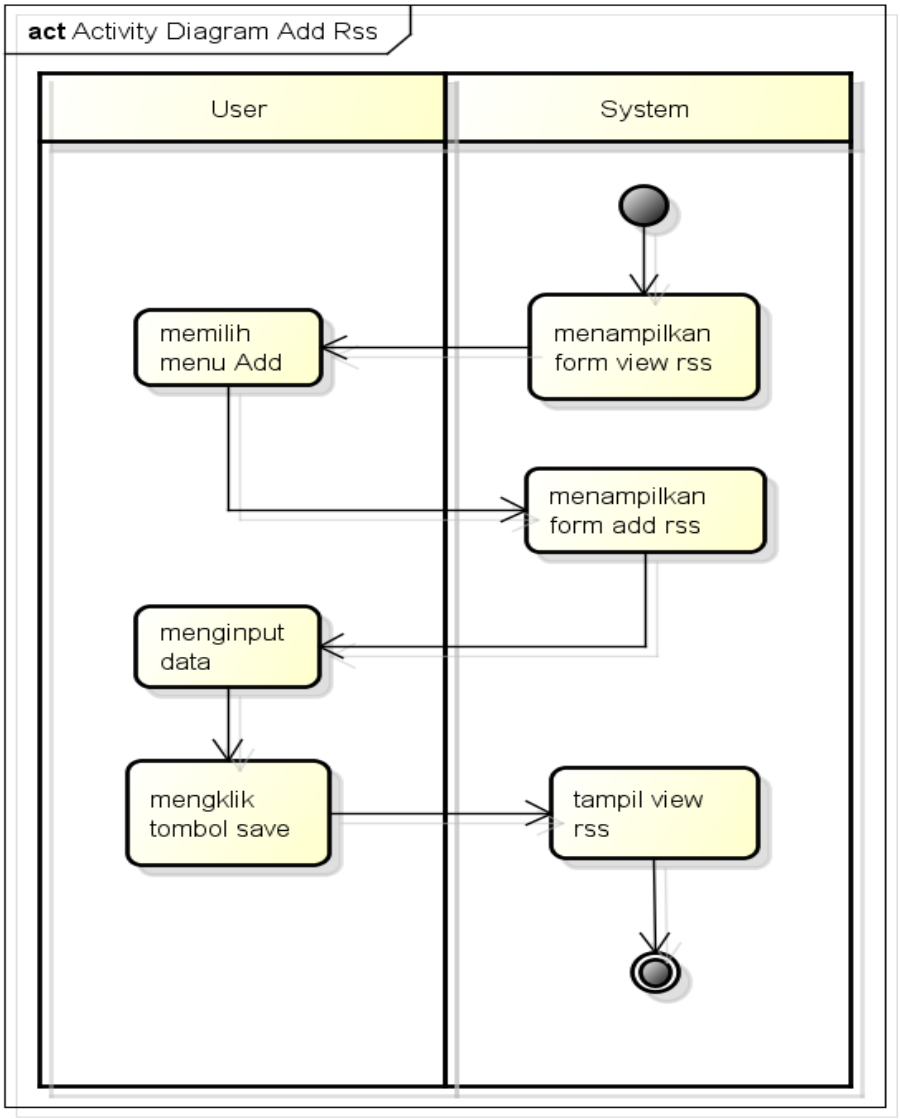
Activity diagram Delete Kategori menjelaskan aliran kerja aktor pengguna pada saat menghapus kategori ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram Delete* Kategori dapat dilihat pada gambar berikut ini.



Gambar 3. 13 *Activity diagram Delete* Kategori

3.1.8.4. *Activity Diagram Add* RSS

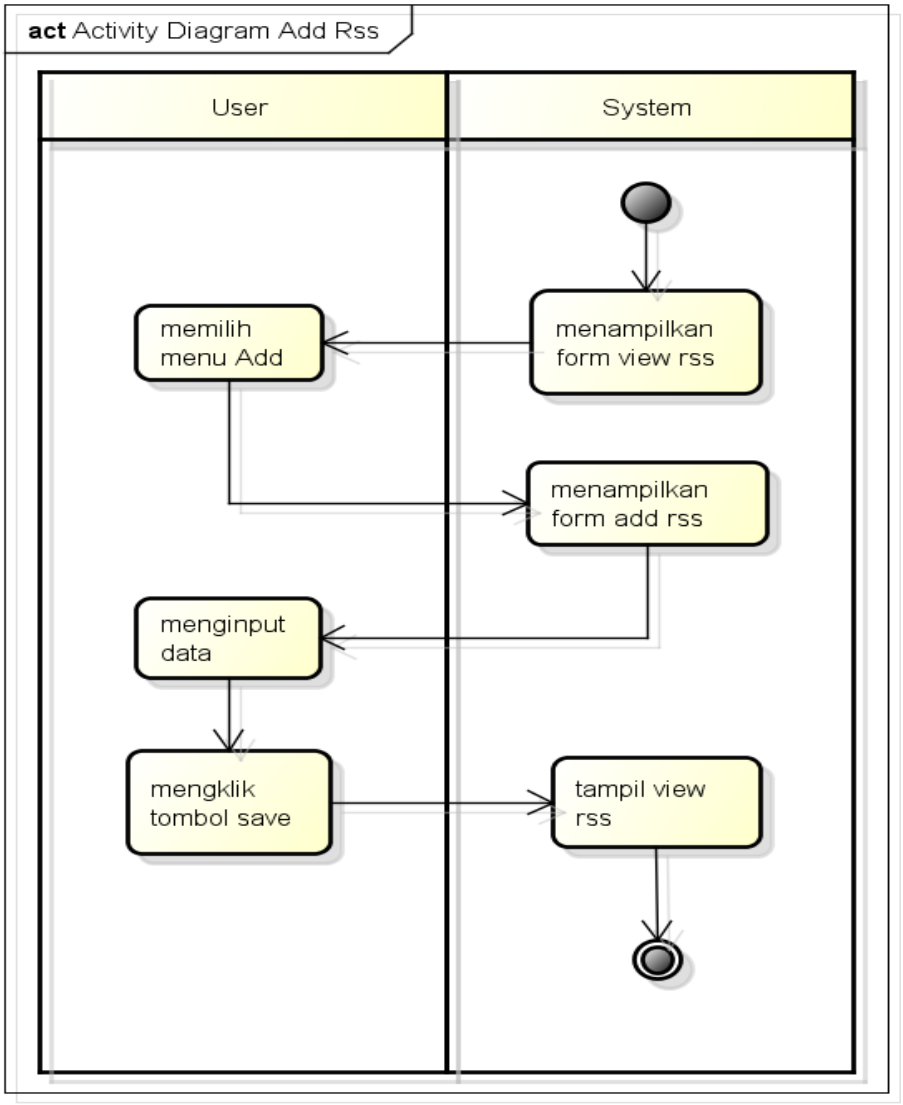
Activity diagram Add RSS menjelaskan aliran kerja aktor pengguna pada saat menambahkan alamat RSS baru ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram Add* RSS dapat dilihat pada gambar berikut ini.



Gambar 3. 14 *Activity diagram Add* RSS

3.1.8.5. *Activity Diagram Edit* RSS

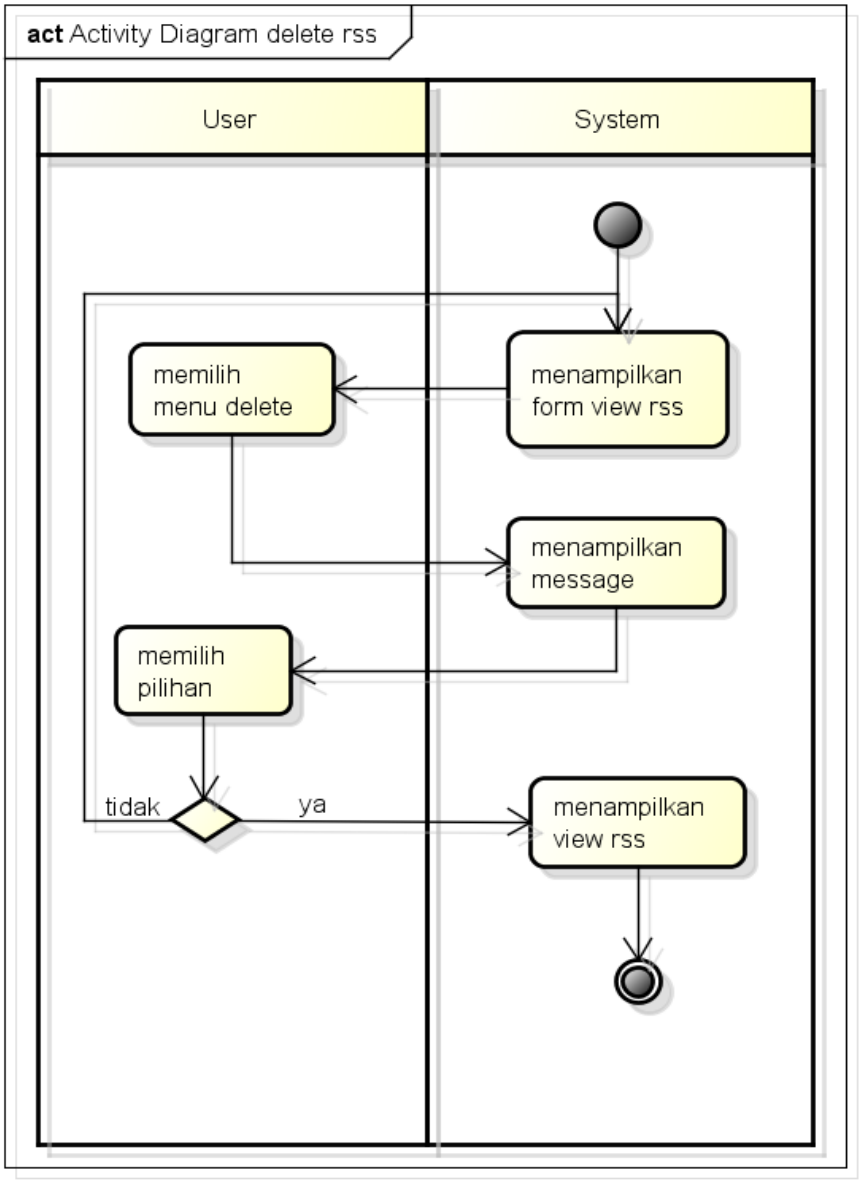
Activity diagram Edit RSS menjelaskan aliran kerja aktor pengguna pada saat merubah alamat RSS ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram Edit RSS* dapat dilihat pada gambar berikut ini.



Gambar 3. 15 *Activity diagram Edit* RSS

3.1.8.6. *Activity Diagram Delete* RSS

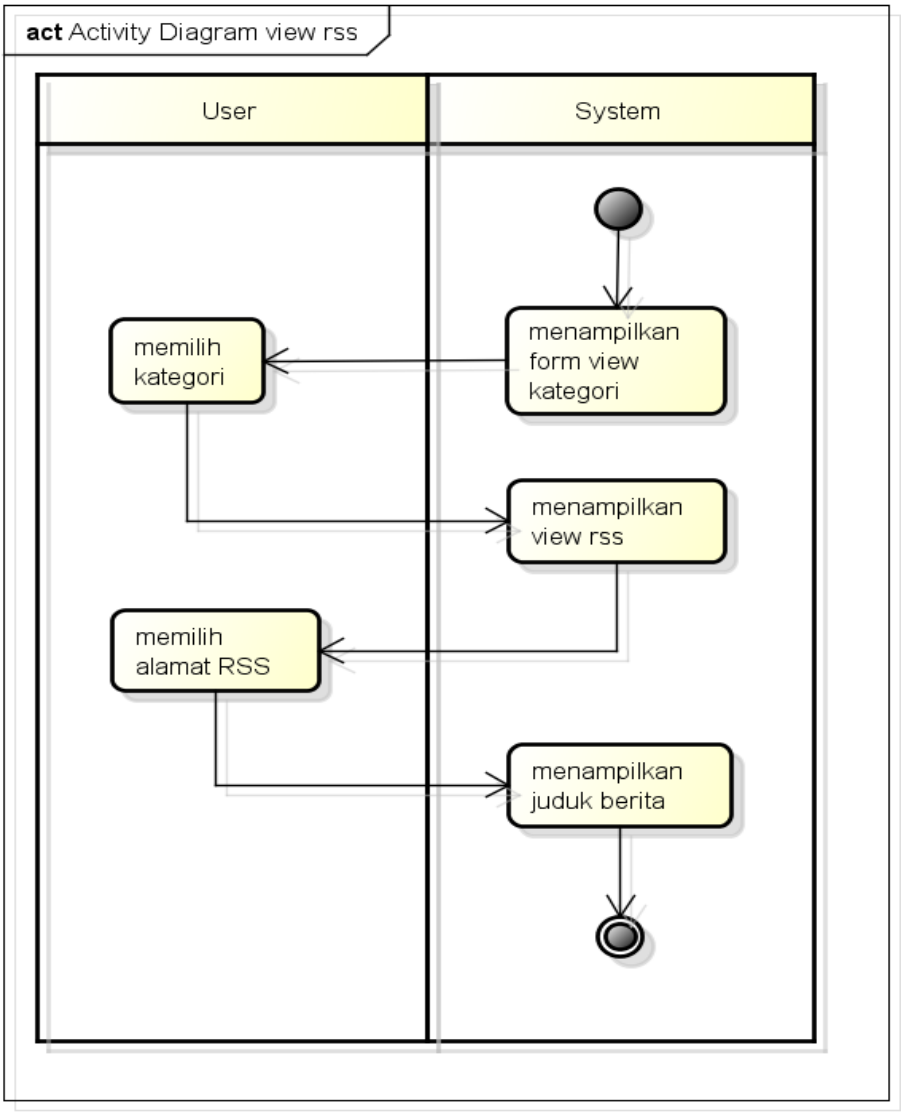
Activity diagram Delete RSS menjelaskan aliran kerja aktor pengguna pada saat menghapus alamat RSS ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram Delete* RSS dapat dilihat pada gambar berikut ini.



Gambar 3. 16 *Activity diagram Delete* Kategori

3.1.8.7. *Activity Diagram View* RSS

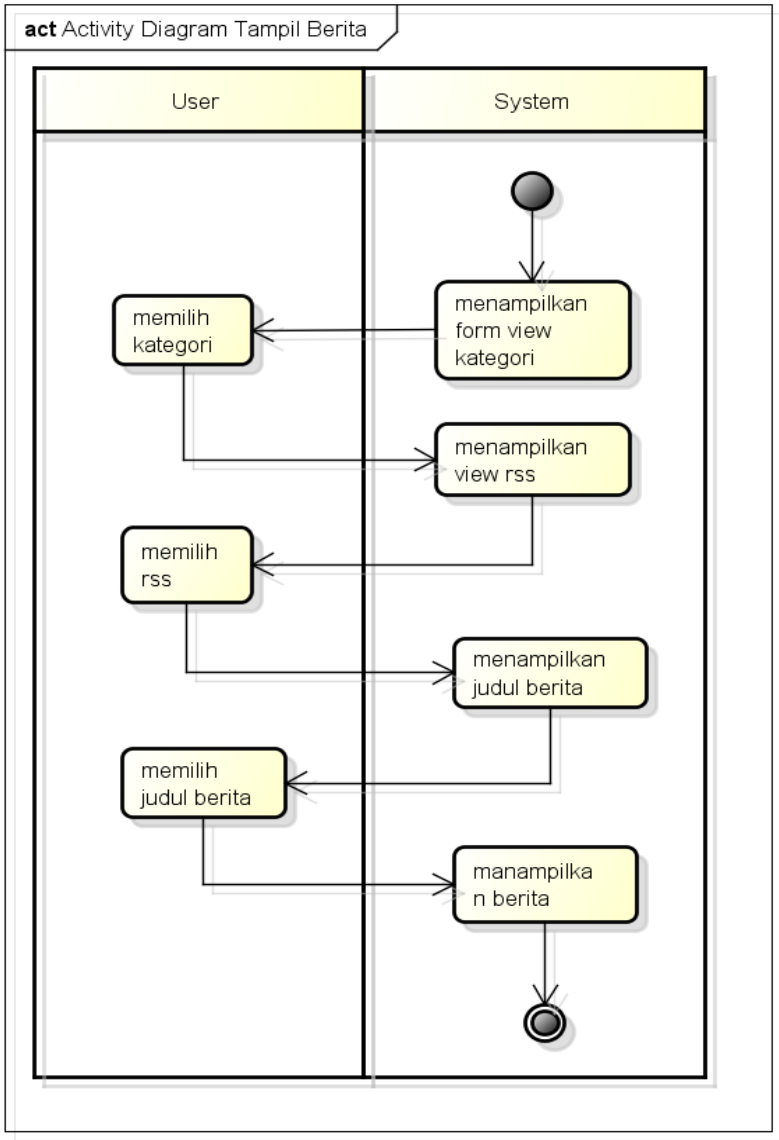
Activity diagram View RSS menjelaskan aliran kerja aktor pengguna pada saat melihat alamat RSS ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram View* RSS dapat dilihat pada gambar berikut ini.



Gambar 3. 17 *Activity diagram View* RSS

3.1.8.8. *Activity Diagram* Tampil Berita

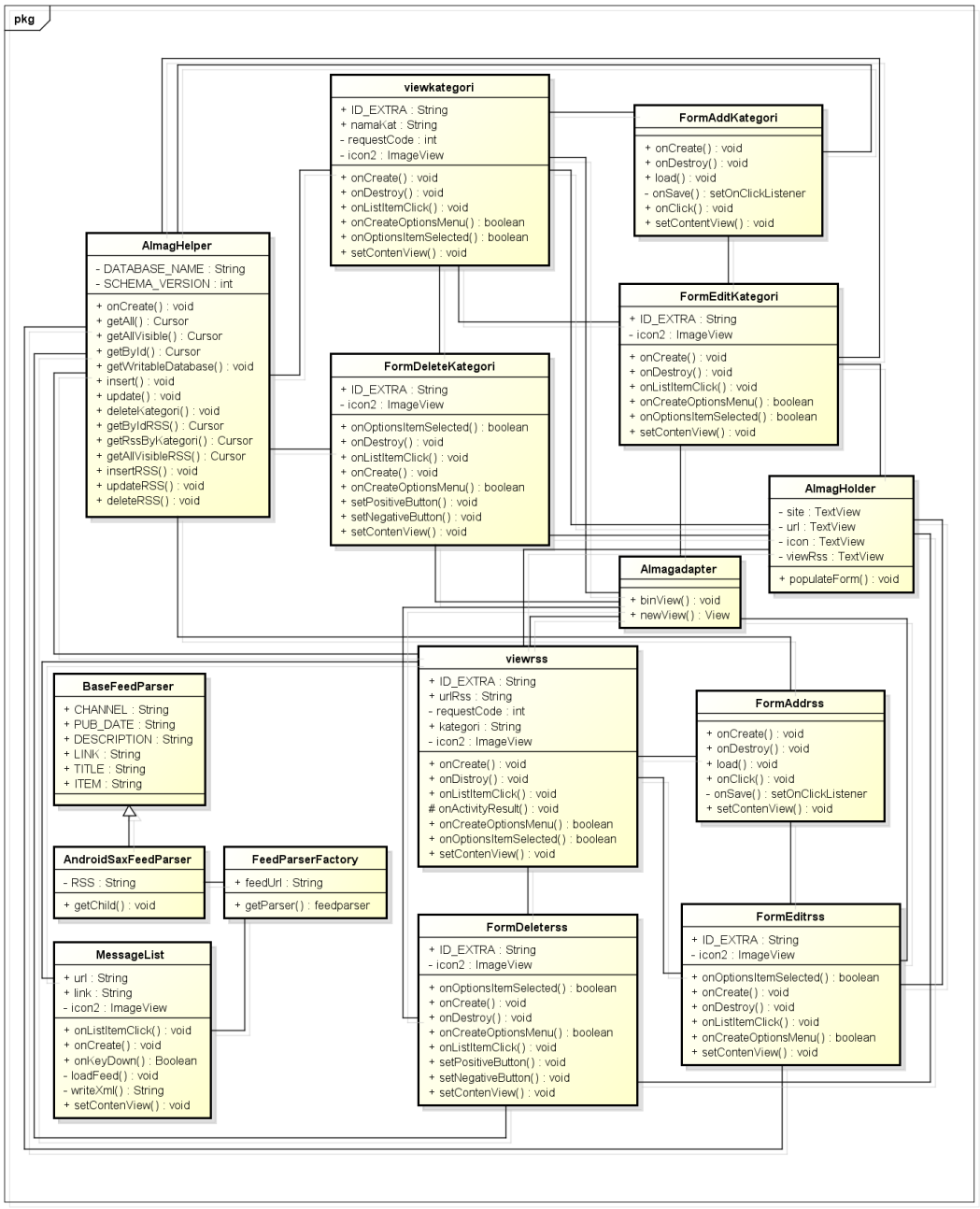
Activity diagram Tampil Berita menjelaskan aliran kerja aktor pengguna pada saat melihat alamat RSS ke dalam aplikasi *mobile Conten Aggregator*. *Activity diagram* Tampil Berita dapat dilihat pada gambar berikut ini.



Gambar 3. 18 Activity diagram Tampil Berita

3.1.9. 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 berikut



Gambar 3. 19 ***Class Diagram***

3.1.10. Operasi dan Atribut

Nama Kelas : viewkategori

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
onListItemClick	public	Pemanggilan activity
onCreateOptionMenu	public	Pembuatan menu
onOptionItemSelected	public	Pemanggilan activity dengan memilih menu
setContenView	public	Pengesetan layout
Nama Atribut	***Visibility (private, public)***	***Tipe***
ID_EXTRA	public	String
namaKat	public	String
requestCode	private	Int
icon2	private	ImageView

Nama Kelas : FormAddKategori

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
load	public	Pengesetan cursor
onClick	public	Pemilihan status kategori
onSave	private	Menyimpan data
setContenView	public	Pengesetan layout
Nama Atribut	***Visibility (private, public)***	***Tipe***

Nama Kelas : FormEditKategori

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
onListItemClick	public	Pemanggilan activity
onCreateOptionMenu	public	Pembuatan menu
onOptionItemSelected	public	Pemanggilan activity dengan memilih menu
startActivity	public	
setContenView	public	Pengesetan layout

Nama Atribut	***Visibility (private, public)***	***Tipe***
ID_EXTRA	public	String
icon2	private	ImageView

Nama Kelas : FormDeleteKategori

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
onListItemClick	public	Pemanggilan alert dialog
onCreateOptionMenu	public	Pembuatan menu
onOptionItemSelected	public	Pemanggilan activity dengan memilih menu
setPositiveButton	public	Pengesetan Alert Dialog
setNegativeButton	public	Pengesetan Alert Dialog
setContenView	public	Pengesetan layout
Nama Atribut	***Visibility (private, public)***	***Tipe***
ID_EXTRA	public	String
icon2	private	ImageView

Nama Kelas : viewrss

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
onListItemClick	public	Pemanggilan activity
onCreateOptionsMenu	public	Pembuatan menu
onOptionItemSelected	public	Pemanggilan activity dengan memilih menu
onActivityResult	protected	Pembuatan ProgressDialog
setContenView	public	Pengesetan layout
Nama Atribut	***Visibility (private, public)***	***Tipe***
ID_EXTRA	public	String
urlrss	public	String
requestCode	private	Int
kategori	public	String
icon2	private	ImageView

Nama Kelas : FormAddrss

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
load	public	Pengesetan cursor
onClick	public	Pemilihan status rss
onSave	public	Menyimpan data
setContenView	public	Pengesetan layout
Nama Atribut	***Visibility (private, public)***	***Tipe***

Nama Kelas : FormEditrss

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
onListItemClick	public	Pemanggilan activity
onCreateOptionMenu	public	Pembuatan menu
onOptionItemSelected	public	Pemanggilan activity dengan memilih menu
getCheckedRadioButtonId	public	Pemilihan status kategori
setContenView	public	Pengesetan layout
Nama Atribut	***Visibility (private, public)***	***Tipe***
ID_EXTRA	public	String
icon2	private	ImageView

Nama Kelas : FormDeleterss

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onDestroy	public	Menutup class helper
onListItemClick	public	Pemanggilan alert dialog
onCreateOptionsMenu	public	Pembuatan menu
onOptionItemSelected	public	Pemanggilan activity dengan memilih menu
setPositiveButton	public	Pengesetan Alert Dialog
setNegativeButton	public	Pengesetan Alert Dialog
setContenView	public	Pengesetan layout
Nama Atribut	***Visibility (private, public)***	***Tipe***

ID_EXTRA	public	String
icon2	private	ImageView

Nama Kelas : BaseFeedParser

Nama Operasi	***Visibility (private, public)***	***Keterangan***
Nama Atribut	***Visibility (private, public)***	***Tipe***
CHANEL	public	String
PUB_DATE	public	String
DESCRIPTION	public	String
LINK	public	String
TITLE	public	String
ITEM	public	String

Nama Kelas : AndroidSaxFeedParser

Nama Operasi	***Visibility (private, public)***	***Keterangan***
getChild	public	
Nama Atribut	***Visibility (private, public)***	***Tipe***
RSS	private	String

Nama Kelas : MessageList

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan layout
onKeyDown	public	Mengaktifkan tombol pada keyboard
onListItemClick	public	Pemanggilan activity
loadFeed	public	Pemanggilan class FeedParserFactory
writeXml	public	Pengesetan dokumen xml
Nama Atribut	***Visibility (private, public)***	***Tipe***
url	public	String
link	public	String
icon2	private	ImageView

Nama Kelas : FeedFactorFactory

Nama Operasi	***Visibility (private, public)***	***Keterangan***
getParser	public	Mengambil type parser
Nama Atribut	***Visibility (private, public)***	***Tipe***
feedUrl	public	String

Nama Kelas : AlmagAdapter

Nama Operasi	***Visibility (private, public)***	***Keterangan***
binView	public	Menampilkan data dari database ke dalam form
newView	public	Menampilkan jumlah baris
Nama Atribut	***Visibility (private, public)***	***Tipe***

Nama Kelas : AlmagHolder

Nama Operasi	***Visibility (private, public)***	***Keterangan***
populateForm	public	Mengisikan data ke dalam form
Nama Atribut	***Visibility (private, public)***	***Tipe***
site	private	TextView
url	private	TextView
icon	private	ImageView
viewRss	private	TextView

Nama Kelas : AlmagHelper

Nama Operasi	***Visibility (private, public)***	***Keterangan***
onCreate	public	Pembuatan database
getAll	public	Menampilkan isi field dari tabel kategori
getAllVisible	public	Menampilkan isi field dari tabel kategori berdasarkan status kategori
getById	public	Menampilkan isi filed dari tabel kategori berdasarkan id
insert	public	Menambahkan data kedalam tabel kategori
update	public	Mengupdate data dari tabel

		kategori
deleteKategori	public	Menghapus data dari tabel kategori
getWritableDatabase	public	Menyimpan data masukan
getByIdRss	public	Menampilkan isi filed dari tabel rss berdasarkan id
getRssByKategori	public	Menampilkan isi field dari tabel rss berdasarkan nama kategori
getAllVisibleRss	public	Menampilkan isi field dari tabel rss berdasarkan nama kategori
insertRss	public	Menambahkan data kedalam tabel rss
updatRss	public	Mengupdate data dari tabel kategori
deleteRss	public	Menghapus data dari tabel kategori
Nama Atribut	***Visibility (private, public)***	***Tipe***
DATABSE_NAME	private	String
SCHEMA_VERSION	private	Int

3.1.11. 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 content Aggregator 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.

3.1.11.1. Analisis dan Kebutuhan Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pembuatan aplikasi content aggregator ini adalah :

1. Sistem Operasi Microsoft Windows 7
2. Eclipse Helios

3.1.11.2. 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 content aggregator ini adalah :

1. Processor : Intel Core i3 M330 2.13GHz
2. RAM : 2 Giga
3. VGA : Onboard
4. HDD : 320 Giga
5. Monitor : 14”

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. Perancangan Data

Perancangan data merupakan tahapan untuk memetakan model konseptual ke model basis data yang akan digunakan. Perancangan data dari aplikasi *mobile Content Aggregator* 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 berikut:

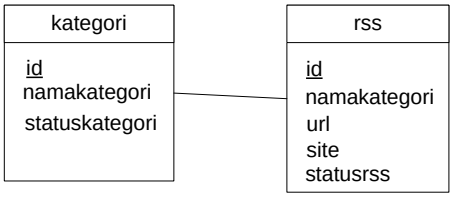
Tabel 3. 14 Perancangan Data Kategori

Nama Field	Tipe Data	Keterangan
id	int	Primary key
namakategori	txt	
status	txt	

Tabel 3. 15 perancangan Data Rss

Nama Field	Tipe Data	Keterangan
id	int	Primary key
namakategori	txt	
status	txt	
url	txt	
Site	txt	

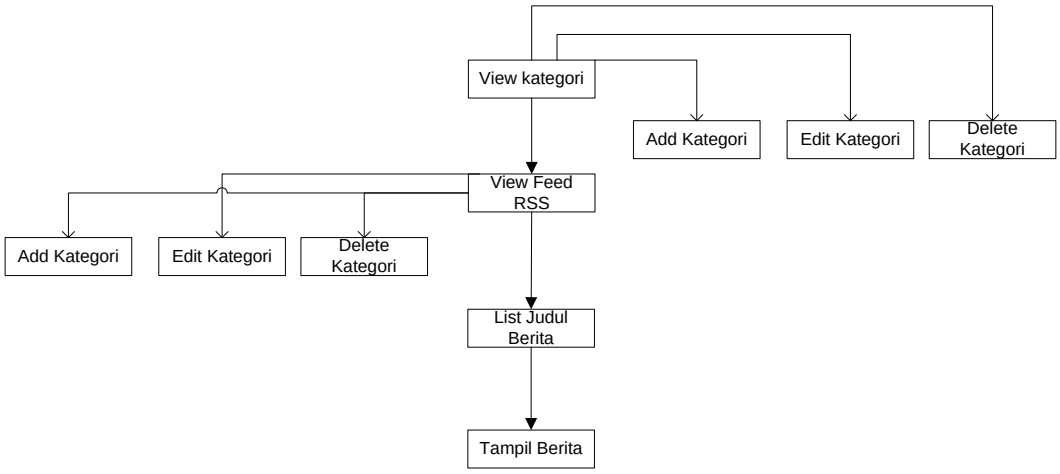
3.2.2. Skema Relasi



Gambar 3. 20 Skema Relasi

3.2.3. Struktur Menu

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi. Perancangan struktur menu dari aplikasi *mobile Content Aggregator* dapat dilihat pada gambar berikut ini.



Gambar 3. 21 Struktur Menu aplikasi *mobile Content Aggregator*

3.2.4. Perancangan Antarmuka

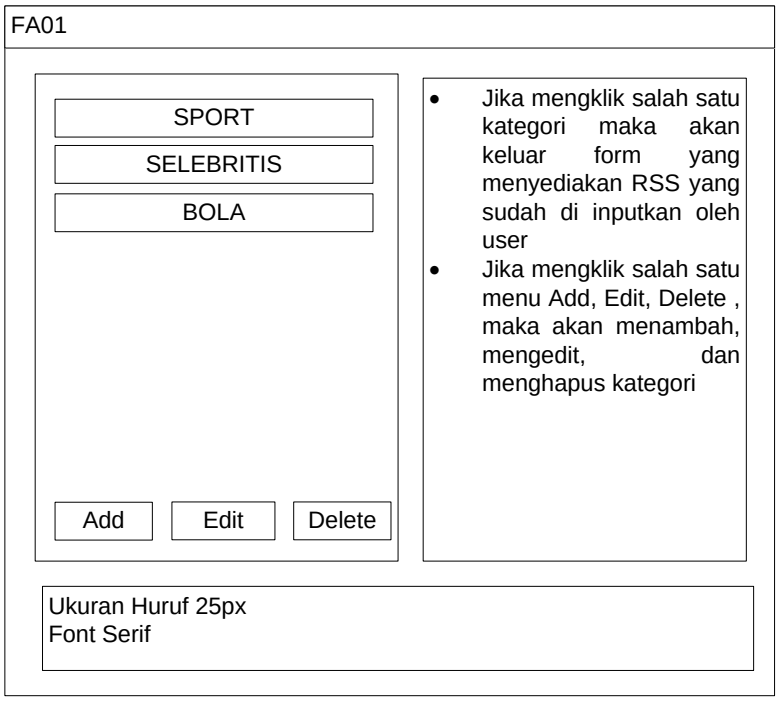
Perancangan antarmuka (*interface*) menggambarkan bagaimana perangkat lunak berkomunikasi dengan dirinya sendiri, dengan sistem yang berinteroperasi dengannnya, dan dengan manusia yang menggunakannya.

3.2.4.1. Perancangan Tampilan

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

1. Perancangan form Utama

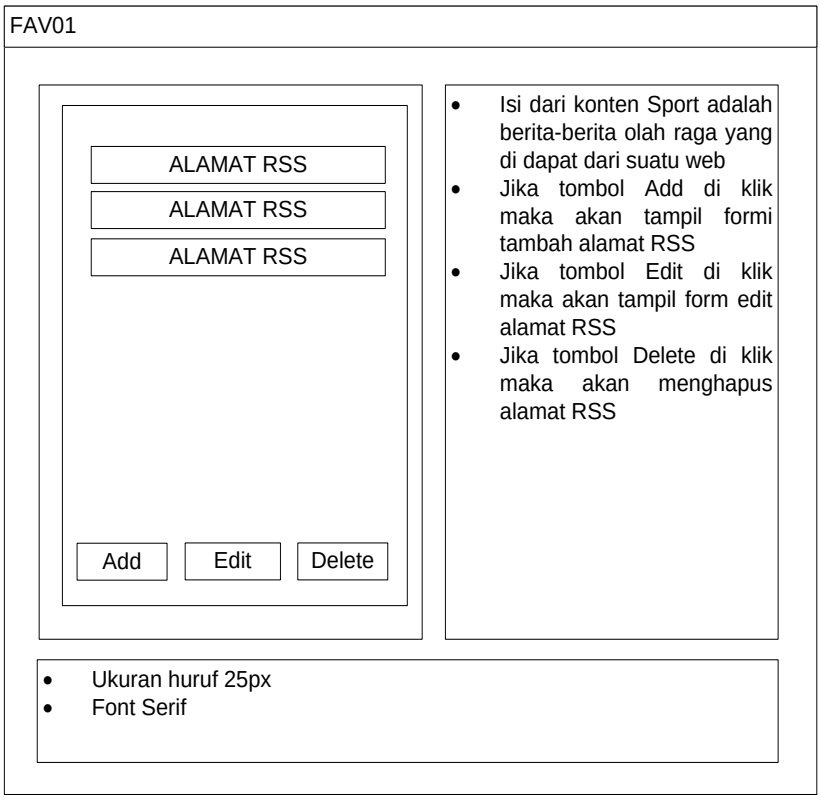
Gambaran umum dari perancangan antarmuka form utama pada aplikasi *mobile Content Anggregator* dapat dilihat pada Gambar berikut ini



Gambar 3. 22 Tampilan Utama

2. Perancangan Form View RSS

Gambaran umum dari perancangan antarmuka form *View* RSS pada aplikasi mobile content Anggregator dapat dilihat pada Gambar berikut ini



Gambar 3. 23 Tampilan Form *View* RSS

3. Perancangan Form Pilih Judul Berita

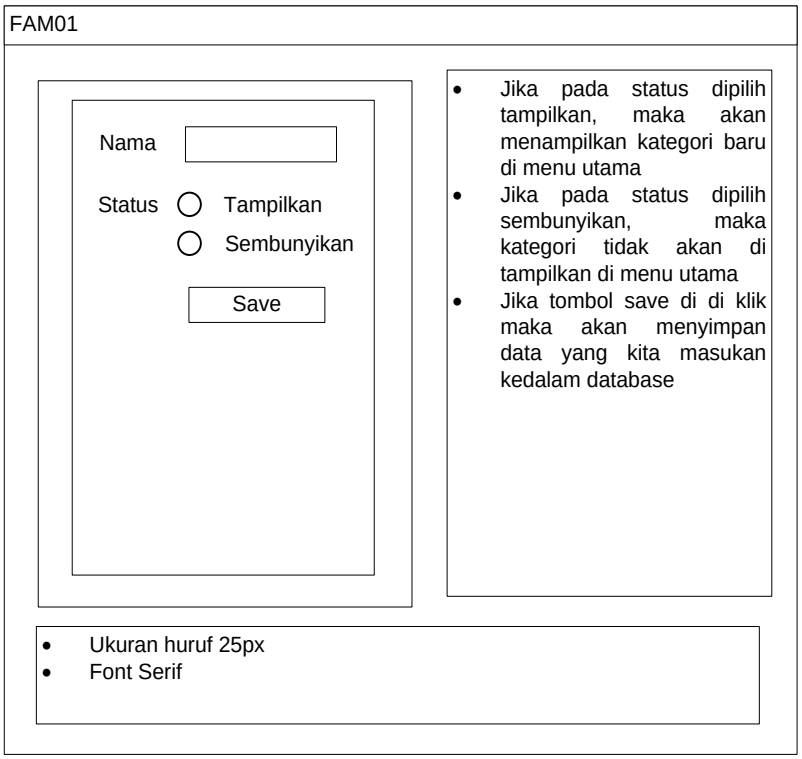
Gambaran umum dari perancangan antarmuka form Pilih Judul Berita pada aplikasi *mobile* content Anggregator dapat dilihat pada Gambar berikut ini



Gambar 3. 24 Tampilan Form Pilih Judul Berita

4. Perancangan Form *Add* Kategori

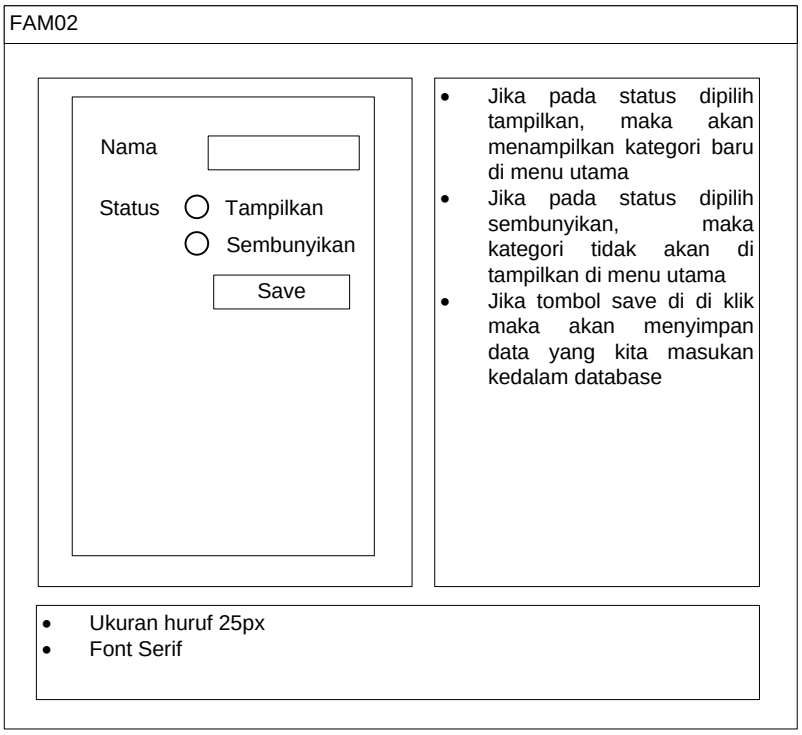
Gambaran umum dari perancangan antarmuka form *Add* Kategori pada aplikasi mobile content Anggregator dapat dilihat pada Gambar berikut ini



Gambar 3. 25 Tampilan form Add Kategori

5. Perancangan Form *Edit* Kategori

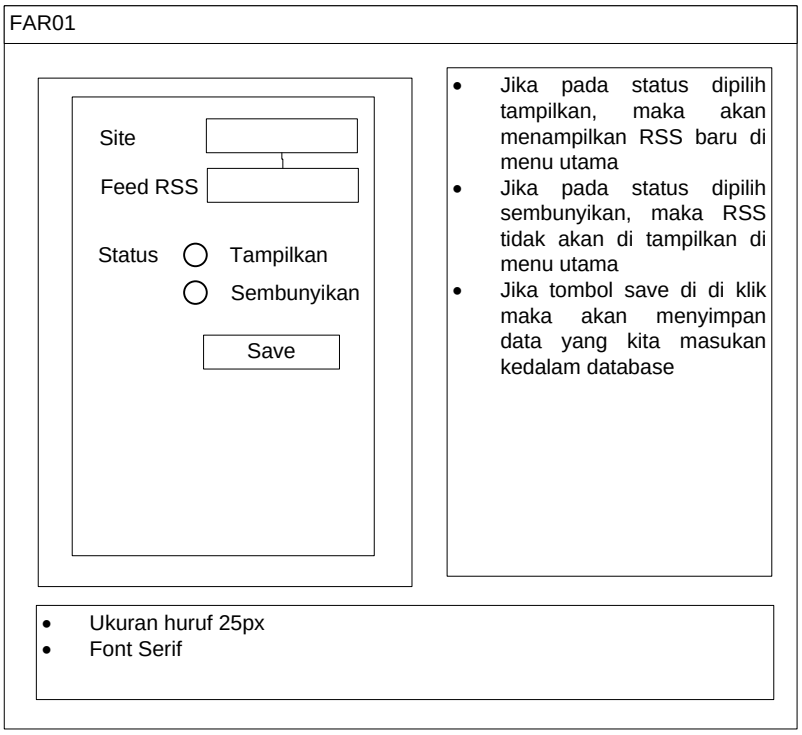
Gambaran umum dari perancangan antarmuka form *Edit* Kategori pada aplikasi mobile content Anggregator dapat dilihat pada Gambar berikut ini



Gambar 3. 26 Tampilan Form *Edit* Kategori

6. Perancangan Form *Add* RSS

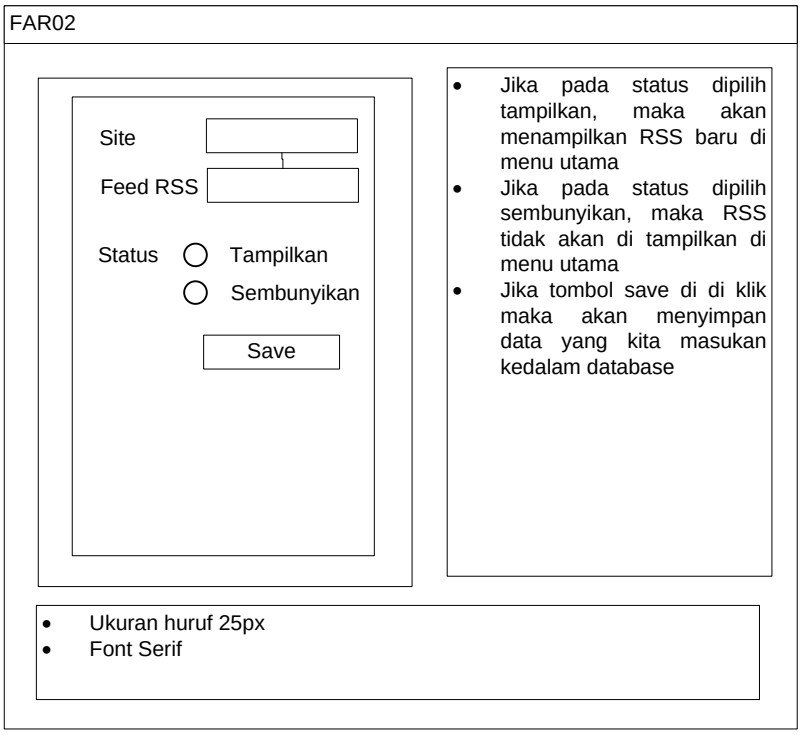
Gambaran umum dari perancangan antarmuka form *Add* RSS pada aplikasi mobile content Anggregator dapat dilihat pada Gambar berikut ini



Gambar 3. 27 Tampilan Form *Add* RSS

7. Perancangan Form *Edit* RSS

Gambaran umum dari perancangan antarmuka form *Edit* RSS pada aplikasi mobile content Anggregator dapat dilihat pada Gambar berikut ini



Gambar 3. 28 Tampilan Form *Edit* RSS