BAB 3

ANALISIS DAN PERANCANGAN

3.1. Analisis Sistem

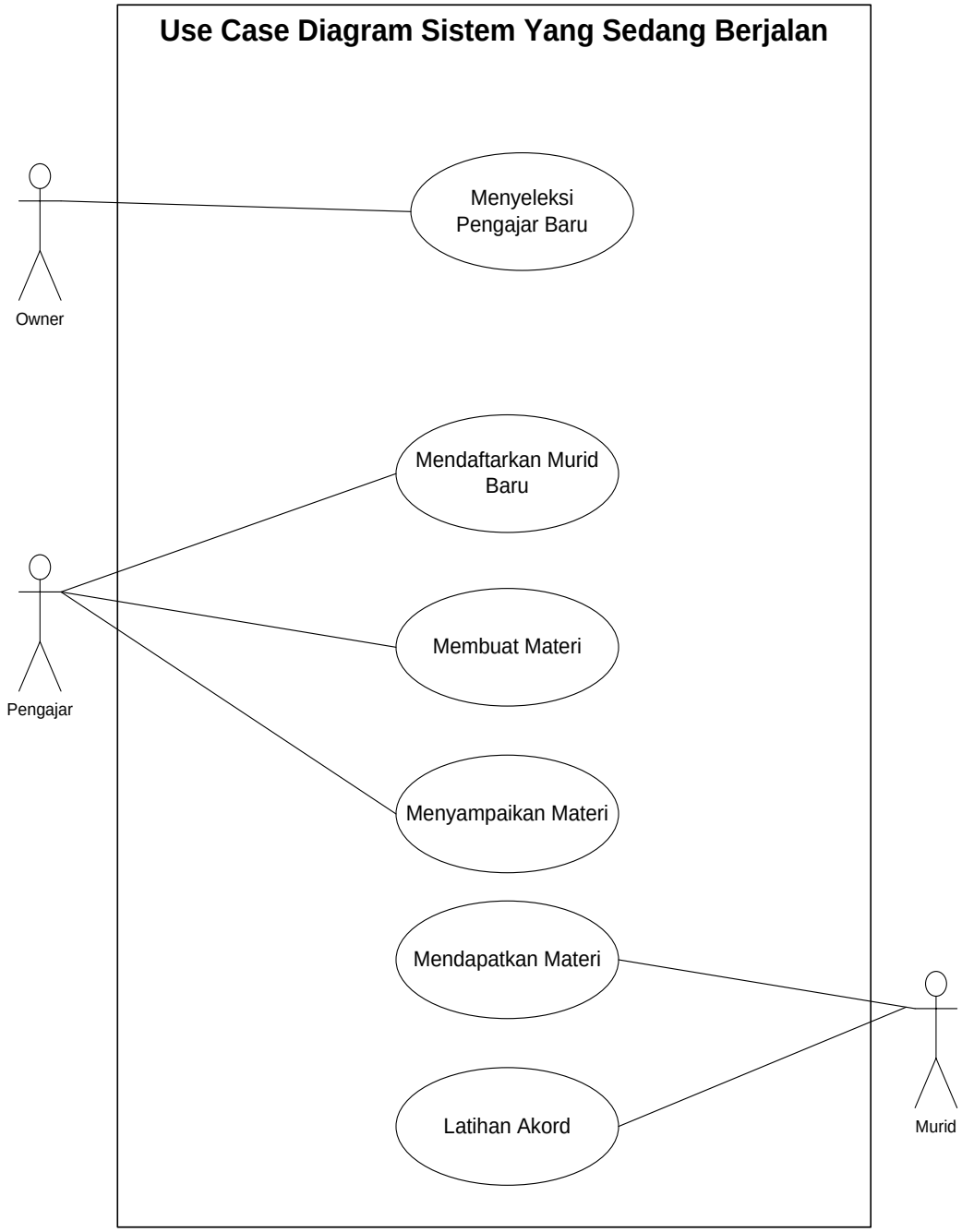
Pada bab ini dilakukan analisis sistem yang sedang berjalan dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi, dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya.

3.1.1. Analisis Sistem Berjalan

Di Gentra Rineka kegiatan pembelajaran dilakukan dengan privat antara satu pengajar dengan satu murid. Pelaksanaan kursus gitar di Gentra Rineka dilakukan dua kali pertemuan dalam satu minggu. Satu pertemuan pembelajaran privat tersebut memiliki waktu yang cukup singkat yaitu satu jam. Hal tersebut membuat pengajarpun menyarankan murid untuk lebih banyak berlatih di luar jam kursus.

Sistem yang berjalan di Gentra Rineka memiliki beberapa kendala yaitu keterbatasan waktu, kecanggungan seorang murid terhadap pengajar, kesulitan murid dalam menentukan benar atau salah pada saat pembelajaran, dan kesulitan dalam mensetting gitar. Solusi yang diambil dalam menyelesaikan kendala tersebut adalah dibuatnya aplikasi pengenalan dan pembelajaran akord gitar berbasis *web* sebagai salah satu media pembelajaran di luar jam kursus gitar Gentra Rineka.

Analisis yang sedang berjalan di Gentra Rineka Production dapat dilihat pada diagram *use case* pada gambar 3.1:



Gambar 3.1 *Use Case* Diagram Sistem Yang Sedang Berjalan

3.2. Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional merupakan proses analisis yang memilah-milah antara kebutuhan perangkat keras, perangkat lunak, dan pengguna sebagai kebutuhan dalam pembuatan aplikasi.

3.2.1. Analisis Perangkat Keras

Analisis perangkat keras (*hardware*) merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan perangkat keras. Berdasarkan hasil analisis, kursus gitar Gentra Rineka menggunakan perangkat dengan spesifikasi sebagai berikut:

1. *Processor* intel core 2 duo 2.0 GHz.
2. RAM 1 GB.
3. VGA Ati Radeon 128 MB
4. *Soundcard*
5. Gitar elektrik dengan *line jack* kabel gitar.
6. Atau gitar akustik dengan menggunakan mikrofon.

Sedangkan kebutuhan minimal perangkat keras yang diperlukan untuk menjalankan aplikasi ini adalah :

1. *Processor* Pentium IV 2.0 GHz.
2. RAM 512 MB.
3. VGA *On Board* 64 MB.
4. *Sound Card.*

5. Gitar elektrik dengan *line jack* kabel gitar.
6. Atau gitar akustik dengan menggunakan mikrofon.
7. Modem internet 3.1 Mbps.

Berdasarkan hasil perbandingan dari perangkat keras yang ada, Gentra Rineka telah memiliki beberapa perangkat keras yang sesuai dengan kebutuhan yang diperlukan untuk menjalankan aplikasi ini, akan tetapi masih diperlukannya sebuah perangkat keras yang memiliki fasilitas internet, yaitu modem internet.

3.2.2. Analisis Perangkat Lunak

Analisis perangkat lunak (*software*) merupakan proses analisis yang lebih menekankan kepada aspek pemanfaatan perangkat lunak. Berdasarkan hasil analisis, kursus gitar Gentra Rineka memiliki perangkat lunak sebagai berikut:

1. Sistem operasi *Windows 7.*
2. Adobe Photoshop CS4.
3. Aplikasi *browser* (seperti Mozilla Firefox, dan Google Chrome).

Perangkat lunak yang dibutuhkan untuk membangun aplikasi ini adalah:

1. *Java Development Kit* (JDK), sebagai sistem kompilasi kode-kode program.
2. *Java Runtime Edition*, sebagai *platform* untuk menjalankan aplikasi.
3. *Wampserver*, sebagai *webserver* untuk sistem *website*.
4. NetBeans IDE, sebagai perangkat lunak yang digunakan untuk membangun aplikasi.

5. Adobe Dreamweaver beserta *license*, sebagai perangkat lunak yang digunakan untuk membangun *website*.
6. Adobe Photoshop beserta *license*, sebagai perangkat lunak yang digunakan untuk edit gambar-gambar yang akan digunakan.
7. Aplikasi *browser* (seperti Mozilla Firefox, dan Google Chrome) sebagai perambah untuk pengguna.

Berdasarkan hasil perbandingan perangkat lunak yang dibutuhkan untuk menjalankan aplikasi ini adalah:

1. *Java Development Kit* (JDK), sebagai sistem kompilasi kode-kode program.
2. *Java Runtime Edition* (JRE), sebagai *platform* untuk menjalankan aplikasi.
3. *License* Adobe Photoshop CS4.
4. *Web Hosting* disertai nama domain sebagai layanan jasa penyewaan tempat di internet.

3.2.3. Analisis Pengguna

Analisis pengguna (*user*) merupakan proses analisis yang lebih menekankan kepada aspek pengguna aplikasi. Berdasarkan hasil analisis, kursus gitar Gentra Rineka memiliki tiga pengguna, karakteristik pengguna di kursus Gentra Rineka dapat dilihat pada tabel 3.1:

Tabel 3.1 Karakteristik Pengguna di Kursus Gitar Gentra Rineka

Pengguna	Tanggung Jawab	Tingkat Pendidikan	Tingkat Keterampilan	Pengalaman
Owner	Melakukan Pendataan	Lulus SMA	- Memiliki kemampuan dalam bermain gitar - Dapat mengoperasikan komputer	-
Pengajar	Melakukan kegiatan mengajar	Lulus SMA	- Memiliki kemampuan dalam bermain gitar - Dapat mengoperasikan komputer	-
Murid	Melakukan kegiatan pembelajaran	Apa Saja	- Memiliki kemampuan dalam bermain gitar - Dapat mengoperasikan komputer	-

Untuk menggunakan sistem yang dibangun terdapat beberapa karakteristik yang harus dimiliki oleh pengguna (user), kebutuhan karakteristik dapat dilihat pada tabel 3.2:

Tabel 3.2 Karakteristik Pengguna Sistem

Pengguna	Tanggung Jawab	Hak Akses	Keterampilan
Owner (Super Admin)	Melakukan olah data	Mengolah data pengajar	- Dapat mengoperasikan komputer dengan baik - Mengerti dalam proses mengelola data
Pengajar (admin)	Melakukan olah data	Mengolah data murid, data materi	- Dalam mengoperasikan komputer - Mengerti dalam proses mengelola data

Murid	Melakukan kegiatan pembelajran	Menggunakan aplikasi pembelajaran dengan berinteraksi langsung dengan *hardware* (gitar)	- Dapat mengoperasikan komputer - Dapat mensetting audio komputer - Dapat bermain gitar
Pengunjung	Melihat Halaman Muka	Melihat daftar materi	- Dapat mengoperasikan komputer - Dapat mensetting audio komputer - Dapat bermain gitar

Berdasarkan hasil perbandingan karakteristik pengguna yang dibutuhkan untuk menggunakan aplikasi ini dapat dilihat pada tabel 3.3:

Tabel 3.3 Karakteristrik Pengguna Sistem Yang Dibutuhkan

Pengguna	Tanggung Jawab	Hak Akses	Pelatihan
Owner (Super Admin)	Melakukan olah data	Mengolah data pengajar	- Pelatihan dalam mengoperasikan komputer - Pelatihan dalam mengelola data
Pengajar (admin)	Melakukan olah data	Mengolah data murid, data materi	- Pelatihan dalam mengoperasikan komputer - Pelatihan dalam mengelola data
Murid	Melakukan kegiatan pembelajran	Menggunakan aplikasi pembelajaran dengan berinteraksi langsung dengan *hardware* (gitar)	- Pelatihan dalam mengoperasikan komputer - Pelatihan dalam men mensetting audio komputer - Pelatihan dalam bermain gitar

Pengunjung	Melihat Halaman Muka	Melihat daftar materi	-

3.3. Analisis Proses Sound Recognition

Sound recognition merupakan sebuah teknik dengan tujuan untuk memungkinkan mesin untuk mendengar dan memahami sebuah masukan dalam bentuk suara. Pada sistem yang dibangun terdapat proses *sound recognition* dalam pembuatan aplikasi pengenalan dan pembelajaran akord. Sebuah nada dikenali dengan besaran *spectrum* yang bermula dari masukan sebuah sinyal digital. Secara sederhana pendeteksian sebuah nada atau akord dapat dilihat pada gambar 3.2:



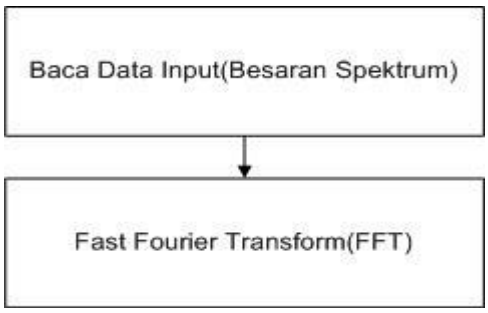
Gambar 3.2 Diagram Blok Sistem Untuk Proses *Sound Recognition*

3.3.1. Akusisi Data

Data input sinyal digital diambil dari suara gitar yang terhubungan dengan *sound card*. Suara gitar yang diambil hanya dalam bentuk sinyal digital yang dihasilkan dari gitar elektrik.

3.3.2. *Processing*

Processing pada tahap awal, yaitu dilakukannya baca data input berupa besaran spektrum dalam bentuk frekuensi dan dianalisis menggunakan *Fast Fourier Transform* (FFT). *Processing* dapat dilihat pada gambar 3.2:



Gambar 3.3 Transformasi Sinyal Digital Ke Dalam Besaran Frekuensi

Sinyal input digital dibawa ke dalam kawasan frekuensi dengan algotitma *Fast Fourier Transform. Fast Fourier Transform* (FFT) merupakan algoritma yang cukup efisien dalam pemrosesan sinyal digital (dalam hal ini nada) dalam bentuk diskrit. Konsep utama algoritma ini adalah mengubah sinyal suara berbasis waktu menjadi berbasis frekuensi dengan membagi masalah menjadi masalah yang lebih kecil, setiap masalah diselesaikan dengan cara melakukan pencocokan pola digital suara. Algoritma ini akan diimplementasikan pada bahasan mengenai proses *sound recognition*. Dalam *sound recognition* diperlukan dalam penentuan frekuensi yang memiliki fungsi periodik tertentu. Fungsi ini bisa mewakili sinyal digital atau audio. Setiap fungsi periodik periode P dapat dinyatakan sebagai rangkaian fungsi sin dan cos. Persamaan *fast fourier transform* dapat dilihat pada persamaan 3.1 dan 3.2.

$$f(t) = a_0 + \sum_{j=1}^{\infty} a_j \cos\left(\frac{2\pi j}{P} t\right) + b_j \sin\left(\frac{2\pi j}{P} t\right) =$$

$$= a_0 + a_1 \cos\left(\frac{2\pi}{P} t\right) + b_1 \sin\left(\frac{2\pi}{P} t\right) + a_2 \cos\left(\frac{4\pi}{P} t\right) + b_2 \sin\left(\frac{4\pi}{P} t\right) + \ldots \quad \ldots (3.1)$$

$$f(t) = \cos(t) + 4\sin(t) - 3\sin(2t) \quad \ldots (3.2)$$

Untuk menyelesaikan sebuah masalah memerlukan sebuah fungsi terurai diwakili oleh array panjang N dan periode P. Algoritma untuk menghitung FFT untuk $N=2^n$, Algoritma *Fast Fourier Transform* yang telah diimplentasikan kedalam bentuk bahasa C# dapat dilihat pada tabel 3.4.

Tabel 3.4 Algoritma *Fast Fourier Transform*

```
BitReverseData(data)
mmax = 1
while (n > mmax)
   istep = 2 * mmax
   theta = 3.14159265358979323846 / mmax
   wp = cos(theta) + i * sin(theta)
   w = 1
   for (m = 1; m <= mmax; m = m + 1)
      for (k = m - 1; k < n; k = k + istep)
         j = k + mmax
         temp = w * x[j]
         x[j] = x[k] - temp
         x[k] = x[k] + temp
      endfor
   w = w*wp
   endfor
   mmax = istep
endwhile
```

3.3.3. Analisis Frekuensi

Dalam analisis frekuensi ini, hanya menentukan nilai-nilai frekuensi yang terdapat dalam setiap segmen. Dari nilai frekuensi tersebut dapat ditentukan kemungkinan nada berdasarkan frekuensi yang telah ditetapkan. Frekuensi nada

pada umumnya telah ditetapkan, setiap nada memiliki frekuensi dengan besaran yang membedakan dengan nada lainnya. Setiap nada memiliki tingkatan tingginya sebuah nada yang disebut juga dengan oktaf. Contoh, Nada dasar A memiliki ketetapan frekuensi yaitu 110 Hz, nada A memiliki tingkatan oktaf mulai dari oktaf pertama yaitu 220 Hz hingga oktaf ke n. Tiap oktaf memiliki asumsi 2 x nilai oktaf A sebelumnya hingga oktaf ke n. Data frekuensi nada dapat dilihat pada tabel 3.5.

Tabel 3.5 Nada Dasar A-D

Nada Dasar A (Hz)	1	2	3	4	5	6
	A	A#	B	C	C#	D
A=110	110	117	123	131	139	147
A=220	220	233	247	262	277	294
A=440	440	466	494	523	554	587
A=880	880	932	988	1047	1109	1175

Data frekuensi nada dapat dilihat pada tabel 3.6 untuk nada dasar D#-G#.

Tabel 3.6 Nada D#-G#

Nada Dasar A (Hz)	7	8	9	10	11	12
	D#	E	F	F#	G	G#
A=110	156	165	175	185	196	208
A=220	311	330	349	370	392	415
A=440	622	659	698	740	784	831

A=880	1245	1319	1397	1480	1568	1661

3.3.4. Penentuan Nada/Akord

Dalam penentuan nada atau akord keluaran terdapat besaran frekuensi. Setiap nada memiliki frekuensi yang tetap sesuai dengan jenis nadanya sedangkan akord memiliki beberapa frekuensi yang membentuk sebuah nada. Pada setiap frekuensi nada dalam suatu rentang waktu tertentu ditentukan beberapa frekuansi nada utama yang dominan dibandingkan dengan lainnya. Jika terdapat tiga atau minimal dua nada yang bersesuaian, nada-nada tersebut dapat dikatakan sebagai akord.

3.4. Analisis Kebutuhan Fungsional

3.4.1. *Use Case Diagram*

Terdapat lima aktor dalam aplikasi pengenalan dan pembelajaran akord gitar ini, yaitu *owner* (superadmin), pengajar (admin), murid, gitar (*hardware*), dan pengunjung. Tugas setiap aktor dapat dilihat pada diagram *Use Case* pada gambar 3.4:



Gambar 3.4 *Use Case* Diagram Aplikasi Pembelajaran Akord

3.4.2. *Use Case* Skenario

3.4.2.1. Autentikasi *Owner*

Autentikasi *Owner* digunakan untuk cara hak akses terhadap *Owner* yang akan menggunakan sistem dengan memasukan *username* dan *password* yang telah terdaftar pada *database*. Skenario untuk autentikasi *Owner* dapat dilihat pada tabel 3.6.

Tabel 3.6 Skenario Autentikasi *Owner*

Identifikasi		
Nomor	1	
Nama	Autentikasi *Owner*	
Tujuan	Memastikan hak akses pengguna terhadap sistem	
Deskripsi	Sistem menerima data identifikasi pengguna, berdasarkan data tersebut memutuskan apakah pengguna memiliki hak akses pada sistem	
Tipe		
Aktor	*Owner* (super admin)	
Skenario Utama		
Kondisi awal	Menampilkan halaman *login*	
Aksi Aktor		**Reaksi Sistem**
1) Memasukkan halaman *login owner* 3) Memasukan *username* dan *password*		2) Menampilkan *form login owner* 4) Mengecek *username* dan *password* 5) Apabila *username* atau *password* salah, sistem akan menampilkan pesan kesalahan dan akan tetap pada *form login* 6) Apabila *username* dan *password* benar maka sistem akan mengecek data pada tabel 7) Apabila data pada tabel salah maka sistem akan menampilkan pesan kesalahan dan akan tetap pada *form login* 8) Apabila data pada tabel benar maka sistem akan menampilkan antarmuka *owner*
Kondisi akhir	Jika pada akhir interaksi *username* dan *password* yang dimasukan pengguna benar maka pengguna dapat melakukan proses yang terdapat pada sistem sesuai dengan hak akses	

	pengguna. Jika salah maka sistem akan menampilkan pesan konfirmasi bahwa *username* dan *password* yang dimasukan salah.

3.4.2.2. Pengolahan Data Pengajar

Pengolahan data pengajar digunakan untuk melakukan proses pengolahan data pengajar yang dilakukan oleh *Owner* (super admin). Skenario untuk pengolahan data pengajar dapat dilihat pada tabel 3.7.

Tabel 3.7 Skenario Pengolahan Data Pengajar

Identifikasi		
Nomor	2	
Nama	Pengolahan Data Pengajar	
Tujuan	Melakukan proses penambahan dan ubah data pengajar	
Deskripsi	Pengguna dapat melakukan penambahan dan ubah data pengajar	
Tipe		
Aktor	*Owner* (super admin)	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai super admin	
Aksi Aktor		**Reaksi Sistem**
1) Memasuki halaman *owner* dan memilih pengolahan data pengajar		2) Sistem akan menampilkan beberapa fungsi, yaitu cari pengajar, tambah pengajar dan ubah pengajar
Kondisi akhir	Pengguna melakukan aktifitas dalam pengolahan data pengajar	

3.4.2.3. Tambah Data Pengajar

Tambah data pengajar digunakan untuk melakukan proses penambahan data pengajar yang dilakukan oleh *Owner* (super admin). Skenario tambah data pengajar dapat dilihat pada tabel 3.8.

Tabel 3.8 Skenario Tambah Data Pengajar

Identifikasi	
Nomor	3
Nama	Tambah Data Pengajar

Tujuan	Melakukan proses penambahan pengajar pada sistem	
Deskripsi	Pengguna dapat melakukan penambahan data pengajar baru pada sistem	
Tipe		
Aktor	*Owner* (super admin)	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin	
Aksi Aktor		**Reaksi Sistem**
1) Memilih tombol tambah data pengajar 3) Mengisi *form* tambah data pengajar 4) Memilih tombol simpan		2) Menampilkan *form* tambah data pengajar 5) Mengecek data yang diisi 6) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan pesan kesalahan dan tetap pada *form* tambah data pengajar 7) Apabila benar sistem akan mengecek data pengajar 8) Apabila data sudah ada pada tabel maka sistem akan menampilkan kesalahan/sudah ada dan tetap pada *form* tambah data pengajar 9) Apabila belum ada pada tabel maka data akan disimpan ke dalam tabel pengajar
Kondisi akhir	Pengguna berhasil melakukan kegiatan penambahan data pengajar	

3.4.2.4. Ubah Data Pengajar

Ubah data pengajar digunakan untuk melakukan ubah data pengajar yang dilakukan oleh *Owner* (super admin). Skenario ubah data pengajar dapat dilihat pada tabel 3.9.

Tabel 3.9 Skenario Ubah Data Pengajar

Identifikasi	
Nomor	4
Nama	Ubah Data Pengajar
Tujuan	Mengubah data pengajar pada sistem
Deskripsi	Pengguna dapat melakukan ubah data pengajar
Tipe	

Aktor	*Owner* (super admin)
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai super admin
Aksi Aktor	**Reaksi Sistem**
1) Memilih tombol ubah data pengajar 4) Memilih data pengajar 6) Mengubah data pengajar 7) Memilih tombol simpan	2) Mencari data pengajar 3) Menampilkan daftar data pengajar 5) Menampilkan *form* ubah data pengajar 8) Mengecek data yang diisi 9) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan kesalahan, tetap pada *form* ubah data pengajar 10) Apabila benar maka sistem akan mengecek data pada tabel pengajar 11) Apabila sudah ada maka sistem akan menampilkan pesan kesalahan atau sudah ada, tetap pada *form* ubah data pengajar 12) Apabila data yang diisi belum ada dan benar maka data akan disimpan kedalam tabel pengajar
Kondisi akhir	Pengguna berhasil melakukan kegiatan ubah data pengajar

3.4.2.5. Autentikasi Pengajar

Autentikasi pengajar digunakan untuk cara hak akses terhadap pengajar yang akan menggunakan sistem dengan memasukan *username* dan *password* yang telah terdaftar pada *database*. Skenario untuk autentikasi pengajar dapat dilihat pada tabel 3.10.

Tabel 3.10 Skenario Autentikasi Pengajar

Identifikasi	
Nomor	5
Nama	Autentikasi Pengajar
Tujuan	Memastikan hak akses pengguna terhadap sistem
Deskripsi	Sistem menerima data identifikasi pengguna, berdasarkan data tersebut memutuskan apakah pengguna memiliki hak akses pada

	sistem	
Tipe		
Aktor	Pengajar (super admin)	
Skenario Utama		
Kondisi awal	Menampilkan halaman *login*	
Aksi Aktor		**Reaksi Sistem**
1) Memasukkan halaman *login* pengajar 3) Memasukan *username* dan *password*		2) Menampilkan *form login* pengajar 4) Mengecek *username* dan *password* 5) Apabila *username* atau *password* salah, sistem akan menampilkan pesan kesalahan dan akan tetap pada *form login* 6) Apabila *username* dan *password* benar maka sistem akan mengecek data pada tabel 7) Apabila data pada tabel salah maka sistem akan menampilkan pesan kesalahan dan akan tetap pada *form login* 8) Apabila data pada tabel benar maka sistem akan menampilkan antarmuka pengajar
Kondisi akhir	Jika pada akhir interaksi *username* dan *password* yang dimasukan pengguna benar maka pengguna dapat melakukan proses yang terdapat pada sistem sesuai dengan hak akses pengguna. Jika salah maka sistem akan menampilkan pesan konfirmasi bahwa *username* dan *password* yang dimasukan salah.	

3.4.2.6. Pengolahan Data Murid

Pengolahan data murid digunakan untuk melakukan proses pengolahan data murid yang dilakukan oleh pengajar (admin). Skenario pengolahan data murid dapat dilihat pada tabel 3.11.

Tabel 3.11 Skenario Pengolahan Data Murid

Identifikasi	
Nomor	6
Nama	Pengolahan Data Murid
Tujuan	Melakukan proses penambahan dan ubah data murid

Deskripsi	Pengguna dapat melakukan penambahan dan ubah data murid
Tipe	
Aktor	Pengajar (admin)
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin
Aksi Aktor	**Reaksi Sistem**
1) Memasuki halaman pengajar dan memilih pengolahan data murid	2) Sistem akan menampilkan beberapa fungsi, yaitu cari murid, tambah murid dan ubah murid
Kondisi akhir	Pengguna melakukan aktifitas dalam pengolahan data murid

3.4.2.7. Tambah Data Murid

Tambah data murid digunakan untuk melakukan proses penambahan data murid yang dilakukan oleh pengajar (admin). Skenario tambah data murid dapat dilihat pada tabel 3.12.

Tabel 3.12 Skenario Tambah Data Murid

Identifikasi	
Nomor	7
Nama	Tambah Data Murid
Tujuan	Melakukan proses penambahan data murid pada sistem
Deskripsi	Pengguna dapat melakukan penambahan data murid baru pada sistem
Tipe	
Aktor	Pengajar (admin)
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin
Aksi Aktor	**Reaksi Sistem**
1) Memilih tombol tambah data murid 3) Mengisi *form* tambah data murid 4) Memilih tombol simpan	2) Menampilkan *form* tambah data murid 5) Mengecek data yang diisi 6) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan pesan kesalahan dan tetap pada *form* tambah data murid 7) Apabila benar sistem akan mengecek data murid 8) Apabila data sudah ada pada tabel maka sistem akan menampilkan

	kesalahan/sudah ada dan tetap pada *form* tambah data murid 9) Apabila belum ada pada tabel maka data akan disimpan ke dalam tabel murid
Kondisi akhir	Pengguna berhasil melakukan kegiatan penambahan data murid

3.4.2.8. Ubah Data Murid

Ubah data murid digunakan untuk melakukan ubah data murid yang dilakukan oleh pengajar (admin). Skenario ubah data murid dapat dilihat pada tabel 3.13.

Tabel 3.13 Skenario Ubah Data Murid

Identifikasi	
Nomor	8
Nama	Ubah Data Murid
Tujuan	Mengubah data murid pada sistem
Deskripsi	Pengguna dapat melakukan ubah data murid
Tipe	
Aktor	Pengajar (admin)
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin
Aksi Aktor	**Reaksi Sistem**
1) Memilih tombol ubah data murid	2) Mencari data murid 3) Menampilkan daftar data murid
4) Memilih data murid	5) Menampilkan *form* ubah data murid
6) Mengubah data murid	
7) Memilih tombol simpan	8) Mengecek data yang diisi 9) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan kesalahan, tetap pada *form* ubah data murid 10) Apabila benar maka sistem akan mengecek data pada tabel pengajar 11) Apabila sudah ada maka sistem akan menampilkan pesan kesalahan atau sudah ada, tetap pada *form* ubah data murid 12) Apabila data yang diisi belum ada dan benar maka data akan disimpan

	kedalam tabel murid
Kondisi akhir	Pengguna berhasil melakukan kegiatan ubah data murid

3.4.2.9. Pengolahan Data Materi

Pengolahan data materi digunakan untuk melakukan proses pengolahan data materi yang dilakukan oleh pengajar (admin). Skenario pengolahan data materi dapat dilihat pada tabel 3.14.

Tabel 3.14 Skenario Pengolahan Data Materi

Identifikasi	
Nomor	9
Nama	Pengolahan Data Materi
Tujuan	Melakukan proses penambahan dan ubah data materi
Deskripsi	Pengguna dapat melakukan penambahan dan ubah data materi
Tipe	
Aktor	Pengajar (admin)
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin
Aksi Aktor	**Reaksi Sistem**
1) Memasuki halaman pengajar dan memilih pengolahan data materi	2) Sistem akan menampilkan beberapa fungsi, yaitu cari materi, tambah materi dan ubah materi
Kondisi akhir	Pengguna melakukan aktifitas dalam pengolahan data materi

3.4.2.10. Tambah Data Materi

Tambah data materi digunakan untuk melakukan proses penambahan data materi yang dilakukan oleh pengajar (admin). Skenario tambah data materi dapat dilihat pada tabel 3.15.

Tabel 3.15 Skenario Tambah Data Materi

Identifikasi	
Nomor	10

Nama	Tambah Data Materi	
Tujuan	Membuat materi pada sistem	
Deskripsi	Pengguna dapat melakukan penambahan materi pada sistem	
Tipe		
Aktor	Pengajar (admin)	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin	
Aksi Aktor		**Reaksi Sistem**
1) Memilih tombol tambah data materi 3) Mengisi *form* tambah data materi 4) Memilih tombol simpan		2) Menampilkan *form* tambah data materi 5) Mengecek data yang diisi 6) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan pesan kesalahan dan tetap pada *form* tambah data materi 7) Apabila benar sistem akan mengecek data materi 8) Apabila data sudah ada pada tabel maka sistem akan menampilkan kesalahan/sudah ada dan tetap pada *form* tambah data materi 9) Apabila belum ada pada tabel maka data akan disimpan ke dalam tabel materi
Kondisi akhir	Pengguna berhasil melakukan kegiatan penambahan data materi	

3.4.2.11. Ubah Data Materi

Ubah data materi digunakan untuk melakukan ubah data materi yang dilakukan oleh pengajar (admin). Skenario ubah data materi dapat dilihat pada tabel 3.16.

Tabel 3.16 Skenario Ubah Data Materi

Identifikasi	
Nomor	11
Nama	Ubah Data Materi
Tujuan	Mengubah data materi pada sistem
Deskripsi	Pengguna dapat melakukan ubah data materi
Tipe	
Aktor	Pengajar (admin)

Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin
Aksi Aktor	**Reaksi Sistem**
1) Memilih tombol ubah data materi 4) Memilih data materi 6) Mengubah data materi 7) Memilih tombol simpan	2) Mencari data materi 3) Menampilkan daftar data materi 5) Menampilkan *form* ubah data materi 8) Mengecek data yang diisi 9) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan kesalahan, tetap pada *form* ubah data materi 10) Apabila benar maka sistem akan mengecek data pada tabel pengajar 11) Apabila sudah ada maka sistem akan menampilkan pesan kesalahan atau sudah ada, tetap pada *form* ubah data materi 12) Apabila data yang diisi belum ada dan benar maka data akan disimpan kedalam tabel materi
Kondisi akhir	Pengguna berhasil melakukan kegiatan ubah data materi

3.4.2.12. Hapus Data Materi

Hapus data materi digunakan untuk melakukan hapus data materi yang dilakukan oleh pengajar (admin). Skenario hapus data materi dapat dilihat pada tabel 3.17.

Tabel 3.17 Skenario Hapus Data Materi

Identifikasi	
Nomor	12
Nama	Hapus Data Materi
Tujuan	Menghapus data materi pada sistem
Deskripsi	Pengguna dapat melakukan hapus data materi
Tipe	
Aktor	Pengajar (admin)
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin
Aksi Aktor	**Reaksi Sistem**

1) Memilih hapus data materi 3) Memilih data yang dihapus 4) Memilih tombol hapus	2) Menampilkan *form* hapus data materi 5) Menampilkan konfirmasi ya atau tidak 6) Apabila tidak maka data tidak dihapus, apabila ya maka sistem akan menghapus data materi
Kondisi akhir	Pengguna berhasil melakukan kegiatan hapus data materi

3.4.2.13. Pengolahan Data Soal

Pengolahan data soal digunakan untuk melakukan proses pengolahan data soal yang dilakukan oleh pengajar (admin). Skenario pengolahan data soal dapat dilihat pada tabel 3.18.

Tabel 3.18 Skenario Pengolahan Data Soal

Identifikasi	
Nomor	13
Nama	Pengolahan Data Soal
Tujuan	Melakukan proses penambahan soal, ubah data soal, dan melihat hasil latihan soal
Deskripsi	Pengguna dapat melakukan penambahan soal, ubah data soal dan melihat hasil latihan soal
Tipe	
Aktor	Pengajar (admin)
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin
Aksi Aktor	**Reaksi Sistem**
1) Memasuki halaman pengajar dan memilih pengolahan data soal	2) Sistem akan menampilkan beberapa fungsi, yaitu cari soal, tambah soal dan ubah soal
Kondisi akhir	Pengguna melakukan aktifitas dalam pengolahan data soal

3.4.2.14. Tambah Data Soal

Tambah data soal digunakan untuk melakukan proses penambahan data soal yang dilakukan oleh pengajar (admin). Skenario tambah data soal dapat dilihat pada tabel 3.19.

Tabel 3.19 Skenario Tambah Data Soal

Identifikasi		
Nomor	14	
Nama	Tambah Data Soal	
Tujuan	Membuat soal pada sistem	
Deskripsi	Pengguna dapat melakukan penambahan soal pada sistem	
Tipe		
Aktor	Pengajar (admin)	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin	
Aksi Aktor		**Reaksi Sistem**
1) Memilih tombol tambah data soal 3) Mengisi *form* tambah data soal 4) Memilih tombol simpan		2) Menampilkan *form* tambah data soal 5) Mengecek data yang diisi 6) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan pesan kesalahan dan tetap pada *form* tambah data soal 7) Apabila benar sistem akan mengecek data soal 8) Apabila data sudah ada pada tabel maka sistem akan menampilkan kesalahan/sudah ada dan tetap pada *form* tambah data soal 9) Apabila belum ada pada tabel maka data akan disimpan ke dalam tabel soal
Kondisi akhir	Pengguna berhasil melakukan kegiatan penambahan data soal	

3.4.2.15. Ubah Data Soal

Ubah data soal digunakan untuk melakukan ubah data soal yang dilakukan oleh pengajar (admin). Skenario ubah data soal dapat dilihat pada tabel 3.20.

Tabel 3.20 Skenario Ubah Data Soal

Identifikasi		
Nomor	15	
Nama	Ubah Data Soal	
Tujuan	Mengubah data soal pada sistem	
Deskripsi	Pengguna dapat melakukan ubah data soal	
Tipe		
Aktor	Pengajar (admin)	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin	
Aksi Aktor		**Reaksi Sistem**
1) Memilih tombol ubah data soal 4) Memilih data soal 6) Mengubah data soal 7) Memilih tombol simpan		2) Mencari data soal 3) Menampilkan daftar data soal 5) Menampilkan *form* ubah data soal 8) Mengecek data yang diisi 9) Apabila terjadi kesalahan pada data yang diisi maka sistem akan menampilkan kesalahan, tetap pada *form* ubah data soal 10) Apabila benar maka sistem akan mengecek data pada tabel pengajar 11) Apabila sudah ada maka sistem akan menampilkan pesan kesalahan atau sudah ada, tetap pada *form* ubah data soal 12) Apabila data yang diisi belum ada dan benar maka data akan disimpan kedalam tabel soal
Kondisi akhir	Pengguna berhasil melakukan kegiatan ubah data soal	

3.4.2.16. Hasil Latihan Soal

Hasil latihan soal digunakan untuk melihat hasil latihan soal yang telah dilakukan oleh murid. Skenario hasil latihan soal dapat dilihat pada tabel 3.21.

Tabel 3.21 Skenario Hasil Latihan Soal

Identifikasi	
Nomor	16
Nama	Hasil Latihan Soal
Tujuan	Melihat hasil latihan soal
Deskripsi	Pengguna dapat melihat hasil latihan soal yang dilakukan oleh

	murid	
Tipe		
Aktor	Pengajar (admin)	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login* sebagai admin	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu hasil latihan soal		2) Sistem akan menampilkan daftar nilai dari latihan soal
Kondisi akhir	Pengguna berhasil melakukan kegiatan melihat hasil latihan soal	

3.4.2.17. Halaman Utama

Halaman utama digunakan untuk menampilkan halaman utama yang dilakukan oleh semua pengguna. Skenario halaman utama dapat dilihat pada tabel 3.22.

Tabel 3.22 Skenario Halaman Utama

Identifikasi		
Nomor	17	
Nama	Halaman Utama	
Tujuan	Menampilkan halaman utama sistem	
Deskripsi	Tampilan utama sistem pengenalan dan pembelajaran akord gitar	
Tipe		
Aktor	*Owner* (super admin), Pengajar (admin), Murid, dan Pengunjung	
Skenario Utama		
Kondisi awal	Halaman utama sistem	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu beranda		2) menampilkan halaman utama
Kondisi akhir	Sistem akan menampilkan informasi pembelajaran yang ada pada sistem	

3.4.2.18. Lihat Materi

Lihat materi digunakan untuk melakukan lihat materi yang dilakukan oleh semua pengguna. Skenario lihat meteri dapat dilihat pada tabel 3.23.

Tabel 3.23 Skenario Lihat Materi

Identifikasi		
Nomor	18	
Nama	Lihat Materi	
Tujuan	Menampilkan daftar materi yang ada pada sistem	
Deskripsi	Pengguna dapat melihat materi yang terdapat pada sistem	
Tipe		
Aktor	Owner (super admin), Pengajar (admin), Murid dan Pengunjung	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam halaman utama	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu materi 3) Memilih materi yang dilihat		2) Menampilkan daftar materi 4) Menampilkan materi yang dipilih
Kondisi akhir	Sistem akan menampilkan materi yang dipilih pengguna	

3.4.2.19. Pengenalan Akord

Pengenalan akord digunakan sebagai salah satu materi yang diterapkan pada sistem berupa jenis dan bentuk akord. Skenario untuk pengenalan akord dapat dilihat pada tabel 3.24.

Tabel 3.24 Skenario Akord

Identifikasi		
Nomor	19	
Nama	Pengenalan Akord	
Tujuan	Memberikan pengenalan tentang akord gitar pada murid	
Deskripsi	Pengguna mencari dan mempelajari akord sebagai bagian paling penting dalam pembelajaran gitar	
Tipe		
Aktor	Murid	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login*	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu pengenalan akord 3) Memilih jenis akord dan tipe user 4) Memilih tombol tampil akord		2) Menampilkan antarmuka pengenalan akord 5) Mengecek data yang dipilih 6) Apabila tipe user belum dipilih maka menampilkan peringatan tipe user belum dipilih 7) Apabila tipe user sudah dipilih maka

	menampilkan akord yang dipilih
Kondisi akhir	Murid menggunakan aplikasi pengenalan akord sebagai bagian paling penting dalam pembelajaran gitar

3.4.2.20. Lihat Skor (Akord *Game*)

Skord Akord *Game* digunakan untuk melihat hasil dari evaluasi (Akord *Game*) yang telah dilakukan murid. Skenario untuk skor akord *game* dapat dilihat pada tabe Tabel 3.25.

Tabel 3.25 Skenario Skor Akord *Game*

Identifikasi	
Nomor	20
Nama	Skor Akord *Game*
Tujuan	Melihat hasil dari Evaluasi (Akord *Game*)
Deskripsi	Pengguna dapat meliihat sepuluh nilai tertinggi dari hasil Evaluasi (Akord *Game*)
Tipe	
Aktor	Murid
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login*
Aksi Aktor	**Reaksi Sistem**
1) Murid memilih menu skor akord *game*	2) Sistem akan menampilkan dafar skor maksimal 10 skor tertinggi
Kondisi akhir	Murid mengetahui hasil skor akor gitar tertinggi

3.4.2.21. Autentikasi Murid

Autentikasi murid digunakan untuk cara hak akses terhadap pengguna yang akan menggunakan sistem dengan memasukan *username* dan *password* yang telah terdaftar pada *database*. Skenario untuk autentikasi murid dapat dilihat pada tabel 3.26.

Tabel 3.26 Skenario Autentikasi Murid

Identifikasi	
Nomor	21

Nama	Autentikasi Murid	
Tujuan	Memastikan hak akses pengguna terhadap sistem	
Deskripsi	Sistem menerima data identifikasi murid, berdasarkan data tersebut memutuskan apakah murid memiliki hak akses pada sistem	
Tipe		
Aktor	Murid	
Skenario Utama		
Kondisi awal	Menampilkan halaman *login*	
Aksi Aktor		**Reaksi Sistem**
1) Memasukkan halaman *login* murid 3) Memasukan *username* dan *password*		2) Menampilkan *form login* murid 4) Mengecek *username* dan *password* 5) Apabila *username* atau *password* salah, sistem akan menampilkan pesan kesalahan dan akan tetap pada *form login* 6) Apabila *username* dan *password* benar maka sistem akan mengecek data pada tabel 7) Apabila data pada tabel salah maka sistem akan menampilkan pesan kesalahan dan akan tetap pada *form login* 8) Apabila data pada tabel benar maka sistem akan menampilkan antarmuka murid
Kondisi akhir	Jika pada akhir interaksi *username* dan *password* yang dimasukan murid benar maka murid dapat melakukan hak akses pada sistem *website* dan dekstop. Jika salah maka sistem akan menampilkan pesan konfirmasi bahwa *username* dan *password* yang dimasukan salah.	

3.4.2.22. Latihan Soal

Latihan soal digunakan sebagai salah satu evaluasi dalam bentuk tebak akord yang diterapkan pada sistem berupa jenis dan bentuk akord. Skenario untuk latihan soal dapat dilihat pada tabel 3.27.

Tabel 3.27 Skenario Latihan Soal

Identifikasi		
Nomor	22	
Nama	Latihan Soal	
Tujuan	Melatih dalam memahami dan mengingat akord gitar	
Deskripsi	Pengguna dapat memilih jenis akord sebagai bahan evaluasi dalam bentuk latihan soal	
Tipe		
Aktor	Murid	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login*	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu latihan soal 3) Memilih jenis akord 4) Memilih tombol milai 6) Menjawab soal		2) Menampilkan antarmuka latihan soal 5) Menampilkan soal dan durasi waktu untuk menjawab 7) Mengecek jawaban 8) Apabila jawaban belum diisi maka menampilkan pesan jawaban belum diisi dan tetap menampilkan hasil latihan soal 9) Apabila jawaban sudah diisi maka sistem akan menghitung jumlah jawaban 10) Menampilkan hasil latihan
Kondisi akhir	Murid menggunakan aplikasi latihan soal sebagai tahap evaluasi dalam pembelajaran gitar	

3.4.2.23. *Download* Materi

Download materi digunakan untuk melakukan *download* materi yang dilakukan oleh semua pengguna dalam mendapatkan materi yang ada pada sistem dalam bentuk beberapa file. Skenario *download* materi dapat dilihat pada tabel 3.28.

Tabel 3.28 Skenario *Download* Materi

Identifikasi	
Nomor	23
Nama	*Download* Materi
Tujuan	Memberikan materi dalam bentuk file

Deskripsi	Pengguna dapat me-*download* materi pada sistem dalam bentuk file
Tipe	
Aktor	*Owner* (super admin), Pengajar (admin), Murid, dan Pengunjung
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam halaman utama
Aksi Aktor	**Reaksi Sistem**
1) Memilih menu *download* 3) Memilih materi yang akan di*download*	2) Menampilkan daftar *download* materi 4) Mengarahkan file download
Kondisi akhir	Pengguna mendapatkan materi yang telah di-*download* dalam bentuk file

3.4.2.24. Gitar *Tuning*

Gitar *Tuning* digunakan untuk mensetting dan menstandarkan *tune* (nada) gitar yang telah ditetapkan pada aplikasi desktop. Skenario untuk gitar *tuning* dapat dilihat pada tabel 3.29.

Tabel 3.29 Skenario Gitar *Tuning*

Identifikasi	
Nomor	24
Nama	Gitar *Tuning*
Tujuan	Mensetting *tune* (nada) gitar sesuai standar
Deskripsi	Pengguna mensetting *tune* (nada) gitar sesuai standard sebagai langkah awal untuk melakukan evaluasi (akord *game*)
Tipe	
Aktor	Murid
Skenario Utama	
Kondisi awal	Pengguna sedang berada dalam *session login*
Aksi Aktor	**Reaksi Sistem**
1) Memilih menu gitar *tuning* 3) Mensetting nada gitar	2) Menampilkan antarmuka gitar *tuning* 3) Menerima masukan suara 4) Proses *sound recognition* 5) Apabila settingan belum akurat sistem tidak akan menampilkan hasil nada yang disetting 6) Apabila settingan telah akurat maka sistem akan menampilkan nada yang sedang disetting

Kondisi akhir	Pengguna sudah memiliki *setting*-an nada standar pada gitarnya

3.4.2.25. Evaluasi (Akord *Game*)

Evaluasi (Akord *Game*) digunakan untuk pembelajaran akord yang diterapkan pada aplikasi desktop. Dalam tahap Evaluasi (Akord *Game*) dibuat dalam bentuk latihan dan *game*. Skenario Evaluasi (Akord *Game*) dapat dilihat pada tabel 3.30.

Tabel 3.30 Skenario Evaluasi (Akord *Game*)

Identifikasi		
Nomor	25	
Nama	Evaluasi (Akord *Game*)	
Tujuan	Memberikan pembelajaran tentang akord gitar pada murid	
Deskripsi	Pengguna melakukan Evaluasi (Akord *Game*) dengan memainkan akord *game* sebagai tahap untuk mengetahui tingkatan kemampuan dalam mempelajari gitar khususnya akord gitar	
Tipe		
Aktor	Murid	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login*	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu evaluasi		2) Menampilkan antarmuka evaluasi
Kondisi akhir	Pengguna dapat menggunakan aplikasi akord *game* sebagai tahap pembelajarn akord gitar	

3.4.2.26. Latihan (Akord *Game*)

Latihan (Akord *Game*) digunakan untuk pembelajaran akord yang diterapkan pada aplikasi desktop. Dalam tahap Latihan (Akord *Game*) dibuat dalam bentuk latihan tebak akord. Skenario Latihan (Akord *Game*) dapat dilihat pada tabel 3.31.

Tabel 3.31 Skenario Latihan (Akord *Game*)

Identifikasi		
Nomor	26	
Nama	Latihan (Akord *Game*)	
Tujuan	Memberikan pembelajaran tentang akord gitar pada murid	
Deskripsi	Pengguna melakukan Latihan (Akord *Game*) dengan memainkan akord *game* sebagai tahap untuk mengetahui tingkatan kemampuan dalam mempelajari gitar khususnya akord gitar	
Tipe		
Aktor	Murid	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login*	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu latihan 4) Menjawab dengan memainkan akord langsung		2) Menampilkan antarmuka latihan 3) menampilkan akord yang harus ditebak 5) Menerima masukan suara 6) Proses *sound recognition* 7) Apabila jawaban benar maka sistem akan menampilkan soal berikutnya dan sistem akan menambahkan jumlah skor 8) Apabila dalam durasi jawaban tidak benar makan sistem akan menampilkan konfirmasi salah dan akan menyimpan jawaban yang salah 9) Menampilkan jumlah skor
Kondisi akhir	Pengguna dapat menggunakan aplikasi akord *game* sebagai tahap pembelajarn akord gitar	

3.4.2.27. *Game* (Akord *Game*)

Game (Akord *Game*) digunakan untuk pembelajaran akord yang diterapkan pada aplikasi desktop. Dalam tahap *Game* (Akord *Game*) dibuat dalam bentuk game tebak akord. Skenario *Game* (Akord *Game*) dapat dilihat pada tabel 3.32.

Tabel 3.32 Skenario Latihan (Akord *Game*)

Identifikasi	
Nomor	27

Nama	*Game* (Akord *Game*)	
Tujuan	Memberikan pembelajaran tentang akord gitar pada murid	
Deskripsi	Pengguna melakukan *Game* (Akord *Game*) dengan memainkan akord *game* sebagai tahap untuk mengetahui tingkatan kemampuan dalam mempelajari gitar khususnya akord gitar	
Tipe		
Aktor	Murid	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login*	
Aksi Aktor		**Reaksi Sistem**
1) Memilih menu game 4) Menjawab dengan memainkan akord langsung		2) Menampilkan antarmuka game 3) Menampilkan akord yang harus ditebak 5) Menerima inputan suara 6) Proses *sound recognition* 7) Apabila jawaban benar maka sistem akan menampilkan konfirmasi benar dan menjumlahkan jumlah skor 8) Apabila salah dalam durasi waktu maka game berakhir 9) Menampilkan dan mengupdate jumlah skor
Kondisi akhir	Pengguna dapat menggunakan aplikasi akord *game* sebagai tahap pembelajarn akord gitar	

3.4.2.28. Update Skor (Akord *Game*)

Update Skor (Akord *Game*) digunakan untuk mengupdate data skor yang diterapkan pada aplikasi desktop maupun *web*site. Dalam tahap Update Skor (Akord *Game*) dibuat dalam bentuk hasil dari tahap Evaluasi (Akord *Game*). Skenario Update Skor (Akord *Game*) dapat dilihat pada tabel 3.33.

Tabel 3.33 Skenario Update Skor

Identifikasi	
Nomor	28
Nama	Update Skor (Akord *Game*)
Tujuan	Mengupdate skor akord *game* pada tabel skor
Deskripsi	Pengguna melakukan Update Skor setelah pengguna telah melakukan evaluasi dengan menekan tombol Ok
Tipe	

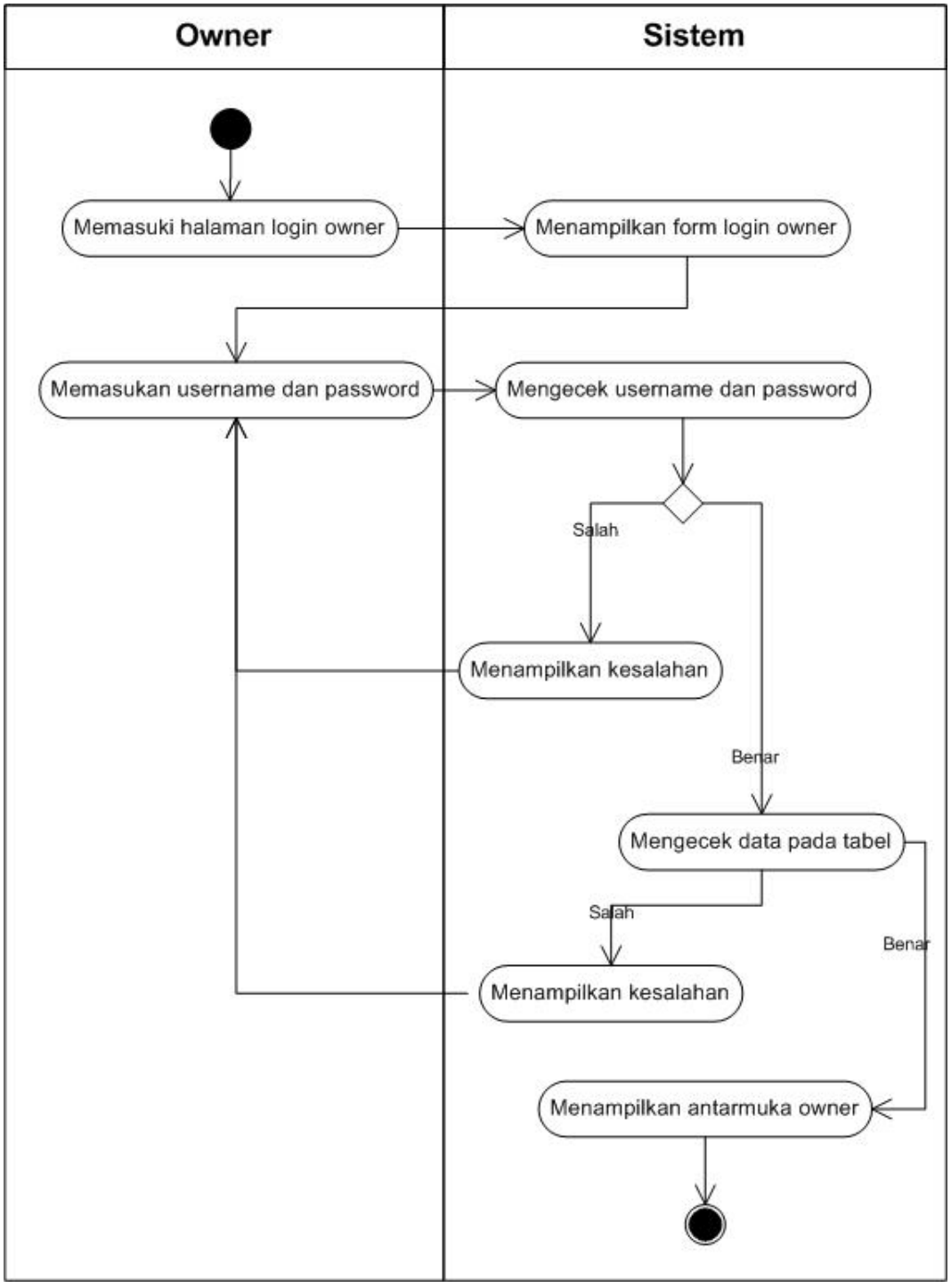
Aktor	Murid	
Skenario Utama		
Kondisi awal	Pengguna sedang berada dalam *session login*	
Aksi Aktor		**Reaksi Sistem**
1) Telah melakukan dan menyelesaikan evaluasi, yaitu game (akord game).		2) Sistem akan mengupdate skor ke dalam tabel skor dan menampilkan skor pada aplikasi dekstop maupun *web*site
Kondisi akhir	Pengguna telah mengupdate skor hasil dari evaluasi (akord *game*)	

3.4.3. *Activity Diagram*

Activity Diagram menggambarkan alur kerja (*workflow*) dari suatu aktivitas ke aktivitas lainnya dalam sebuah sistem yang sedang dirancang.

3.4.3.1. Autentikasi *Owner*

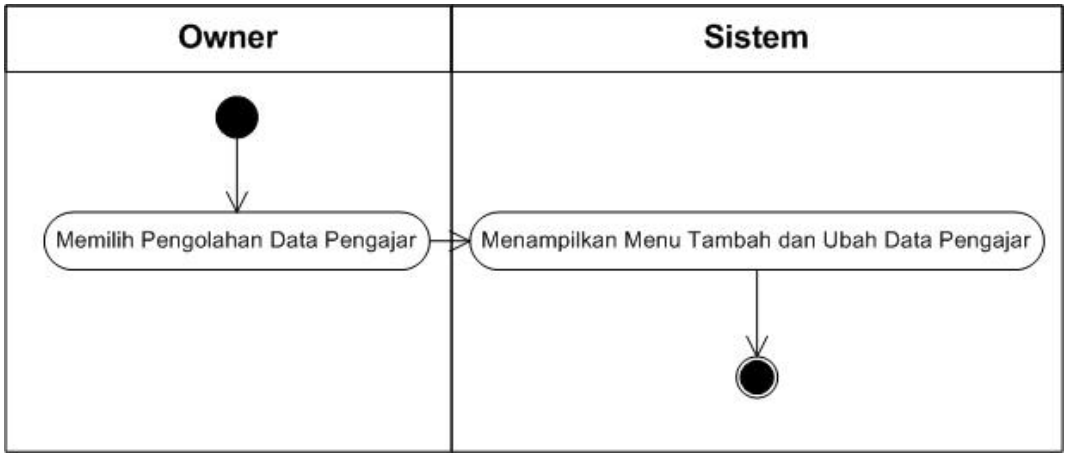
Aktifitas autentikasi *Owner* merupakan aktifitas yang dilakukan oleh *Owner* (super admin) untuk mendapatkan hak akses pada sistem. Seluruh tahapan aktifitas autentikasi *Owner* dapat dilihat pada gambar 3.5.



Gambar 3.5 Activity *Diagram* Autentikasi *Owner*

3.4.3.2. Pengolahan Data Pengajar

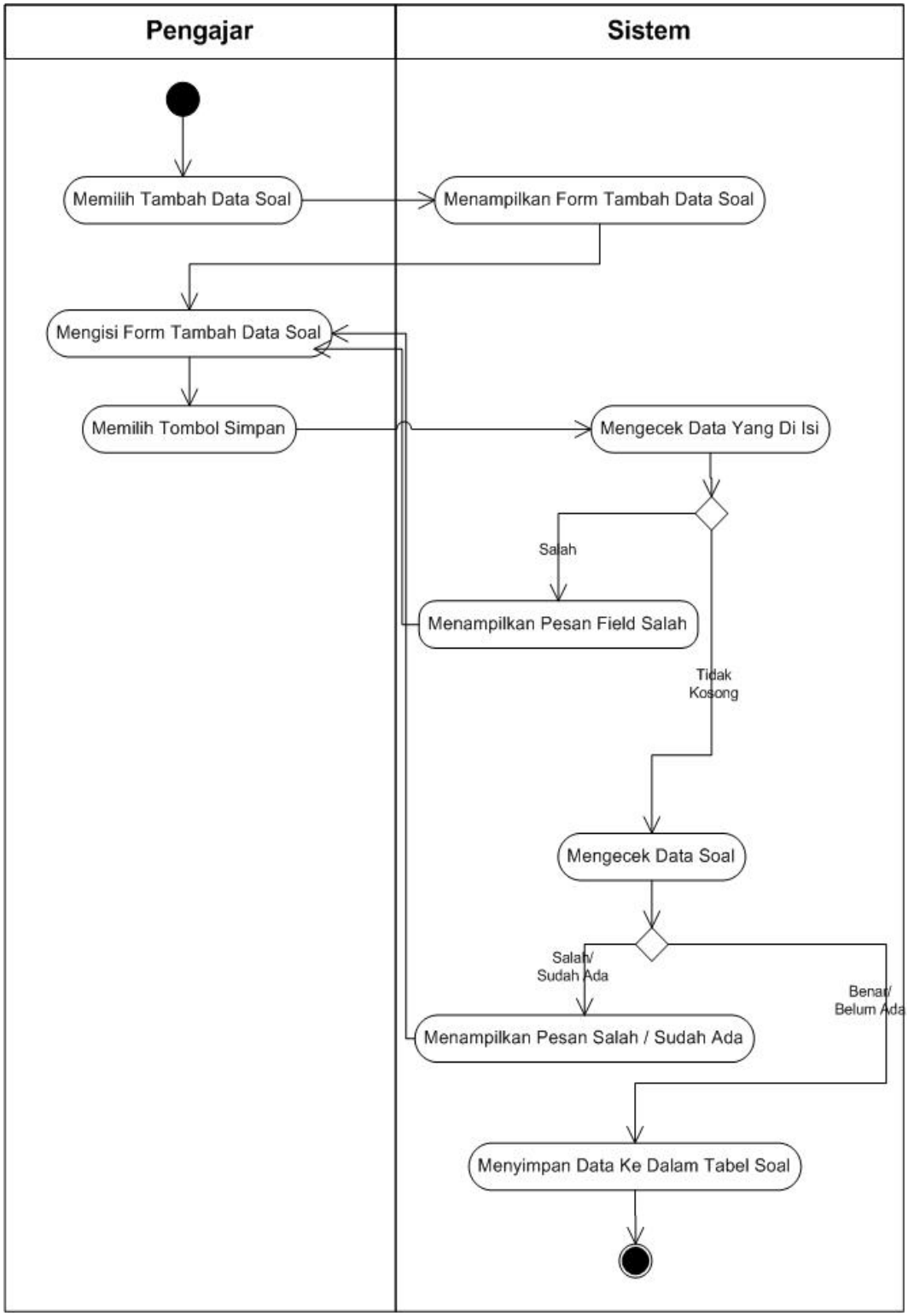
Aktifitas pengolahan data pengajar merupakan aktifitas yang dilakukan pengguna yaitu *Owner* (super admin) yang memiliki hak akses dalam melakukan tambah dan ubah data pengajar pada sistem. Seluruh tahapan aktifitas pengolahan data pengajar dapat dilihat pada gambar 3.6.



Gambar 3.6 *Activity Diagram* Pengolahan Data Pengajar

3.4.3.3. Tambah Data Pengajar

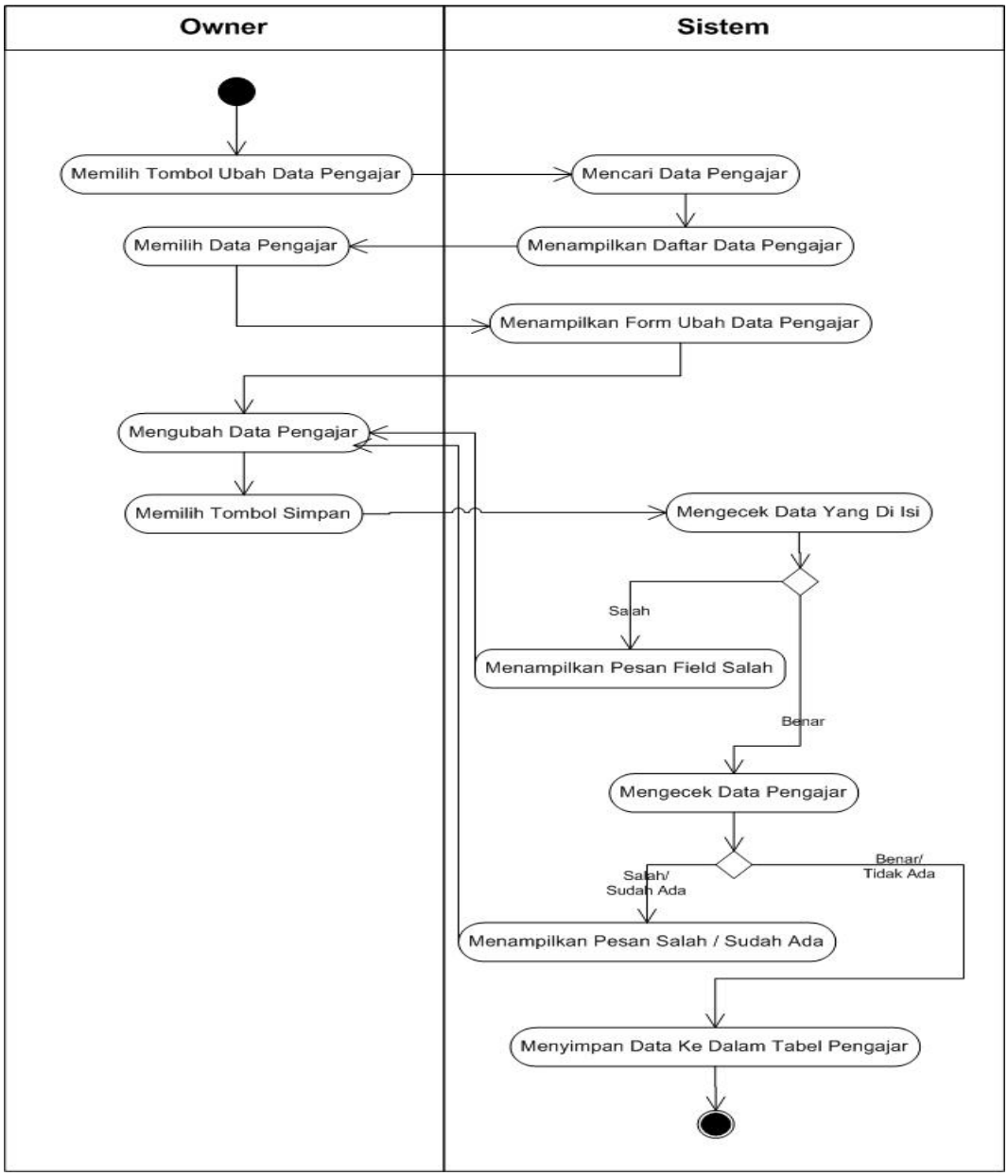
Aktifitas tambah data pengajar merupakan aktifitas yang dilakukan *Owner* (super admin) untuk menambahkan data pengajar. Seluruh tahapan aktifitas tambah data pengajar dapat dilihat pada gambar 3.7.



Gambar 3.7 *Activity Diagram* Tambah Data Pengajar

3.4.3.4. Ubah Data Pengajar

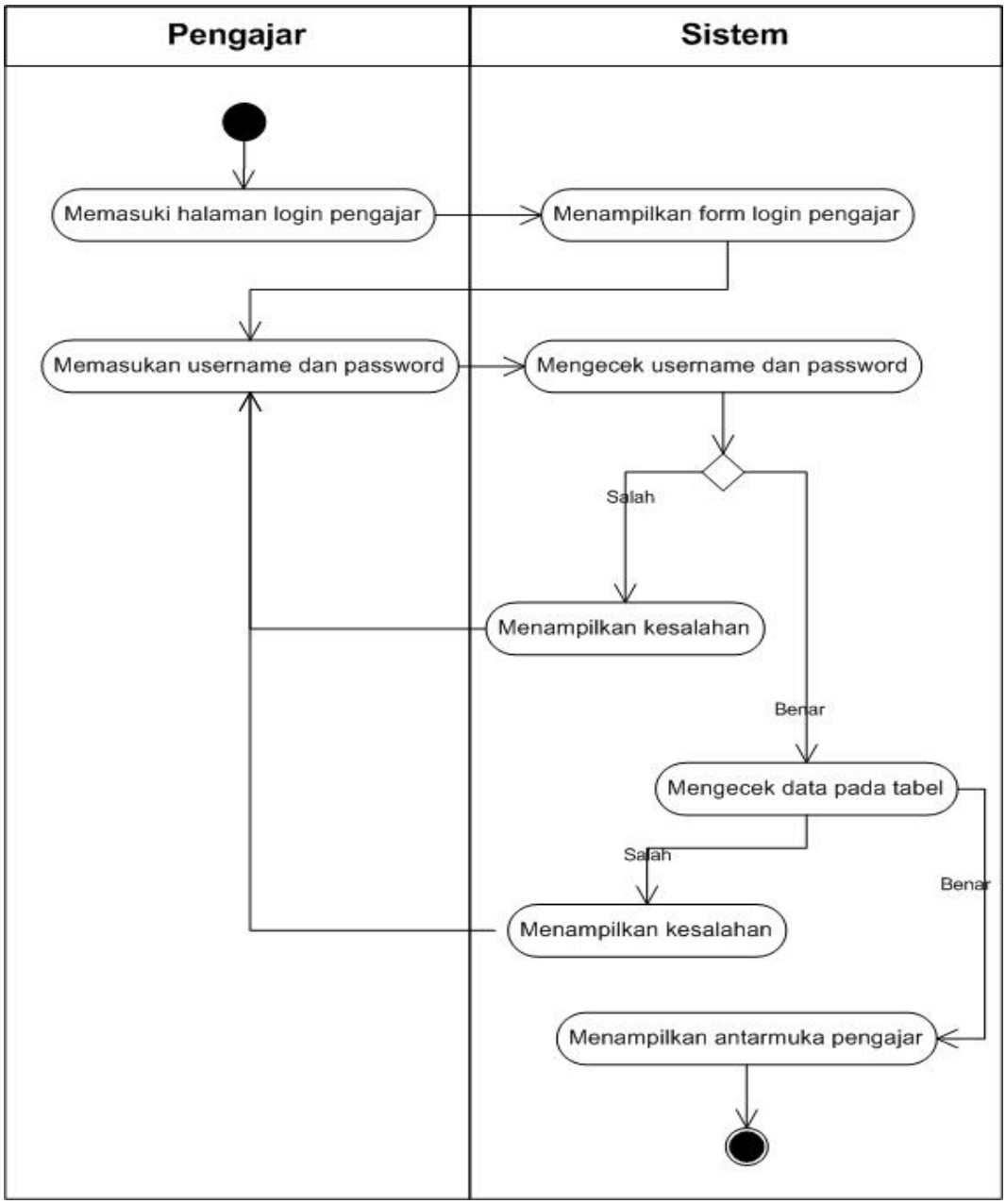
Aktifitas ubah data pengajar merupakan aktifitas yang dilakukan *Owner* (super admin) untuk mengubah data pengajar. Seluruh tahapan aktifitas ubah data pengajar dapat dilihat pada gambar 3.8.



Gambar 3.8 *Activity Diagram* Ubah Data Pengajar

3.4.3.5. Autentikasi Pengajar

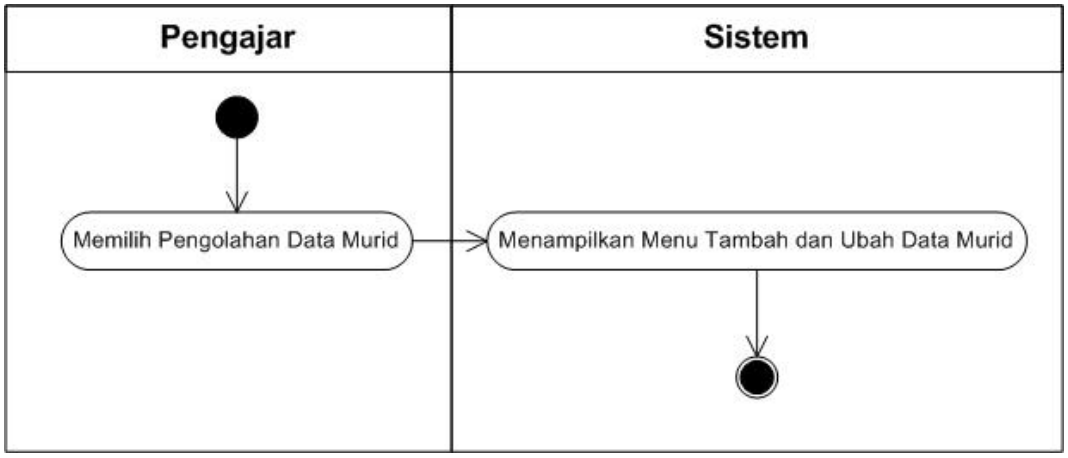
Aktifitas autentikasi Pengajar merupakan aktifitas yang dilakukan oleh Pengajar (admin) untuk mendapatkan hak akses pada sistem. Seluruh tahapan aktifitas autentikasi Pengajar dapat dilihat pada gambar 3.9.



Gambar 3.9 *Activity Diagram* Autentikasi Pengajar

3.4.3.6. Pengolahan Data Murid

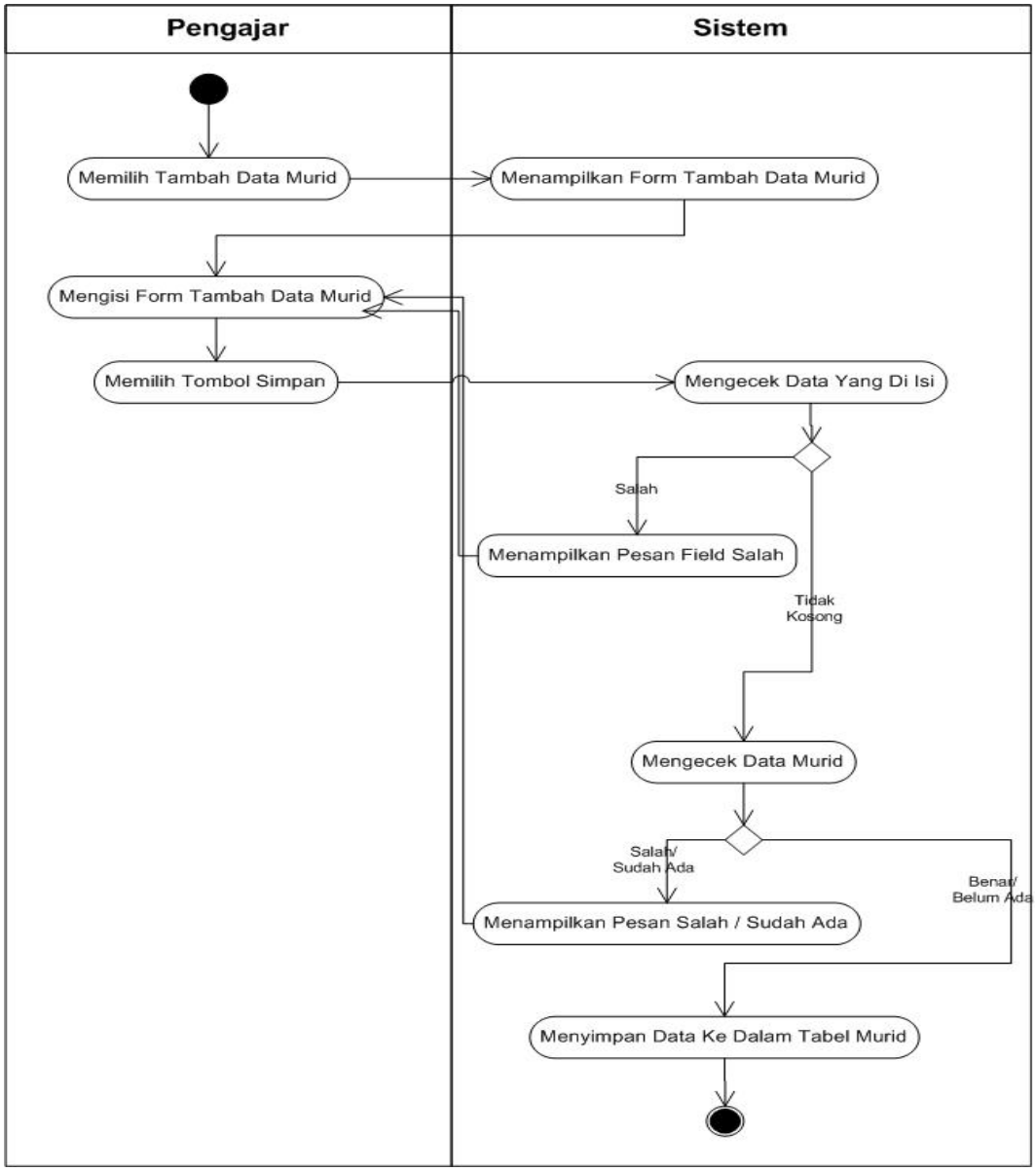
Aktifitas pengolahan data murid merupakan aktifitas yang dilakukan pengguna yaitu Pengajar (admin) yang memiliki hak akses dalam melakukan tambah dan ubah data murid pada sistem. Seluruh tahapan aktifitas pengolahan data murid dapat dilihat pada gambar 3.10.



Gambar 3.10 *Activity Diagram* Pengolahan Data Murid

3.4.3.7. Tambah Data Murid

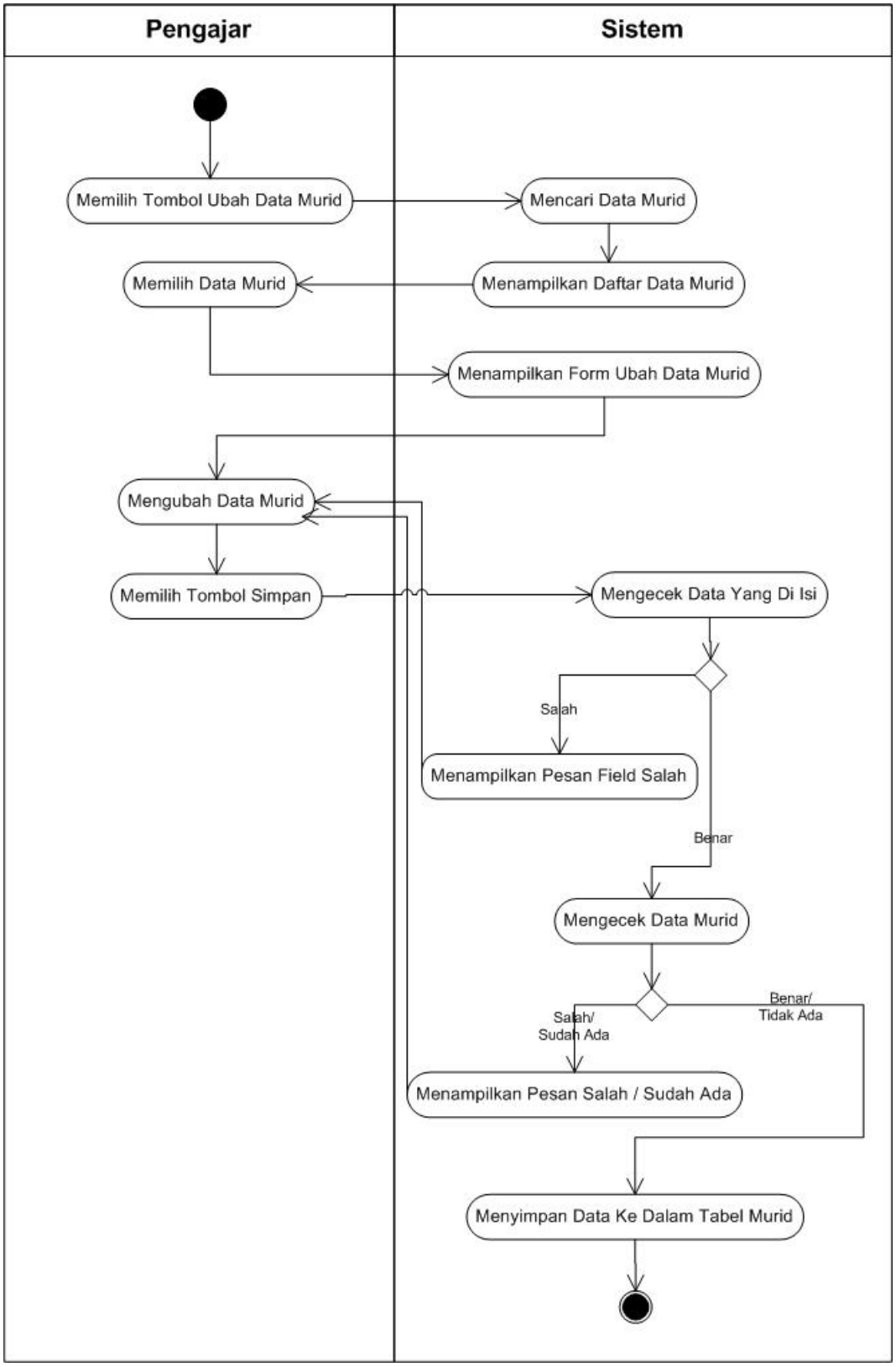
Aktifitas tambah data murid merupakan aktifitas yang dilakukan Pengajar (admin) untuk menambahkan murid. Seluruh tahapan aktifitas tambah data murid dapat dilihat pada gambar 3.11.



Gambar 3.11 *Activity Diagram* Tambah Data Murid

3.4.3.8. Ubah Data Murid

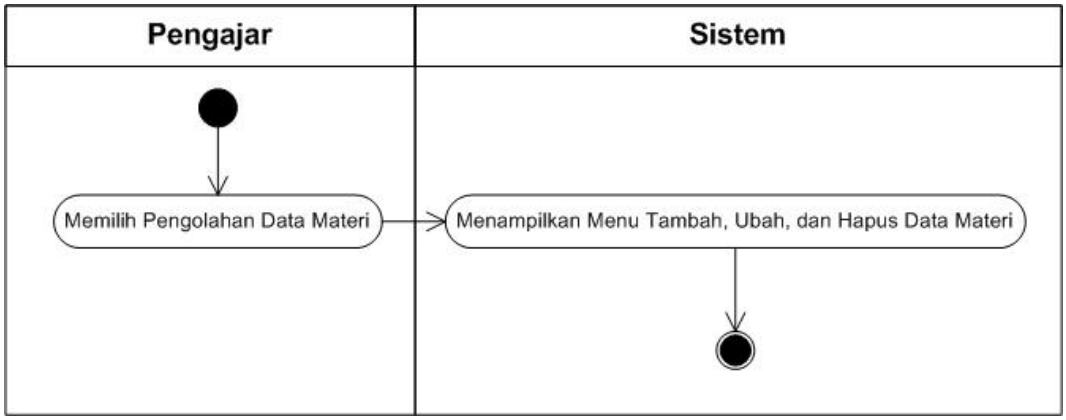
Aktifitas ubah data murid merupakan aktifitas yang dilakukan Pengajar (admin) untuk mengubah data murid. Seluruh tahapan aktifitas ubah data murid dapat dilihat pada gambar 3.12.



Gambar 3.12 *Activity Diagram* Ubah Data Murid

3.4.3.9. Pengolahan Data Materi

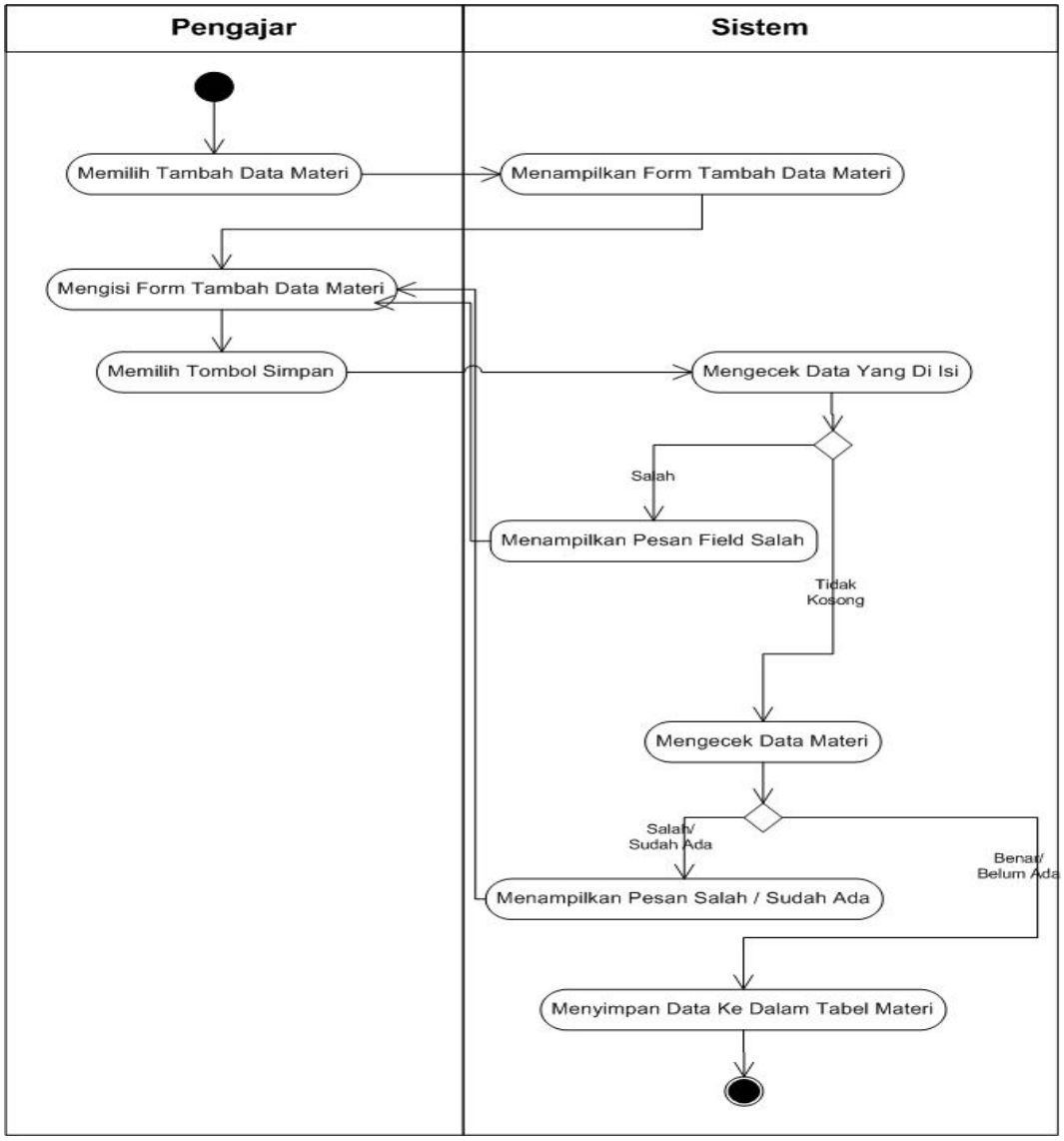
Aktifitas pengolahan data materi merupakan aktifitas yang dilakukan pengguna yaitu Pengajar (admin) yang memiliki hak akses dalam melakukan tambah dan ubah data materi pada sistem. Seluruh tahapan aktifitas pengolahan data materi dapat dilihat pada gambar 3.13.



Gambar 3.13 *Activity Diagram* Pengolahan Data Materi

3.4.3.10. Tambah Data Materi

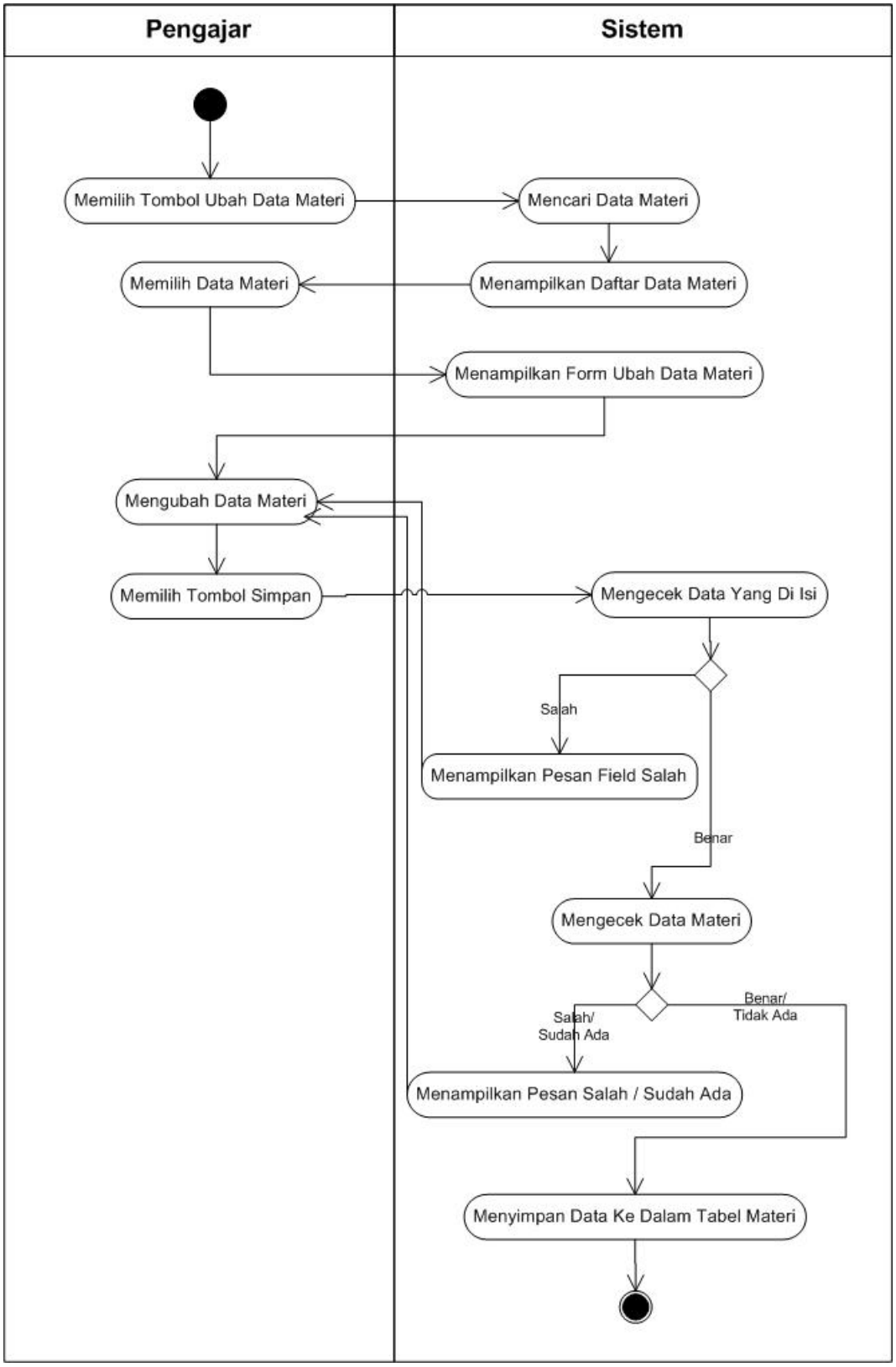
Aktifitas tambah materi merupakan aktifitas yang dilakukan Pengajar (admin) untuk menambahkan materi. Seluruh tahapan aktifitas tambah data materi dapat dilihat pada gambar 3.14.



Gambar 3.14 *Activity Diagram* Tambah Data Materi

3.4.3.11. Ubah Data Materi

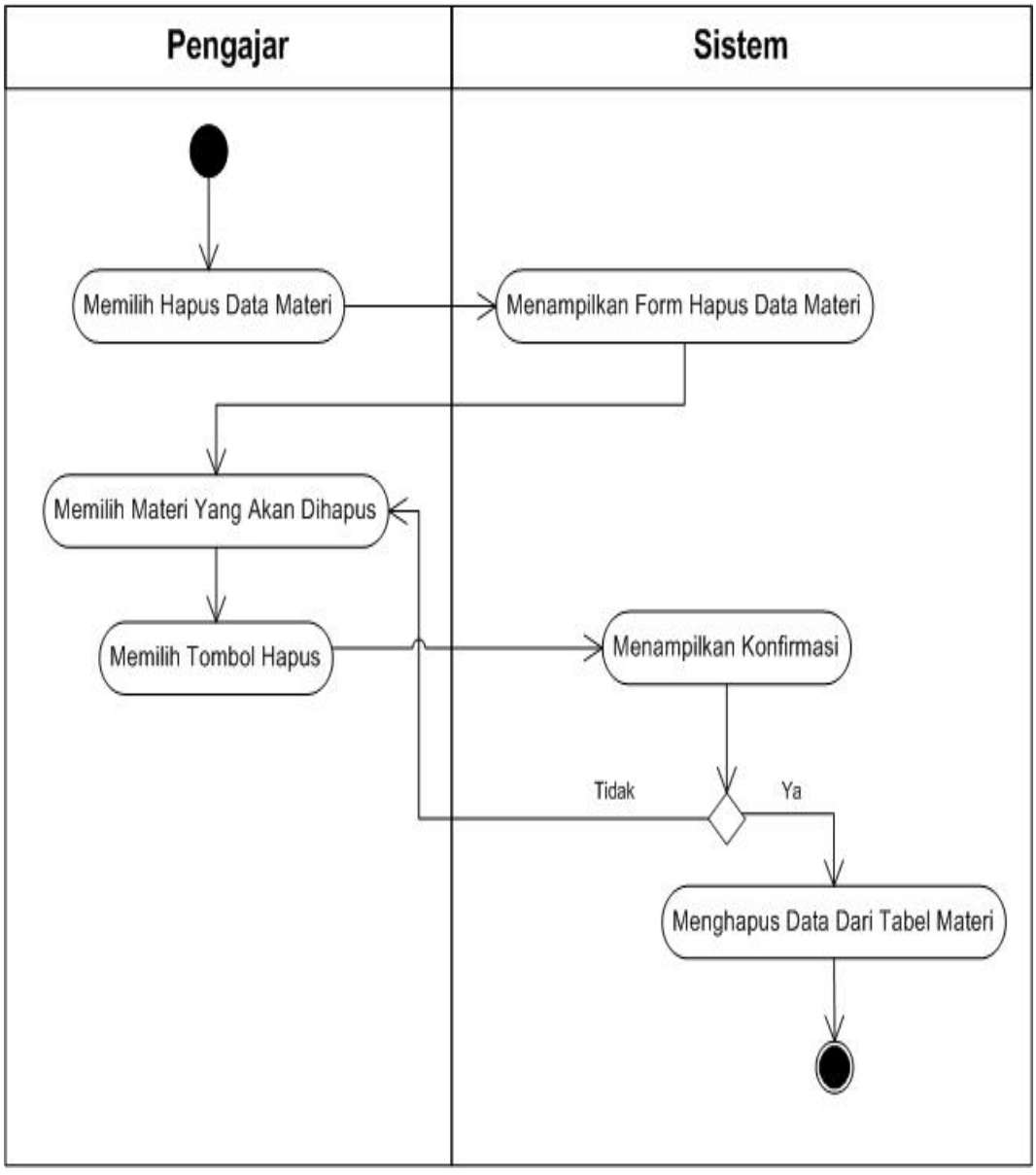
Aktifitas ubah data materi merupakan aktifitas yang dilakukan Pengajar (admin) untuk mengubah data materi. Seluruh tahapan aktifitas ubah data materi dapat dilihat pada gambar 3.15.



Gambar 3.15 ***Activity Diagram*** **Ubah Materi**

3.4.3.12. Hapus Data Materi

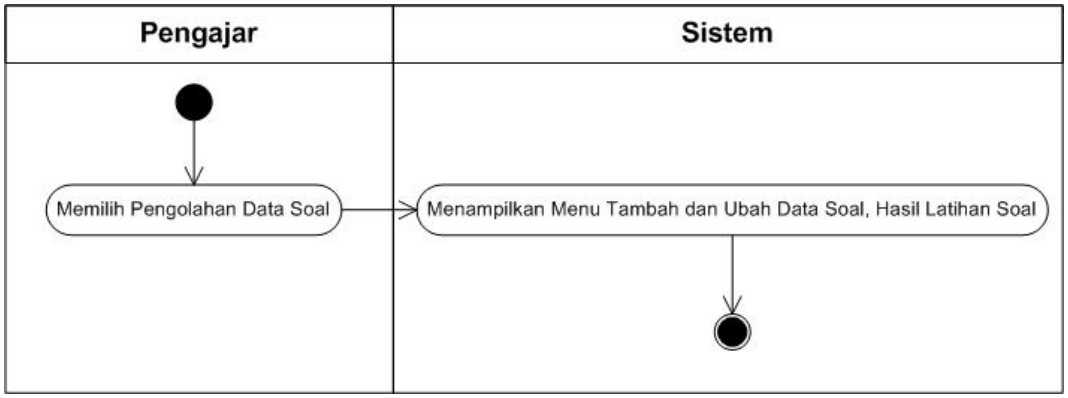
Pengajar memilih menu hapus data materi untuk menghapus data materi yang dipilih. Seluruh tahapan aktifitas hapus data materi dapat dilihat pada gambar 3.16.



Gambar 3.16 *Activity Diagram* Hapus Data Materi

3.4.3.13. Pengolahan Data Soal

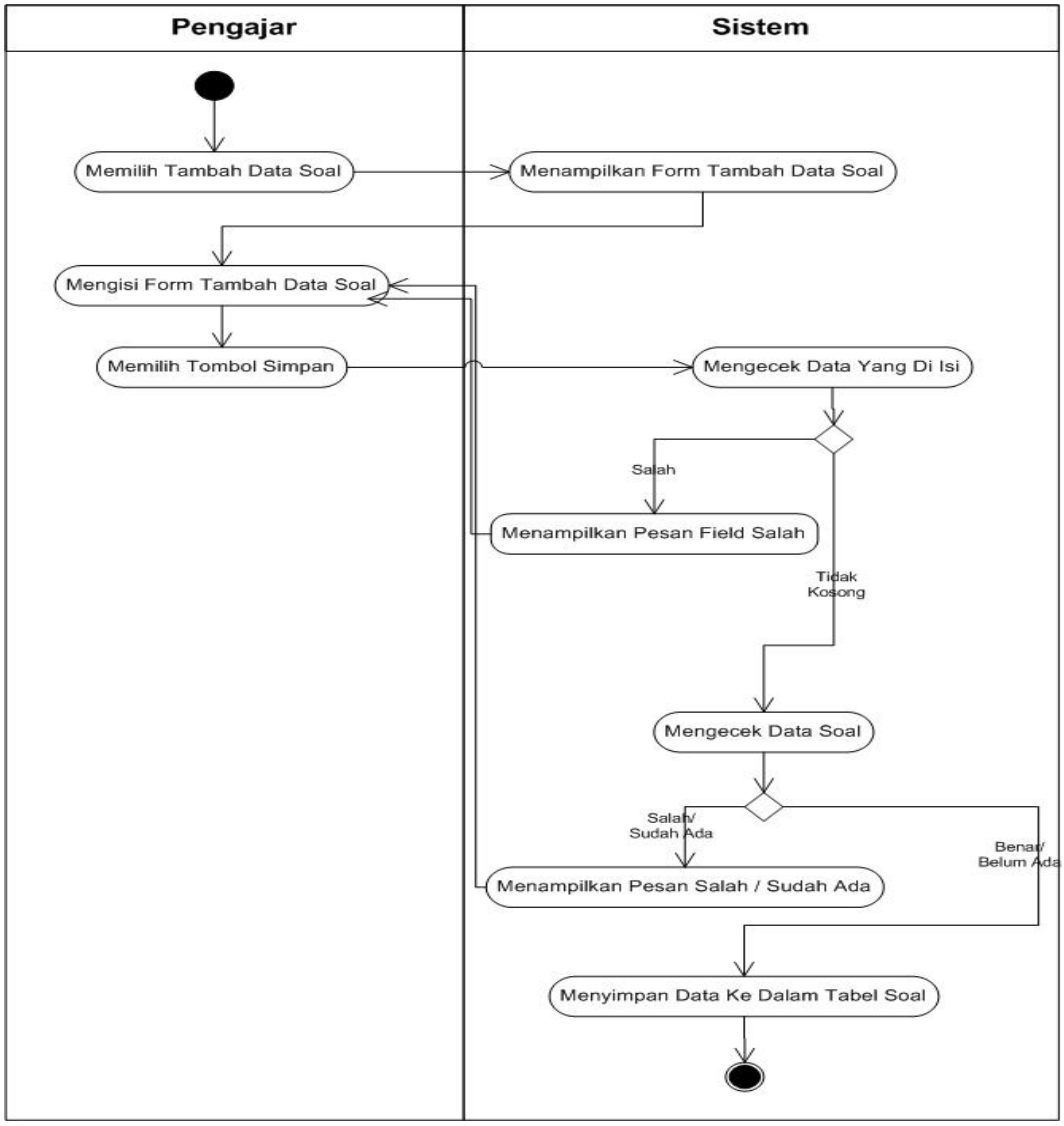
Aktifitas pengolahan data soal merupakan aktifitas yang dilakukan pengguna yaitu Pengajar (admin) yang memiliki hak akses dalam melakukan tambah dan ubah data soal pada sistem serta melihat hasil latihan soal. Seluruh tahapan aktifitas pengolahan data materi dapat dilihat pada gambar 3.177.



Gambar 3.17 Activity *Diagram* Pengolahan Data Soal

3.4.3.14. Tambah Data Soal

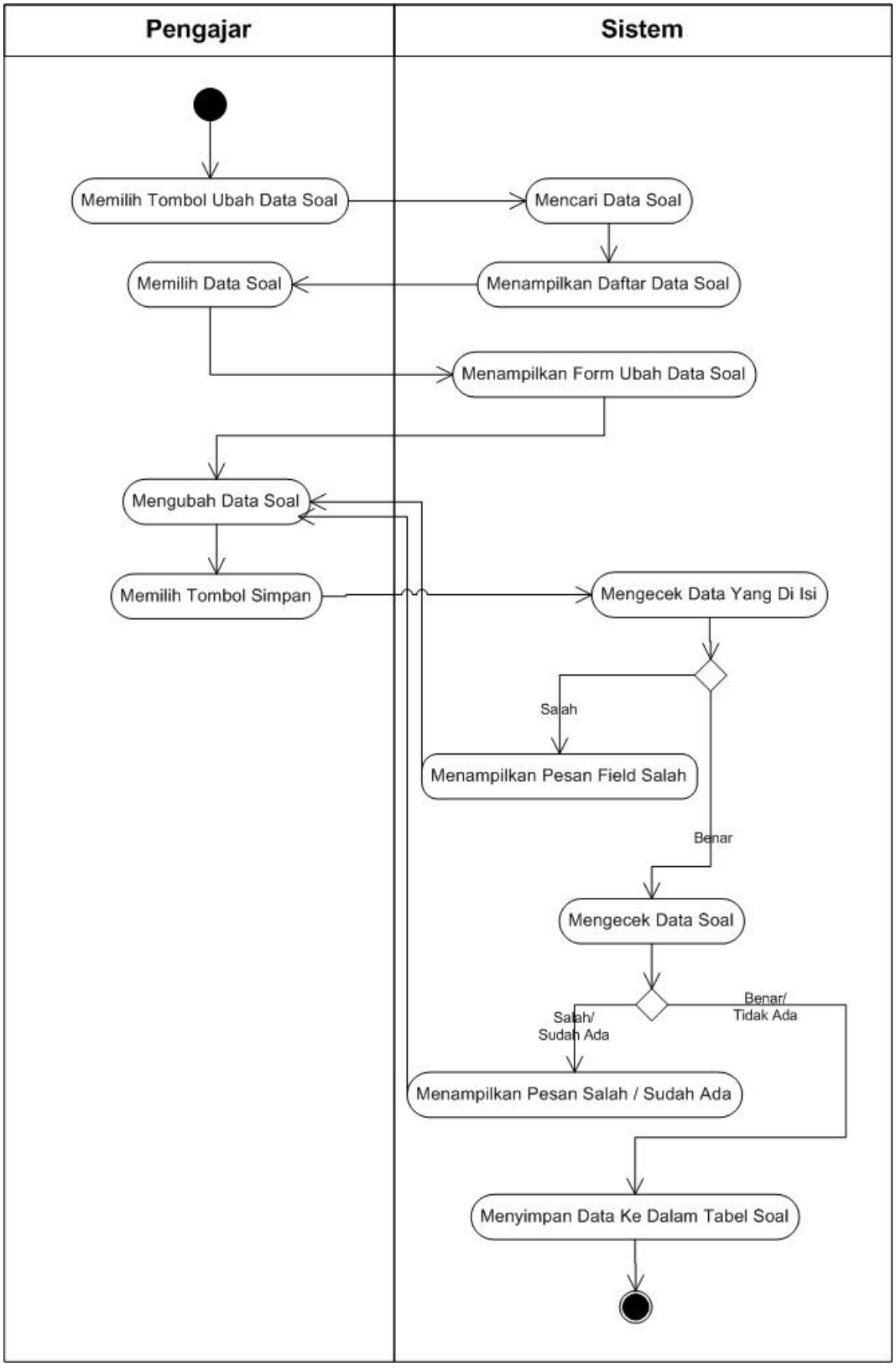
Tambah data soal merupakan aktifitas yang dilakukan Pengajar (admin) untuk menambahkan soal. Seluruh tahapan aktifitas tambah data soal dapat dilihat pada gambar 3.18.



Gambar 3.18 *Activity Diagram* Tambah Data Soal

3.4.3.15. Ubah Data Soal

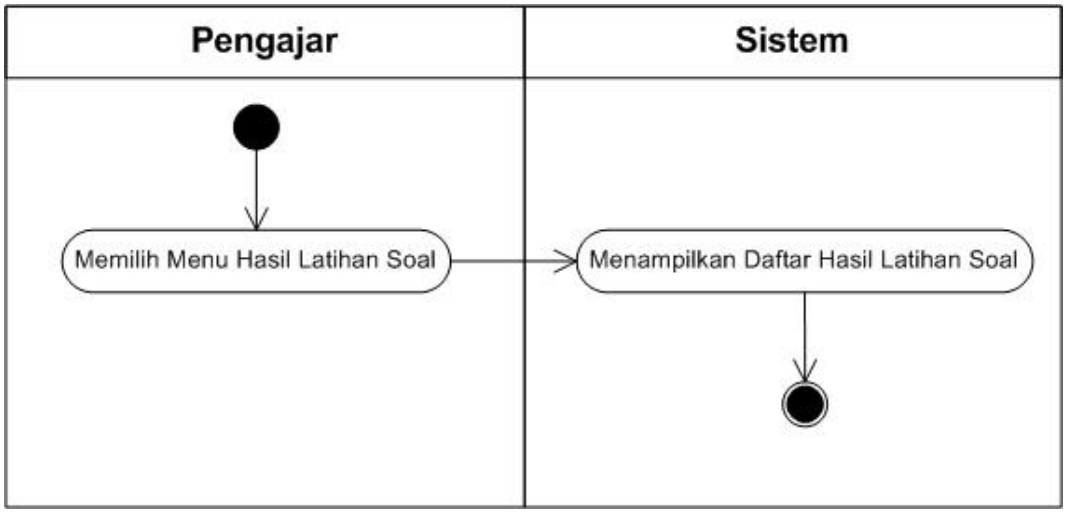
Aktifitas ubah data soal merupakan aktifitas yang dilakukan Pengajar (admin) untuk mengubah data soal. Seluruh tahapan aktifitas ubah data soal dapat dilihat pada gambar 3.199.



Gambar 3.19 ***Activity Diagram*** **Ubah Data Soal**

3.4.3.16. Hasil Latihan Soal

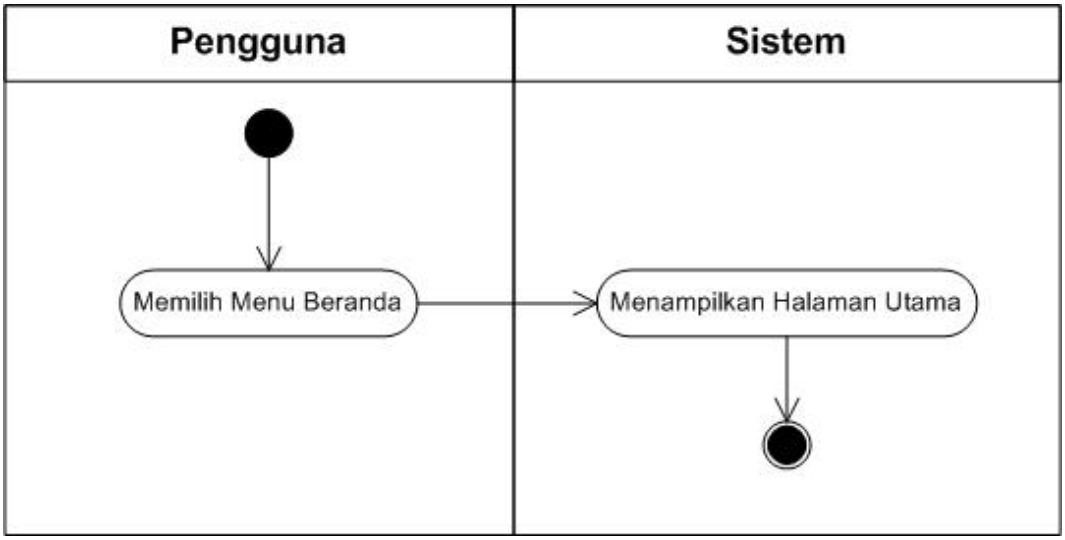
Aktifitas hasil latihan soal merupakan aktifitas yang dilakukan pengguna yaitu Pengajar (admin) untuk melihat hasil evaluasi dalam bentuk latihan soal. Seluruh tahapan aktifitas hasil latihan soal dapat dilihat pada gambar 3.20.



Gambar 3.20 *Activity Diagram* Hasil Latihan Soal

3.4.3.17. Halaman Utama

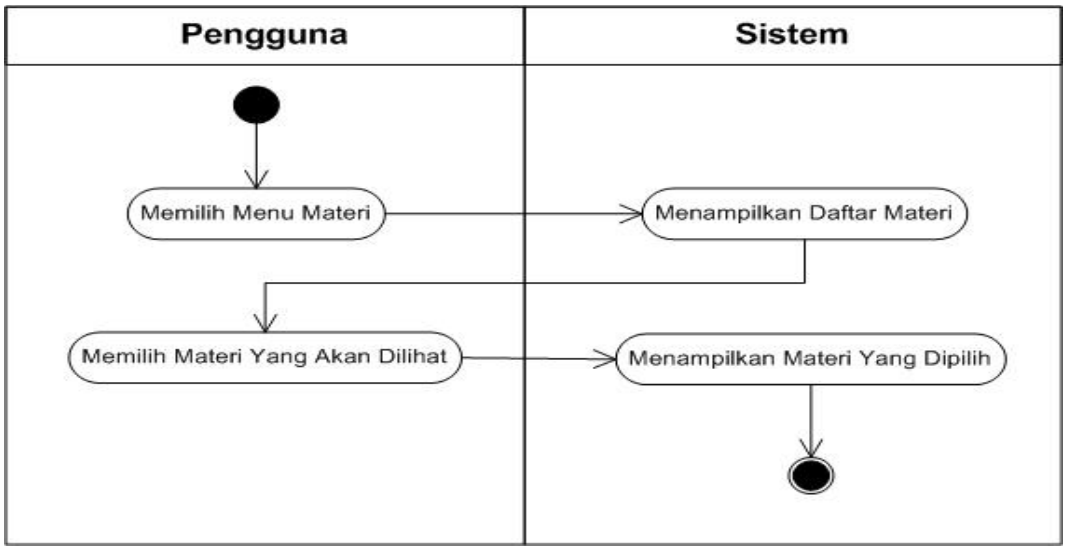
Aktivitas halaman utama merupakan aktifitas di mana halaman utama pada sistem akan ditampilkan kepada pengguna. Seluruh tahapan aktifitas hapus soal dapat dilihat pada gambar 3.21.



Gambar 3.21 *Activity Diagram* Halaman Utama

3.4.3.18. Lihat Materi

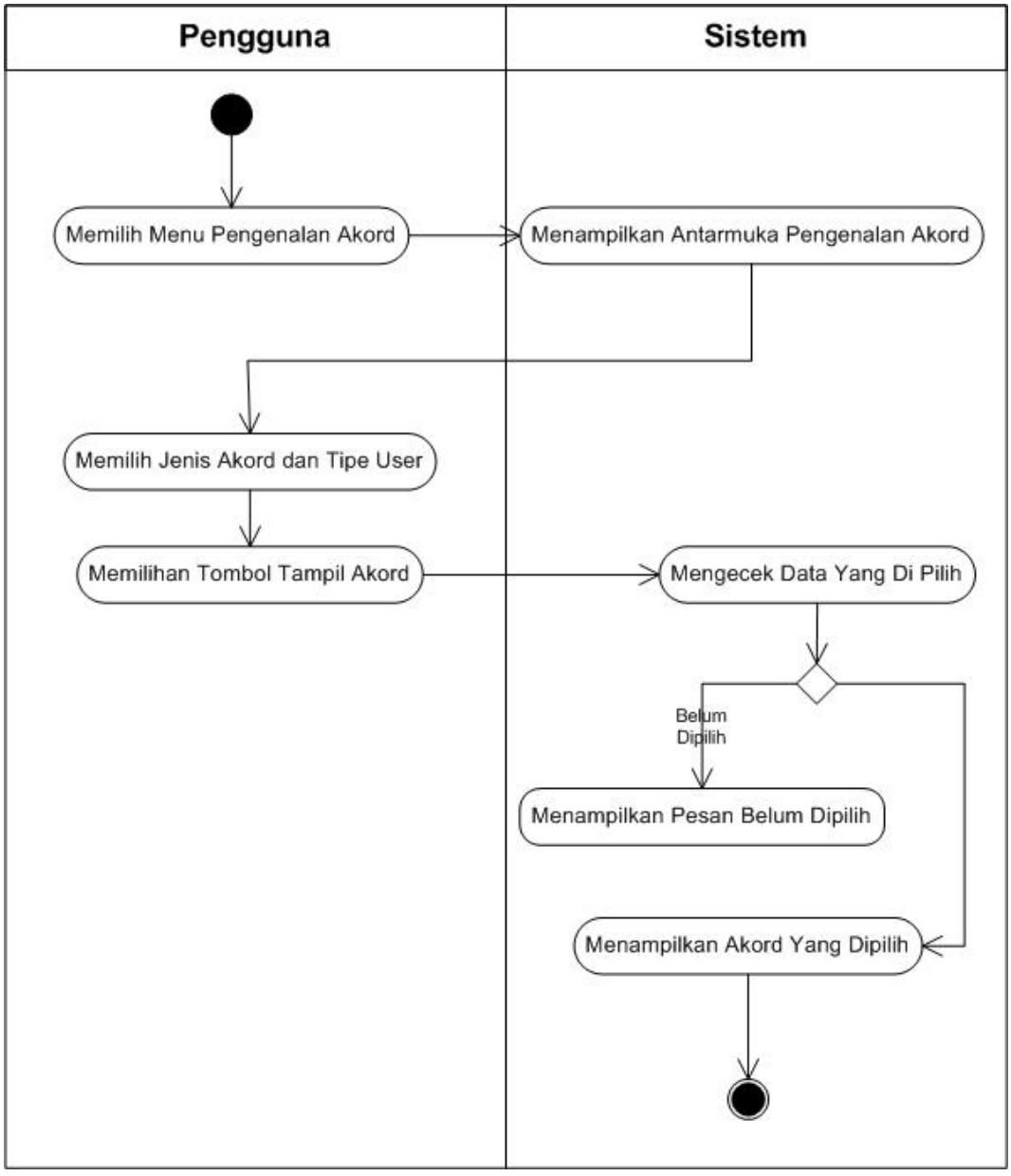
Aktifitas lihat materi merupakan aktifitas yang dapat dilakukan semua pengguna dalam melihat materi. Seluruh tahapan aktifitas lihat materi dapat dilihat pada gambar 3.22.



Gambar 3.22 *Activity Diagram* Lihat Materi

3.4.3.19. Pengenalan Akord

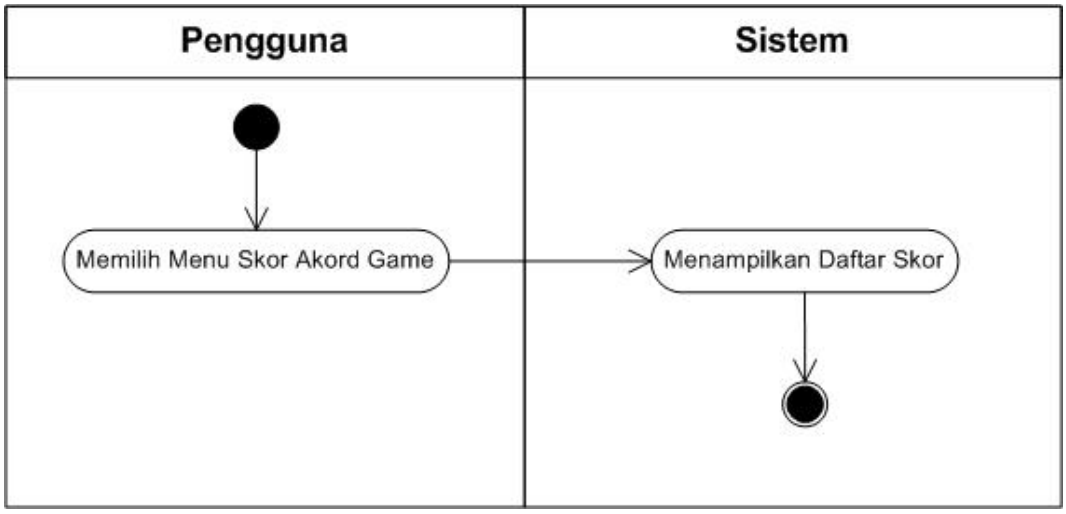
Aktifitas pengenalan akord merupakan aktifitas yang dilakukan murid untuk pengenalan akord dalam mempelajari bentuk akord. Murid dapat memilih nama akord yang akan ditampilkan sesuai jenis dan tipe pengguna. Seluruh tahapan aktifitas pengenalan akord dapat dilihat pada gambar 3.23.



Gambar 3.23 *Activity Diagram* Pengenalan Akord

3.4.3.20. Lihat Skor (Akord *Game*)

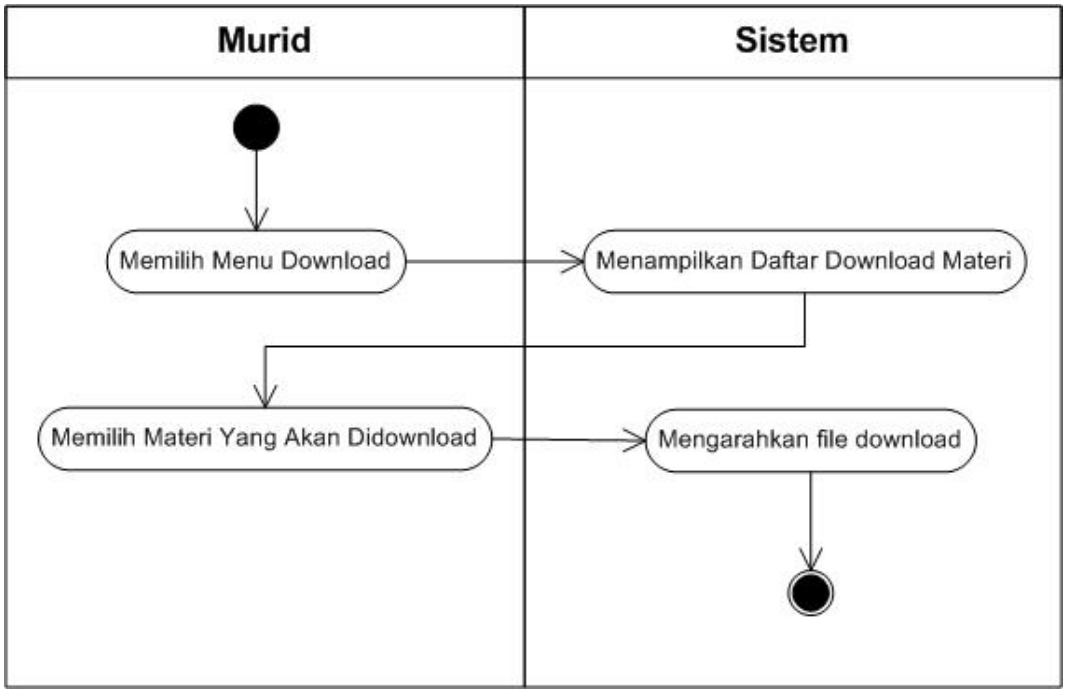
Aktifitas skor akord *game* merupakan aktifitas yang dilakukan murid untuk melihat skor tertinggi hasil dari *game* (Akord *Game*). Aktifitas akord *game* dapat dilihat pada sistem *website* dan *desktop base*. Seluruh tahapan aktifitas skor akord *game* dapat dilihat pada gambar 3.24.



Gambar 3.24 *Activity Diagram* Skor Akord *Game*

3.4.3.21. *Download* Materi

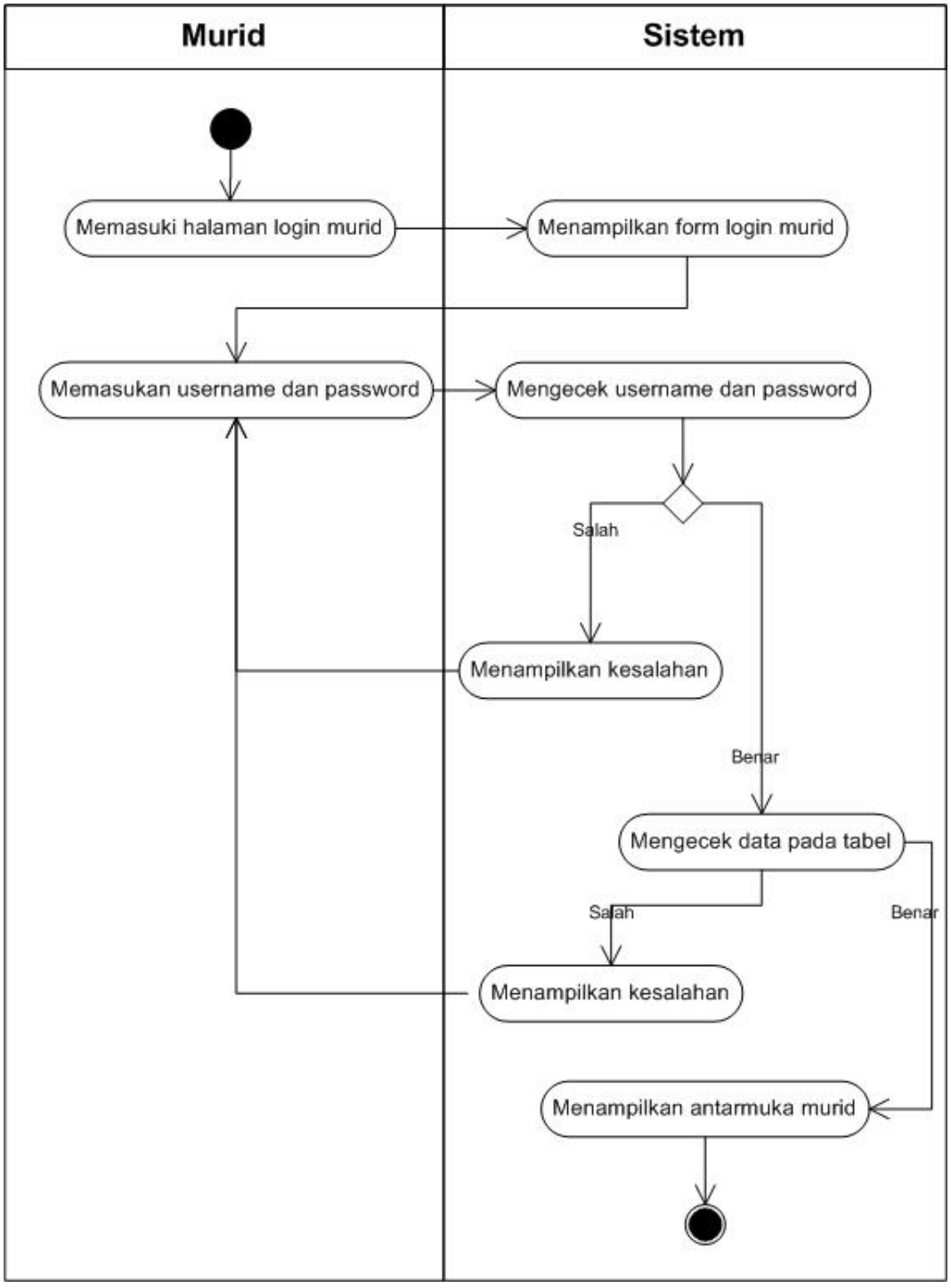
Aktifitas *download* materi merupakan aktifitas yang dapat dilakukan oleh semua pengguna untuk mengambil materi dalam bentuk file. Seluruh tahapan aktifitas *download* materi dapat dilihat pada gambar 3.25.



Gambar 3.25 *Activity Diagram Download* Materi

3.4.3.22. Autentikasi Murid

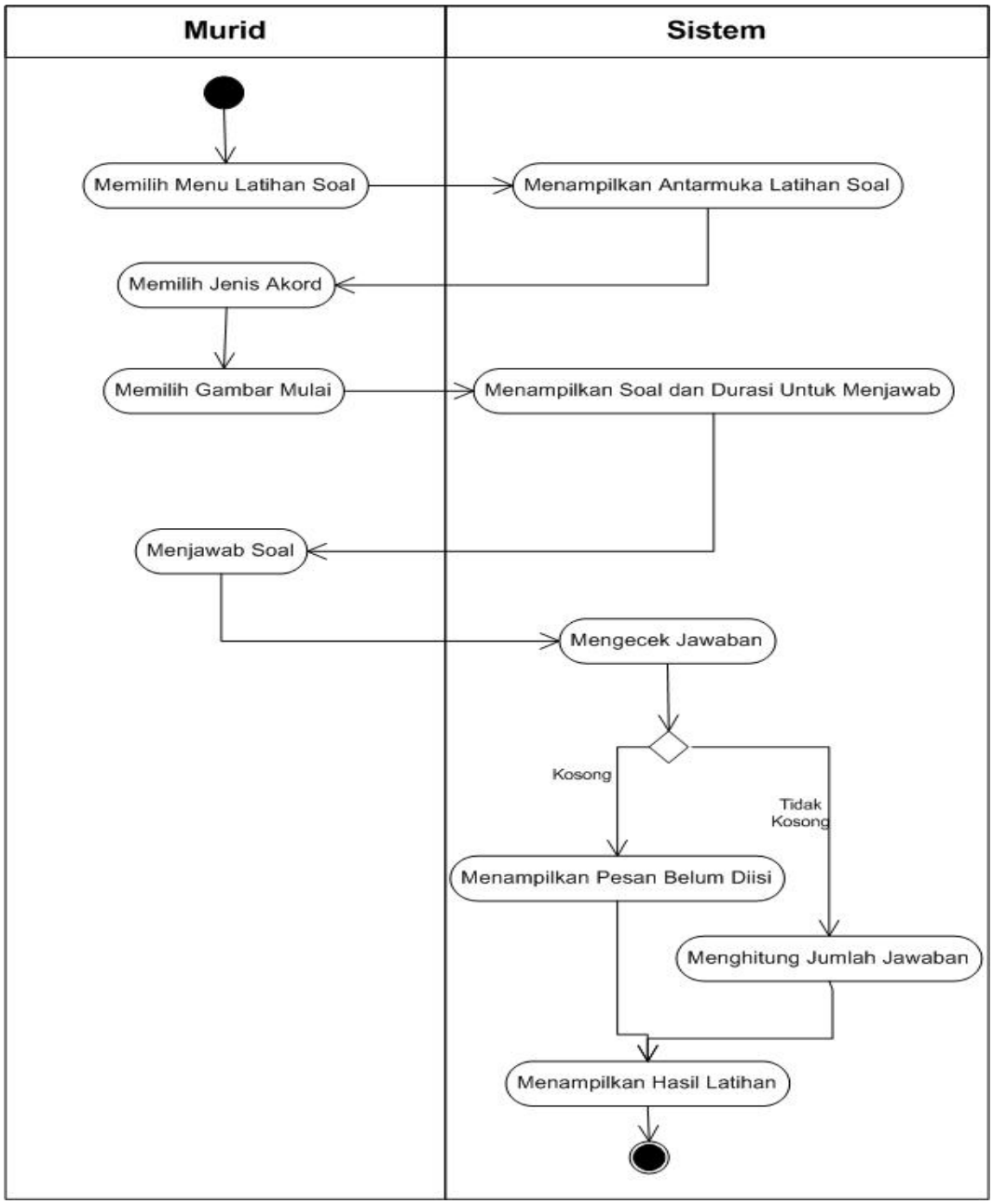
Aktifitas autentikasi murid merupakan aktifitas yang dilakukan oleh Murid untuk mendapatkan hak akses pada sistem. Aktivitas autentikasi murid berlaku pada sistem *website* dan desktop. Seluruh tahapan aktifitas autentikasi murid dapat dilihat pada gambar 3.26.



Gambar 3.26 *Activity Diagram* Autentikasi Murid

3.4.3.23. Latihan Soal

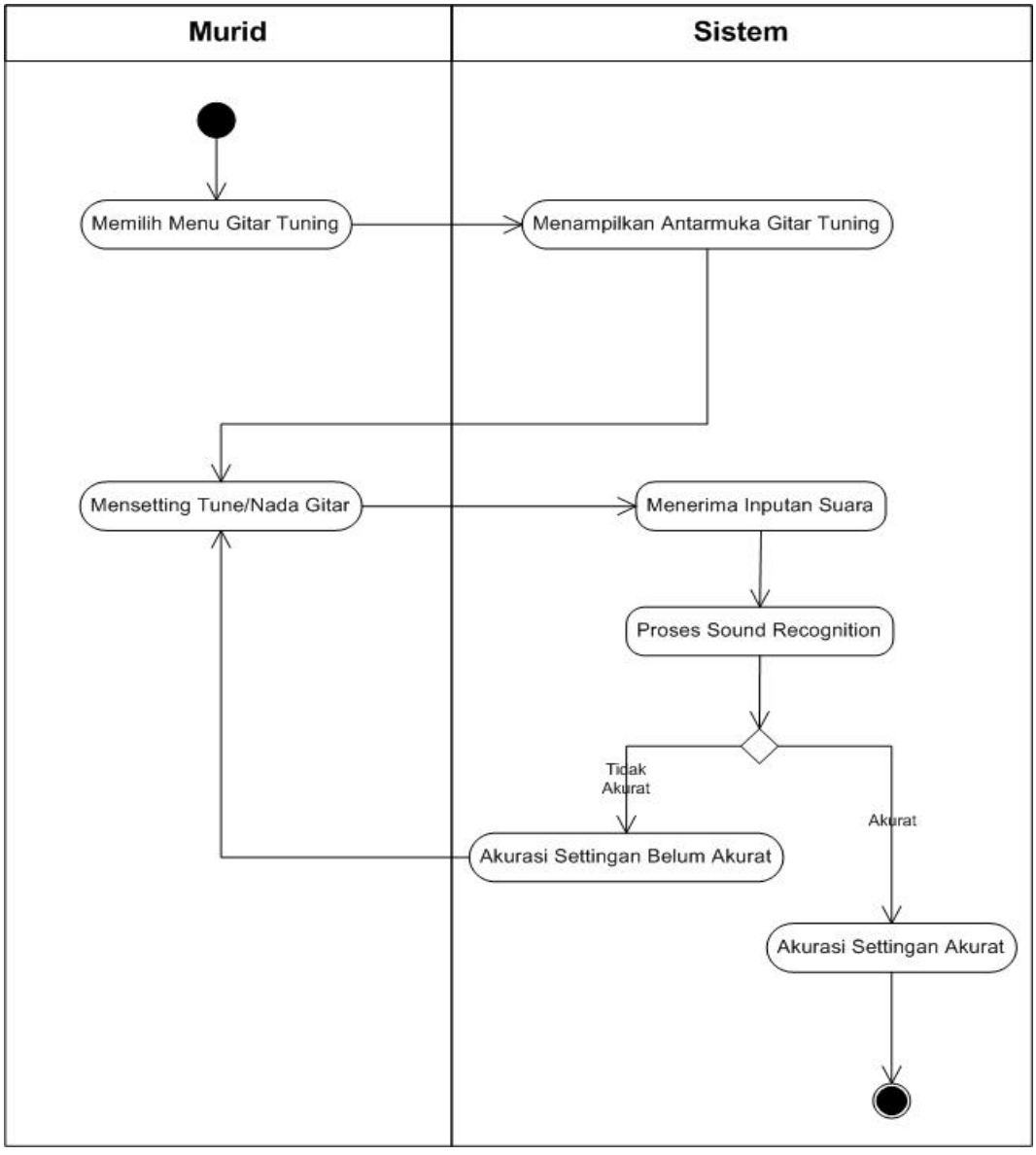
Aktifitas latihan soal merupakan aktifitas yang dilakukan murid untuk mempelajari akord gitar dalam bentuk tebak akord. Seluruh tahapan aktifitas latihan soal dapat dilihat pada gambar 3.27.



Gambar 3.27 *Activity Diagram* Latihan Soal

3.4.3.24. Gitar Tuning

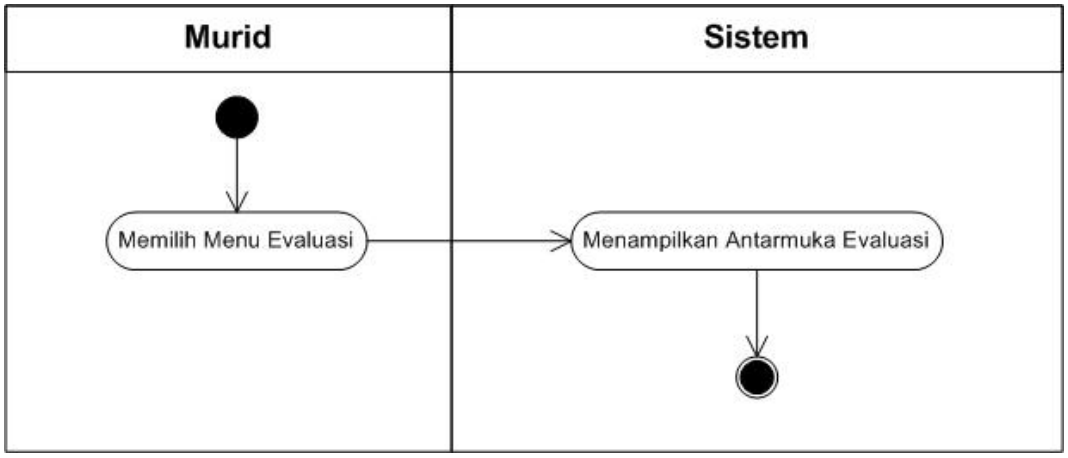
Aktifitas gitar *tuning* merupakan aktifitas yang dilakukan murid untuk mensetting *tune* (nada) gitarnya sesuai standar yang telah ditetapkan sistem. Gitar *tuning* terdapat pada sistem *desktop base*. Seluruh tahapan aktifitas gitar *tuning* dapat dilihat pada gambar 3.28.



Gambar 3.28 *Activity Diagram* Gitar *Tuning*

3.4.3.25. Evaluasi (Akord *Game*)

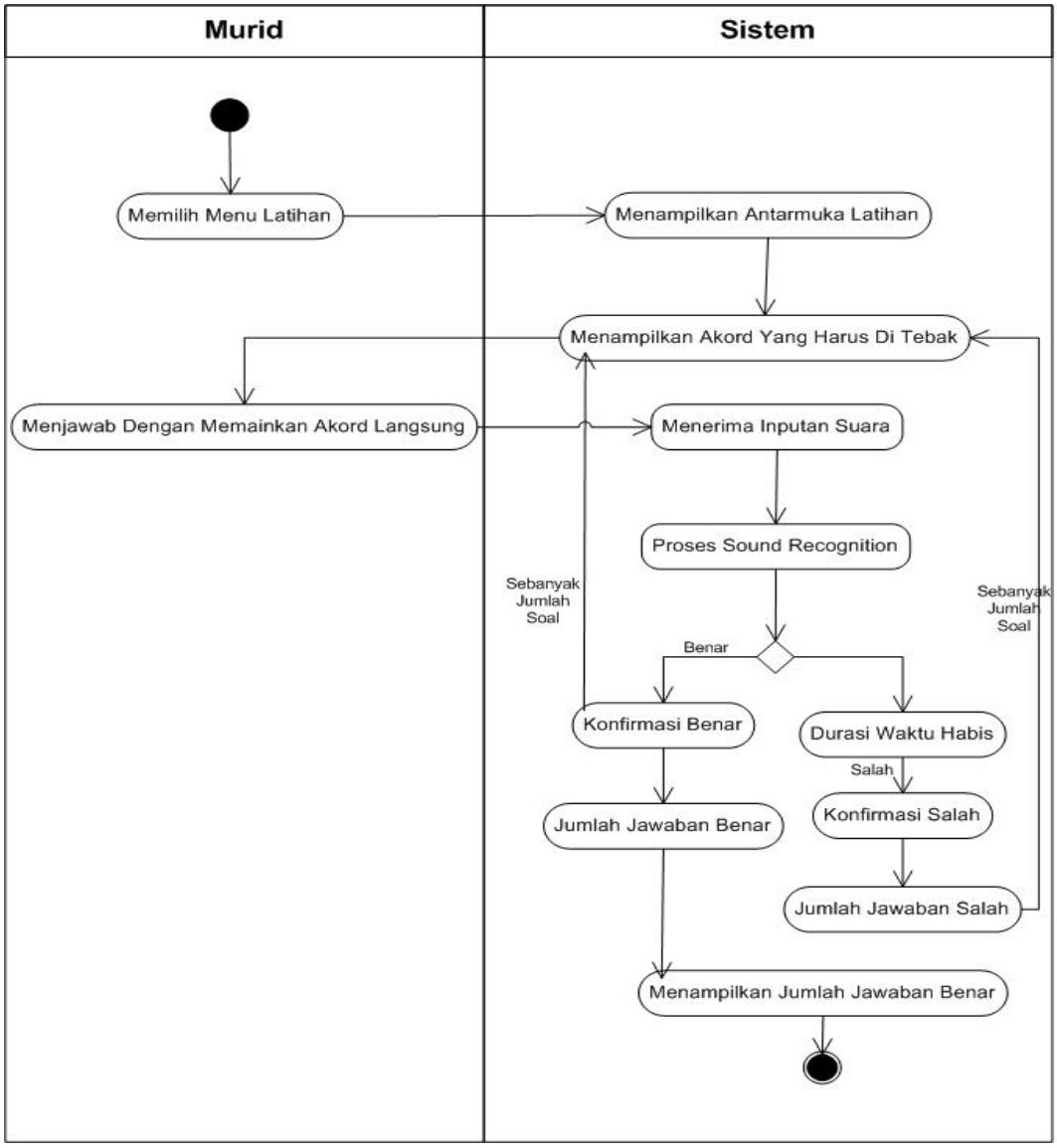
Aktifitas Evaluasi (Akord *Game*) merupakan aktifitas pembelajaran akord yang dapat diakses murid untuk memilih jenis evaluasi, yaitu latihan dan game. Seluruh tahapan aktifitas Evaluasi (Akord *Game*) dapat dilihat pada gambar 3.29.



Gambar 3.29 *Activity Diagram* Evaluasi (Akord *Game*)

3.4.3.26. Latihan (Akord *Game*)

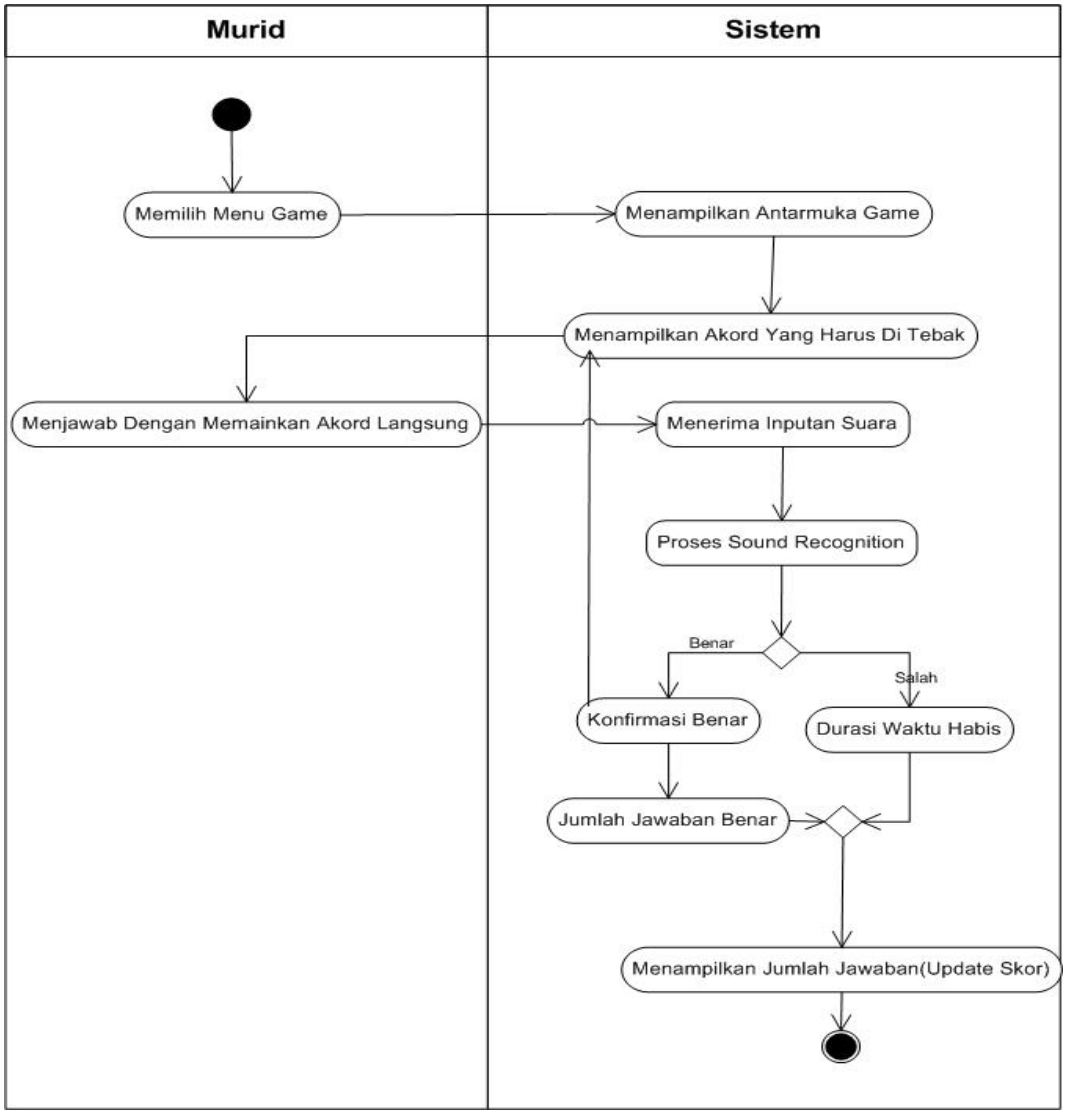
Aktifitas Latihan (Akord *Game*) merupakan aktifitas pembelajaran akord yang dapat diakses murid untuk mempelajari dan menguji kemampuan tentang akord gitar yang telah dipelajarinya. Seluruh tahapan aktifitas Latihan (Akord *Game*) dapat dilihat pada gambar 3.30.



Gambar 3.30 *Activity Diagram* Latihan (Akord *Game*)

3.4.3.27. Game (Akord *Game*)

Aktifitas *Game* (Akord *Game*) merupakan aktifitas pembelajaran akord yang dapat diakses murid untuk mempelajari dan menguji kemampuan tentang akord gitar yang telah dipelajarinya. Seluruh tahapan aktifitas *Game* (Akord *Game*) dapat dilihat pada gambar 3.31.



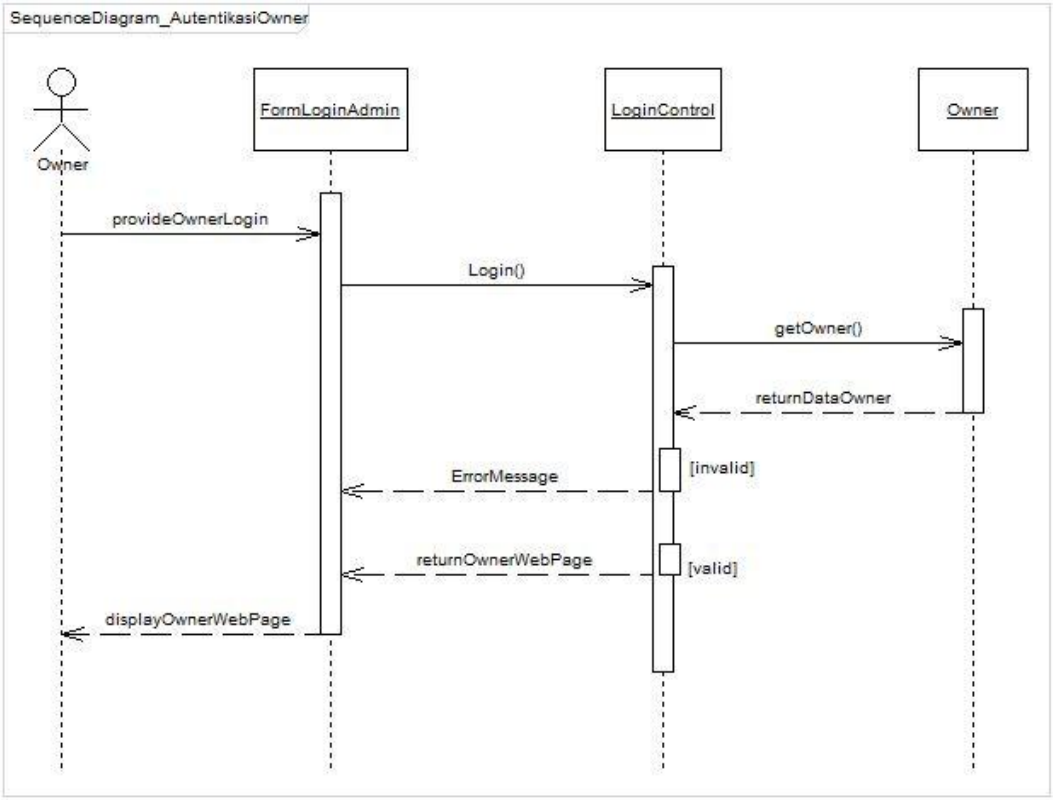
Gambar 3.31 *Activity Diagram Game* (Akord *Game*)

3.4.4. *Sequence Diagram*

Sequence diagram merupakan penerapan skenario sistem pembelajaran akord gitar dengan menunjukan interaksi dengan menampilkan setiap partisipan garis alir secara vertikal dan pengurutan pesan dari atas ke bawah.

3.4.4.1. *Sequence Diagram* Autentikasi *Owner*

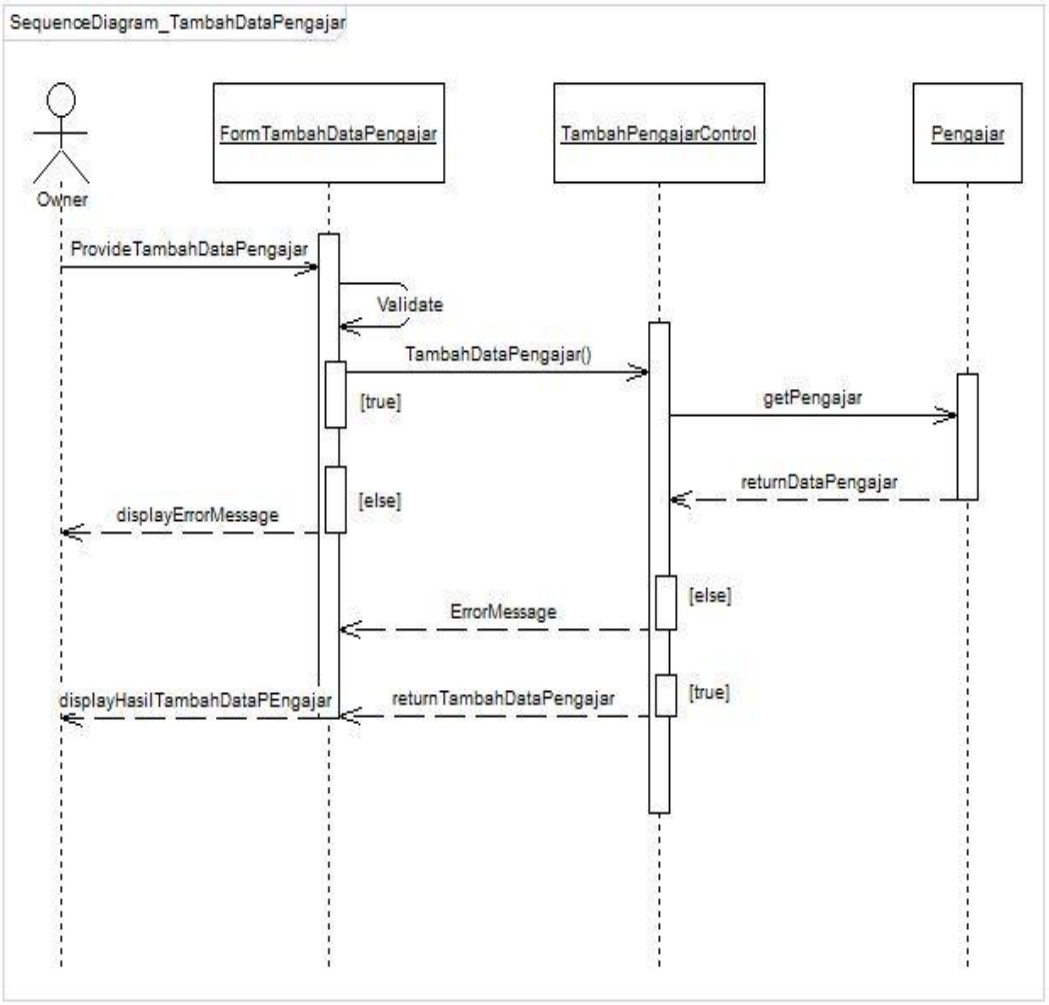
Interaksi antar kelas yang terjadi pada saat proses autentikasi *Owner* dapat dilihat pada gambar 3.32.



Gambar 3.32 *Sequence Diagram* Autentikasi *Owner*

3.4.4.2. *Sequence Diagram* Tambah Data Pengajar

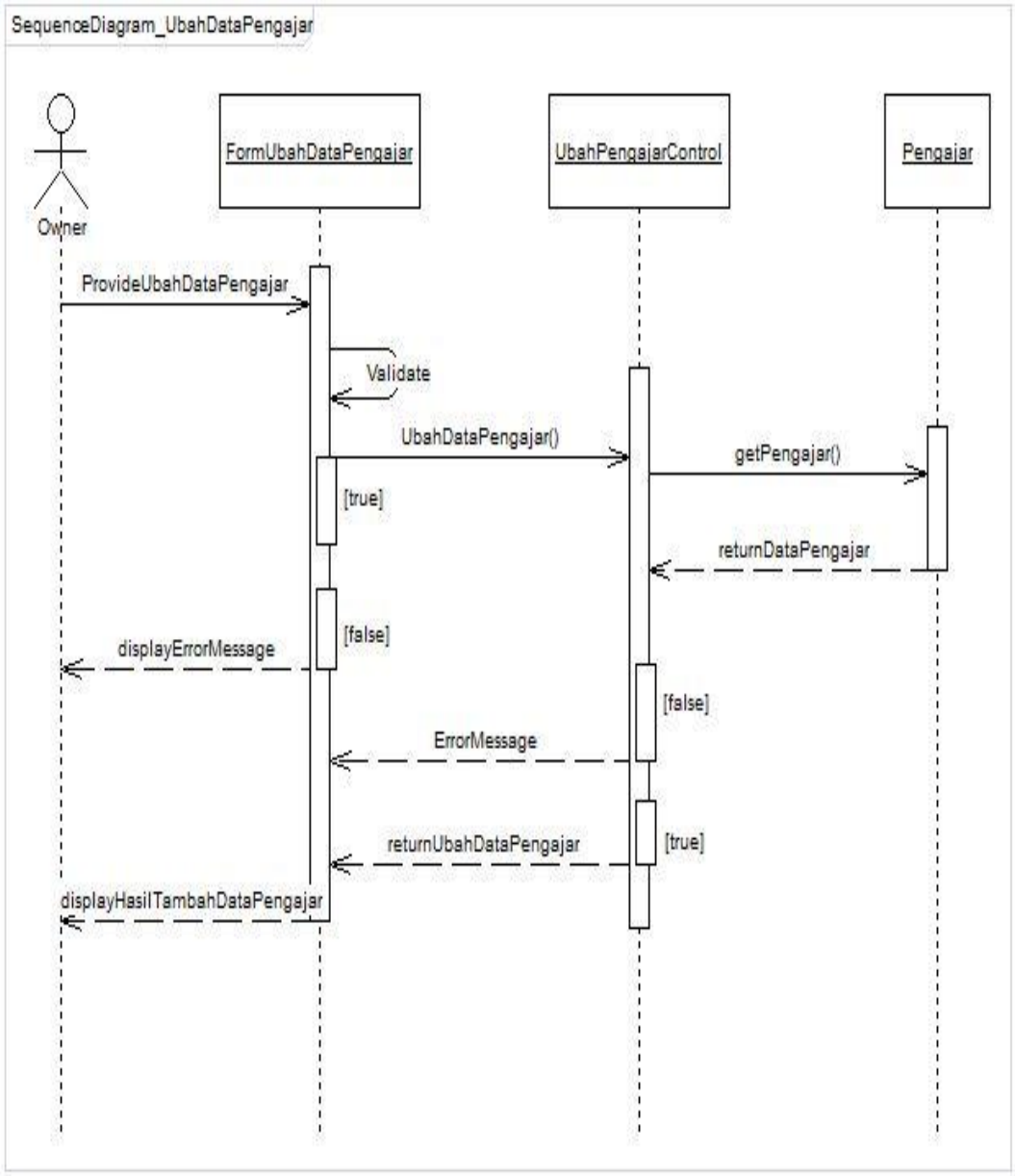
Interaksi antar kelas yang terjadi pada saat proses tambah data pengajar dapat dilihat pada gambar 3.33.



Gambar 3.33 *Sequence Diagram* Tambah Data Pengajar

3.4.4.3. *Sequence Diagram* Ubah Data Pengajar

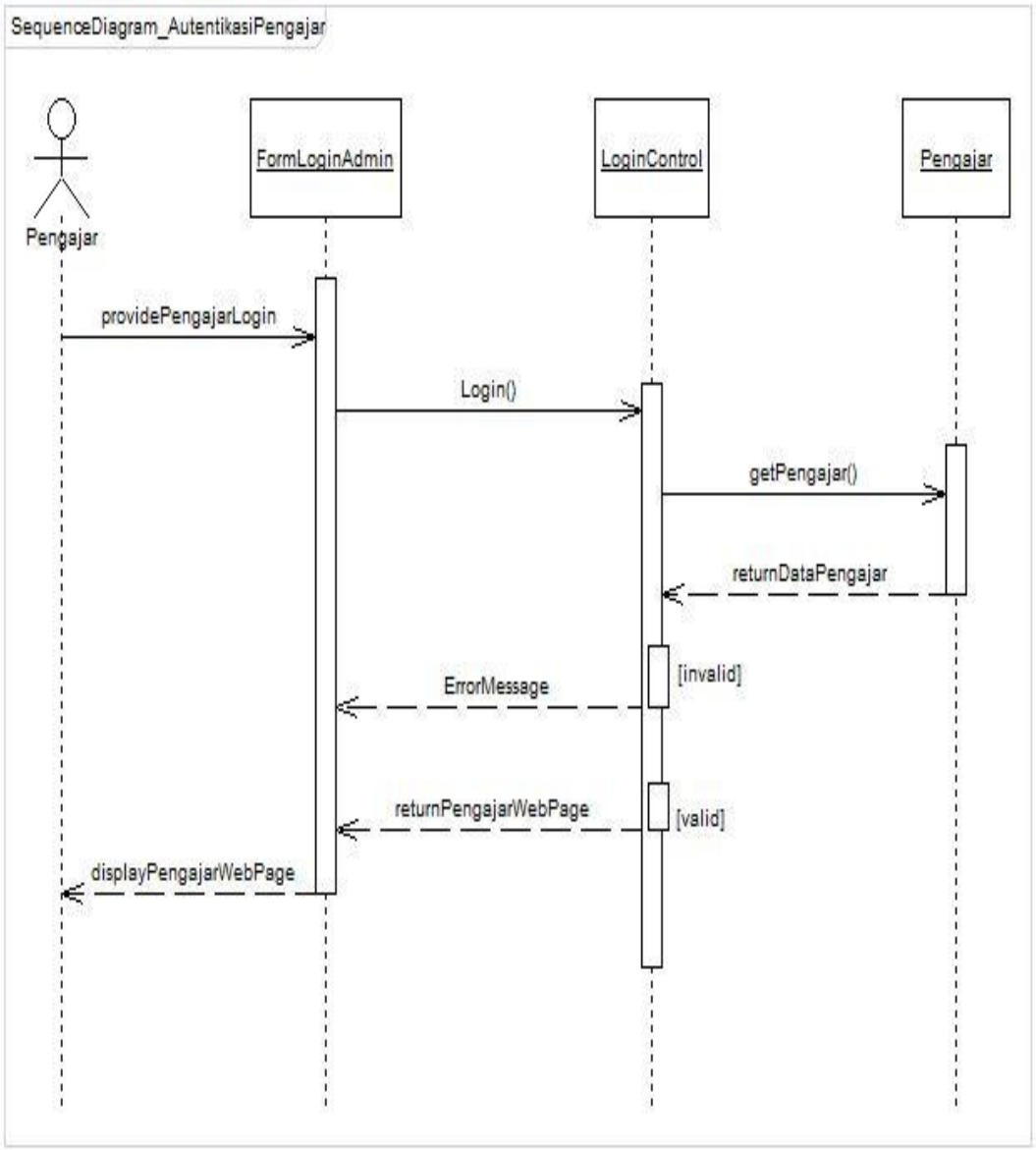
Interaksi antar kelas yang terjadi pada saat proses ubah data pengajar dapat dilihat pada gambar 3.34.



Gambar 3.34 *Sequence Diagram* Ubah Data Pengajar

3.4.4.4. *Sequence Diagram* Autentikasi Pengajar

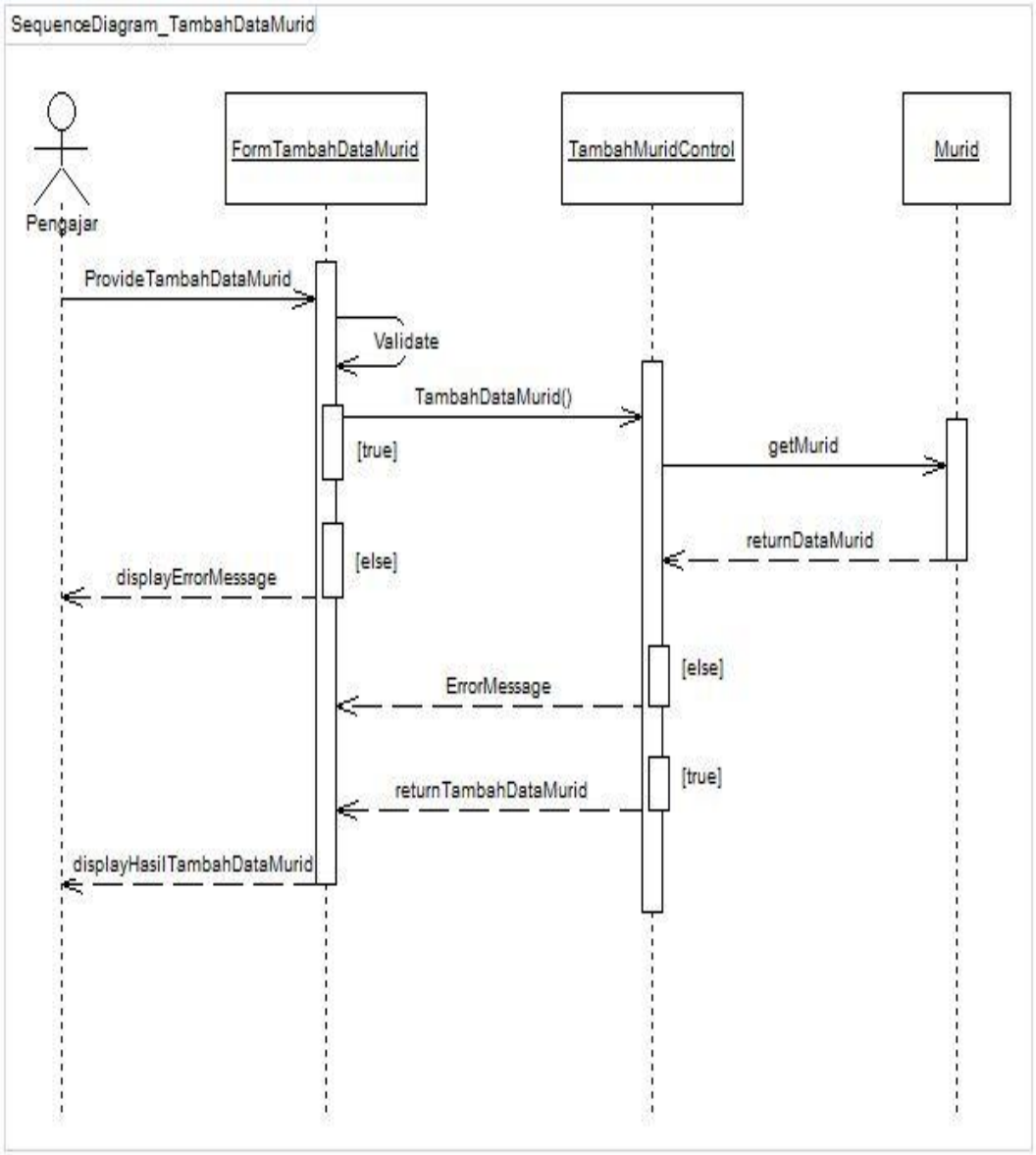
Interaksi antar kelas yang terjadi pada saat proses autentikasi Pengajar dapat dilihat pada gambar 3.35.



Gambar 3.35 *Sequence Diagram* Autentikasi Pengajar

3.4.4.5. *Sequence Diagram* Tambah Data Murid

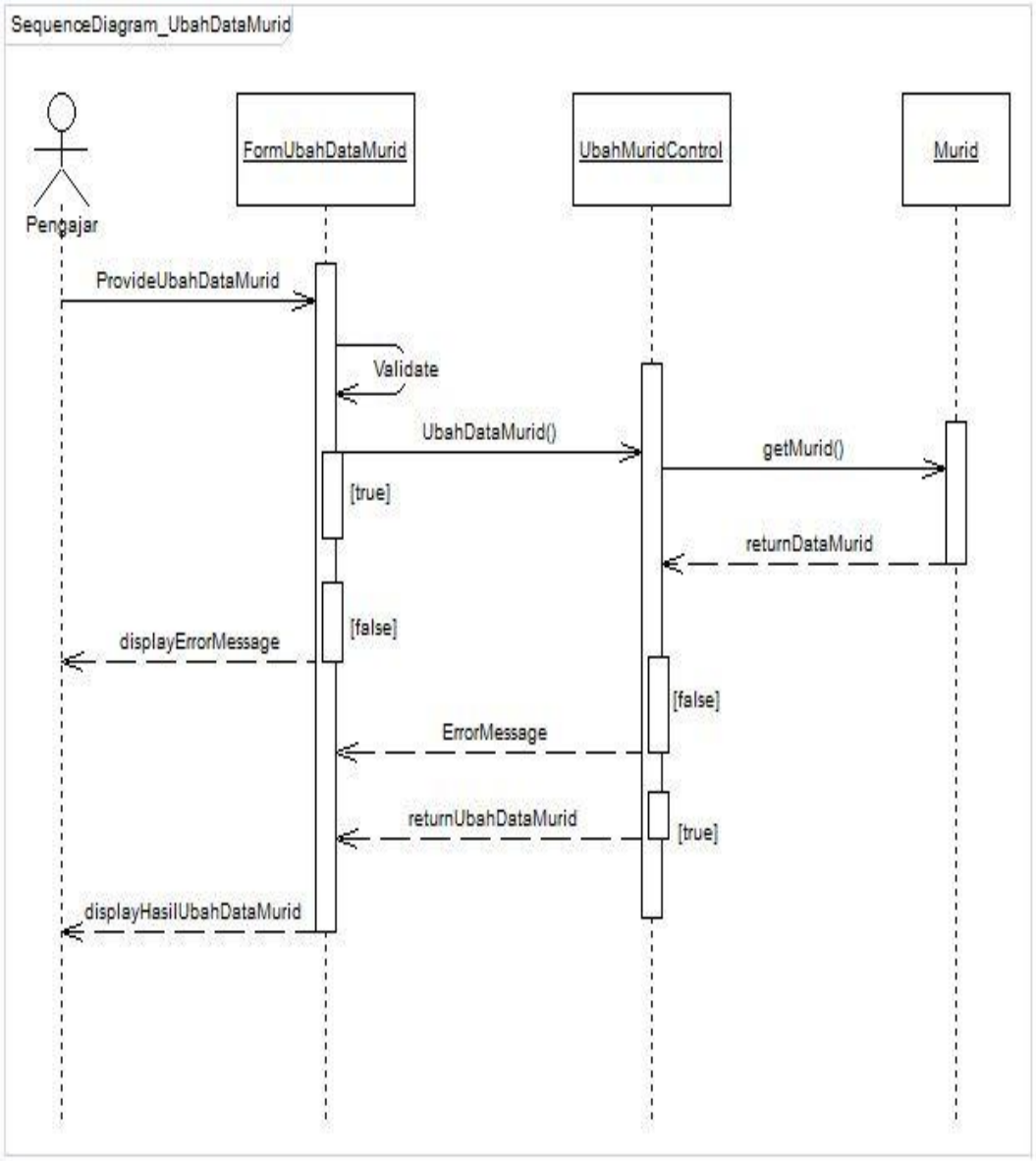
Interaksi antar kelas yang terjadi pada saat proses tambah data murid dapat dilihat pada gambar 3.36.



Gambar 3.36 *Sequence Diagram* Tambah Data Murid

3.4.4.6. *Sequence Diagram* Ubah Data Murid

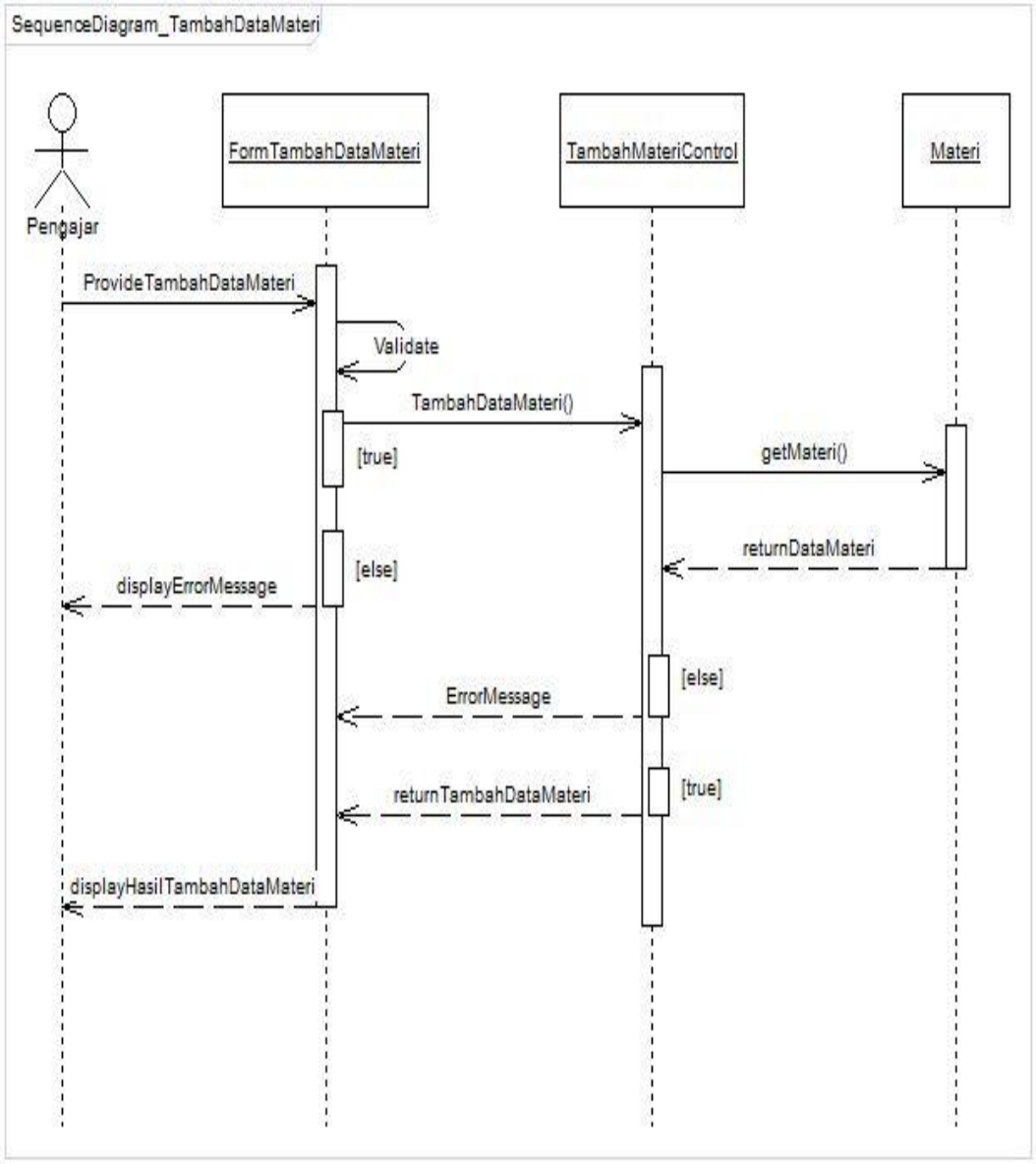
Interaksi antar kelas yang terjadi pada saat proses ubah data murid dapat dilihat pada gambar 3.37.



Gambar 3.37 *Sequence Diagram* Ubah Data Murid

3.4.4.7. *Sequence Diagram* Tambah Data Materi

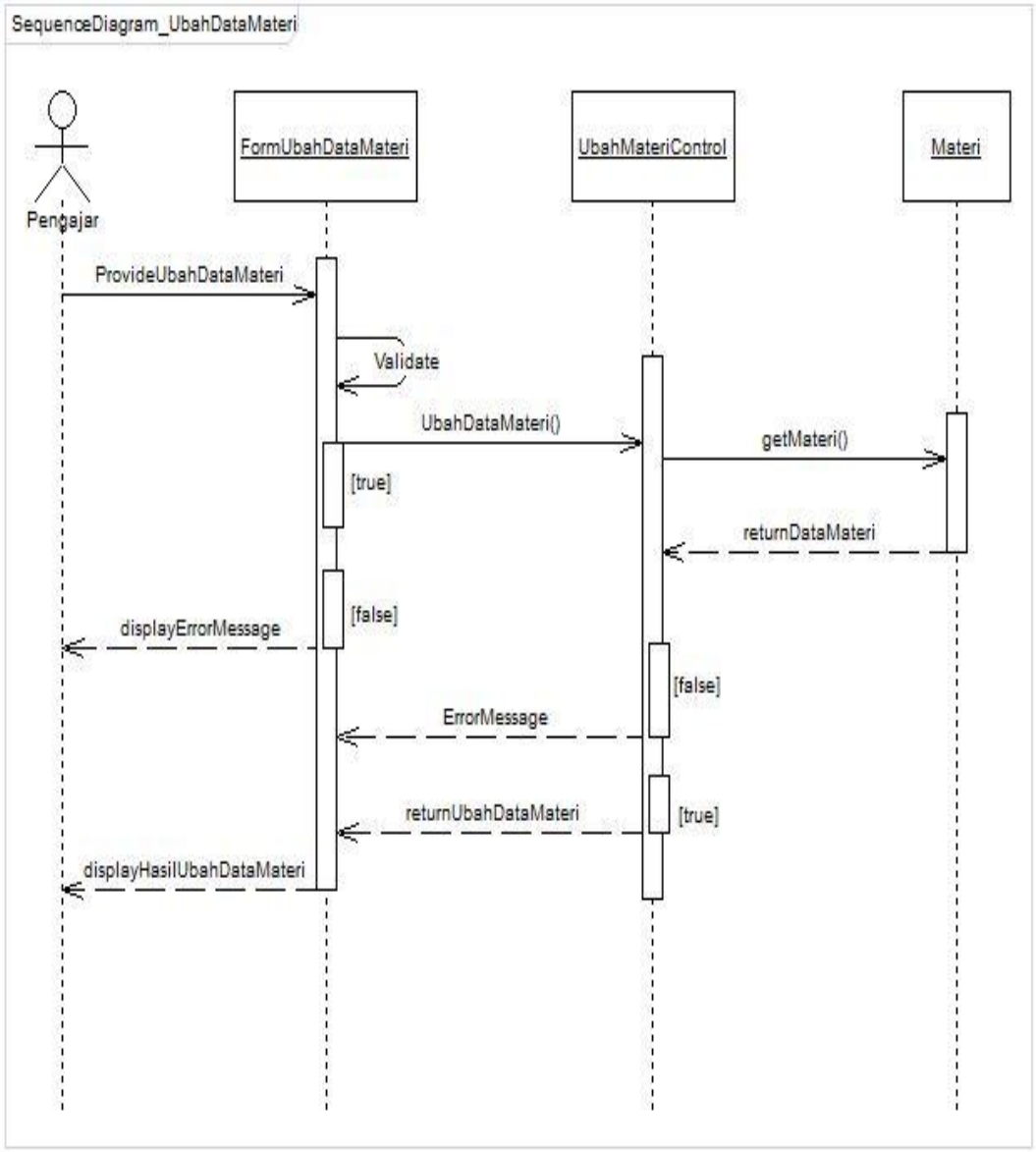
Interaksi antar kelas yang terjadi pada saat proses tambah data materi dapat dilihat pada gambar 3.38.



Gambar 3.38 *Sequence Diagram* Tambah Data Materi

3.4.4.8. *Sequence Diagram* Ubah Data Materi

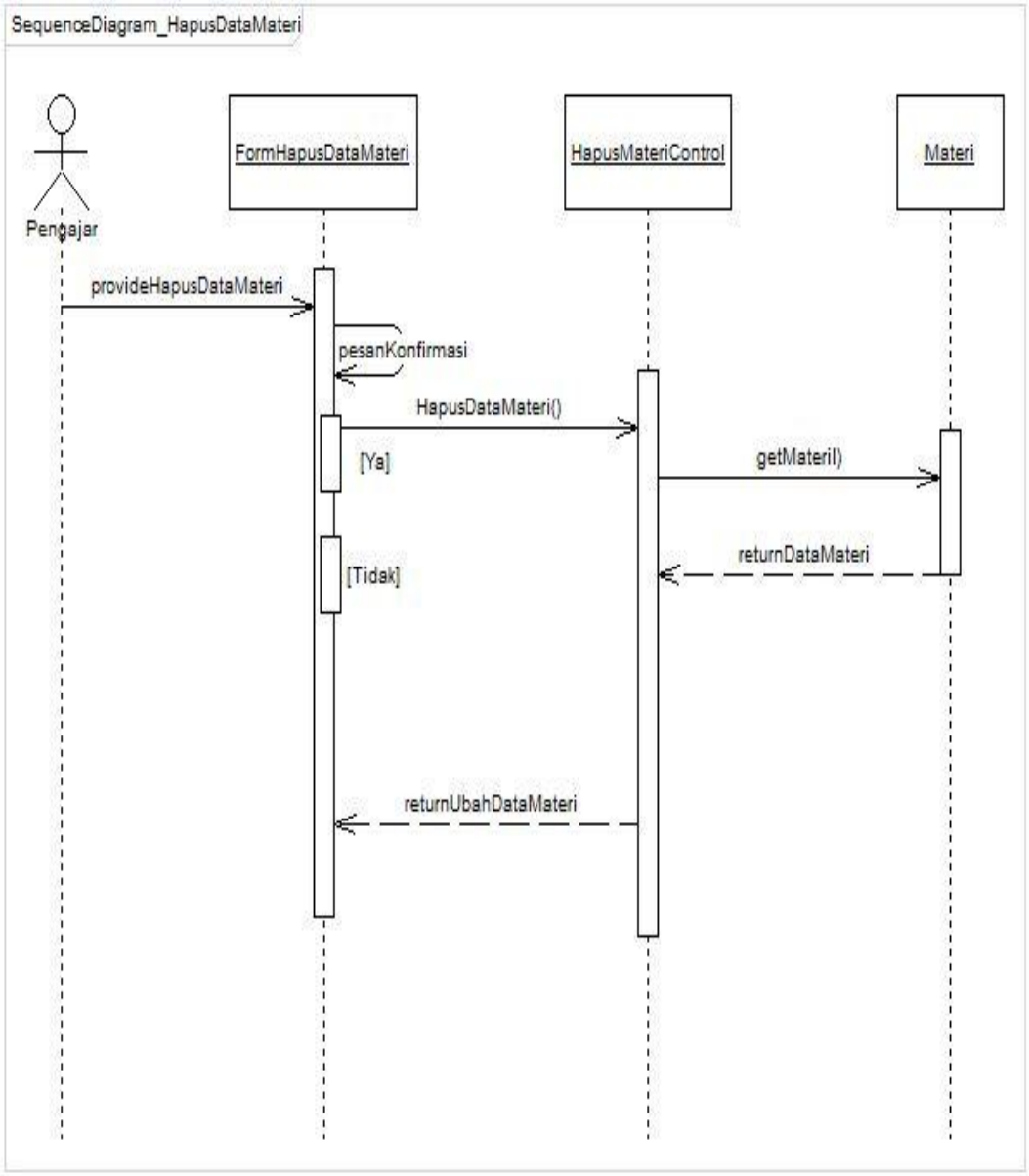
Interaksi antar kelas yang terjadi pada saat proses ubah data materi dapat dilihat pada gambar 3.39.



Gambar 3.39 *Sequence Diagram* Ubah Data Materi

3.4.4.9. *Sequence Diagram* Hapus Data Materi

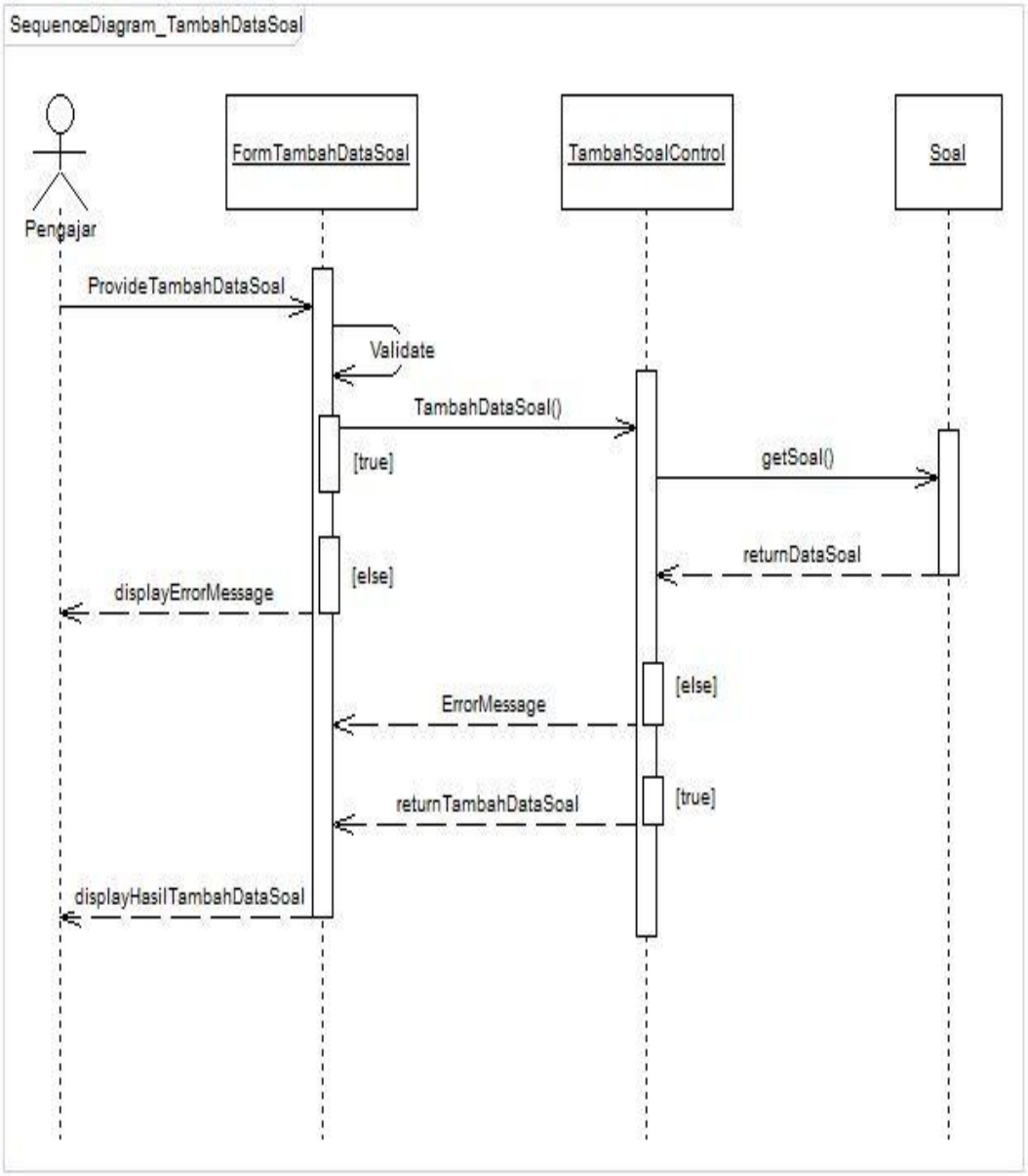
Interaksi antar kelas yang terjadi pada saat proses hapus data soal materi dilihat pada gambar 3.40.



Gambar 3.40 *Sequence Diagram* Hapus Data Materi

3.4.4.10. *Sequence Diagram* Tambah Data Soal

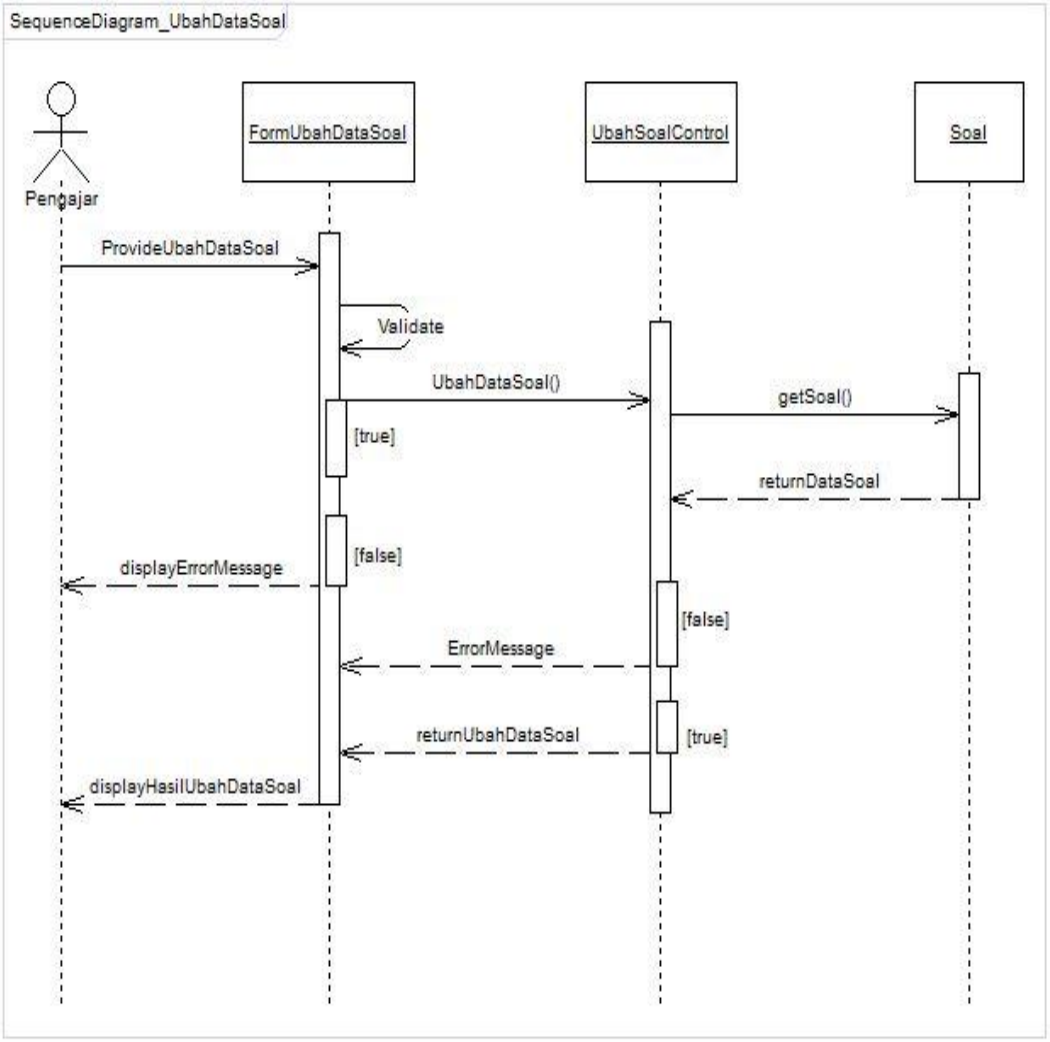
Interaksi antar kelas yang terjadi pada saat proses tambah data soal dapat dilihat pada gambar 3.41.



Gambar 3.41 *Sequence Diagram* Tambah Data Soal

3.4.4.11. *Sequence Diagram* Ubah Data Soal

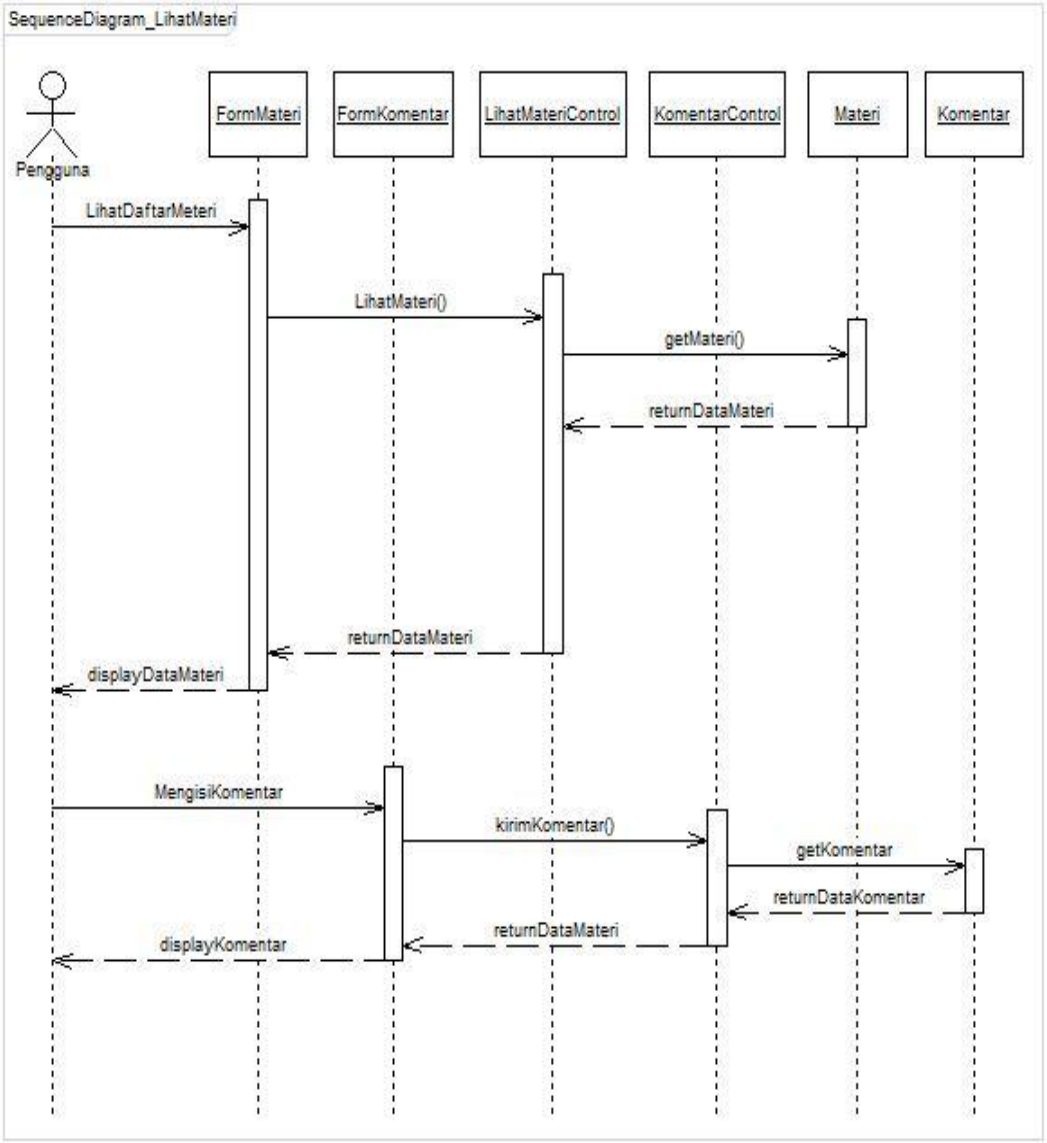
Interaksi antar kelas yang terjadi pada saat proses ubah data soal dapat dilihat pada gambar 3.42.



Gambar 3.42 *Sequence Diagram* Ubah Data Soal

3.4.4.12. *Sequence diagram* Lihat Materi

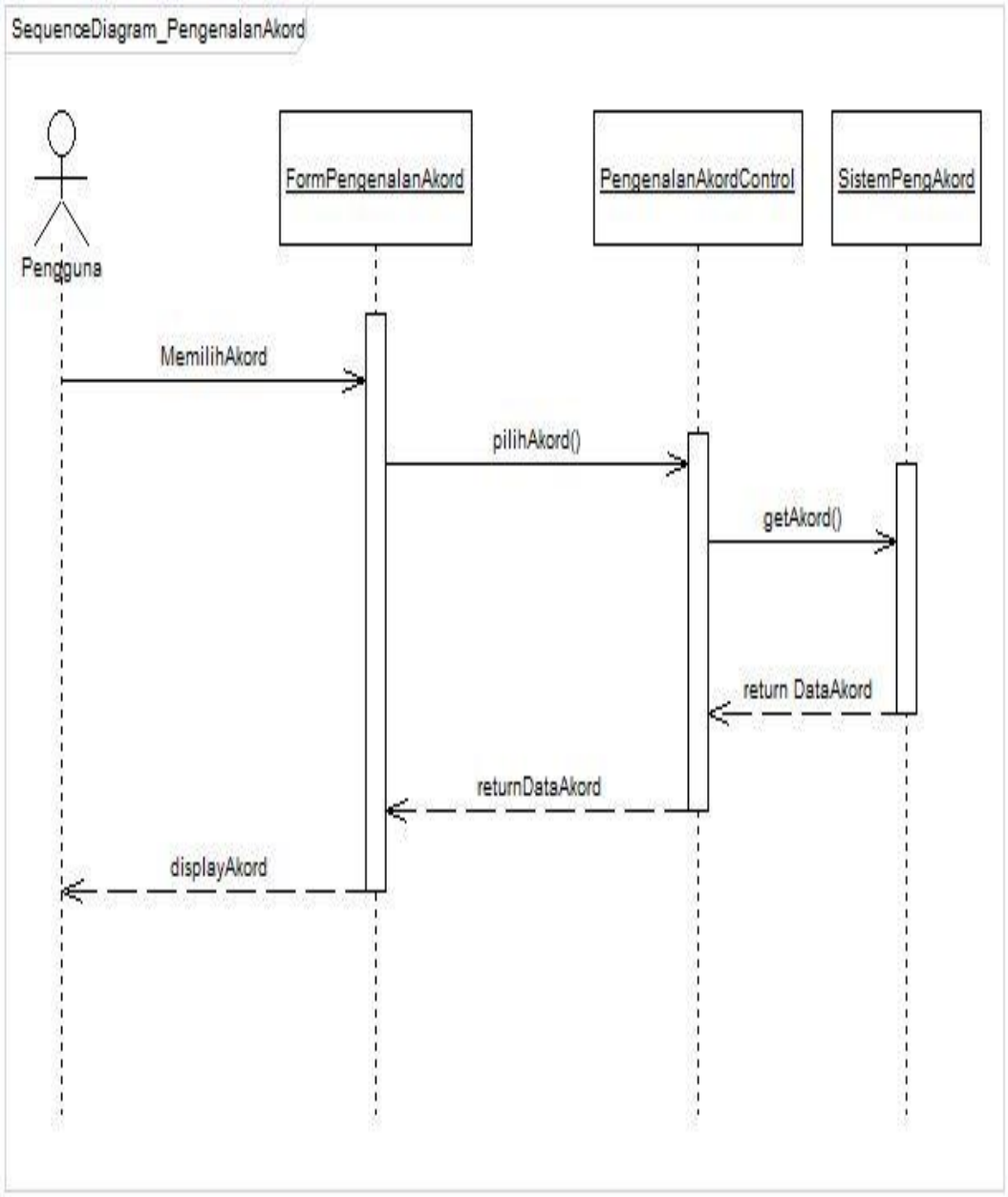
Interaksi antar kelas yang terjadi pada saat proses lihat materi dapat dilihat pada gambar 3.43.



Gambar 3.43 *Sequence Diagram* Lihat Materi

3.4.4.13. *Sequence Diagram* Pengenalan Akord

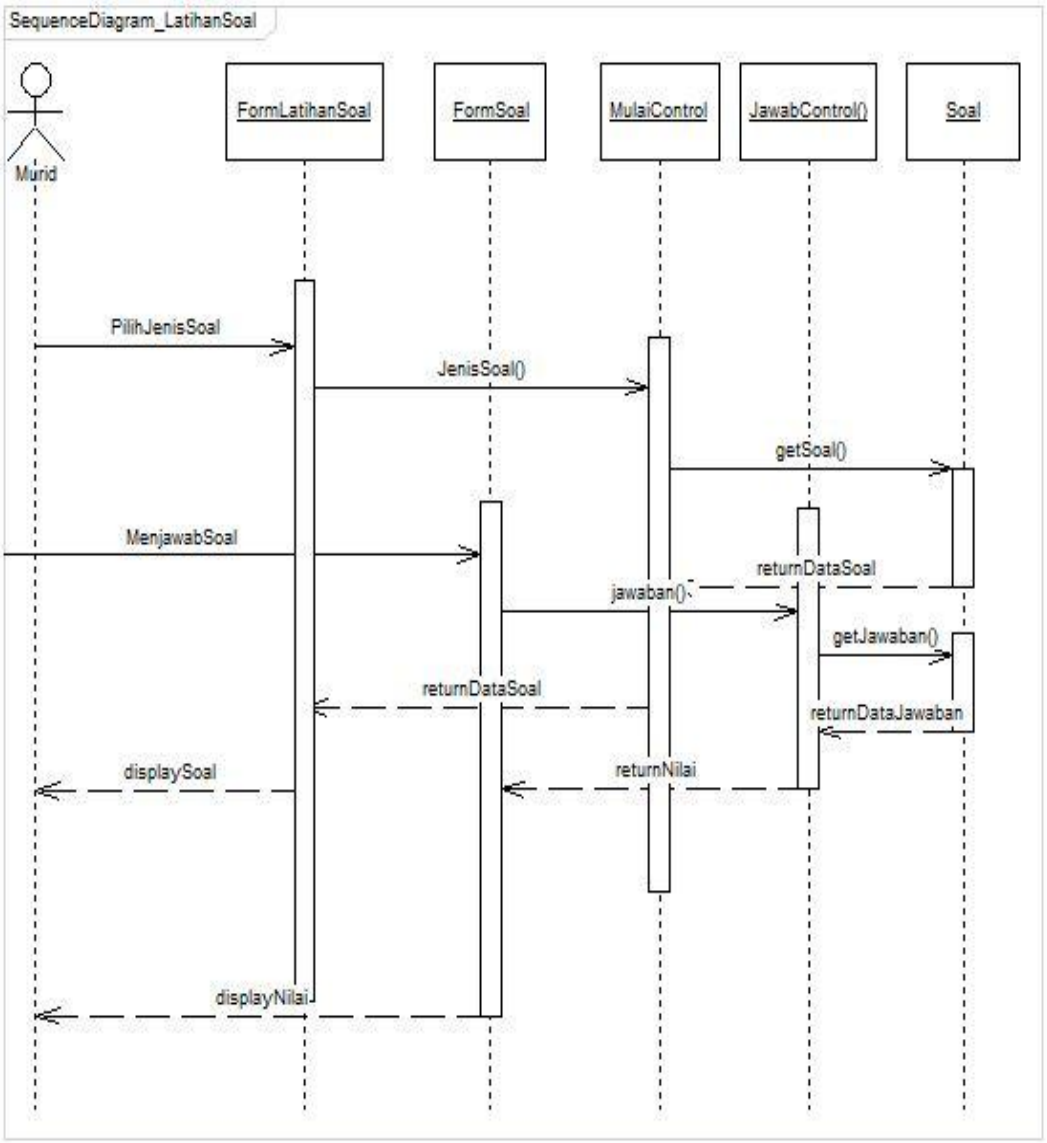
Interaksi antar kelas yang terjadi pada saat proses pengenalan akord dapat dilihat pada gambar 3.44.



Gambar 3.44 *Sequence Diagram* Pengenalan Akord

3.4.4.14. *Sequence Diagram* Latihan Soal

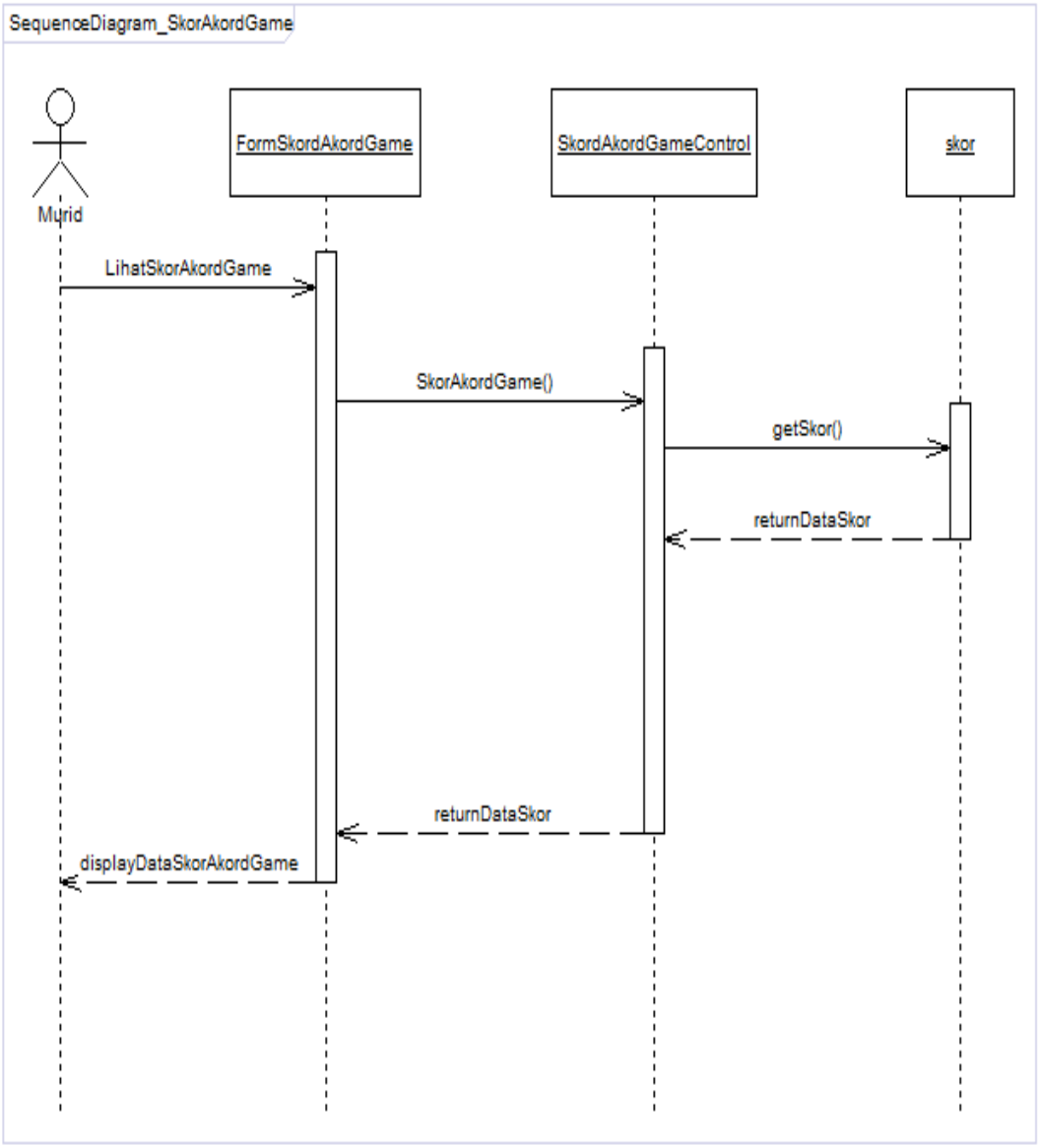
Interaksi antar kelas yang terjadi pada saat proses latihan soal dapat dilihat pada gambar 3.45.



Gambar 3.45 *Sequence Diagram* Latihan Soal

3.4.4.15. *Sequence Diagram* Skor Akord *Game*

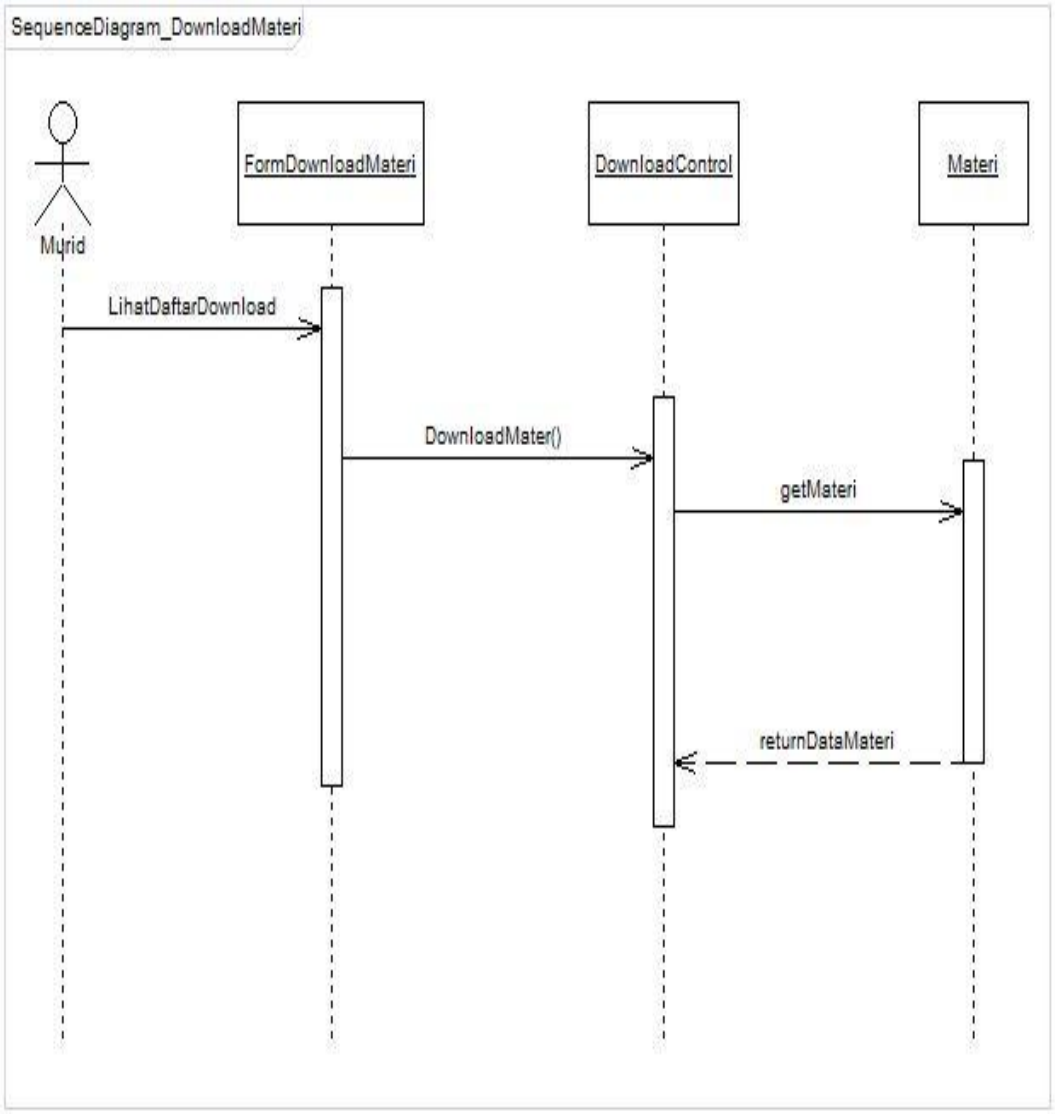
Interaksi antar kelas yang terjadi pada saat proses skor akord *game* dapat dilihat pada gambar 3.46.



Gambar 3.46 *Sequence Diagram* Skor Akord *Game*

3.4.4.16. *Sequence Diagram Download* Materi

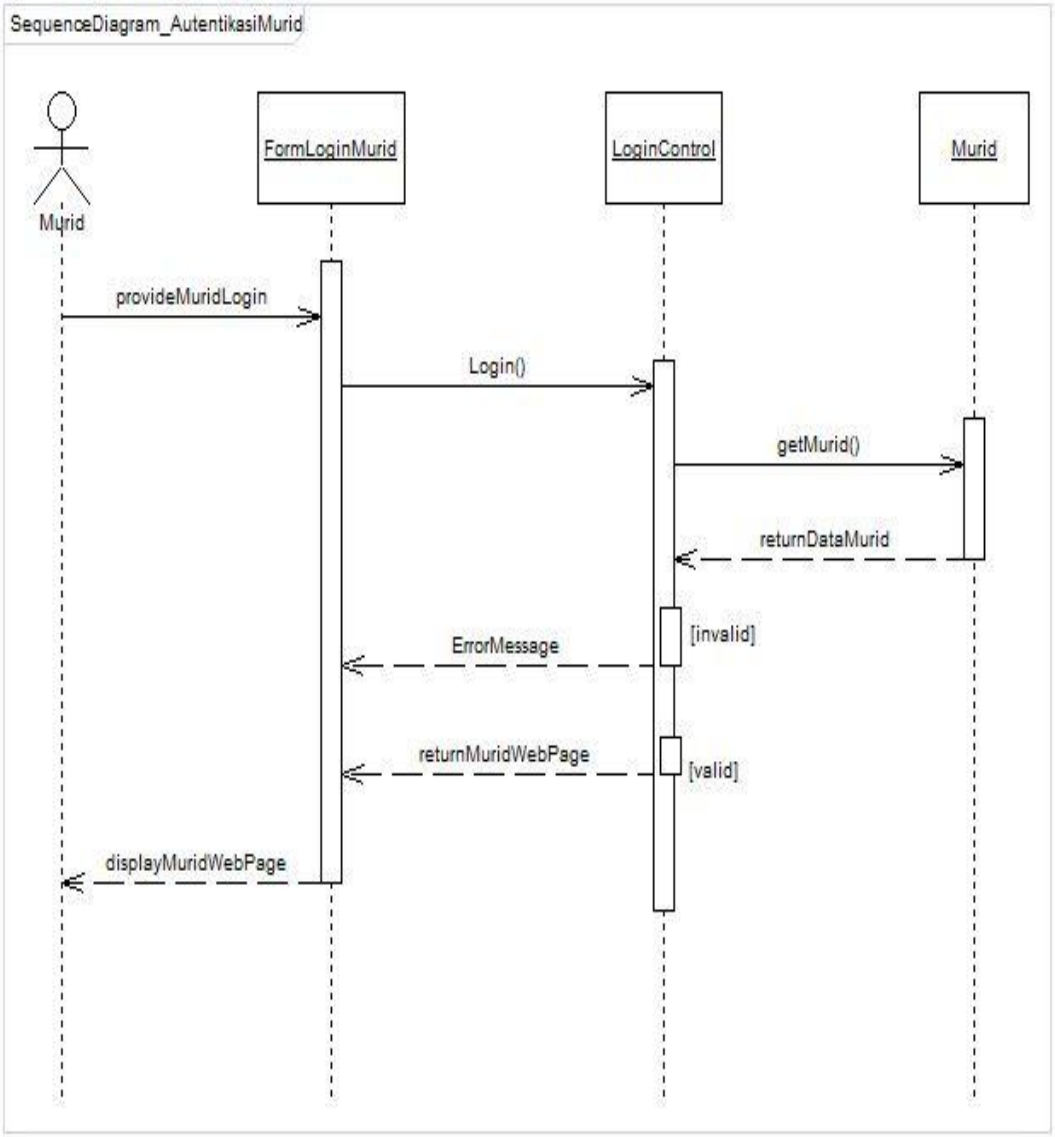
Interaksi antar kelas yang terjadi pada saat proses *download* materi dapat dilihat pada gambar 3.47.



Gambar 3.47 *Sequence Diagram Download* Materi

3.4.4.17. *Sequence Diagram* Autentikasi Murid

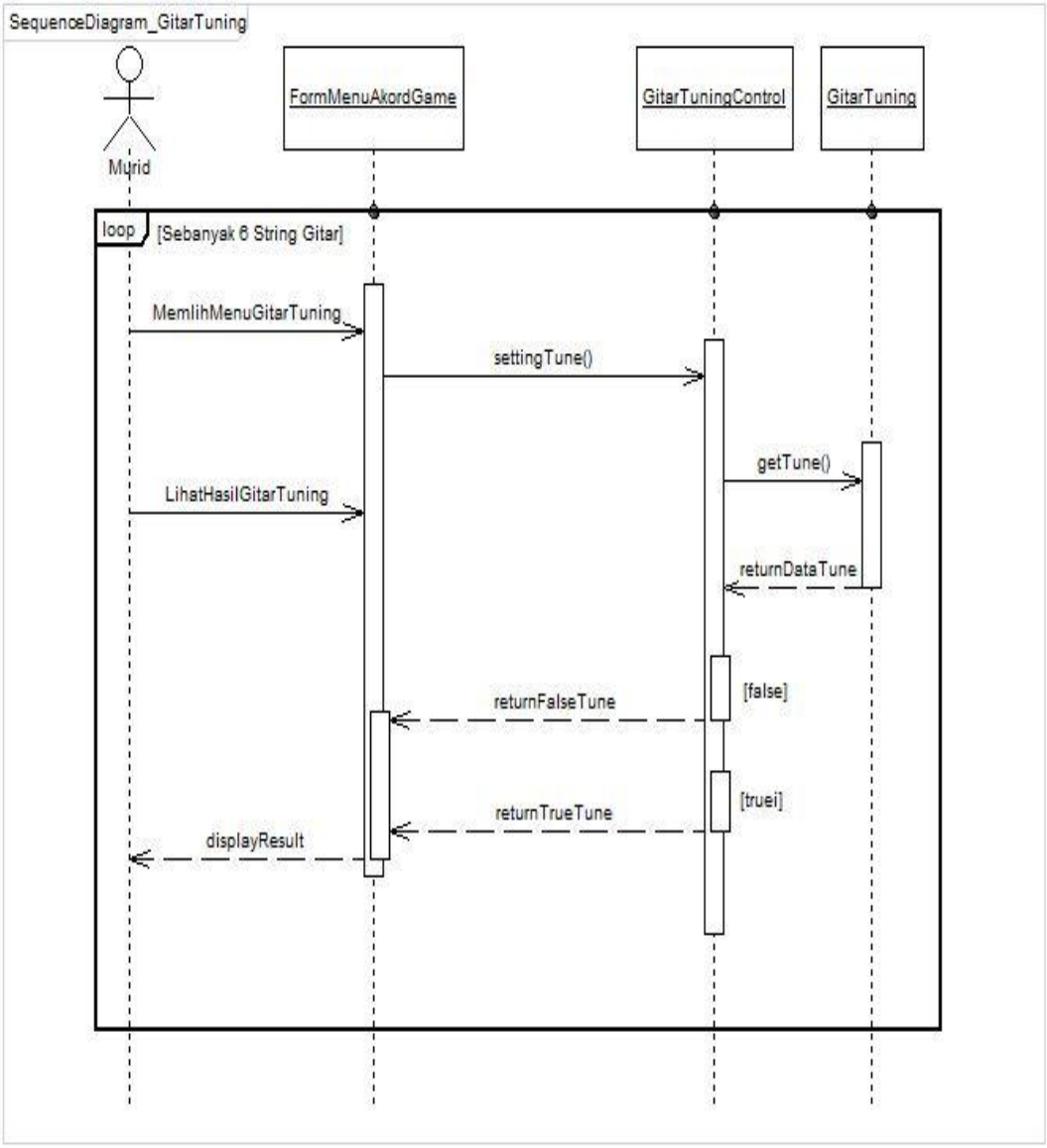
Interaksi antar kelas yang terjadi pada saat proses autentikasi Murid dapat dilihat pada gambar 3.48.



Gambar 3.48 *Sequence Diagram* Autentikasi Murid

3.4.4.18. *Sequence diagram* Gitar Tuning

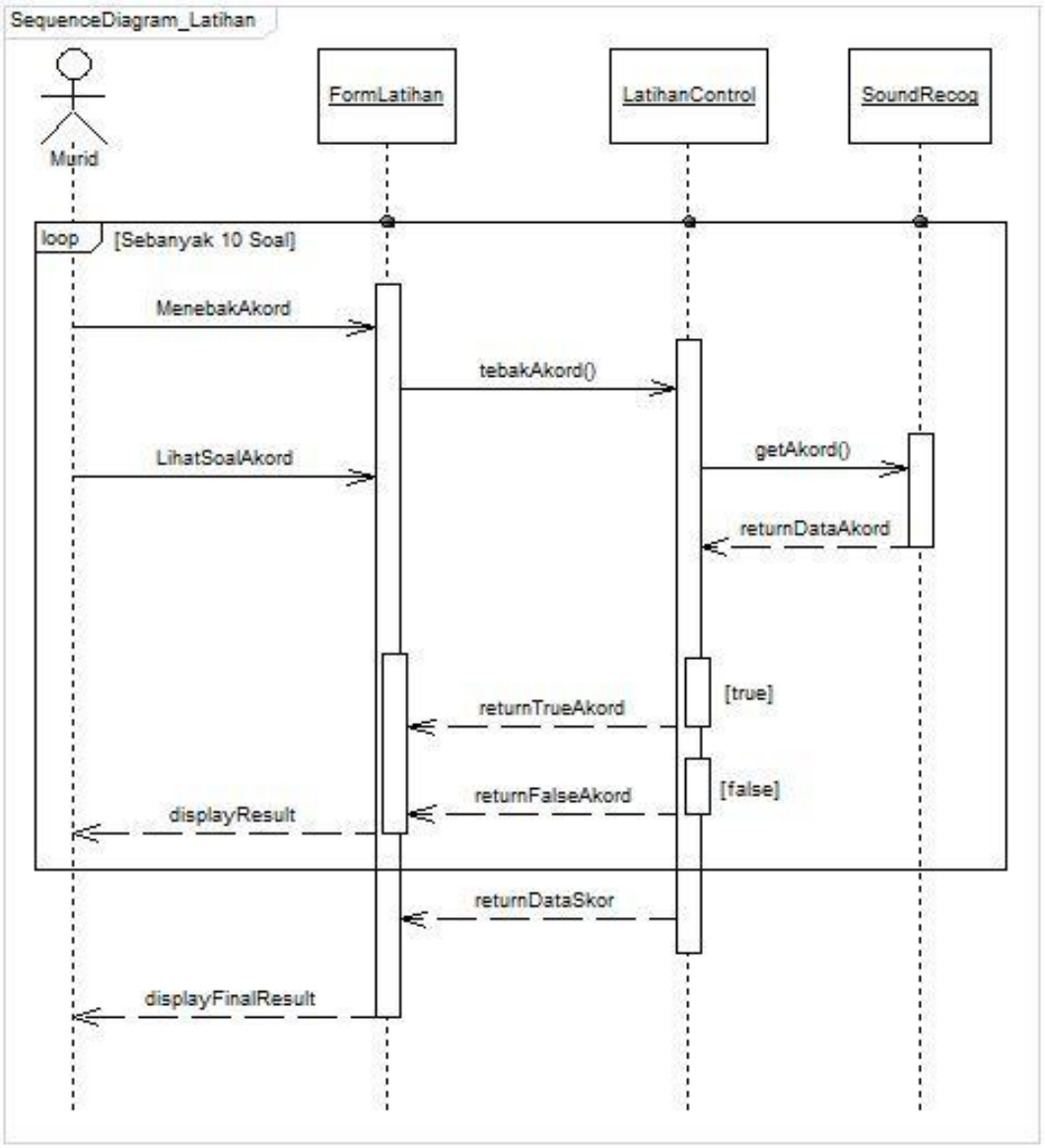
Interaksi antar kelas yang terjadi pada saat proses gitar tuning dapat dilihat pada gambar 3.49.



Gambar 3.49 *Sequence Diagram* Gitar Tuning

3.4.4.19. *Sequence Diagram* Latihan (Akord *Game*)

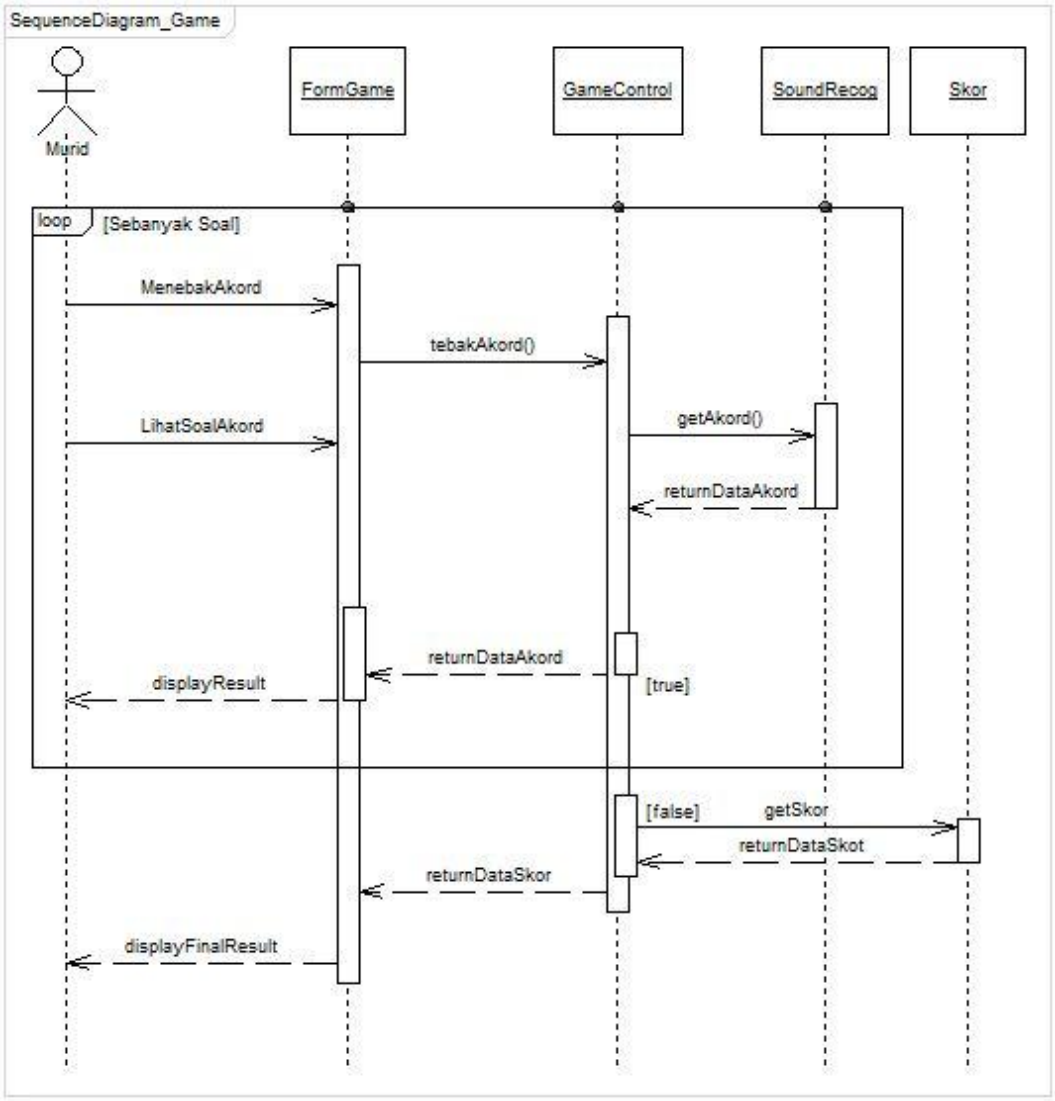
Interaksi antar kelas yang terjadi pada saat proses Latihan (Akord *Game*) dapat dilihat pada gambar 3.50.



Gambar 3.50 *Sequence Diagram* Latihan (Akord *Game*)

3.4.4.20. *Sequence Game* (Akord *Game*)

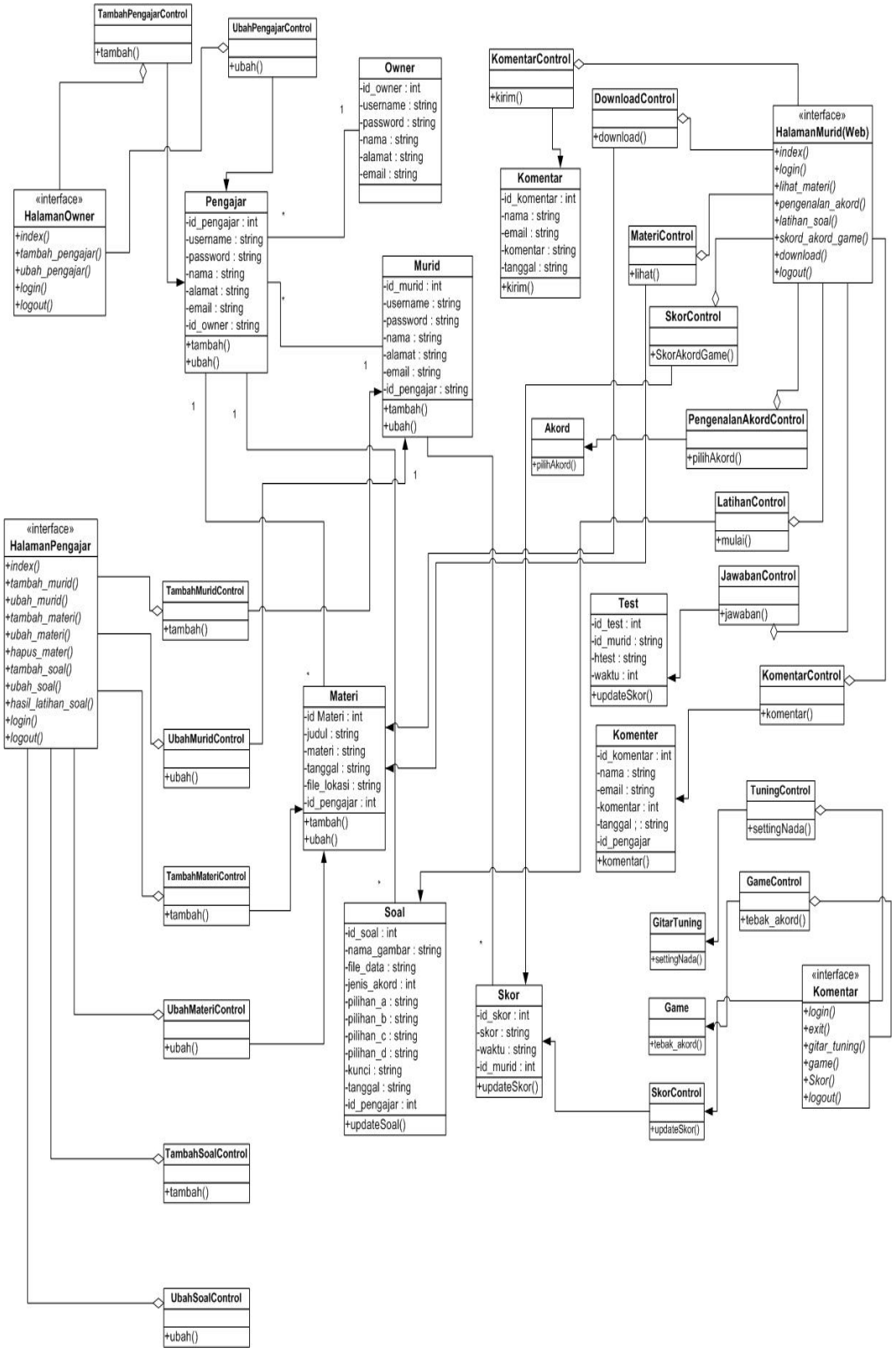
Interaksi antar kelas yang terjadi pada saat proses *Game* (Akord *Game*) dapat dilihat pada gambar 3.51.



Gambar 3.51 *Sequence Diagram Game* (Akord *Game*)

3.4.5. Class Diagram

Class Diagram ditujukan untuk mendefinisikan proses yang berjalan pada aplikasi pengenalan dan pembelajaran akord gitar. Terdapat tiga jenis kelas dalam sistem, yaitu kelas *interface*, kelas *controller* dan kelas data. Seluruh kelas-kelas yang ada dalam sistem dapat dilihat pada gambar 3.52.



Gambar 3.52 Class Diagram Pembelajaran Akord

3.5. Perancangan Sistem

Perancangan merupakan bagian dari metodologi pembangunan suatu perangkat lunak yang harus dilakukan setelah melalui tahapan analisis. Pada bagian ini akan dijelaskan perancangan sistem yang dimaksudkan untuk menggambarkan perbedaan antara sistem yang sedang berjalan dengan sistem yang diusulkan.

3.5.1. Perancangan Data

Pada tahap data merupakan transformasi model data yang dihasilkan oleh proses analisis menjadi objek data yang dibutuhkan pada saat implementasi.

3.5.1.1. Skema Relasi

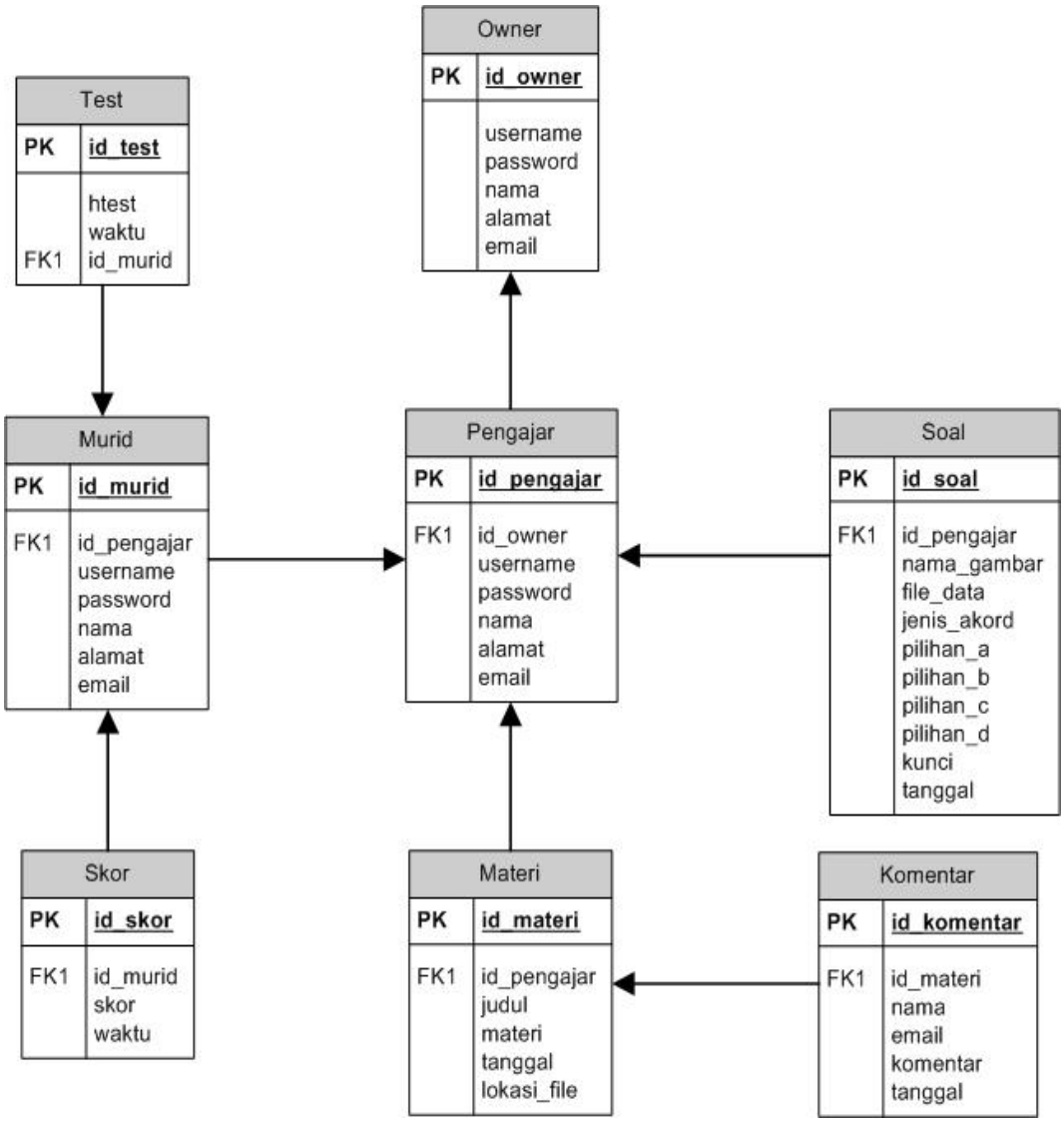
Skema relasi pada sistem pembelajaran akord memiliki empat skema relasi, yaitu:

1. *Owner* =(id_*owner*, *username*, *password*, nama, alamat, email)
2. Pengajar= (id_pengajar, *username*, *password*, nama, alamat, email, id_owner)
3. Murid= (id_murid, *username*, *password*, nama, alamat, email, id_pengajar)
4. Materi=(id_materi, judul, materi, tanggal,file_lokasi, id_pengajar)
5. Skor=(id_skor, waktu, id_murid)
6. Soal=(id_soal, nama_gambar, file_data, jenis_akord, pilihan_a, pilihan_b, pilihan_c, pilihan_d, kunci, tanggal, id_pengajar)
7. Test=(id_test, htest, waktu, id_murid)

8. Komentar=(id_komentar, nama, email, komentar, tanggal, id_materi)

3.5.1.2. Diagram Relasi

Kelas data pada diagram relasi yang mempunyai hubungan antar kelas yaitu kelas data pengajar, murid, dan materi. Diagram relasi dapat dilihat pada gambar 3.53.



Gambar 3.53 Diagram Relasi

3.5.1.3. Spesifikasi Kelas Data

Spesifikasi kelas data pada sistem terdapat empat kelas data, yaitu kelas *Owner*, Pengajar, Murid, Materi dan Skor. Kelas data merupakan kelas yang digunakan sebagai penampung data atau representasi data itu sendiri.

3.5.1.3.1. Deskripsi Atribut Kelas *Owner*

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas *Owner* dapat dilihat pada tabel 3.34.

Tabel 3.34 Kelas *Owner*

Nama	Tipe	Panjang	Kunci	Keterangan
id_owner	Int	8	*PrimaryKey*	Tidak boleh kosong, Unix
username	String	50		Tidak boleh kosong, Unix
password	String	50		Tidak boleh kosong
nama	String	50		Tidak boleh kosong
alamat	String	50		Tidak boleh kosong
email	String	50		Tidak boleh kosong, Tidak boleh sama

3.5.1.3.2. Deskripsi Atribut Kelas Pengajar

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas Pengajar dapat dilihat pada tabel 3.35.

Tabel 3.35 Kelas Pengajar

Nama	Tipe	Panjang	Kunci	Keterangan
id_pengajar	Int	8	*PrimaryKey*	Tidak boleh kosong, Tidak boleh sama
username	String	50		Tidak boleh kosong, Tidak boleh sama
password	String	50		Tidak boleh kosong
nama	String	50		Tidak boleh kosong
alamat	String	50		Tidak boleh kosong
email	String	50		Tidak boleh kosong, Tidak boleh sama
id_owner	int	8	*ForeignKey*	Tidak boleh kosong, Tidak boleh sama

3.5.1.3.3. Deskripsi Atribut Kelas Murid

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas Murid dapat dilihat pada tabel 3.36.

Tabel 3.36 Kelas Murid

Nama	Tipe	Panjang	Kunic	Keterangan
id_murid	Int	8	*PrimaryKey*	Tidak boleh kosong, Tidak boleh sama
username	String	50		Tidak boleh kosong, Tidak boleh sama

password	String	50		Tidak boleh kosong
nama	String	50		Tidak boleh kosong
alamat	String	50		Tidak boleh kosong
email	String	50		Tidak boleh kosong, Tidak boleh sama
id_pengajar	Int	8	*ForeignKey*	Tidak boleh kosong, Tidak boleh sama

3.5.1.3.4. Deskripsi Atribut Kelas Materi

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas Materi dapat dilihat pada tabel 3.37.

Tabel 3.37 Kelas Materi

Nama	Tipe	Panjang	Kunci	Keterangan
id_materi	Int	8	*PrimaryKey*	Tidak boleh kosong, Tidak boleh sama
judul	String	50		Tidak boleh kosong
materi	String	50		Tidak boleh kosong
tanggal	Date			
file_lokasi	String	200		
id_pengajar	Int	8	*ForeignKey*	Tidak boleh kosong, Tidak boleh sama

3.5.1.3.5. Deskripsi Atribut Kelas Skor

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas Skor dapat dilihat pada tabel 3.38.

Tabel 3.38 Kelas Skor

Nama	Tipe	Panjang	Kunci	Keterangan
id_materi	Int	8	*PrimaryKey*	Tidak boleh kosong, TIdak boleh sama
waktu	Date			
Id_murid	Int	8	*ForeignKey*	Tidak boleh kosong, Tidak boleh sama

3.5.1.3.6. Deskripsi Atribut Kelas Soal

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas Soal dapat dilihat pada tabel 3.39.

Tabel 3.39 Kelas Soal

Nama	Tipe	Panjang	Kunci	Keterangan
id_soal	Int	5	*PrimaryKey*	Tidak boleh kosong, Tidak boleh sama
nama_gambar	String	50		Tidak boleh kosong, Tidak boleh sama
file_data	Longblob			Tidak boleh kosong , Tidak Boleh dari 1 mb

jenis_akord	String	50		Tidak boleh kosong
pilihan_a	String	10		Tidak boleh kosong
pilihan_b	String	10		Tidak boleh kosong
pilihan_c	String	10		Tidak boleh kosong
pilihan_d	String	10		Tidak boleh kosong
kunci	enum	(‘A’,’B’,’C’,’D’)		Tidak boleh kosong
tanggal	Date			
id_pengajar	Int	8	*ForeignKey*	Tidak boleh kosong, Tidak boleh sama

3.5.1.3.7. Deskripsi Atribut Kelas Test

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas test dapat dilihat pada tabel 3.40.

Tabel 3.40 Kelas Test

Nama	Tipe	Panjang	Kunci	Keterangan
id_test	Int	5	*PrimaryKey*	Tidak boleh kosong, Tidak boleh sama

htest	String	10		Tidak boleh kosong
waktu	Date			
Id_murid	Int	8	*ForeignKey*	Tidak boleh kosong, Tidak boleh sama

3.5.1.3.8. Deskripsi Atribut Kelas Komentar

Deskripsi atribut kelas menjelaskan isi data yang berada dalam satu record pada satu tabel. Deskripsi atribut kelas komentar dapat dilihat pada tabel 3.41.

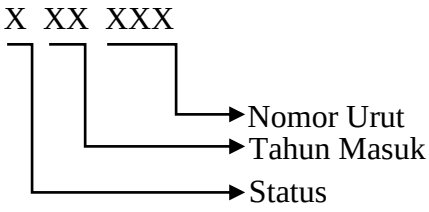
Tabel 3.41 Kelas Komentar

Nama	Tipe	Panjang	Kunci	Keterangan
id_komentar	Int	10	*PrimaryKey*	Tidak boleh kosong, Tidak boleh sama
nama	String	50		Tidak boleh kosong
email	String	50		Tidak boleh kosong
komentar	Longtext			Tidak boleh kosong
tanggal	Date			
id_materi	Int	8	*ForeignKey*	Tidak boleh kosong, Tidak boleh sama

3.5.2. Perancangan Pengkodean

Perancangan pengkodean digunakan untuk menjelaskan kode-kode dalam primary key yang ada dalam sistem sebagai nomor id untuk pengguna sistem. Pengkodean yang digunakan dalam sistem ini adalah:

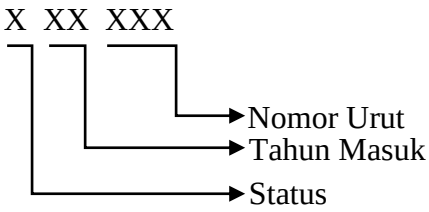
1. Pengkodean Pengajar



Contoh : 2 10 001

Keterangan : Pengajar masuk pada tahun 2010 dengan status 2 adalah pengajar dan nomor urut pengajar 01.

2. Pengkodean Murid



Contoh : 3 10 001

Keterangan : Murid masuk pada tahun 2010 dengan status 3 adalah murid dan nomor urut murid 001.

3.5.3. Perancangan Arsitektur

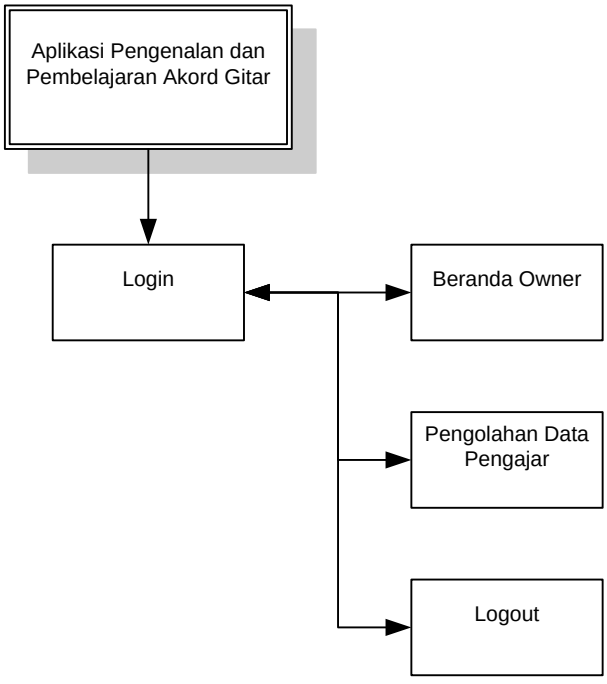
Pada tahap ini merupakan struktur program yang berfungsi untuk mempresentasikan dan mendefinisikan antarmuka yang memungkinkan data dapat mengalir pada seluruh program.

3.5.3.1. Perancangan Struktur Menu

Perancangan struktur menu berisi menu dan submenu yang berfungsi memudahkan pengguna dalam menggunakan sistem. Menu tersebut dibagi berdasarkan hak akses masing-masing pengguna. Berikut ini gambaran mengenai struktur menu aplikasi pengenalan dan pembelajaran akord gitar berbasis *web*:

1. Struktur Menu *Owner*

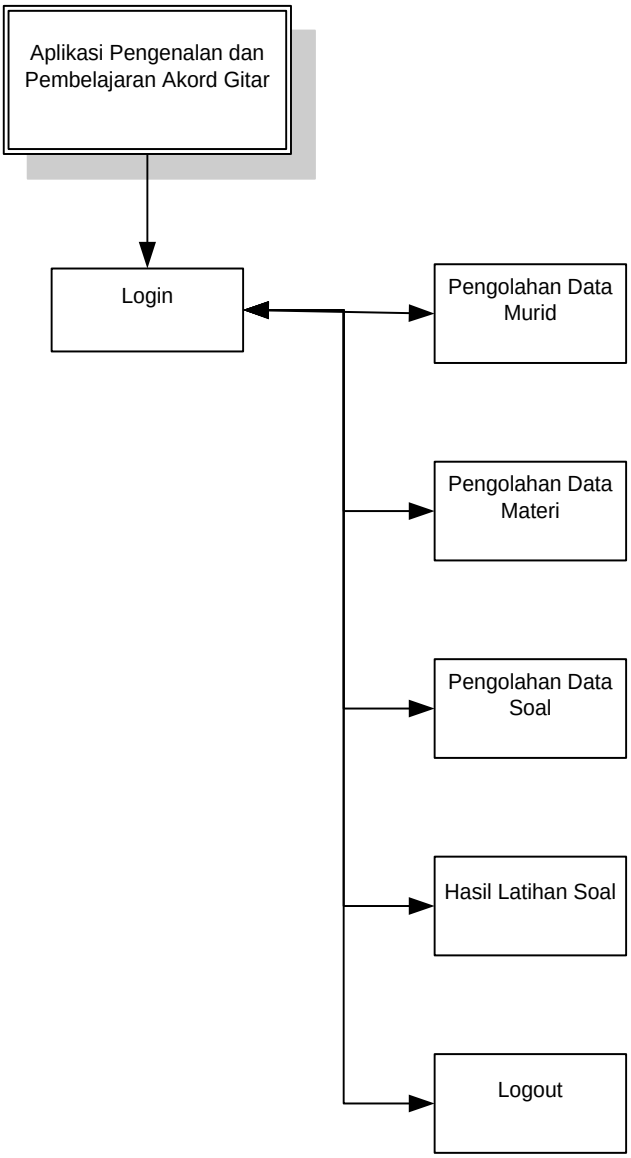
   Struktur menu *Owner* terdiri dari *login,* pengolahan data pengajar, tambah data pengajar, ubah data pengajar dan *logout*. Struktur menu *Owner* dapat dilihat pada gambar 3.54.



Gambar 3.54 Struktur Menu *Owner*

2. Struktur Menu Pengajar

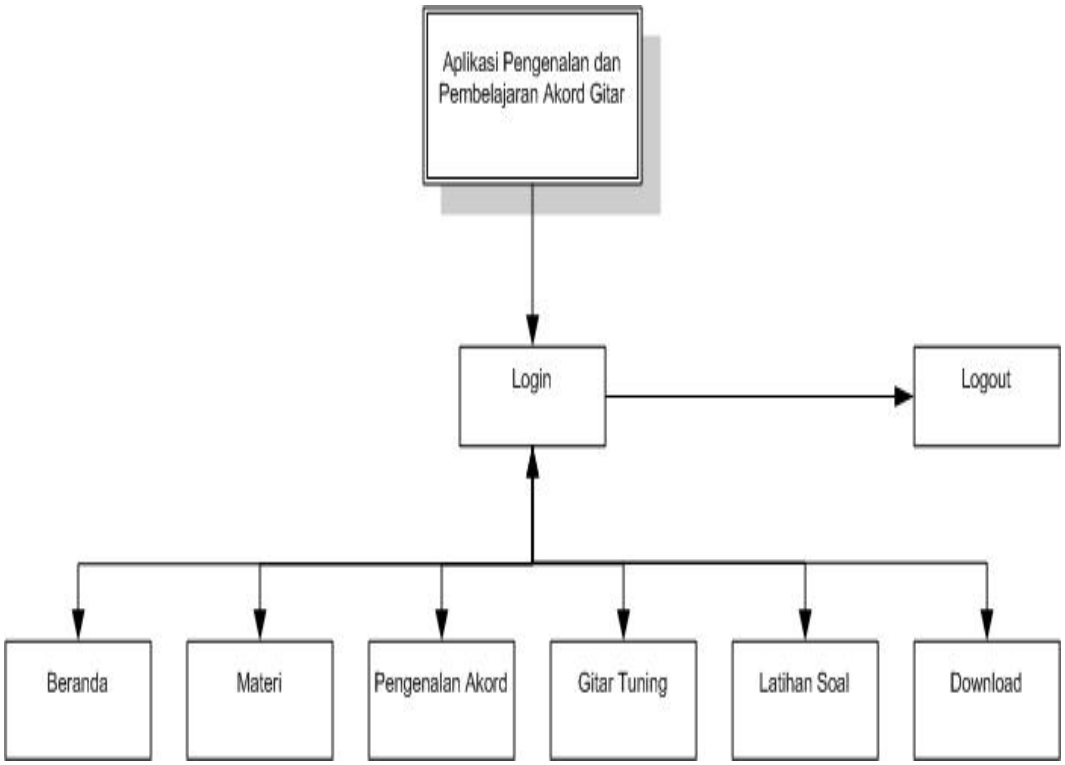
Struktur menu Pengajar terdiri dari *login*, pengolahan data murid, pengolahan data materi, pengolahan data soal dan *logout*. Struktur menu Pengajar dapat dilihat pada gambar 3.55.



Gambar 3.55 Struktur Menu Pengajar

3. Struktur Menu Murid

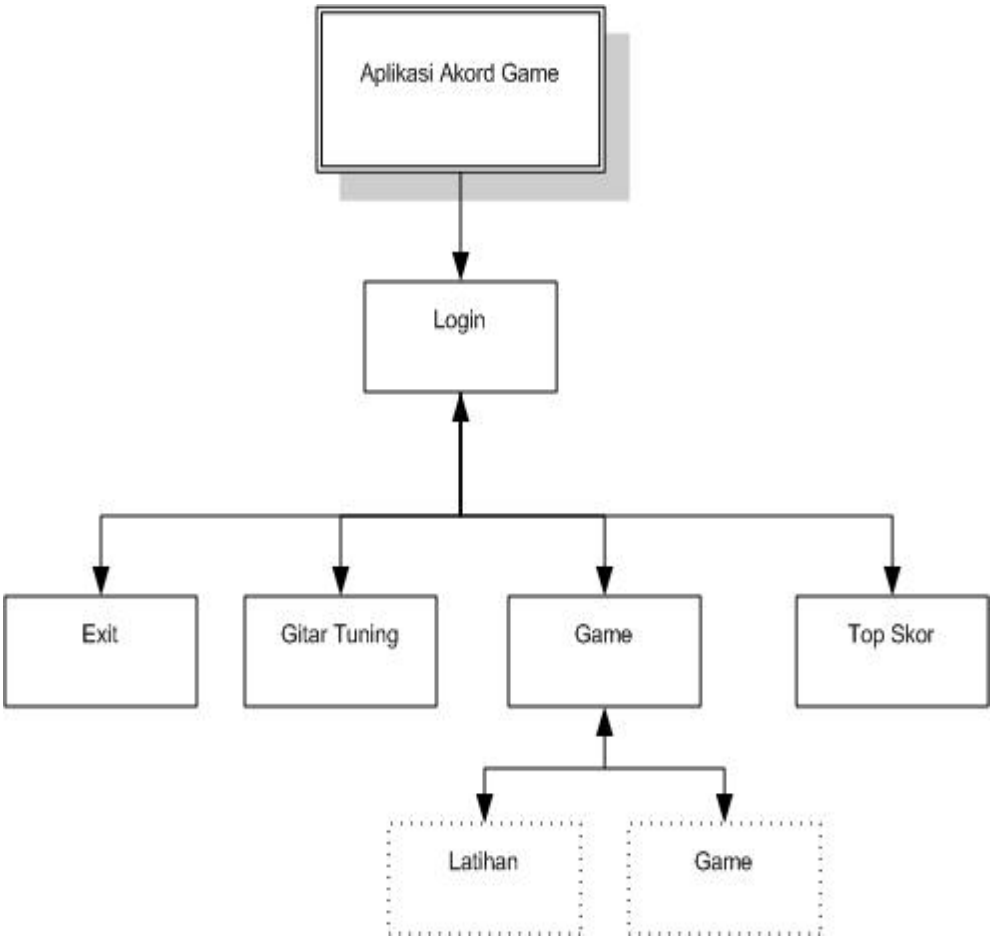
   Struktur menu Murid terdiri dari *login*, beranda, materi, pengenalan akord, gitar tuning, Evaluasi (Akord *Game*), download dan *logout*. Struktur menu Murid dapat dilihat pada gambar 3.56.



Gambar 3.56 Struktur Menu Murid

4. Struktur Menu Aplikasi Akord Game

   Struktur menu Aplikasi Akord Game terdiri dari *login*, exit, gitar tuning, *game*, dan top skor. Struktur menu Aplikasi Akord Game dapat dilihat pada gambar 3.57.



Gambar 3.57 Struktur Menu Aplikasi Akord Game

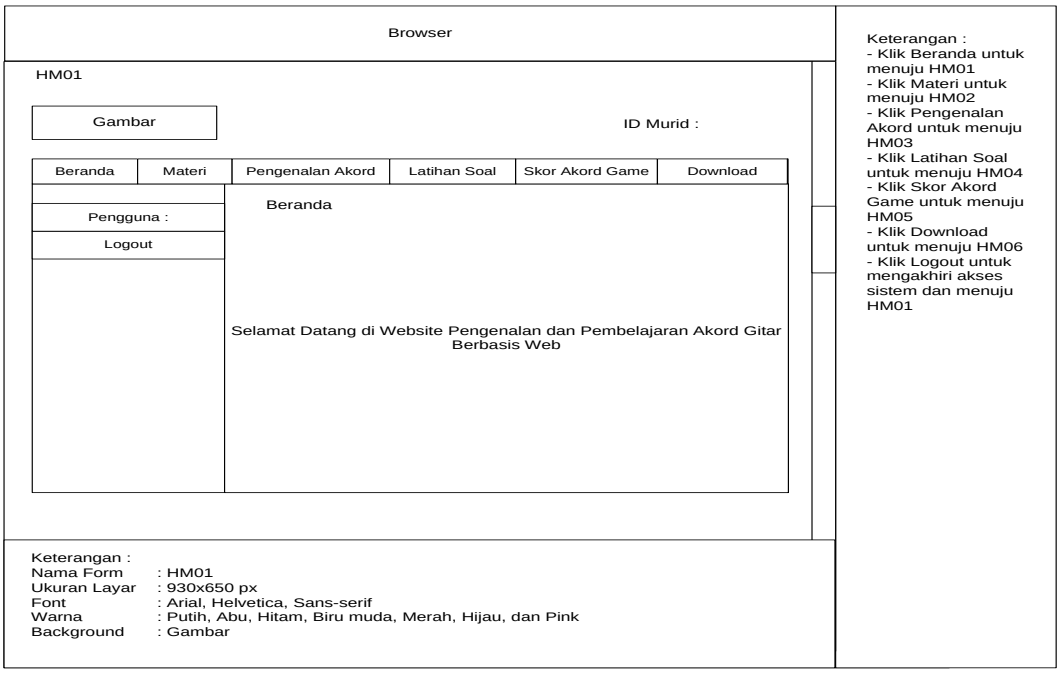
3.5.4. Perancangan Antarmuka

Perancangan antarmuka bertujuan untuk memberikan gambaran tentang sistem yang akan dibangun. Perancangan ini diimplementasikan menjadi sebuah program yang utuh dan dapat digunakan oleh pengguna sistem.

3.5.4.1. Perancangan Form

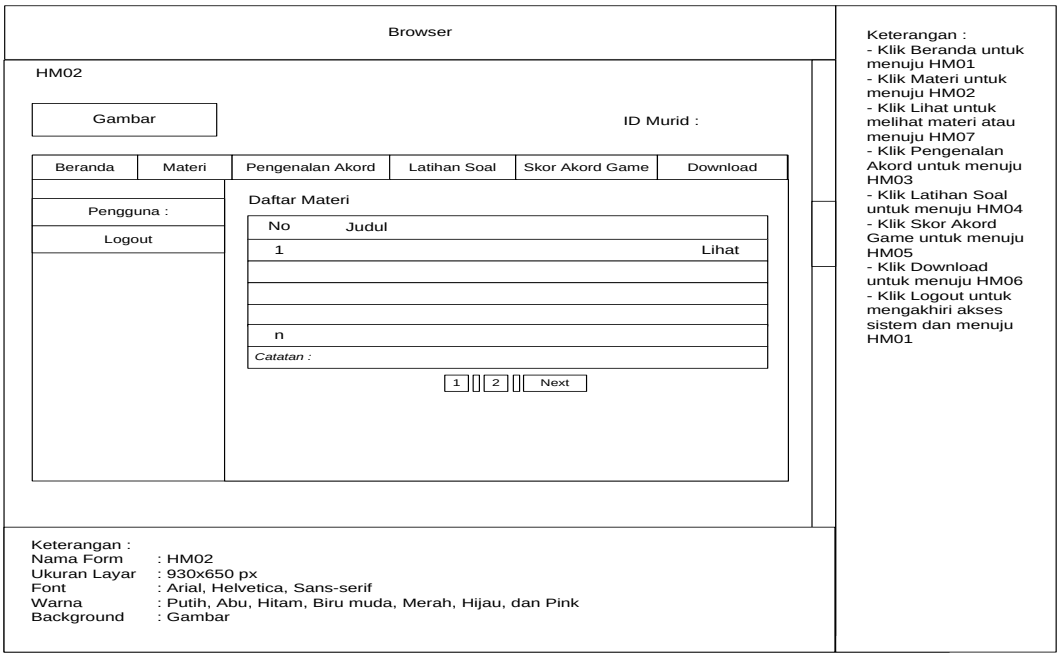
Perancangan form bertujuan untuk memberikan gambaran pada setiap halaman sistem. Perancangan yang akan diimplementasikan, yaitu:

1. Perancangan Antarmuka Beranda dapat dilihat pada gambar 3.58.



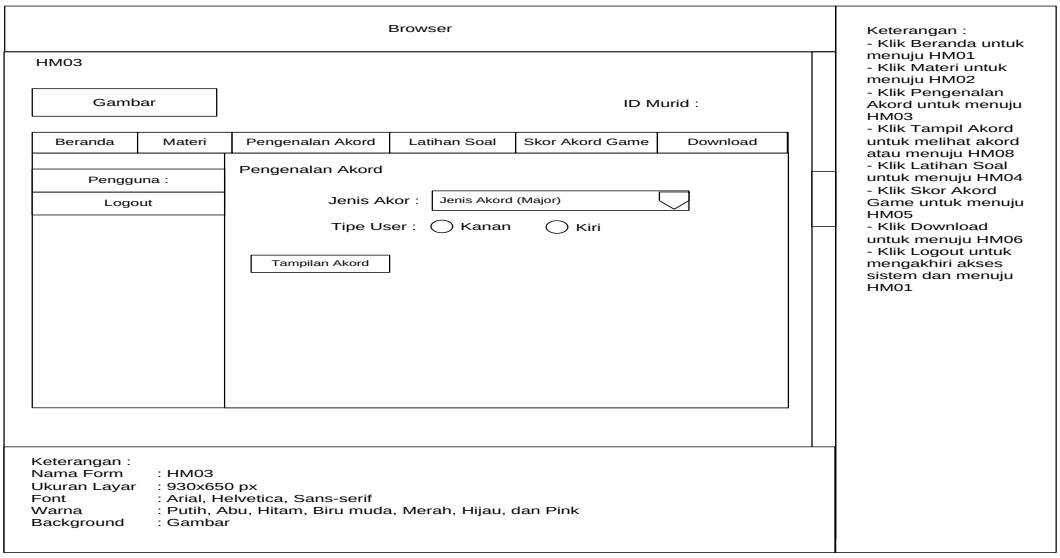
Gambar 3.58 Perancangan Antarmuka Beranda

2. Perancangan Antarmuka Materi dapat dilihat pada gambar 3.59.



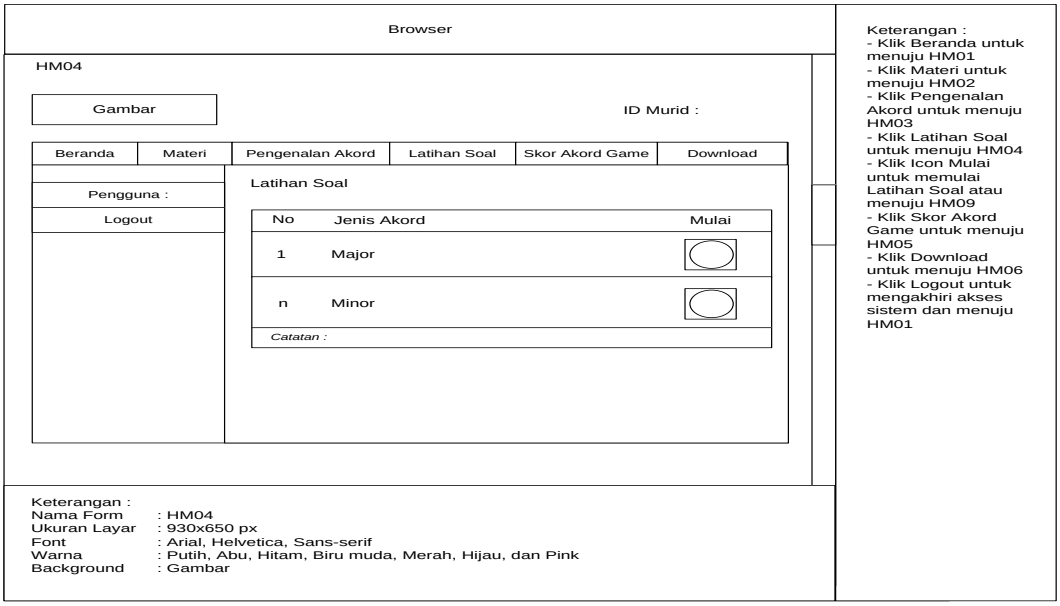
Gambar 3.59 Perancangan Antarmuka Materi

3. Perancangan Antarmuka Pengenalan Akord dapat dilihat pada gambar 3.60.



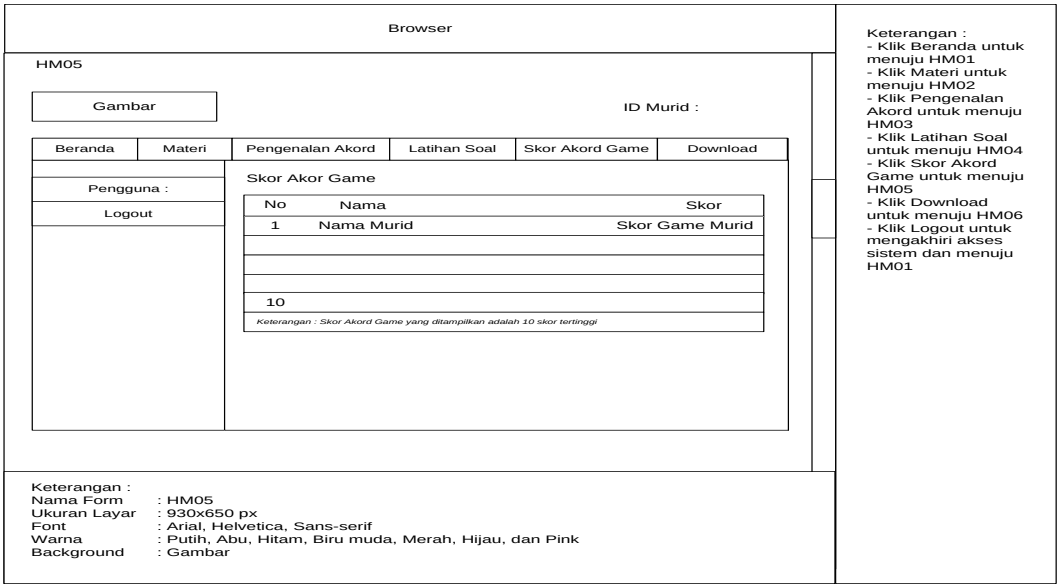
Gambar 3.60 Perancangan Antarmuka Pengenalan Akord

4. Perancangan Antarmuka Latihan Soal dapat dilihat pada gambar 3.61.



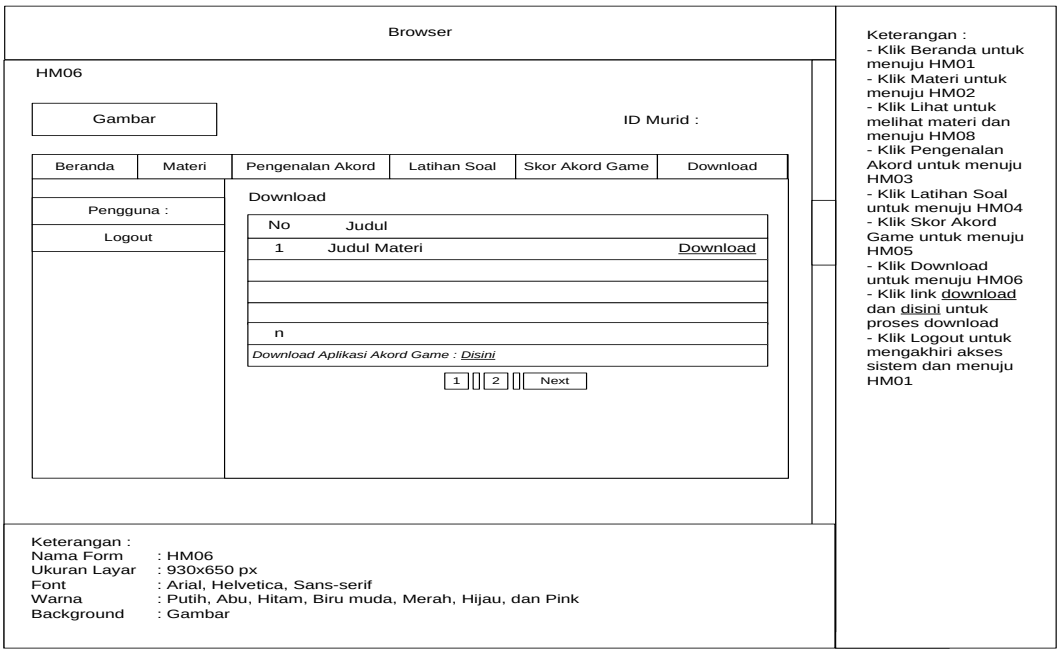
Gambar 3.61 Perancangan Antarmuka Latihan Soal

5. Perancangan Antarmuka Skor Akord *Game* dapat dilihat pada gambar 3.62.



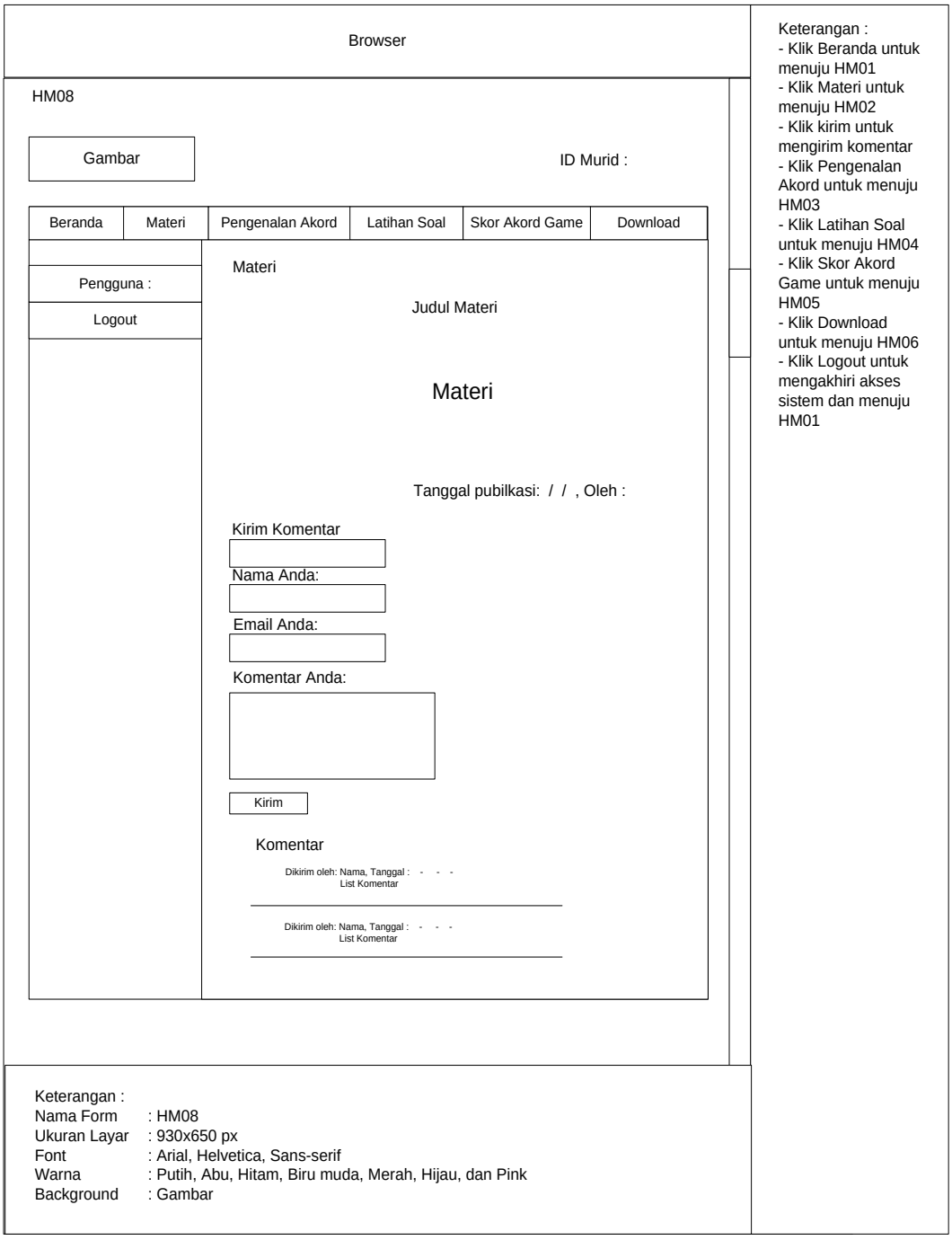
Gambar 3.62 Perancangan Antarmuka Skor Akord *Game*

6. Perancangan Antarmuka *Download* dapat dilihat pada gambar 3.63.



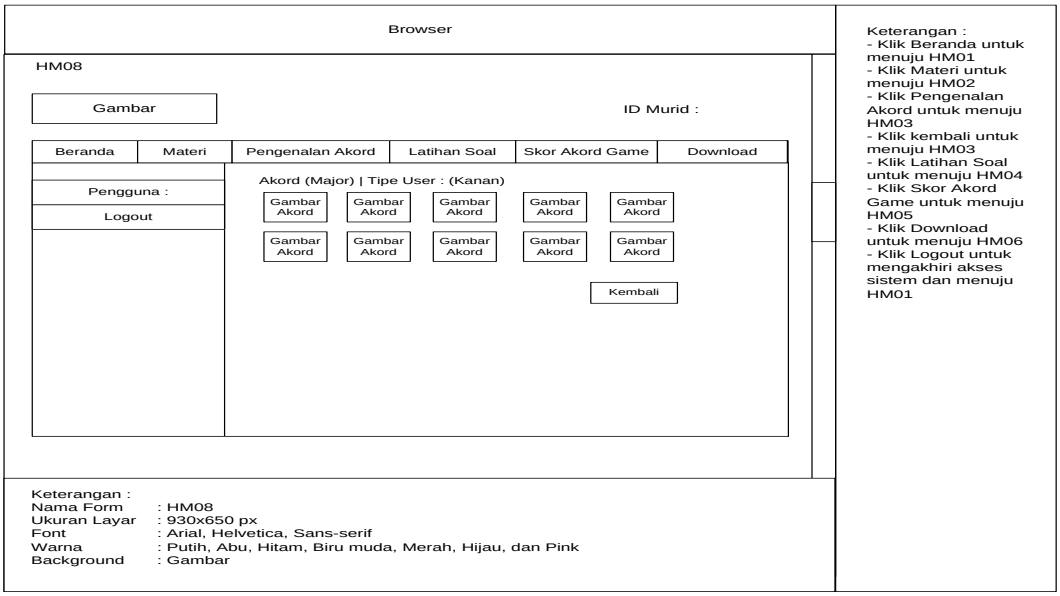
Gambar 3.63 Perancangan Antarmuka *Download*

7. Perancangan Antarmuka Lihat Materi dapat dilihat pada gambar 3.64.



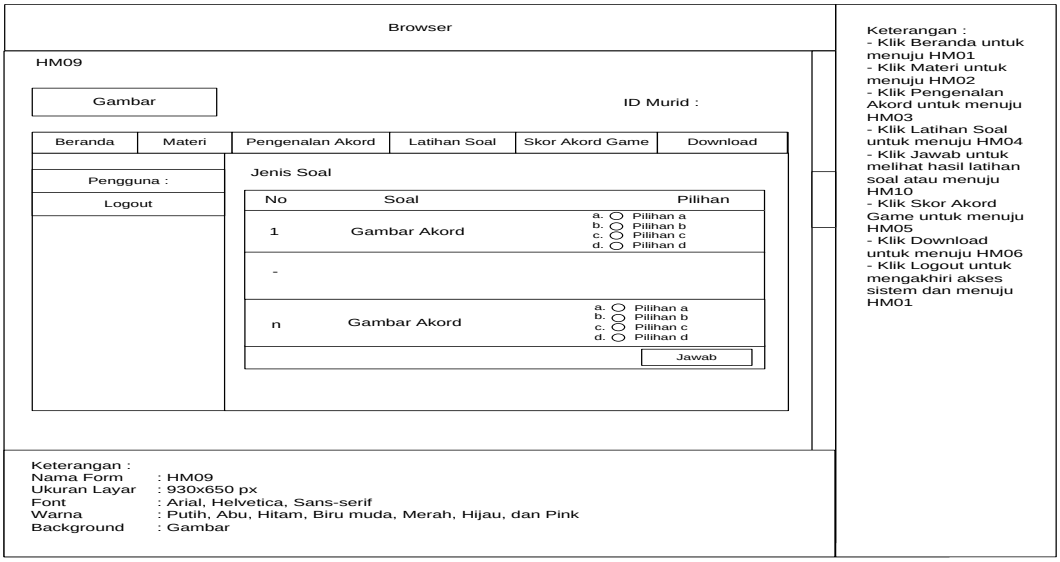
Gambar 3.64 Perancangan Antarmuka Lihat Materi

8. Perancangan Antarmuka Tampil Akord dapat dilihat pada gambar 3.65.



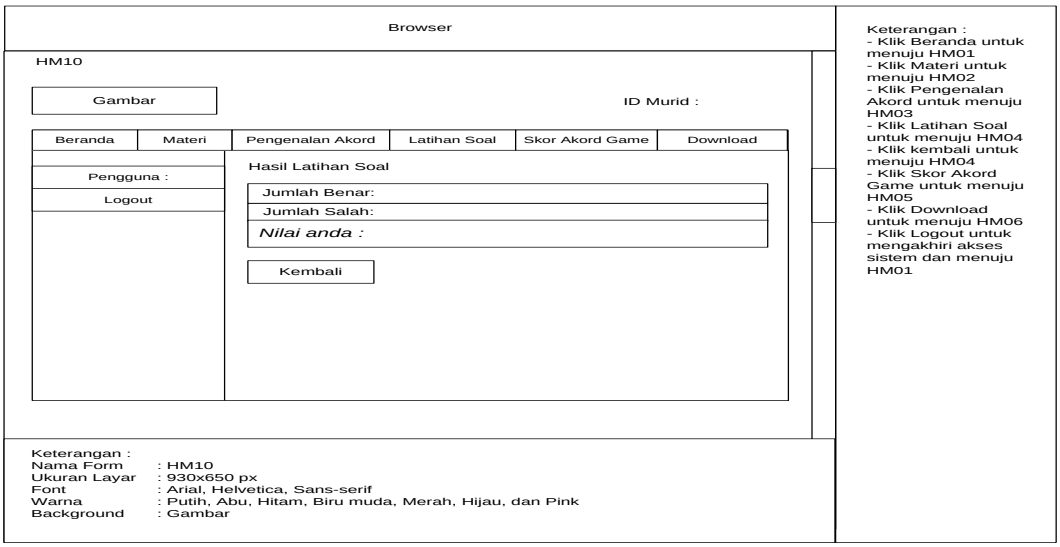
Gambar 3.65 Perancangan Antarmuka Tampil Akord

9. Perancangan Antarmuka Proses Latihan Soal dapat dilihat pada gambar 3.66.



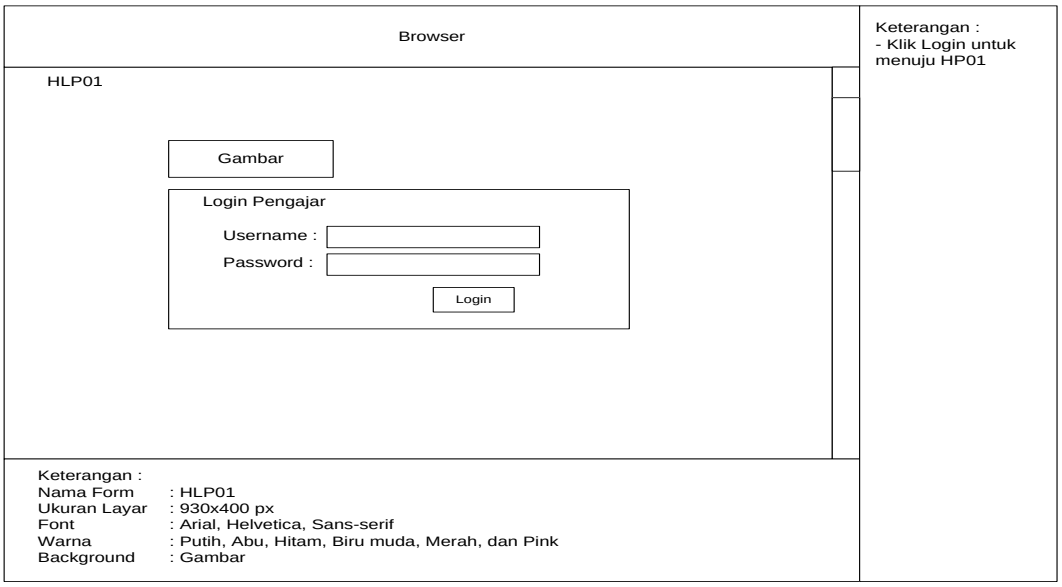
Gambar 3.66 Perancangan Antarmuka Proses Latihan Soal

10. Perancangan Antarmuka Hasil Latihan dapat dilihat pada gambar 3.67.



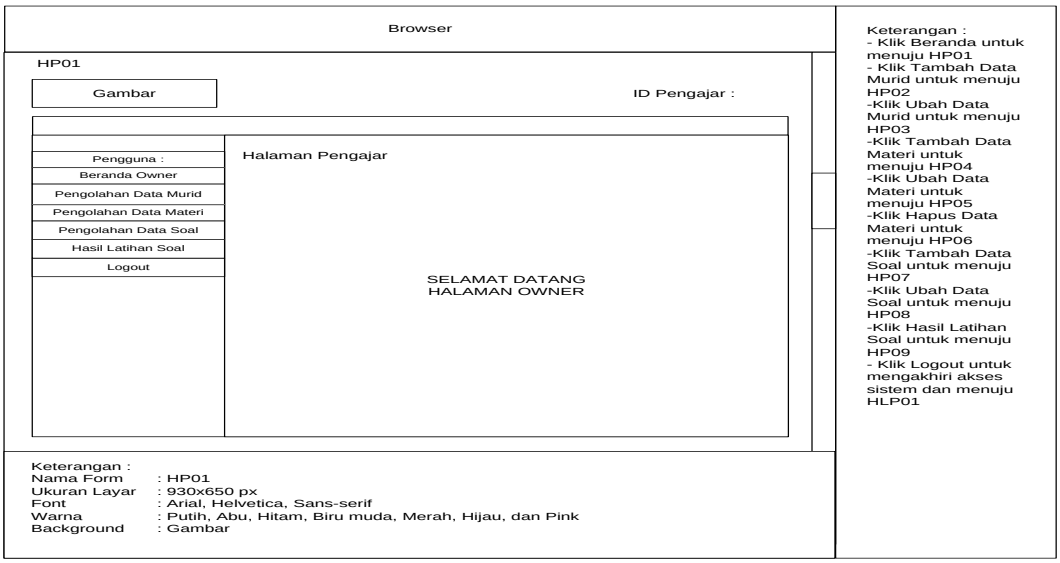
Gambar 3.67 Perancangan Antarmuka Tambah Data Pengajar

11. Perancangan Antarmuka *Login* Pengajar dapat dilihat pada gambar 3.68.



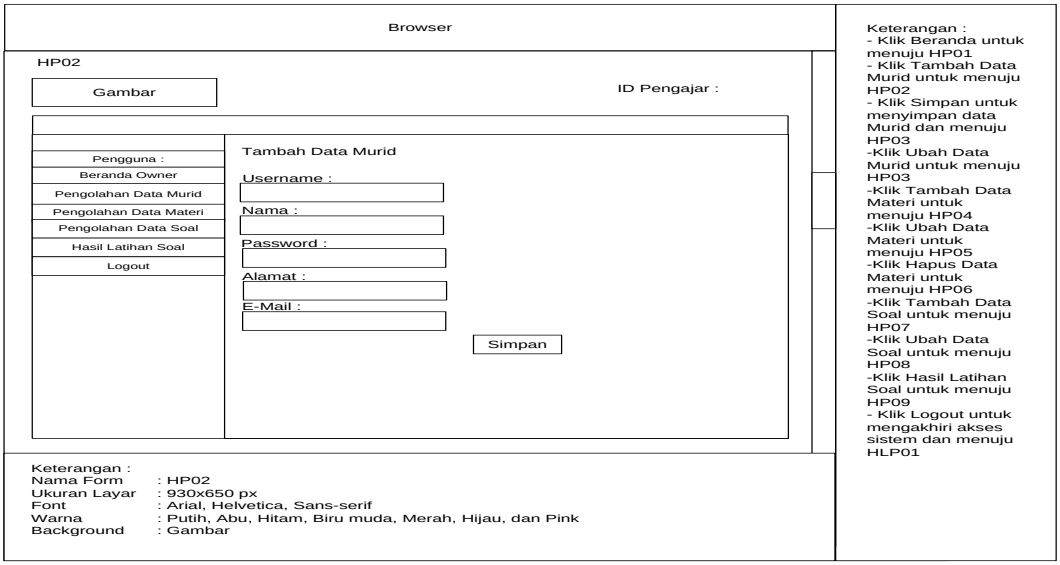
Gambar 3.68 Halaman *Login* Pengajar

12. Perancangan Antarmuka Beranda Pengajar dapat dilihat pada gambar 3.69.



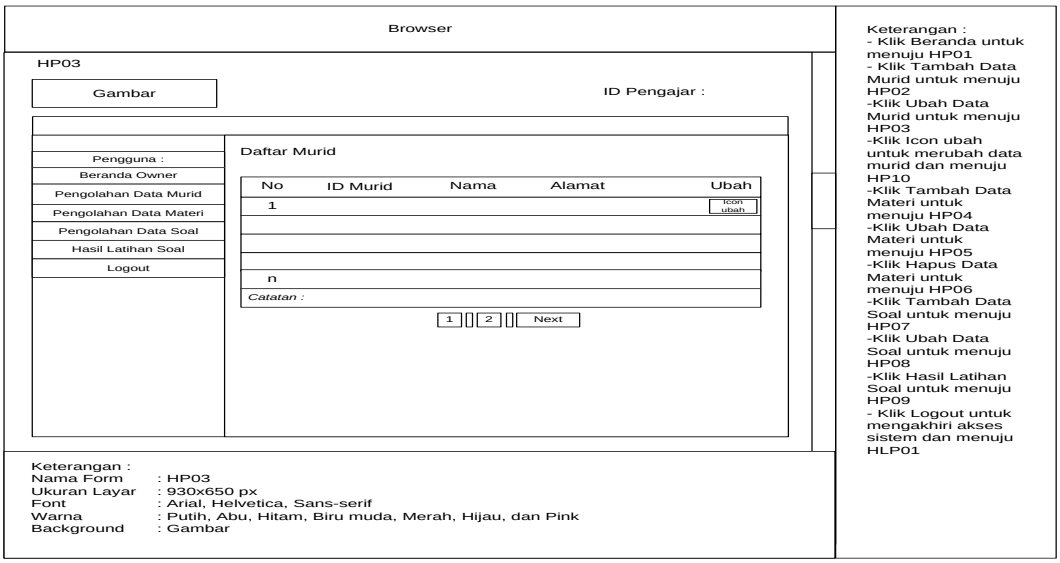
Gambar 3.69 Perancangan Antarmuka Beranda Pengajar

13. Perancangan Antarmuka Tambah Data Murid dapat dilihat pada gambar 3.70.



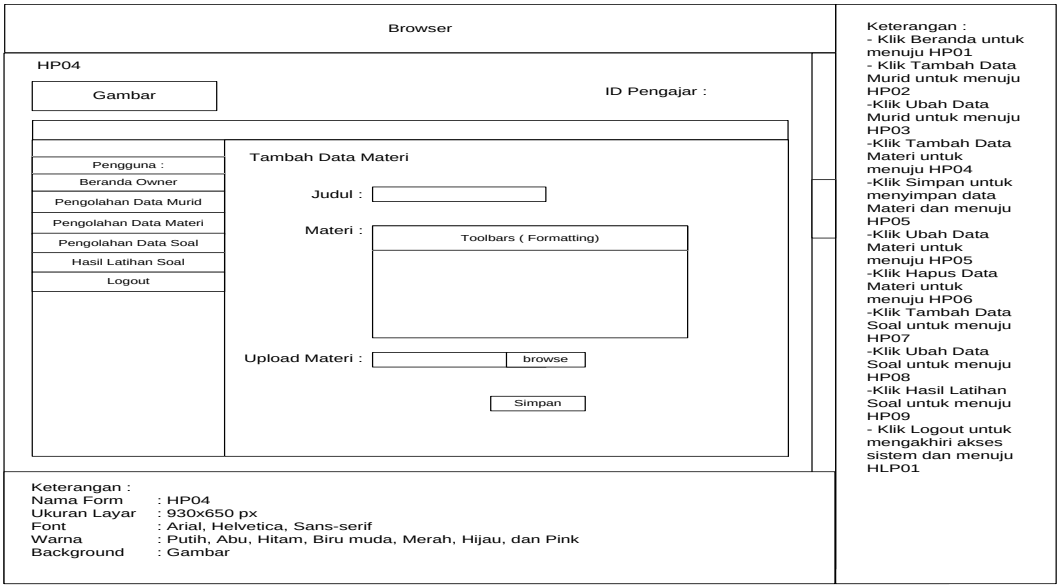
Gambar 3.70 Perancangan Antarmuka Tambah Data Murid

14. Perancangan Antarmuka Ubah Data Murid dapat dilihat pada gambar 3.71.



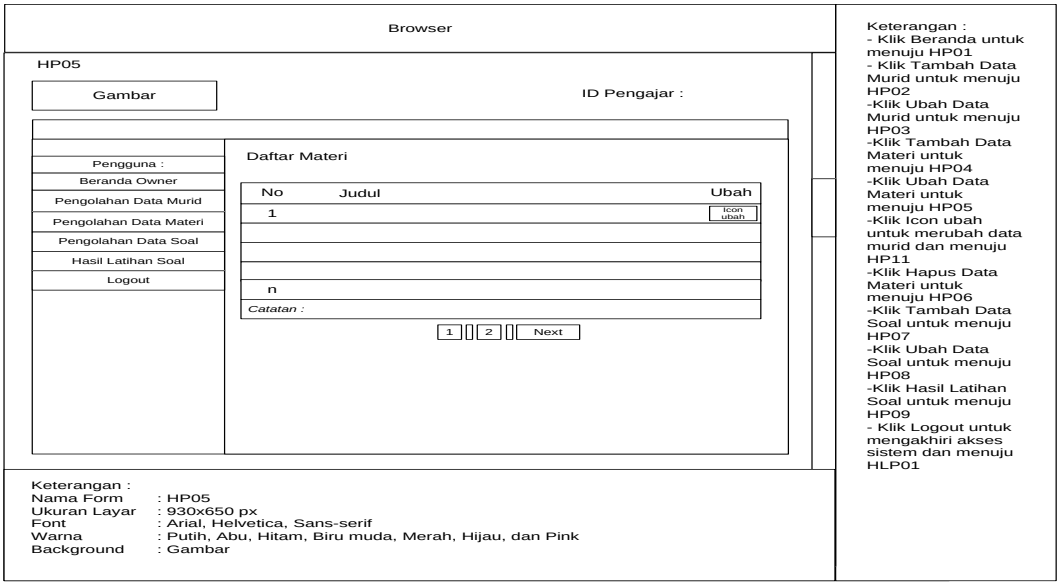
Gambar 3.71 Perancangan Antarmuka Ubah Data Murid

15. Perancangan Antarmuka Tambah Data Materi dapat dilihat pada gambar 3.72.



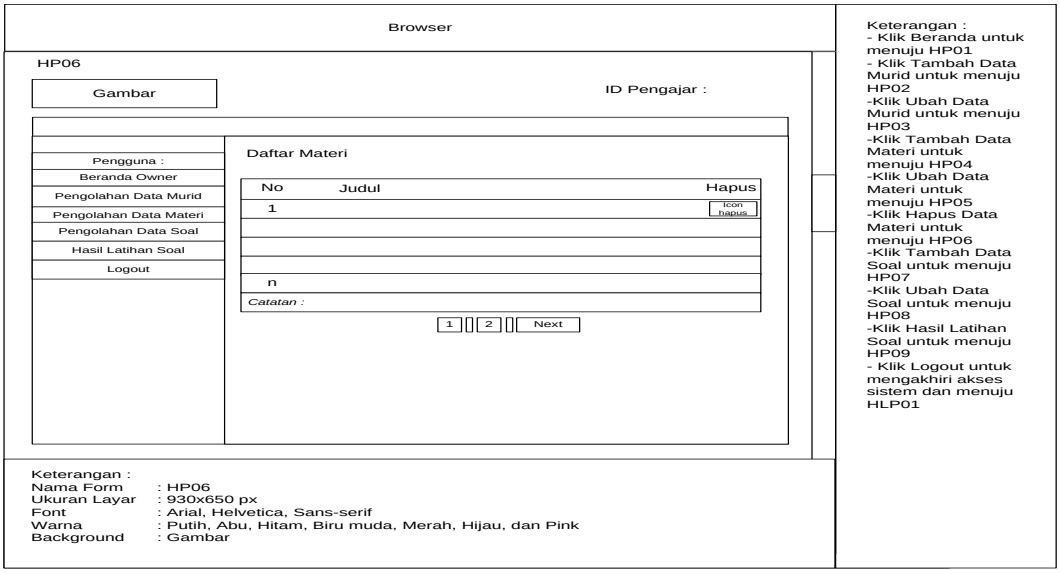
Gambar 3.72 Perancangan Antarmuka Tambah Data Materi

16. Perancangan Antarmuka Ubah Data Materi dapat dilihat pada gambar 3.73.



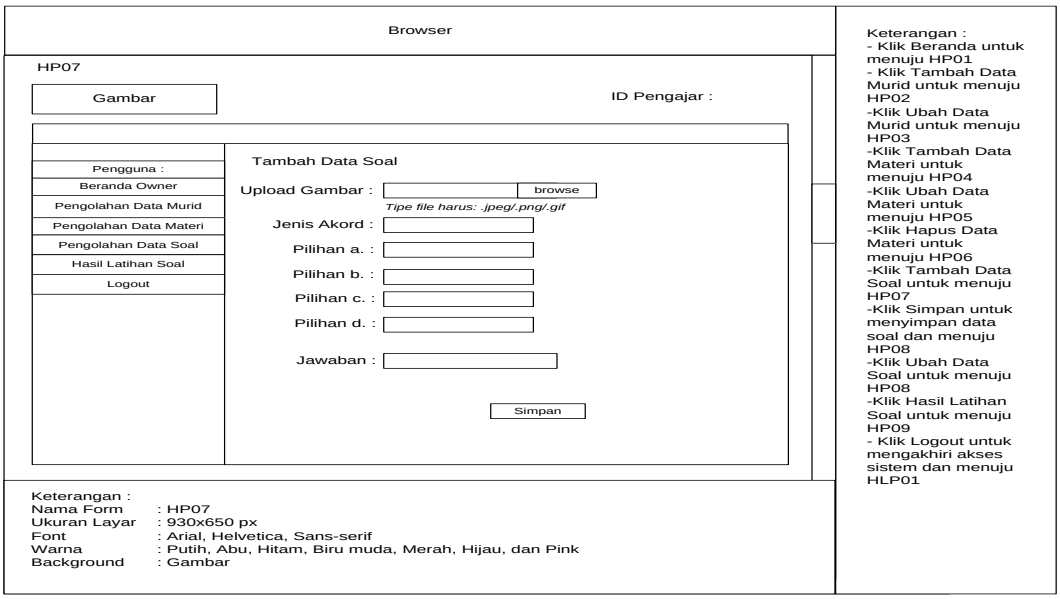
Gambar 3.73 Perancangan Antarmuka Ubah Data Materi

17. Perancangan Antarmuka Hapus Data Materi dapat dilihat pada gambar 3.74.



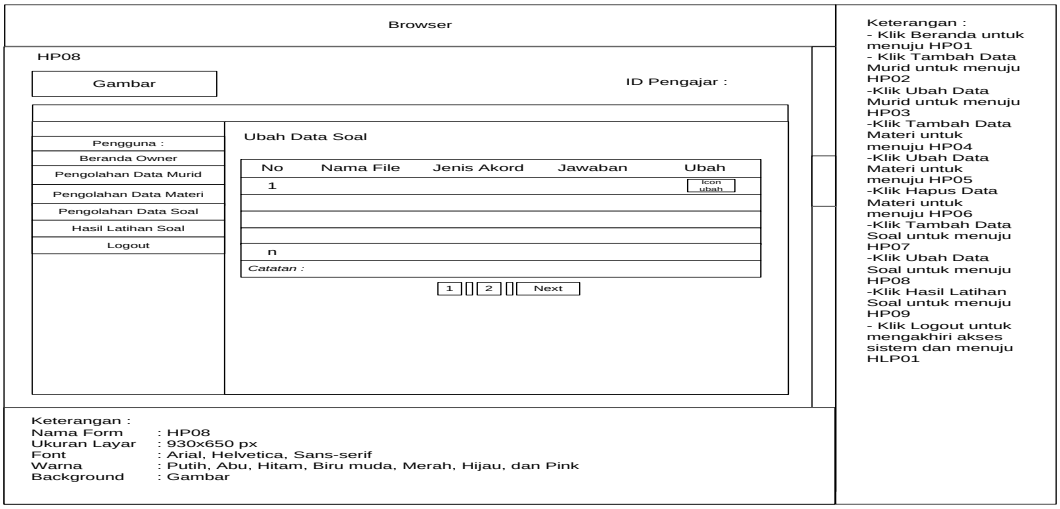
Gambar 3.74 Perancangan Antarmuka Hapus Data Materi

18. Perancangan Antarmuka Tambah Data Soal dapat dilihat pada gambar 3.75.



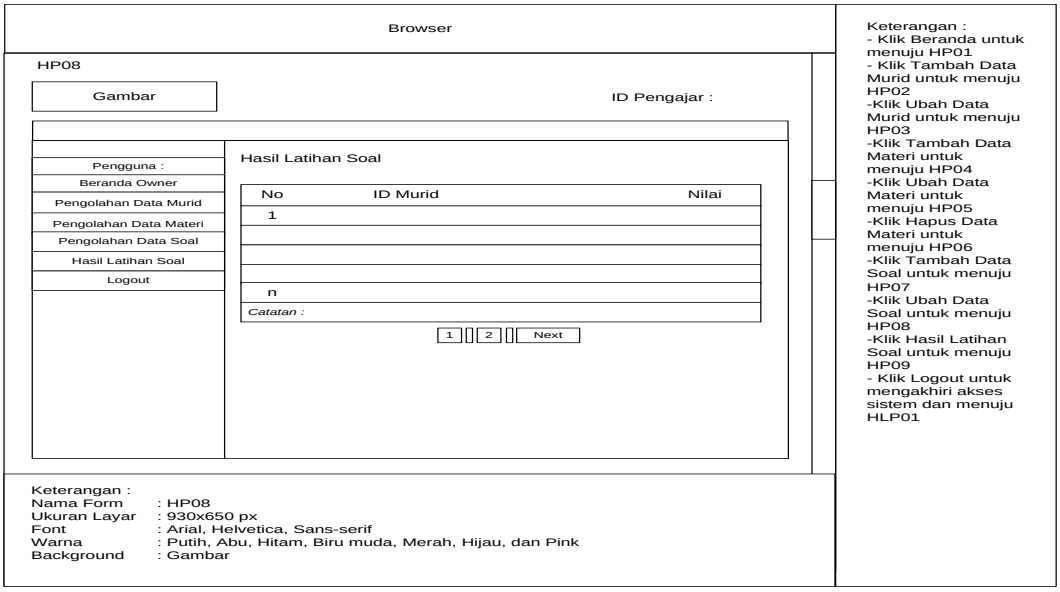
Gambar 3.75 Perancangan Antarmuka Tambah Data Soal

19. Perancangan Antarmuka Ubah Data Soal dapat dilihat pada gambar 3.76.



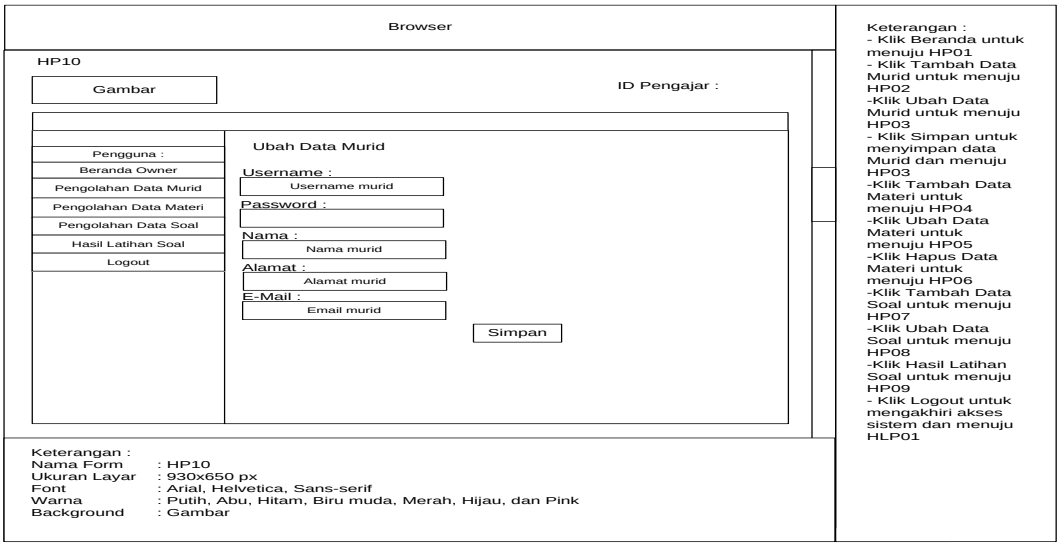
Gambar 3.76 Perancangan Antarmuka Ubah Data Soal

20. Perancangan Antarmuka Hasil Latihan Soal dapat dilihat pada gambar 3.77.



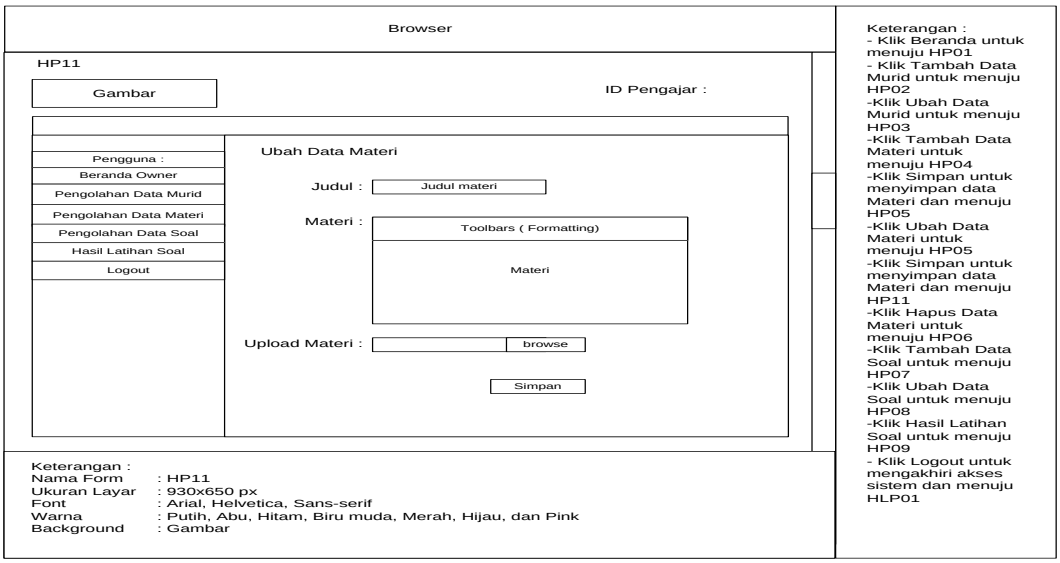
Gambar 3.77 Perancangan Antarmuka Hasil Latihan Soal

21. Perancangan Antarmuka Proses Ubah Data Murid dapat dilihat pada gambar 3.78.



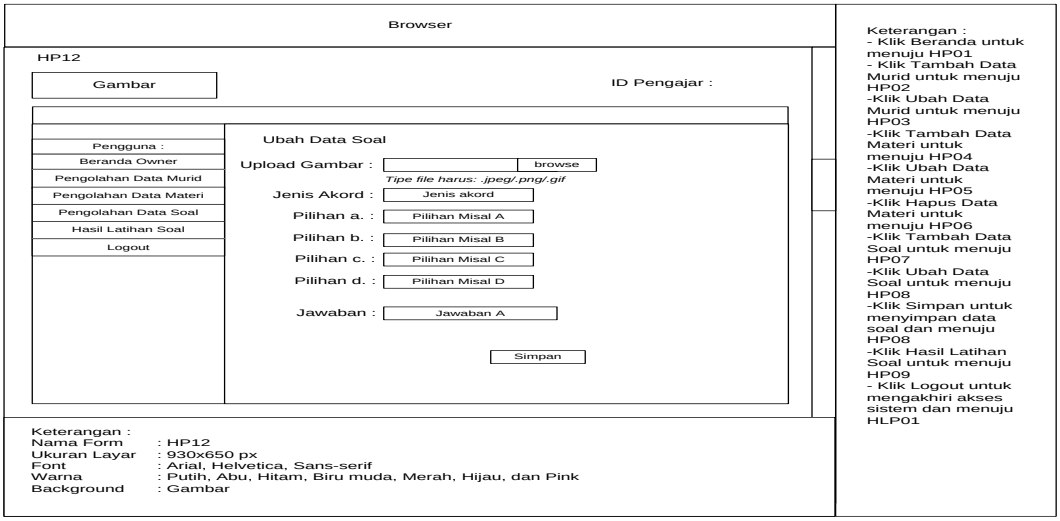
Gambar 3.78 Perancangan Antarmuka Proses Ubah Data Murid

22. Perancangan Antarmuka Proses Ubah Data Materi dapat dilihat pada gambar 3.79.



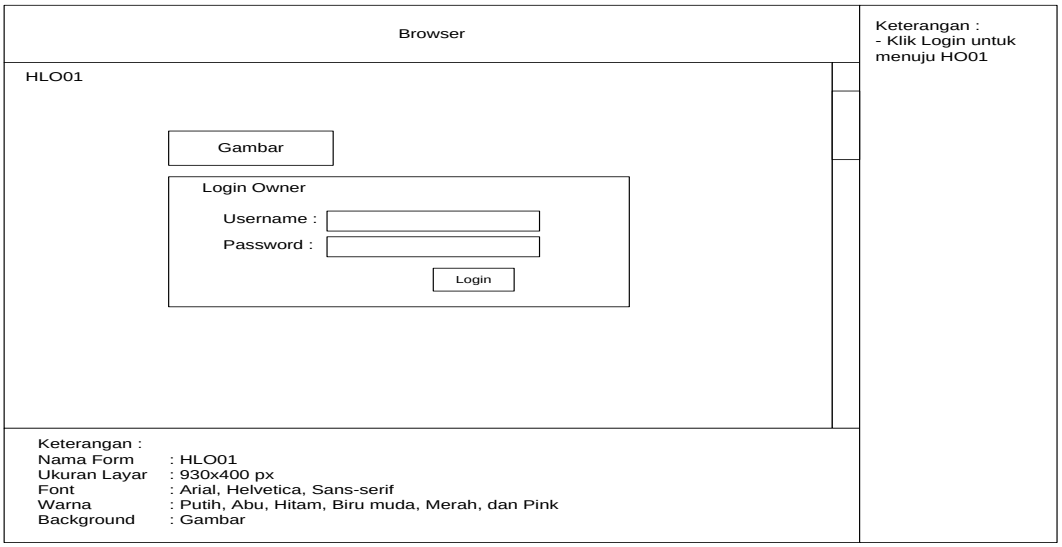
Gambar 3.79 Perancangan Antarmuka Proses Ubah Data Materi

23. Perancangan Antarmuka Proses Ubah Data Soal dapat dilihat pada gambar 3.80.



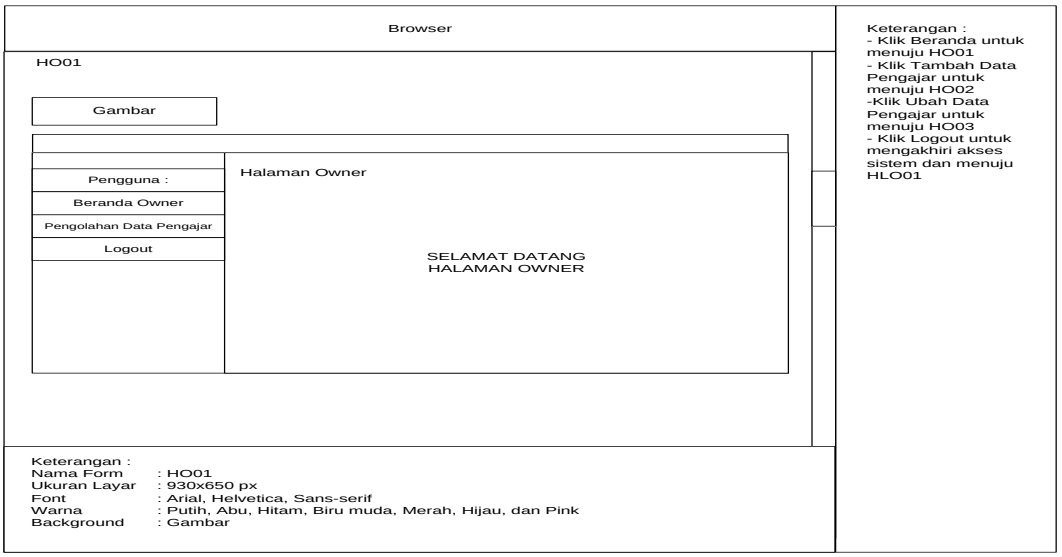
Gambar 3.80 Perancangan Antarmuka Proses Ubah Data Soal

24. Perancangan Antarmuka *Login Owner* dapat dilihat pada gambar 3.81.



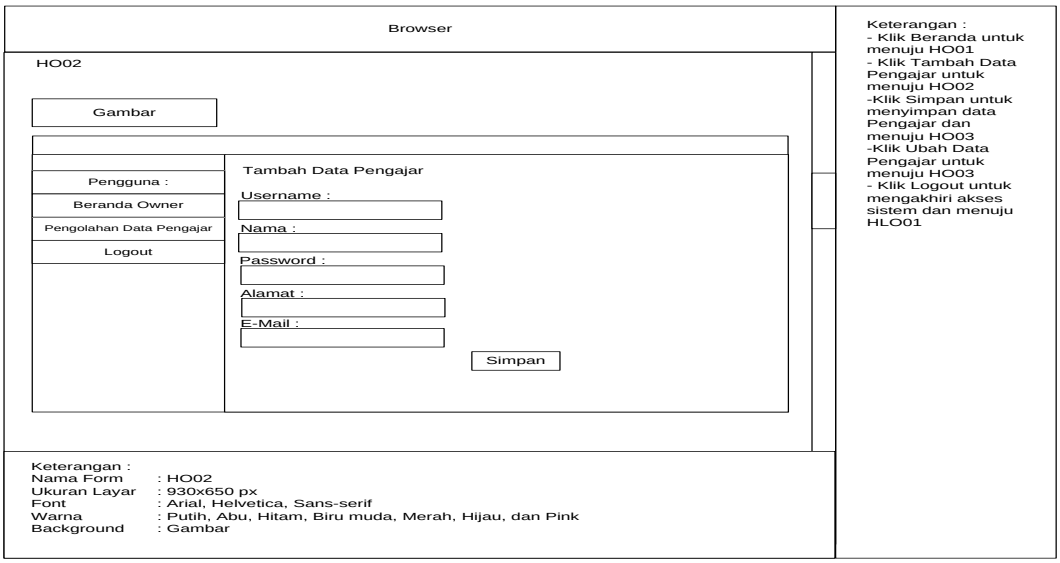
Gambar 3.81 Perancangan Antarmuka *Login Owner*

25. Perancangan Antarmuka Beranda *Owner* dapat dilihat pada gambar 3.82.



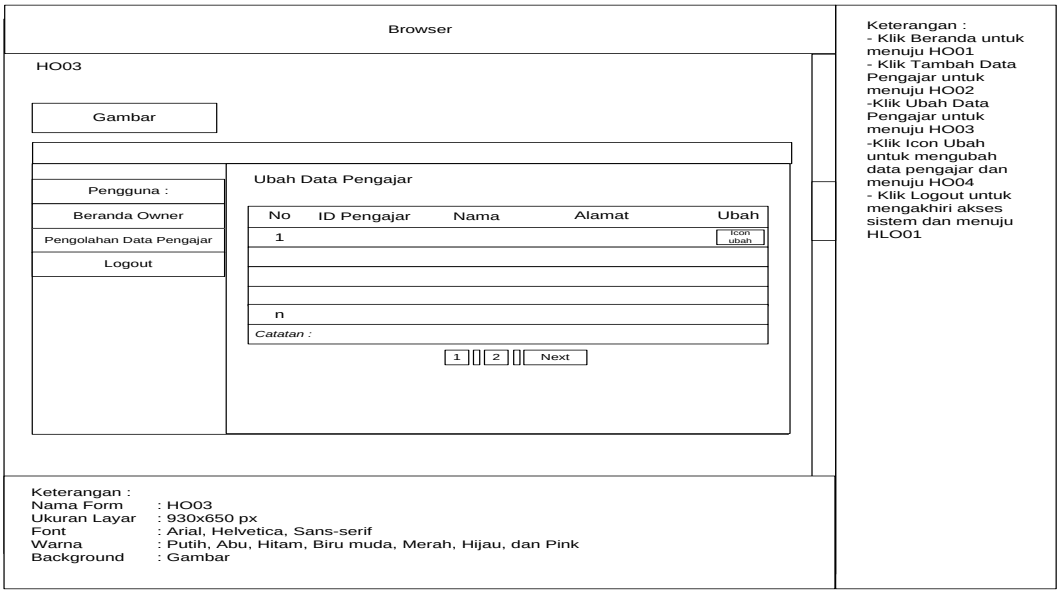
Gambar 3.82 Perancangan Antarmuka Beranda *Owner*

26. Perancangan Antarmuka Tambah Data Pengajar dapat dilihat pada gambar 3.83.



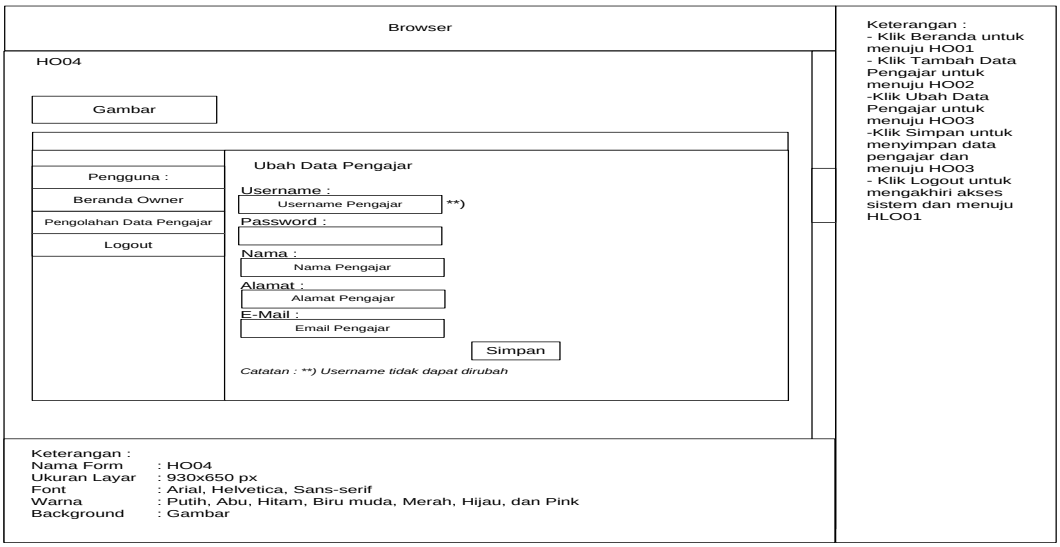
Gambar 3.83 Perancangan Antarmuka Tambah Data Pengajar

27. Perancangan Antarmuka Ubah Data Pengajar dapat dilihat pada gambar 3.84.



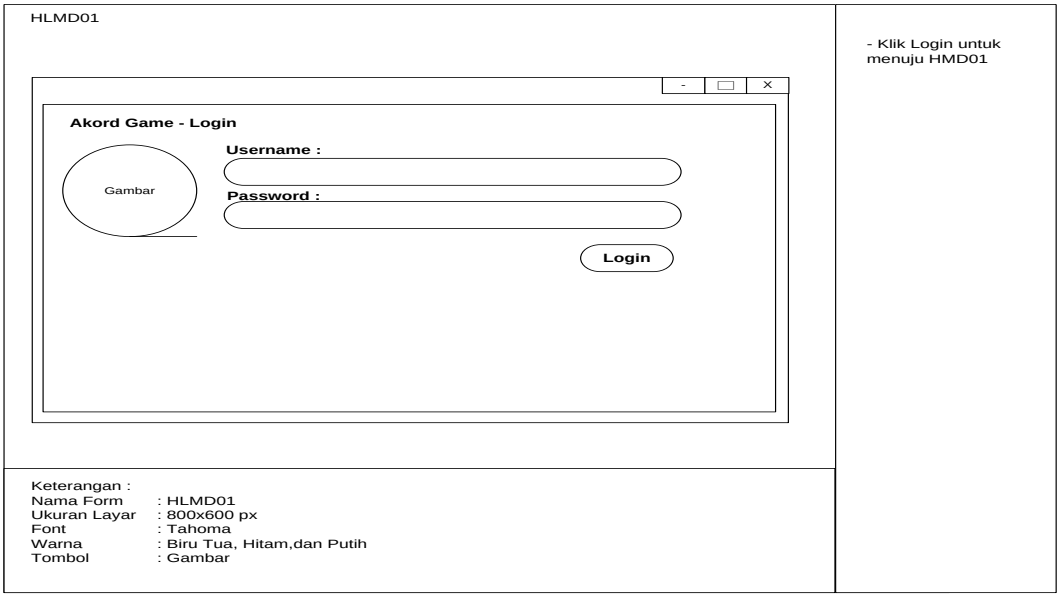
Gambar 3.84 Perancangan Antarmuka Ubah Data Pengajar

28. Perancangan Antarmuka Proses Ubah Data Pengajar dapat dilihat pada gambar 3.85.



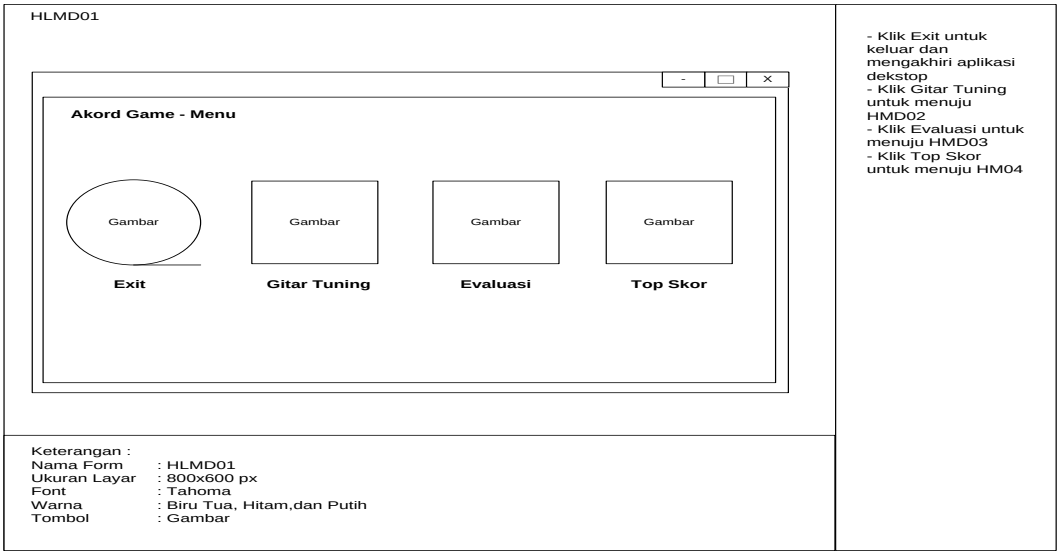
Gambar 3.85 Perancangan Antarmuka Proses Ubah Data Pengajar

29. Perancangan Antarmuka *Login* Aplikasi Akord *Game* dapat dilihat pada gambar 3.86:



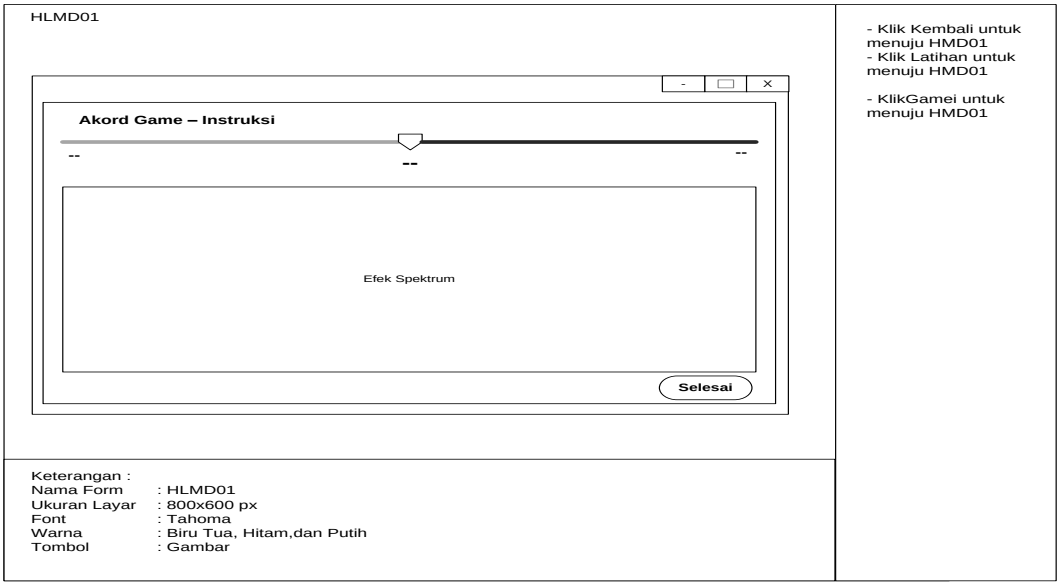
Gambar 3.86 Perancangan Antarmuka *Login* Akord *Game*

30. Perancangan Antarmuka Menu Aplikasi (Akord *Game*) dapat dilihat pada gambar 3.87.



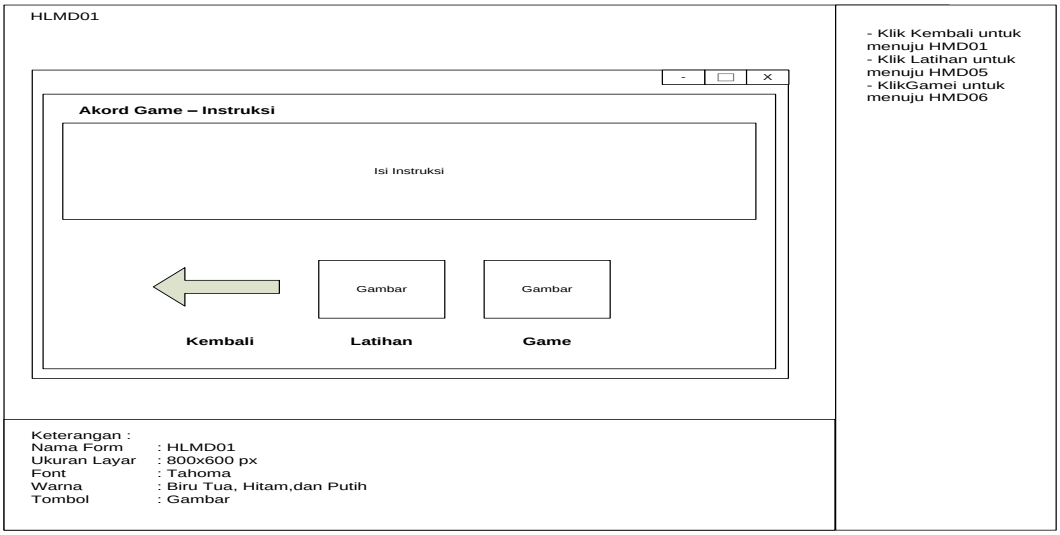
Gambar 3.87 Perancangan Antarmuka Menu Aplikasi (Akord *Game*)

31. Perancangan Antarmuka Gitar Tuning (Akord *Game*) dapat dilihat pada gambar 3.88:



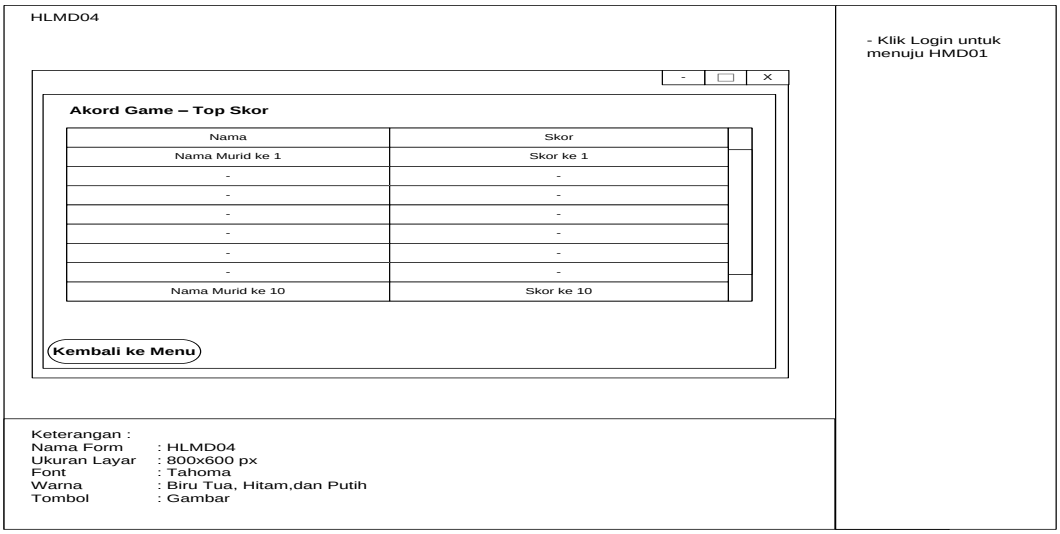
Gambar 3.88 Perancangan Antarmuka Gitar Tuning (Akord *Game*)

32. Perancangan Antarmuka Instruksi Evaluasi (Akord *Game*) dapat dilihat pada gambar 3.89.



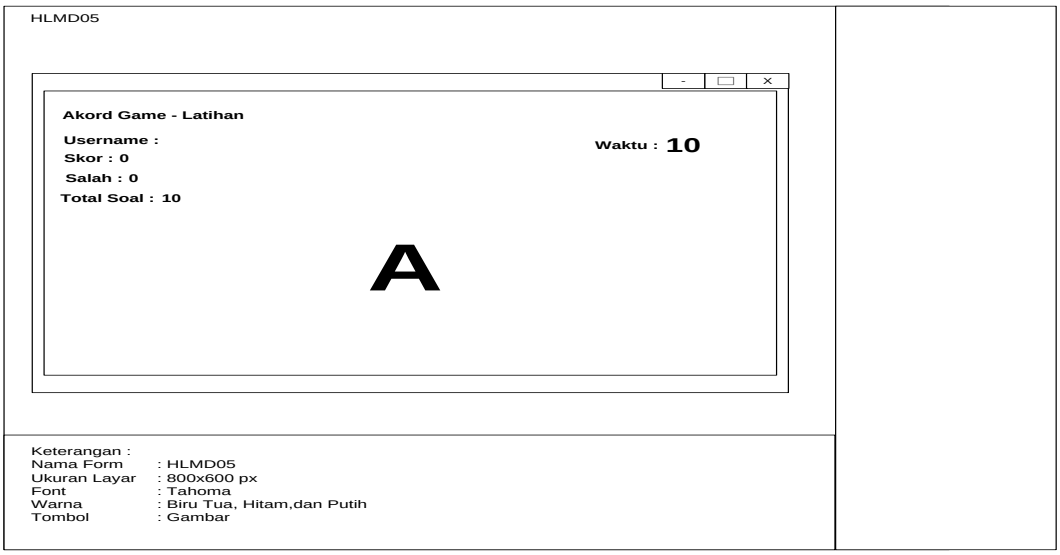
Gambar 3.89 Perancangan Antarmuka Instruksi *Game* (Akord *Game*)

33. Perancangan Antarmuka Top Skor Aplikasi Evaluasi (Akord *Game*) dapat dilihat pada gambar 3.90:



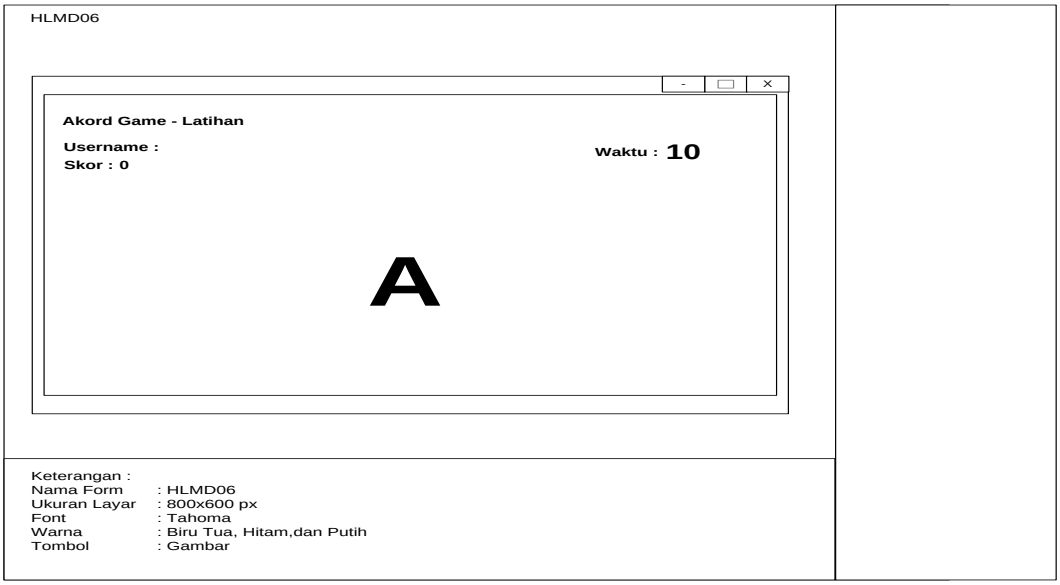
Gambar 3.90 Perancangan Antarmuka Instruksi Aplikasi (Akord *Game*)

34. Perancanga Antarmuka Latihan Aplikasi Evaluasi (Akord *Game*) dapat dilihat pada gambar 3.91.



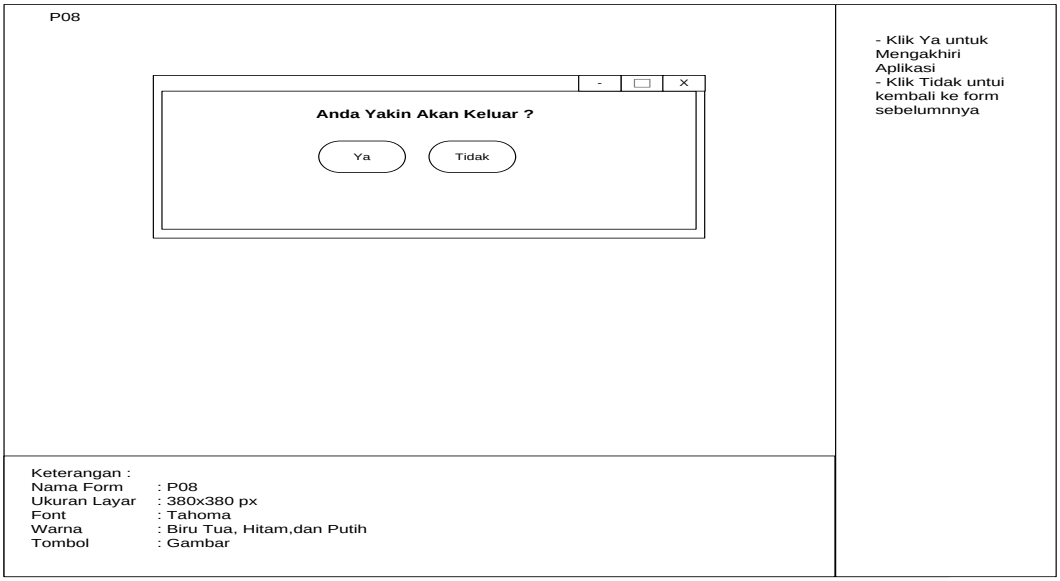
Gambar 3.91 Perancangan Antarmuka Latihan

35. Perancangan Antarmuka *Game* Aplikasi Evaluasi (Akord *Game*) dapat dilihat pada gambar 3.92:



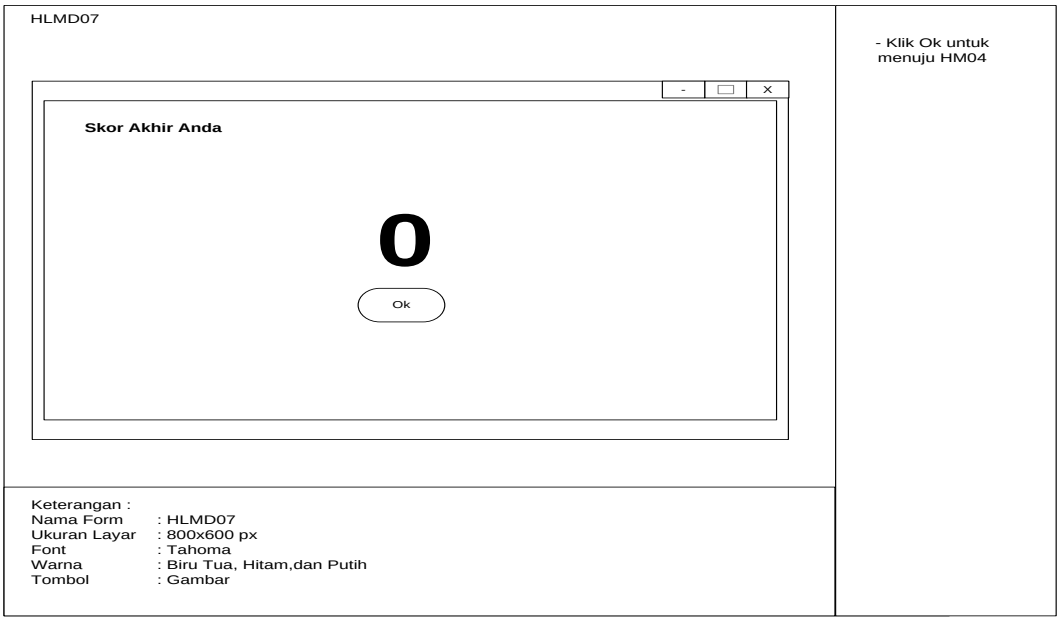
Gambar 3.92 Perancangan Antarmuka *Game*

36. Perancangan tampilan konfirmasi exit aplikasi evaluasi (Akord *Game*) dapat dilihat pada gambar 3.93.



Gambar 3.93 Tampilan Konfirmasi Exit Akord *Game*

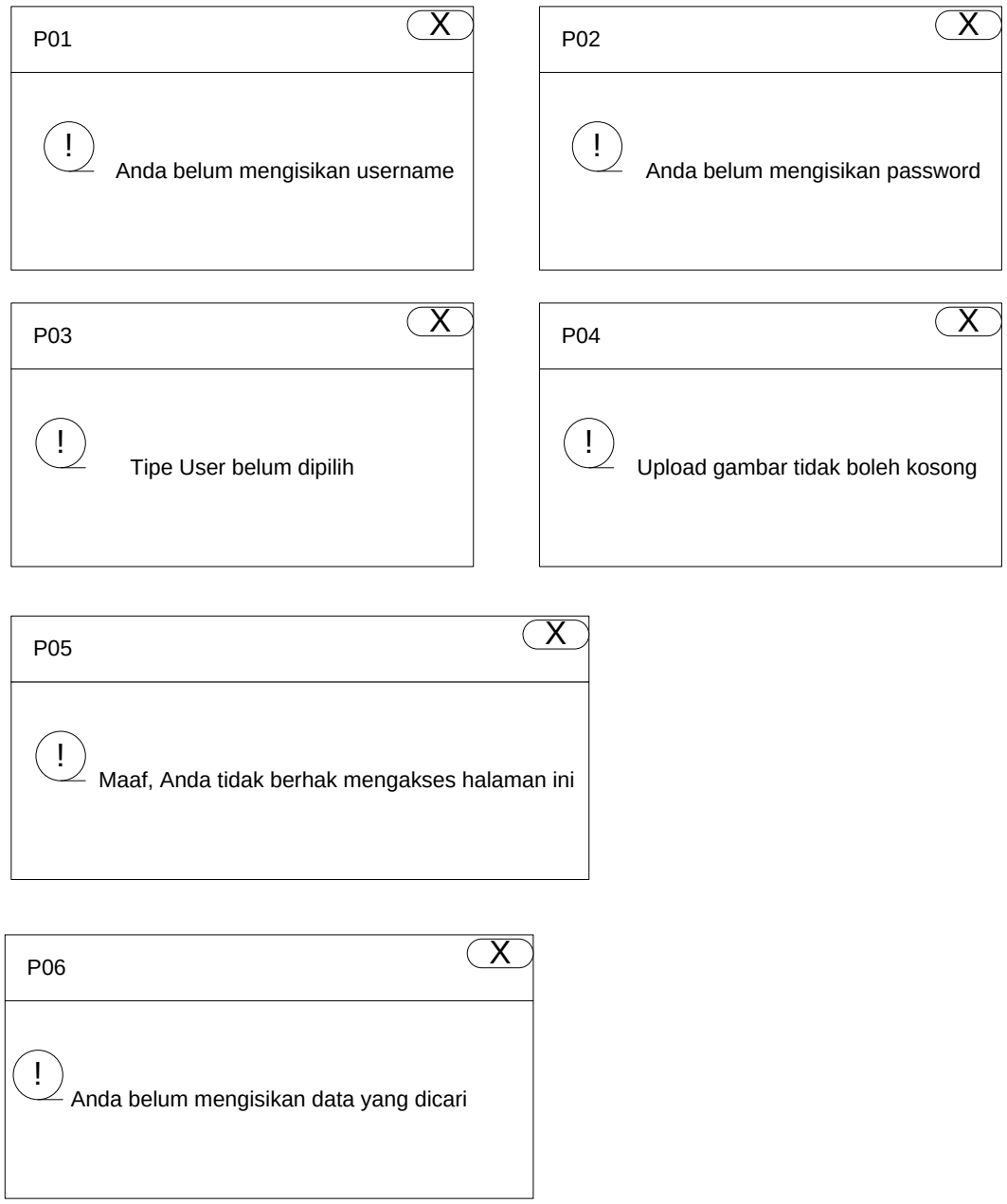
37. Perancangan Antarmuka Hasil Skor dapat dilihat pada gambar 3.94:



Gambar 3.94 Perancangan Antarmuka Hasil Skor

3.5.4.2. Perancangan Pesan

Perancangan pesan bertujuan untuk memberikan pesan dari setiap kondisi yang ada pada sistem dengan menggunaka *Java Script* dan *Java.* Perancanga pesan yang digunakan pada sistemdapa dilihat pada gambar 3.95.

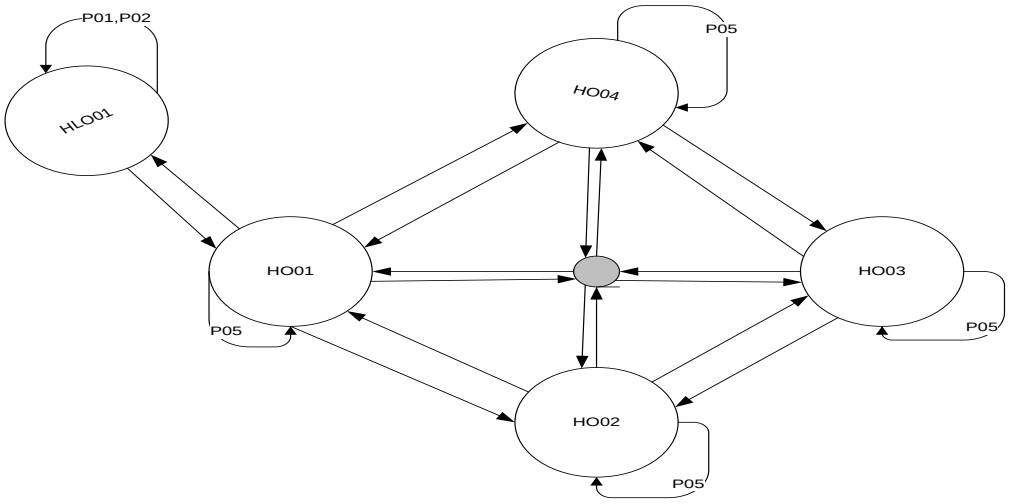


Gambar 3.95 Perancangan Pesan

3.5.4.3. Jaringan Semantik

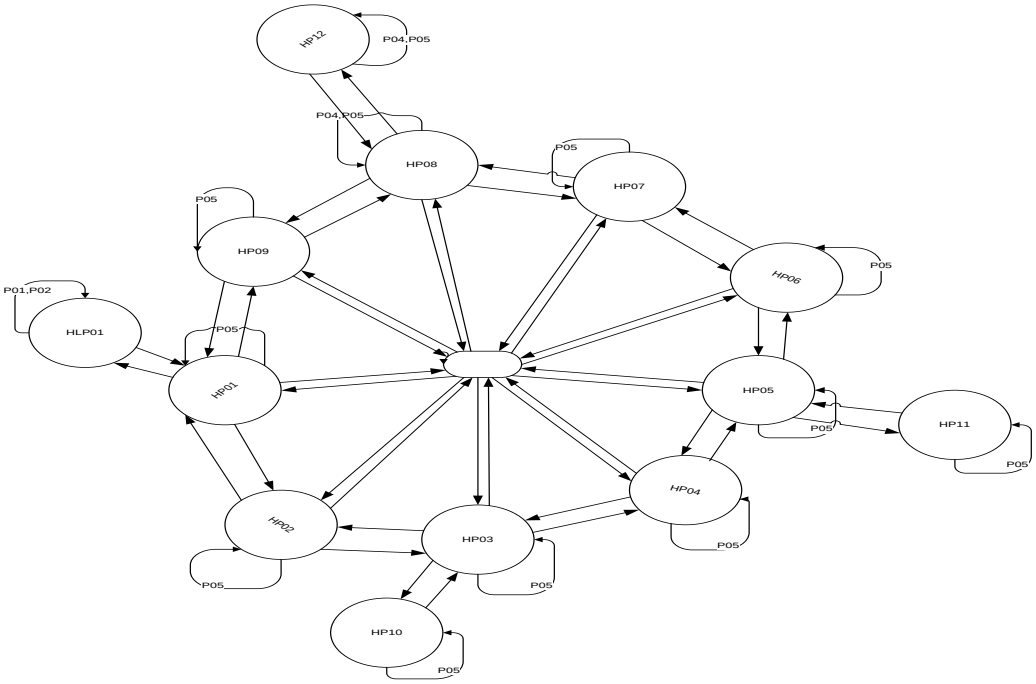
Jaringan semantik merupakan gambaran alur grafis yang menunjukan hubungan antar halaman. Alur antarmuka yang terdapat pada sistem pengenalan dan pembelajaran akord, yaitu:

1. Jaringan Semantik *Owner* dapat dilihat pada gambar 3.96.



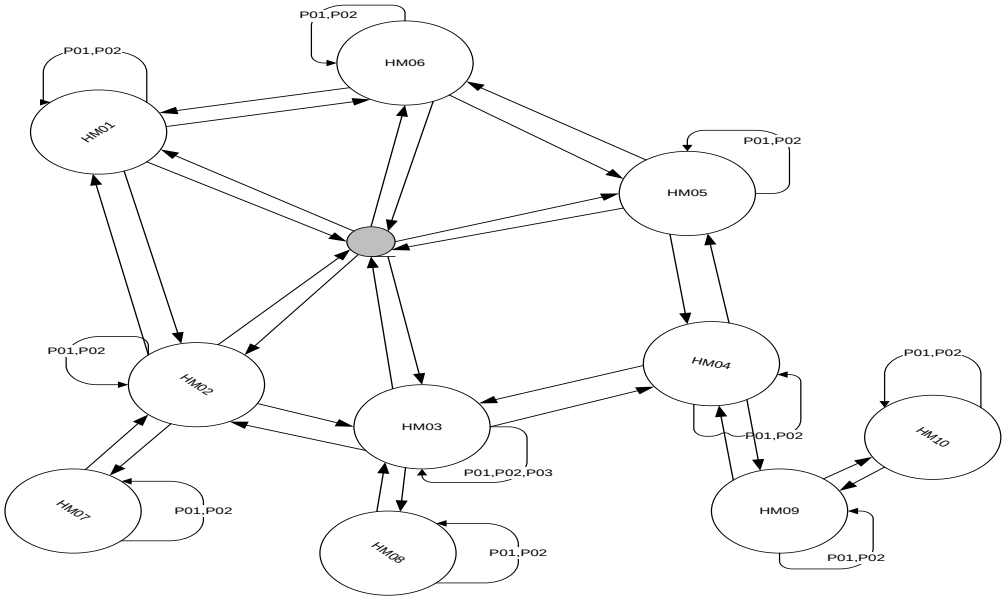
Gambar 3.96 Jaringan Semantik *Owner*

2. Jaringan Semantik Pengajar dapat dilihat pada gambar 3.97.



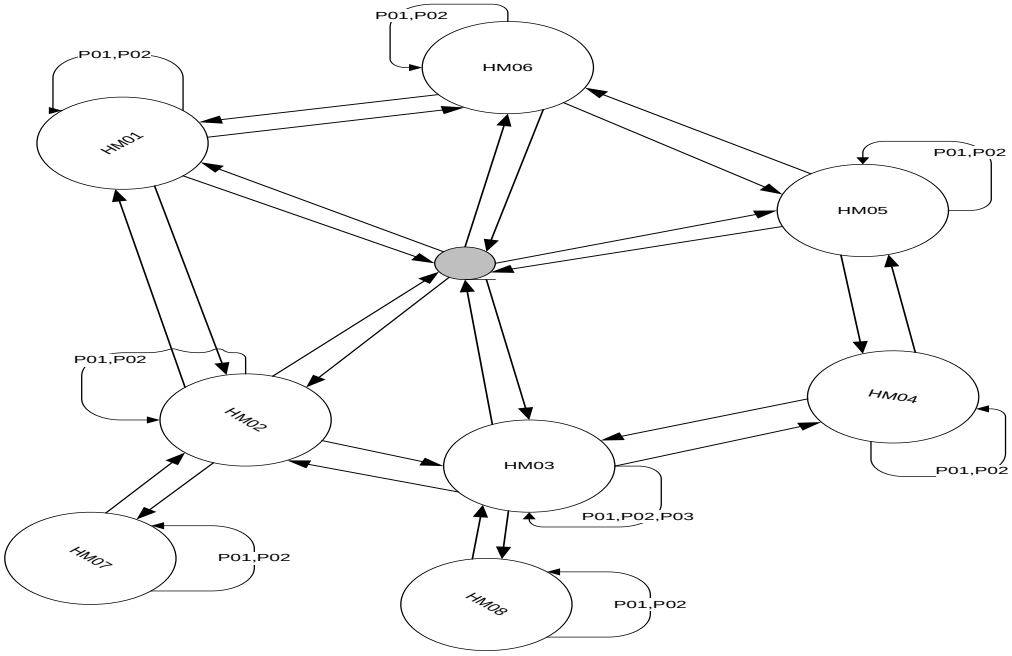
Gambar 3.97 Jaringan Semantik Pengajar

3. Jaringan Semantik Murid dapat dilihat pada gambar 3.98.



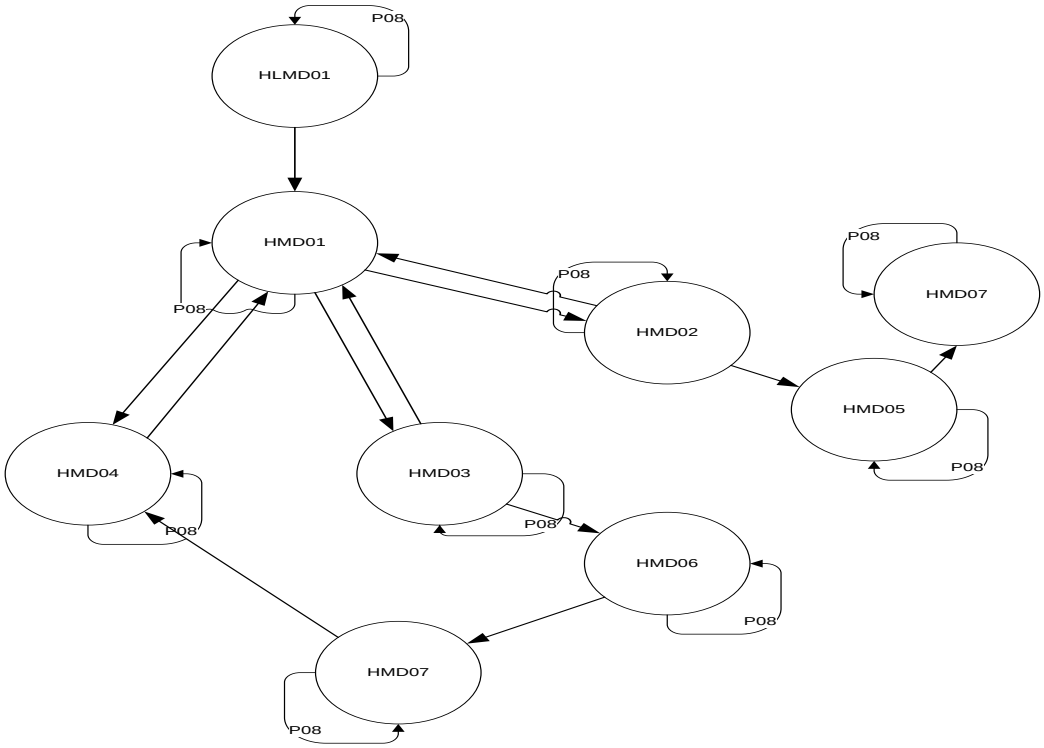
Gambar 3.98 Jaringan Semantik Murid

4. Jaringan Semantik Pengunjung dapat dilihat pada gambar 3.99.



Gambar 3.99 Jaringan Semantik Pengunjung

5. Jaringan Semantik Murid Dekstop dapat dilihat pada gambar 3.100.

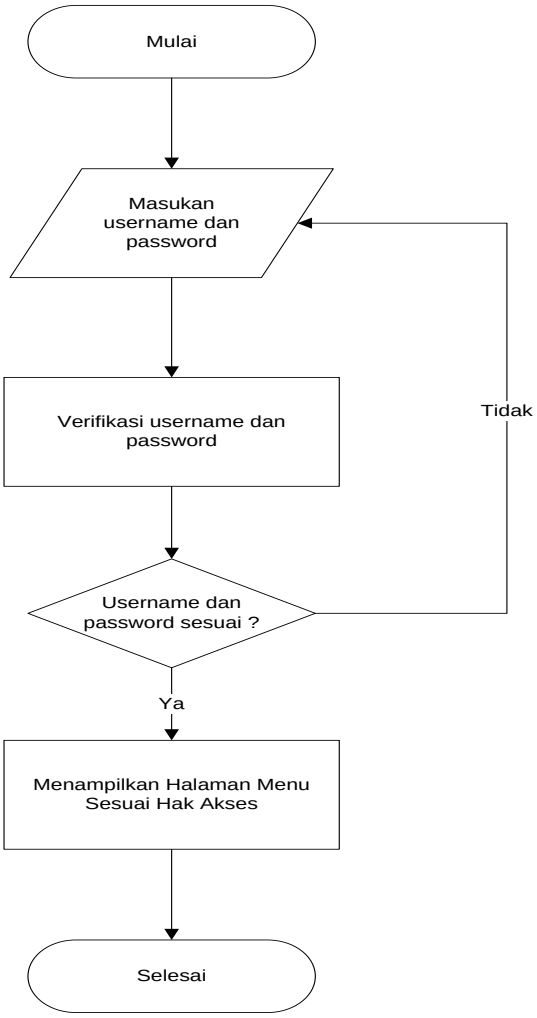


Gambar 3.100 Jaringan Semantik Murid (Akord Game)

3.5.5. Perancangan Prosedural

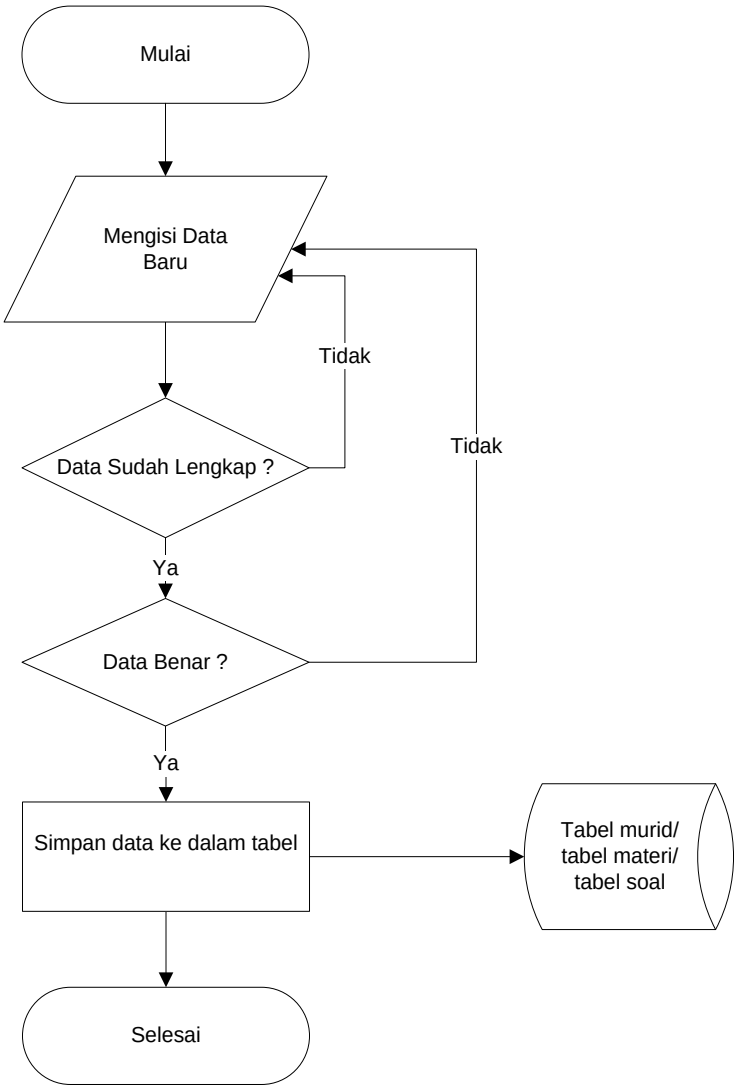
Perancangan prosedural merupakan perancangan yang dilakukan untuk menetapkan *detail* algoritma yang akan dinyatakan ke dalam suatu program. Adapun perancangan prosedural untuk aplikasi pengenalan dan pembelajaran akord, yaitu:

1. *Flowchart login*, merupakan prosedur yang terjadi ketika pengguna mulai mengakses sistem. *Flowchart login* dapat dilihat pada gambar 3.101.



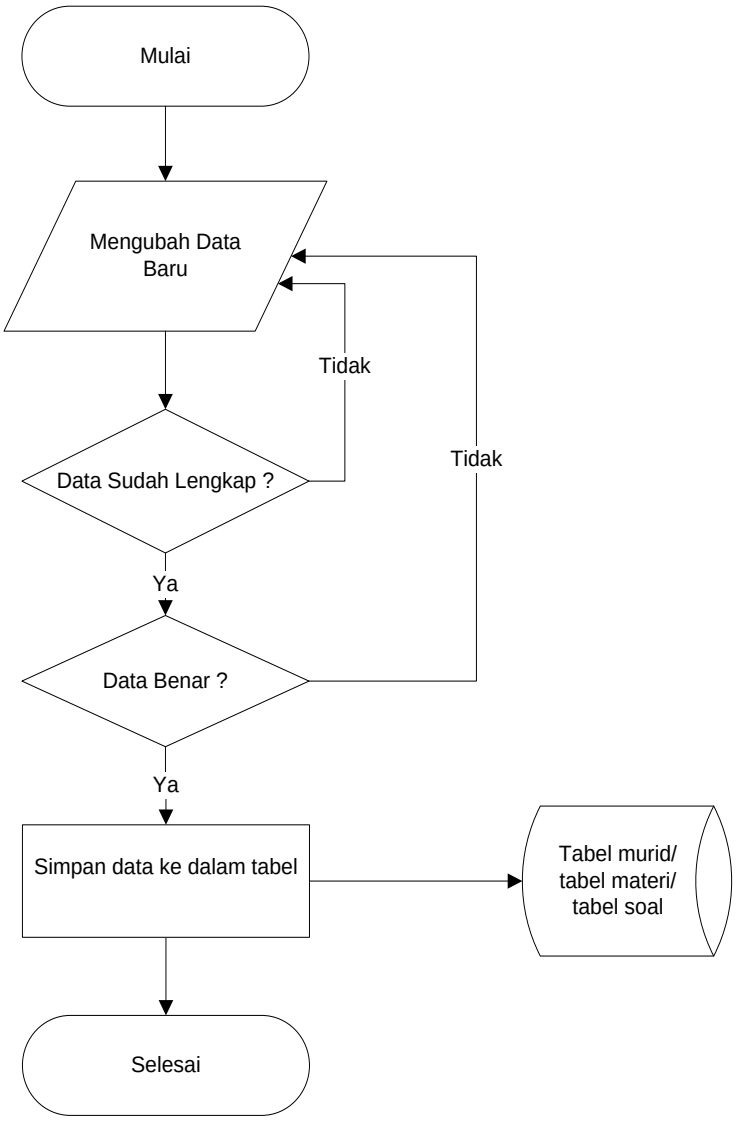
Gambar 3.101 *Flowchart Login*

2. *Flowchart* tambah data, merupakan prosedur yang terjadi ketika pengguna menambahkan sebuah data. *Flowchart* tambah data dapat dilihat pada gambar 3.102.



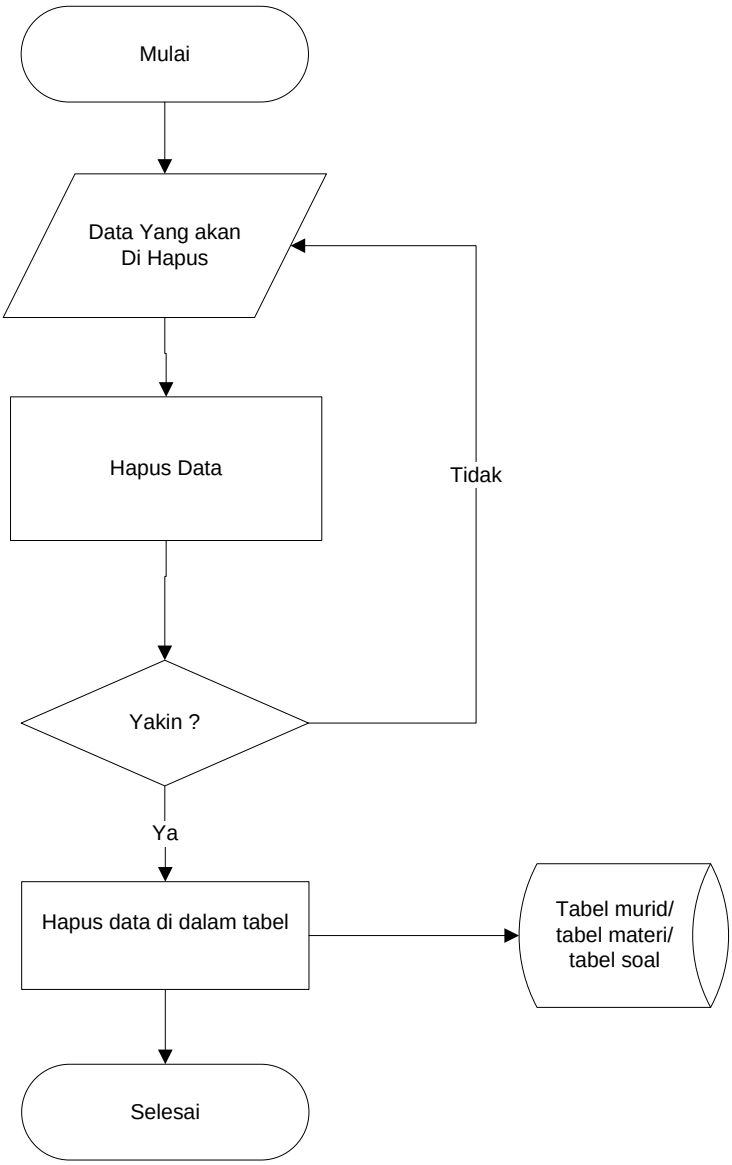
Gambar 3.102 *Flowchart* Tambah Data

3. *Flowchart* ubah data, merupakan prosedur yang terjadi ketika pengguna mengubah sebuah data pengajar. *Flowchart* ubah data dapat dilihat pada gambar 3.103.



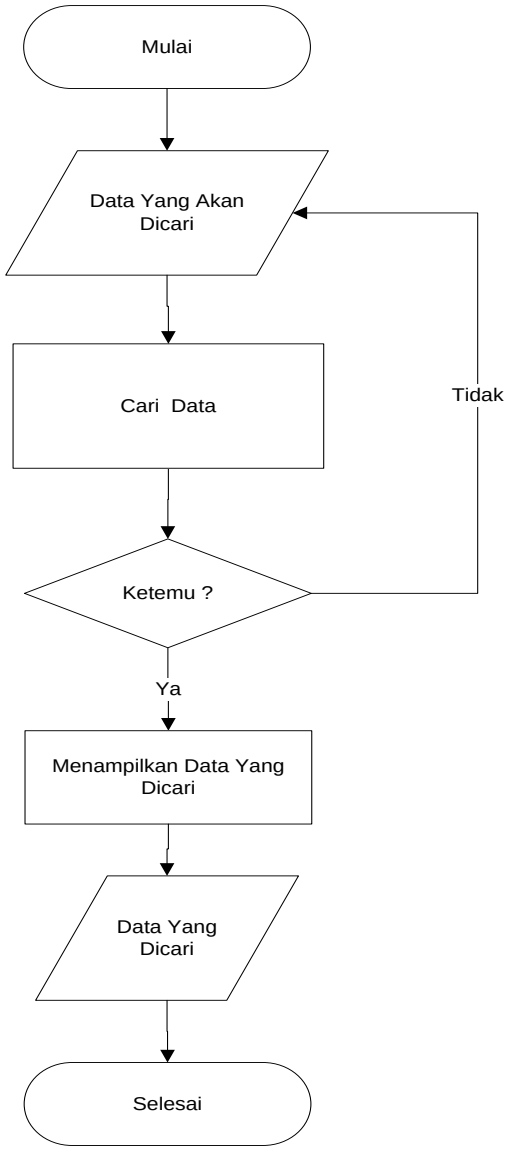
Gambar 3.103 *Flowchart* Ubah Data

4. *Flowchart* hapus data, merupakan prosedur yang terjadi ketika pengguna untuk menghapus sebuah data. *Flowchart* hapus data dapat dilihat pada gambar 3.104.



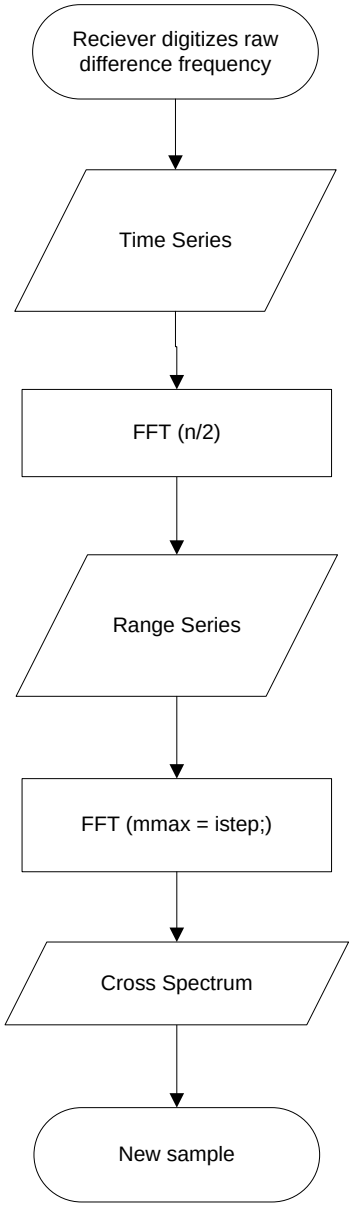
Gambar 3.104 *Flowchart* Hapus Data

5. *Flowchart* cari data, merupakan prosedur yang terjadi ketika pengguna untuk mencari sebuah data. *Flowchart* cari data dapat dilihat pada gambar 3.105.



Gambar 3.105 *Flowchart* Cari Data

6. Flowchart *fast fourier transform,* merupakan prosedur untuk mencari dan membagi sinyal digital menjadi pertapis (sebagai penyederhanaan sinyal digital yang kemudian menjadi sebuah sinyal baru yang akan dicocokan dengan data yang telah ditetapkan). *Flowchart fast fourier transform* dapat dilihat pada gambar 3.106.



Gambar 3.106 ***Flowchart Fast Fourier Transform***