BAB III

PEMBAHASAN

3.1 Analisis Sistem

Analisis sistem merupakan penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan, dengan berlandaskan pada sistem yang telah berjalan.

Dalam proses analisis akan dibahas mengenai analisis yang sedang berjalan, analisis yang akan dikembangkan, analisis kebutuhan non fungsional, analisis kebutuhan fungsional, dan analisis web analytics.

3.1.1 Analisis Sistem Yang Berjalan

Prosedur adalah serangkaian aksi yang spesifik, tindakan atau operasi yang harus dijalankan atau dieksekusi dengan cara yang sama agar selalu memperoleh hasil yang sama dari keadaan yang sama. Prosedur mengindikasikan rangkaian aktivitas, tugas-tugas, langkah-langkah, keputusan-keputusan, perhitungan-perhitungan dan proses-proses, yang dijalankan melalui serangkaian pekerjaan yang menghasilkan suatu tujuan yang diinginkan, suatu produk atau sebuah akibat.

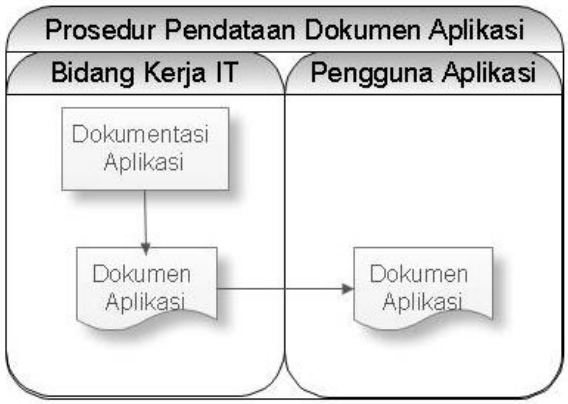
Berdasarkan hasil wawancara dan observasi yang dilakukan kepada pihak PLN prosedur dalam pendataan aplikasi dapat diklasifikasikan antara lain : prosedur pendataan aplikasi, prosedur pendataan dokumen aplikasi.

3.1.1.1 Prosedur Pendataan Aplikasi

Prosedur yang dilakukan untuk pendataan aplikasi adalah sebagai berikut :

1. Setiap Aplikasi baru yang telah dibuat atau dikembangkan bagian bidang kerja IT mencatat nama, deskripsi, versi, pembangun, pemilik dan bidang kerja yang akan menggunakan aplikasi ke dalam list.
2. Aplikasi yang baru dibuat akan di sosialisasikan cara penggunaan kepada bidang kerja yang akan menggunakan aplikasi tersebut.

Berikut ini adalah Flowmap nya :



Gambar 3.1.1.1 Prosedur Pendataan Aplikasi

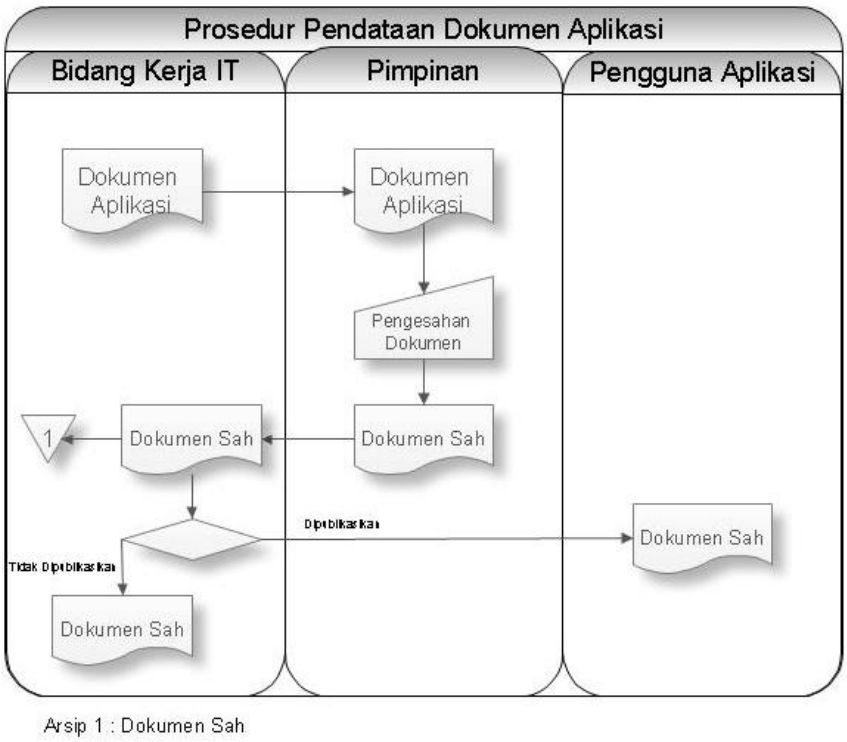
3.1.1.2 Prosedur Pendataan dokumen aplikasi

Prosedur yang dilakukan untuk pendataan dokumen aplikasi adalah sebagai berikut :

1. Setiap dokumen Aplikasi baru yang telah dibuat atau dikembangkan dan disetujui oleh pimpinan PLN akan di arsipkan oleh bidang IT.
2. Bila dokumen tertentu yang di anggap bisa dipublikasikan ke pada bidang kerja lain, misalnya : dokumen cara penggunaan, dokumen pengesahan aplikasi, atau dokumen

catatan lainnya. Bidang kerja lain bisa meminta duplikasi baik berupa softcopy atau hardcopy kepada bidang IT.

Berikut adalah Flowmap nya



Gambar 3.1.1.2 Prosedur Pendataan Dokumen Aplikasi

3.1.2 Perancangan

Berdasarkan analisis yang telah dilakukan di prosedur yang berjalan, dibutuhkan pengembangan dan tambahan prosedur untuk melengkapi prosedur yang telah ada. Prosedur tersebut antara lain :

3.1.2.1 Login

Proses login merupakan proses validasi pengunjung untuk dapat menggunakan hak aksesnya. Proses login dibutuhkan untuk menjaga keamanan dan kenyamanan pada saat melakukan kunjungan.

Pengunjung diharuskan untuk melakukan proses login pada saat pertama kali pengaksesan. Pengunjung dapat mencantumkan username dan password yang telah di daftarkan sebelumnya.

3.1.2.2 Pengolahan Data Pengguna dan Admin

Untuk mengoprasikan aplikasi dibutuhkan Admin yang bertugas sebagai pengolah data mulai dari data detail aplikasi, gambar, data dokumen aplikasi, kategori aplikasi dan data pengguna yang bisa mengakses aplikasi ini.

Admin bisa menambah, mengubah dan menghapus pengguna. Untuk bisa menambah dan mengubah data pengguna Admin harus mencantumkan username, password dan level hak akses user.

3.1.1.1. Pengolahan Data Kategori Aplikasi

Pengkategorian adalah salah satu hal yang paling penting, dengan adanya pengkategorian yang baik data aplikasi dapat dikelompokan sesuai ciri-ciri tertentu, sehingga dapat mempermudah pengunjung untuk melakukan penelusuran daftar aplikasi yang telah

ditambahkan. Setiap data aplikasi bisa memiliki lebih dari satu kategori dan setiap kategori bisa mempunyai parent kategori atau bisa disebut subkategori.

Dalam hal ini admin yang mempunyai akses untuk menambah, mengubah, dan menghapus kategori. Untuk bisa menambah dan mengubah data kategori admin harus mencantumkan nama kategori, dan jika kategori tersebut mempunyai parent admin harus memilih satu perent kategori dari kategori lainnya.

3.1.2.3 Pengolahan Data Aplikasi

Data detail aplikasi hanya bisa ditambah, diubah dan dihapus oleh admin. Prosedur untuk menambah dan mengubah data detail aplikasi adalah sebagai berikut :

1. Admin mengisi nama, detail, versi, tanggal rilis, pembangun aplikasi, dan pemilik aplikasi
2. Admin dapat memilih kategori aplikasi, jika kategori tidak dipilih maka data aplikasi diasumsikan memiliki kategori : “lain-lain”.
3. Setelah form telah diisi dengan benar tekan tombol simpan, jika validasi gagal akan muncul pesan kesalahan.

3.1.2.4 Pengolahan Data Galeri Aplikasi

Galeri merupakan fasilitas untuk memperjelas detail aplikasi itu sendiri. Dalam pengolahan data galeri aplikasi terdapat beberapa syarat yang harus dipenuhi, yaitu :

1. Pengguna yang bisa menambahkan atau menghapus gambar galeri adalah admin.
2. Detail aplikasi telah berhasil ditambahkan atau telah ada pada database.
3. File gambar yang akan ditambahkan memiliki extensi *.jpg, *.jpeg atau *.png.
4. File gambar yang akan ditambahkan memiliki ukuran file maksimal 2 MB.

Prosedur yang dilakukan untuk menambahkan gambar galeri aplikasi adalah sebagai berikut :

1. Admin membuka daftar list aplikasi dan menekan tombol olah galeri
2. Admin bisa menginputkan keterangan (alt) gambar.
3. Admin memilih lokasi gambar pada perangkat client.
4. Setelah semua telah selesai, tekan tombol simpan. Jika validasi gagal akan muncul pesan kesalahan.

3.1.2.5 Pengolahan Data Dokumen Aplikasi

Dalam pengolahan data dokumen aplikasi terdapat beberapa syarat yang harus dipenuhi, yaitu:

1. Pengguna yang bisa menambahkan atau menghapus dokumen adalah admin.
2. Detail aplikasi telah berhasil ditambahkan atau telah ada pada database.
3. File dokumen yang akan ditambahkan memiliki extensi *.pdf, *.doc, *.docx, *.xlsx, *.txt, *.rar atau *.zip.
4. File dokumen yang akan ditambahkan memiliki ukuran file maksimal 20 MB.

Prosedur yang dilakukan untuk menambahkan gambar galeri aplikasi adalah sebagai berikut :

1. Admin membuka daftar list aplikasi dan menekan tombol olah dokumen.
2. Admin menginputkan nama dokumen.
3. Admin memilih lokasi dokumen pada perangkat client.
4. Admin menginputkan keterangan dokumen.
5. Setelah semua telah selesai, tekan tombol simpan. Jika validasi gagal akan muncul pesan kesalahan.

3.1.3 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional adalah analisis yang dilakukan untuk mengetahui bagaimana keadaan sistem yang ada di PLN, yang meliputi komponen pendukung, sehingga akan diusulkan kebutuhan untuk menjalankan sistem yang akan dibangun sampai dengan sistem diimplementasikan, sehingga apa yang dibangun dapat berjalan optimal. Analsisis kebutuhan non fungsional meliputi analisis perangkat keras, analisis perangkat lunak, dan analisis pengguna.

Analsis kebutuhan non fungsional meliputi analisis perangkat keras, analisis perangkat lunak, dan analisis pengguna.

3.1.3.1 Analisis Perangkat Keras

Perangkat keras minimum yang dibutuhkan untuk menjalankan aplikasi yang akan dibangun adalah sebagai berikut:

1. Prosessor : Intel Pentium 4 2.0GHz
2. Monitor : Monitor VGA 15”
3. Memori : DDR2 256GB
4. Harddisk : 20 GB
5. Keyboard
6. Mouse
7. Terkoneksi Jaringan

3.1.3.2 Analisis Perangkat Lunak

Untuk membangun aplikasi ini dibutuhkan spesifikasi perangkat lunak sebagai berikut:

1. XAMPP, yang mencakup web server Apache, DBMS MySQL, dan bahasa pemrograman PHP.
2. Dreamweaver, sebagai aplikasi pembangun aplikasi.

3. Web browser, Mozilla Firefox, Google Chrome.

Untuk menjalankan aplikasi ini dibutuhkan spesifikasi minimal perangkat lunak yang setara dengan:

1. Sistem operasi Windows XP.
2. Browser Google Chrome, Mozilla Firefox.

3.1.3.3 Analisis Pengguna

Suatu sistem akan berjalan optimal bukan hanya karena teknologinya, namun juga harus diperhatikan sumber daya manusia yang menggunakan. Pengguna aplikasi ini akan dibagi menjadi dua bagian menyesuaikan dengan apa yang dibutuhkan, berdasarkan analisis yang telah dilakukan untuk pembagian pengguna aplikasi yang akan dikembangkan dapat dilihat pada tabel 3.1.3.1.

Pengguna	**Tanggung Jawab**	**Hak Akses**	**Tingkat Keterampilan**	**Pelatihan yang Dibutuhkan**
Admin	Mengolah data pengguna, mengolah data daftar aplikasi, data kategori aplikasi, dokumen aplikasi dan galeri aplikasi.	Mengakses dan mengolah data yang berhubungan dengan informasi web yang akan di tampilkan, selain itu mengolah data user.	Tidak asing dengan dunia web dan internet, berpengalaman menggunakan sistem operasi windows	Pengenalan terhadap aplikasi yang dibangun.
Pengunjung	Mengakses fitur umum aplikasi	Mengakses fitur umum aplikasi	mengenal internet dan mampu menggunakan	-

Tabel 3.1.3.1 Pembagian Pengguna

3.1.4 Analisis Kebutuhan Fungsional

Analisis fungsional merupakan tindakan mengidentifikasi proses yang akan berjalan didalam sistem, yang akan dikembangkan dan menjelaskan kebutuhan yang diperlukan. Sehingga sistem dapat berjalan seperti yang diharapkan.

Analisis fungsional dimodelkan dengan menggunakan UML (Unified Modeling Language). Tahapan pemodelan dalam analisis tersebut antara lain mengidentifikasi aktor, pembuatan diagram use case, scenario use case, diagram sequence, diagram class, diagram package dan diagram deployment.

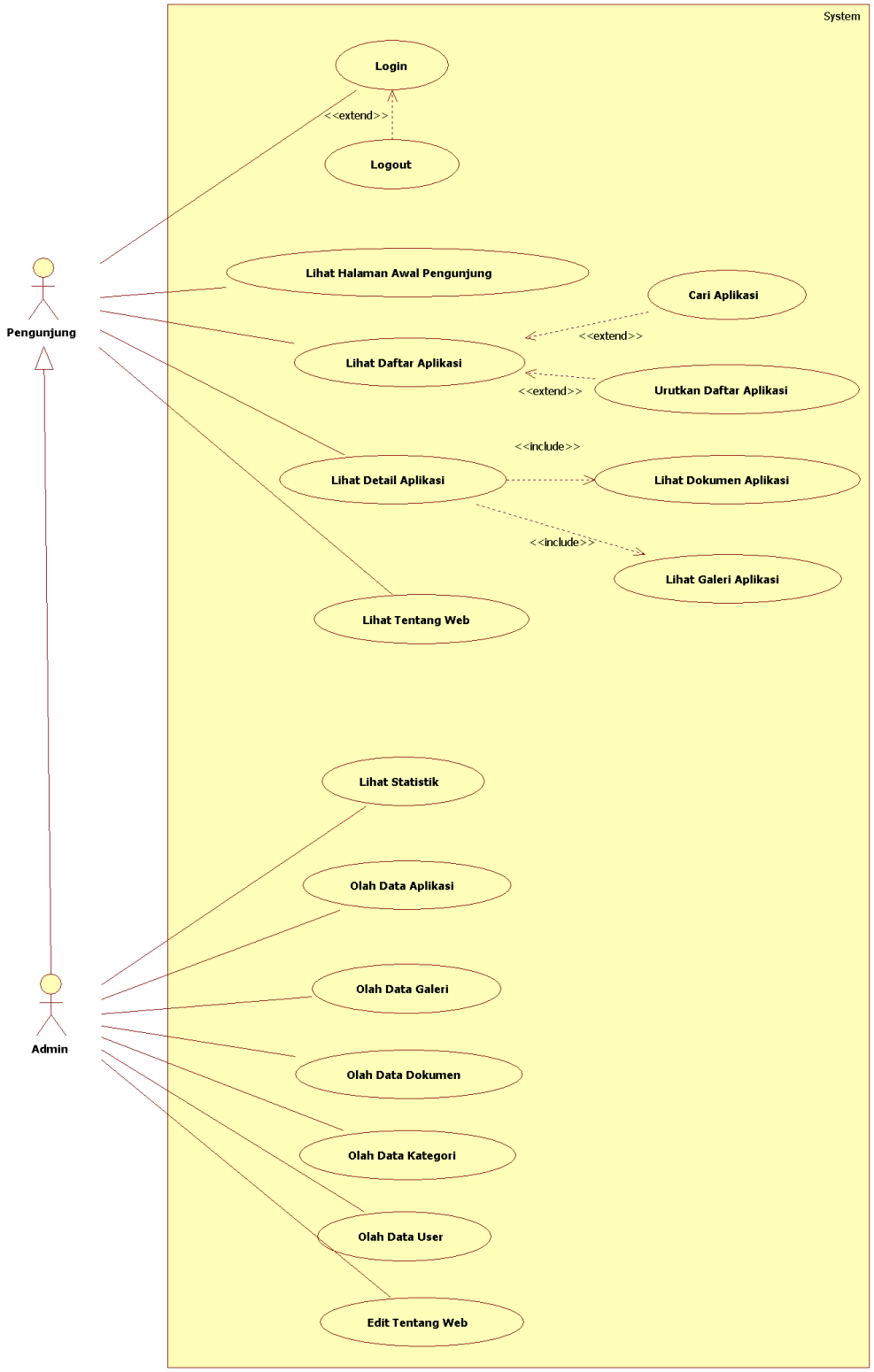
3.1.4.1 Identifikasi Aktor

Aktor yang teridentifikasi dan terlibat dalam sistem yaitu Admin dan pengunjung.

3.1.4.2 Diagram Use Case

Use Case adalah interaksi yang terjadi antara sistem dan aktor yang terlibat, didalamnya termasuk pertukaran pesan dan tindakan yang dilakukan oleh sistem. Use Case diagram menggambarkan setiap aktivitas yang dilakukan oleh sistem dari sudut pengamatan tertentu.

Gambar 3.1.4.1 adalah diagram use case untuk Web Profil Aplikasi Corporate di PT PLN.



Gambar 3.1.4.1 Diagram use case

3.1.4.3 Skenario Use Case

Skenario use case merupakan skenario (flow of event) interaksi aktor yang terlibat dengan beberapa usecase utama. Skenario usecase menggambarkan urutan interaksi actor dengan use case tersebut, dari awal sampai akhir. Berikut adalah skenario usecase utama untuk Web Profil Aplikasi Corporate berdasarkan diagram usecase pada gambar 3.1.4.1.

1. Skenario Use Case Login

Identifikasi			
Nama		Login	
Tujuan		Validasi hak akses bagi setiap user baik pengunjung ataupun administrator.	
Deskripsi		Validasi atau autentikasi sistem untuk setiap aktor, untuk memperoleh hak khusus sesuai perannya.	
Aktor		Pengunjung, Administrator	
Skenario Utama			
Kondisi Awal		Sistem menampilkan form login yang terdiri dari username dan password	
1.	Mengisi form login	2	Melakukan validasi masukan
		3	Memeriksa inputan ke database user
		4	Bila data di database cocok, maka akan menuju menu utama, tergantung hak akses masing masing aktor.
Skenario Alternatif (login gagal)			
		2.1	menampilkan pesan “Masukan username dan password tidak valid”
		3.1	Menampilkan pesan “login tidak berhasil, username atau password salah”

2. Skenario Use Case Logout

Identifikasi	
Nama	Logout

Tujuan		Untuk menghapus session login user aktif dan menghilangkan hak akses aktor.	
Deskripsi		menghapus session aktif dan menghilangkan hak akses bagi setiap aktor	
Aktor		Pengunjung, Administrator	
Skenario Utama			
Kondisi Awal		User telah login sebelumnya, dan session login masih aktif.	
1.	Menekan tombol logout	2	Menghapus session aktif sesui id session user
		3	Sistem kembali mengalihkan ke halaman login

3. Skenario User Case Lihat Halaman Awal Pengunjung

Identifikasi			
Nama		Lihat Halaman Awal Pengunjung	
Tujuan		Menampilkan halaman pertama setelah user login	
Deskripsi		Sistem menampilkan judul web profil dan empat aplikasi secara acak.	
Aktor		Pengunjung, Administrator	
Skenario Utama			
Kondisi Awal		User telah login	
1.	User berada di halaman utama	2	Setem mengambil empat data aplikasi secara acak
		3	Sistem menampilkan judul web profil dan empat aplikasi sebagai headline.

4. Skenario Use Case Lihat Daftar Aplikasi

Identifikasi	
Nama	Lihat Daftar Aplikasi

Tujuan		Menampilkan daftar aplikasi sesuai kreteria yang di inginkan user	
Deskripsi		Sistem menampilkan daftar aplikasi yang telah pernah di simpan dan user bisa meminta daftar aplikasi menurut kriteria tertentu	
Aktor		Pengunjung, Administrator	
Skenario Utama			
Kondisi Awal		User telah login	
1.	User berada di halaman daftar aplikasi	2	pengecekan parameter (GET) di url
		3	jika tidak terdapat parameter khusus, sistem mengambil semua data aplikasi di database
		4	Sitem menampilkan daftar aplikasi kepada user
Skenario Alternatif			
		3.1	Jika terdapat parameter id kategori, sistem mengambil data berdasarkan id kategori yang cocok di database
		3.2	Jika terdapat parameter query pencarian, sistem mengambil data aplikasi yang memiliki nama yang mendekati query
		3.3	Jika terdapat parameter pengurutan, sistem mengambil data dan mengurutkan data daftar aplikasi

5. Skenario Use Case Lihat Detail Aplikasi

Identifikasi	
Nama	Lihat Detail Aplikasi
Tujuan	Menampilkan detail aplikasi
Deskripsi	Sistem menampilkan detail aplikasi berupa nama , kategori,versi, tanggal rilis, pembangun, pemilik, gallery dan dokumen aplikasi

Aktor		Pengunjung, Administrator	
Skenario Utama			
Kondisi Awal		User telah login	
1.	User berada di halaman detail aplikasi	2	pengecekan parameter (GET) di url
		3	sistem mengambil semua data aplikasi di database
		4	Sitem menampilkan detail aplikasi kepada user
Skenario Alternatif			
		3.1	jika tidak ada data, detail tidak akan ditampilkan dan akan menampilkan pesan .

6. Skenario Use Case Lihat Tentang Web

Identifikasi			
Nama		Lihat Tentang Web	
Tujuan		Menampilkan profile tentang web dan profile perusahaan PLN.	
Deskripsi		Sistem menampilkan profile web dan profile perusahaan.	
Aktor		Pengunjung, Administrator	
Skenario Utama			
Kondisi Awal		User telah login	
1.	User berada di halaman ‘tentang’	2	Sistem mengambil data profile di database.
		3	Sistem menampilkan profile kepada user

7. Skenario Use Case Lihat Statistik

Identifikasi	
Nama	Lihat Statistik

Tujuan		Menunjukan secara detail jumlah user, jumlah aplikasi yang ada, dan statistik pengunjung web	
Deskripsi		Sistem menampilkan jumlah user, jumlah aplikasi yang ada, dan statistik pengunjung web.	
Aktor		Administrator	
Skenario Utama			
Kondisi Awal		User telah login sebagi administrator	
1.	User berada di halaman utama admin	2	Sistem mengambil data statistik database
		3	Sistem menampilkan statistik kepada user

8. Skenario Use Case Olah Data Aplikasi

Identifikasi			
Nama		Olah Data Aplikasi	
Tujuan		Mengolah data aplikasi	
Deskripsi		Sistem mengolah data aplikasi	
Aktor		Administrator	
Skenario Utama (Tambah Aplikasi)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User memasuki tampilan tambah aplikasi	2	sistem mengambil data kategori dari database
		3	sistem menampilkan form tambah aplikasi
4.	User mengisi form dan menekan tombol “simpan”	5	sistem memeriksa dan memvalidasi inputan

		6	jika sudah inputan telah benar, sistem menyimpan ke database.
		7	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan
Skenario Alternatif (Tambah Aplikasi)			
		6.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”
		7.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”
Skenario Utama (Edit Aplikasi)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User memasuki tampilan edit aplikasi	2	sistem mengambil data aplikasi dari database
		3	sistem mengambil data kategori dari database
		4	sistem menampilkan form tambah aplikasi
5	User mengisi form dan menekan tombol “simpan”	6	sistem memeriksa dan memvalidasi inputan
		7	jika sudah inputan telah benar, sistem menyimpan ke database.
		8	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan
Skenario Alternatif (Edit Aplikasi)			
		7.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”

		8.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”
Skenario Utama (Hapus Aplikasi)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User menekan tombol hapus aplikasi	2	sistem menghapus aplikasi
		3	jika berhasil menghapus, sistem menampilkan pesan “data berhasil di hapus”
Skenario Alternatif (Hapus Aplikasi)			
		3.1	sistem tidak berhasil menghapus data, sistem menampilkan pesan “data tidak berhasil di hapus”

9. Skenario Use Case Olah Data Kategori

Identifikasi			
Nama		Olah Data Kategori	
Tujuan		Mengolah data kategori	
Deskripsi		Sistem mengolah data kategori	
Aktor		Administrator	
Skenario Utama (Tambah Kategori)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User memasuki tampilan tambah kategori	2	sistem mengambil data kategori dari database
		3	sistem menampilkan form tambah aplikasi
4.	User mengisi form dan menekan tombol “simpan”	5	sistem memeriksa dan memvalidasi inputan
		6	jika sudah inputan telah benar, sistem menyimpan ke database.

		7	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan”
Skenario Alternatif (Tambah Kategori)			
		6.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”
		7.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”
Skenario Utama (Hapus Kategori)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User menekan tombol hapus kategori	2	sistem menghapus kategori
		3	jika berhasil menghapus, sistem menampilkan pesan “data berhasil di hapus”
Skenario Alternatif (Hapus Kategori)			
		3.1	sistem tidak berhasil menghapus data, sistem menampilkan pesan “data tidak berhasil di hapus”

10. Skenario Use Case Olah Data Dokumen

Identifikasi	
Nama	Olah Data Dokumen
Tujuan	Mengolah data dokumen aplikasi
Deskripsi	Sistem mengolah data dokumen aplikasi
Aktor	Administrator
Skenario Utama (Tambah Dokumen)	
Kondisi Awal	User telah login dan user adalah administrator

1.	User memasuki tampilan tambah dokumen	2	sistem menampilkan form tambah dokumen
3.	User mengisi form dan menekan tombol “tambah dokumen”	4	sistem memeriksa dan memvalidasi inputan
		5	jika sudah inputan telah benar, sistem menyimpan ke database.
		6	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan”
Skenario Alternatif (Tambah Dokumen)			
		5.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”
		6.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”
Skenario Utama (Hapus Dokumen)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User menekan tombol hapus dokumen	2	sistem menghapus dokumen
		3	jika berhasil menghapus, sistem menampilkan pesan “data berhasil di hapus”
Skenario Alternatif (Hapus Kategori)			
		3.1	sistem tidak berhasil menghapus data, sistem menampilkan pesan “data tidak berhasil di hapus”

11. Skenario Use Case Olah Data Galeri

Identifikasi			
Nama		Olah Data galeri	
Tujuan		Mengolah data galeri aplikasi	
Deskripsi		Sistem mengolah data galeri aplikasi	
Aktor		Administrator	
Skenario Utama (Tambah Galeri)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User memasuki tampilan tambah galeri	2	sistem menampilkan form tambah galeri
3.	User mengisi form dan menekan tombol “tambah galeri”	4	sistem memeriksa dan memvalidasi inputan
		5	jika sudah inputan telah benar, sistem menyimpan ke database.
		6	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan”
Skenario Alternatif (Tambah Galeri)			
		5.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”
		6.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”
Skenario Utama (Hapus Galeri)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User menekan tombol hapus galeri	2	sistem menghapus galeri
		3	jika berhasil menghapus, sistem menampilkan pesan “data berhasil di hapus”

Skenario Alternatif (Hapus Galeri)			
		3.1	sistem tidak berhasil menghapus data, sistem menampilkan pesan “data tidak berhasil di hapus”

12. Skenario Use Case Olah Data User

Identifikasi			
Nama		Olah Data User	
Tujuan		Mengolah data user	
Deskripsi		Sistem mengolah data user	
Aktor		Administrator	
Skenario Utama (Tambah User)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User memasuki tampilan tambah user	2	sistem menampilkan form tambah aplikasi
3.	User mengisi form dan menekan tombol “simpan”	4	sistem memeriksa dan memvalidasi inputan
		5	jika sudah inputan telah benar, sistem menyimpan ke database.
		6	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan”
Skenario Alternatif (Tambah User)			
		5.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”
		6.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”
Skenario Utama (Edit User)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User memasuki tampilan edit user	2	sistem mengambil data user dari database
		3	sistem menampilkan form tambah kategori

4	User mengisi form dan menekan tombol “simpan”	5	sistem memeriksa dan memvalidasi inputan
		6	jika sudah inputan telah benar, sistem menyimpan ke database.
		7	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan
Skenario Alternatif (Edit user)			
		6.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”
		7.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”
Skenario Utama (Hapus User)			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User menekan tombol hapus user	2	sistem menghapus user
		3	jika berhasil menghapus, sistem menampilkan pesan “data berhasil di hapus”
Skenario Alternatif (Hapus User)			
		3.1	sistem tidak berhasil menghapus data, sistem menampilkan pesan “data tidak berhasil di hapus”

13. Skenario Use Case Edit Tentang Web

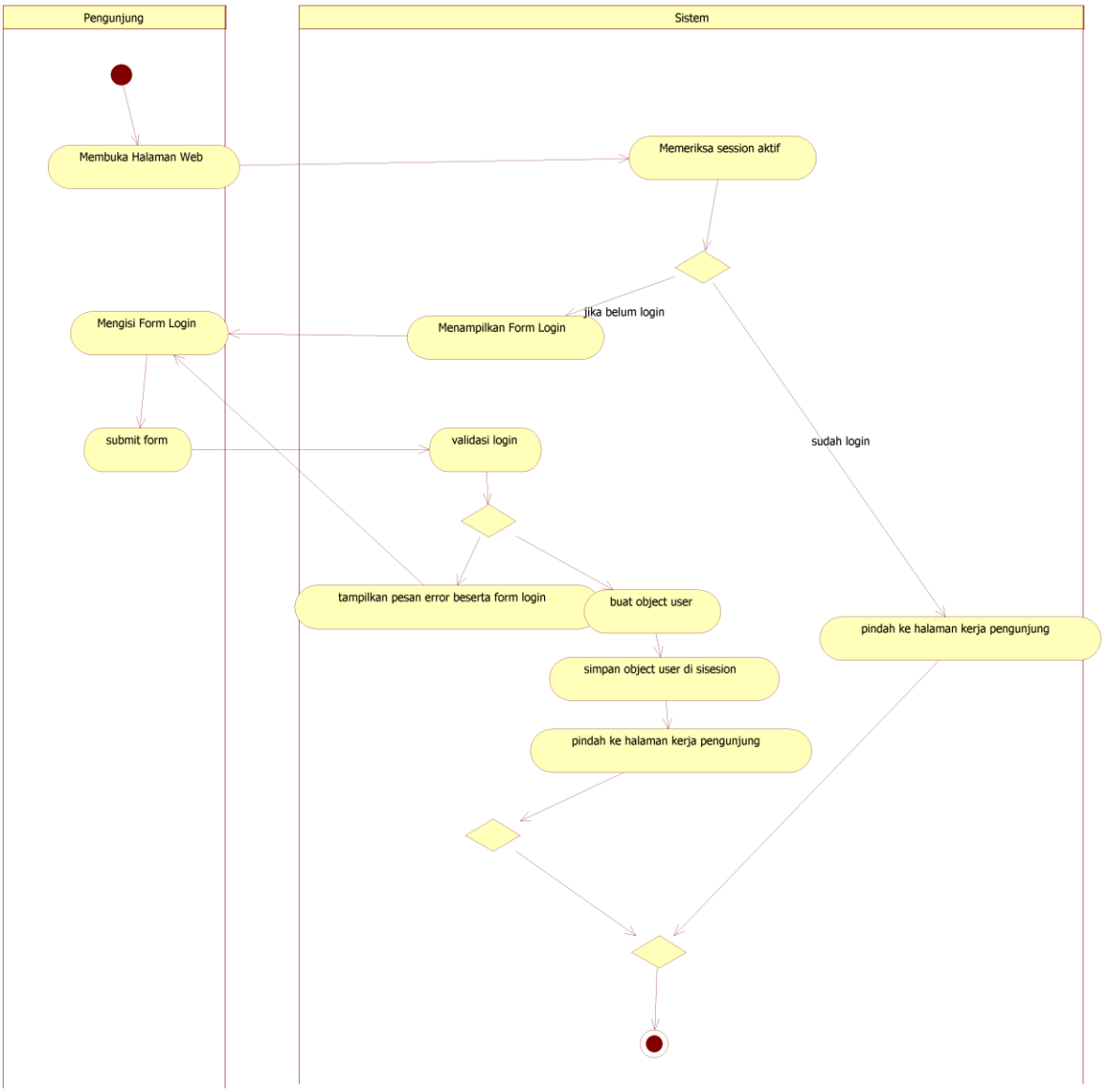
Identifikasi	
Nama	Edit Tentang Web
Tujuan	Mengedit profile web

Deskripsi		Sistem mengolah data profile web	
Aktor		Administrator	
Skenario Utama			
Kondisi Awal		User telah login dan user adalah administrator	
1.	User memasuki tampilan edit tentang web	2	Sitem mengambil data profile web dari database
		3	sistem menampilkan form edit tentang web
4.	User mengisi form dan menekan tombol “simpan”	5	sistem memeriksa dan memvalidasi inputan
		6	jika sudah inputan telah benar, sistem menyimpan ke database.
		7	jika berhasil menyimpan sistem menampilkan pesan “data berhasil di simpan”
Skenario Alternatif			
		6.1	jika inputan tidak valid, sistem menampilkan pesan “inputan tidak valid”
		7.1	sistem tidak berhasil menyimpan data, sistem menampilkan pesan “data tidak berhasil di simpan”

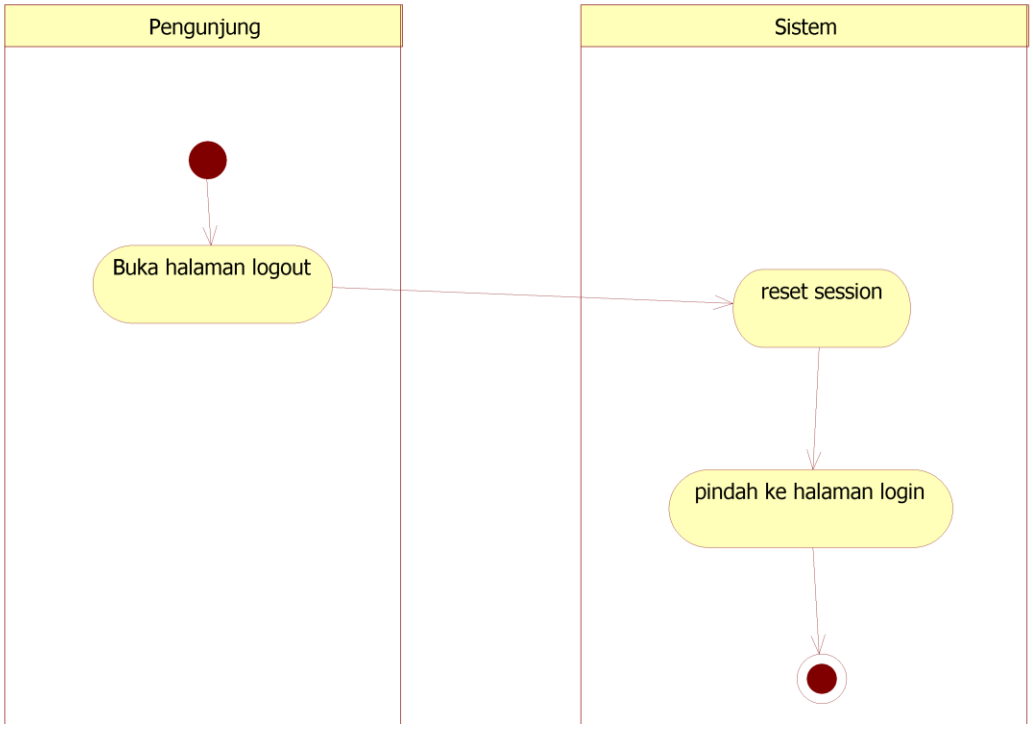
3.1.4.4 Activity Diagram

Activity diagram merupakan teknik untuk menggunakan logika prosedural, proses bisnis, dan jalur kerja. Dari skenario yang dibuat, berikut akan digambarkan *activity diagram*nya.

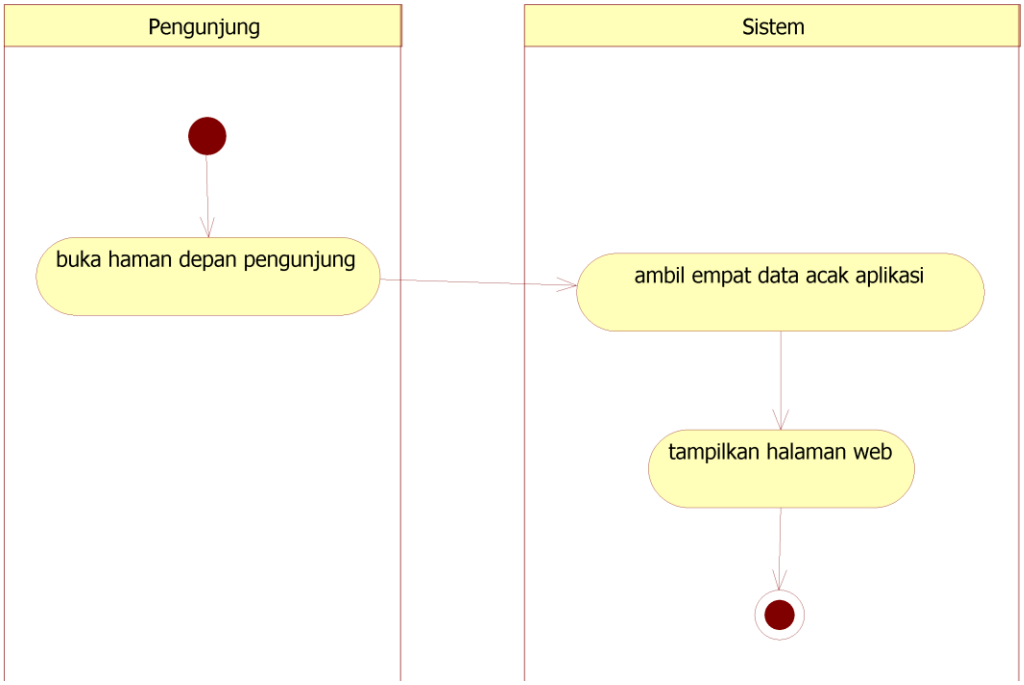
1. *Activity Diagram* Login



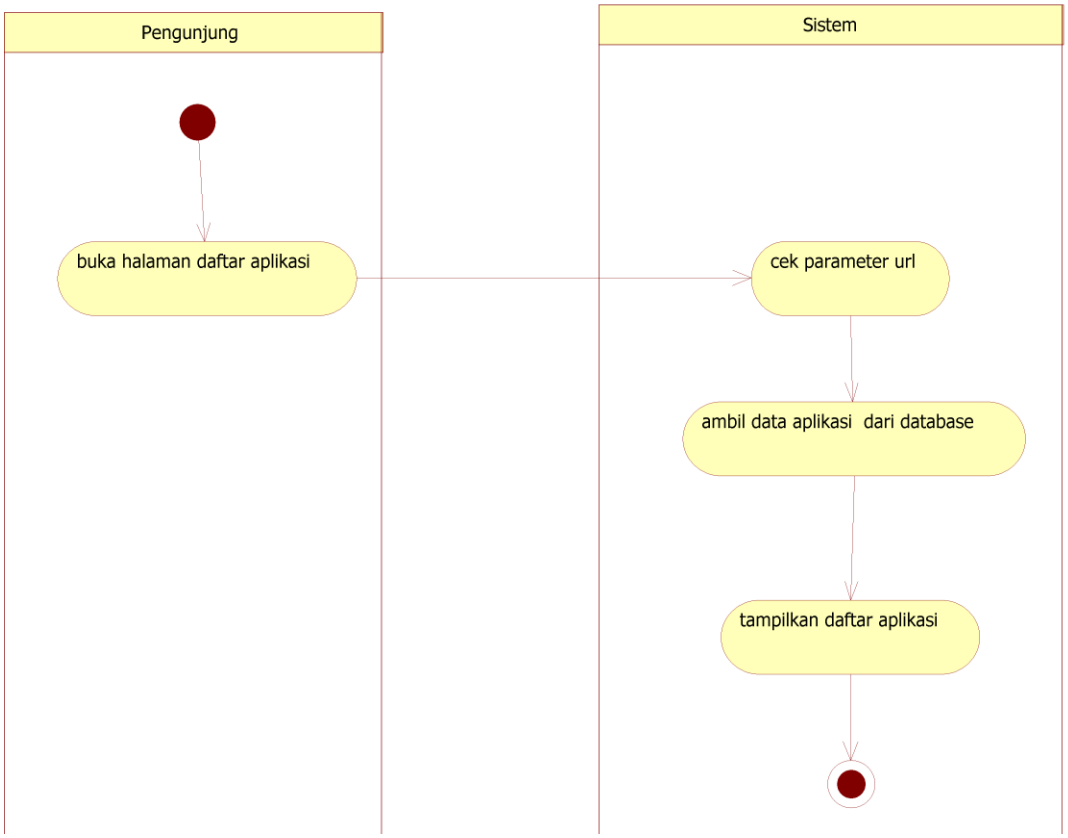
2. *Activity Diagram* Logout



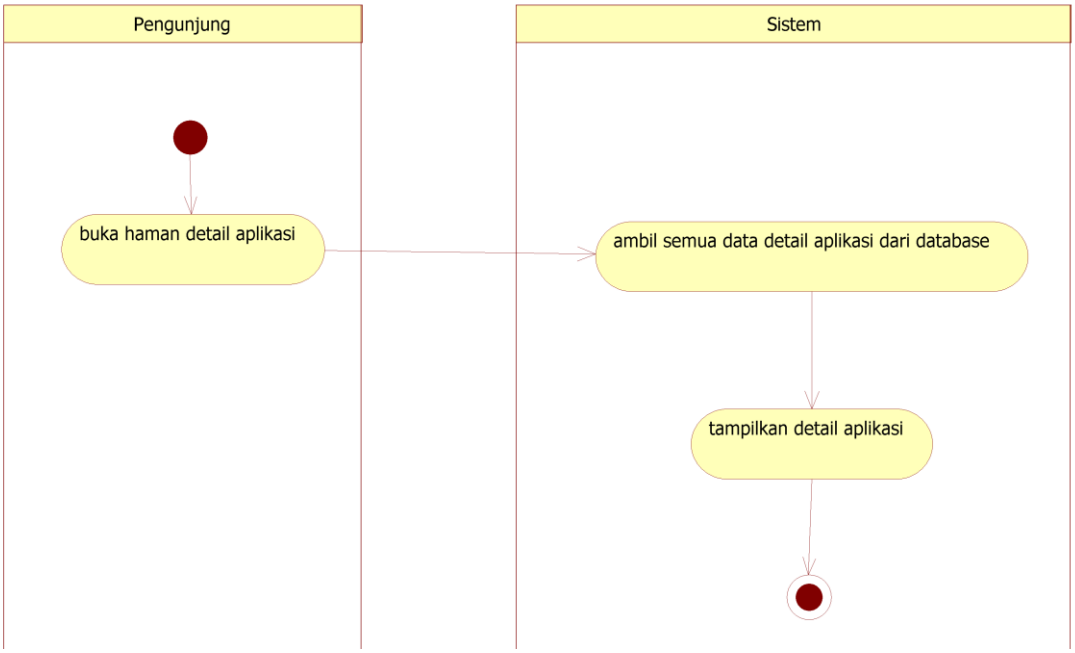
3. *Activity Diagram* Lihat Halaman Awal Pengunjung



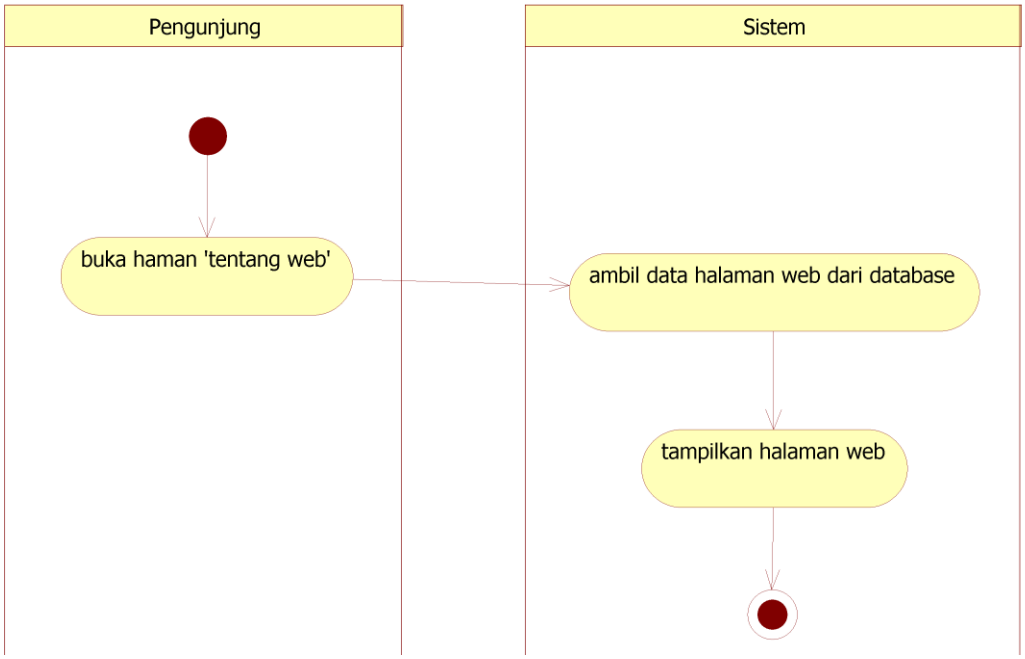
4. *Activity Diagram* Lihat Daftar Aplikasi



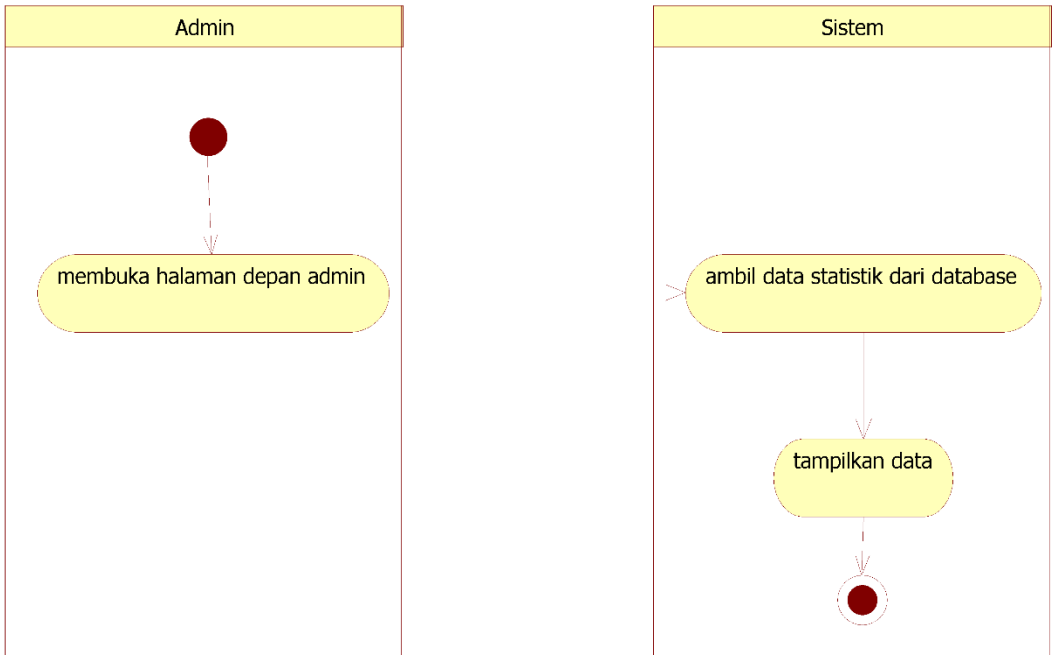
5. *Activity Diagram* Lihat Detail Aplikasi



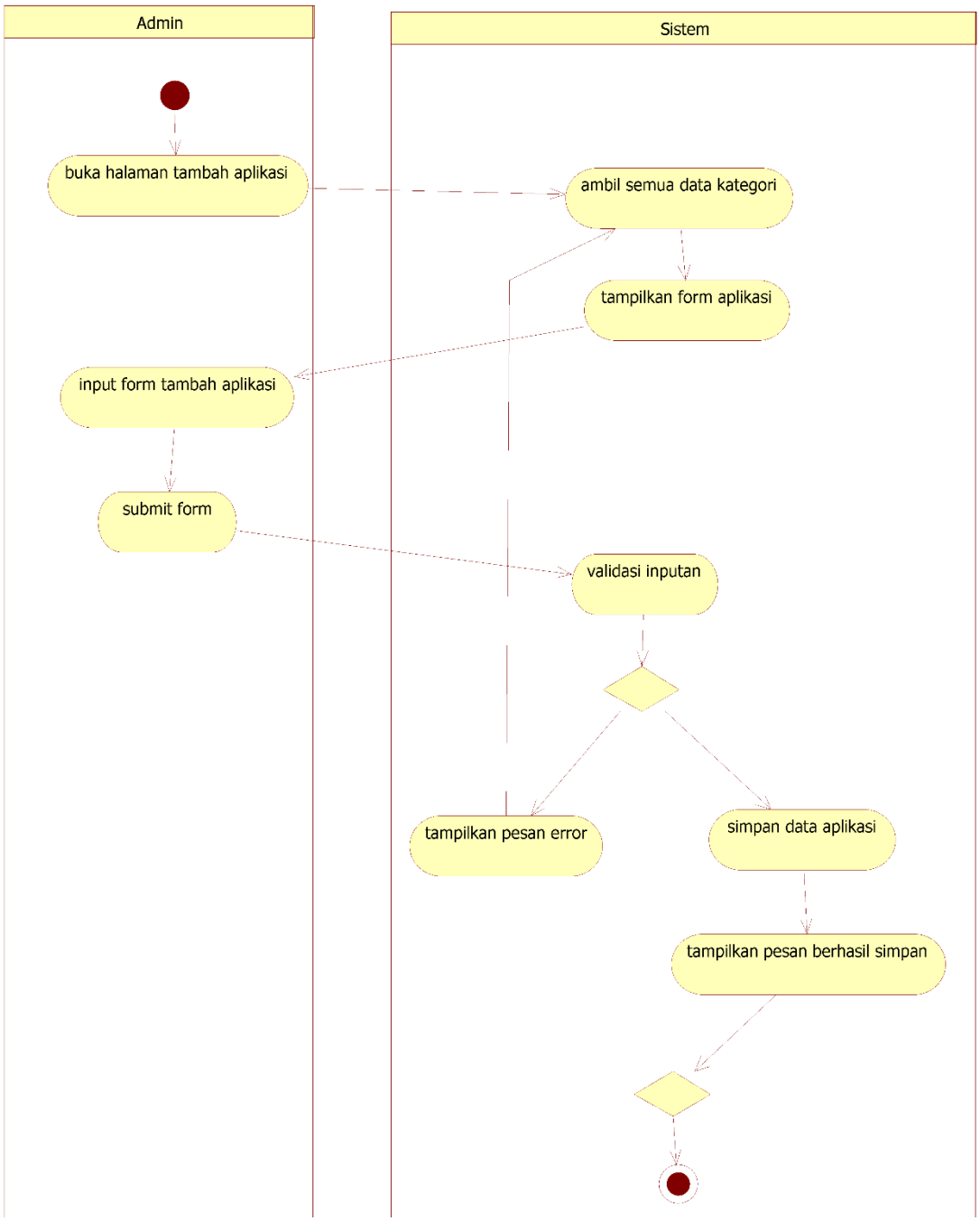
6. *Activity Diagram* Lihat Tentang Web



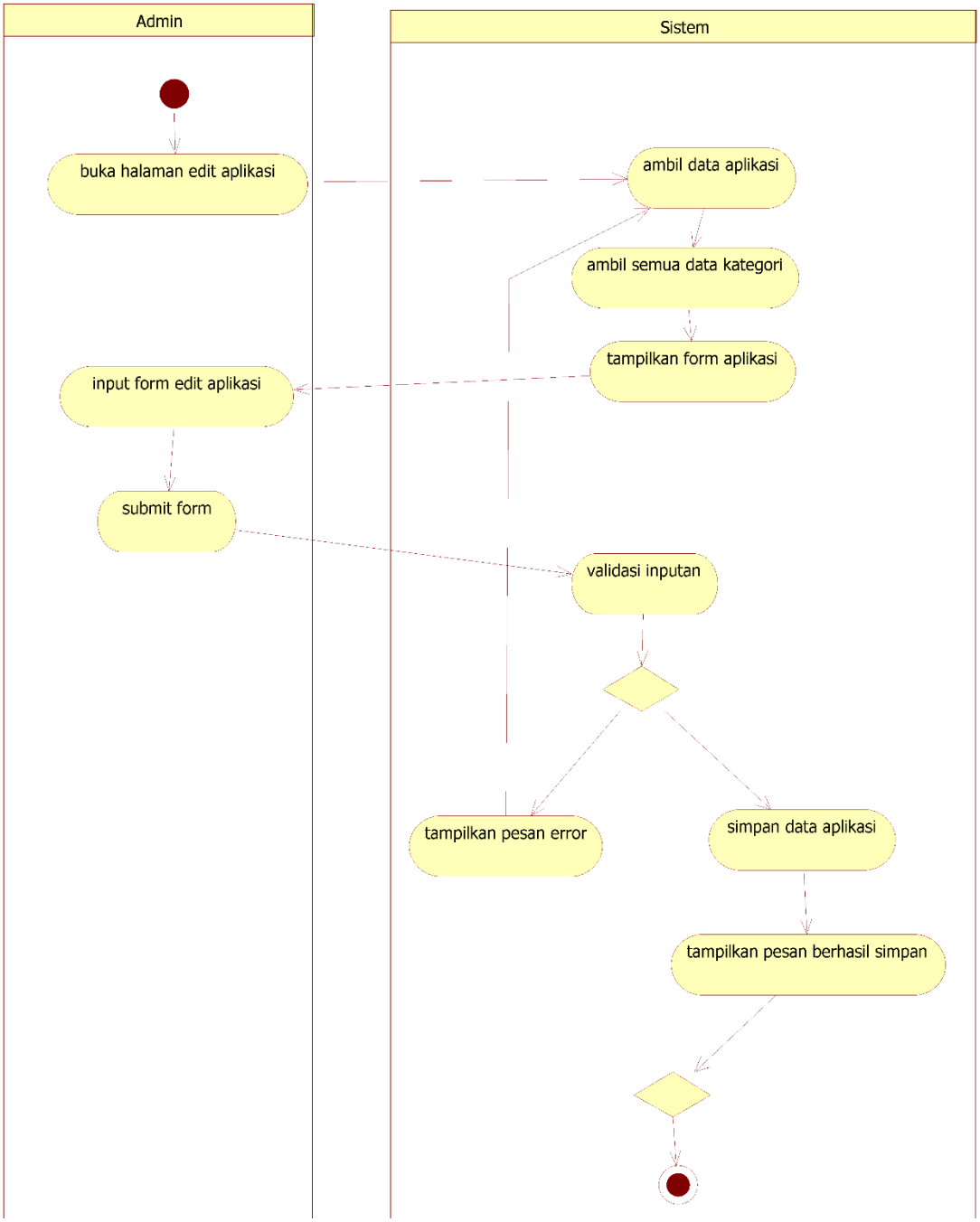
7. *Activity Diagram* Lihat Statistik



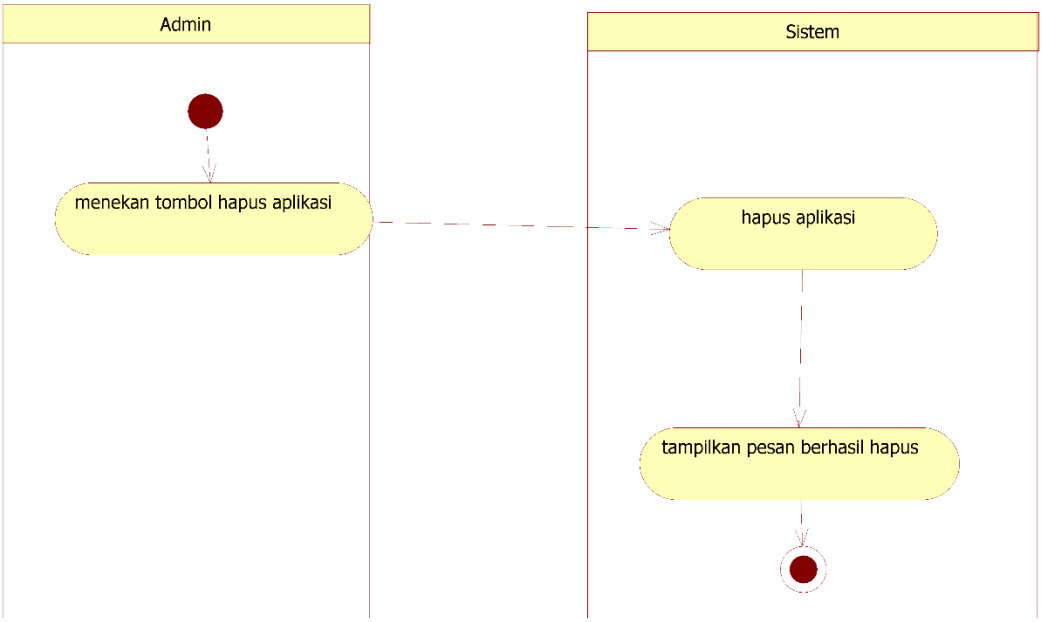
8. *Activity Diagram* Tambah Data Aplikasi



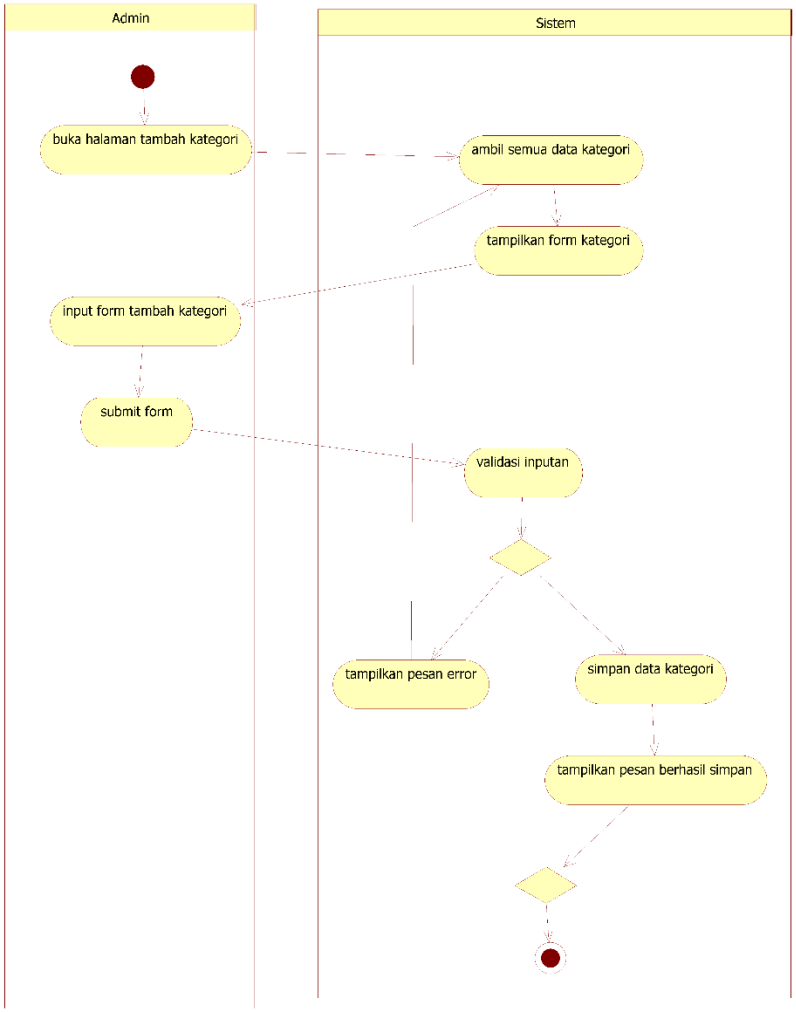
9. *Activity Diagram* Edit Data Aplikasi



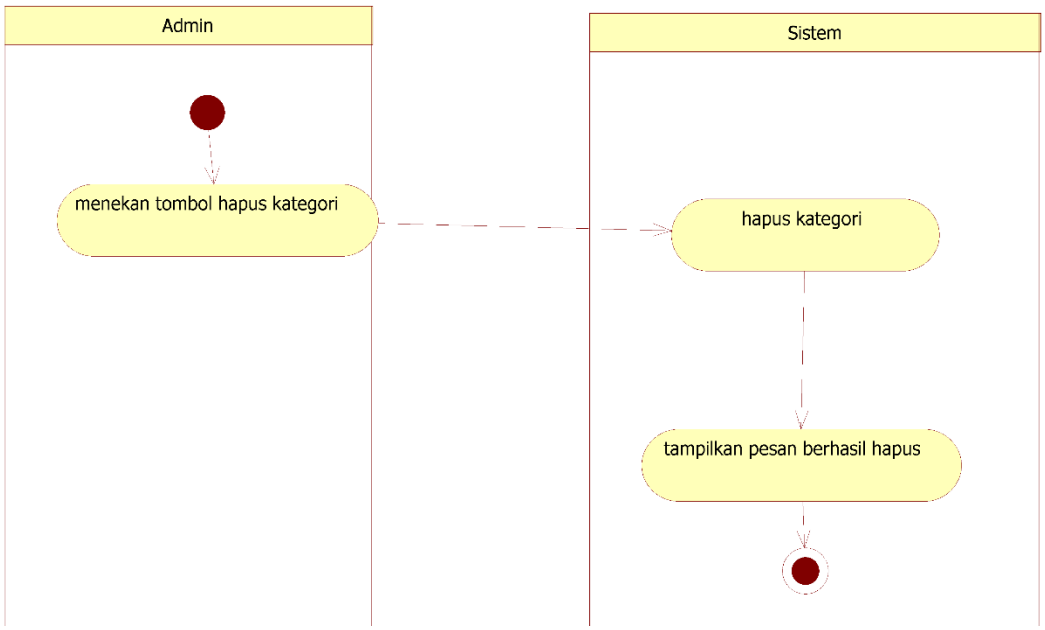
10. *Activity Diagram* Hapus Data Aplikasi



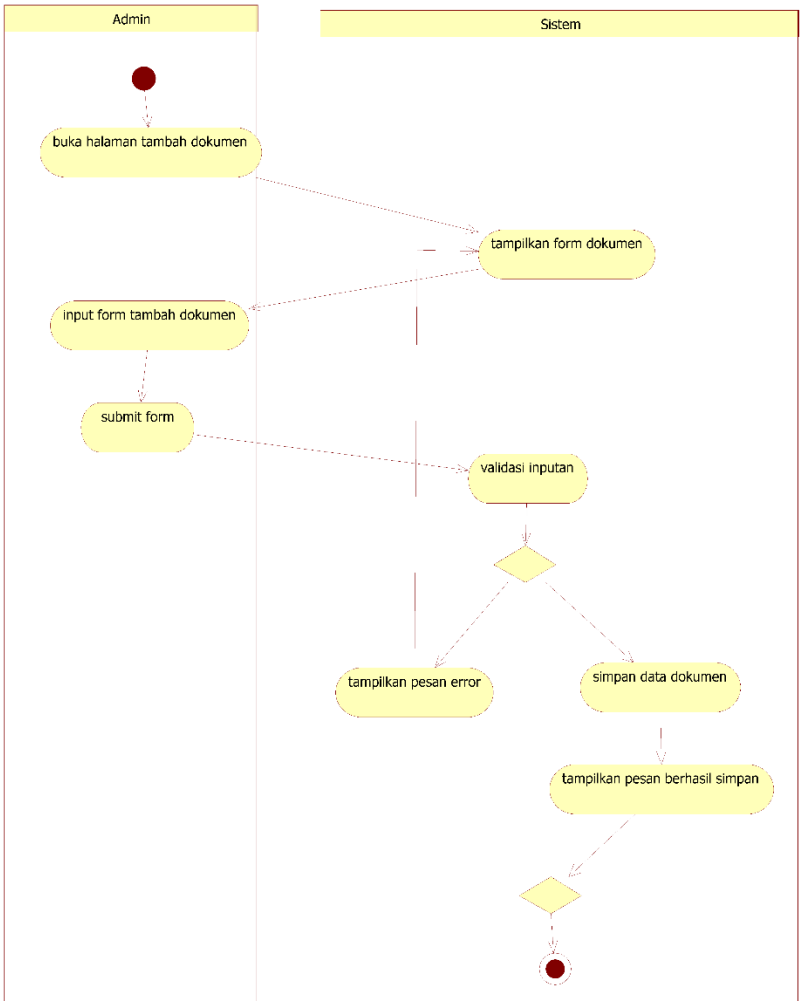
11. *Activity Diagram* Tambah Data Kategori



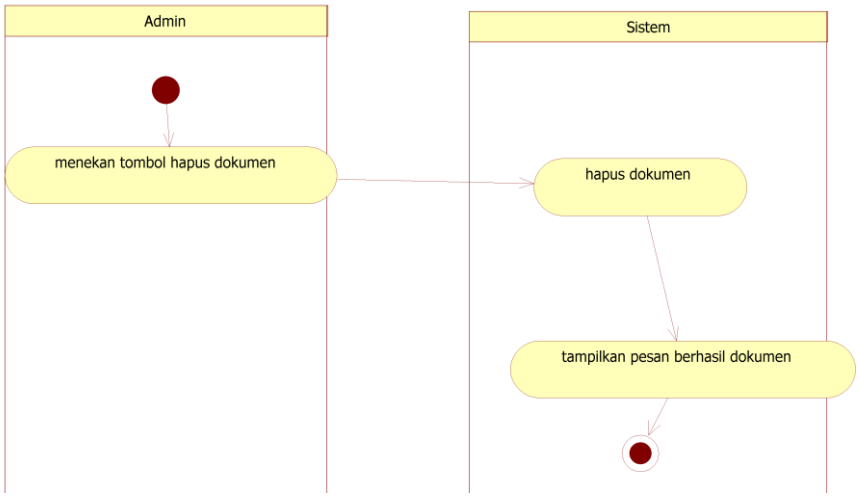
12. *Activity Diagram* Hapus Data Kategori



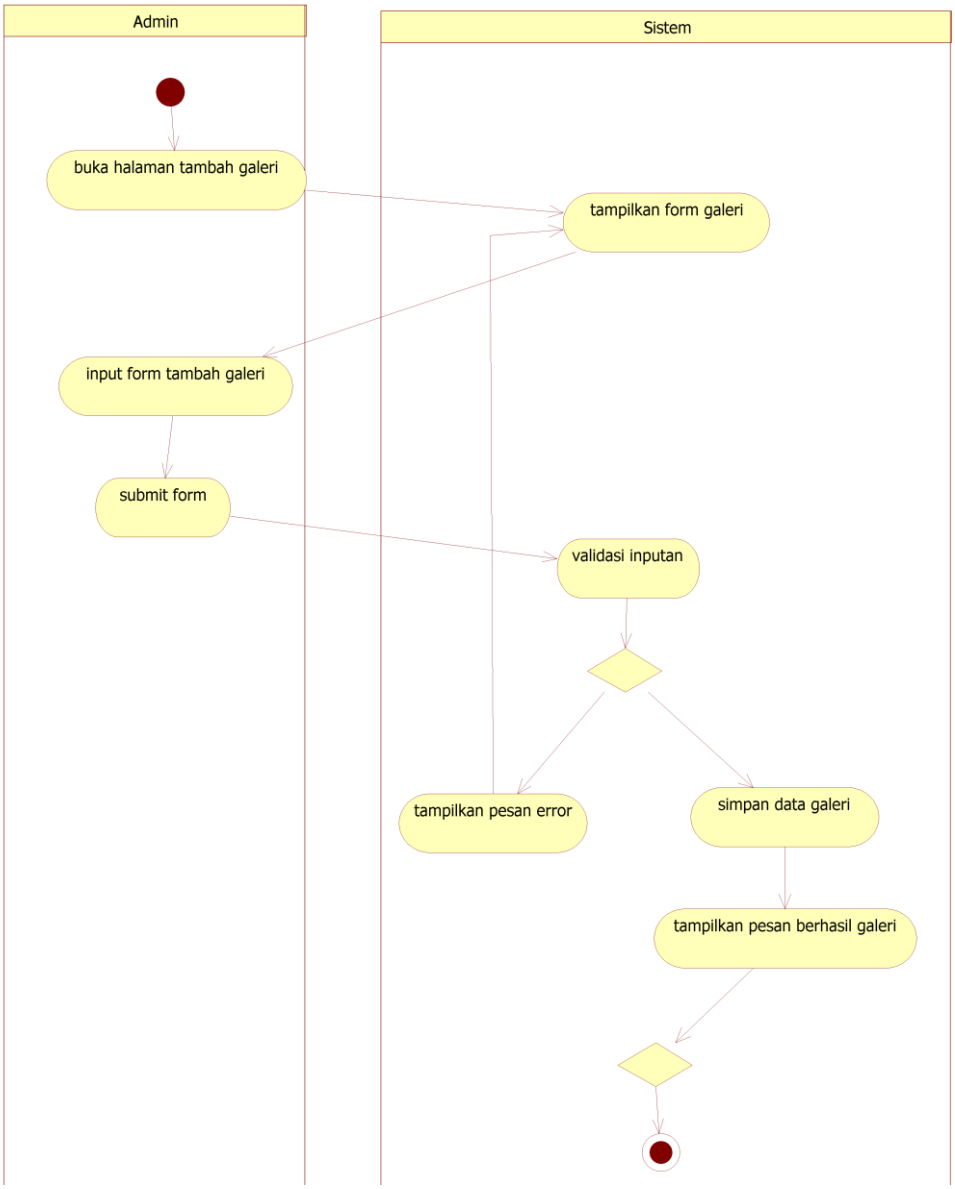
13. *Activity Diagram* Tambah Data Dokumen



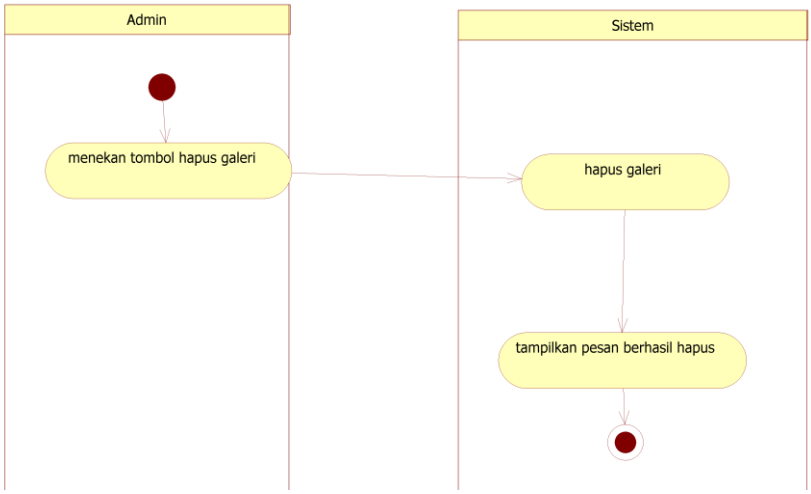
14. *Activity Diagram* Hapus Data Dokumen



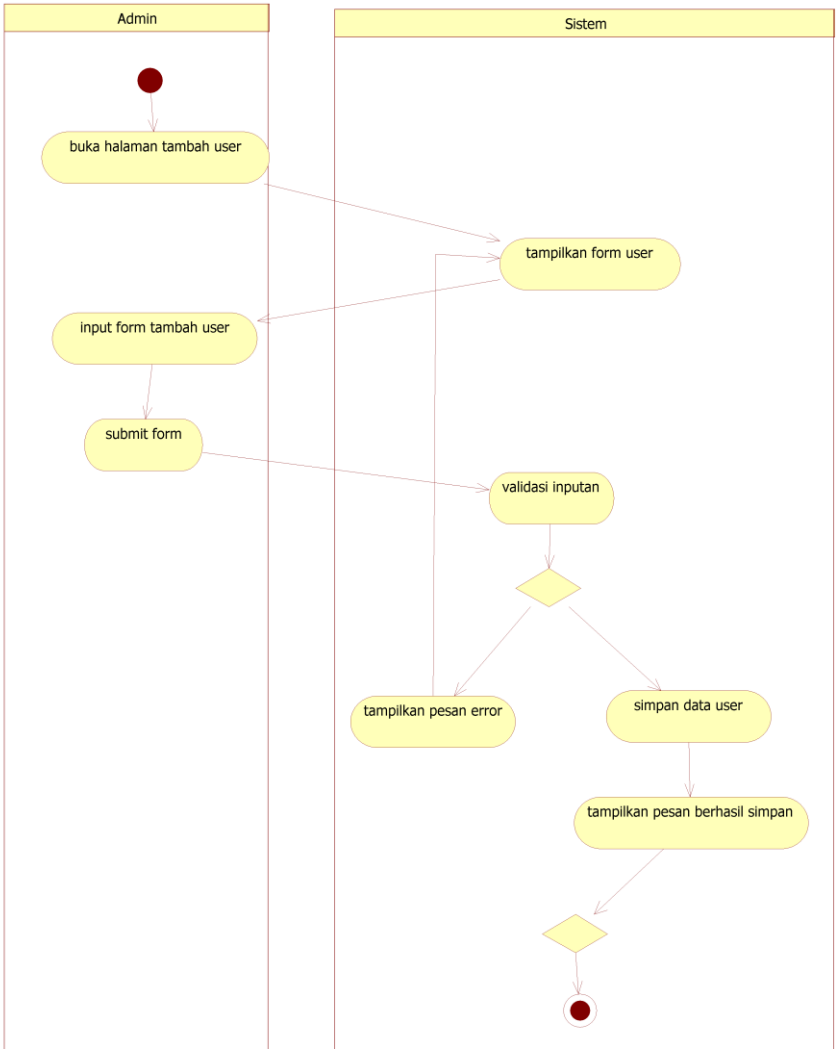
15. *Activity Diagram* Tambah Data Galeri



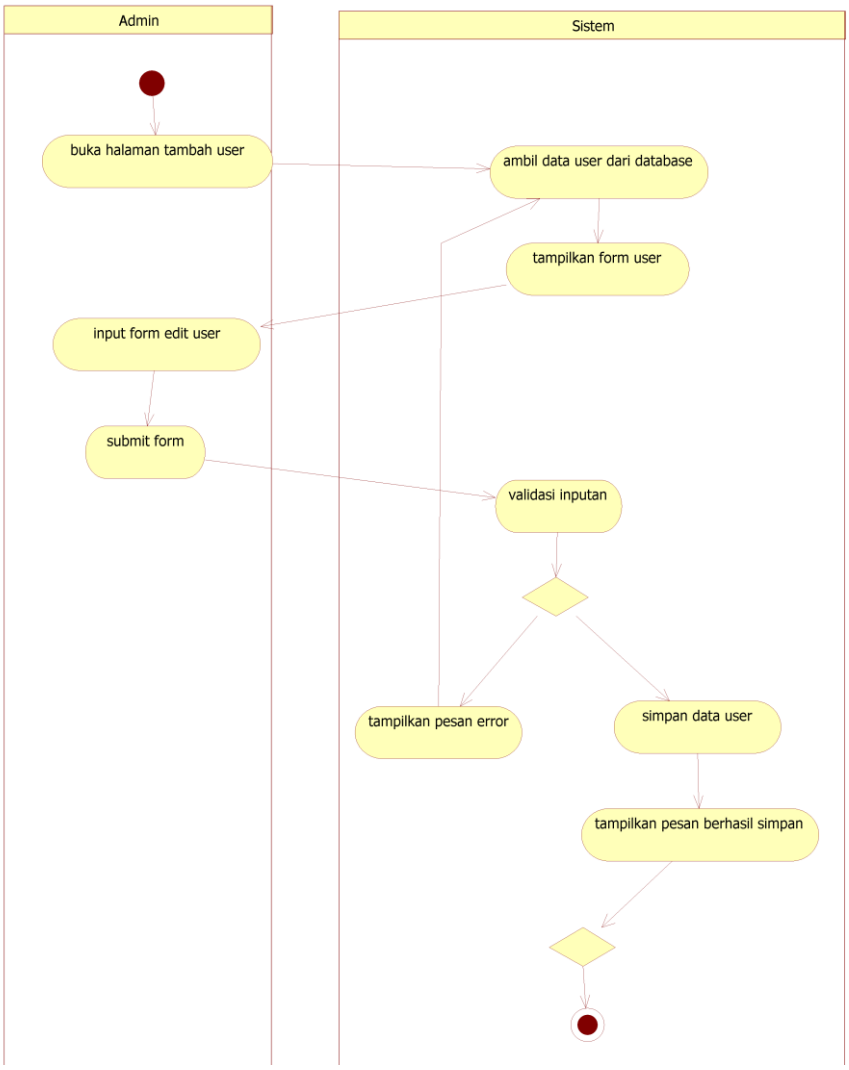
16. *Activity Diagram* Hapus Data Galeri



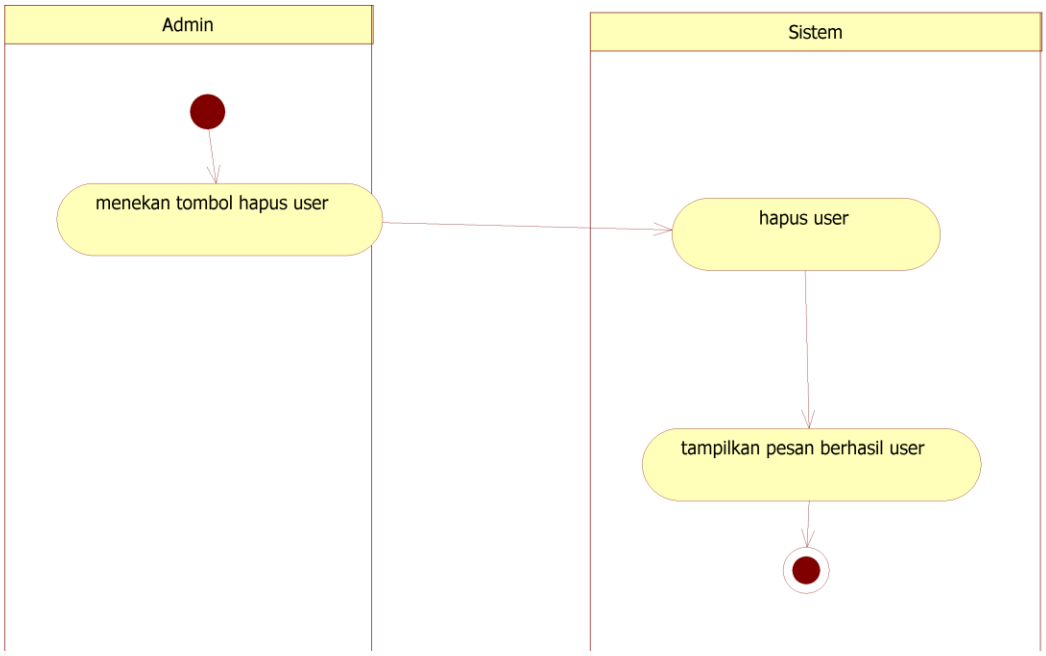
17. *Activity Diagram* Tambah Data User



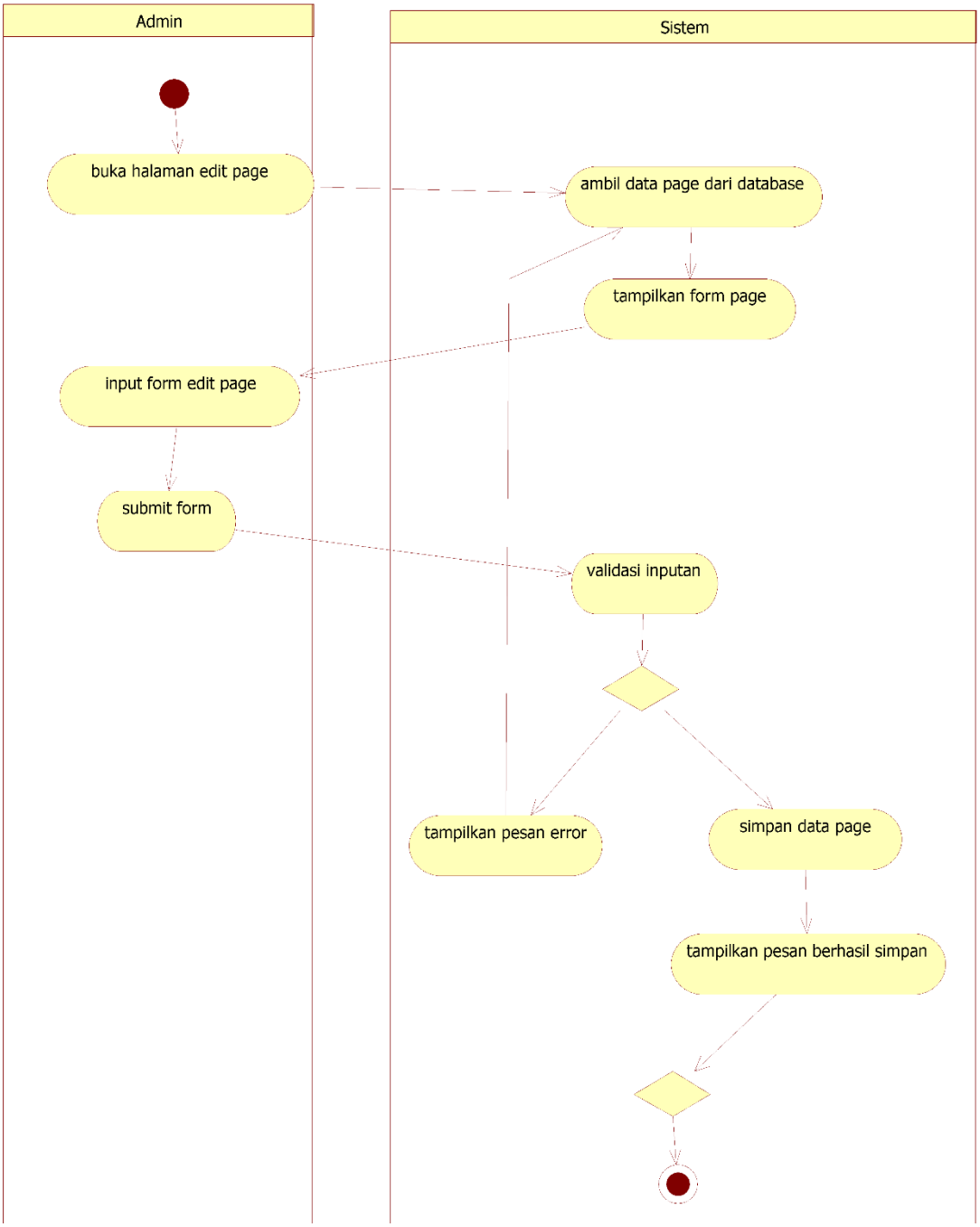
18. *Activity Diagram* Edit Data User



19. *Activity Diagram* Hapus Data User



20. *Activity Diagram* Edit Tentang Web

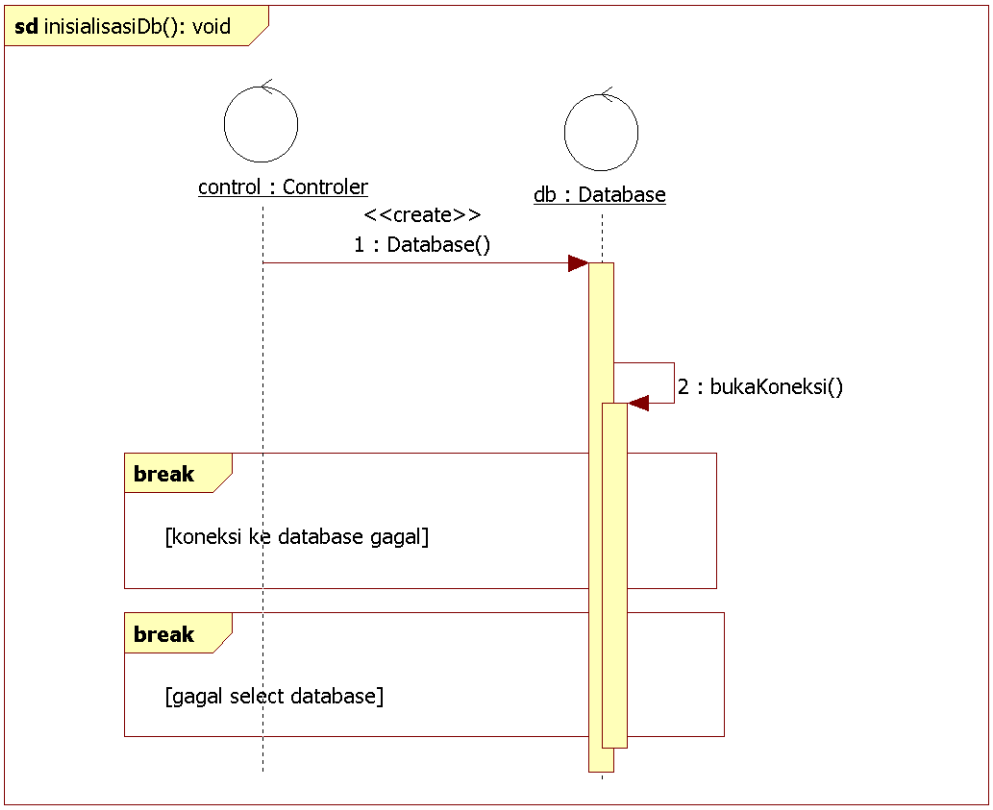


3.1.4.5 Diagram Sequence

Sequence diagram adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Diagram sequence untuk Web Profil Aplikasi Corporate berdasarkan diagram usecase pada gambar adalah sebagai berikut :

1. Diagram Sequence Inisialisasi Koneksi Database

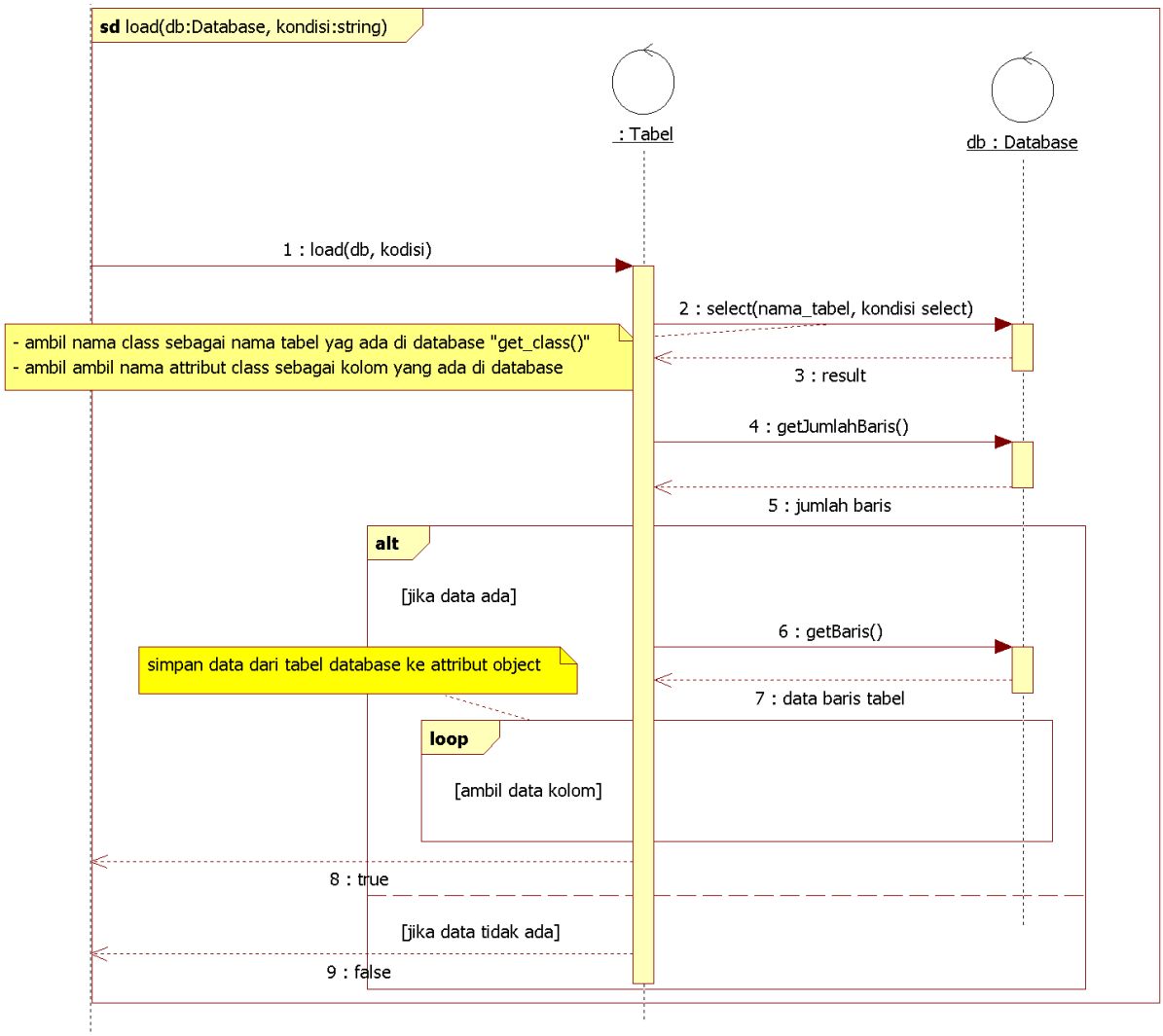
   Diagram sequence Inisialisasi Koneksi Database dapat dilihat pada gambar 3.1.4.2.



Gambar 3.1.4.2 Diagram Squence Inisialisasi Koneksi Database

2. Diagram Sequence Mengambil Data Tabel Dari Database

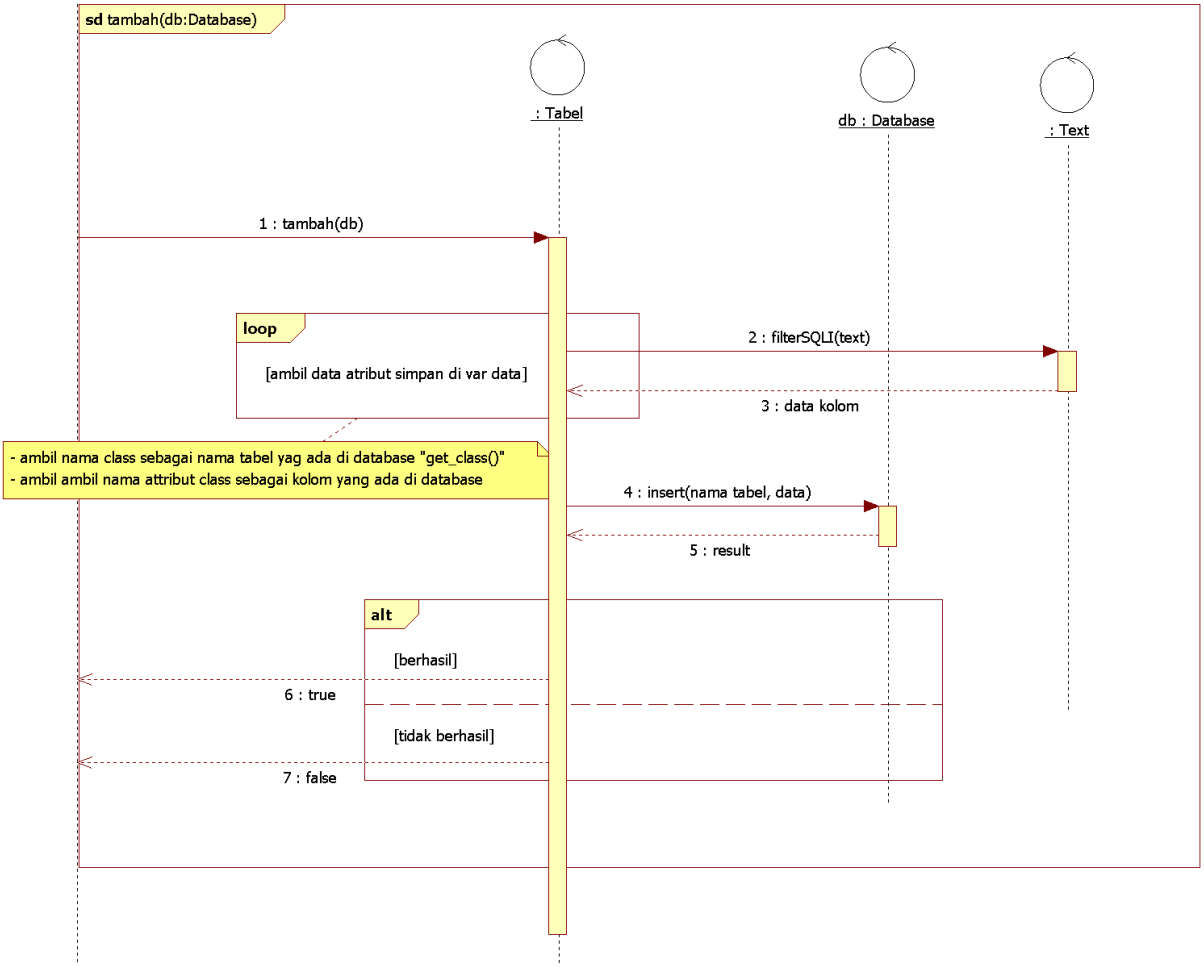
Diagram sequence Mengambil Data Tabel Dari Databasedapat dilihat pada gambar 3.1.4.3.



Gambar 3.1.4.3Diagram Sequence Ambil data table dari database

3. Diagram Sequence Tambah Data Tabel Database

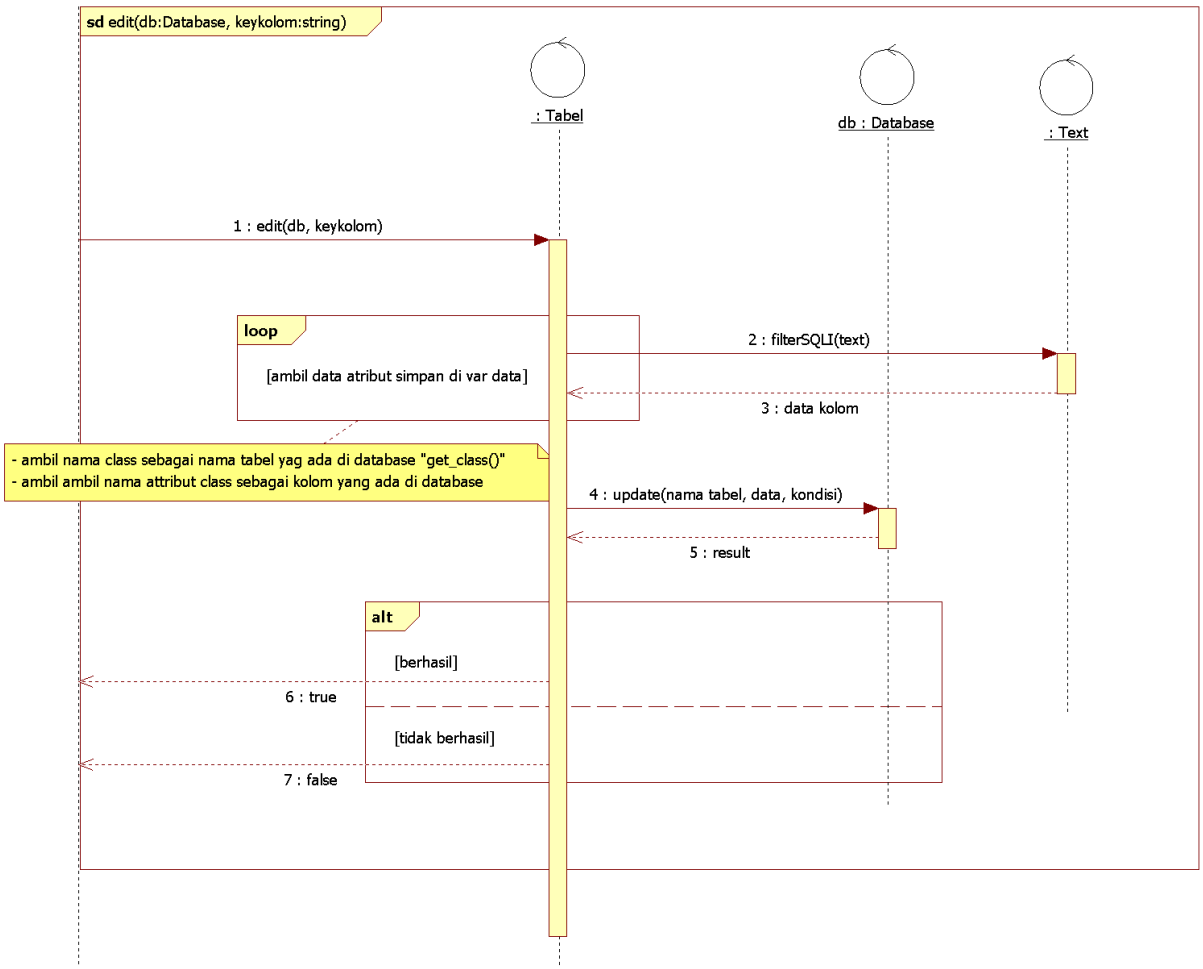
Diagram sequence Tambah Data Tabel Databasedapat dilihat pada gambar 3.1.4.4.



Gambar 3.1.4.4Diagram Sequence Tambah data tabel database

4. Diagram Sequence Edit Data Tabel Database

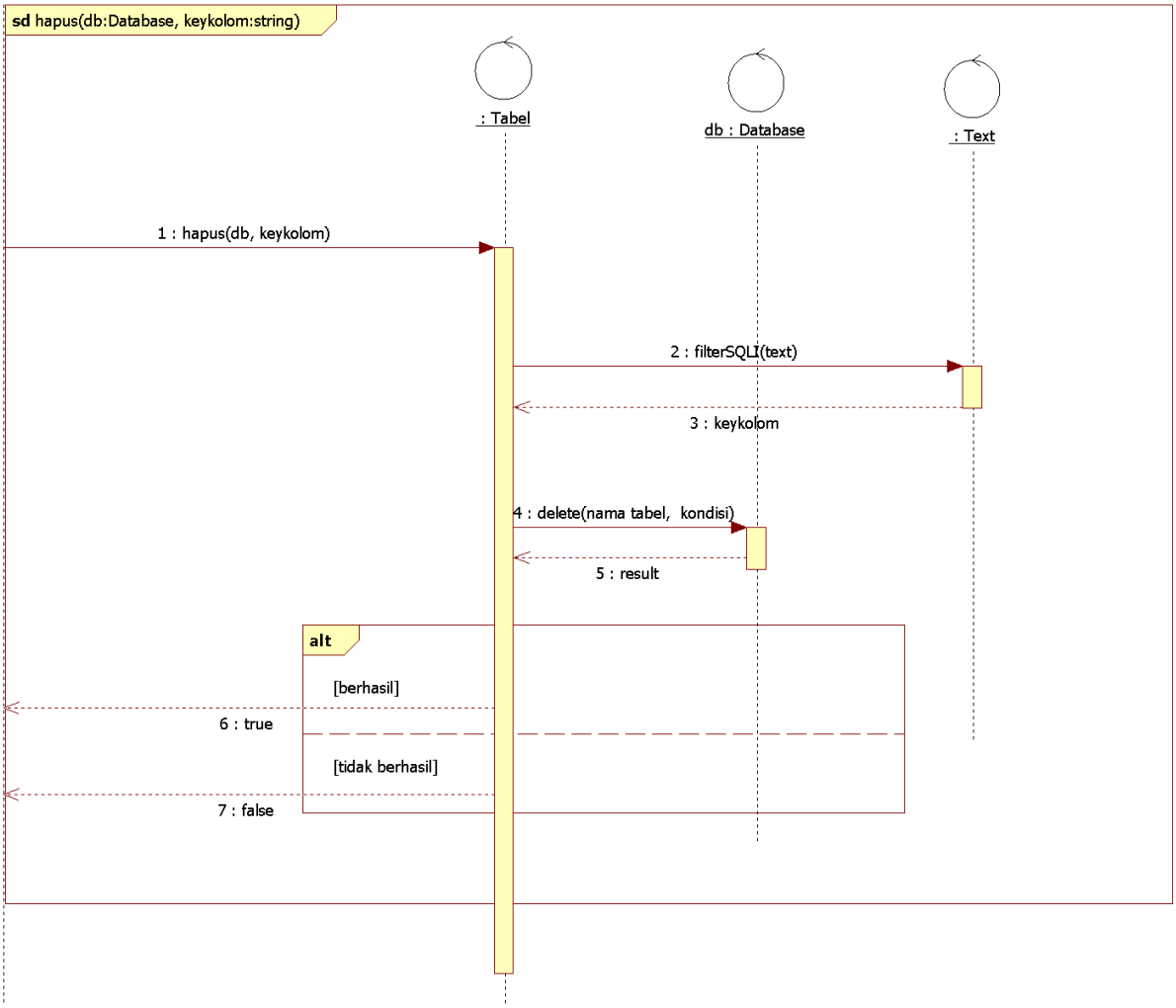
Diagram sequence Edit Data Tabel Database dapat dilihat pada gambar 3.1.4.5.



Gambar 3.1.4.5Diagram sequence Edit Data Tabel Database

5. Diagram Sequence Hapus Data Tabel Database

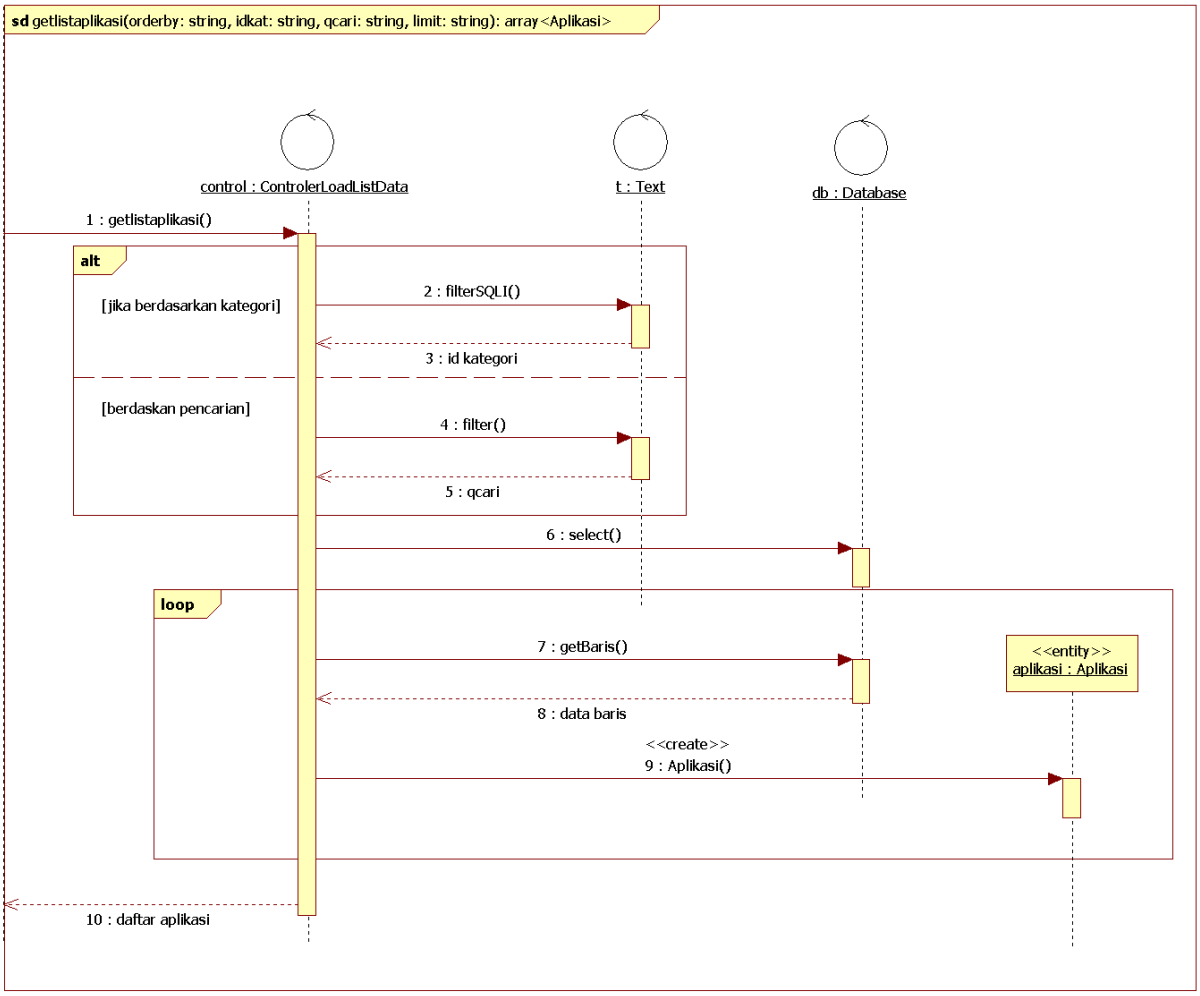
Diagram sequence Hapus Data Tabel Database dapat dilihat pada gambar 3.1.4.6.



Gambar 3.1.4.6Diagram sequence Hapus Data Tabel Database

6. Diagram Sequence Ambil Daftar Aplikasi

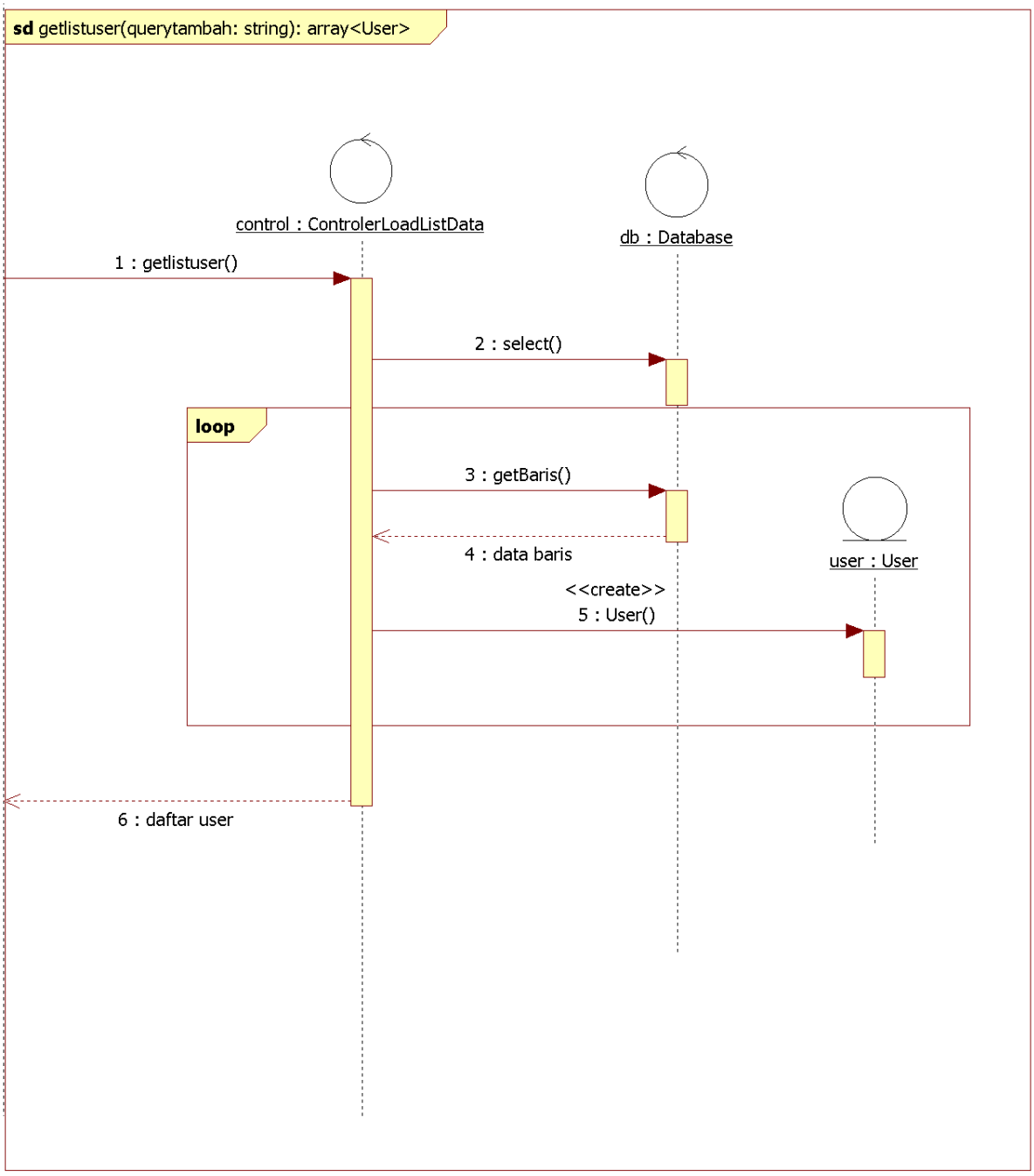
Diagram sequence Ambil Daftar Aplikasi dapat dilihat pada gambar 3.1.4.7.



Gambar 3.1.4.7Diagram sequence Ambil Daftar Aplikasi

7. Diagram Sequence Ambil Daftar User

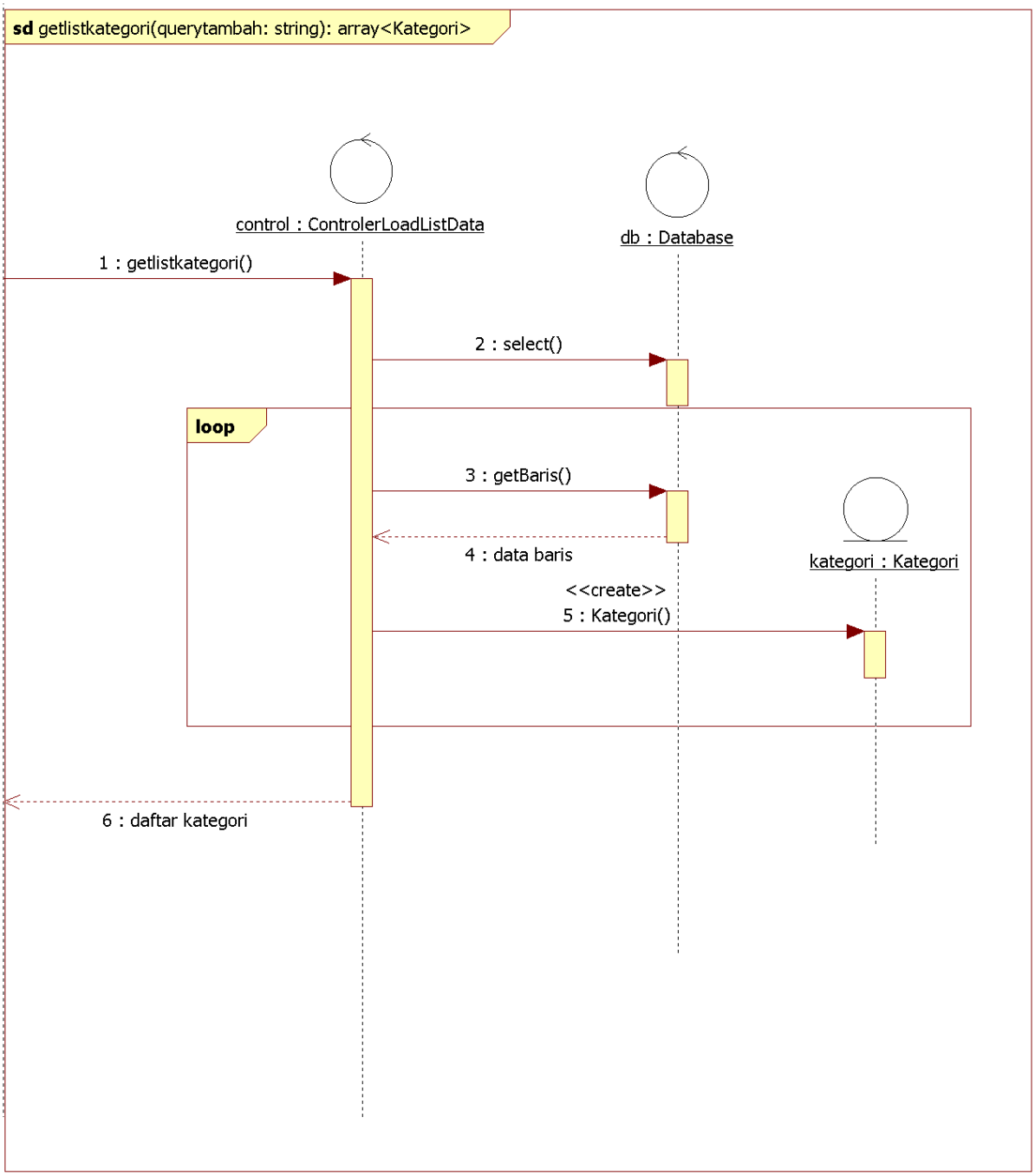
Diagram sequence Ambil Daftar User dapat dilihat pada gambar 3.1.4.8.



Gambar 3.1.4.8Diagram sequence Ambil Daftar User

8. Diagram Sequence Ambil Daftar Kategori

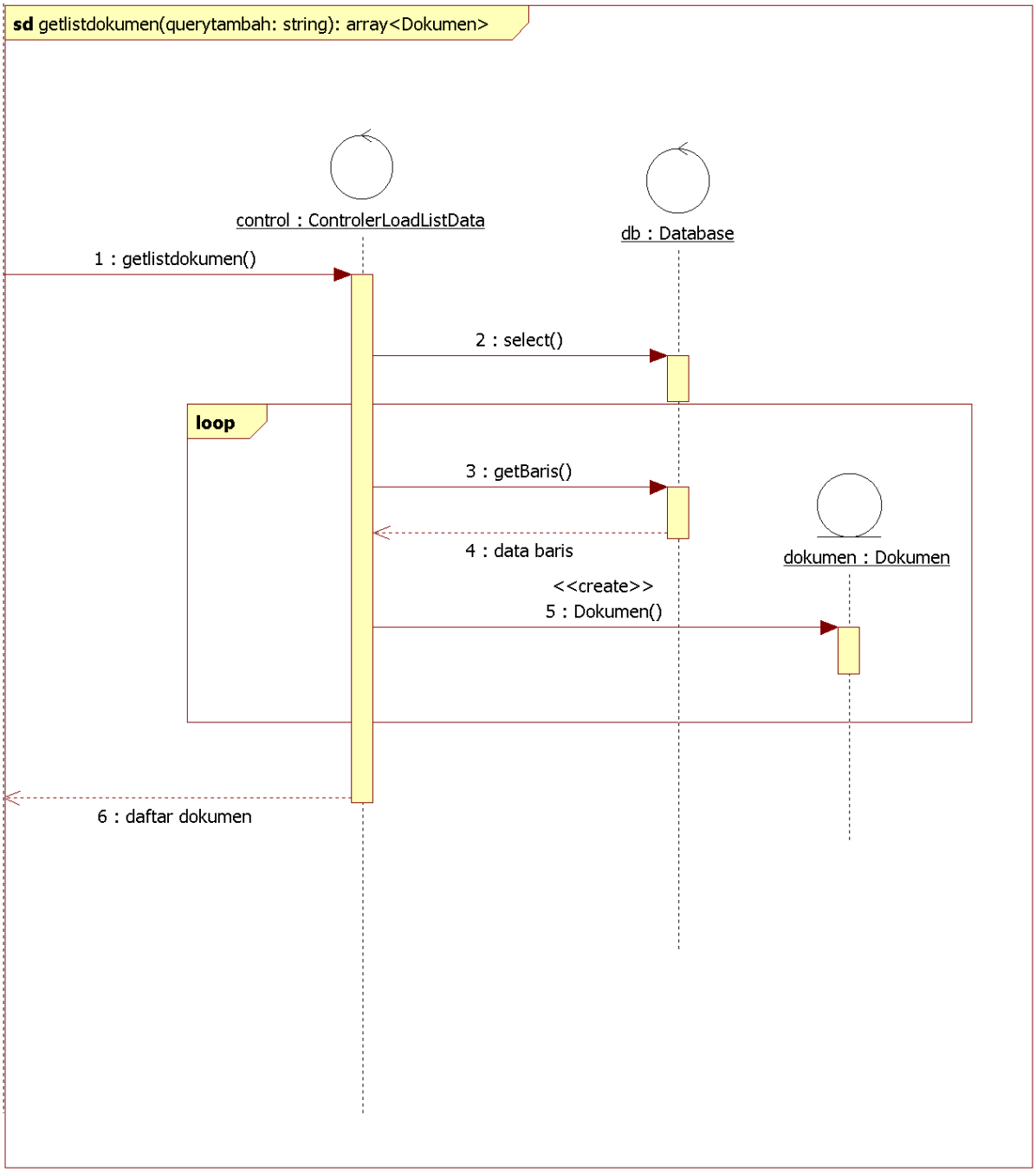
Diagram sequence Ambil Daftar Kategori dapat dilihat pada gambar 3.1.4.9.



Gambar 3.1.4.9Diagram sequence Ambil Daftar Kategori

9. Diagram Sequence Ambil Daftar Dokumen

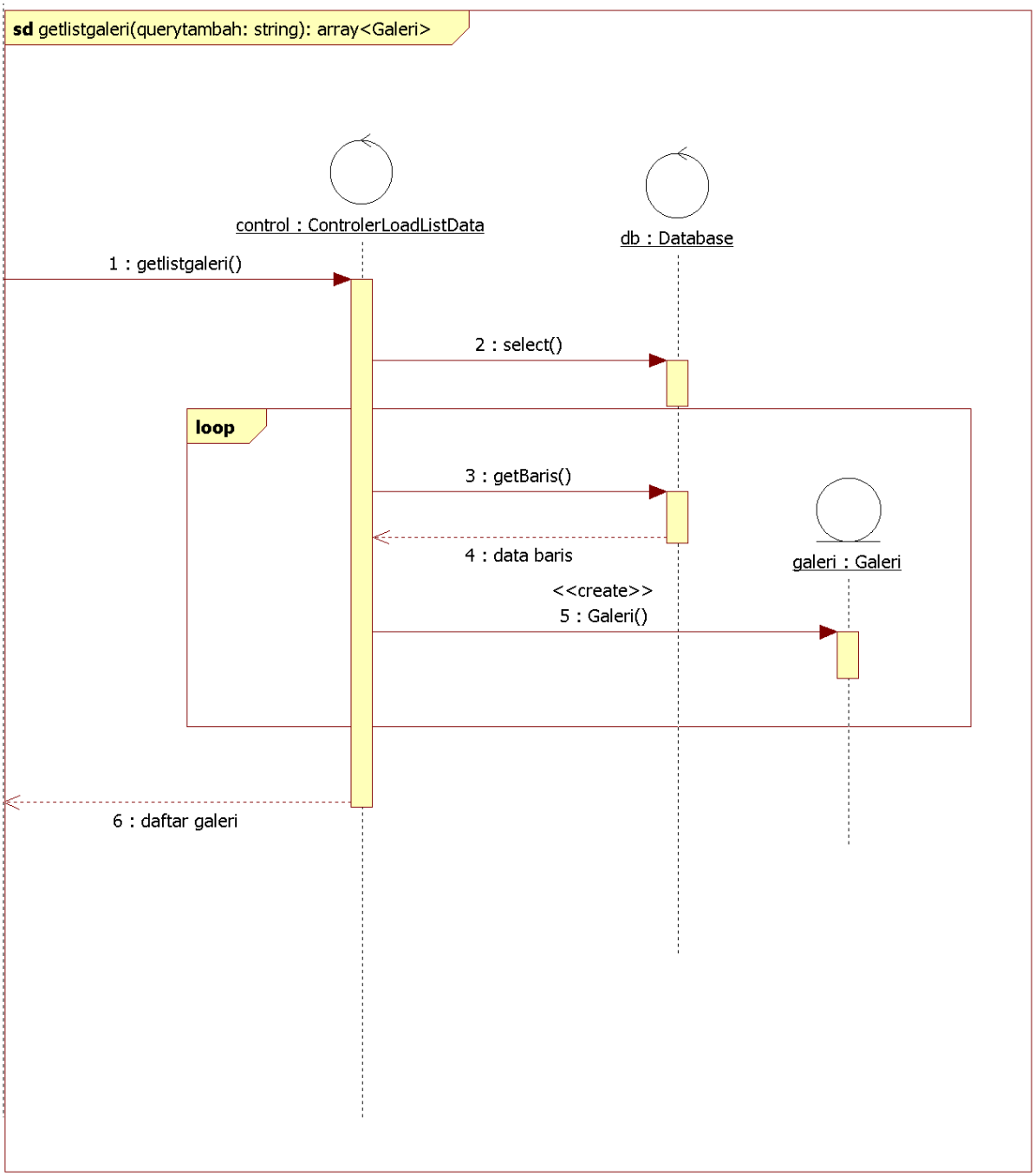
Diagram sequence Ambil Daftar Dokumen dapat dilihat pada gambar 3.1.4.10.



Gambar 3.1.4.10Diagram sequence Ambil Daftar Dokumen

10. Diagram Sequence Ambil Daftar Galeri

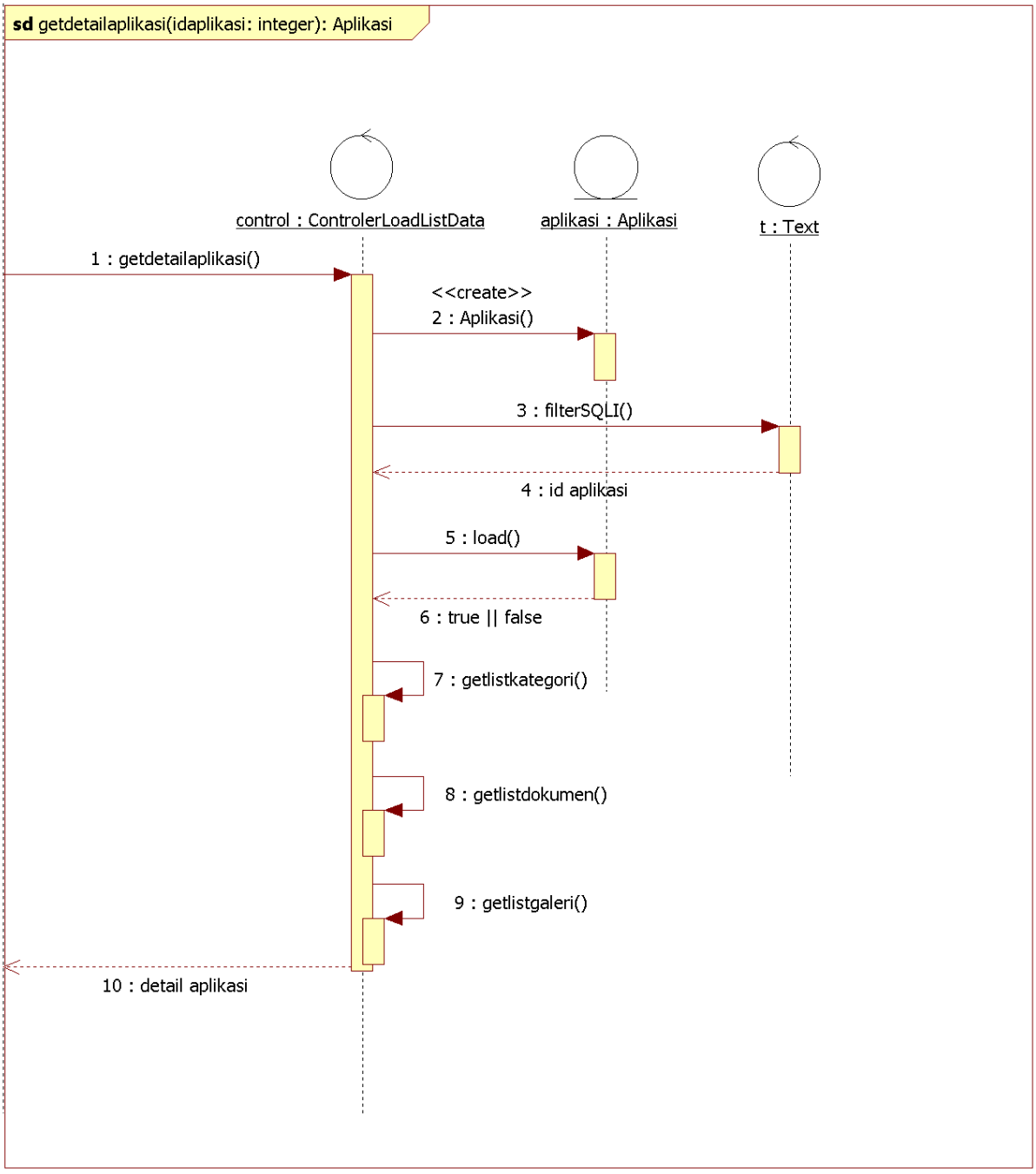
Diagram sequence Ambil Daftar Galeri dapat dilihat pada gambar 3.1.4.11.



Gambar 3.1.4.11Diagram sequence Ambil Daftar Galeri

11. Diagram Sequence Ambil Detail Aplikasi

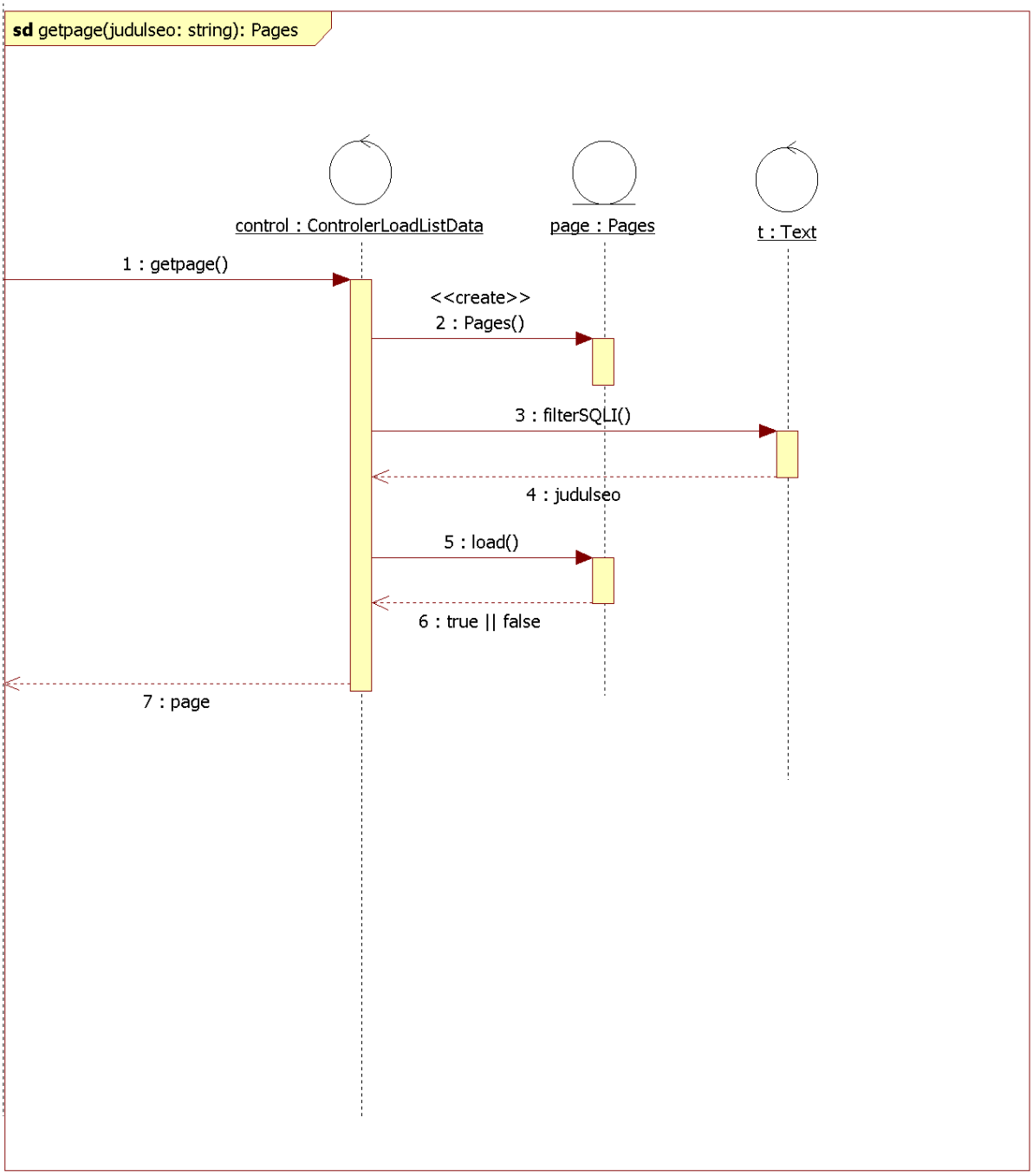
Diagram sequence Ambil Detail Aplikasi dapat dilihat pada gambar 3.1.4.12.



Gambar 3.1.4.12Diagram sequence Ambil Detail Aplikasi

12. Diagram Sequence Ambil Pages

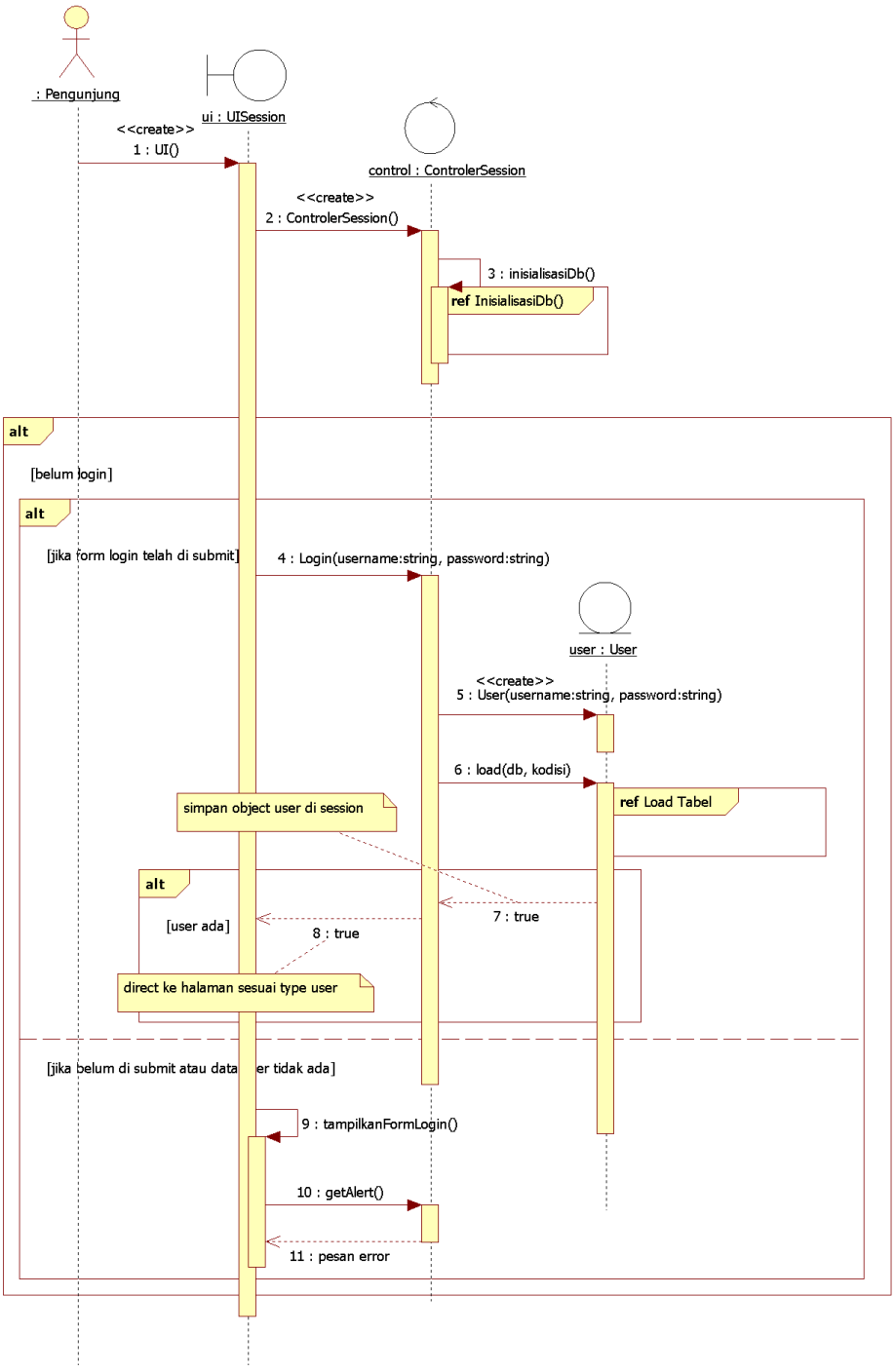
Diagram sequence Ambil Pages dapat dilihat pada gambar 3.1.4.13.



Gambar 3.1.4.13Diagram sequence Ambil Pages

13. Diagram Sequence Login

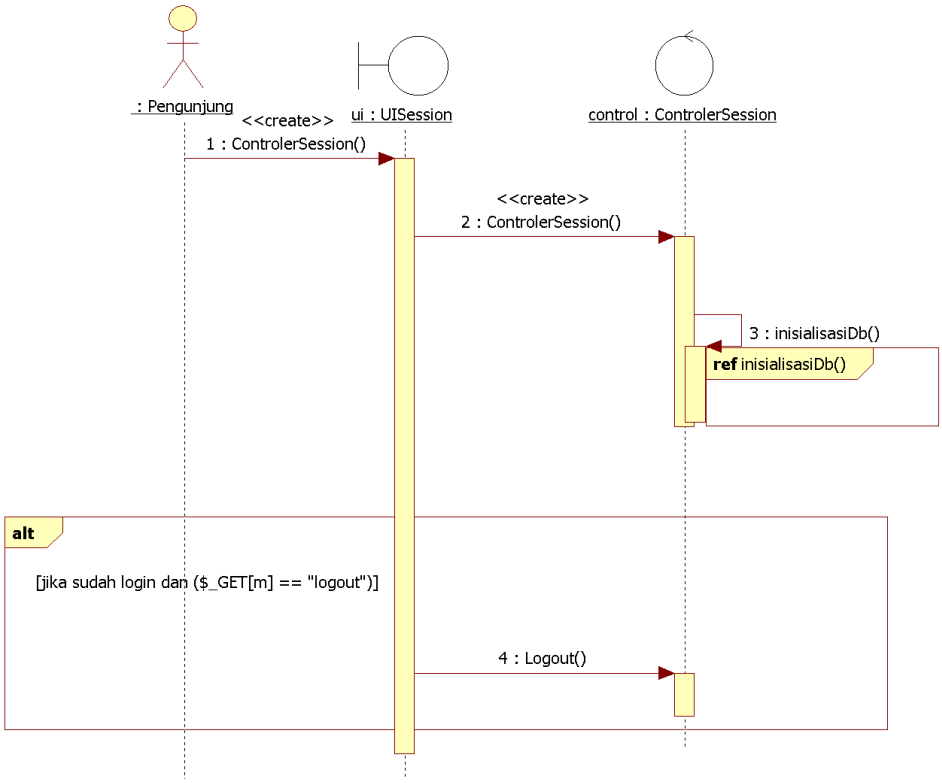
Diagram sequence Login dapat dilihat pada gambar 3.1.4.14.



Gambar 3.1.4.14Diagram sequence Login

14. Diagram Sequence Logout

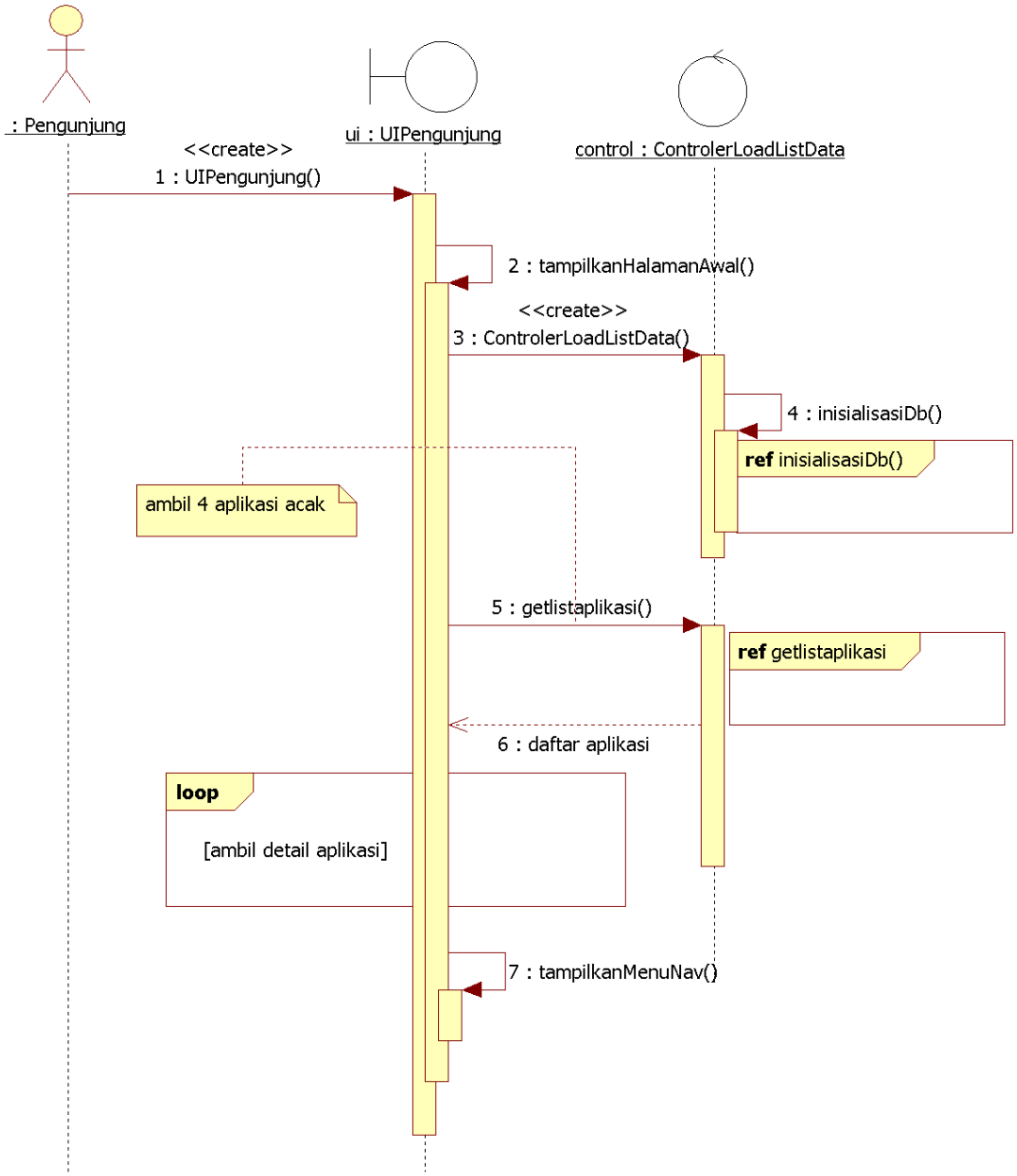
Diagram sequence Logout dapat dilihat pada gambar 3.1.4.15.



Gambar 3.1.4.15Diagram sequence Logout

15. Diagram Sequence Halaman Apal Pengunjung

Diagram sequence Halaman Apal Pengunjung dapat dilihat pada gambar 3.1.4.16.



Gambar 3.1.4.16Diagram sequence Halaman Apal Pengunjung

16. Diagram Sequence Lihat Daftar Aplikasi

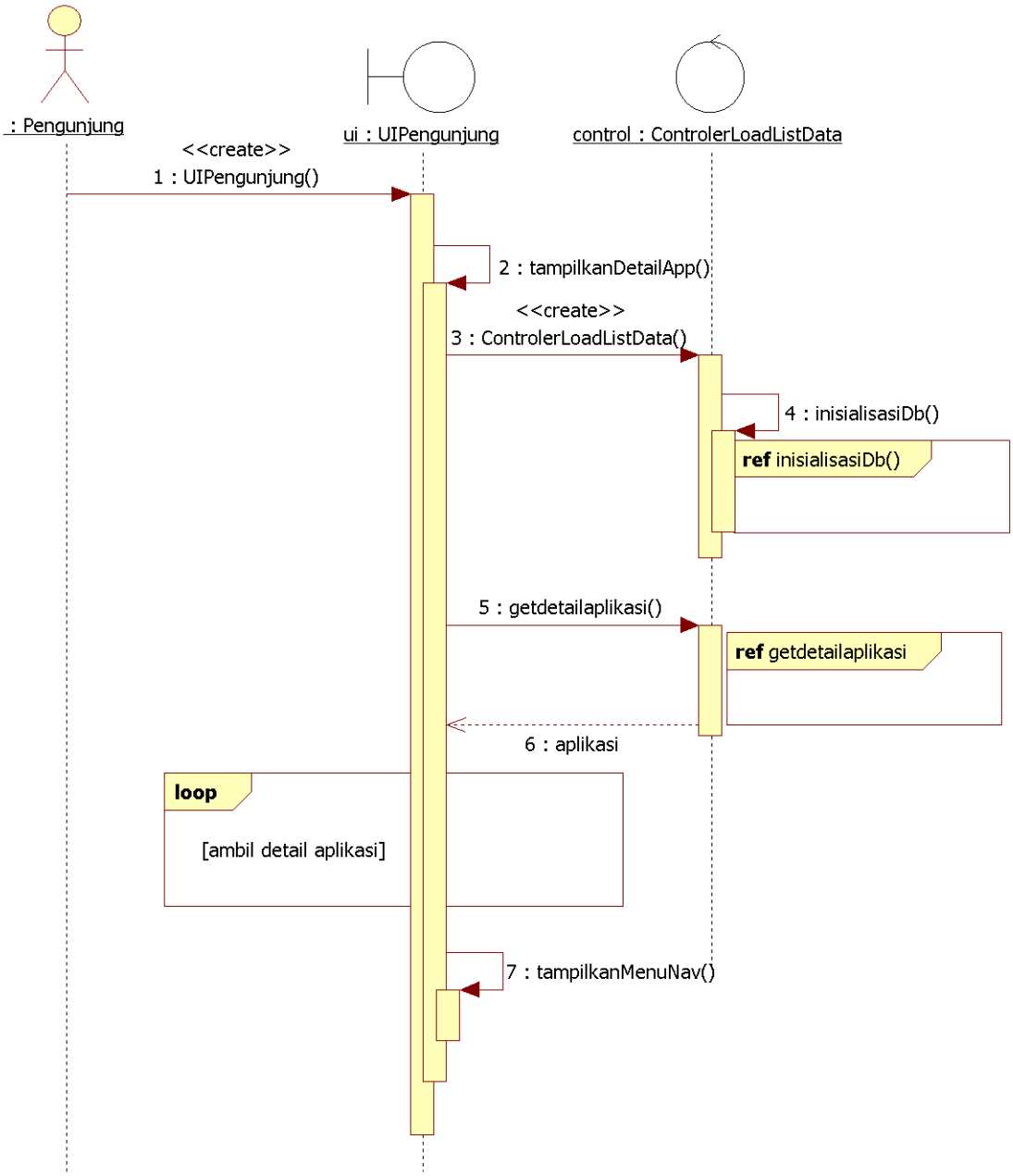
Diagram sequence Lihat Daftar Aplikasi dapat dilihat pada gambar 3.1.4.17.



Gambar 3.1.4.17Diagram sequence Lihat Daftar Aplikasi

17. Diagram Sequence Lihat Detail Aplikasi

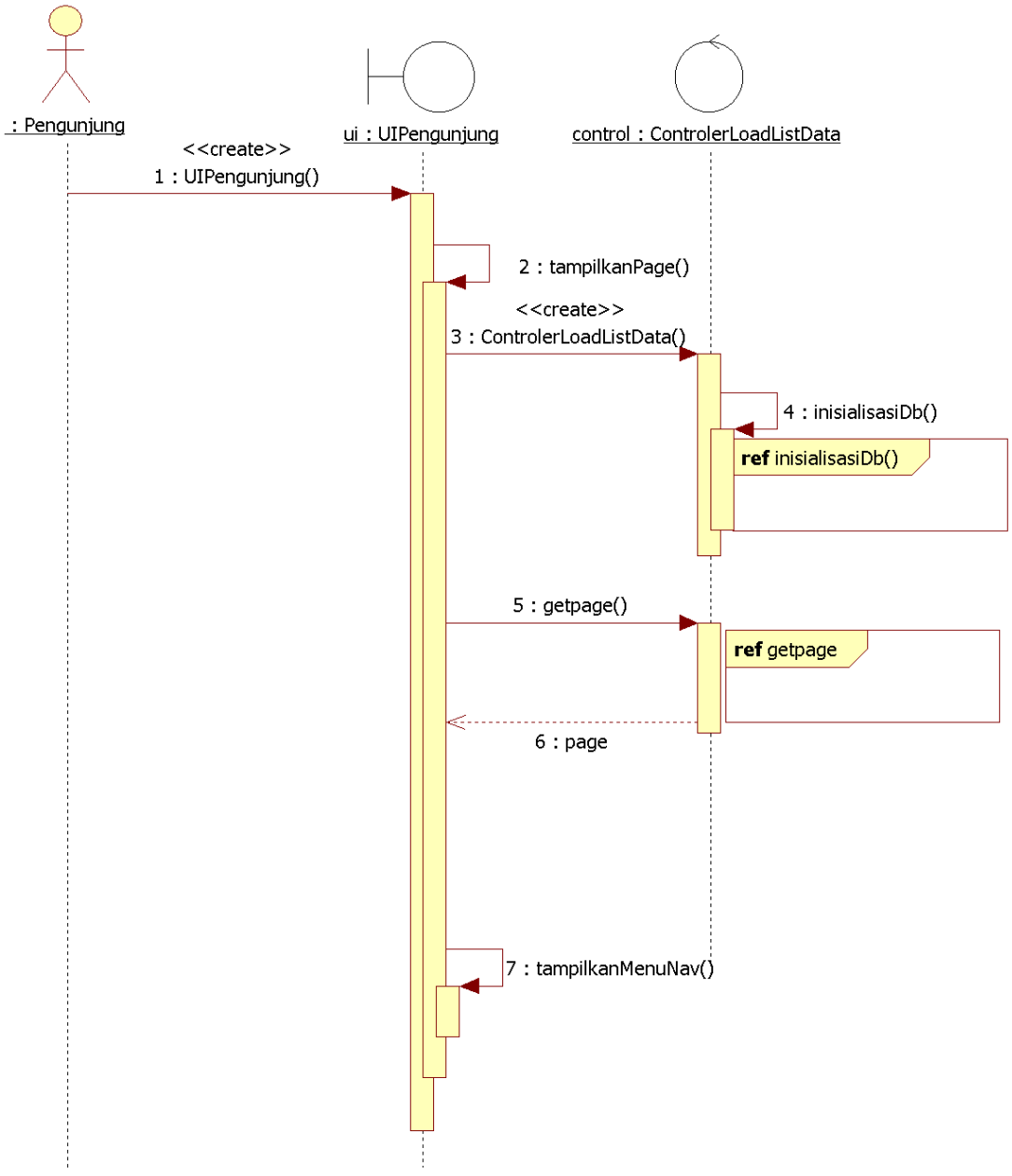
Diagram sequence Lihat Detail Aplikasi dapat dilihat pada gambar 3.1.4.18.



Gambar 3.1.4.18Diagram sequence Lihat Detail Aplikasi

18. Diagram Sequence Lihat Tentang Web

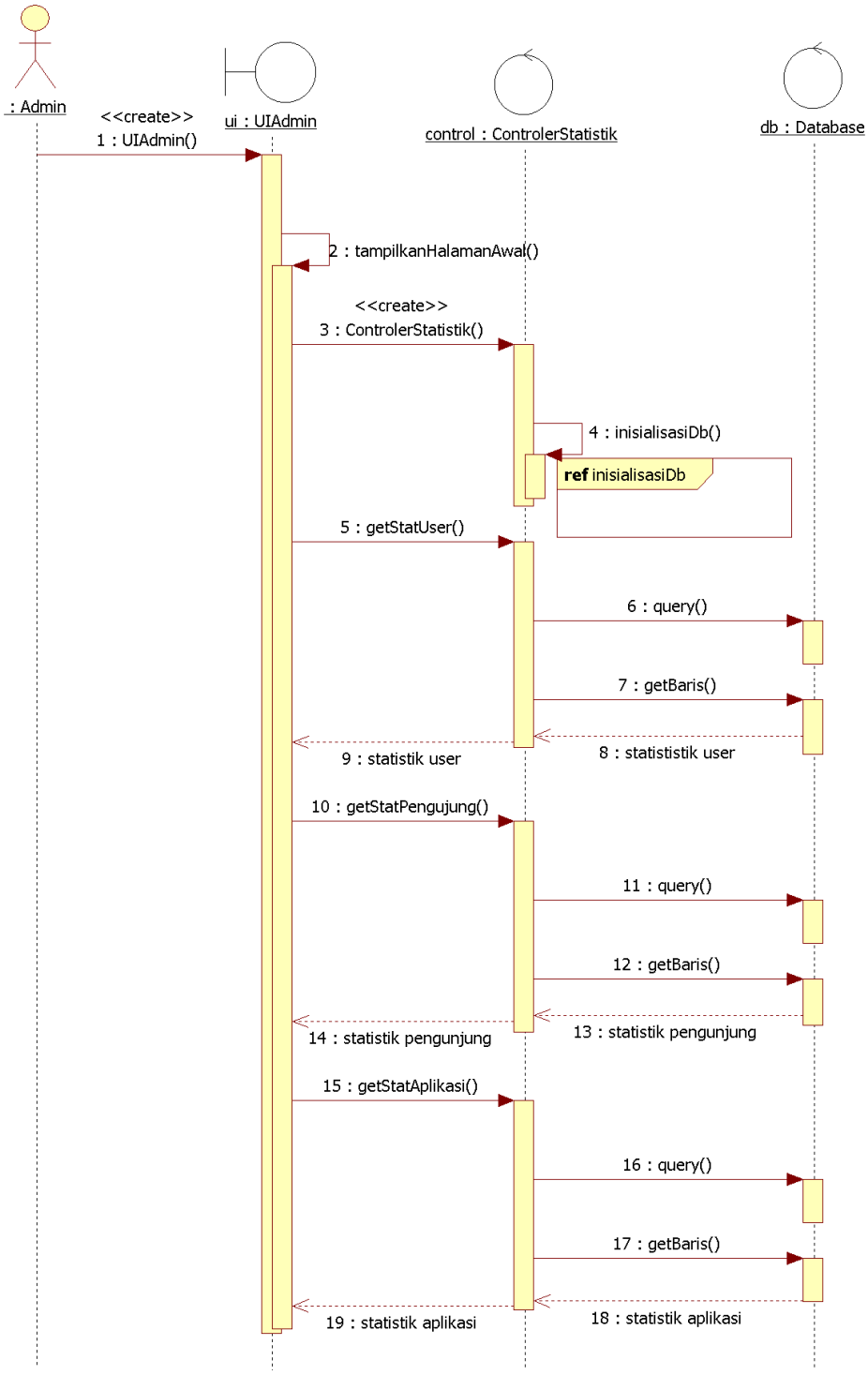
Diagram sequence Lihat Tentang Web dapat dilihat pada gambar 3.1.4.19.



Gambar 3.1.4.19Diagram sequence Lihat Tentang Web

19. Diagram Sequence Lihat Statistik

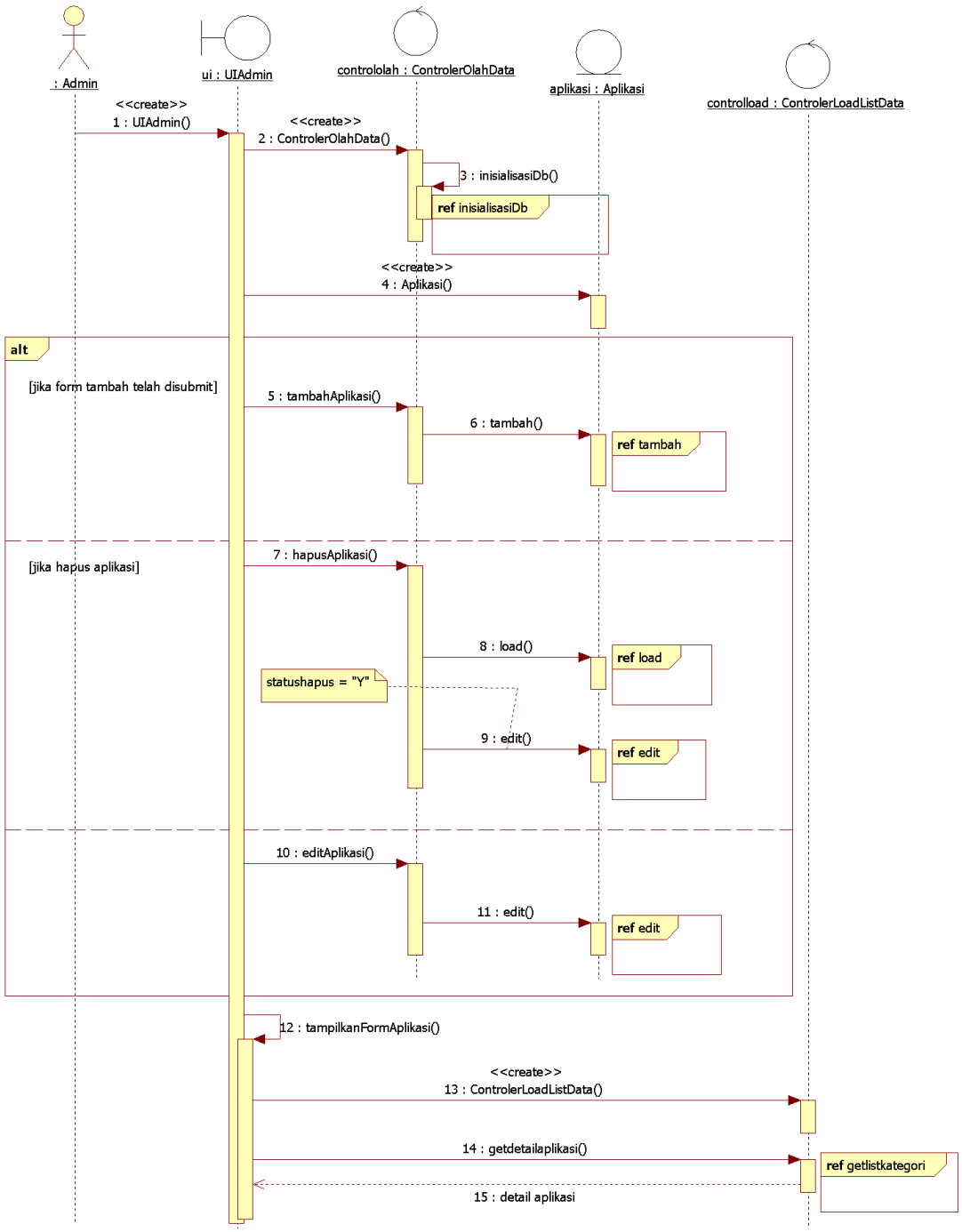
Diagram sequence Lihat Statistik dapat dilihat pada gambar 3.1.4.20.



Gambar 3.1.4.20Diagram sequence Lihat Statistik

20. Diagram Sequence Olah Data Aplikasi

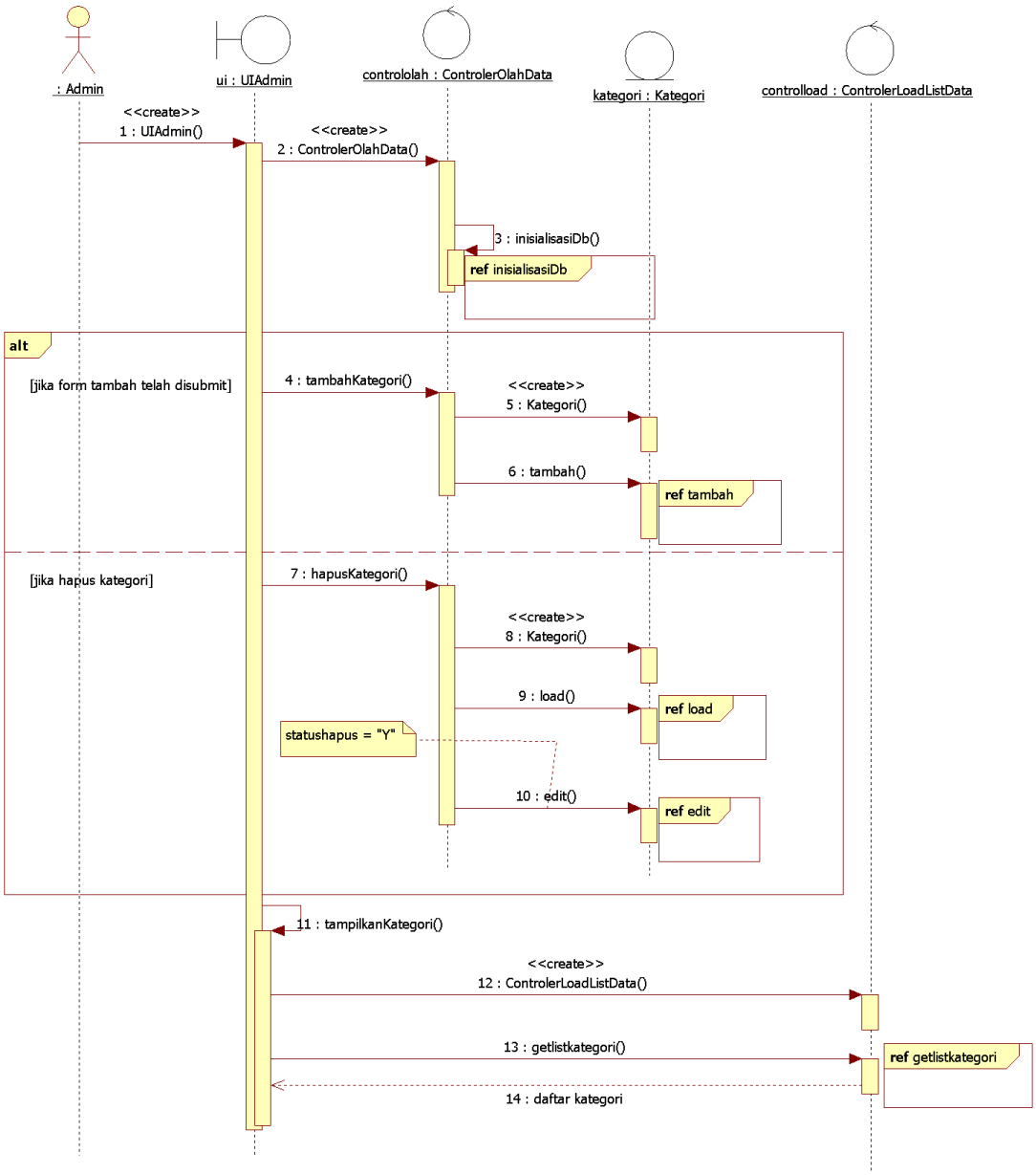
Diagram sequence Olah Data Aplikasi dapat dilihat pada gambar 3.1.4.21.



Gambar 3.1.4.21Diagram sequence Olah Data Aplikasi

21. Diagram Sequence Olah Data Kategori

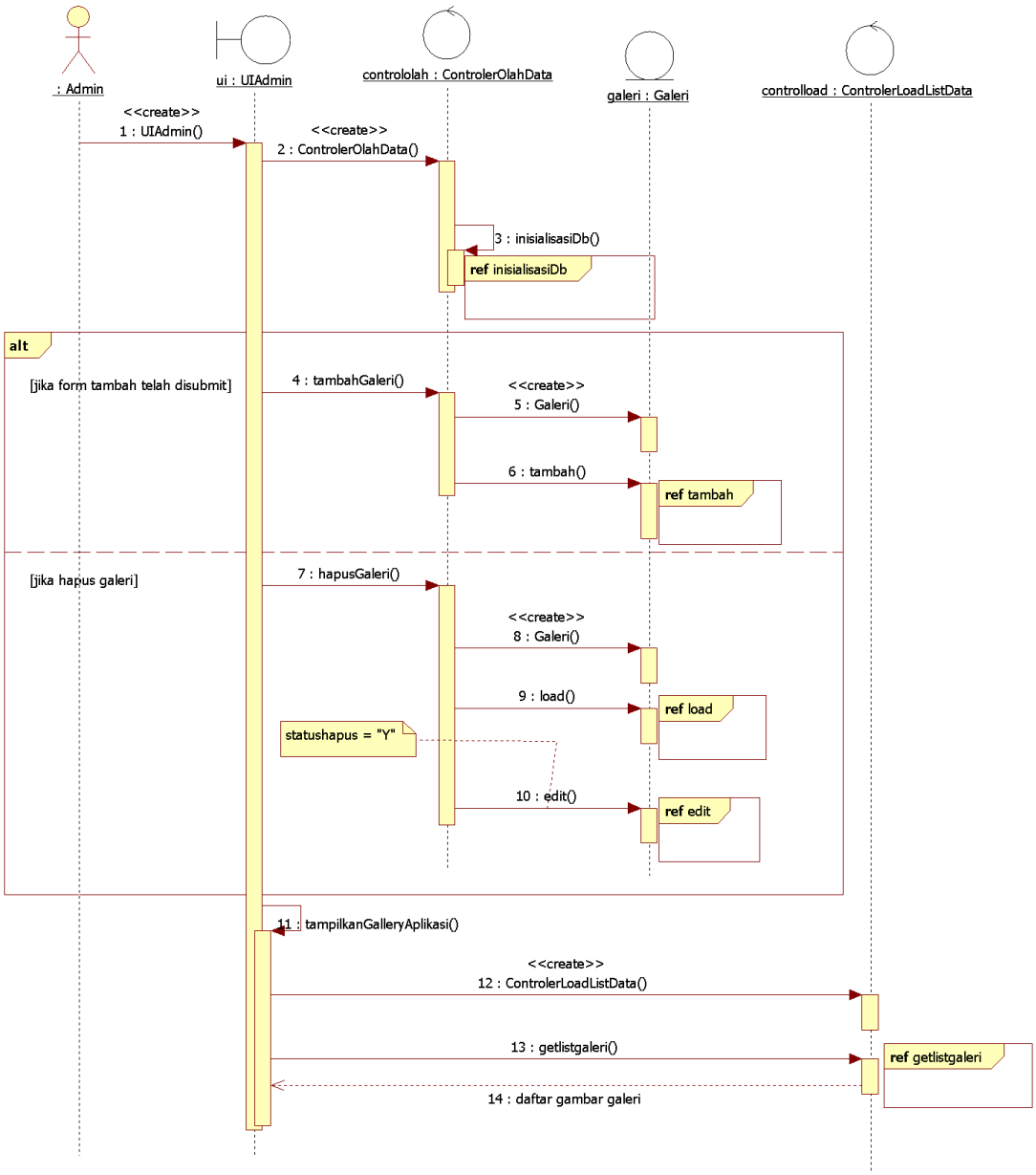
Diagram sequence Olah Data Kategori dapat dilihat pada gambar 3.1.4.22.



Gambar 3.1.4.22Diagram sequence Olah Data Kategori

22. Diagram Sequence Olah Data Galeri

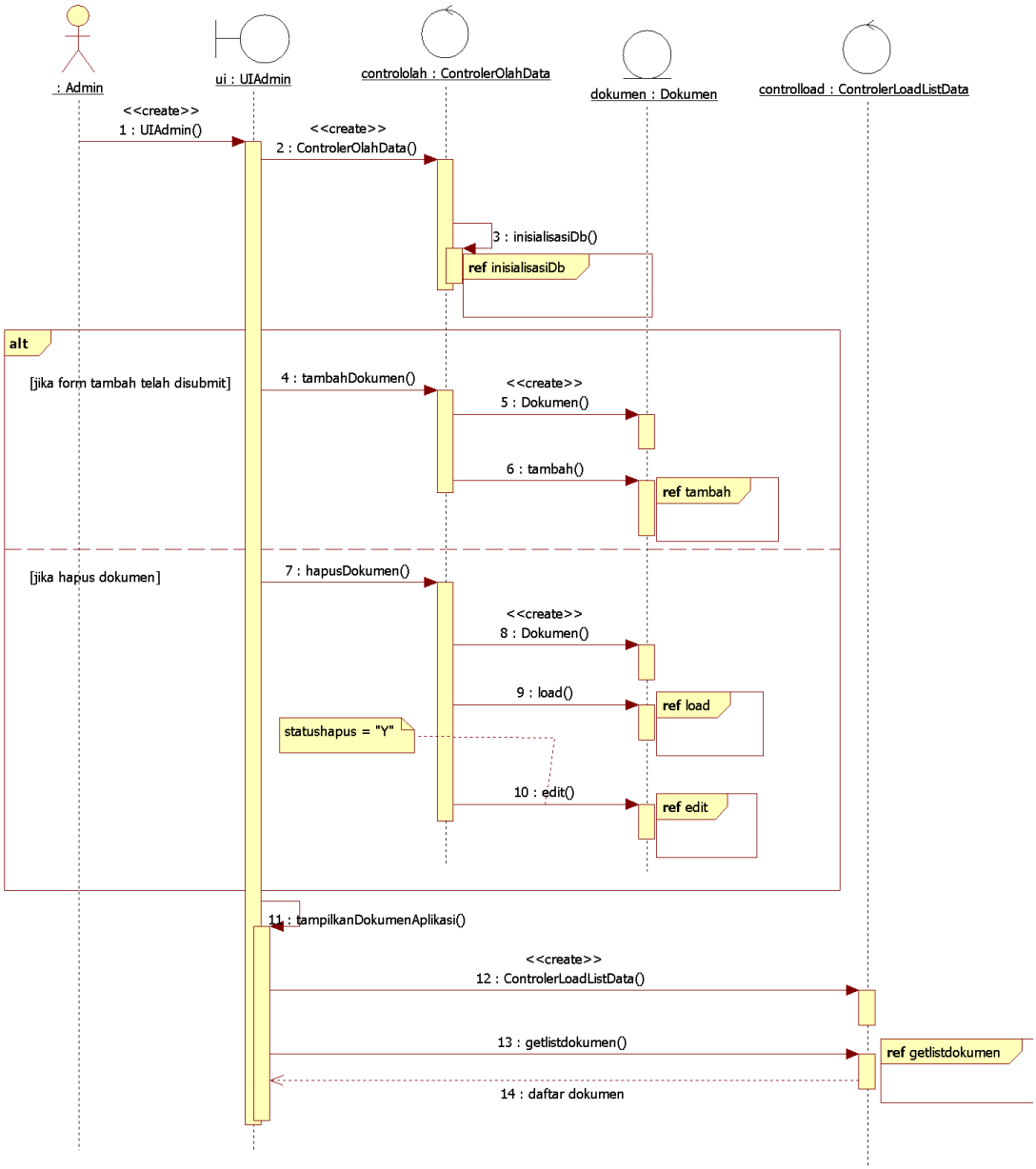
Diagram sequence Olah Data Galeri dapat dilihat pada gambar 3.1.4.23.



Gambar 3.1.4.231. Diagram Sequence Olah Data Galeri

23. Diagram Sequence Olah Data Dokumen

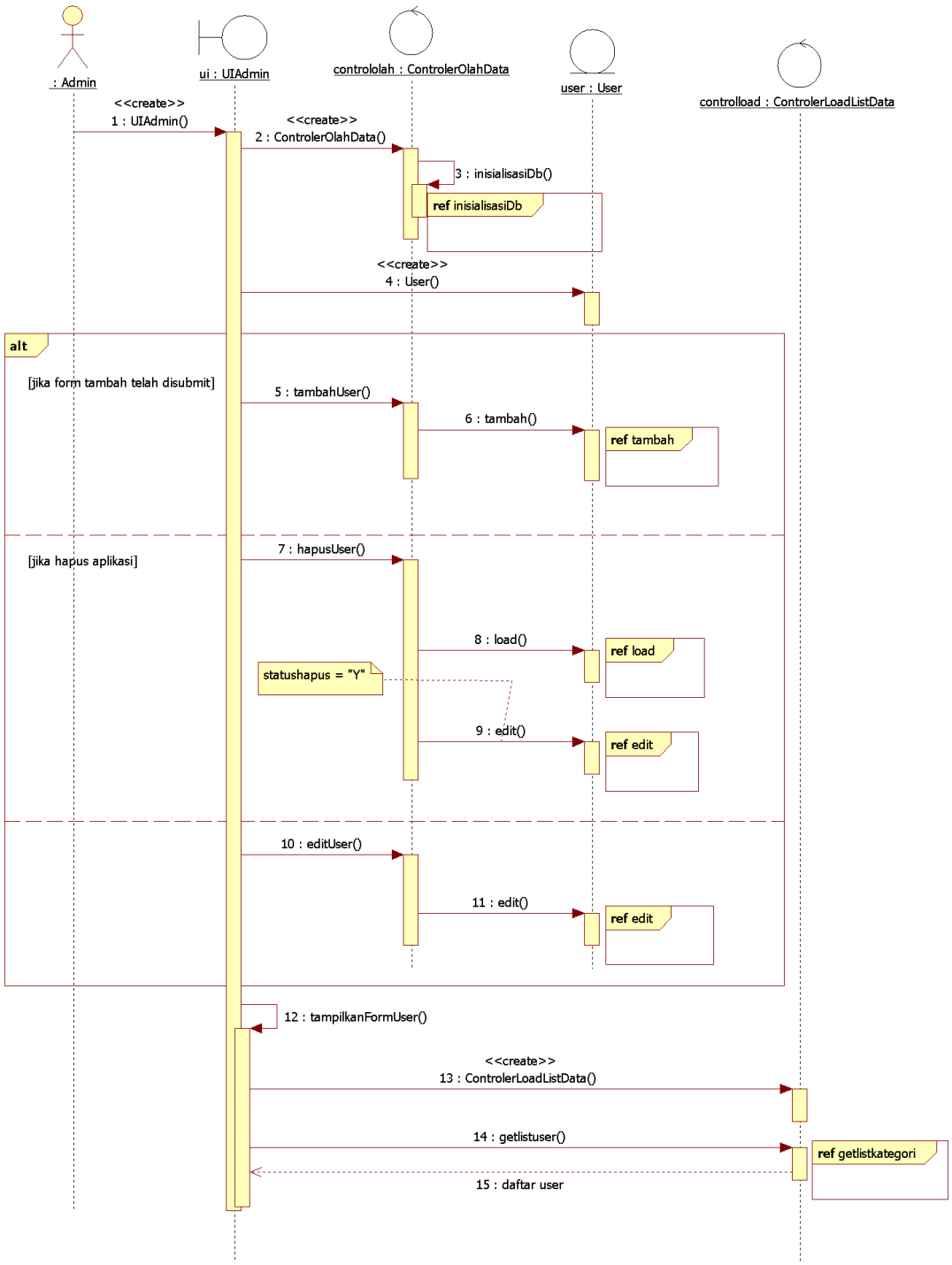
Diagram sequence Olah Data Dokumen dapat dilihat pada gambar 3.1.4.24.



Gambar 3.1.4.24Diagram sequence Olah Data Dokumen

24. Diagram Sequence Olah Data User

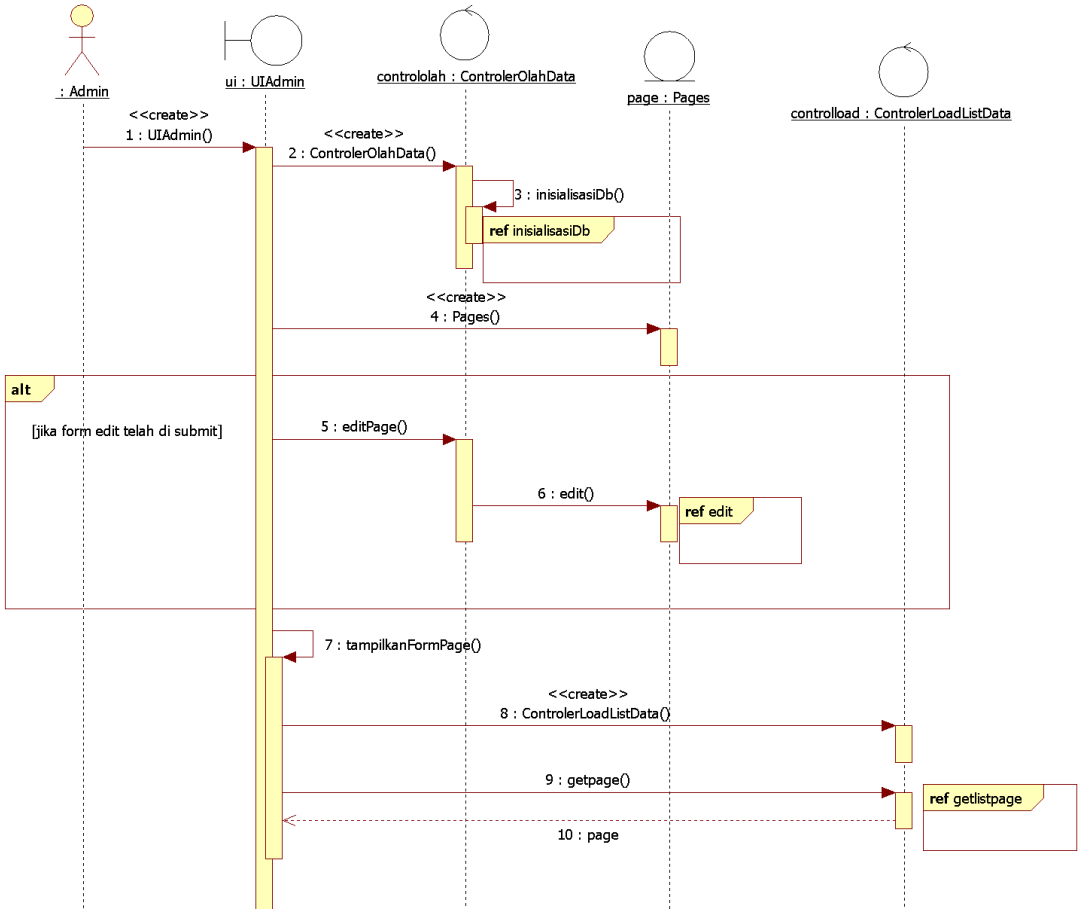
Diagram sequence Olah Data User dapat dilihat pada gambar 3.1.4.25.



Gambar 3.1.4.25Diagram sequence Olah Data User

25. Diagram Sequence Edit Tentang Web

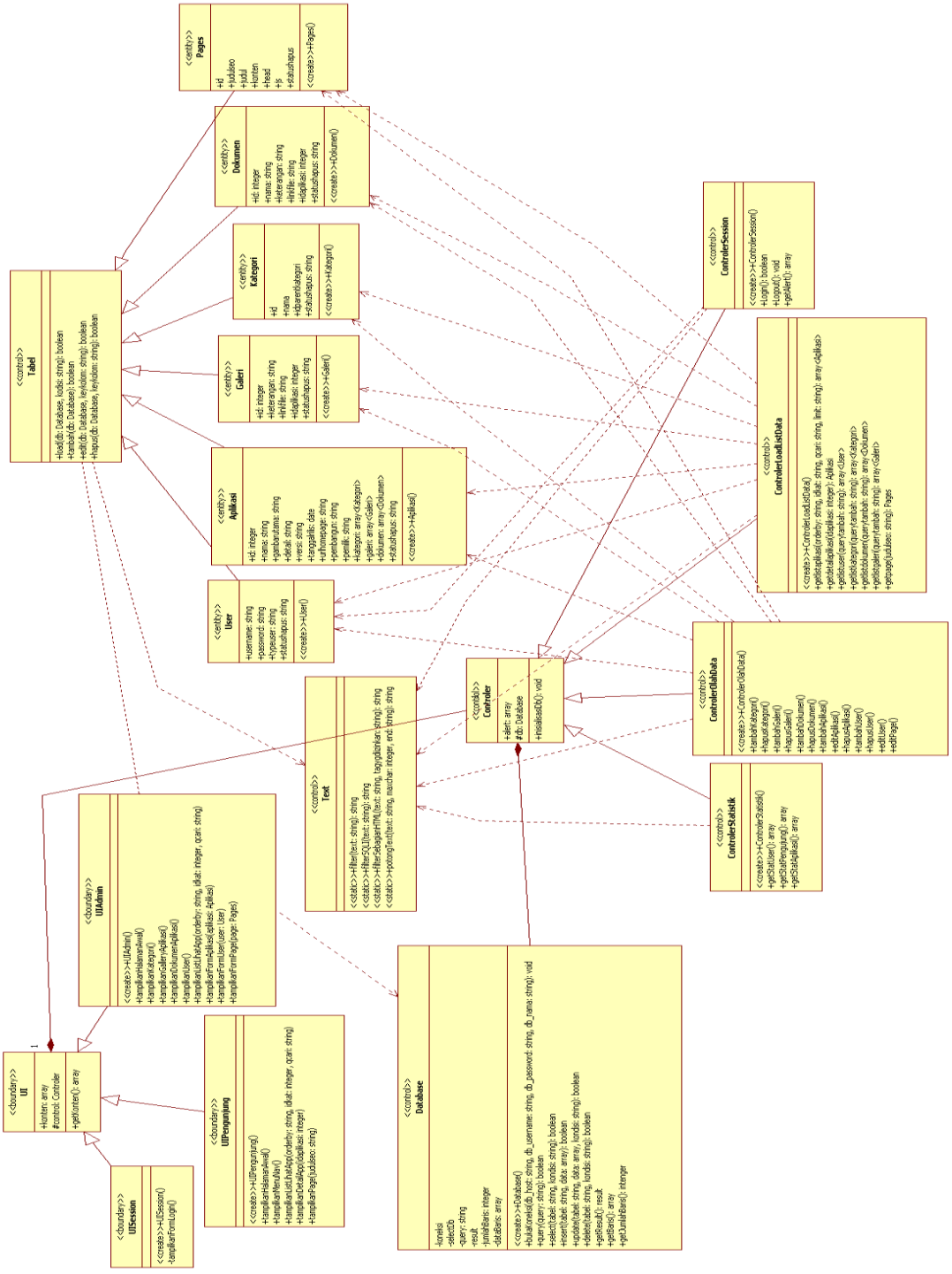
Diagram sequence Edit Tentang Web dapat dilihat pada gambar 3.1.4.26.



Gambar 3.1.4.26Diagram sequence Edit Tentang Web

3.1.4.6 Class Diagram

Class diagram mendeskripsikan jenis–jenis objek dalam sistem dan berbagai macam hubungan statis yang ada. Berikut adalah class diagram untuk Web Profil Aplikasi Corporate PLN.



Gambar 3.1.4.27 Class Diagram

Setiap class memiliki jenis class masing–masing seperti terlihat pada tabel berikut :

No	Nama	Jenis
1.	UI	Boundary
2.	UISession	Boundary
3.	UIPengunjung	Boundary
4.	UIAdmin	Boundary
5.	Controler	Control
6.	ControlerListData	Control
7.	ControlerSession	Control
8.	ControlerStatistik	Control
9.	ControlerOlahData	Control
10.	Database	Control
11.	Text	Control
12.	Tabel	Control
13.	Aplikasi	Entity
14.	Dokumen	Entity
15.	Galeri	Entity
16.	Kategori	Entity
17.	Pages	Entity
18.	User	Entity

Deskripsi tiap class untuk data tersebut adalah sebagai berikut:

1. Class UI

Class UI merupakan class yang berisi prosedur-prosedur pengatur tampilan yang berinteraksi langsung dengan user.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
UI	*public*	*merupakan constructor yang langsung membuat object Controler*
getKonten	*public*	*getter untuk atribut webKonten*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
kontenWEB	*private*	*array[]*
control	*Protected*	*Controler*

2. Class UISession
   UISession merupakan turunan dari class UI yang berisi prosedur-prosedur pengatur tampilan yang berinteraksi langsung dengan user saat user belum login.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
UISession	*public*	*merupakan constructor*
tampilkanFormLogin	*public*	*untuk menampilkan form Login*

3. Class UIPengunjung
   UIPengunjung merupakan turunan dari class UI yang berisi prosedur-prosedur pengatur tampilan yang berinteraksi langsung dengan user bertype pegunjung.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
UIPengunjung	*public*	*merupakan constructor yang langsung membuat object Controler*

tampilkanHalamanAwal	*public*	*Menampilkan halaman awal*
TampilkanMenuNav	*Public*	*Menampilkan Navigasi*
tampilkanListLihatApp	*Public*	*Menampilkan daftar aplikasi*
tampilkanDetailApp	*Public*	*Menampilkan detail dari aplikasi*

4. Class UIAdmin
   UIAdmin merupakan turunan dari class UI yang berisi prosedur-prosedur pengatur tampilan yang berinteraksi langsung dengan user bertype admin.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
UIAdmin	*public*	*merupakan constructor yang langsung membuat object Controler*
tampilkanHalamanAwal	*public*	*Menampilkan halaman awal sealigus statistik*
TampilkanKategori	*Public*	*MenampilkanDaftarKategori*
tampilkanGaleriAplikasi	*Public*	*MenampilkanDaftar Galeri*
tampilkanDokumenAplikasi	*Public*	*Menampilkan Daftar Dokumen Aplikasi*
TampikanFormAplikasi	*Public*	*Menampilkan Form Aplikasi*
Tampilkan Form User	*Public*	*Menampilkan Form User*
tampilkanformPage	*Public*	*Menampilkan form page*

5. Class Controler
   Class Controler merupakan class control yang berisi prosedur-prosedur alud data sistem.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Controler	*public*	*merupakan constructor yang langsung membuat object Database*
InisialisasiDb	*public*	*Untuk membuat object database*
Nama Atribut	***Visibility***	***Tipe***

	(private, public)	
Alert	*Public*	*Array*
Db	*Protected*	*Database*

6. Class ControlerLoadListData
   Class ControlerListData merupakan control turunan dari class Controler yang berisi prosedur prosedur untuk mengambil data dari database melewati class Database.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
getlistaplikasi	*public*	*Mengambil data aplikasi dari database dan mengembalikan daftar aplikasi berupa array dari object aplikasi*
getdetailAplikasi	*public*	*Untuk mengambil detail aplikasi, function ini mengembalikan object aplikasi*
getListUser	*Public*	*Mengambil data user dari database dan mengembalikan daftar aplikasi berupa array dari object user*
getlistKategori	*Public*	*Mengambil data kategori dari database dan mengembalikan daftar kategori berupa array dari object kategori*
Getlistdokumen	*Public*	*Mengambil data dokumen dari database dan mengembalikan daftar dokumen berupa array dari object dokumen*
Getlistgaleri	*Public*	*Mengambil data galeri dari database dan mengembalikan daftar galeri berupa array dari object galeri*
Getpage	*Public*	*Untuk mengambil data page dari database, function ini mengembalikan object pages*

7. Class ControlerSession

Class ControlerSession merupakan control turunan dari class Controler yang berisi prosedur prosedur untuk memberikan hak akses login untuk user.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
ControlerSession	*public*	*Merupakan constructor yang langsung membuat object database*
Login	*Public*	*Untuk melakukan validasi login user*
Logout	*Public*	*Untuk menghapus session aktif user*

8. Class ControlerStatistik

Class ControlerStatistik merupakan control turunan dari class Controler yang berisi prosedur prosedur untuk mengambil informasi statistic konjungan web, jumlah data aplikasi, dan jumlah user.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
ControlerStatistik	*public*	*Merupakan constructor yang langsung membuat object database*
getStatUser	*Public*	*Mengambil banyak user berdasarkan jenis user*
getStatPengunjung	*Public*	*Mengambil data banyaknya kunjungan hari ini dan total kunjungan*
getStatAplikasi	*Public*	*Mengambil banyaknya data aplikasi berdasarkan kategori ataupun tidak memiliki kategori*

9. Class ControlerOlahData

Class ControlerOlahData merupakan control turunan dari class Controler yang berisi prosedur prosedur untuk mengolah data.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
ControlerOlahData	*public*	*Merupakan constructor yang langsung membuat object database*
Tambahkategori	*Public*	*Menyimpan kategori baru*
Hapuskategori	*Public*	*Menghapus kategori*
Tambahgaleri	*Public*	*Menyimpan galeri baru*
Tambahdokumen	*Public*	*Menyimpanpan dokumen baru*
hapusDokumen	*Public*	*Menghapus dokumen*
tambahAplikasi	*Public*	*Menambah Aplikasi*
Editaplikasi	*Public*	*Untuk edit aplikasi*
Hapusaplikasi	*Public*	*Untuk menghapusaplikasi*
Edituser	*Public*	*Untuk mengedit user*
Tambahuser	*Public*	*Untuk menambah user baru*
Hapususer	*Public*	*Untuk menghapus user*
Edit page	*Public*	*Untuk mengedit page*

10. Class Text

Class Text merupakan class control yang berisi prosedur-prosedur untuk kepentingan text / string di dalam sistem.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
filterSQLI	*public*	*merupakan procedure untuk memfilter kutip tunggal untuk keaman pengoprasian query database*
encrypt	*public*	*merupakan prosedur untuk mengencrypt sebuah text*
filterSebagianHTHML	*public*	*menghapus sebagian tag-tag HTML yang terdapat pada text*
filter	*public*	*menghapus tag-tag HTML yang terdapat pada text*
potongText	*public*	*untuk memotong text dengan panjang karakter tertentu.*

11. Class Database

Class Database merupakan class control yang berisi prosedur-prosedur untuk pengaksesan database.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Database	*public*	*merupakan constructor*
bukaKoneksi	*public*	*prosedur untuk koneksi ke database*
query	*public*	*untuk mengoprasikan query*
select	*public*	*Untuk membuat query select*
insert	*public*	*Untuk membuat query insert*
update	*public*	*Untuk membuat query update*

delete	*public*	*Untuk membuat query delete*
getResult	*public*	*Getter darai result*
getBaris	*public*	*untuk mengambil data baris*
jumlahBaris	*public*	*untuk mengambil jumbal baris*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
koneksi	*private*	*koneksi*
selectDb	*Private*	*Select db*
query	*private*	*string*
result	*private*	*result*
jumlahBaris	*private*	*integer*
dataBaris	*private*	*array*

12. Class Tabel

Class Tabel merupakan class control yang berisi prosedur-prosedur untuk menghungukan class entity dengan database.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Load	*public*	*Mengabil data dari database ke object*
Tambah	*public*	*Menyimpan data object kedatabase*
Edit	*public*	*Untuk menyamakan object dengan data yang ada di database*
Hapus	*public*	*Untuk menghapus data di database yang sesuai dengan objectnya*

13. Class Aplikasi

Class Aplikasi merupakan class entity sebagai entity aplikasi.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Aplikasi	*public*	*merupakan constructor*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
id	*public*	*integer*
nama	*public*	*string*
gambarUtama	*public*	*string*
detail	*public*	*string*
versi	*public*	*string*
tanggalRilis	*public*	*Date*
url	*public*	*string*
pembangun	*public*	*string*
pemilik	*public*	*string*
kategori	*public*	*Array Kategori*
galeri	*public*	*Array Galeri*
dokumen	*public*	*Array Dokumen*
statushapus	*public*	*boolean*

14. Class Kategori

Class Kategori merupakan class entity sebagai entity kategori.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Kategori	*public*	*merupakan constructor*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
id	*public*	*integer*
nama	*public*	*string*
parent	*public*	*integer*
statushapus	*public*	*boolean*

15. Class Dokumen

Class dokumen merupakan class entity sebagai entity dokumen.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Dokumen	*public*	*merupakan constructor*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
id	*public*	*integer*
Idaplikasi	*Public*	*integer*
nama	*public*	*string*
keterangan	*public*	*string*
linkFile	*public*	*String*

statushapus	*public*	*boolean*

16. Class Galeri

Class galeri merupakan class entity sebagai entity galeri.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Galeri	*public*	*merupakan constructor*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
id	*public*	*integer*
idaplikasi	*Public*	*integer*
keterangan	*public*	*string*
linkFile	*public*	*String*
statushapus	*public*	*boolean*

17. Class Pages

Class pages merupakan class entity sebagai entity pages.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
Pages	*public*	*merupakan constructor*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
id	*public*	*integer*
judulseo	*Public*	*integer*
judul	*public*	*string*
Konten	*public*	*String*

head	*public*	*String*
js	*public*	*string*
statushapus	*public*	*boolean*

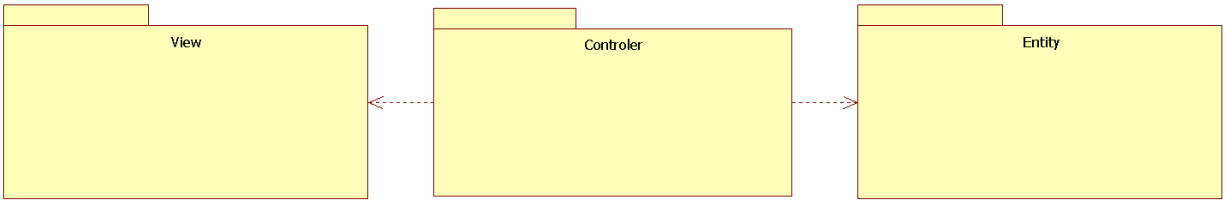
18. Class User

Class user merupakan class entity sebagai entity user.

Nama Operasi	***Visibility*** ***(private, public)***	***Keterangan***
User	*public*	*merupakan constructor*
Nama Atribut	***Visibility*** ***(private, public)***	***Tipe***
username	*private*	*string*
password	*private*	*string*
typeuser	*private*	*String*
statushapus	*public*	*boolean*

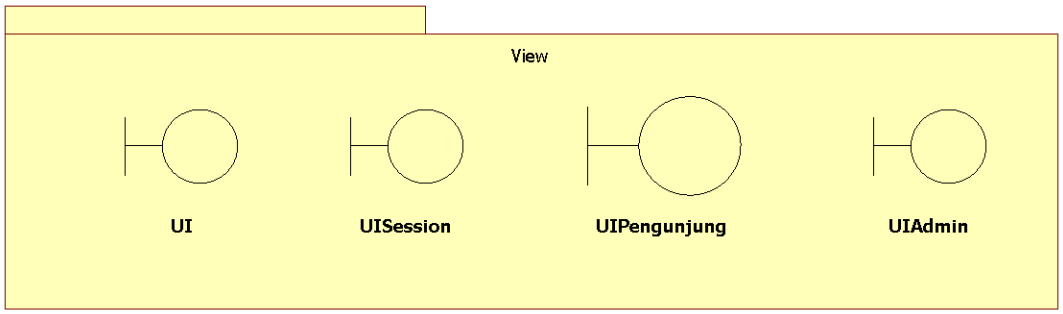
3.1.4.7 Diagram Package

Package diagram merupakan pengelompokan kesamaan jenis class untuk memudahkan dalam pengorganisasian file. Berikut adalah package diagram untuk Web Profil Aplikasi Corporate PLN.



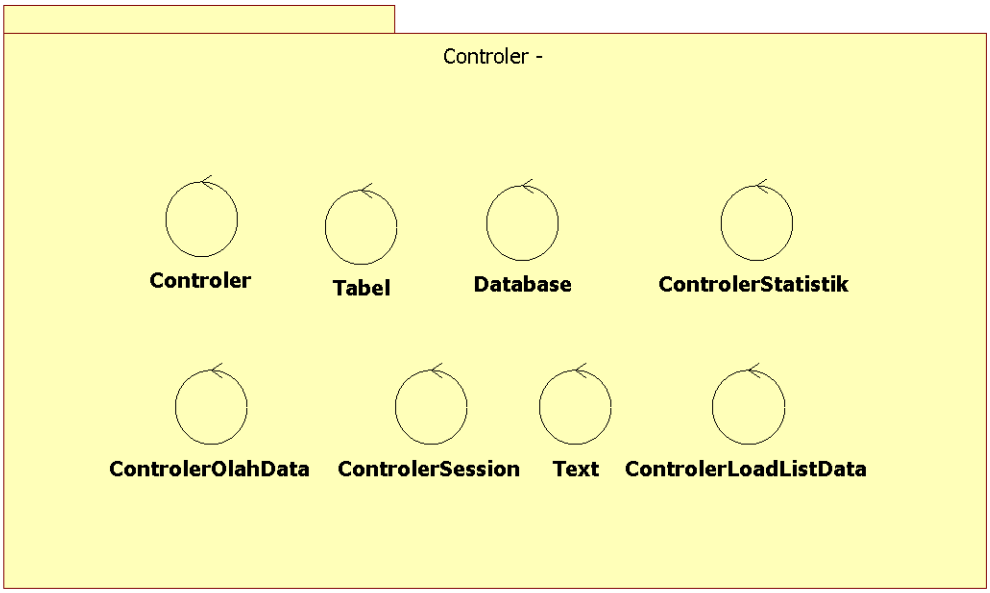
1. View Package

Package view atau boundary berisi class-class view yang digunakan untuk menampilkan interface yang dapat berupa tampilan data atau form.



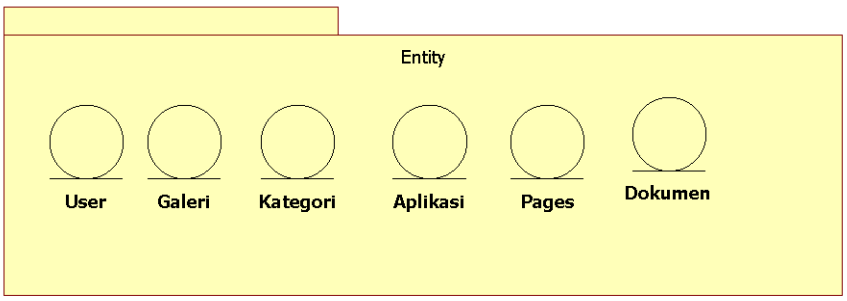
2. Controller Package

Package controllers atau control berisi class-class control yang merupakan pengatur dalam setiap proses yang ada.



3. Entity Package

Package model atau entities berisi class-class model yang mengakses langsung pada database untuk dapat mengolah data.

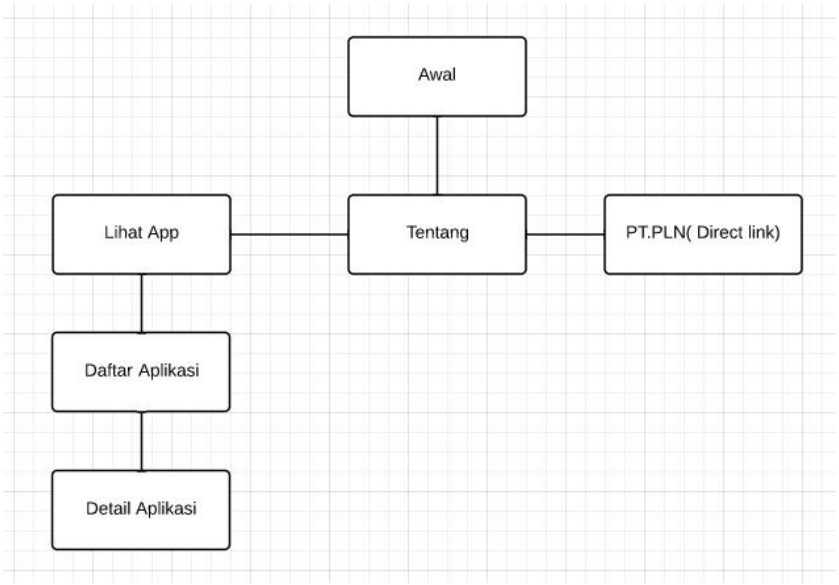


3.2 Perancangan Menu

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

3.2.1 Struktur Menu FrontEnd

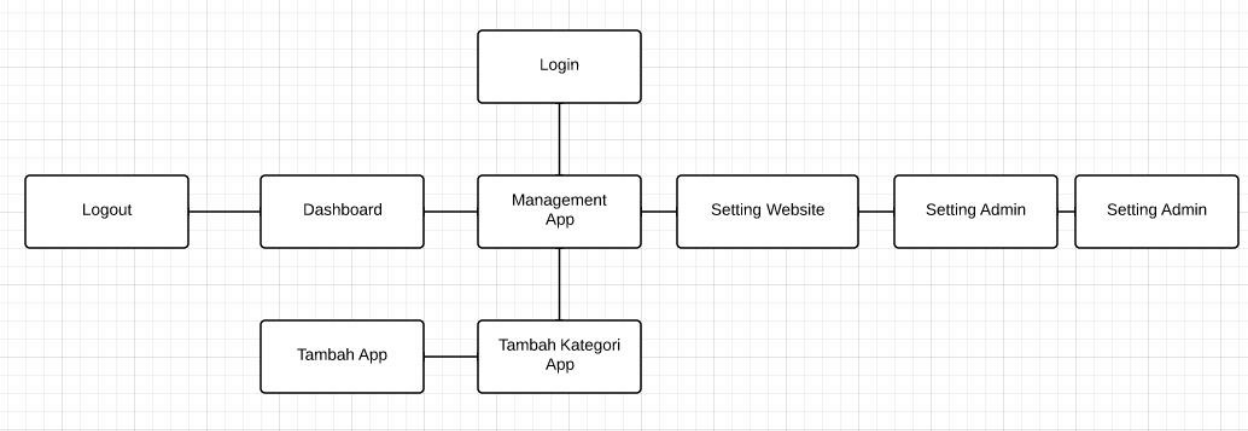
Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi user pengguna website ini. Perancangan struktur menu dari aplikasi ini dapat dilihat pada gambar.



Gambar 3.2.1.1 Struktur Menu Frontend

3.2.2 Struktur Menu BackEnd

Sedangkan struktur menu pada BackEnd dapat dilihat pada gambar 3.18



Gambar 3.2.2.1 Struktur Menu Backend

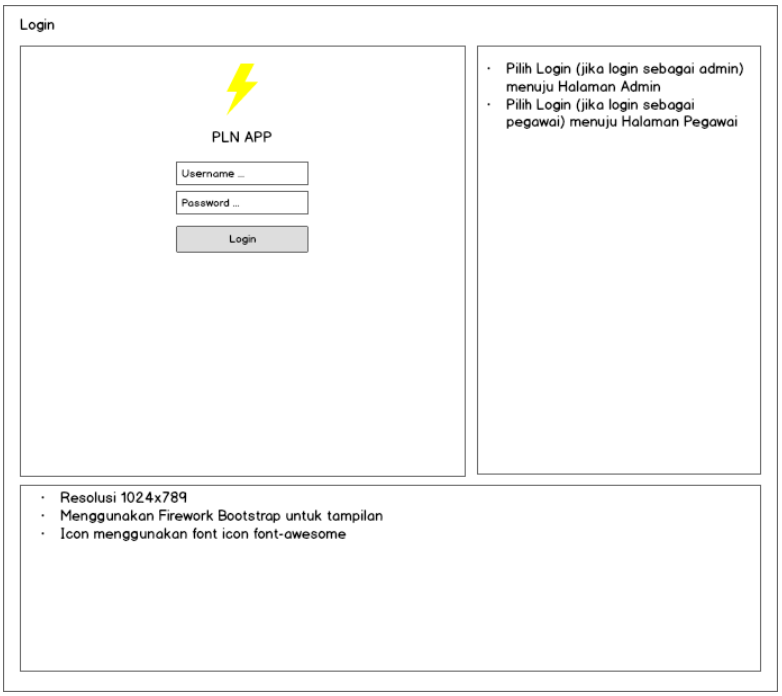
3.3 Perancangan Antarmuka Sistem

Tahap perancangan antarmuka sistem bertujuan untuk mencari bentuk optimal dari aplikai yang akan dibangun dengan pertimbangan factor-faktor permasalahan dan kebutuhan uang ada pada sistem seperti yang telah ditetapkan pada tahap analisis.

Perancangan antarmuka sisetm terbagi menjadi 2 bagian yaitu antarmuka backend dan frontend.

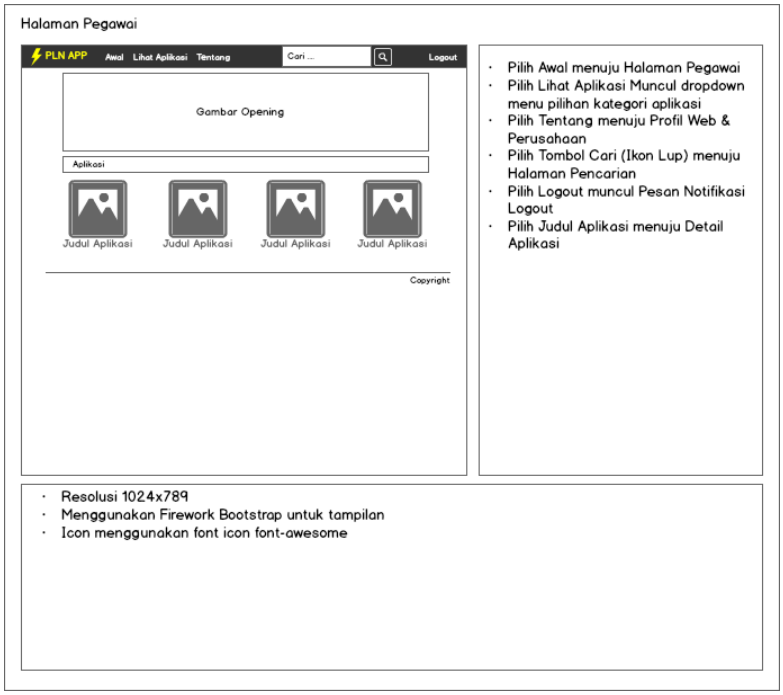
3.3.1 Perancangan Antarmuka Frontend

3.3.1.1 Login



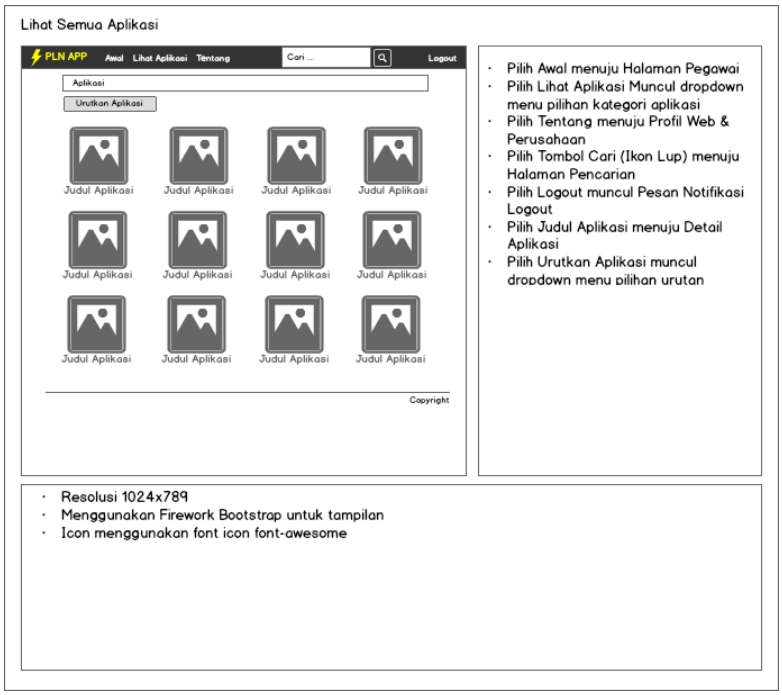
Gambar 3.3.1.1. Login

3.3.1.1 Halaman Awal Pegawai



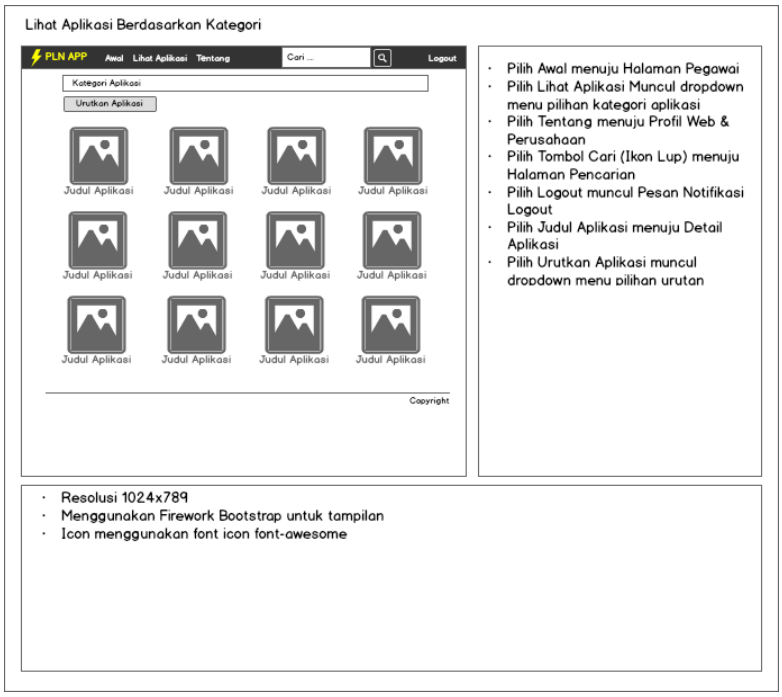
Gambar 3.3.1.2. Halaman Awal Pegawai

3.3.1.2 Halaman Lihat Semua Aplikasi



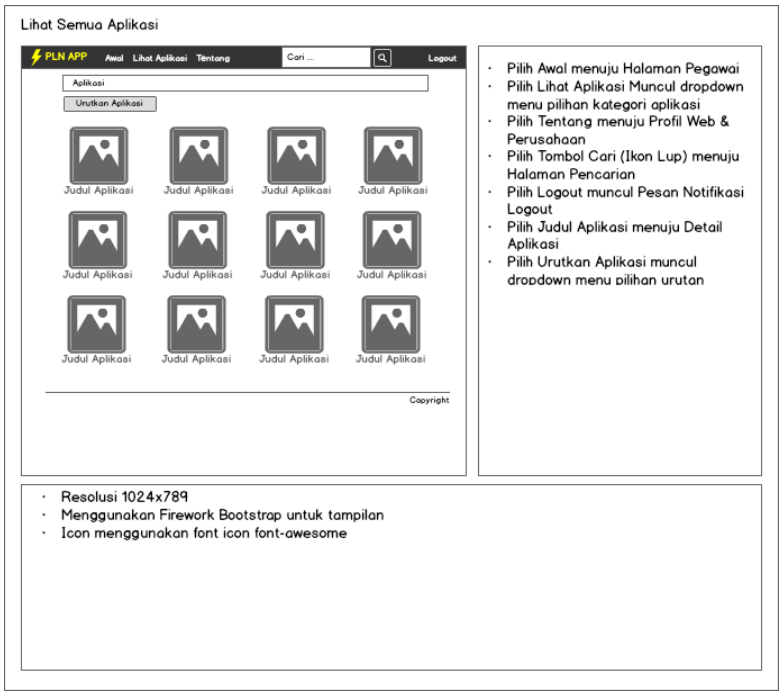
Gambar 3.3.1.3. Halaman Lihat Semua Aplikasi

3.3.1.3 Halaman Kategori Aplikasi



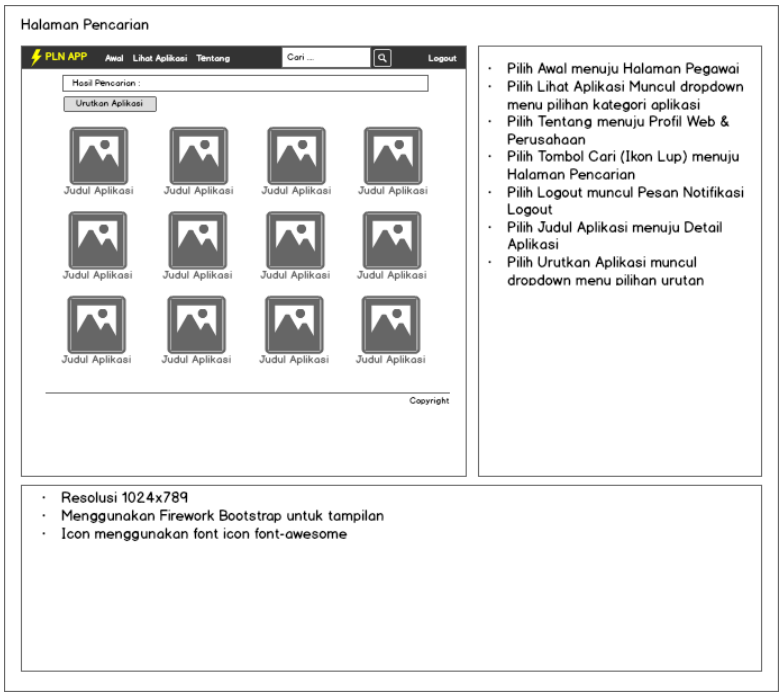
Gambar 3.3.1.4. Halaman Kategori Aplikasi

3.3.1.4 Halaman Lihat Semua Aplikasi



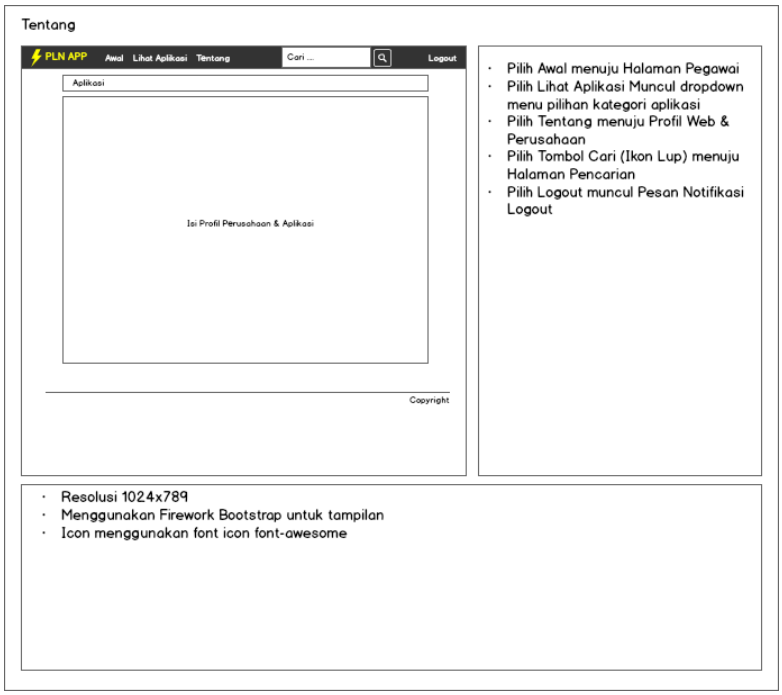
Gambar 3.3.1.5. Halaman Lihat Semua Aplikasi

3.3.1.5 Halaman Pencarian



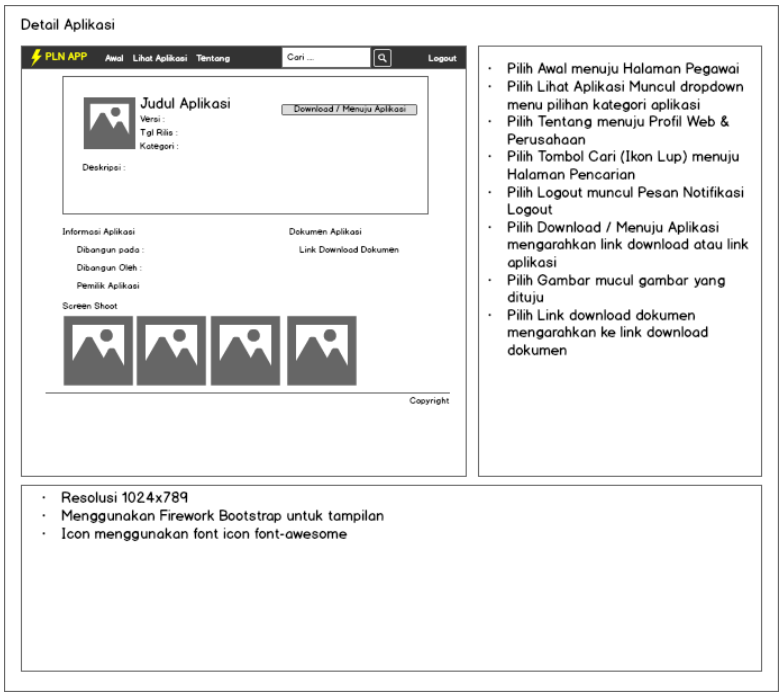
Gambar 3.3.1.6. Halaman Pencarian

3.3.1.6 Halaman Tentang (Profile Web & Perusahaan)



Gambar 3.3.1.7. Halaman Tentang (Profile Web & Perusahaan)

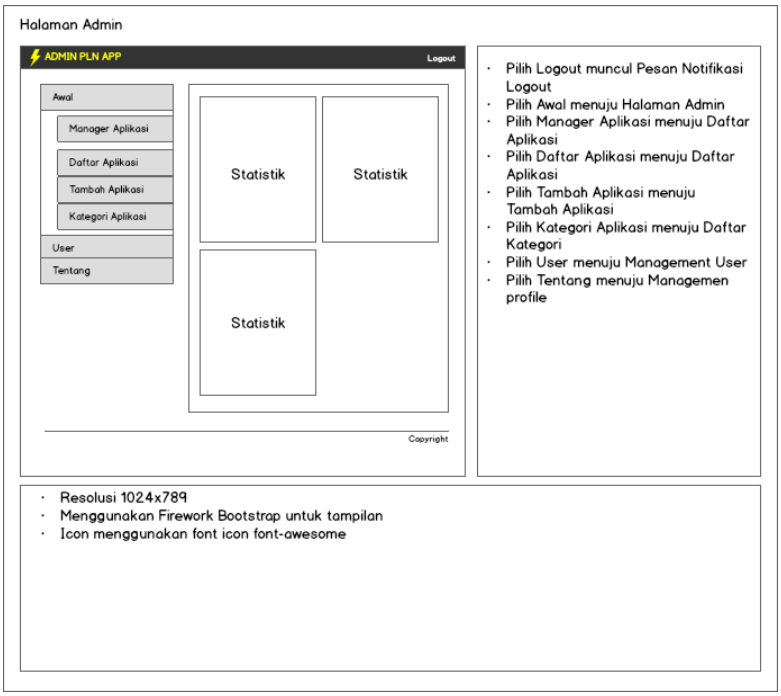
3.3.1.7 Halaman Detail Aplikasi



Gambar 3.3.1.8. Halaman Detail Aplikasi

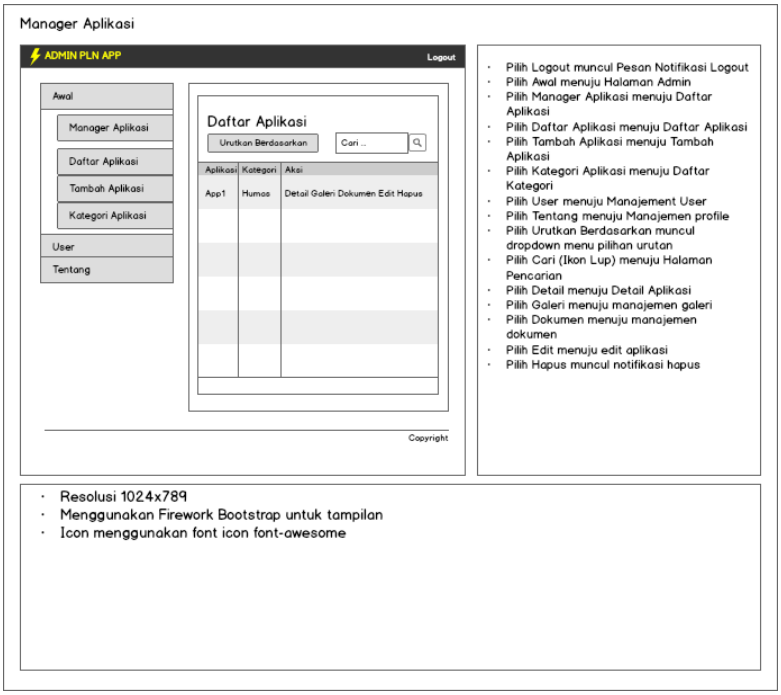
3.3.2 Perancangan Antarmuka Backend

3.3.2.1. Awal



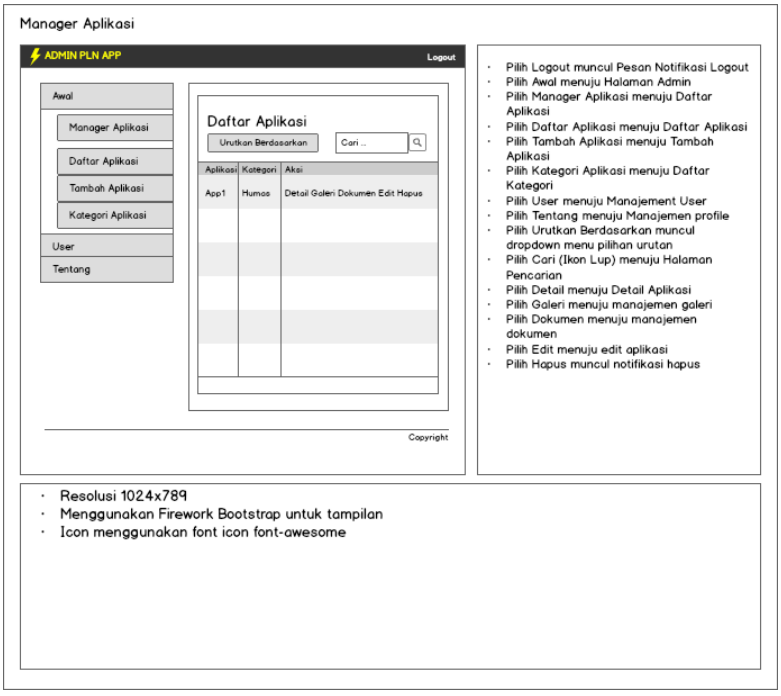
Gambar 3.3.2.1. Halaman Awal

3.3.2.2. Manager Aplikasi



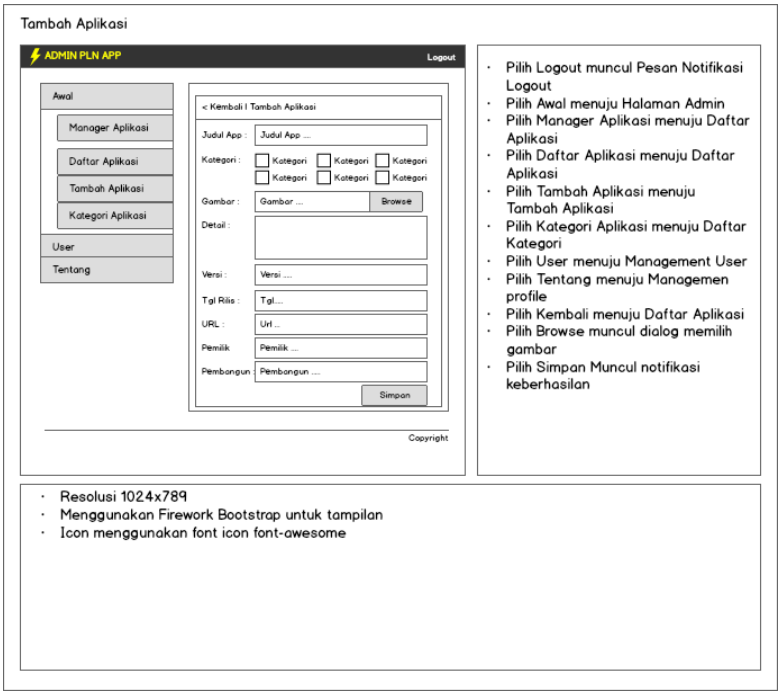
Gambar 3.3.2.2. Halaman Manager Aplikasi

3.3.2.3. Daftar Aplikasi



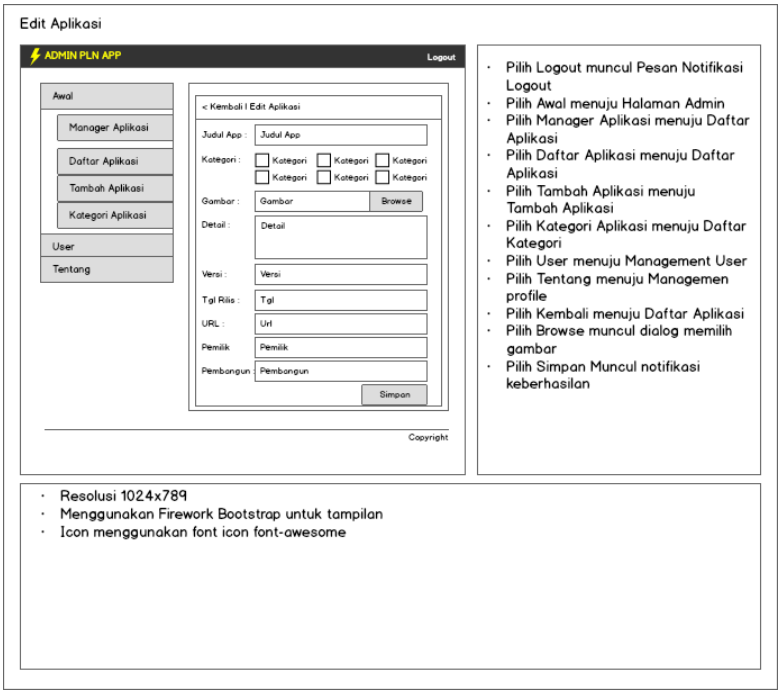
Gambar 3.3.2.3. Halaman Daftar Aplikasi

3.3.2.4. Tambah Aplikasi



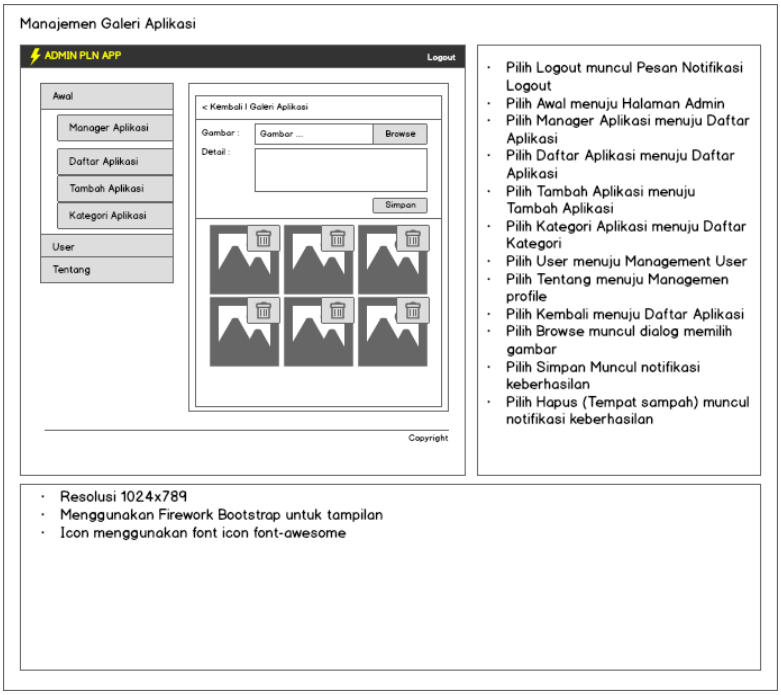
Gambar 3.3.2.4. Halaman Tambah Aplikasi

3.3.2.5. Edit Aplikasi



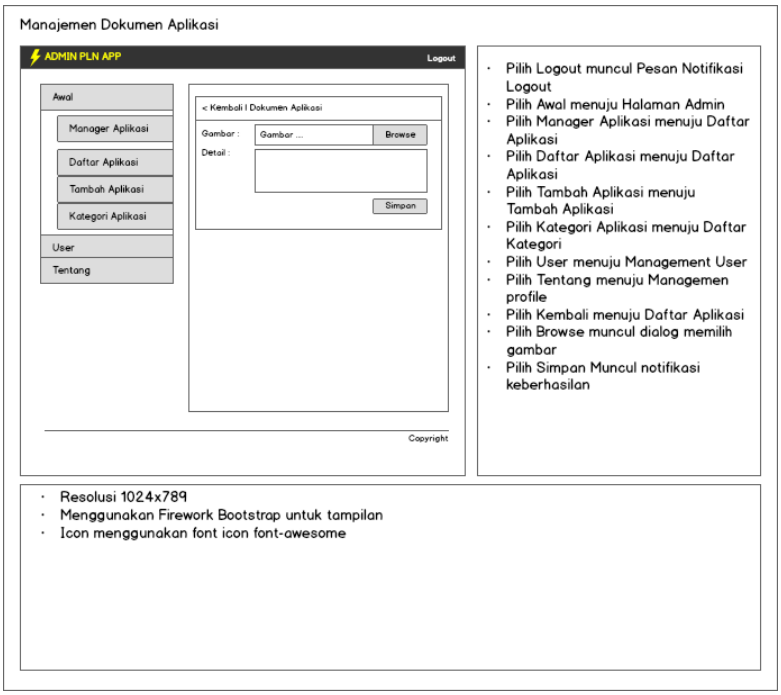
Gambar 3.3.2.5. Halaman Edit Aplikasi

3.3.2.6. Manajemen Galeri Aplikasi



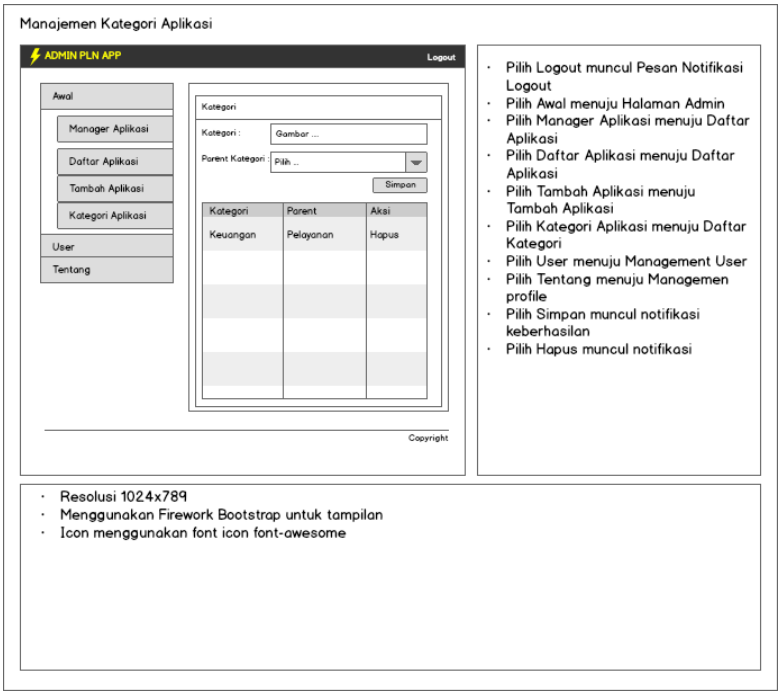
Gambar 3.3.2.6. Halaman Galeri Aplikasi

3.3.2.7. Manajemen Dokumen Aplikasi



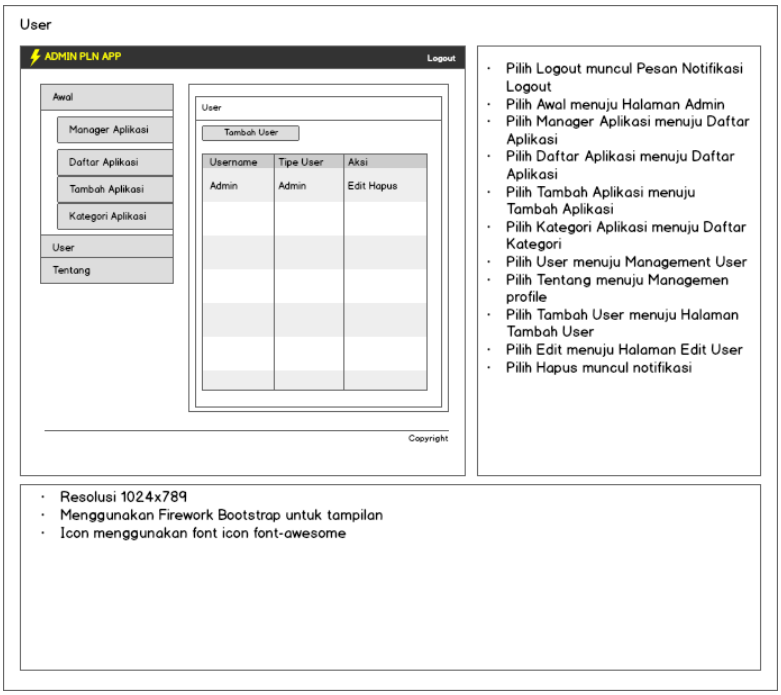
Gambar 3.3.2.7. Halaman Dokumen Aplikasi

3.3.2.8. Kategori



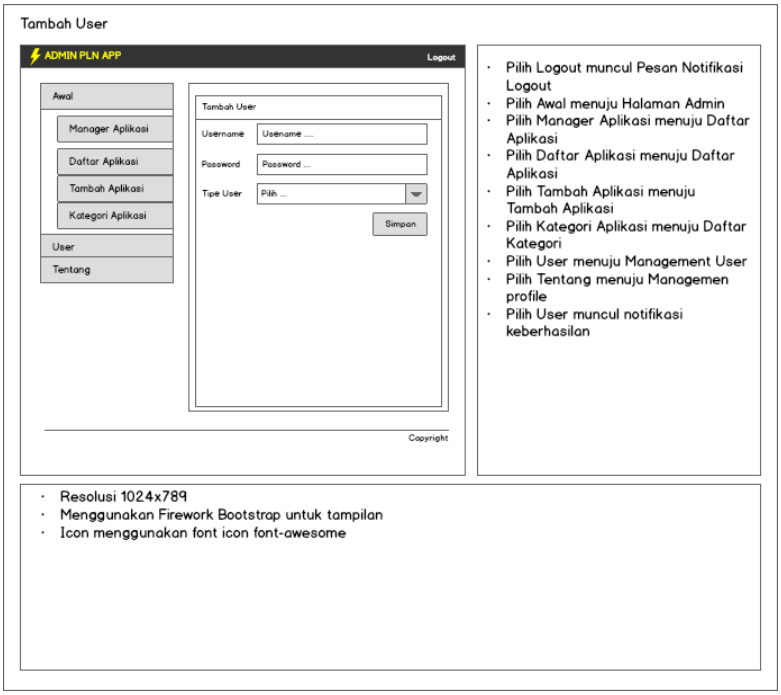
Gambar 3.3.2.8. Halaman Kategori Aplikasi

3.3.2.9. User



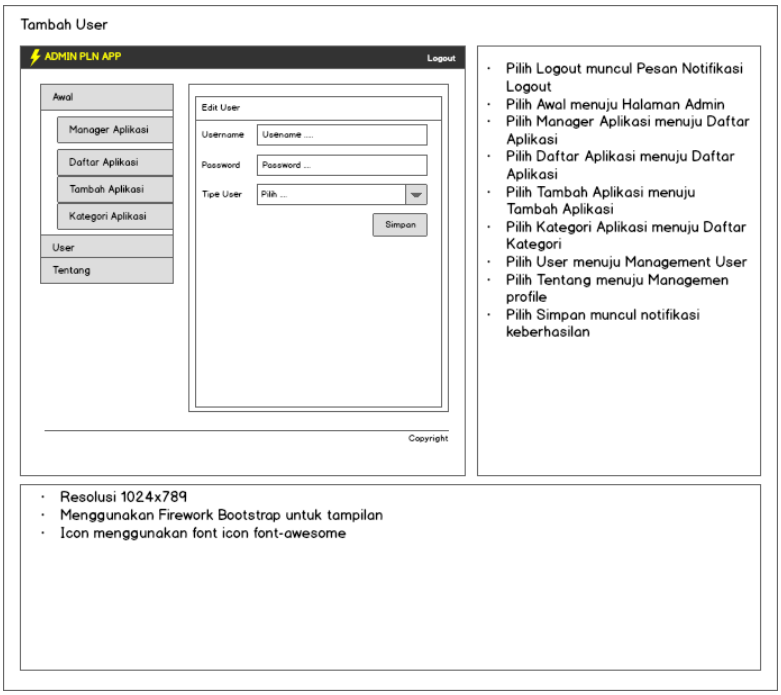
Gambar 3.3.2.9. Halaman User

3.3.2.10. Tambah User



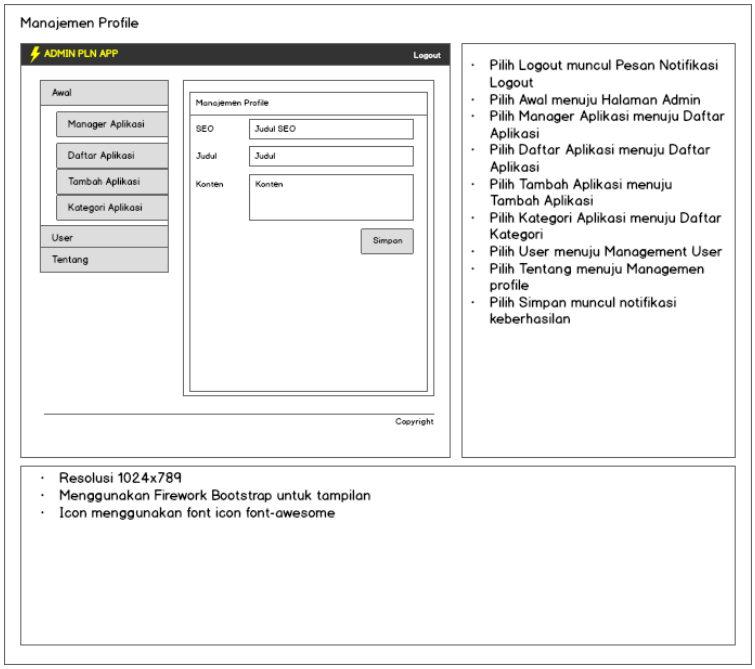
Gambar 3.3.2.10. Halaman Tambah User

3.3.2.11. Edit User



Gambar 3.3.2.11. Halaman Edit User

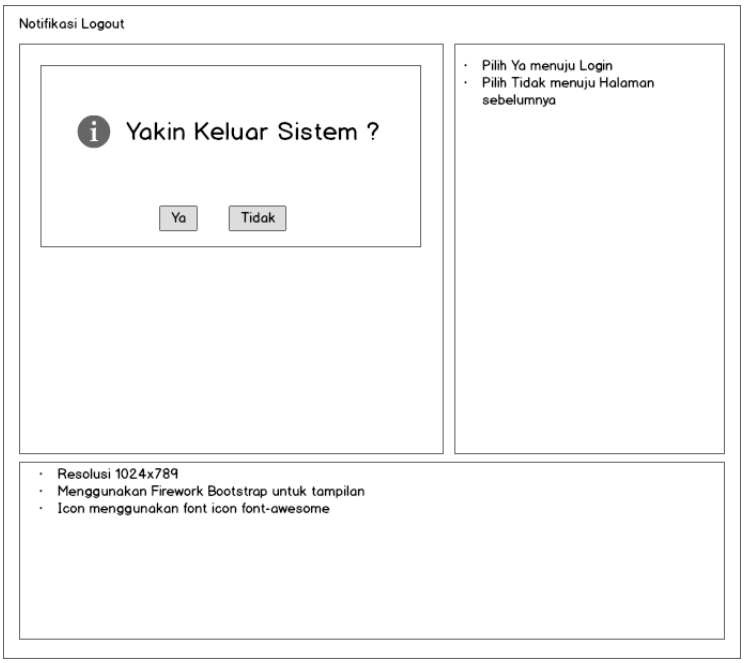
3.3.2.12. Manajemen Profile



Gambar 3.3.2.12. Halaman Manajemen Profile

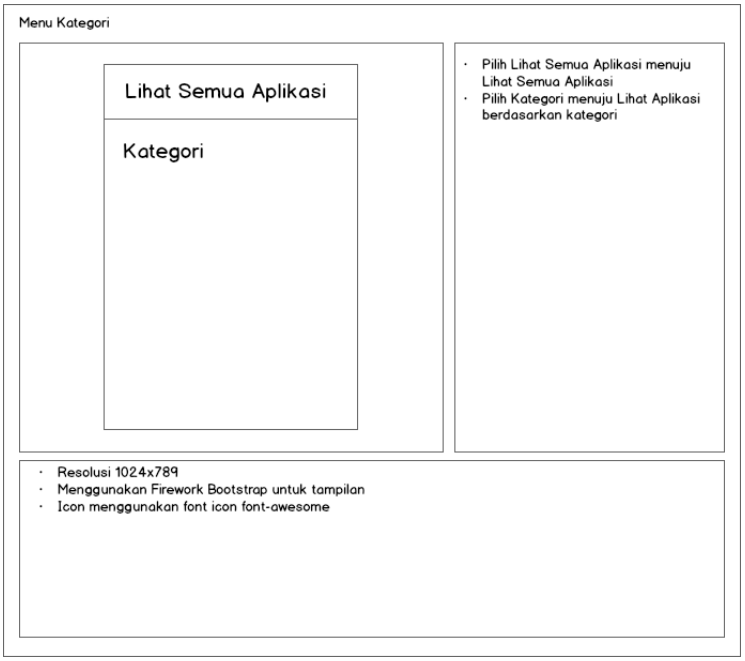
3.3.3 Antarmuka Pesan

3.3.3.1 Notifikasi Logout



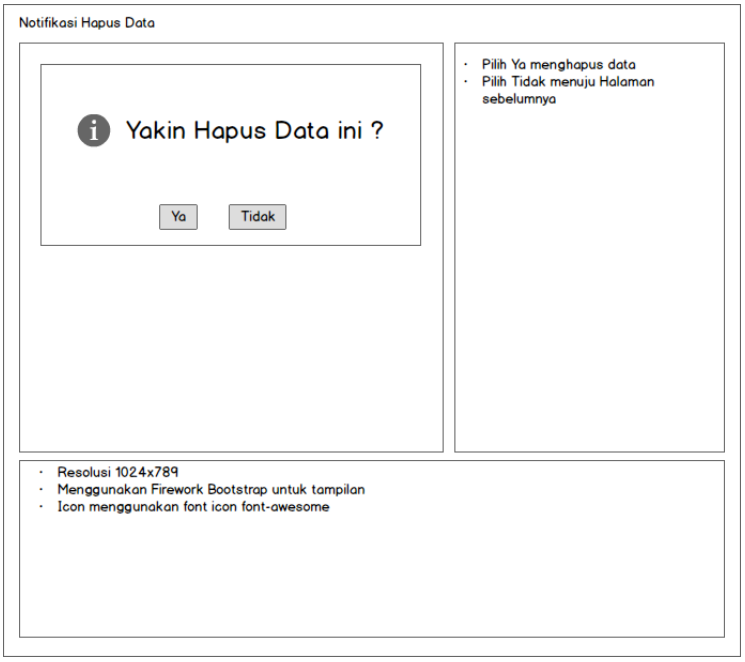
Gambar 3.3.3.1. Notifikasi Pesan Keluar Sistem (logout)

3.3.3.1 Menu Kategori



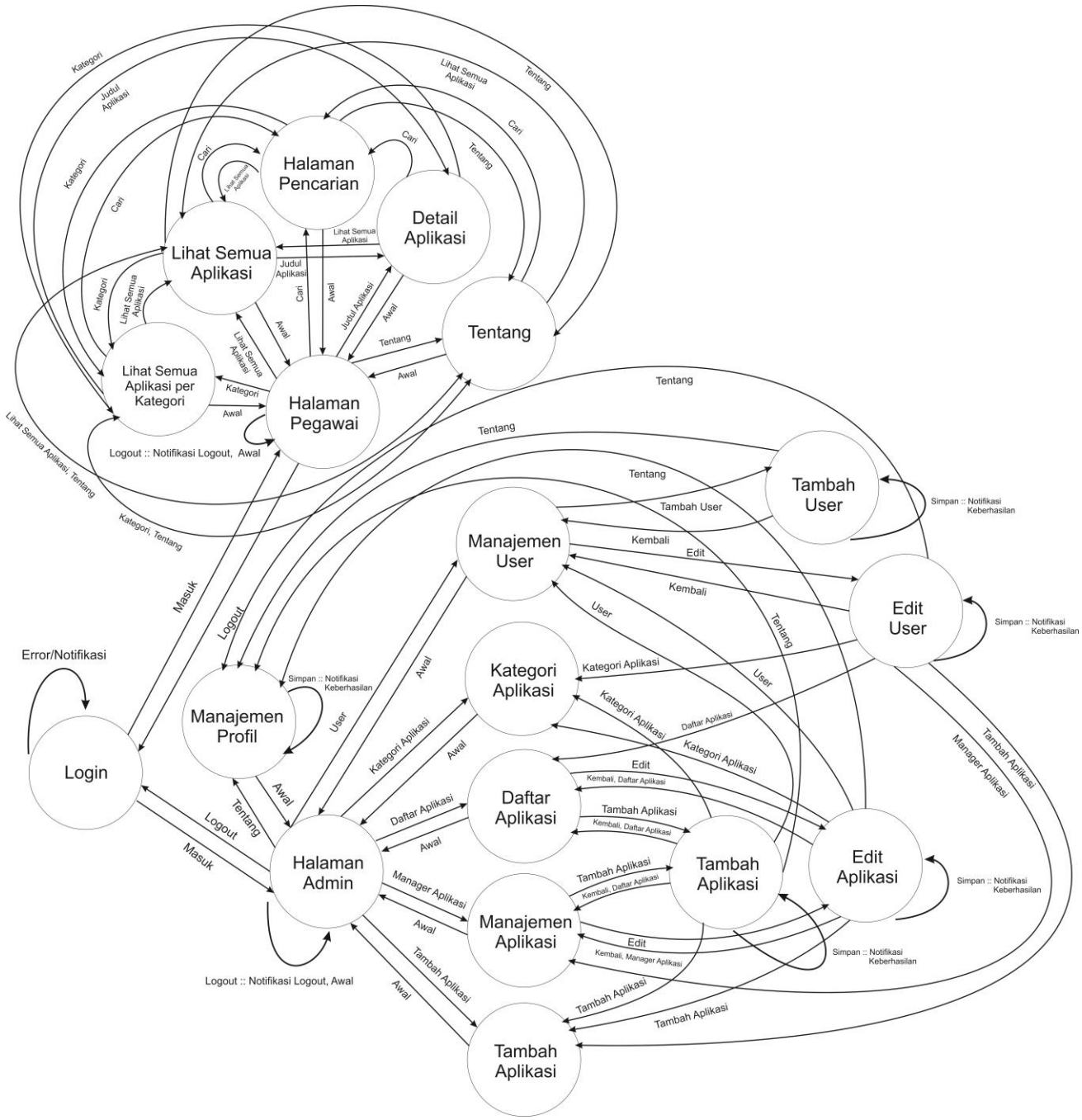
Gambar 3.3.3.2. Menu Kategori

3.3.3.2 Menu Kategori



Gambar 3.3.3.3. Notifikas Hapus Data

3.3.4 Jaringan Semantik Antarmuka



Gambar 3.3.4. Jaringan Semantik Antarmuka