BAB 3
ANALISIS DAN PERANCANGAN

3.1 Gambaran Umum Sistem

Sistem *e-learning* yang akan dibangun terdiri dari empat bagian utama, yaitu:

1. Bagian admin.

   Pada bagian admin akan mengelola sistem secara keseluruhan yaitu pengolahan data guru, siswa, tahun ajaran, mata pelajaran, forum komunikasi, kelas, kepala sekolah, pengumuman, bantuan, mengajar.
2. Bagian guru.

   Pada bagian ini akan berisi fitur pengolahan, materi pelajaran, tugas, soal latihan, pengumuman, forum komunikasi.
3. Bagian siswa.

   Berisi fitur untuk mengikuti kegiatan belajar, materi pelajaran, tugas, nilai, soal latihan, forum komunikasi.
4. Bagian kepala sekolah.

   Bagian ini akan berisi fitur berupa grafik perkembangan siswa terhadap materi yang telah diberikan oleh guru berdasarkan latihan yang telah dikerjakan.

3.2 Analisis Sistem

Analisis sistem merupakan analisis yang dilakukan untuk mengidentifikasi masalah-masalah pada sistem yang sedang berjalan di SMK-PU. Dari hasil analisis sistem akan digunakan menjadi dasar dari perancangan sistem *e-learning.*

3.2.1 Analisis Masalah

Siswa sulit mendapatkan materi pelajaran ketika guru yang bersangkutan berhalangan hadir, hal ini disebabkan karena tidak ada guru piket yang dapat menggantikan guru yang berhalangan hadir tersebut. Kurangnya waktu dan tempat komunikasi antara siswa dan guru apabila masih ada pelajaran yang belum

dipahami oleh siswa. Kurangnya latihan soal terhadap suatu materi pada akhir proses belajar mengajar yang mengakibatkan siswa sulit memahami materi yang telah disampaikan. Penerimaan laporan perkembangan siswa pada akhir semester mengakibatkan kepala sekolah sulit mengetahui perkembangan siswa secara langsung.

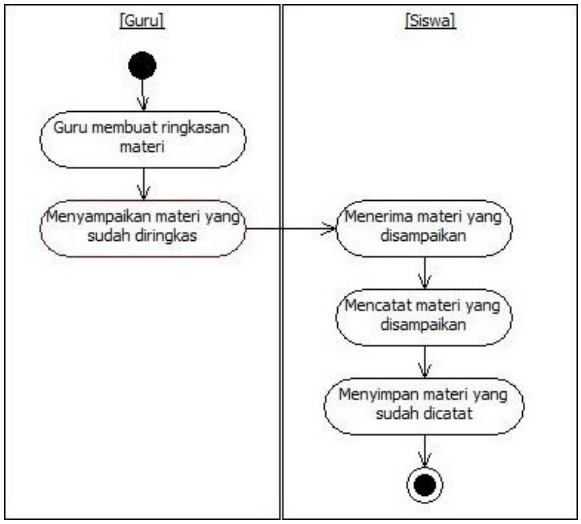
3.2.2 Analisis Prosedur yang Sedang Berjalan

Analisis prosedur yang sedang berjalan bertujuan untuk mengetahui secara detail sistem pembelajaran yang sedang berjalan pada SMK-PU Prov. Jawa Barat. Adapun prosedur yang sedang berjalan saat ini meliputi prosedur penyampaian materi, pemberian tugas, latihan.

1. Prosedur penyampaian materi yang sedang berjalan.

   Tahapan proses penyampaian materi digambarkan pada Gambar 3.1 dan akan dijelaskan sebagai berikut :

   a. Guru membuat ringkasan materi.
   b. Ringkasan materi yang telah dibuat oleh guru akan disimpan sebagai arsip guru.
   c. Ringkasan materi yang telah dibuat oleh guru disampaikan ke siswa.
   d. Siswa mencatat ringkasan materi yang telah dibuat oleh guru.
   e. Catatan ringkasan materi yang telah dibuat oleh guru disimpan sebagai arsip siswa.

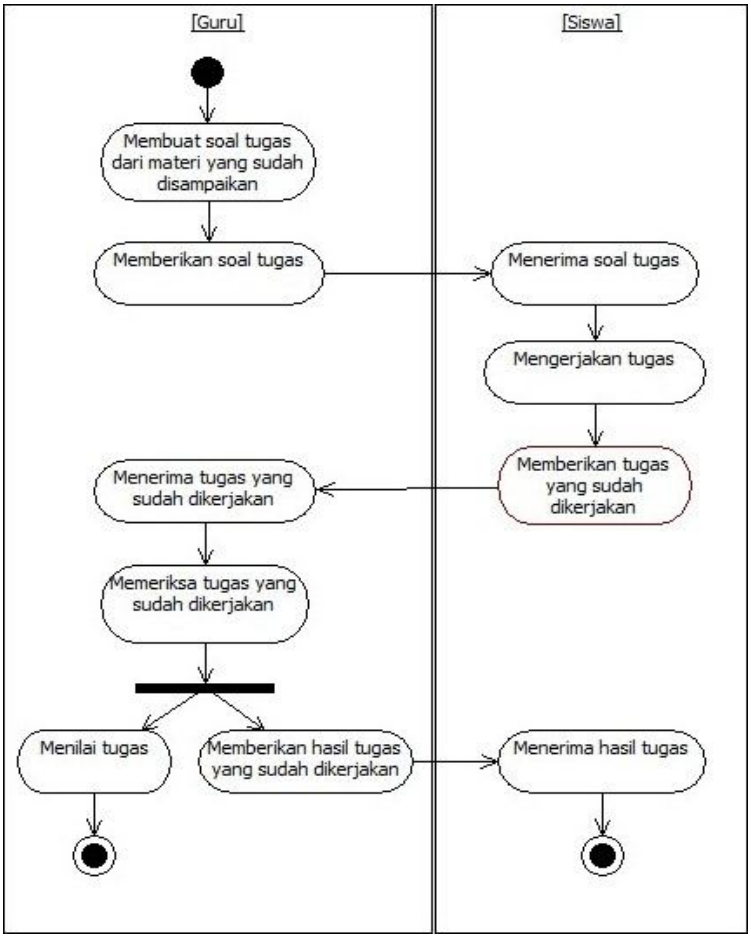


Gambar 3.1 ***Activity Diagram Penyampaian Materi***

2. Prosedur pemberian tugas.

   Tahapan proses pemberian tugas digambarkan pada Gambar 3.2 dan akan dijelaskan sebagai berikut :

   a. Guru membuat tugas dari materi yang sudah disampaikan ke siswa.
   b. Siswa mengerjakan tugas yang diberikan oleh guru.
   c. Tugas yang sudah dikerjakan oleh siswa dikumpulkan ke guru.
   d. Guru memeriksa tugas yang telah dikerjakan oleh siswa dan menentukan hasil nilai.
   e. Guru memberikan hasil tugas yang telah diperiksa ke siswa.



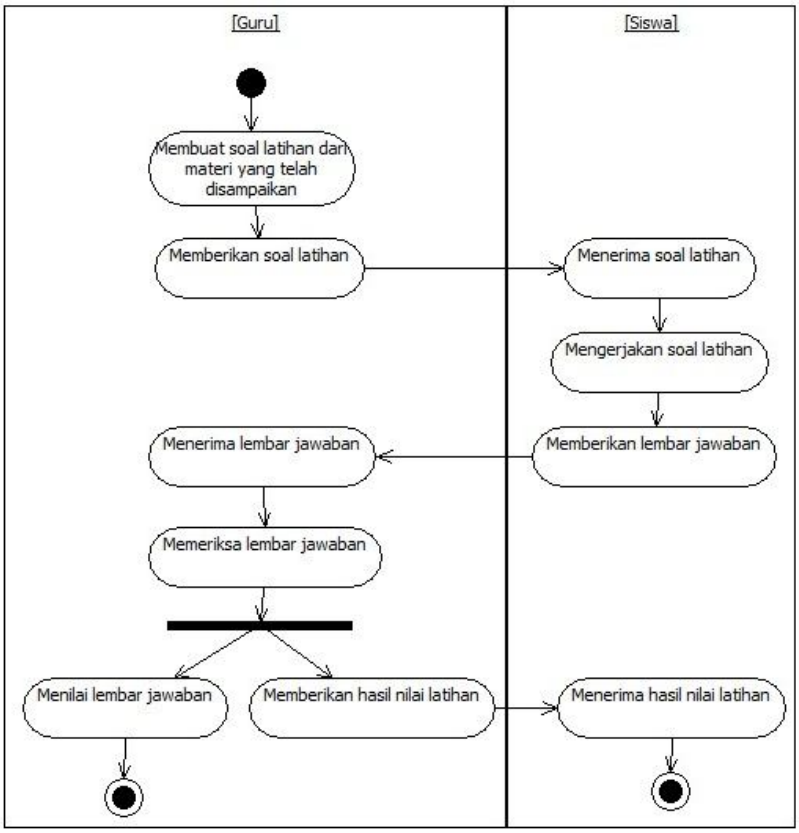
Gambar 3.2 ***Activity Diagram Pemberian Tugas***

3. Prosedur latihan.

   Tahapan proses latihan digambarkan pada Gambar 3.3 dan akan dijelaskan sebagai berikut :

   a. Guru membuat soal latihan dari materi yang sudah disampaikan.

b. Siswa mengerjakan soal latihan yang diberikan oleh guru pada lembar jawaban.
c. Lembar jawaban dan soal latihan yang telah dikerjakan oleh siswa dikumpulkan ke guru.
d. Guru memeriksa lembar jawaban siswa dan menentukan hasil nilai.
e. Hasil nilai dari lembar jawaban yang telah diperiksa oleh guru dimasukkan pada daftar nilai.
f. Lembar jawaban yang telah diperiksa dibagikan ke siswa.



Gambar 3.3 ***Activity Diagram Latihan***

3.2.3 Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi Kebutuhan Perangkat Lunak dibedakan menjadi dua yaitu Kebutuhan Fungsional dan Kebutuhan Non Fungsional. Daftar kebutuhan fungsional pada aplikasi *e-learning* SMK-PU Negeri Prov. Jawa Barat dapat dilihat pada Tabel 3.1.

Tabel 3.1 Daftar Kebutuhan Fungsional

No	SKPL-ID	Kebutuhan
1	SKPL-F-01	Sistem menyediakan fungsionalitas *login* untuk admin, guru, dan siswa
2	SKPL -F-02	Sistem menyediakan fungsionalitas untuk pengolahan data guru yang dilakukan oleh admin
3	SKPL -F-03	Sistem menyediakan fungsionalitas untuk pengolahan data siswa yang dilakukan oleh admin
4	SKPL -F-04	Sistem menyediakan fungsionalitas untuk pengolahan data kepala sekolah yang dilakukan oleh admin
5	SKPL -F-05	Sistem menyediakan fungsionalitas untuk pengolahan data mata pelajaran yang dilakukan oleh admin
6	SKPL -F-06	Sistem menyediakan fungsionalitas untuk pengolahan data mengajar yang dilakukan oleh admin
7	SKPL -F-07	Sistem menyediakan fungsionalitas untuk pengolahan data kelas yang dilakukan oleh admin
8	SKPL -F-08	Sistem menyediakan fungsionalitas untuk pengolahan data tahun ajaran yang dilakukan oleh admin
9	SKPL -F-09	Sistem menyediakan fungsionalitas untuk pengolahan data bantuan yang dilakukan oleh admin
10	SKPL -F-10	Sistem menyediakan fungsionalitas untuk pengolahan data forum yang dilakukan oleh admin
11	SKPL -F-11	Sistem menyediakan fungsionalitas untuk melihat data siswa yang dilakukan oleh guru
12	SKPL -F-12	Sistem menyediakan fungsionalitas untuk melihat data guru yang dilakukan oleh guru
13	SKPL -F-13	Sistem menyediakan fungsionalitas untuk pengolahan data materi yang dilakukan oleh guru. Dapat dilakukan jika telah mengajar mata pelajaran.
14	SKPL -F-14	Sistem menyediakan fungsionalitas untuk pengelolaan data tugas yang dilakukan oleh guru. Dapat dilakukan jika telah mengajar mata pelajaran.
15	SKPL -F-15	Sistem menyediakan fungsionalitas untuk pengelolaan data latihan yang dilakukan oleh guru. Dapat dilakukan jika telah mengajar mata pelajaran.
16	SKPL -F-16	Sistem menyediakan fungsionalitas untuk pengelolaan data pengumuman yang dilakukan oleh guru
17	SKPL -F-17	Sistem menyediakan fungsionalitas untuk melihat daftar guru yang dilakukan oleh siswa
18	SKPL -F-18	Sistem menyediakan fungsionalitas untuk penyajian data materi yang dilakukan oleh siswa
19	SKPL -F-19	Sistem menyediakan fungsionalitas untuk melakukan latihan yang dilakukan oleh siswa
20	SKPL -F-20	Sistem menyediakan fungsionalitas untuk mengunduh materi yang dilakukan oleh siswa

21	SKPL -F-21	Sistem menyediakan fungsionalitas untuk pengelolaan data forum, yang dilakukan oleh siswa
22	SKPL -F-22	Sistem menyediakan fungsionalitas untuk melihat daftar guru yang dilakukan oleh kepala sekolah
23	SKPL -F-23	Sistem menyediakan fungsionalitas untuk melihat daftar siswa yang dilakukan oleh kepala sekolah

Sedangkan daftar kebutuhan non fungsional aplikasi *e-learning* SMK-PU Prov. Jawa Barat dapat dilihat pada Tabel 3.2.

Tabel 3.2 ***Analisis Kebutuhan Non Fungsional***

No	SKPL-Id	Kebutuhan
1	SKPL-NF-01	Sistem tersedia dalam bentuk web yang dapat diakses manggunakan *internet* dan *web browser*
2	SKPL-NF-02	Resolusi website yang disarankan 1024x768
3	SKPL-NF-03	Sistem menggunakan bahasa Indonesia sebagai bahasa komunikasi
4	SKPL-NF-04	Sistem *online* 24x7 dalam seminggu
5	SKPL-NF-05	Jenis data yang digunakan untuk *database* adalah sql
6	SKPL-NF-06	Tipe gambar yang digunakan dalam sistem adalah jpg, jpeg, dan png.
7	SKPL-NF-07	Sistem menggunakan *web hosting* sebagai media tempat penyimpanan data *web* dan *database*
8	SKPL-NF-08	Pengguna sistem yaitu admin, guru, siswa, kepala sekolah

3.2.4 Analisis Kebutuhan Non Fungsional

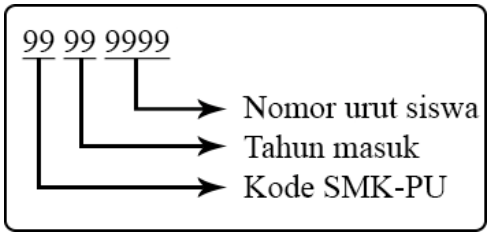
Analisis kebutuhan non fungsional mencangkup analisis pengkodean, analisis perangkat keras, analisis perangkat lunak, identifikasi aktor.

3.2.4.1 Analisis Pengkodean

Pengkodean pada SMK-PU terdiri dari pengkodean Nomor Induk Siswa (NIS) untuk siswa/i, Nomor Induk Pegawai (NIP) untuk guru dan kode kelas.

1. Pengkodean Nomor Induk Siswa (NIS)

   Pengkodean NIS pada SMK-PU ini terdiri dari 8 digit, dengan format yang dapat dilihat pada Gambar 3.4 berikut ini :

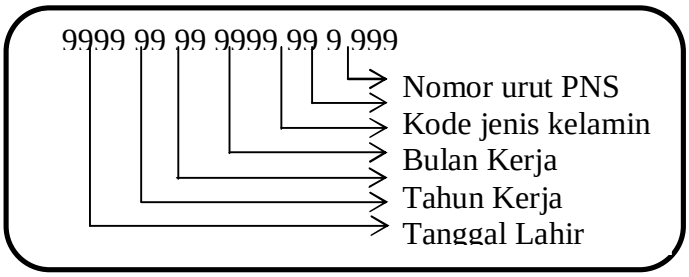


Gambar 3.4 ***Format Pengkodean NIS***

Contoh 10127643 diartikan bahwa siswa tersebut bernomor urut 7643, masuk pada 10 yaitu SMK-PU pada tahun 2012.

2. Pengkodean Nomor Induk Pegawai (NIP)

   Pengkodean NIP terdiri dari 18 digit, dengan format yang dapat dilihat pada Gambar 3.5 berikut ini :

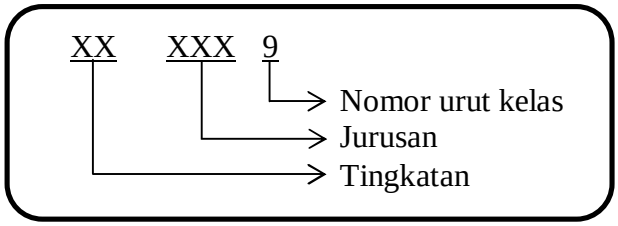


Gambar 3.5 ***Format Pengkodean NIP***

Contoh 197305091999031108 diartikan bahwa guru tersebut lahir pada 09 Mei 1973, diangkat menjadi pegawai negeri pada bulan Maret 1999, berjenis kelamin laki-laki dengan nomor urut PNS (Pegawai Negeri Sipil) 108.

3. Pengkodean kelas

   Pengkodean kelas terdiri dari 6 digit, dengan format pengkodean dapat dilihat pada Gambar 3.6 berikut ini :



Gambar 3.6 ***Format Pengkodean Kelas***

Contoh XII TKJ 1 diartikan bahwa tingkatan XII jurusan Teknik Komputer Jaringan kelas dengan nomor urut 1.

3.2.4.2 Analisis Perangkat Keras

Sistem *e-learning* yang akan dibangun diimplementasikan pada jaringan *internet*. Perangkat keras yang digunakan pada SMK-PU Negeri Prov. Jawa Barat adalah sebagai berikut :

1. Prosessor Intel Celeron atau setara dengan kecepatan 2.2 GHz.
2. RAM 128-512MB.
3. *Harddisk* dengan kapasitas 40GB.
4. Monitor dengan resolusi sebesar 1024x768 *pixel.*

Untuk menjalankan aplikasi *e-learning* yang akan dibangun membutuhkan perangkat keras dengan spesifikasi minimum sebagai berikut :

1. Pentium III atau setara dengan kecepatan 800MHz.
2. RAM 128MB.
3. *Harddisk* dengan ruang kosong 500MB.
4. Monitor dengan resolusi sebesar 1024x768 *pixel.*

Berdasarkan data tersebut, maka sekolah telah mencukupi kebutuhan perangkat keras dari sistem *e-learning* yang akan dibangun.

3.2.4.3 Analisis Perangkat Lunak

Perangkat lunak merupakan program komputer yang menghubungkan antara pengguna dengan perangkat keras. Adapun perangkat lunak yang digunakan pada sekolah ini adalah sebagai berikut :

1. Sistem operasi *Windows XP* dengan *Web browser* menggunakan *Mozilla Firefox, Google Chorme,* dan *Internet Explorer.*
2. *MySql* sebagai *Database Management System (DBMS).*

Perangkat lunak yang dibutuhkan untuk membangun dan menjalankan sistem aplikasi *e-learning* ini adalah sebagai berikut :

1. Sistem operasi *Windows XP.*
2. *Macromedia Dreamwever CS4* untuk pembuatan coding

3. *Adobe Photoshop CS6* untuk editing gambar yang akan digunakan pada aplikasi *e-learning.*
4. *Microsoft Visio, StarUML* untuk perancangan.
5. XAMPP 1.7.3 sebagai *web server* saat pembangunan aplikasi.
6. DBMS menggunakan *MySql.*
7. *Mozilla firefox, Google Chorme, Internet Explorer* sebagai *browser.*

3.2.4.4 Identifikasi Aktor

Aplikasi *E-learning* yang akan dibangun digunakan oleh empat orang pengguna, diantaranya admin, kepala sekolah, guru, dan siswa. Berikut deskripsi masing-masing aktor :

1. Aktor pertama adalah admin sebagai pengelola sistem secara keseluruhan dan mempunyai hak akses penuh yaitu menambah, mengubah, dan menghapus data guru, siswa, tahun ajaran, mata pelajaran, tugas, forum komunikasi, kelas, kepala sekolah, pengumuman
2. Aktor kedua yaitu kepala sekolah yang memiliki hak akses hanya melihat perkembangan siswa dan guru yang aktif dalam sistem *e-learning* ini.
3. Aktor ketiga yaitu guru memiliki hak akses untuk melakukan penambahan, pengubahan, penghapusan, materi pelajaran, tugas, forum komunikasi, soal latihan.
4. Aktor keempat yaitu siswa memiliki hak akses untuk melakukan melihat pengumuman, data materi pelajaran, data mata pelajaran, nilai serta penambahan dan pengubahan data pada forum komunikasi.

3.2.4.5 Aturan Bisnis

Aturan bisnis pada SMK-PU Prov. Jawa Barat yang sedang berlangsung saat ini adalah sebagai berikut :

1. Guru tidak dapat melakukan upload materi, upload tugas, dan membuat latihan soal apabila guru tersebut tidak mengajar mata pelajaran

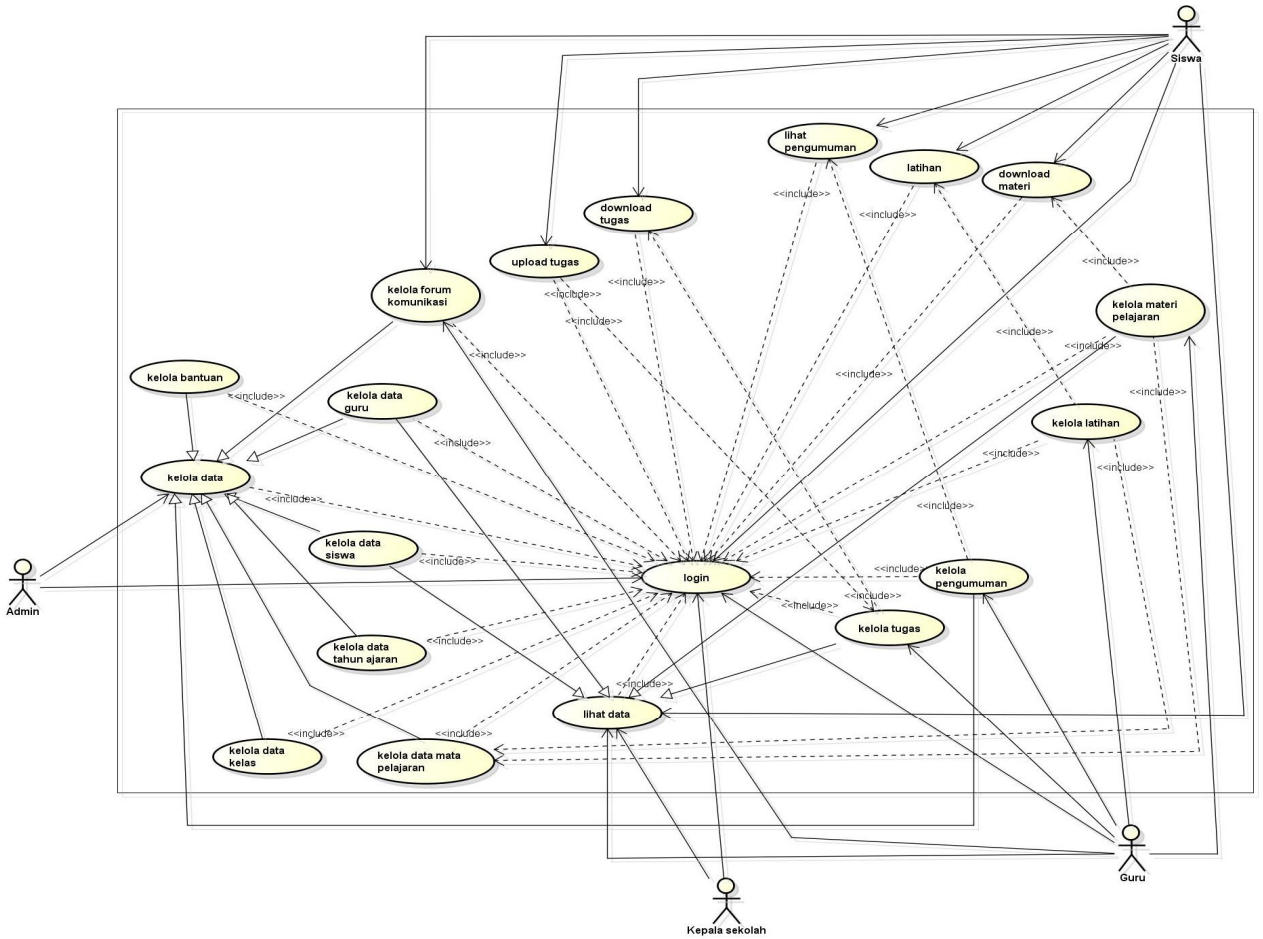
2. Materi diupload oleh guru berdasarkan mata pelajaran yang diajar oleh guru tersebut.
3. Tugas yang diberikan oleh guru berupa file yang berisi soal tugas berdasarkan mata pelajaran yang diajar.
4. Jika pengumpulan tugas melebihi waktu yang telah ditentukan maka tugas yang sudah dikerjakan tidak bisa di upload oleh siswa.
5. Tugas yang sudah diupload oleh siswa akan dikoreksi dan diberikan nilai sesuai tugas yang telah diupload.
6. Siswa mengerjakan latihan yang telah dibuat oleh guru.
7. Siswa hanya bisa mengerjakan soal berdasarkan waktu yang telah ditentukan.
8. Latihan soal akan automatis berhenti apabila waktu yang dibutuhkan telah berakhir.

3.2.5 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional meliputi *use case* diagram, *use case* sekenario, *sequence* diagram, dan *class* diagram.

3.2.5.1 Use Case Diagram

Use case diagram merupakan interaksi antara sistem dengan aktor, termasuk pertukaran pesan dan tindakan yang sedang berlangsung. *Use case* diagram menggambarkan fungsionalitas dari sistem, sehingga aktor atau pengguna sistem mengerti mengenai kegunaan sistem yang akan dibangun. Adapun *use case* diagram pada SMK-PU dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Use Case* Diagram *E-Learning SMK-PU Negeri Prov. Jawa Barat*

3.2.5.2 Use Case Sekenario

Use case sekenario merupakan penjelasan interaksi antar aktor dan *use case* diagram dalam melakukan sebuah proses. Berikut *use case* sekenario aplikasi *e-learning* pada SMK-PU Prov. Jawa Barat :

1. *Use case* sekenario login

   Interaksi antara aktor admin, kepala sekolah, guru, dan siswa dengan *use case login* dapat dilihat pada Tabel 3.3 di bawah ini.

Tabel 3.3 ***Sekenario Use Case Login***

		Identifikasi	
Nama		Login	
Tujuan		Masuk kedalam sistem	
Deskripsi		Proses login merupakan proses autentikasi menggunakan hak akses sebagai admin, kepala sekolah, guru, siswa untuk menggunakan sistem	
Aktor		Admin, kepala sekolah, guru, siswa	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Sistem menampilkan form login, dimana form login ada pada halaman *index* pada sistem	
Aksi		**Reaksi Sistem**	
1	Aktor mengisi form login		
2	Aktor menekan tombol login	3	Sistem mengecek data pada tabel di *database.*
		4	Bila data ada dan cocok dengan yang dimaksud maka akan masuk kedalam sistem dan akan menampilkan menu utama sesuai dengan hak akses.
Sekenario Alternatif (Autentikasi gagal)			
Aksi			**Reaksi Sistem**
			Sistem menampilkan pesan data login salah.
3	Aktor mengisi kembali form login		
4	Aktor menekan tombol login	5	Sistem mengecek data yang diisi pada *database.*
			Bila data id ada dan cocok maka akan masuk kedalam sistem dan menampilkan menu utama sesuai hak

			akses.
Kondisi Akhir			Berhasil masuk kedalam sistem dan menampilkan menu utama.

2. Interaksi antara aktor admin dengan *use case* pengelolaan tahun ajaran dapat dilihat pada Tabel 3.4 di bawah ini.

Tabel 3.4 ***Sekenario Use Case Kelola Tahun Ajaran***

		Identifikasi	
Nama		Kelola data tahun ajaran	
Tujuan		Mengolah data tahun ajaran	
Deskripsi		Proses untuk tambah tahun ajaran	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi		**Reaksi Sistem**	
1	Aktor memilih menu tahun ajaran	2	Sistem menampilkan tahun ajaran beserta form tahun ajaran yang telah diisi secara automatis oleh sistem, dimana form tahun ajaran berisikan tahun ajaran yang akan dating.
3	Aktor memilih tombol simpan	4	Sistem menyimpan data tahun ajaran pada database
Kondisi Akhir		Admin berhasil mengolah tahun ajaran	

3. Interaksi antara aktor admin dengan *use case* pengelolaan data guru dibagi menjadi empat proses yaitu proses import data guru, tambah data, ubah data, dan hapus data.
   a. Interaksi antar aktor admin dengan use case pengelolaan data guru pada proses import data guru dapat dilihat pada Tabel 3.5 di bawah ini.

Tabel 3.5 ***Sekenario Use Case Import Data Guru***

	Identifikasi
Nama	Import data guru
Tujuan	Import data guru
Deskripsi	Proses untuk import data guru
Aktor	Admin
Use case yang berkaitan	-
	Sekenario yang berkaitan
Kondisi Awal	Menampilkan halaman admin

Aksi		Reaksi Sistem	
1	Memilih menu guru	2	Menampilkan data guru
3	Memilih import data	4	Menampilkan direktori komputer
5	Memilih file yang akan diimport		
6	Memlih tombol simpan	7	Mengecek file yang diimport dengan format xls
		8	Bila file yang dipilih sesuai dengan format xls maka data disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Tidak sesuai dengan format xls)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error file yang dipilih tidak sesuai dengan format
9	Memilih kembali import file	10	Menampilkan direktori komputer
11	Memilih file yang akan diimport		
12	Memlih tombol simpan	13	Mengecek file yang diimport dengan format xls
		14	Bila file yang dipilih sesuai dengan format xls maka data disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil import data guru	

b. Interaksi antar aktor admin dengan use case pengelolaan data guru pada proses tambah data guru dapat dilihat pada Tabel 3.6 di bawah ini.

Tabel 3.6 ***Sekenario Use Case Tambah Data Guru***

		Identifikasi	
Nama		Tambah data guru	
Tujuan		Menambah data guru	
Deskripsi		Proses untuk tambah data guru	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu guru	2	Menampilkan data guru
3	Memilih menu tambah	4	Menampilkan form tambah data guru
5	Mengisi data guru		

6	Memilih tombol simpan	7	Mengecek validasi data guru
		8	Bila data valid maka data guru disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data guru tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data guru		
10	Memilih tombol simpan	11	Mengecek validasi data guru
		12	Bila data valid maka data guru disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil tambah data guru	

c. Interaksi antar aktor admin dengan use case pengelolaan data guru pada proses ubah data guru dapat dilihat pada Tabel 3.7 di bawah ini.

Tabel 3.7 ***Sekenario Use Case Ubah Data Guru***

		Identifikasi	
Nama		Ubah data guru	
Tujuan		Mengubah data guru	
Deskripsi		Proses untuk ubah data guru	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu guru	2	Menampilkan data guru
3	Memilih menu ubah	4	Menampilkan form ubah data guru
5	Mengisi data guru yang baru		
6	Memilih tombol simpan	7	Mengecek validasi data guru
		8	Bila data valid maka data guru disimpan pada database dan menampilkan pesan data berhasil disimpan

Sekenario Alternatif (Data guru tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data guru yang baru		
10	Memilih tombol simpan	11	Mengecek validasi data guru
		12	Bila data valid maka data guru disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil merubah data guru	

d. Interaksi antar aktor admin dengan use case pengelolaan data guru pada proses hapus data guru dapat dilihat pada Tabel 3.8 di bawah ini.

Tabel 3.8 ***Sekenario Use Case Hapus Data Guru***

		Identifikasi	
Nama		Hapus data guru	
Tujuan		Menghapus data guru	
Deskripsi		Proses untuk hapus data guru	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu guru	2	Menampilkan data guru
3	Memilih hapus guru	4	Menampilkan form konfirmasi hapus
		5	Konfirmasi Ok maka data guru dihapus dari database
			Konfirmasi Batal data guru batal dihapus
Kondisi Akhir		Admin berhasil mengolah data guru	

4. Interaksi antara aktor dengan *use case* pengelolaan data kelas dibagi menjadi dua proses yaitu tambah data kelas dan hapus data kelas
   a. Interaksi antara aktor dengan *use case* pengelolaan data kelas pada porses tambah data kelas dapat dilihat pada Tabel 3.9 di bawah ini.

Tabel 3.9 ***Sekenario Use Case Tambah Data Kelas***

		Identifikasi	
Nama		Tambah data kelas	
Tujuan		Menambah data kelas	
Deskripsi		Proses untuk tambah data kelas	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi			**Reaksi Sistem**
1	Memilih menu kelas	2	Menampilkan data kelas dan form kelas
3	Mengisi data kelas pada form kelas		
4	Memilih tombol simpan	5	Mengecek validasi data kelas
		6	Bila data kelas valid maka data kelas disimpan pada database
Sekenario Alternatif (Data kelas tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
7	Mengisi kembali data kelas		
8	Memilih tombol simpan	9	Mengecek validasi data kelas
		10	Bila data kelas valid maka data kelas disimpan pada database
Kondisi Akhir		Admin berhasil menambah data kelas	

b. Interaksi antara aktor dengan *use case* pengelolaan data kelas pada porses hapus data kelas dapat dilihat pada Tabel 3.10 di bawah ini.

Tabel 3.10 Sekenario Use Case Hapus Data Kelas

		Identifikasi
Nama		Hapus data kelas
Tujuan		Menghapus data kelas
Deskripsi		Proses untuk hapus data kelas
Aktor		Admin
Use case yang berkaitan		-
		Sekenario yang berkaitan
Kondisi Awal		Menampilkan halaman admin
Aksi		**Reaksi Sistem**

1	Memilih menu kelas	2	Menampilkan data kelas
3	Memilih menu hapus kelas	4	Menampilkan form konfirmasi hapus
		5	Konfirmasi Ok maka data kelas dihapus dari database
			Konfirmasi Batal data kelas batal dihapus
Kondisi Akhir		Admin berhasil menghapus data kelas	

5. Interaksi antara aktor admin dengan *use case* pengelolaan data siswa dibagi menjadi empat proses yaitu proses import data siswa, tambah data, ubah data, dan hapus data
   a. Interaksi antara aktor admin dengan use case pengelolaan data siswa pada proses import data siswa dapat dilihat pada Tabel 3.11 di bawah ini.

Tabel 3.11 ***Sekenario Use Case Import Data Siswa***

		Identifikasi	
Nama		Import data siswa	
Tujuan		Import data siswa	
Deskripsi		Proses untuk import data siswa	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan menu utama admin	
Aksi			**Reaksi Sistem**
1	Memilih menu siswa	2	Menampilkan data siswa
3	Memilih import data	4	Menampilkan direktori komputer
5	Memilih file yang akan diimport		
6	Memlih tombol simpan	7	Mengecek file yang diimport dengan format xls
		8	Bila file yang dipilih sesuai dengan format xls maka data disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Tidak sesuai dengan format xls)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error file yang dipilih tidak sesuai dengan format
9	Memilih kembali import file	10	Menampilkan direktori komputer

11	Memilih file yang akan diimport		
12	Memlih tombol simpan	13	Mengecek file yang diimport dengan format xls
		14	Bila file yang dipilih sesuai dengan format xls maka data disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil import data siswa	

b. Interaksi antara aktor admin dengan use case pengelolaan data siswa pada proses tambah data siswa dapat dilihat pada Tabel 3.12 di bawah ini.

Tabel 3.12 ***Sekenario Use Case Tambah Data Siswa***

		Identifikasi	
Nama		Tambah data siswa	
Tujuan		Menambah data siswa	
Deskripsi		Proses untuk tambah data siswa	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu siswa	2	Menampilkan data siswa
3	Memilih menu tambah	4	Menampilkan form tambah data siswa
5	Mengisi data siswa		
6	Memilih tombol simpan	7	Mengecek validasi data siswa
		8	Bila data valid maka data siswa disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data siswa tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data siswa		
10	Memilih tombol simpan	11	Mengecek validasi data siswa
		12	Bila data valid maka data siswa disimpan pada database dan menampilkan pesan data berhasil

			disimpan
Kondisi Akhir		Admin berhasil tambah data siswa	

c. Interaksi antara aktor admin dengan use case pengelolaan data siswa pada proses ubah data siswa dapat dilihat pada Tabel 3.13 di bawah ini.

Tabel 3.13 ***Sekenario Use Case Ubah Data Siswa***

		Identifikasi	
Nama		Ubah data siswa	
Tujuan		Mengubah data siswa	
Deskripsi		Proses untuk ubah data siswa	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu siswa	2	Menampilkan data siswa
3	Memilih menu ubah	4	Menampilkan form ubah data siswa
5	Mengisi data siswa yang baru		
6	Memilih tombol simpan	7	Mengecek validasi data siswa
		8	Bila data valid maka data siswa disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data siswa tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data siswa yang baru		
10	Memilih tombol simpan	11	Mengecek validasi data siswa
		12	Bila data valid maka data siswa disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil mengubah data siswa	

d. Interaksi antara aktor admin dengan use case pengelolaan data siswa pada proses hapus data siswa dapat dilihat pada Tabel 3.14 di bawah ini.

Tabel 3.14 ***Sekenario Use Case Hapus Data Siswa***

		Identifikasi	
Nama		Hapus data siswa	
Tujuan		Menghapus data siswa	
Deskripsi		Proses untuk hapus data siswa	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman admin	
Aksi			**Reaksi Sistem**
1	Memilih menu siswa	2	Menampilkan data siswa
3	Memilih hapus siswa	4	Menampilkan form konfirmasi hapus
		5	Konfirmasi Ok maka data siswa dihapus dari database
			Konfirmasi Batal data siswa batal dihapus
Kondisi Akhir		Admin berhasil menghapus data siswa	

6. Interaksi antara aktor kepala sekolah dengan *use case* lihat data guru dapat dilihat pada Tabel 3.15 di bawah ini.

Tabel 3.15 ***Sekenario Use Case Lihat Data Guru***

		Identifikasi	
Nama		Lihat data guru	
Tujuan		Menampilkan data guru	
Deskripsi		Proses untuk menampilkan data guru	
Aktor		Kepala sekolah	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan menu utama kepala sekolah	
Aksi			**Reaksi Sistem**
1	Memilih menu guru	2	Mengambil data guru dari database dan menampilkan data guru berdasarkan nip guru
Kondisi Akhir		Halaman data guru berhasil ditampilkan	

7. Interaksi antara aktor kepala sekolah dengan *use case* lihat data siswa dapat dilihat pada Tabel 3.16 di bawah ini.

Tabel 3.16 ***Sekenario Use Case Lihat Data Siswa***

		Identifikasi	
Nama		Lihat data siswa	
Tujuan		Menampilkan data siswa	
Deskripsi		Proses untuk menampilkan data siswa	
Aktor		Kepala sekolah	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan menu utama kepala sekolah	
Aksi		**Reaksi Sistem**	
1	Memilih menu jumlah siswa	2	Mengambil data siswa dari database dan menampilkan jumlah siswa berdasarkan kelas
Kondisi Akhir		Halaman data siswa berhasil ditampilkan	

8. Interaksi antara aktor guru dengan *use case* kelola materi pelajaran dibagi menjadi dua proses yaitu prose tambah materi dan proses hapus materi.
   a. Interaksi antara aktor guru dengan *use case* kelola materi pelajaran pada proses tambah materi dapat dilihat pada Tabel 3.17 di bawah ini.

Tabel 3.17 ***Sekenario Use Case Tambah Materi Pelajaran***

		Identifikasi	
Nama		Tambah materi pelajaran	
Tujuan		Mengambah materi pelajaran	
Deskripsi		Proses untuk tambah materi pelajaran	
Aktor		Guru	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan menu utama guru	
Aksi		**Reaksi Sistem**	
1	Memilih menu materi	2	Menampilkan data materi berdasarkan mata pelajaran yang diajarkan oleh guru
3	Memilih menu tambah materi	4	Menampilkan form tambah materi pelajaran
5	Mengisi data materi pelajaran		
6	Memilih tombol simpan	7	Mengecek validasi data materi
		8	Bila data valid maka data materi disimpan pada database dan

			menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data Materi Pelajaran tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data materi yang baru		
10	Memilih tombol simpan	11	Mengecek validasi data materi
		12	Bila data valid maka data materi disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Guru berhasil menambah materi pelajaran	

b. Interaksi antara aktor guru dengan *use case* kelola materi pelajaran pada proses hapus materi dapat dilihat pada Tabel 3.18 di bawah ini.

Tabel 3.18 ***Sekenario Use Case Hapus Materi Pelajaran***

		Identifikasi	
Nama		Hapus materi pelajaran	
Tujuan		Menghapus materi pelajaran	
Deskripsi		Proses untuk hapus materi pelajaran	
Aktor		Guru	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan menu utama guru	
Aksi		**Reaksi Sistem**	
1	Memilih menu hapus	2	Menampilkan pesan konfirmasi hapus
		3	Konfirmasi OK, materi pelajaran dihapus dari database
			Konfirmasi Batal, materi pelajaran batal dihapus
Kondisi Akhir		Guru berhasil menghapus materi pelajaran	

9. Interaksi antara aktor admin dengan *use case* kelola mata pelajaran dibagi menjadi tiga proses yaitu proses tambah, ubah, dan hapus mata pelajaran

a. Interaksi antara aktor admin dengan *use case* kelola mata pelajaran pada proses tambah mata pelajaran dapat dilihat pada Tabel 3.19 di bawah ini.

Tabel 3.19 ***Sekenario Use Case Tambah Mata Pelajaran***

		Identifikasi	
Nama		Tambah data mata pelajaran	
Tujuan		Menambah data mata pelajaran	
Deskripsi		Proses untuk tambah data mata pelajaran	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan menu utama admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu mata pelajaran	2	Menampilkan data mata pelajaran
3	Memilih menu tambah	4	Menampilkan form tambah data mata pelajaran
5	Mengisi data mata pelajaran	6	
7	Memilih tombol simpan	8	Mengecek validasi data materi
		9	Bila data valid maka data materi disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data Materi Pelajaran tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
10	Mengisi kembali data materi yang baru		
11	Memilih tombol simpan	12	Mengecek validasi data materi
		13	Bila data valid maka data materi disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Guru berhasil menambah materi pelajaran	

b. Interaksi antara aktor admin dengan *use case* kelola mata pelajaran pada proses ubah mata pelajaran dapat dilihat pada Tabel 3.20 di bawah ini.

Tabel 3.20 ***Sekenario Use Case Ubah Mata Pelajaran***

		Identifikasi	
Nama		Ubah data mata pelajaran	
Tujuan		Mengubah data mata pelajaran	
Deskripsi		Proses untuk ubah data mata pelajaran	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan menu utama admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu mata pelajaran	2	Menampilkan data mata pelajaran
3	Memilih menu ubah	4	Menampilkan form ubah data mata pelajaran
5	Mengisi data mata pelajaran yang baru		
6	Memilih tombol simpan	7	Mengecek validasi data mata pelajaran
		8	Bila data valid maka data mata pelajaran disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data mata pelajaran tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data mata pelajaran yang baru		
10	Memilih tombol simpan	11	Mengecek validasi data mata pelajaran
		12	Bila data valid maka data mata pelajaran disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil mengubah data mata pelajaran	

c. Interaksi antara aktor admin dengan *use case* kelola mata pelajaran pada proses hapus mata pelajaran dapat dilihat pada Tabel 3.21 di bawah ini

Tabel 3.21 ***Sekenario Use Case Hapus Mata Pelajaran***

Identifikasi			
Nama		Hapus data mata pelajaran	
Tujuan		Menghapus data mata pelajaran	
Deskripsi		Proses untuk hapus data mata pelajaran	
Aktor		Admin	
Use case yang berkaitan		-	
		Sekenario yang berkaitan	
Kondisi Awal		Menampilkan halaman mata pelajaran	
Aksi		**Reaksi Sistem**	
1	Memilih menu mata pelajaran	2	Menampilkan data mata pelajaran
3	Memilih hapus mata pelajaran	4	Menampilkan form konfirmasi hapus
		5	Konfirmasi Ok maka data mata pelajaran dihapus dari database
			Konfirmasi Batal data mata pelajaran batal dihapus
Kondisi Akhir		Admin berhasil menghapus data mata pelajaran	

10. Interaksi antara aktor guru dengan *use case* pengumuman dibagi menjadi tiga proses yaitu proses tambah, ubah, dan hapus pengumuman.
    a. Interaksi antara aktor guru dengan *use case* kelola pengumuman pada proses tambah pengumuman dapat dilihat pada Tabel 3.22 di bawah ini.

Tabel 3.22 ***Sekenario Use Case Tambah Pengumuman***

Identifikasi			
Nama		Tambah data pengumuman	
Tujuan		Menambah data pengumuman	
Deskripsi		Proses untuk tambah data pengumuman	
Aktor		Guru	
Use case yang berkaitan			-
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama guru	
Aksi		**Reaksi Sistem**	
1	Memilih menu pengumuman	2	Menampilkan data pengumuman
3	Memilih menu tambah	4	Menampilkan form tambah data pengumuman
5	Mengisi data		

	pengumuman		
6	Memilih tombol simpan	7	Mengecek validasi data pengumuman
		8	Bila data valid maka data pengumuman disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data mata pelajaran tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data pengumuman		
10	Memilih tombol simpan	11	Mengecek validasi data pengumuman
		12	Bila data valid maka data pengumuman disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Guru berhasil menambah data pengumuman	

b. Interaksi antara aktor guru dengan *use case* kelola pengumuman pada proses ubah pengumuman dapat dilihat pada Tabel 3.23 di bawah ini.

Tabel 3.23 ***Sekenario Use Case Ubah Pengumuman***

Identifikasi			
Nama	Ubah data pengumuman		
Tujuan	Mengubah data pengumuman		
Deskripsi	Proses untuk ubah data pengumuman		
Aktor	Guru		
Use case yang berkaitan	-		
Sekenario yang berkaitan			
Kondisi Awal	Menampilkan menu utama guru		
Aksi		**Reaksi Sistem**	
1	Memilih menu pengumuman	2	Menampilkan data pengumuman
3	Memilih menu ubah	4	Menampilkan form ubah data pengumuman
5	Mengisi data pengumuman yang baru		
6	Memilih tombol	7	Mengecek validasi data pengumuman

	simpan		
		8	Bila data valid maka data pengumuman disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data mata pelajaran tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data pengumuman yang baru		
10	Memilih tombol simpan	11	Mengecek validasi data pengumuman
		12	Bila data valid maka data pengumuman disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Guru berhasil mengubah data pengumuman	

c. Interaksi antara aktor guru dengan *use case* kelola pengumuman pada proses hapus pengumuman dapat dilihat pada Tabel 3.24 di bawah ini

Tabel 3.24 ***Sekenario Use Case Hapus Pengumuman***

Identifikasi			
Nama		Hapus data pengumuman	
Tujuan		Menghapus data pengumuman	
Deskripsi		Proses untuk hapus data pengumuman	
Aktor		Guru	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan halaman pengumuman	
Aksi		**Reaksi Sistem**	
1	Memilih menu pengumuman	2	Menampilkan data pengumuman
3	Memilih hapus pengumuman	4	Menampilkan form konfirmasi hapus
		5	Konfirmasi Ok maka data pengumuman dihapus dari database
			Konfirmasi Batal data pengumuman batal dihapus
Kondisi Akhir		Guru berhasil menghapus data pengumuman	

11. Interaksi antara aktor guru dengan *use case* kelola tugas dapat dilihat pada Tabel 3.25 di bawah ini.

Tabel 3.25 ***Sekenario Use Case Kelola Tugas***

Identifikasi			
Nama		Kelola data tugas	
Tujuan		Menambah data tugas	
Deskripsi		Proses untuk tambah data tugas	
Aktor		Guru	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama guru	
Aksi		**Reaksi Sistem**	
1	Memilih menu tugas	2	Menampilkan data tugas
3	Memilih menu tambah	4	Menampilkan form tambah data tugas
5	Mengisi data tugas		
6	Memilih tombol simpan	7	Mengecek validasi data pengumuman
			Bila data valid maka data pengumuman disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data tugas tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
8	Mengisi kembali data tugas		
9	Memilih tombol simpan	10	Mengecek validasi data tugas
		11	Bila data valid maka data pengumuman disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Guru berhasil menambah data pengumuman	

12. Interaksi antara aktor guru, siswa dengan *use case* kelola forum komunikasi dapat dilihat pada Tabel 3.26 di bawah ini.

Tabel 3.26 ***Sekenario Use Case Kelola Forum Komunikasi***

Identifikasi	
Nama	Kelola forum komunikasi
Tujuan	Mengolah forum komunikasi
Deskripsi	Proses untuk tambah, hapus, ubah forum komunikasi

Aktor		Guru, siswa	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama berdasarkan hak akses	
Aksi			**Reaksi Sistem**
1	Memilih menu forum komunikasi	2	Menampilkan topik berdasarkan waktu
3	Memilih tambah topik	4	Menampilkan form tambah topik baru
5	Mengisi topik baru	6	Menyimpan data pada database
Kondisi Akhir		Guru dan siswa berhasil mengolah forum	

13. Interaksi antara aktor siswa dengan *use case* lihat pengumuman dapat dilihat pada Tabel 3.27 di bawah ini.

Tabel 3.27 ***Sekenario Use Case Lihat Pengumuman***

Identifikasi			
Nama		Lihat pengumuman	
Tujuan		Menampilkan pengumuman	
Deskripsi		Proses untuk menampilkan pengumuman yang terlah dibuat oleh guru	
Aktor		Siswa	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama siswa	
Aksi			**Reaksi Sistem**
1	Memilih menu pengumuman	2	Mengambil data pengumuman dari database dan menampilkan data pengumuman
Kondisi Akhir		Halaman lihat pengumuman berhasil ditampilkan	

14. Interaksi antara aktor siswa dengan *use case download* materi pelajaran dapat dilihat pada Tabel 3.28 di bawah ini.

Tabel 3.28 ***Sekenario Use Case Download Materi Pelajaran***

Identifikasi			
Nama		*Download* materi pelajaran	
Tujuan		Menyimpan *file* materi pelajaran	
Deskripsi		Proses untuk menyimpan file materi pelajaran yang telah di *upload* oleh guru	
Aktor		Siswa	
Use case yang berkaitan		-	
Sekenario yang berkaitan			

Kondisi Awal		Menampilkan menu utama siswa	
Aksi		**Reaksi Sistem**	
1	Memilih menu mata pelajaran	2	Mengambil data mata pelajaran dari database dan menampilkan data mata pelajaran
3	Memilih menu *download*	4	Menyimpan *file* yang dipilih oleh siswa
Kondisi Akhir		Siswa berhasil menyimpan *file*	

15. Interaksi antara aktor siswa dengan *use case* latihan dapat dilihat pada Tabel 3.29 di bawah ini.

Tabel 3.29 ***Sekenario Use Case Latihan***

Identifikasi			
Nama		Latihan	
Tujuan		Menampilkan latihan yang sedang berlangsung	
Deskripsi		Proses untuk melaksanakan latihan soal	
Aktor		Siswa	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama siswa	
Aksi		**Reaksi Sistem**	
1	Memilih menu latihan	2	Menampilkan data latihan yang belum dikerjakan
3	Memilih menu mulai latihan	4	Menampilkan soal latihan beserta waktu penghitungan latihan
		5	Sistem tidak bisa di akses lagi saat waktu latihan berakhir dan kembali ke halaman latihan
Kondisi Akhir		Siswa berhasil melakukan latihan	

16. Interaksi antara aktor siswa dengan *use case download* tugas dapat dilihat pada Tabel 3.30 di bawah ini.

Tabel 3.30 ***Sekenario Use Case Download Tugas***

Identifikasi	
Nama	*Download* tugas
Tujuan	Menyimpan tugas pelajaran
Deskripsi	Proses menyimpan tugas pelajaran
	yang telah di *upload* oleh guru
Aktor	Siswa
Use case yang berkaitan	-
Sekenario yang berkaitan	

Kondisi Awal		Menampilkan menu utama siswa	
Aksi		**Reaksi Sistem**	
1	Memilih menu tugas	2	Menampilkan data tugas berdasarkan mata pelajaran
3	Memilih menu *download* tugas	4	Menyimpan tugas yang dipilih oleh siswa
Kondisi Akhir		Siswa berhasil menyimpan tugas	

17. Interaksi antara aktor siswa dengan *use case upload* tugas dapat dilihat pada Tabel 3.31dibawah ini.

Tabel 3.31 ***Sekenario Use Case Upload Tugas***

Identifikasi			
Nama		*Upload* tugas	
Tujuan		*Upload* tugas pelajaran	
Deskripsi		Proses untuk *upload* tugas pelajaran ke sistem	
Aktor		Siswa	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama siswa	
Aksi		**Reaksi Sistem**	
1	Memilih menu tugas	2	Menampilkan data tugas berdasarkan mata pelajaran
3	Memilih menu *upload* tugas	4	Menyimpan tugas yang dipilih oleh siswa ke sistem
Kondisi Akhir		Siswa berhasil *upload* tugas	

18. Interaksi antara aktor admin dengan *use case* kelola bantuan dibagi menjadi tiga proses yaitu tambah, ubah, hapus bantuan.
    a. Interaksi antara aktor admin dengan *use case* kelola bantuan pada proses tambah bantuan dapat dilihat pada Tabel 3.32 di bawah ini.

Tabel 3.32 ***Sekenario Use Case Tambah Bantuan***

Identifikasi			
Nama		Tambah data bantuan	
Tujuan		Menambah data pengumuman	
Deskripsi		Proses untuk tambah data pengumuman	
Aktor		Admin	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama admin	
Aksi		**Reaksi Sistem**	
1	Memilih menu bantuan	2	Menampilkan data bantuan

3	Memilih menu tambah	4	Menampilkan form tambah data bantuan
5	Mengisi data bantuan		
6	Memilih tombol simpan	7	Mengecek validasi data bantuan
		8	Bila data valid maka data bantuan disimpan pada database dan menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data mata pelajaran tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
9	Mengisi kembali data bantuan		
10	Memilih tombol simpan	11	Mengecek validasi data bantuan
		12	Bila data valid maka data bantuan disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil menambah data bantuan	

b. Interaksi antara aktor admin dengan *use case* kelola bantuan pada proses ubah bantuan dapat dilihat pada Tabel 3.33 di bawah ini.

Tabel 3.33 ***Sekenario Use Case Ubah Bantuan***

Identifikasi			
Nama		Ubah data bantuan	
Tujuan		Mengubah data bantuan	
Deskripsi		Proses untuk ubah data bantuan	
Aktor		Admin	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan menu utama guru	
Aksi		**Reaksi Sistem**	
1	Memilih menu bantuan	3	Menampilkan data bantuan
4	Memilih menu ubah	5	Menampilkan form ubah data bantuan
6	Mengisi data bantuan yang baru		
7	Memilih tombol simpan	8	Mengecek validasi data bantuan
		9	Bila data valid maka data bantuan disimpan pada database dan

			menampilkan pesan data berhasil disimpan
Sekenario Alternatif (Data mata pelajaran tidak valid)			
Aksi		**Reaksi Sistem**	
			Menampilkan pesan error data tidak valid
10	Mengisi kembali data bantuan yang baru		
11	Memilih tombol simpan	12	Mengecek validasi data bantuan
		13	Bila data valid maka data bantuan disimpan pada database dan menampilkan pesan data berhasil disimpan
Kondisi Akhir		Admin berhasil mengubah data bantuan	

c. Interaksi antara aktor admin dengan *use case* kelola bantuan pada proses hapus bantuan dapat dilihat pada Tabel 3.34 di bawah ini

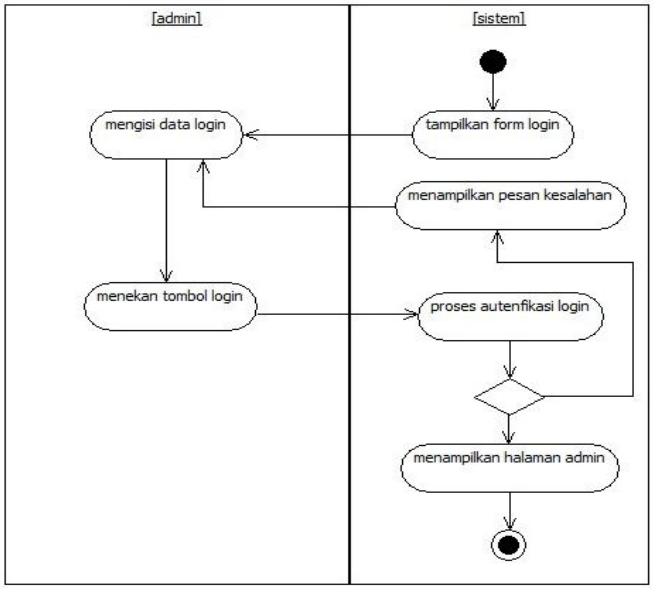
Tabel 3.34 ***Sekenario Use Case Hapus Bantuan***

Identifikasi			
Nama		Hapus data bantuan	
Tujuan		Menghapus data bantuan	
Deskripsi		Proses untuk hapus data bantuan	
Aktor		Admin	
Use case yang berkaitan		-	
Sekenario yang berkaitan			
Kondisi Awal		Menampilkan halaman bantuan	
Aksi		**Reaksi Sistem**	
1	Memilih menu bantuan	2	Menampilkan data bantuan
3	Memilih hapus bantuan	4	Menampilkan form konfirmasi hapus
		5	Konfirmasi Ok maka data bantuan dihapus dari database
			Konfirmasi Batal data bantuan batal dihapus
Kondisi Akhir		Bantuan berhasil menghapus data bantuan	

3.2.5.3 Activity Diagram

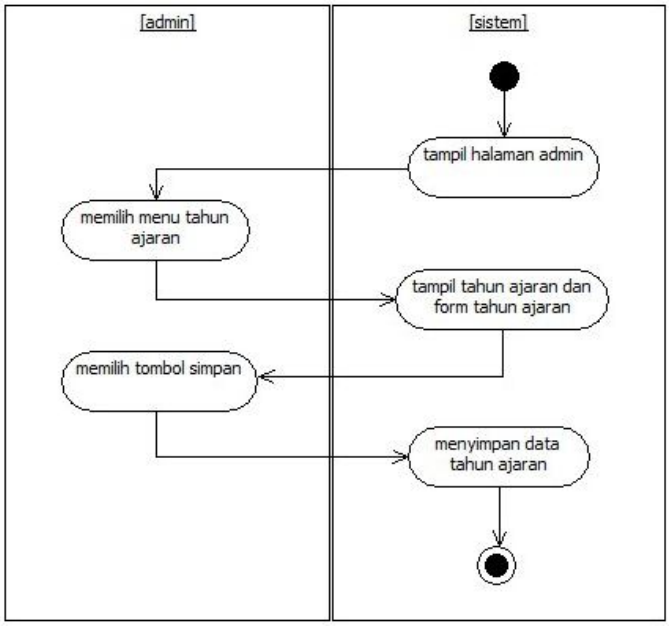
Activity diagram merupakan diagram yang memodelkan aliran kerja atau workflow dari urutan aktifitas dalam suatu proses yang mengacu pada use case diagram yang ada, berikut ini penjelasan dari masing masing activity diagram

1. Activity diagram login admin



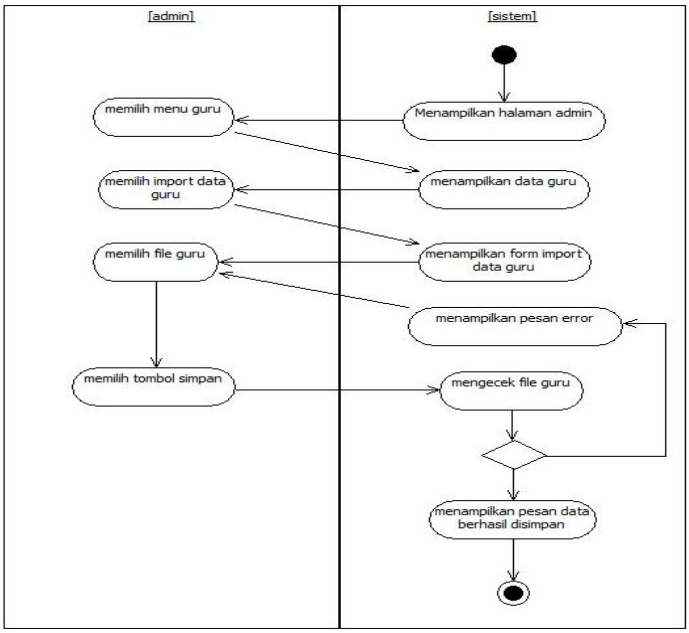
Gambar 3.8 ***Activity Diagram Login Admin***

2. Activity diagram pengolahan tahun ajaran



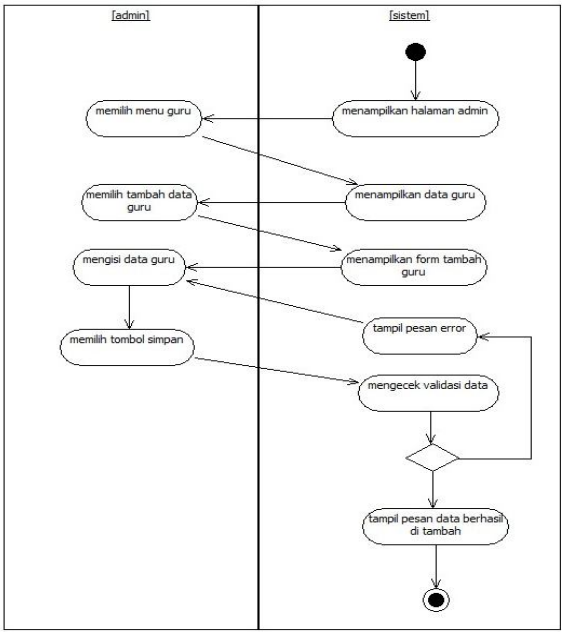
Gambar 3.9 ***Activity Diagram Pengolahan Tahun Ajaran***

3. Activity diagram pengolahan data guru dibagi menjadi empat yaitu import data, tambah data, ubah data, hapus data.
   a. Activity diagram pengolahan data guru pada proses import data



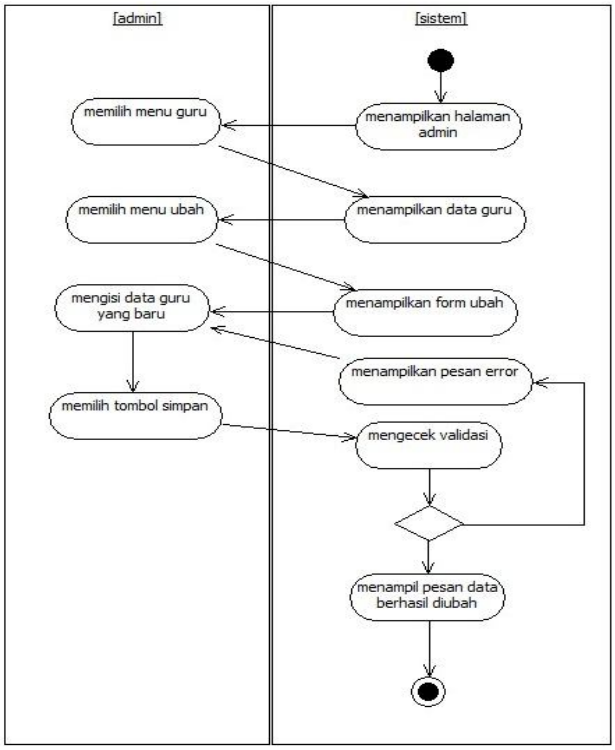
Gambar 3.10 ***Activity Diagram Import Data Guru***

   b. Activity diagram pengolahan data guru pada proses tambah data



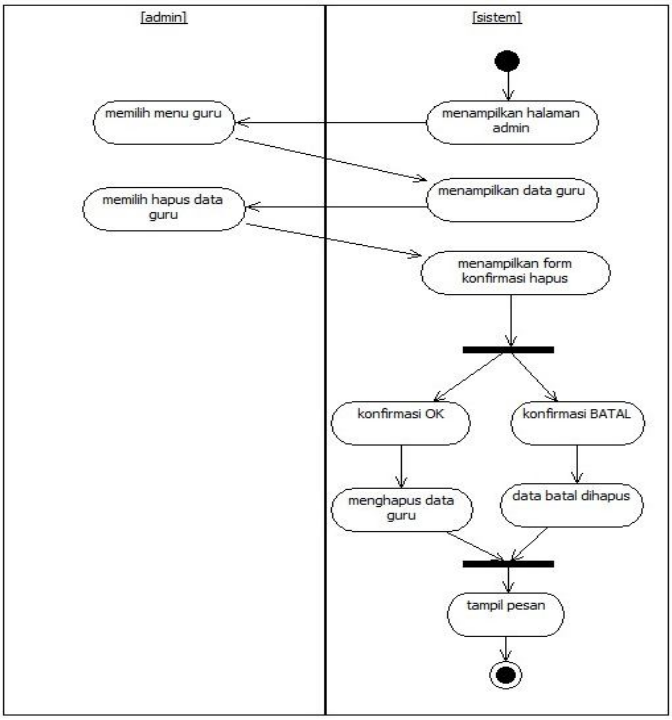
Gambar 3.11 ***Activity Diagram Tambah Data Guru***

c. Activity diagram pengolahan data guru pada proses ubah data



Gambar 3.12 ***Activity Diagram Ubah Data Guru***

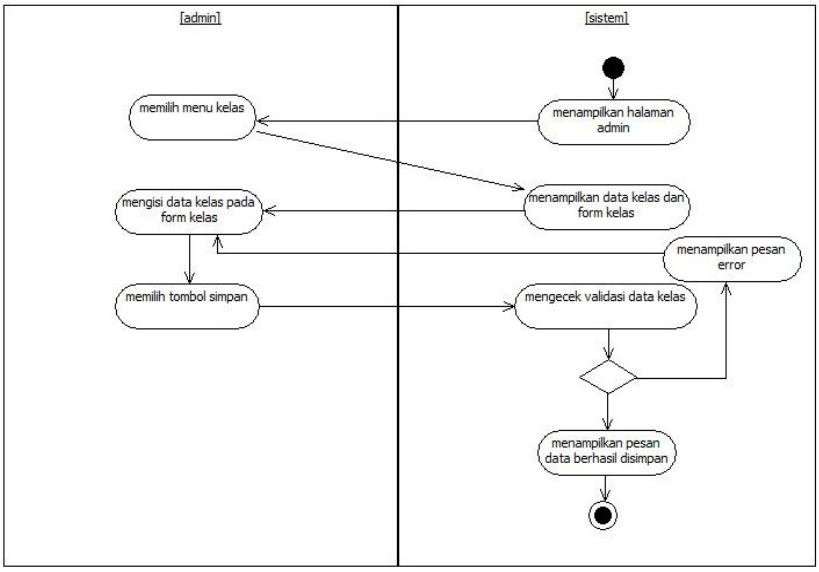
d. Activity diagram pengolahan data guru pada proses hapus data



Gambar 3.13 ***Activity Diagram Hapus Data Guru***

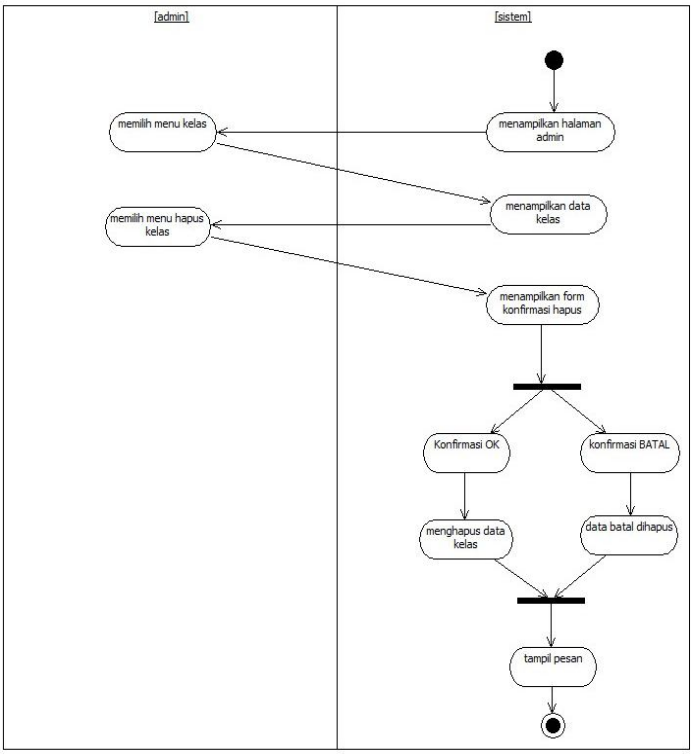
4. Activity diagram pengolahan kelas dibagi menjadi dua proses yaitu tambah data dan hapus data

   a. Activity diagram pengolahan data kelas pada proses tambah data



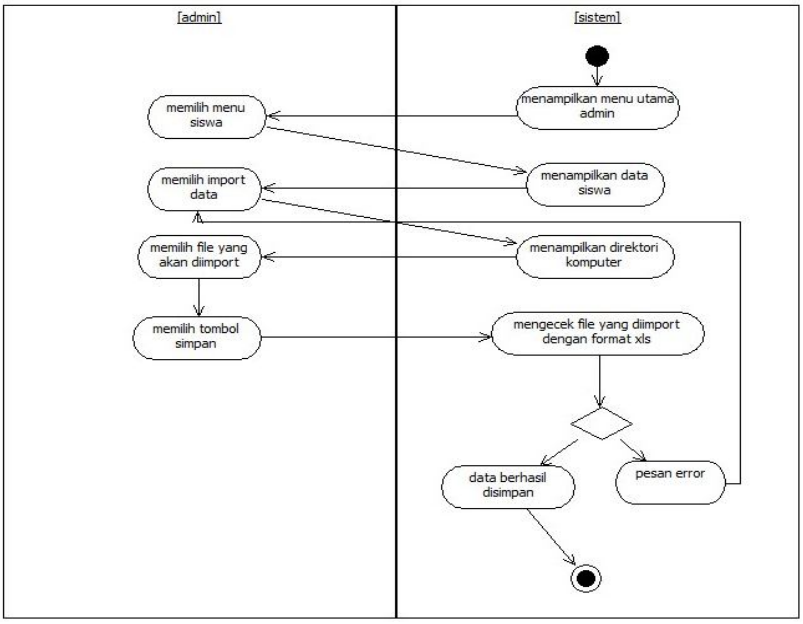
Gambar 3.14 ***Activity Diagram Tambah Data Kelas***

   b. Activity diagram pengolahan data kelas pada proses hapus data



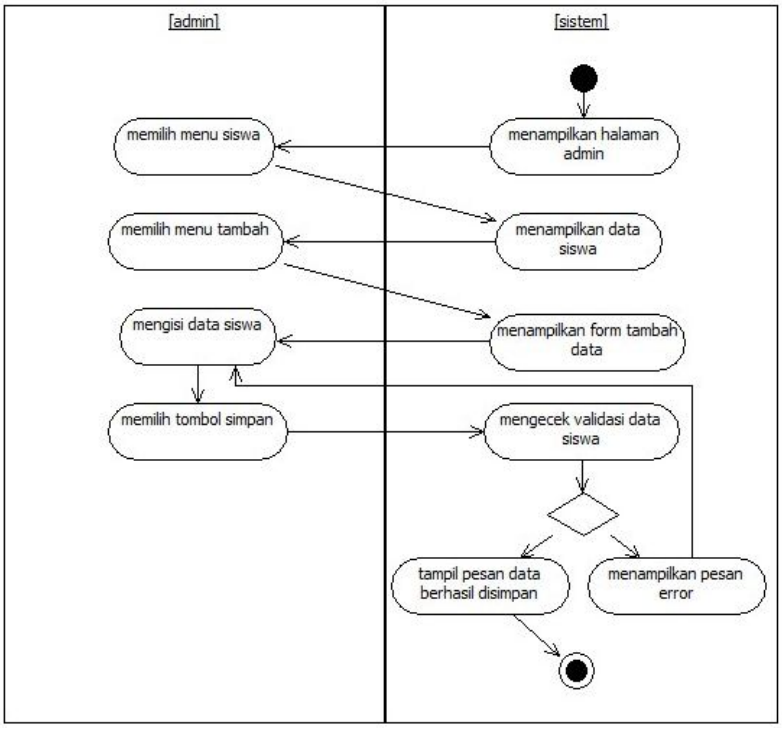
Gambar 3.15 ***Activity Diagram Hapus Data Kelas***

5. Activity diagram pengolahan data siswa dibagi menjadi empat proses yaitu import data, tambah data, ubah, data, dan hapus data
   a. Activity diagram pengolahan data siswa pada proses import data



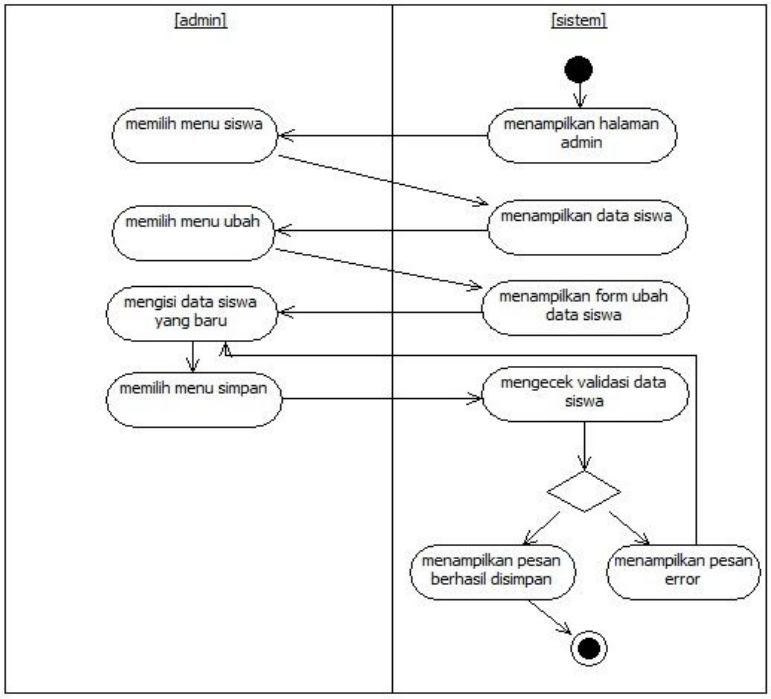
Gambar 3.16 ***Activity Diagram Import Data Siswa***

   b. Activity diagram pengolahan data siswa pada proses tambah data



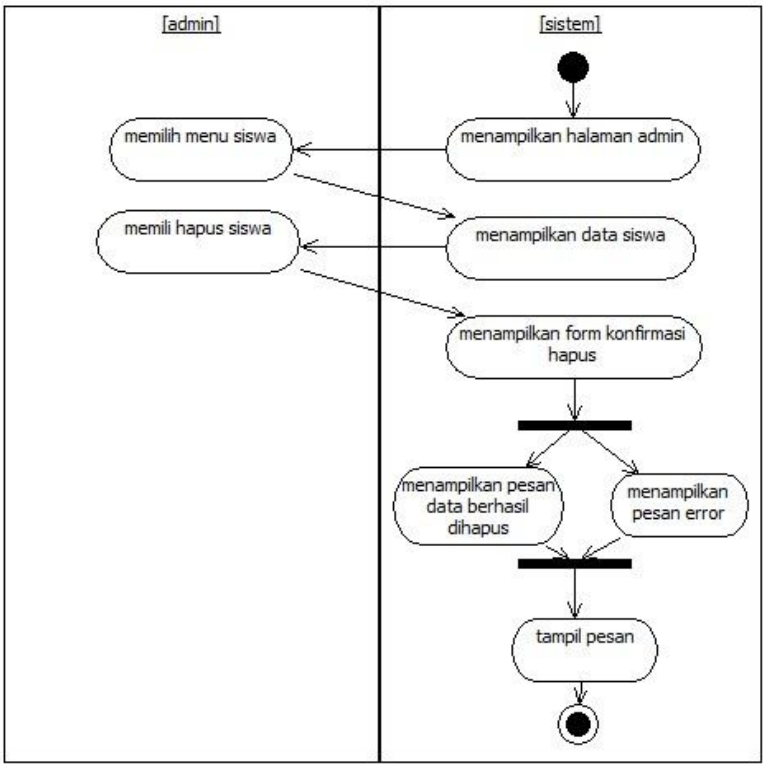
Gambar 3.17 ***Activity Diagram Tambah Data Siswa***

c. Activity diagram pengolahan data siswa pada proses ubah data



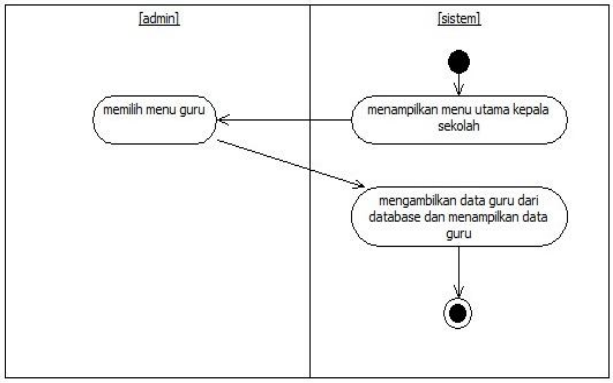
Gambar 3.18 ***Activity Diagram Ubah Data Siswa***

d. Activity diagram pengolahan data siswa pada proses hapus data



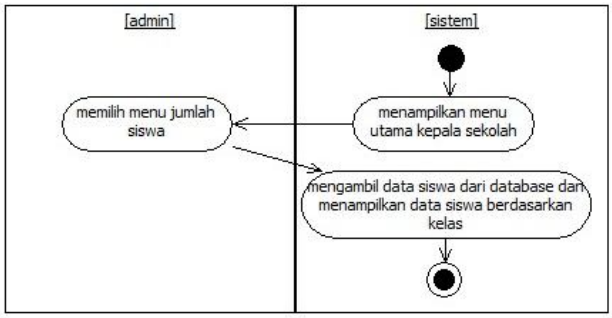
Gambar 3.19 ***Activity Diagram Hapus Data Siswa***

6. Activity diagram lihat data guru



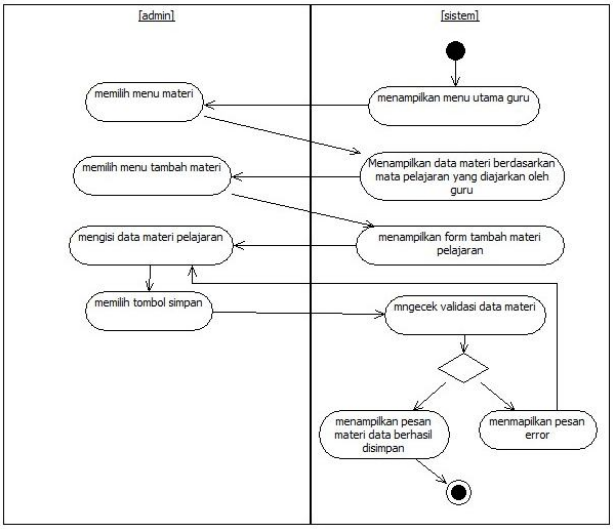
Gambar 3.20 ***Activity Diagram Lihat Data Guru***

7. Activity diagram lihat data siswa



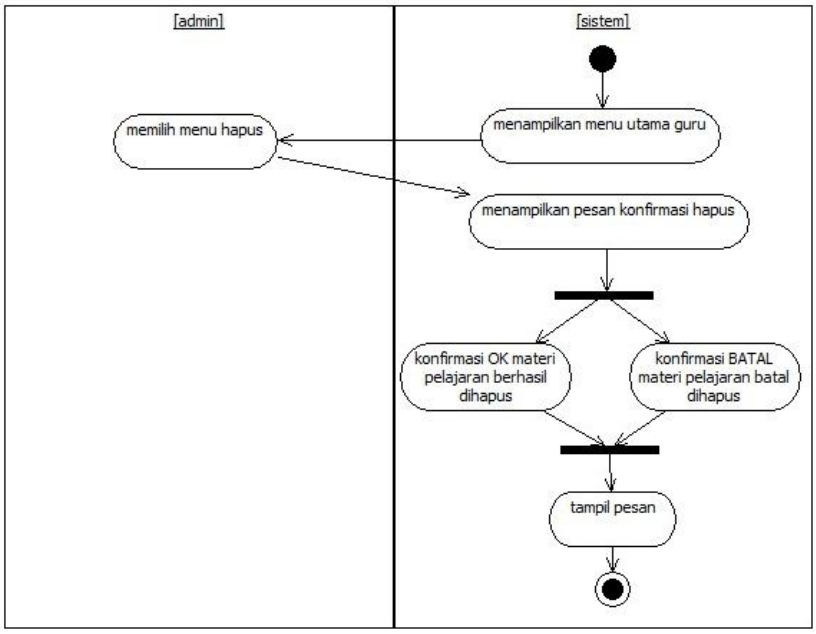
Gambar 3.21 ***Activity Diagram Lihat Data Siswa***

8. Activity diagram pengolahan materi pelajaran dibagi menjadi dua proses
   a. Activity diagram pengolahan materi pelajaran pada proses tambah data



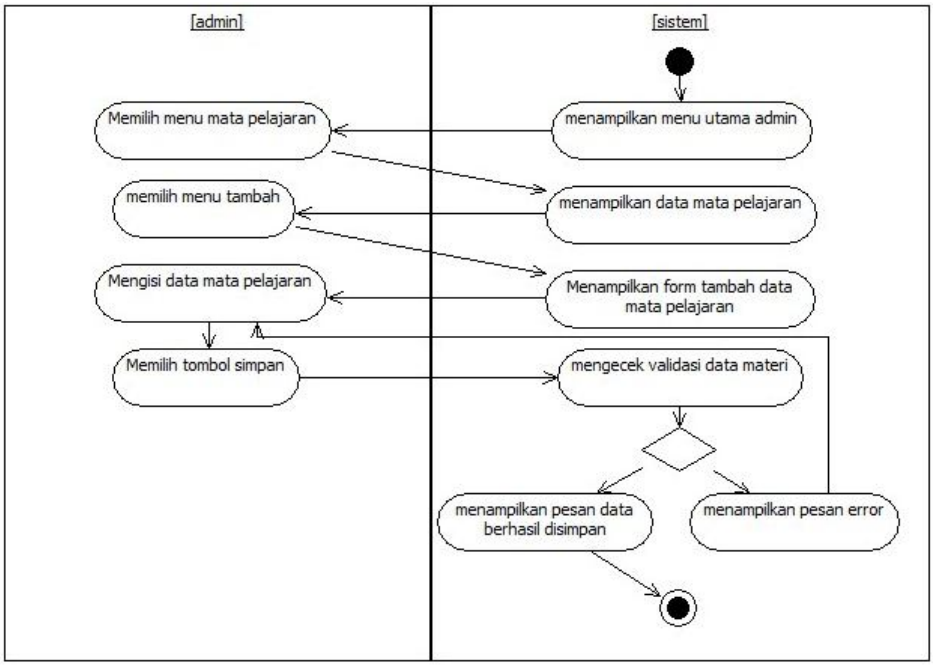
Gambar 3.22 ***Activity Diagram Tambah Materi Pelajaran***

   b. Activity diagram pengolahan materi pelajaran pada proses hapus data



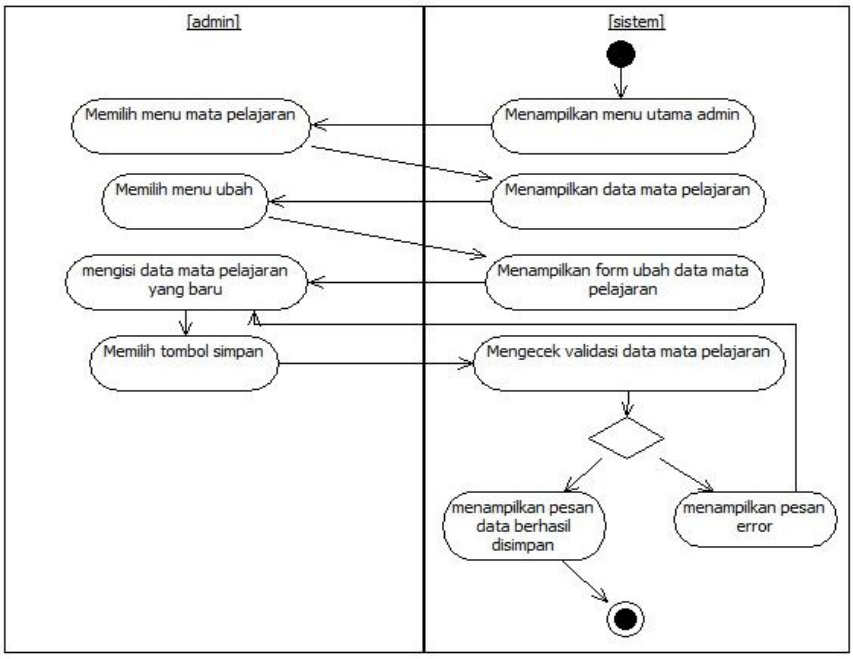
Gambar 3.23 ***Activity Diagram Hapus Materi Pelajaran***

9. Activity diagram pengolahan mata pelajaran dibagi menjadi tiga proses yaitu proses tambah data, ubah data, hapus data
   a. Activity diagram pengolahan data mata pelajaran pada proses tambah data



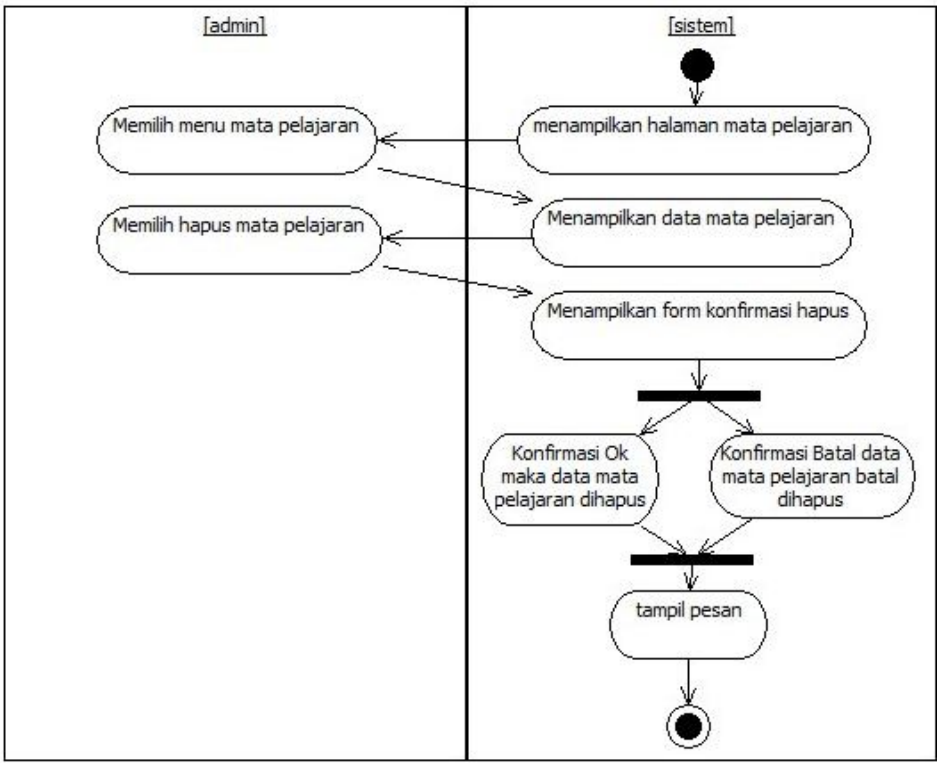
Gambar 3.24 ***Activity Diagram Tambah Mata Pelajaran***

b. Activity diagram pengolahan data mata pelajaran pada proses ubah data



Gambar 3.25 ***Activity Diagram Ubah Mata Pelajaran***

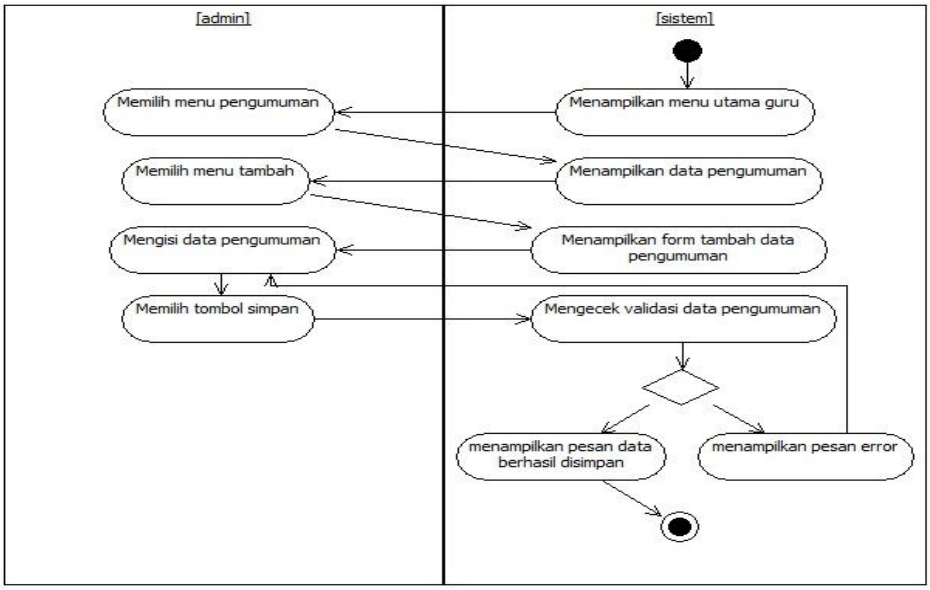
c. Activity diagram pengolahan data mata pelajaran pada proses hapus data



Gambar 3.26 ***Activity Diagram Hapus Mata Pelajaran***

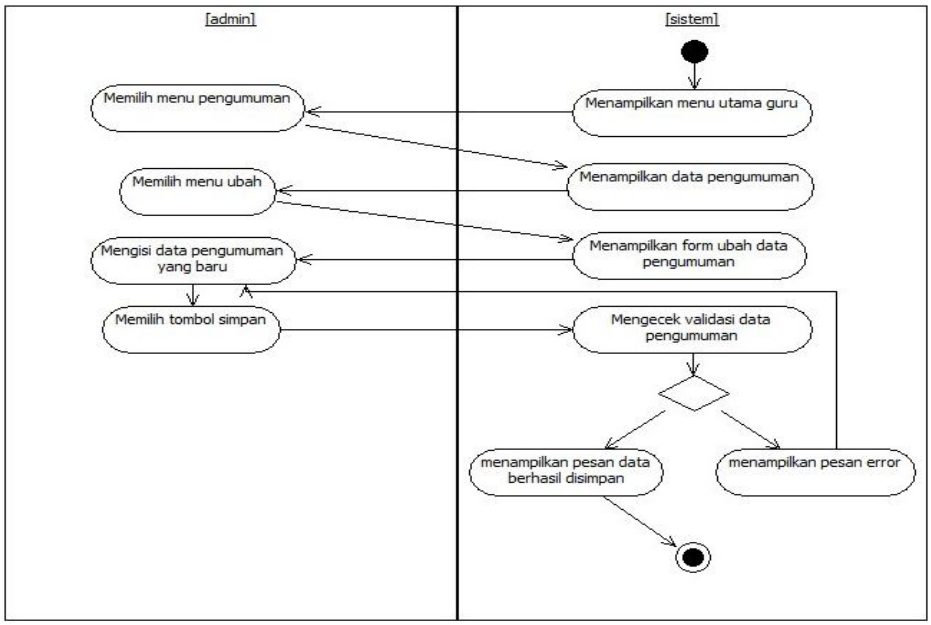
10. Activity diagram pengolahan data pengumuman dibagi menjadi tiga proses yaitu proses tambah data, ubah data, hapus data

    a. Activity diagram pengolahan data pengumuman pada proses tambah data



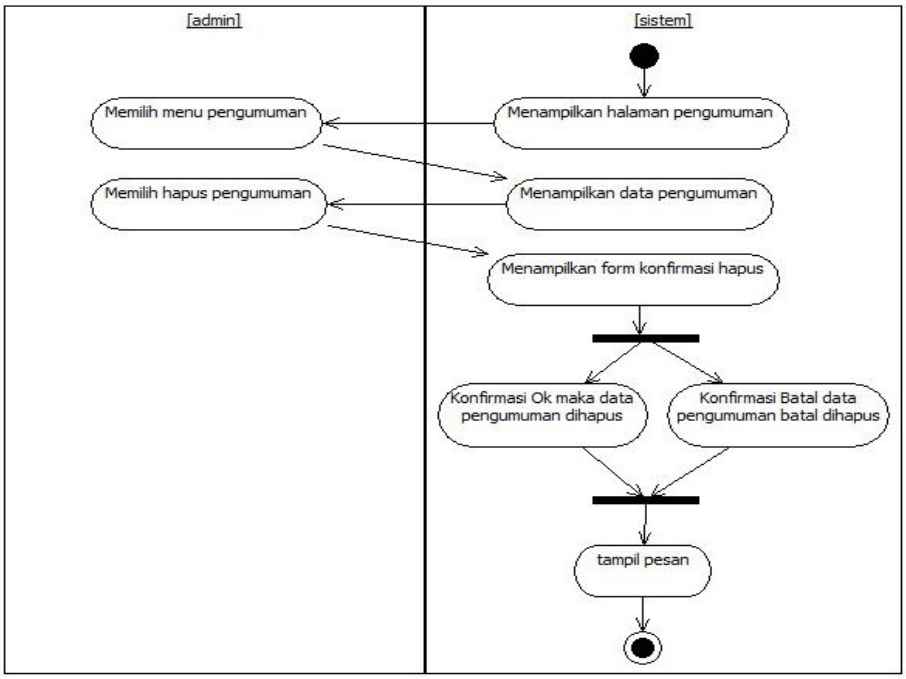
Gambar 3.27 ***Activity Diagram Tambah Data Pengumuman***

    b. Activity diagram pengolahan data pengumuman pada proses ubah data



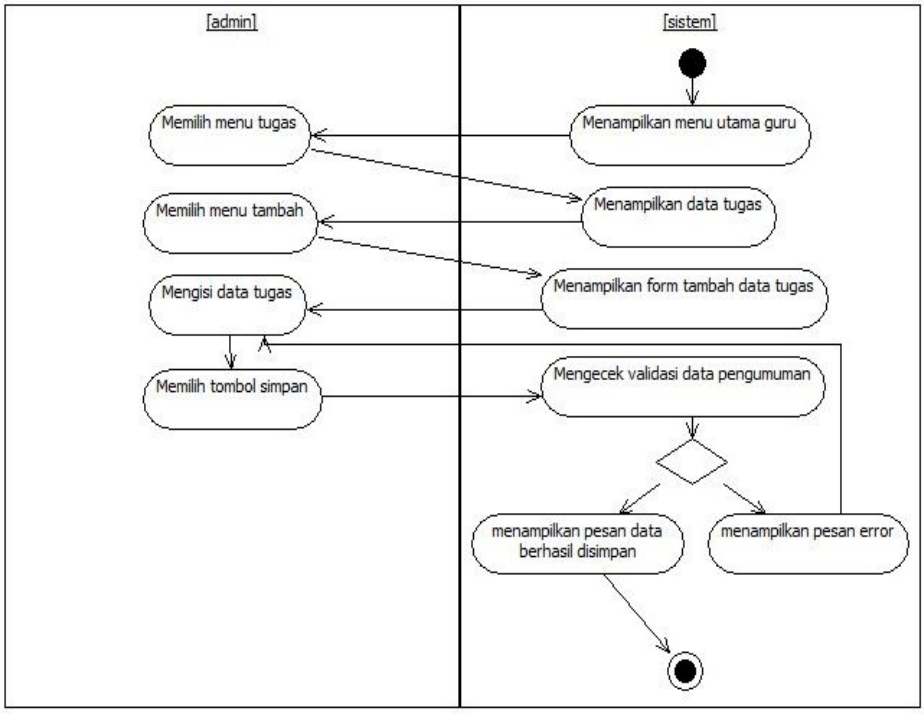
Gambar 3.28 ***Activity Diagram Ubah Data Pengumuman***

c. Activity diagram pengolahan data pengumuman pada proses hapus data



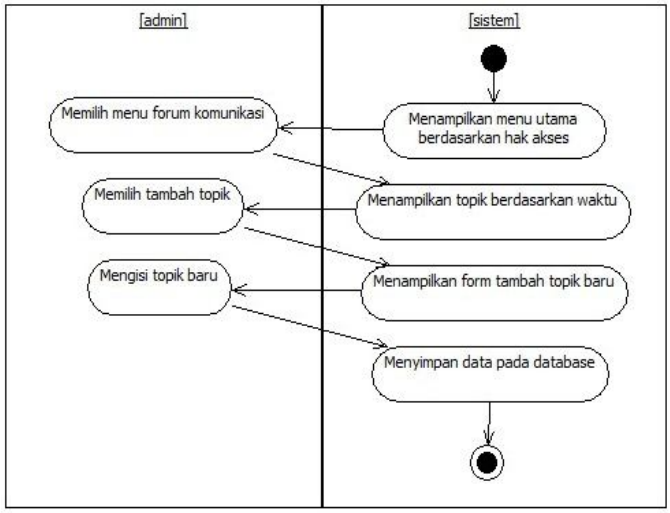
Gambar 3.29 ***Activity Diagram Hapus Data Pengumuman***

11. Activity diagram pengolahan data tugas



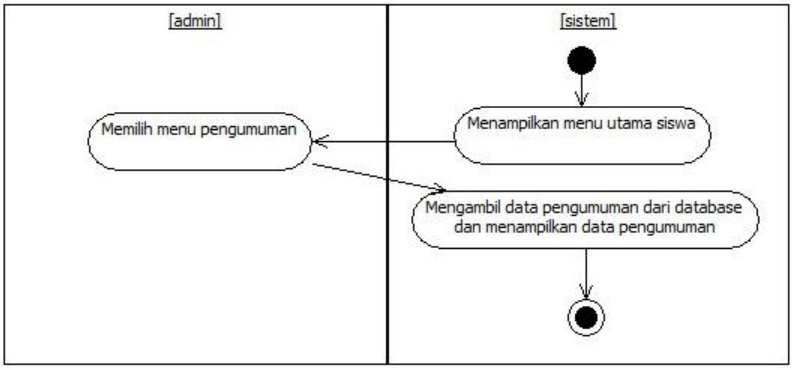
Gambar 3.30 ***Activity Diagram Data Tugas***

12. Activity diagram pengolahan data forum



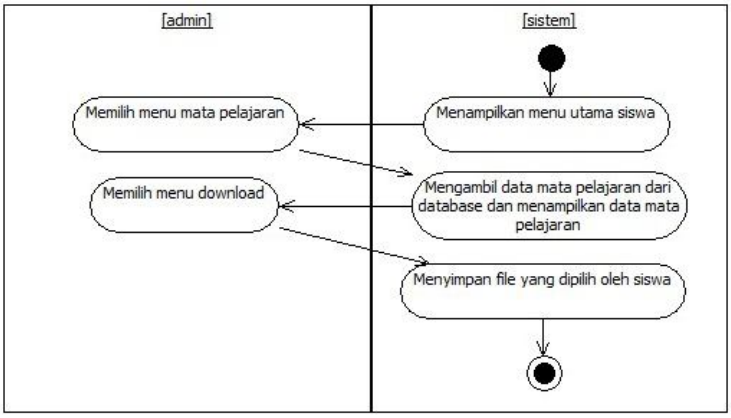
Gambar 3.31 ***Activity Diagram Pengolahan data Forum***

13. Activity diagram pengolahan lihat pengumuman



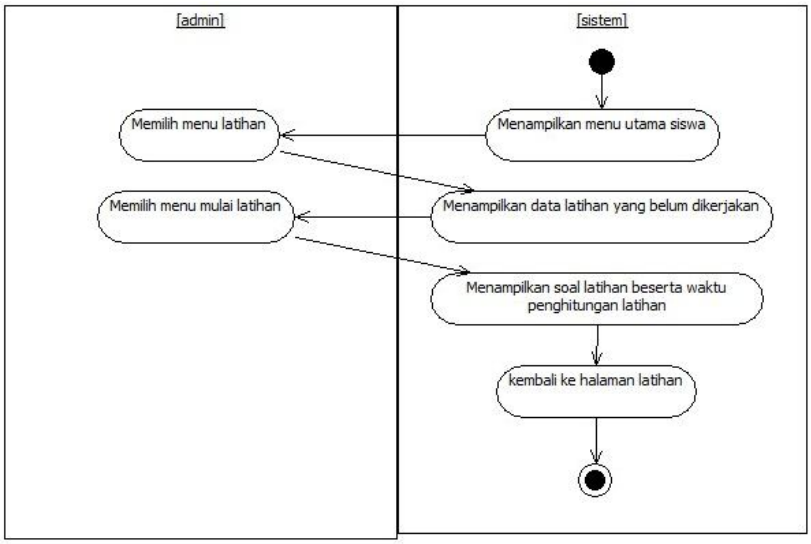
Gambar 3.32 ***Activity Diagram Lihat Pengumuman***

14. Activity diagram pengolahan download materi



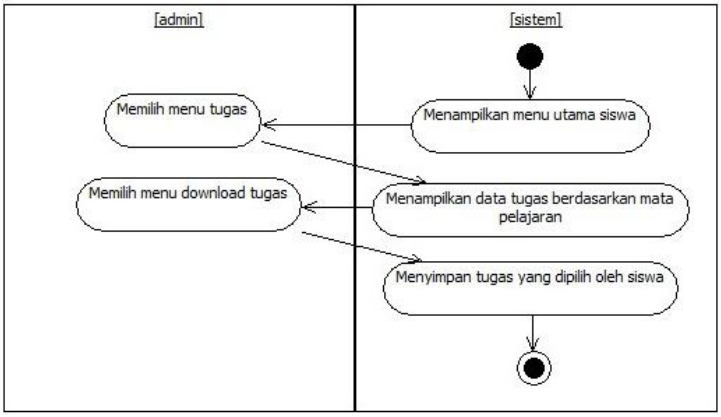
Gambar 3.33 ***Activity Diagram Download Materi***

15. Activity diagram pengolahan latihan



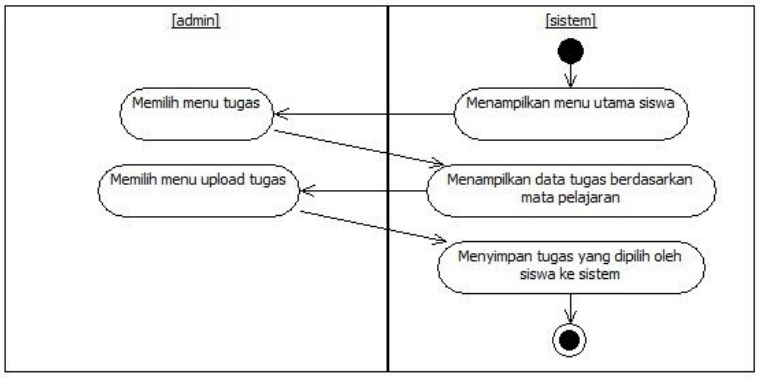
Gambar 3.34 ***Activity Diagram Kelola Latihan***

16. Activity diagram pengolahan download tugas



Gambar 3.35 ***Activity Diagram Download Tugas***

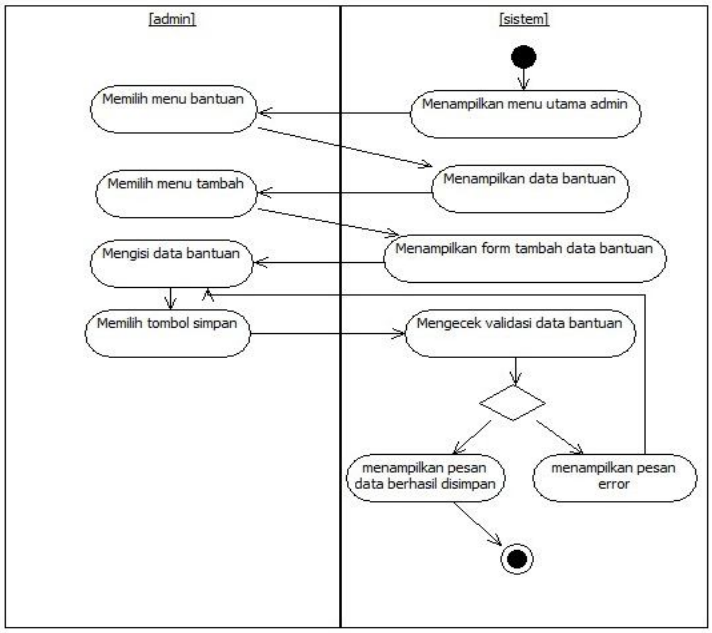
17. Activity diagram pengolahan upload tugas



Gambar 3.36 ***Activity Diagram Upload Tugas***

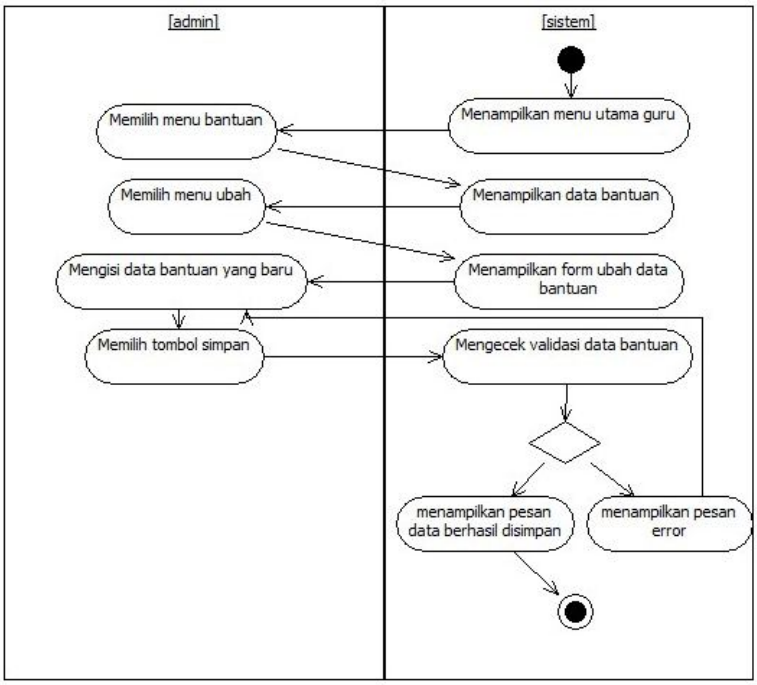
18. Activity diagram pengolahan data bantuan dibagi menjadi tiga proses yaitu tambah data, ubah data, dan hapus data

a. Activity diagram pengolahan data bantuan pada proses tambah data



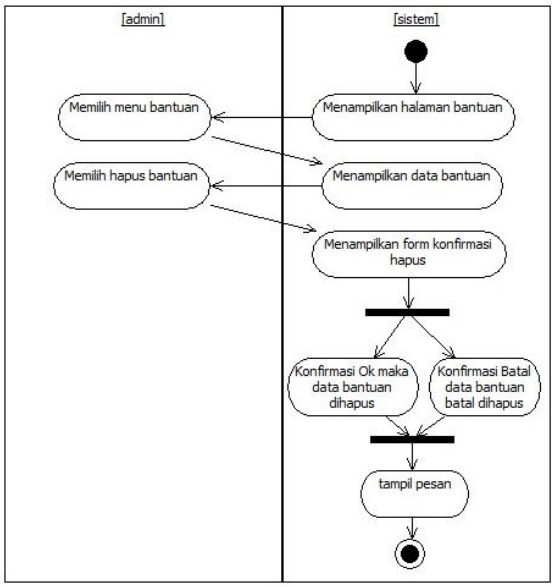
Gambar 3.37 ***Activity Diagram Tambah Data Bantuan***

b. Activity diagram pengolahan data bantuan pada proses tambah data



Gambar 3.38 ***Activity Diagram Ubah data Bantuan***

c. Activity diagram pengolahan data bantuan pada proses tambah data

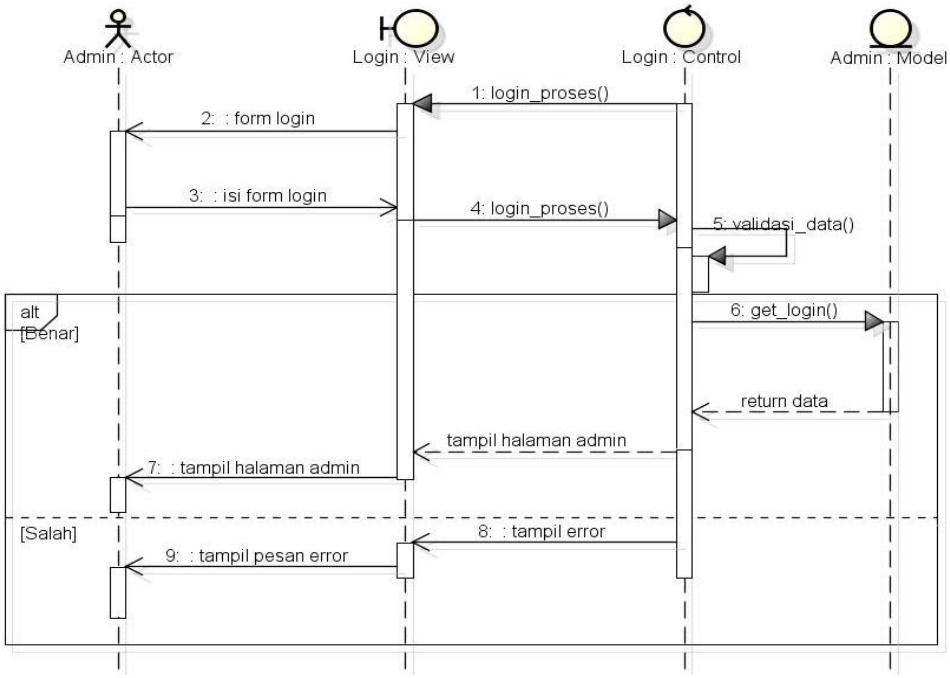


Gambar 3.39 ***Activity Diagram Hapus Data Bantuan***

3.2.5.4 *Sequenc*e Diagram

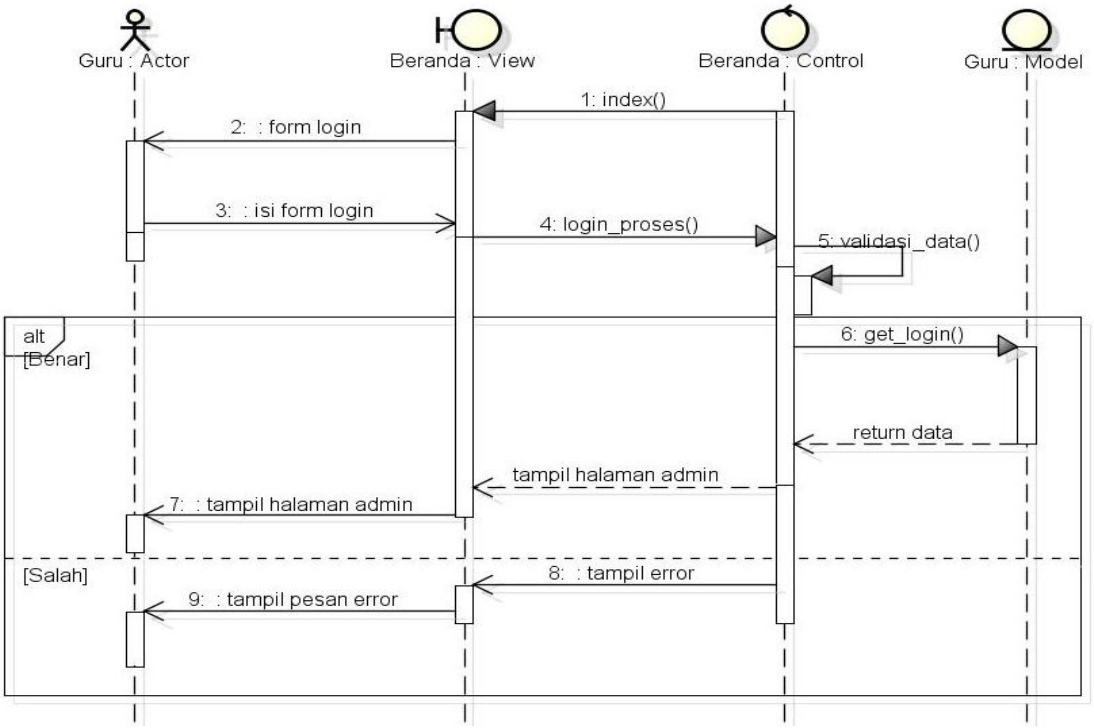
Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem berupa pesan (*message*) yang digambarkan terhadap waktu.

1. *Sequence* diagram *login* admin dapat dilihat pada Gambar 3.40



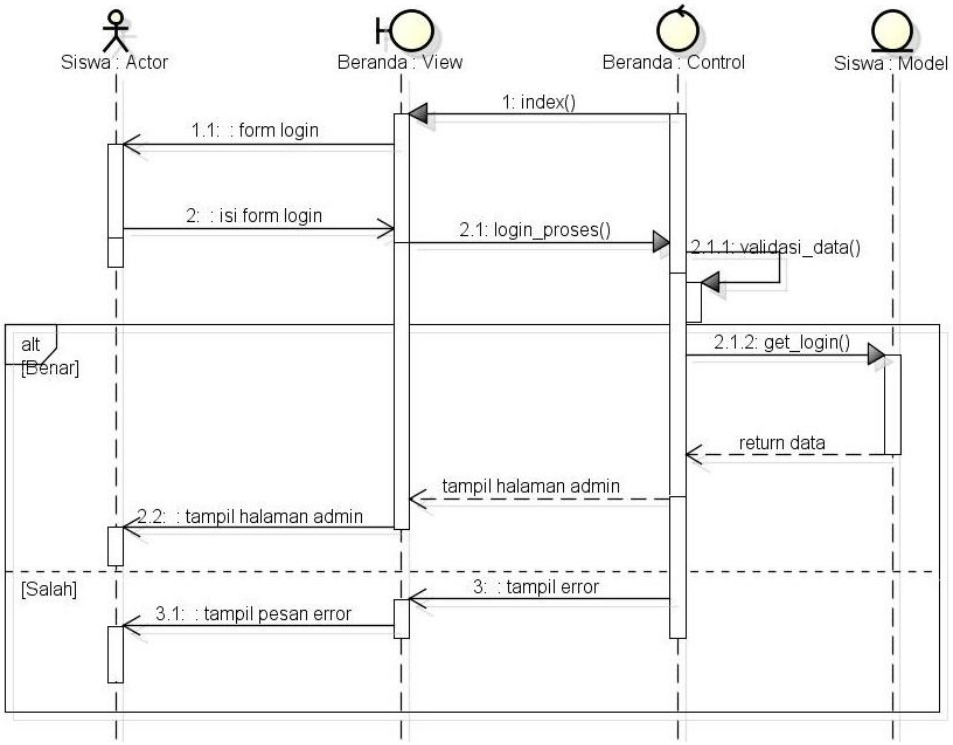
Gambar 3.40 ***Sequence Diagram Login Admin***

2. *Sequence* diagram *login* guru dapat dilihat pada Gambar 3.41



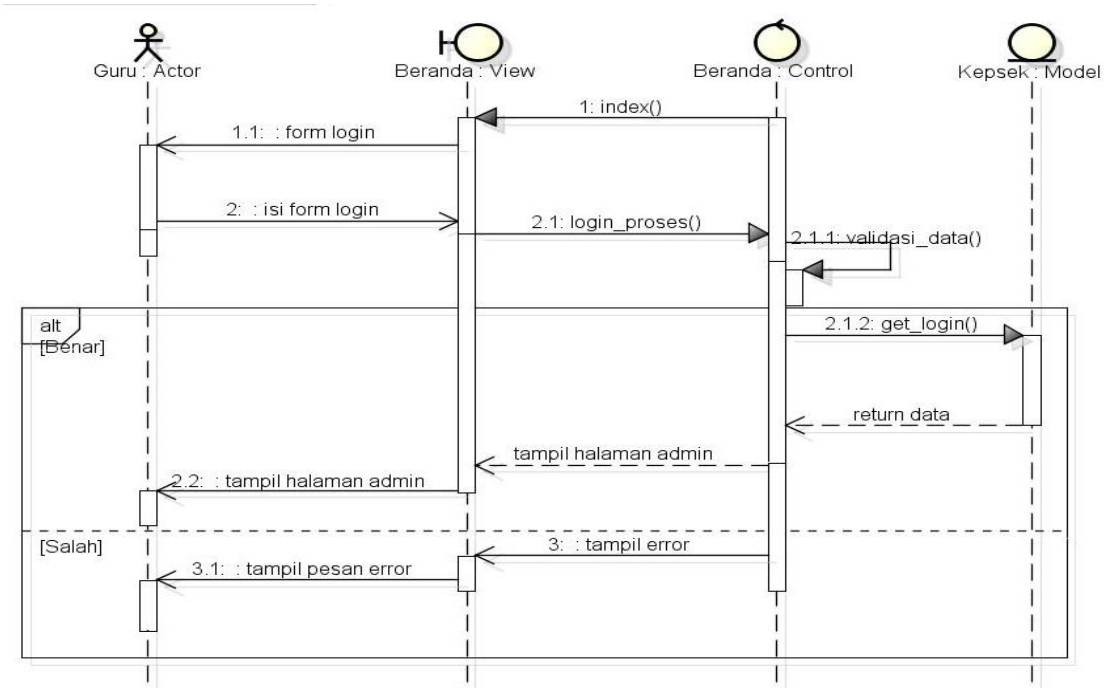
Gambar 3.41 ***Sequence diagram login guru***

3. *Sequence* diagram *login* siswa dapat dilihat pada Gambahr 3.42



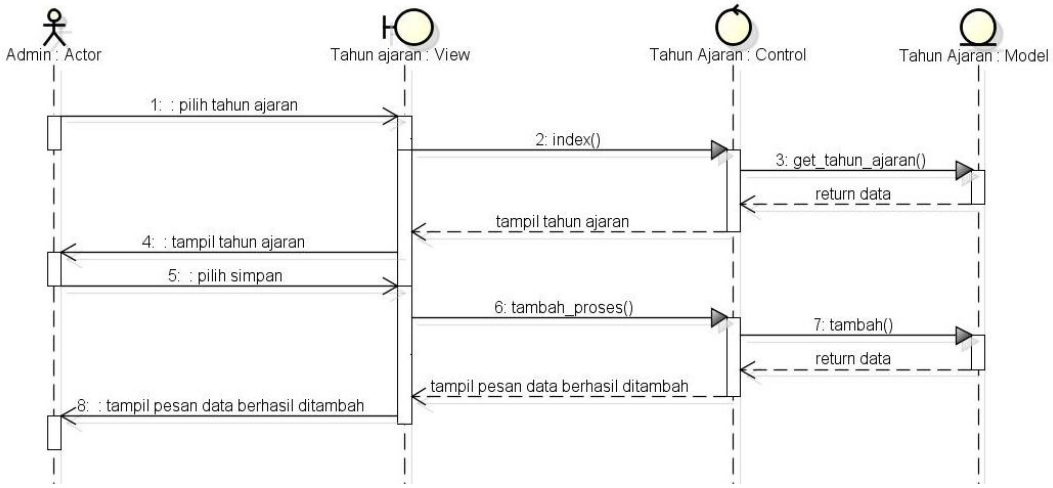
Gambar 3.42 ***Sequence diagram login siswa***

4. *Sequence* diagram *login* kepsek dapat dilihat pada gambahr 3.43



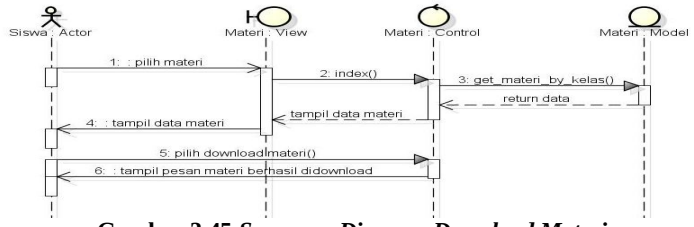
Gambar 3.43 ***Sequence diagram login kepala sekolah***

5. *Sequence* diagram pengolahan tahun ajaran dapat dilihat pada Gambar 3.44



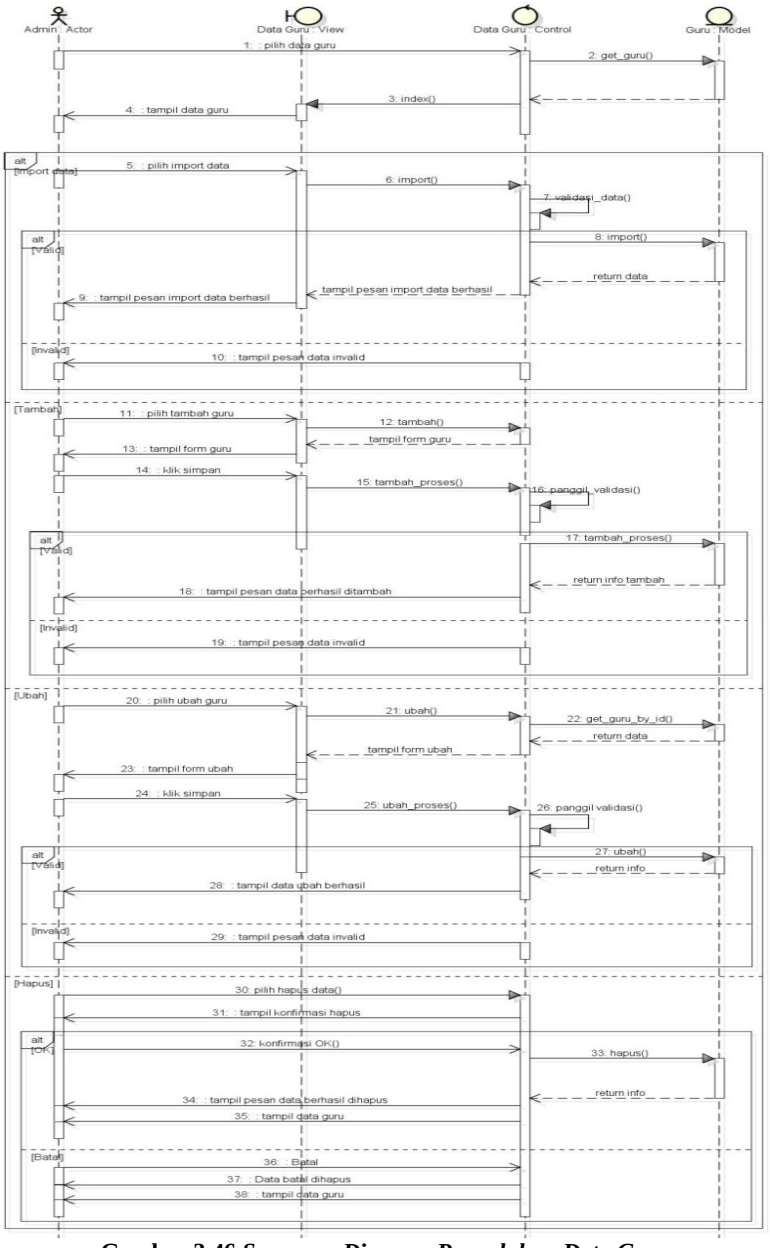
Gambar 3.44 ***Sequence Diagram Pengelohan Tahun Ajaran***

6. *Sequence* diagram *download* materi dapat dilihat pada Gambar 3.45



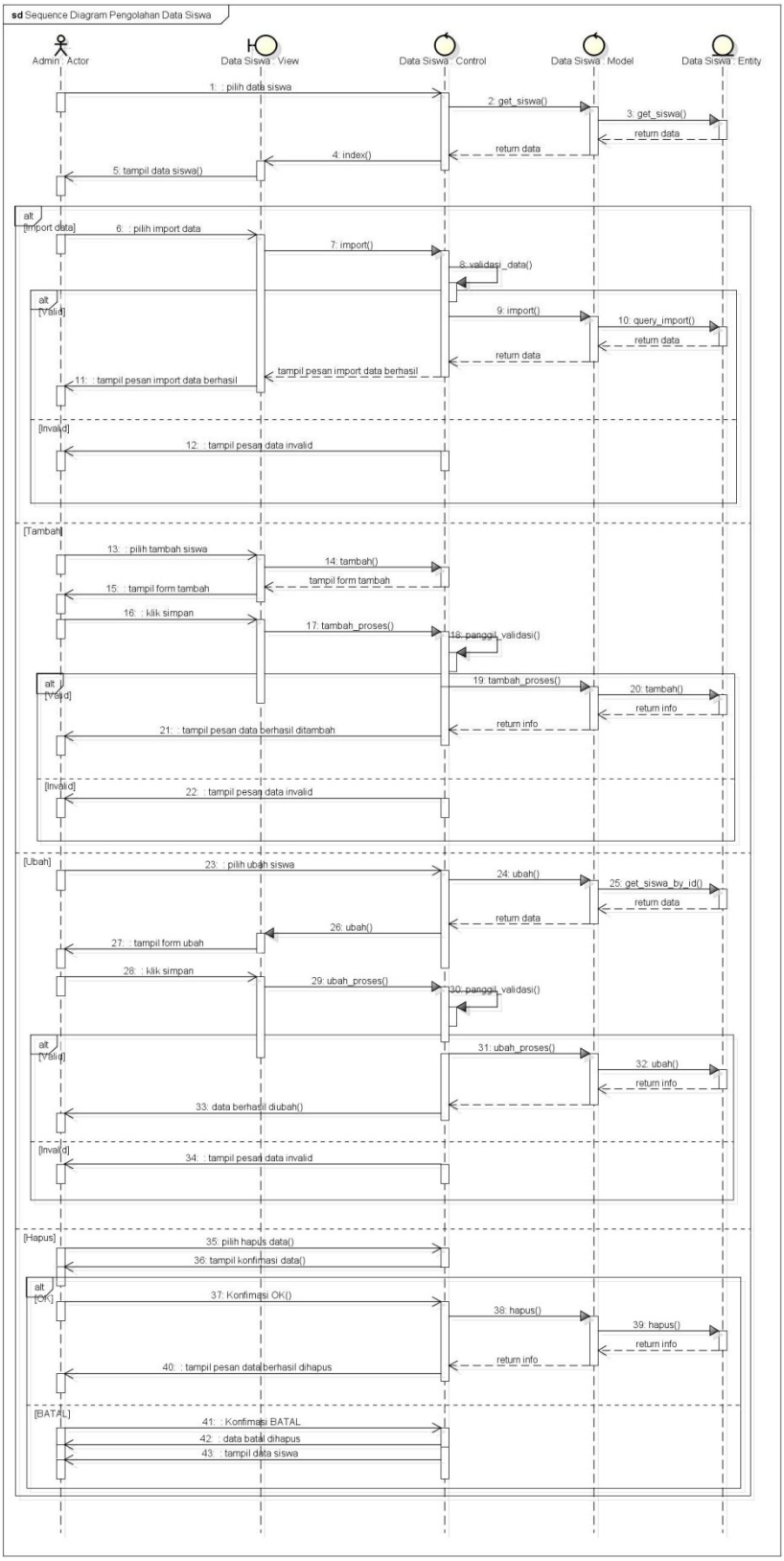
Gambar 3.45 ***Sequence Diagram Download Materi***

7. *Sequence* diagram pengelolaan data guru dapat dilihat pada gambar 3.46



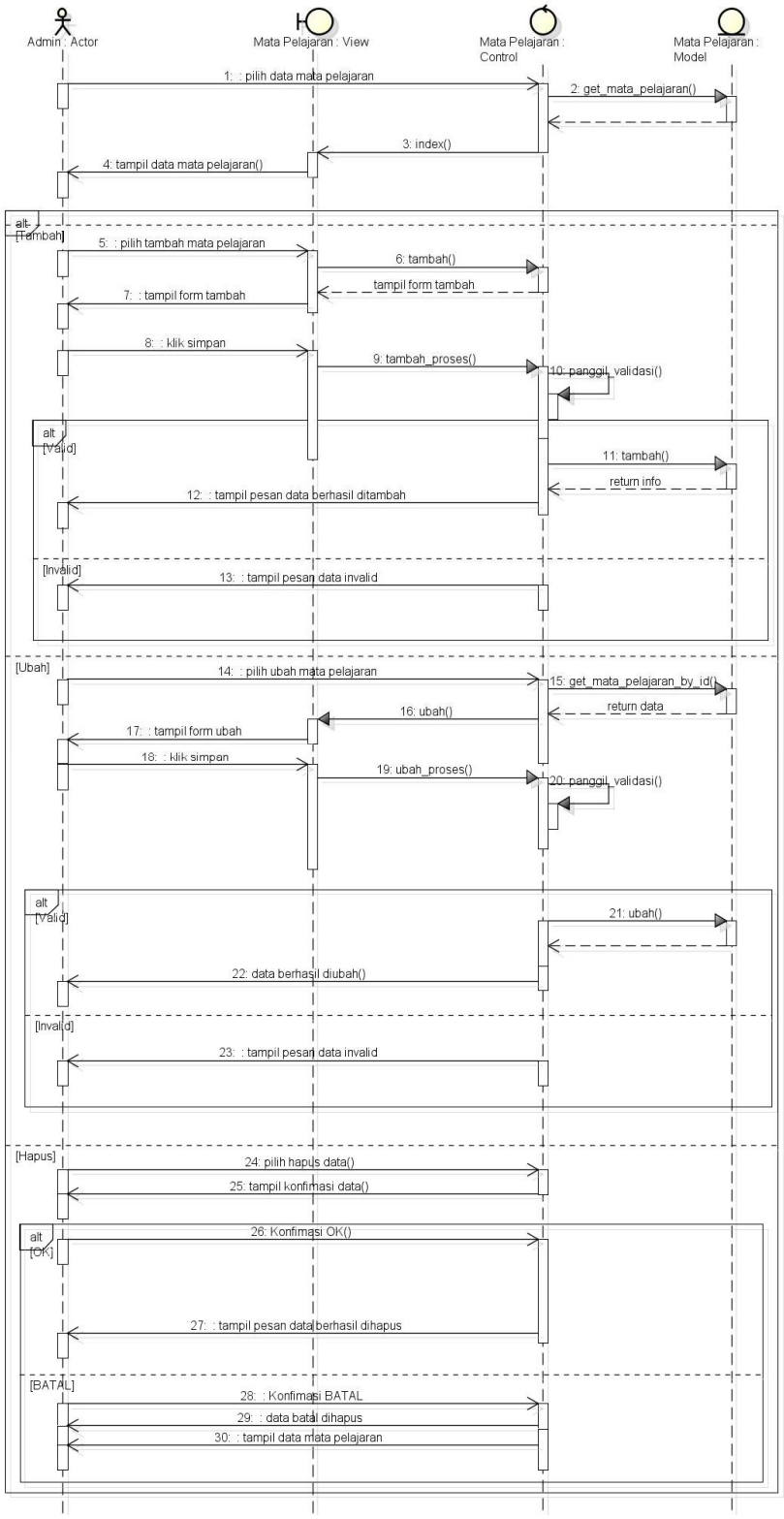
Gambar 3.46 ***Sequence Diagram Pengolahan Data Guru***

8. *Sequence* diagram pengelolaan data siswa dapat dilihat pada Gambar 3..47



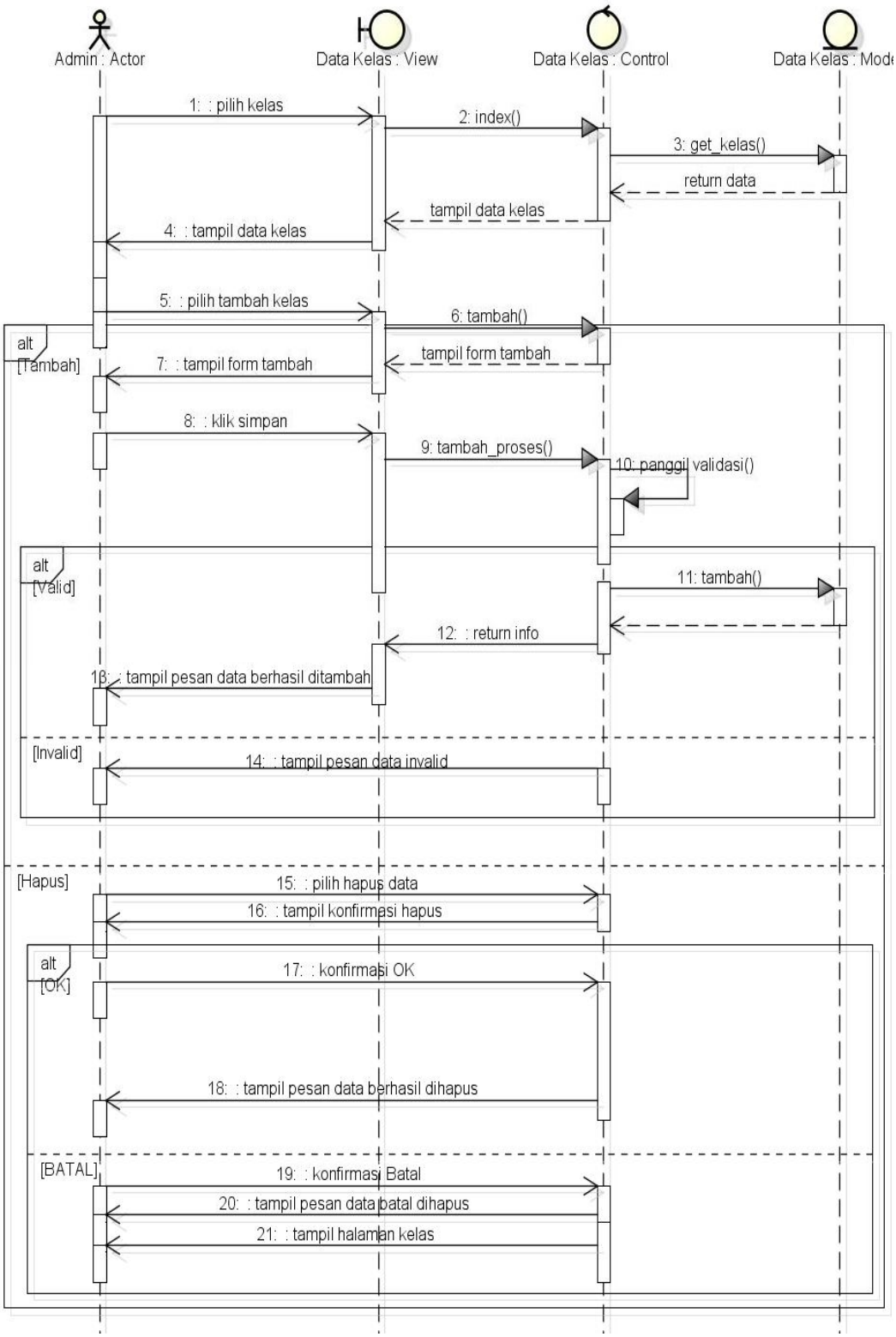
Gambar 3.47 ***Sequence Diagram Pengelolaan Data Siswa***

9. *Sequence* pengelolaan mata pelajaran dapat dilihat pada Gambar 3.48



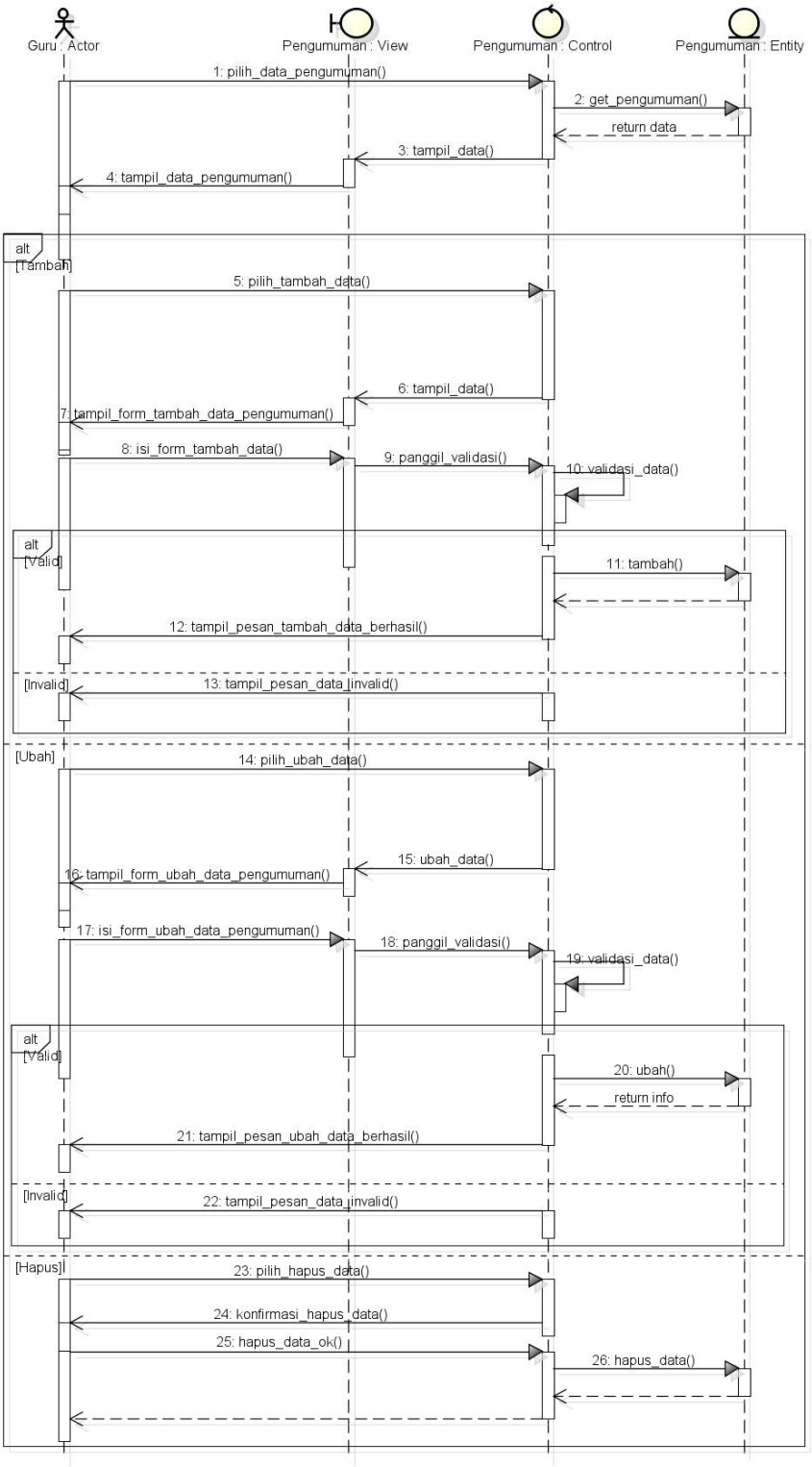
Gambar 3.48 ***Sequence Diagram Pengelolaan Mata Pelajaran***

10. *Sequence* diagram pengelolaan data kelas dapat dilihat pada Gambar 3.49



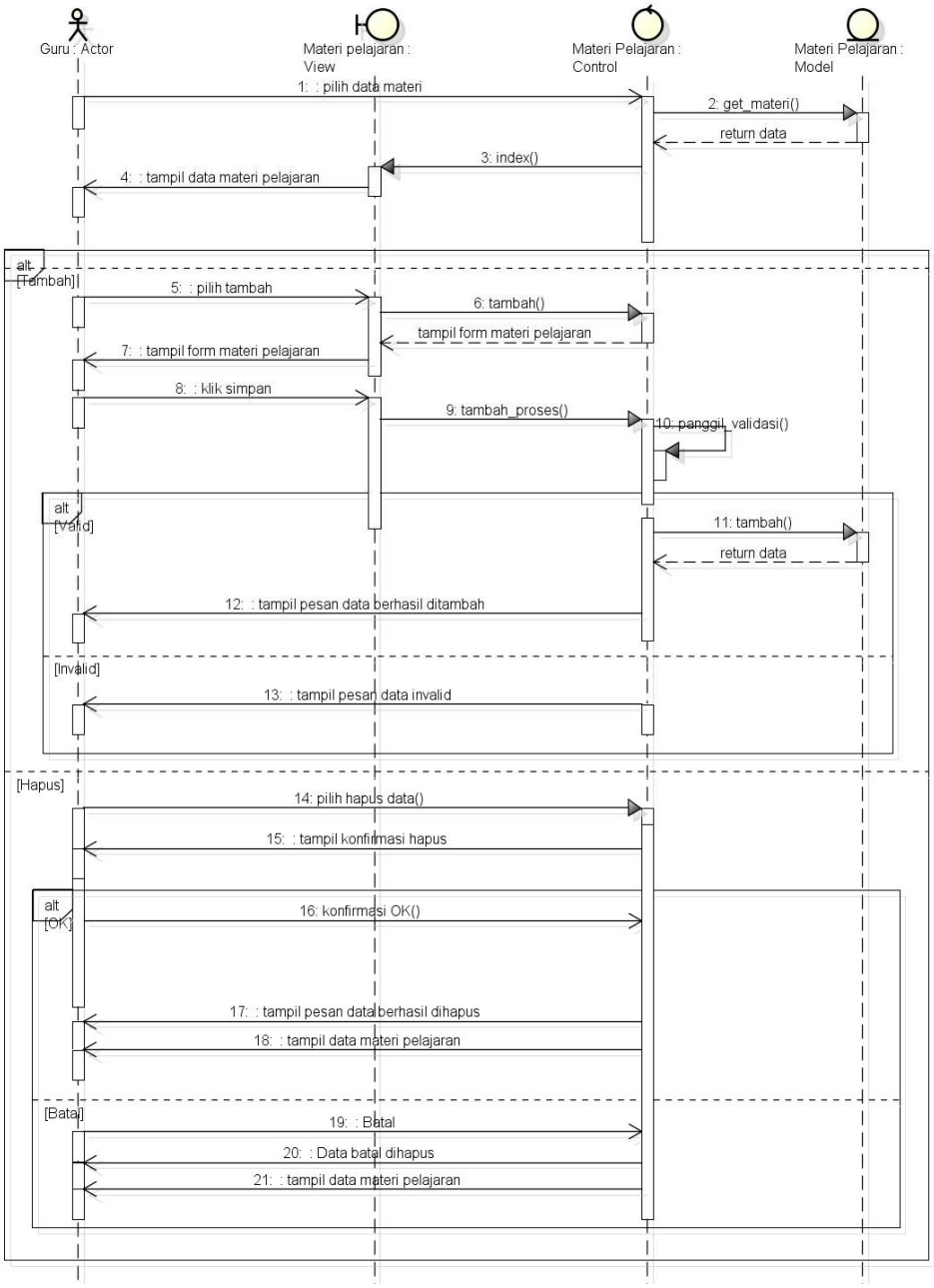
Gambar 3.49 ***Sequence Diagram Pengelolaan Data Kelas***

11. *Sequence* diagram pengolahan pengumuman dapat dilihat pada Gambar 3.50



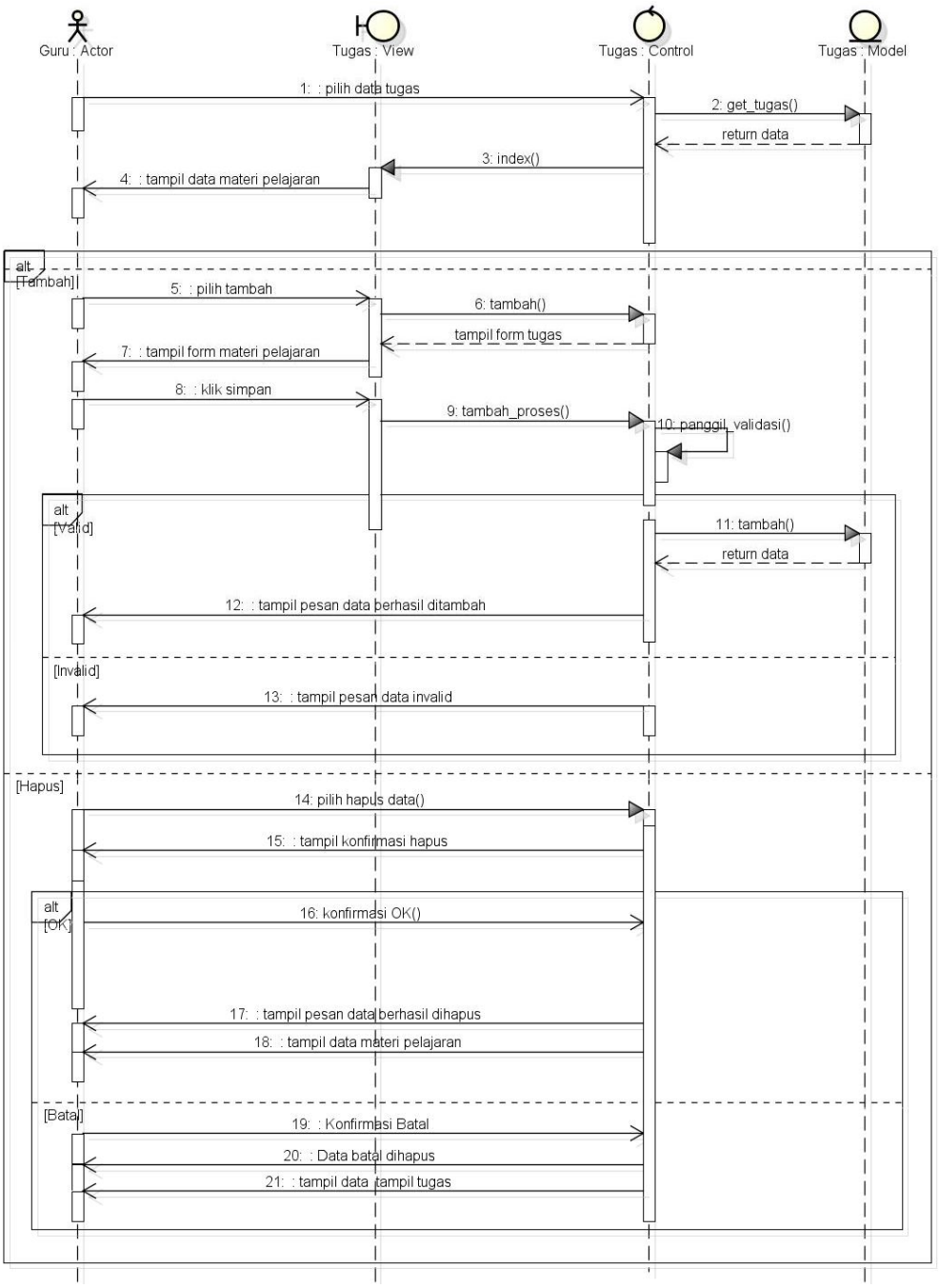
Gambar 3.50 ***Sequence Diagram Pengolahan Pengumuman***

12. *Sequence* diagram pengolahan materi pelajaran dapat dilihat pada Gambar 3.51



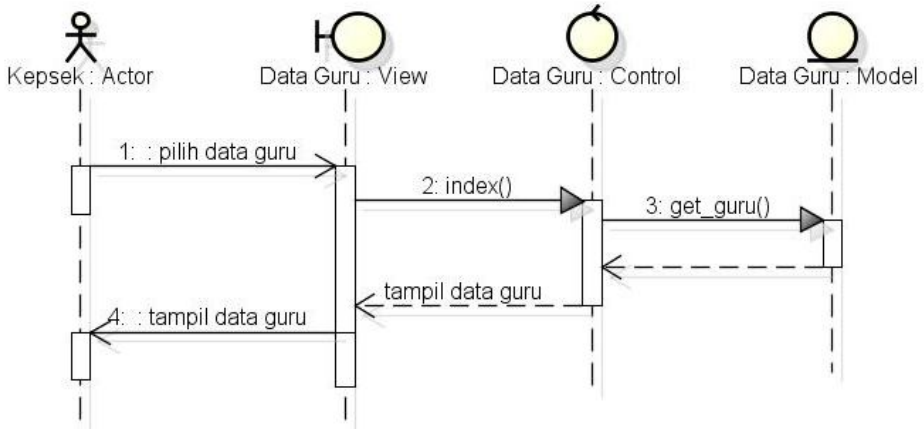
Gambar 3.51 Sequence *Diagram Pengolahan Materi Pelajaran*

13. *Sequence* diagram pengolahan tugas dapat dilihat pada Gambar 3.52



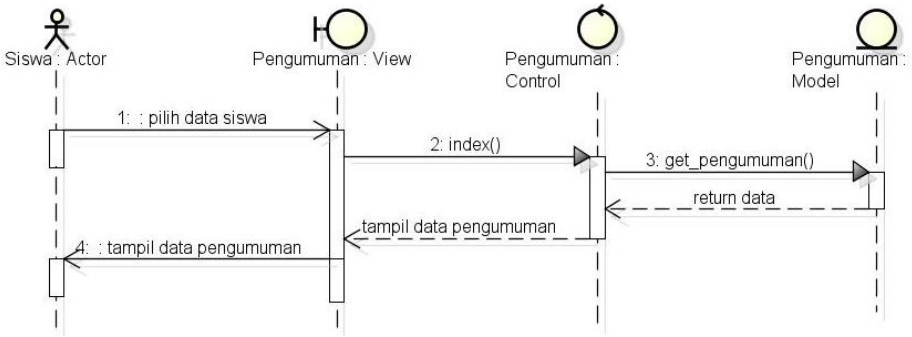
Gambar 3.52 Sequence *Diagram Pengolahan Tugas*

14. *Sequence diagram* lihat data guru dapat dilihat pada Gambar 3.53



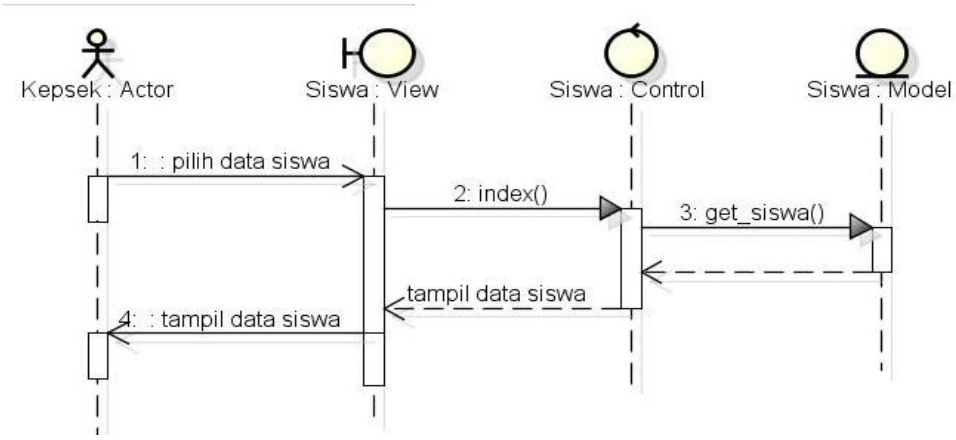
Gambar 3.53 ***Sequence Diagram Lihat Data Guru***

15. *Sequence* lihat pengumuman dapat dilihat pada Gambar 3.54



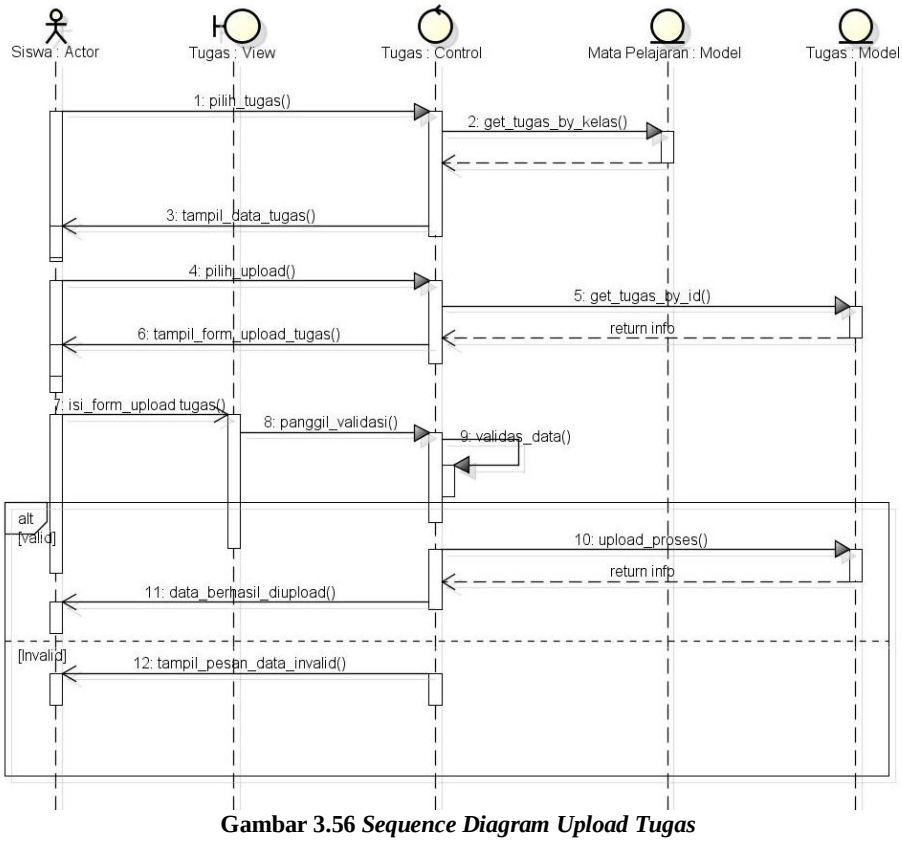
Gambar 3.54 ***Sequence Diagram Lihat Pengumuman***

16. *Sequence* diagram lihat data siswa dapat dilihat pada Gambar 3.55



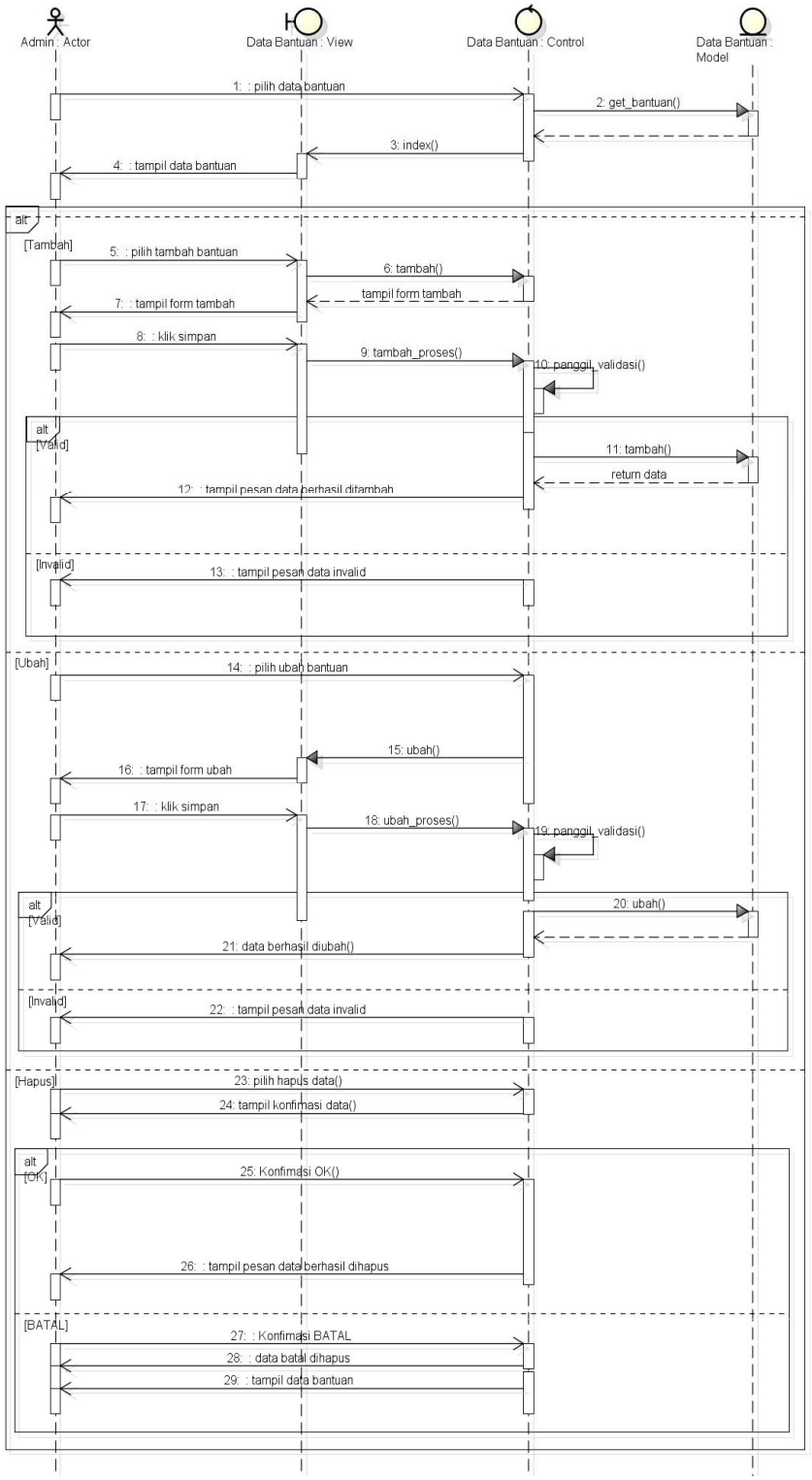
Gambar 3.55 ***Sequence Diagram Lihat Data Siswa***

17. *Sequence* diagram *upload* tugas dapat dilihat pada Gambar 3.56



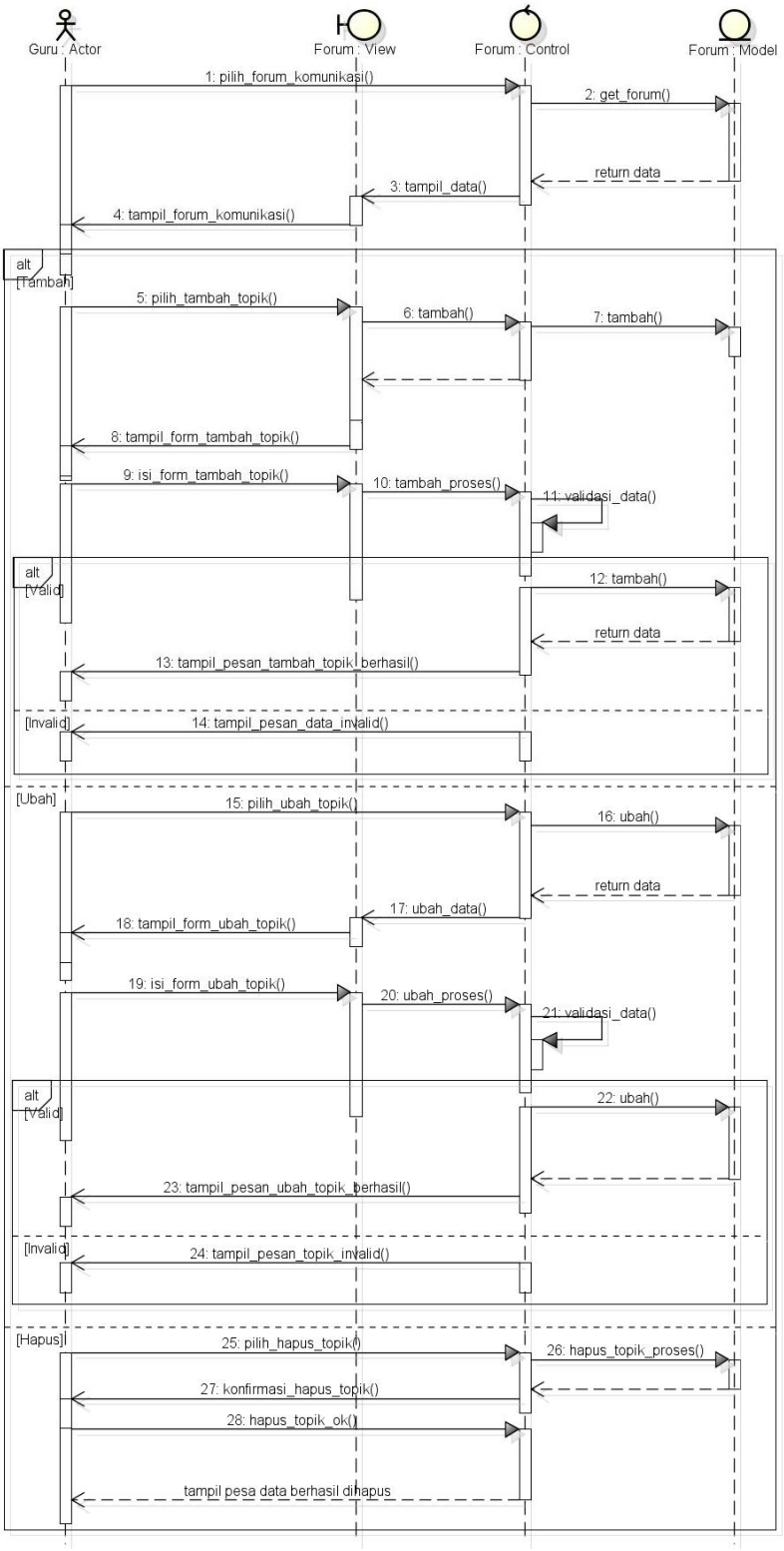
Gambar 3.56 ***Sequence Diagram Upload Tugas***

18. *Sequence* diagram pengelolaan bantuan dapat dilihat pada Gambar 3.57



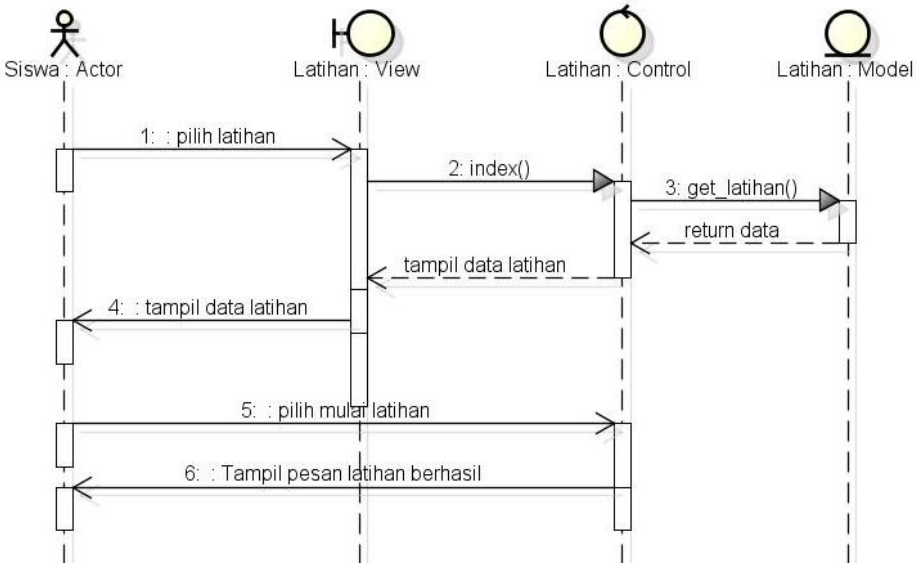
Gambar 3.57 ***Sequence Diagram Pengelolaan bantuan***

19. *Sequence* diagram forum komunikasi dapat dilihat pada Gambar 3.58



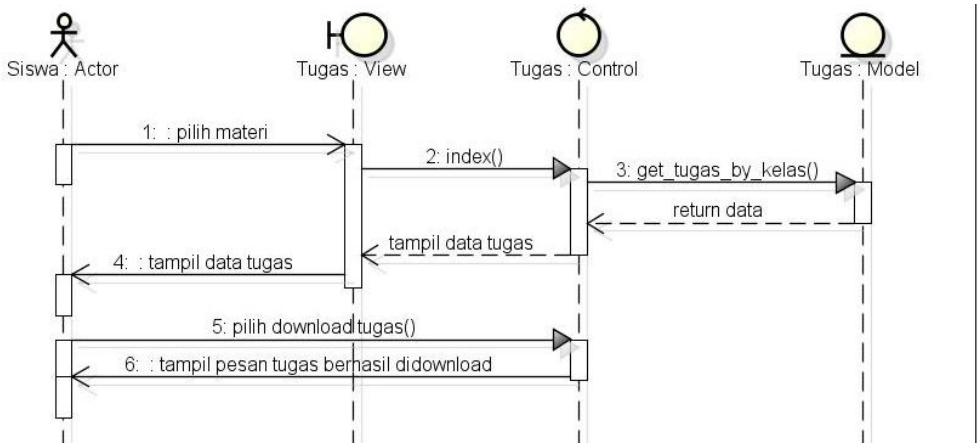
Gambar 3.58 ***Sequence Diagram Forum Komunikasi***

20. *Sequence* diagram latihan soal dapat dilihat pada Gambar 3.59



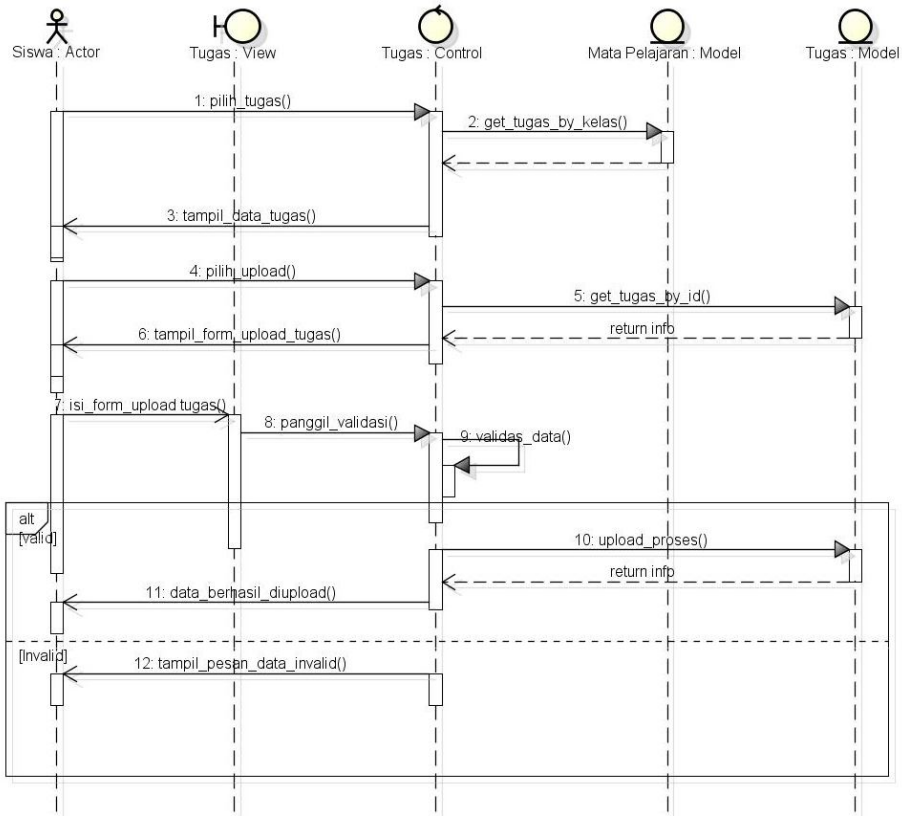
Gambar 3.59 ***Sequence Diagram latihan Soal***

21. *Sequence* diagram *download* tugas dapat dilihat pada Gambar 3.60



Gambar 3.60 Sequence Diagram Download Tugas

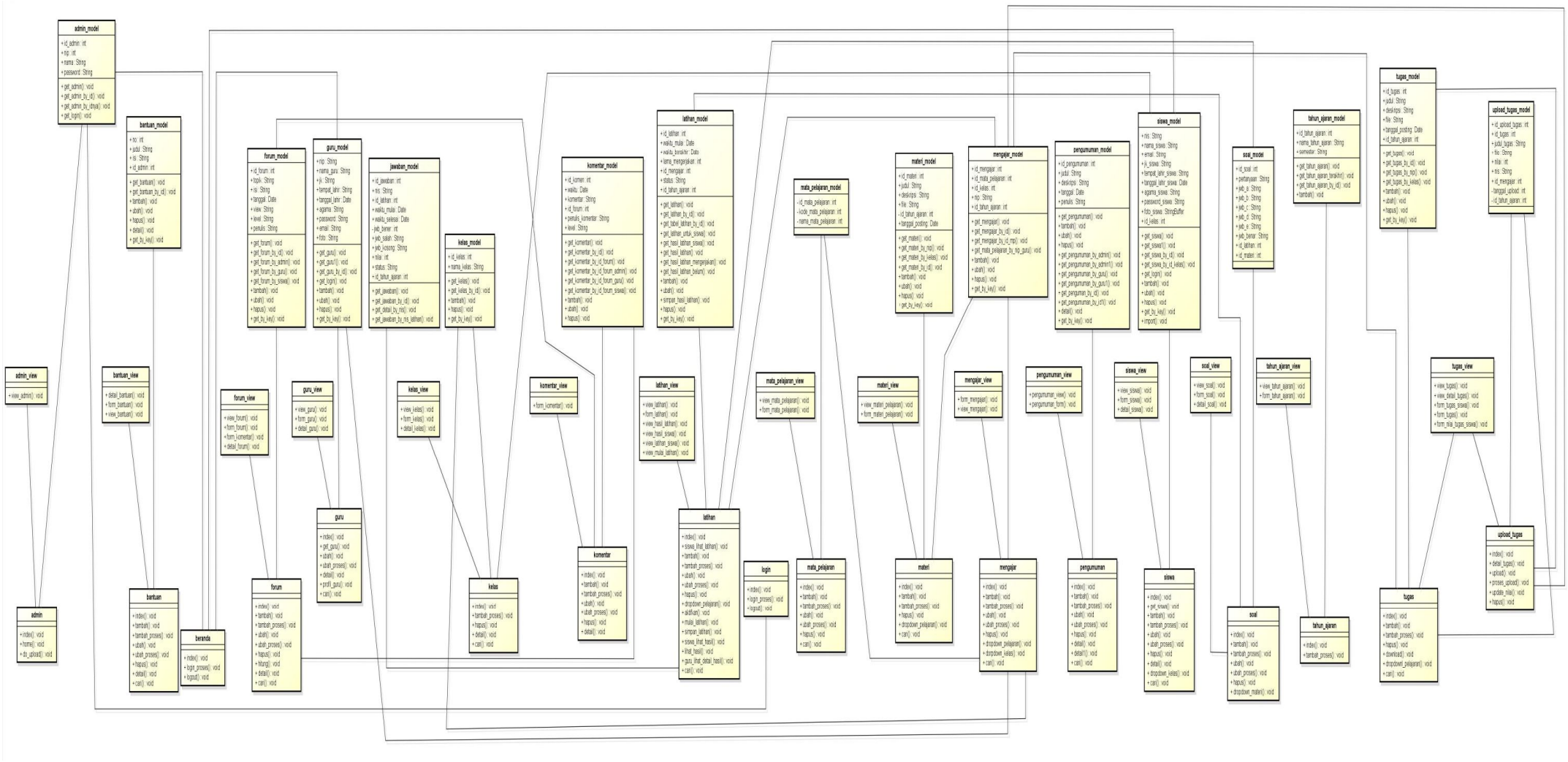
22. *Sequence* diagram *upload* tugas dapat dilihat pada Gambar 3.61.



Gambar 3.61Sequence Diagram

3.2.5.5 Class Diagram

Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara *class* diagram. *Class* diagram juga menunjukkan properti dan operasi sebuah class dan batasan-batasan yang terdapat dalam hubungan objek tersebut. *Class* diagram *e-learning* pada SMK-PU Negeri Prov. Jawa Barat dapat dilihat pada Gambar 3.62.



Gambar 3.62 *Class Diagram E-learning* SMK-PU Negeri Prov. Jawa Barat

Setiap kelas pada Gambar 3.62 memiliki jenis kelasnya masing-masing seperti yang terlihat pada Tabel 3.35.

Tabel 3.35 ***Deskripsi Class***

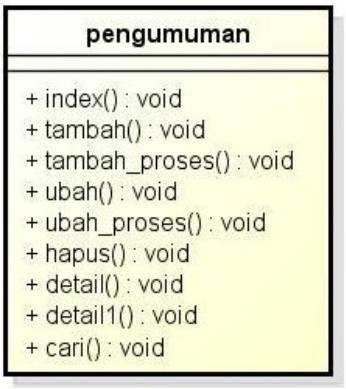
No	Nama	Jenis
1	Admin	Control
2	Bantuan	Control
3	Beranda	Control
4	Forum	Control
5	Guru	Control
6	Kelas	Control
7	Komentar	Control
8	Latihan	Control
9	Login	Control
10	mata_pelajaran	Control
11	Materi	Control
12	Mengajar	Control
13	Pengumuman	Control
14	Siswa	Control
15	tahun_ajaran	Control
16	Tugas	Control
17	upload_tugas	Control
18	admin_view	Boundary
19	bantuan_view	Boundary
20	forum_view	Boundary
21	guru_view	Boundary
22	kelas_view	Boundary
23	komentar_view	Boundary
24	latihan_view	Boundary
25	mata_pelajaran_view	Boundary
26	materi_view	Boundary
27	pengumuman_view	Boundary
28	siswa_view	Boundary
29	soal_view	Boundary
30	tahun_ajaran_view	Boundary
31	tugas_view	Boundary
32	admin_model	Model
33	bantuan_model	Model
34	forum_model	Model
35	guru_model	Model
36	jawaban_model	Model
37	kelas_model	Model
38	komentar	Model
39	latihan_model	Model

40	mata_pelajaran	Model
41	materi_model	Model
42	mengajar_model	Model
43	pengumuman_model	Model
44	siswa_model	Model
45	soal_model	Model
46	tahun_ajaran_model	Model
47	tugas_model	Model
48	upload_tugas	Model

Deskripsi kelas yang ada dijelaskan sebagai berikut.

1. *Class* pengumuman

   Class pengumuman merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data pengumuman dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* pengumuman dapat dilihat pada Gambar 3.63.



Gambar 3.63 ***Class Diagram Pengumuman Control***

a. Deskripsi *attribut*.

   Class pengumuman tidak memiliki *attribut*.

b. Deskripsi operasi

   Deskripsi operasi pengumuman dapat dilihat pada Tabel 3.36

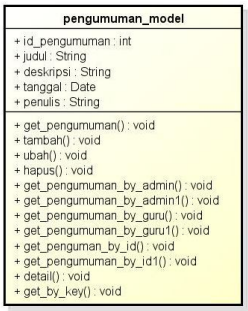
Tabel 3.36 ***Deskripsi Class Pengumuman Control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data pengumuman.
tambah() : void	Prosedur yang dipanggil ketika *user* meminta untuk

	memanggil form untuk mengisi pengumuman.
Tambah_proses() : void	Prosedur untuk mempross data masukkan dari form tambah, memanggil pengumuman model untuk menyimpan data pada *database.*
Ubah : void	Prosedur untuk memanggil form ubah pada pengumuman_view
Ubah_proses() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil pengumuman_model untuk menyimpan data yang sudah diubah.
Hapus_data() : void	Prosedur untuk menghapus data pengumuman, akan menampilkan pesan konfirmasi apakah data pengumuman akan dihapus, jika iya maka akan memanggil pengumuman_model untuk melakukan penghapusan data pada *database*
detail() : void	Prosedur untuk menampilkan detail pengumuman.
Cari() : void	Prosedur yang dipanggil untuk mencari pengumuman pada form pengumuman

2. *Class* pengumuman model

   Class pengumuman_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data model pengumuman dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* pengumuman_model dapat dilihat pada Gambar 3.64.



Gambar 3.64 ***Class Pengumuman Model***

Deskripsi operasi pada pengumuman_model digunakan untuk memanggil fungsi-fungsi yang dibutuhkan dapat dilihat pada Tabel 3.37

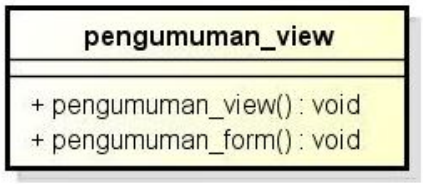
Tabel 3.37 ***Deskripsi Class Pengumuman Model***

Nama	Keterangan
Get_pengumuman(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan mengambil data dari database.

Tambah () : void	Prosedur untuk melakukan tambah data. Memanggil pengumuman_view untuk menampilkan form tambah.
Ubah () : void	Prosedur untuk memproses data masukan dari form ubah, memanggil pengumuman_model untuk menyimpan data yang sudah diubah.
hapus() : void	Prosedur untuk menghapus pada pengumuman.
Get_pengumuman_by_admin():void	Prosedur untuk menampilkan data pengumuman berdasarkan id admin pengumuman
Get_pengumuman_by_id():void	Prosedur untuk menampilkan data pengumuman berdasarkan id pengumuman
Get_pengumuman_by_guru():void	Prosedur untuk menampilkan data pengumuman berdasarkan nip guru pengumuman
Detail():void	Prosedur untuk menampilkan detail pengumuman

3. *Class* pengumuman view

   Class pengumuman_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data pengumuman dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* pengumuman_view dapat dilihat pada Gambar 3.65



Gambar 3.65 *Class Pengumuman View*

   a. Deskripsi *attribut*

      Class pengumuman_view tidak memiliki *attribut*.

   b. Deskripsi operasi

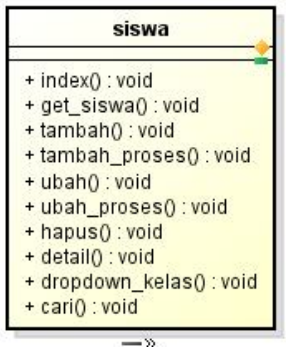
      Deskripsi operasi pengumuman_view dapat dilihat pada Tabel 3.38

Tabel 3.38 ***Deskripsi Class Pengumuman View***

Nama	Keterangan
Pengumuman_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* Pengumuman_model dan Pengumuman_control
Pengumuman_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Pengumuman_control

4. *Class* siswa *control*

   Class siswa_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data siswa dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* siswa_control dapat dilihat pada Gambar 3.66



Gambar 3.66 ***Class Siswa Control***

   a. Deskripsi *attribut*.

      Class siswa_control tidak memiliki *attribut*.

   b. Deskripsi operasi

      Deskripsi operasi class siswa control dapat dilihat pada Tabel III.39

Tabel 3.39 ***Deskripsi Class Siswa Control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data pengumuman.
tambah() : void	Prosedur untuk melakukan tambah data. Memanggil pengumuman_view untuk menampilkan form tambah.
Tambah_proses() : void	Prosedur untuk memproses data masukkan dari form tambah, memanggil siswa_model untuk

	menyimpan data pada database
Ubah() : void	Prosedur untuk memanggil form ubah pada siswa view.
Ubah_data() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil pengumuman_model untuk menyimpan data yang sudah diubah.
Hapus () : void	Prosedur untuk menghapus data pengumuman, akan menampilkan pesan konfirmasi apakah data pengumuman akan dihapus, jika iya maka akan memanggil pengumuman_model untuk melakukan penghapusan data pada *database*
Cari () : void	Prosedur untuk mencari data pada data siswa.
Detail () : void	Prosedur untuk menampilkan detail siswa
Dropdown_kelas () : void	Prosedur untuk menampilkan data kelas yang akan dimunculkan pada form tambah siswa.

5. *Class* siswa model

   Class siswa_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data siswa dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* siswa_model dapat dilihat pada Gambar 3.67



Gambar 3.67 ***Class Siswa Model***

Deskripsi operasi class siswa_model dapat dilihat pada Tabel 3.40

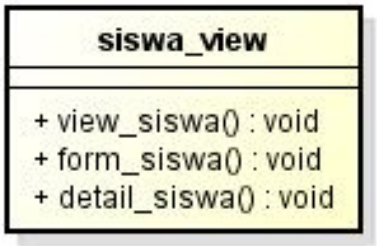
Tabel 3.40 ***Deskripsi Class Siswa Model***

Nama	Keterangan
Get_siswa(): void	Prosedur mengambil data dari database, digunakan untuk menampilkan seluruh data siswa
get_siswa_by_id():void	Prosedur mengambil data siswa berdasarkan

	NIS siswa, digunakan untuk menampilkan siswa berdasarkan NIS
Get_siswa_by_id_kelas():void	Prosedur mengambil data siswa berdasarkan id_kelas kelas, digunakan untuk menampilkan siswa berdasarkan Kelas
get_login():void	Prosedur mengambil data username dan password siswa, digunakan untuk login sistem
cari () : void	Prosedur untuk mencari data pada data siswa.
Tambah () : void	Prosedur untuk melakukan tambah data. Memanggil pengumuman_view untuk menampilkan form tambah.
Ubah () : void	Prosedur untuk memproses data masukan dari form ubah, memanggil pengumuman_model untuk menyimpan data yang sudah diubah.
Hapus () : void	Prosedur untuk menghapus data pengumuman, akan menampilkan pesan konfirmasi apakah data pengumuman akan dihapus, jika iya maka akan memanggil pengumuman_model untuk melakukan penghapusan data pada *database*
import() : void	Prosedur untuk import data dari file yang sudah ada ke dalam database

6. *Class* siswa *view*

   Class siswa_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data siswa dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* siswa_view dapat dilihat pada Gambar 3.68



Gambar 3.68 *Class Siswa View*

   a. Deskripsi *attribut*

      Class siswa_view tidak memiliki *attribut.*

   b. Deskripsi operasi

      Deskripsi operasi class siswa_view dapat dilihat pada Tabel 3.41

Tabel 3.41 ***Deskripsi Class Siswa View***

Nama	Keterangan
View_siswa(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* siswa_model dan siswa_control
Form_siswa() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* siswa_control
Detail_ssifwwww() : void	Prosedur untuk menampilkan detail siswa ketika memanggil prosedur detail siswa pada class siswa_control

7. *Class* soal control

   Class soal_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data soal dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* soal_control dapat dilihat pada Gambar 3.69



Gambar 3.69 ***Class Soal Control***

   a. Deskripsi *attribut*

      Class soal control tidak memiliki *attribut.*

   b. Deskripsi operasi

      Deskripsi operasi class soal_control dapat dilihat pada Tabel 3.42

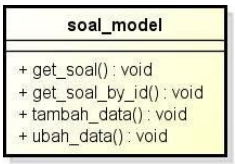
Tabel 3.42 ***Deskripsi Class Siswa Control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data pengumuman.
tambah() : void	Prosedur untuk melakukan tambah data. Memanggil pengumuman_view untuk menampilkan form tambah.

Tambah_proses() : void	Prosedur untuk memproses data masukkan dari form tambah, memanggil siswa_model untuk menyimpan data pada database
Ubah() : void	Prosedur untuk memanggil form ubah pada siswa view.
Ubah_data() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil pengumuman_model untuk menyimpan data yang sudah diubah.
Hapus () : void	Prosedur untuk menghapus data pengumuman, akan menampilkan pesan konfirmasi apakah data pengumuman akan dihapus, jika iya maka akan memanggil pengumuman_model untuk melakukan penghapusan data pada *database*
Cari () : void	Prosedur untuk mencari data pada data siswa.
Detail () : void	Prosedur untuk menampilkan detail siswa
Dropdown_kelas () : void	Prosedur untuk menampilkan data kelas yang akan dimunculkan pada form tambah siswa.

8. *Class* soal model

   Class soal_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data soal dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* soal_model dapat dilihat pada Gambar 3.70



Gambar 3.70 *Class Soal Model*

   a. Deskripsi *attribut*

      Class soal model tidak memiliki *attribut.*

   b. Deskripsi operasi

      Deskripsi operasi class soal_model dapat dilihat pada Tabel 3.43

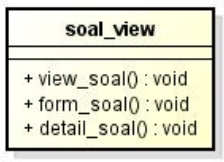
Tabel 3.43 *Deskripsi Soal Model*

Nama	Keterangan
Get_soal(): void	Prosedur mengambil soal dari database, digunakan untuk menampilkan seluruh data soal
get_soal_by_id():void	Prosedur mengambil data soal berdasarkan id_soal, digunakan untuk menampilkan data soal berdasarkan id
Tambah_data() : void	Prosedur untuk melakukan tambah data soal.

	Memanggil soal_view untuk menampilkan form tambah.
Ubah_data() : void	Prosedur untuk memproses data soal masukan dari form ubah, memanggil soal_model untuk menyimpan data yang sudah diubah.

9. *Class* soal view

   Class soal_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data soal dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* soal_view dapat dilihat pada Gambar 3.71



Gambar 3.71 *Class Siswa View*

a. Deskripsi *attribut*

   Class soal_view tidak memiliki *attribut*

b. Deskripsi operasi

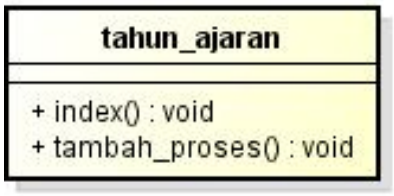
   Deskripsi operasi class soal_view dapat dilihat pada Tabel 3.44

Tabel 3.44 *Deskripsi Class Soal View*

Nama	Keterangan
Soal_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* soal_model dan soal_control
Soal_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* soal_control

10. *Class* tahun ajaran *control*

    Class tahun_ajaran_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data tahun ajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* tahun_ajaran_control dapat dilihat pada Gambar 3.72



Gambar 3.72 ***Class Tahun Ajaran Control***

a. Deskripsi *attribut*

Class tahun_ajaran_control tidak memiliki atribut

b. Deskripsi operasi

Deskripsi operasi class tahun_ajaran dapat dilihat pada Tabel 3.45

Tabel 3.45 ***Deskripsi Class Tahun Ajaran Control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data tahun ajaran.
Tambah_proses() : void	Prosedur untuk melakukan tambah data. Memanggil tahun_ajaran_view untuk menampilkan form tambah.

11. *Class* tahun ajaran model

Class tahun_ajaran_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data tahun ajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* tahun_ajaran_model dapat dilihat pada Gambar 3.73



Gambar 3.73 ***Tahun Ajaran Model***

a. Deskripsi *attribut*

Class tahun_ajaran_model tidak memiliki attribut

b. Deskripsi operasi

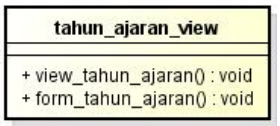
Deskripsi operasi class tahun_ajaran_model dapat dilihat pada Tabel 3.46

Tabel 3.46 ***Deskripsi Class Tahun Ajaran Model***

Nama	Keterangan
Get_tahun_ajaran(): void	Prosedur mengambil data dari database, digunakan untuk menampilkan seluruh data tahun ajaran.
Get_tahun_ajaran():void	Mengambil semua data tahun ajaran dari database
get_tahun_ajaran_by_id():void	Prosedur mengambil data berdasarkan id_tahun_ajaran, digunakan untuk menampilkan data tahun ajaran berdasarkan id
Tambah_data() : void	Prosedur untuk melakukan tambah data tahun ajaran. Memanggil tahun_ajaran_view untuk menampilkan form tambah.

12. *Class* tahun ajaran view

Class tahun_ajaran_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data tahun_ajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* tahun_ajaran_view dapat dilihat pada Gambar 3.74



Gambar 3.74 ***Class Tahun Ajaran View***

a. Deskripsi *attribut*

Class tahun_ajaran_view tidak memiliki attribut.

b. Deskripsi operasi

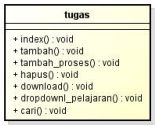
Deskripsi operasi class tahun_ajaran_view dapat dilihat pada Tabel 3.47

Tabel 3.47 ***Deskripsi class tahun ajaran***

Nama	Keterangan
Tahun_ajaran_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* tahun_ajaran_model dan tahun_ajaran_control
Tahun_ajaran_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* tahun_ajaran_control

13. *Class* tugas control

Class tugas_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data tugas dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* tugas_control dapat dilihat pada Gambar 3.75



Gambar 3.75 ***Class Tugas Control***

c. Deskripsi *attribut*

Class tugas_control tidak memiliki atribut

d. Deskripsi operasi

Deskripsi operasi class tugas_control dapat dilihat pada Tabel 3.48

Tabel 3.48 ***Deskripsi Class Tugas Control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data tugas.
Tampil_data() : void	Prosedur yang dipanggil ketika *user* meminta untuk menampilkan data, prosedur ini akan memberi perintah kepada *class* tugas_control untuk mengambil data pada *database.*
Tambah_data() : void	Prosedur untuk melakukan tambah data. Memanggil tugas_view untuk menampilkan form tambah.
Ubah_data() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil tugas_model untuk menyimpan data yang sudah diubah.
Hapus_data() : void	Prosedur untuk menghapus data tugas, akan

	menampilkan pesan konfirmasi apakah data tugas akan dihapus, jika iya maka akan memanggil tugas_model untuk melakukan penghapusan data pada *database*
Cari_data() : void	Prosedur untuk mencari data pada data tugas.

14. *Class* tugas model

Class tugas_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data tugas dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* tugas_model dapat dilihat pada Gambar 3.76



Gambar 3.76 *Class Tugas Model*

a. Deskripsi *attribut*

Class tugas_model tidak memiliki attribut

b. Deskripsi operasi

Deskripsi operasi class tugas_model dapat dilihat pada Tabel 3.49

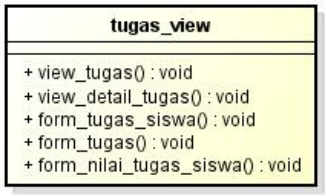
Tabel 3.49 *Deskripsi Class Tugas Model*

Nama	Keterangan
get_tugas(): void	Prosedur mengambil data dari database, digunakan untuk menampilkan seluruh data tugas.
get_tugas_by_id():void	Prosedur mengambil data berdasarkan id_tugas, digunakan untuk menampilkan data tugas berdasarkan id
Get_materi_pelajaran() : void	Prosedur mengambil data tugas dari databases, digunakan untuk menampilkan semua data tugas
Tmbah_data():void	Prosedur untuk melakukan tambah data tugas. Memanggil tugas_view untuk menampilkan form

	tambah.
Hapus_data():void	Prosedur untuk menghapus data tugas, akan menampilkan pesan konfirmasi apakah data tugas akan dihapus, jika iya maka akan memanggil tugas_model untuk melakukan penghapusan data pada *database*
Tampil_data():void	Prosedur yang dipanggil ketika *user* meminta untuk menampilkan data, prosedur ini akan memberi perintah kepada *class* tugas_control untuk mengambil data pada *database.*

15. *Class* tugas view

Class tugas_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data tugas dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* tugas_view dapat dilihat pada Gambar 3.77



Gambar 3.77 ***Class Tugas View***

a. Deskripsi *attribut*

Class tugas_view tidak memiliki attribut.

b. Deskripsi operasi

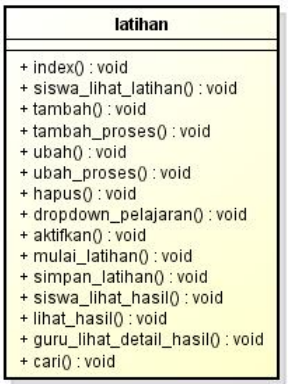
Deskripsi operasi class tugas_view dapat dilihat pada Tabel 3.50

Tabel 3.50 ***Deskripsi Class Tugas Tabel***

Nama	Keterangan
View_tugas(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* tugas_model dan tugas_control
View_detail_tugas	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* tugas_model dan tugas_control
Form_tugas() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* tugas_control

16. *Class* latihan control

Class latihan_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data latihan dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* latihan_control dapat dilihat pada Gambar 3.78



Gambar 3.78 ***Class Latihan Control***

a. Deskripsi *attribut*

Class latihan tidak memiliki atribut

b. Deskripsi operasi

Deskripsi operasi class latihan dapat dilihat pada Tabel 3.51

Tabel 3.51 ***Deskripsi Class Latihan Control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data latihan
Tambah () : void	Prosedur untuk melakukan tambah data. Memanggil latihan_view untuk menampilkan form tambah.
Tambah _proses() : void	Prosedur yang dipanggil ketika proses tambah data return tugas, melakukan perintah tambah data pada database
Ubah () : void	Prosedur untuk memproses data masukan dari form ubah, memanggil latihan_model untuk menyimpan data yang sudah diubah.
Ubah _proses():void	Prosedur yang dipanggil ketika proses ubah data return tugas, melakukan perintah ubah data pada database
Hapus_() : void	Prosedur untuk menghapus data tahun_ajaran, akan menampilkan pesan konfirmasi apakah data

	latihan akan dihapus, jika iya maka akan memanggil latihan_model untuk melakukan penghapusan data pada *database*.

17. *Class* latihan model

Class latihan_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data latihan dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* latihan_model dapat dilihat pada Gambar 3.79



Gambar 3.79 *Class Latihan Model*

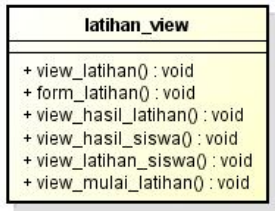
Deskripsi operasi class latihan_model dapat dilihat pada Tabel 3.52

Tabel 3.52 *Deskripsi Class Latihan Model*

Nama	Keterangan
Get_latihan(): void	Prosedur mengambil data dari database, digunakan untuk menampilkan seluruh data latihan.
get_latihan_by_id():void	Prosedur mengambil data latihan berdasarkan id_latihan, digunakan untuk menampilkan data latihan berdasarkan id
Tambah_data() : void	Prosedur untuk melakukan tambah data latihan. Memanggil latihan_view untuk menampilkan form tambah.
Ubah_latihan():void	Prosedur untuk memproses data masukan dari form ubah, memanggil latihan_model untuk menyimpan data yang sudah diubah.
Hapus_latihan():void	Prosedur untuk menghapus latihan, akan menampilkan pesan konfirmasi apakah data latihan akan dihapus, jika iya maka akan memanggil latihan_model untuk melakukan penghapusan data pada *database*

18. *Class* latihan view

    Class latihan_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data latihan dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* latihan_view dapat dilihat pada Gambar 3.80



Gambar 3.80 *Class Latihan View*

a. Deskripsi *attribut*

   Class latihan_view tidak memiliki attribut.

b. Deskripsi operasi

   Deskripsi operasi class latihan_view dapat dilihat pada Tabel 3.53

Tabel 3.53 *Deskripsi Class Latihan View*

Nama	Keterangan
Latihan_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* latihan_model dan latihan_control
Latihan_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* latihan_control

19. *Class* jawaban model

    Class jawaban_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data jawaban dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* jawaban_model dapat dilihat pada Gambar 3.81



Gambar 3.81 Class jawaban_model

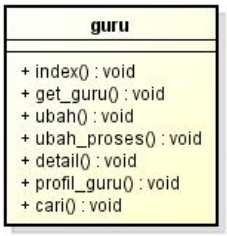
Deskripsi operasi class jawaban_model dapat dilihat pada Tabel 3.54

Tabel 3.54 ***Deskripsi Class Detail Latihan Model***

Nama	Keterangan
Get_jawaban(): void	Prosedur mengambil data dari database, digunakan untuk menampilkan seluruh data detail latihan.
Get_jawaban_by_id() void	Prosedur untuk memanggil jawaban berdasarkan id
Get_detail_by_nis() : void	Prosedur untuk memanggil jawaban berdasarkan nis siswa
Get_jawaban_by_nis_latihan () :void	Prosedur untuk memanggil jawaban berdasarkan nis siswa yang yang mengikuti latihan.

20. *Class* guru control

Class guru_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data guru dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* guru_control dapat dilihat pada Gambar 3.82



Gambar 3.82 ***Class guru_control***

a. Deskripsi *attribut*

Class guru tidak memiliki atribut

b. Deskripsi operasi

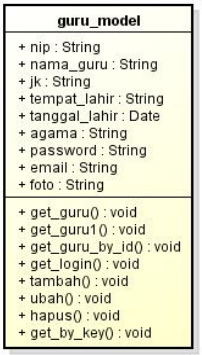
Deskripsi operasi class guru dapat dilihat pada Tabel 3.55

Tabel 3.55 ***Deskripsi class guru control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data guru.
tambah() : void	Prosedur yang dipanggil ketika *user* meminta untuk menampilkan data, prosedur ini akan memberi perintah kepada *class* guru_control untuk mengambil data pada *database.*
Tambah_proses() : void	Prosedur untuk melakukan tambah data. Memanggil guru_view untuk menampilkan form tambah.
Ubah () : void	Prosedur untuk memproses data masukan dari form ubah, memanggil guru_model untuk menyimpan data yang sudah diubah.
Hapus () : void	Prosedur untuk menghapus data guru, akan menampilkan pesan konfirmasi apakah data guru akan dihapus, jika iya maka akan memanggil guru_model untuk melakukan penghapusan data pada *database*
Cari () : void	Prosedur untuk mencari data pada data guru.

21. *Class* guru model

Class guru_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data guru dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* guru_model dapat dilihat pada Gambar 3.83



Gambar 3.83 ***Class Guru Model***

a. Deskripsi *attribut*

Class guru_model tidak memiliki attribut

b. Deskripsi operasi

Deskripsi operasi class guru_model dapat dilihat pada Tabel 3.56

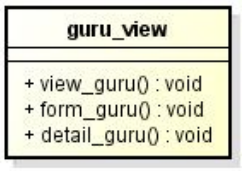
Tabel 3.56 ***Deskripsi class guru model***

Nama	Keterangan
Get_guru(): void	Prosedur mengambil data dari database, digunakan untuk menampilkan seluruh data guru.
get_guru_by_id():void	Prosedur mengambil data guru berdasarkan NIP guru, digunakan untuk menampilkan data guru berdasarkan NIP
Get_login() : void	Prosedur untuk melakukan login dengan menggunakan NIP dan password guru
Tambah_data():void	Prosedur untuk melakukan tambah data guru. Memanggil guru_view untuk menampilkan form tambah.
Ubah_data():void	Prosedur untuk memproses data masukan dari form ubah, memanggil guru_model untuk menyimpan data yang sudah diubah.
Cari_data():void	Prosedur untuk mencari data pada data guru.
Hapus_data():void	Prosedur untuk menghapus data, akan menampilkan pesan konfirmasi apakah data guru akan dihapus, jika iya maka akan memanggil guru_model untuk melakukan penghapusan data pada *database*
Upload_gambar():void	Prosedur yang digunakan untuk upload gambar pada guru

22. *Class* guru view

Class guru_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data guru dan bertugas untuk memanggil *class*-

class yang lainnya. Atribut dan operasi yang dimiliki *class* guru_view dapat dilihat pada Gambar 3.84



Gambar 3.84 *Class Guru View*

a. Deskripsi *attribut*

Class guru_view tidak memiliki attribut.

b. Deskripsi operasi

Deskripsi operasi class guru_view dapat dilihat pada Tabel 3.57

Tabel 3.57 *Deskripsi class guru view*

Nama	Keterangan
guru_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* guru_model dan guru_control
guru_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* guru_control
Guru_detail():void	Prosedur untuk menampilkan detail guru ketika memanggil prosedur detail guru pada class guru_control

23. *Class* guru

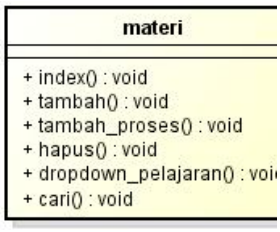
Class guru merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class guru dapat di lihat pada Gambar 3.85



Gambar 3.85 *Class guru*

24. *Class* materi pelajaran control

Class materi_pelajaran_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data mata pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* materi_pelajaran_control dapat dilihat pada Gambar 3.86



Gambar 3.86 ***Class materi pelajaran control***

a. Deskripsi *attribut*

Class materi_pelajaran_control tidak memiliki atribut

b. Deskripsi operasi

Deskripsi operasi class materi_pelajaran_control dapat dilihat pada Tabel 3.58

Tabel 3.58 ***Deskripsi class materi pelajaran control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data tugas.
Tampil_data() : void	Prosedur yang dipanggil ketika *user* meminta untuk menampilkan data, prosedur ini akan memberi perintah kepada *class* materi_pelajaran_control untuk mengambil data pada *database*.
Tambah_data() : void	Prosedur untuk melakukan tambah data. Memanggil materi_pelajaran_view untuk menampilkan form tambah.
Ubah_data() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil materi_pelajaran_model untuk menyimpan data yang sudah diubah.
Hapus_data() : void	Prosedur untuk menghapus data materi pelaran, akan menampilkan pesan konfirmasi apakah data materi pelajaran akan dihapus, jika iya maka akan memanggil materi_pelajaran_model untuk melakukan penghapusan data pada *database*
Cari_data() : void	Prosedur untuk mencari data pada data materi pelajaran.

25. *Class* materi pelajaran model

Class materi_pelajaran_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* materi_pelajaran_model dapat dilihat pada Gambar 3.86



Gambar 3.87 ***Class materi pelajaran model***

a. Deskripsi *attribut*

Class materi_pelajaran_model tidak memiliki attribut

b. Deskripsi operasi

Deskripsi operasi class materi_pelajaran_model dapat dilihat pada Tabel 3.59

Tabel 3.59 ***Deskripsi class materi pelajaran model***

Nama	Keterangan
Get_materi_pelajaran(): void	Prosedur mengambil materi pelajaran dari database, digunakan untuk menampilkan seluruh data materi pelajaran.
get_materi_pelajaran_by_id():void	Prosedur mengambil data materi pelajaran berdasarkan id materi pelajaran, digunakan untuk menampilkan data materi pelajaran berdasarkan id
Tambah_data() : void	Prosedur untuk melakukan tambah data materi pelajaran. Memanggil materi_pelajaran_view untuk menampilkan form tambah.
Hapus_latihan():void	Prosedur untuk menghapus materi pelajaran, akan menampilkan pesan konfirmasi apakah data materi pelajaran akan dihapus, jika iya maka akan memanggil materi_pelajaran_model untuk melakukan penghapusan data pada *database*

26. *Class* materi pelajaran view

Class materi_pelajaran_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* materi_pelajaran_view dapat dilihat pada Gambar 3.88



Gambar 3.88 ***Class materi pelajaran view***

a. Deskripsi *attribut*

Class materi_pelajaran_view tidak memiliki attribut.

b. Deskripsi operasi

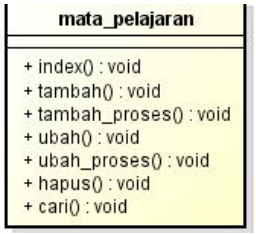
Deskripsi operasi class materi_pelajaran_view dapat dilihat pada Tabel 3.60

Tabel 3.60 ***Deskripsi class materi pelajaran view***

Nama	Keterangan
Materi_pelajaran_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* materi_pelajaran_model dan materi_pelajaran_control
Materi_pelajaran_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* ateri_pelajaran_control

27. *Class* mata pelajaran control

Class mata_pelajaran_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data mata pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* mata_pelajaran_control dapat dilihat pada Gambar 3.89



Gambar 3.89 ***Class mata pelajaran control***

a. Deskripsi *attribut*

Class mata_pelajaran_control tidak memiliki atribut

b. Deskripsi operasi

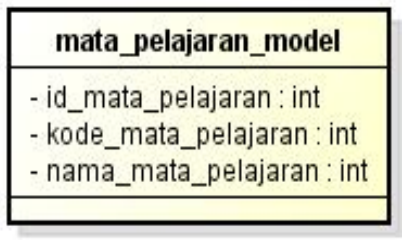
Deskripsi operasi class mata_pelajaran_control dapat dilihat pada Tabel 3.61

Tabel 3.61 ***Deskripsi class mata pelajaran control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data tugas.
Tampil_data() : void	Prosedur yang dipanggil ketika *user* meminta untuk menampilkan data, prosedur ini akan memberi perintah kepada *class* mata_pelajaran_control untuk mengambil data pada *database.*
Tambah_data() : void	Prosedur untuk melakukan tambah data. Memanggil mata_pelajaran_view untuk menampilkan form tambah.
Ubah_data() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil mata_pelajaran_model untuk menyimpan data yang sudah diubah.
Hapus_data() : void	Prosedur untuk menghapus data materi pelaran, akan menampilkan pesan konfirmasi apakah data materi pelajaran akan dihapus, jika iya maka akan memanggil mata_pelajaran_model untuk melakukan penghapusan data pada *database*
Cari_data() : void	Prosedur untuk mencari data pada data materi pelajaran.

28. *Class* mata pelajaran model

Class mata_pelajaran_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data mata pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* mata_pelajaran_model dapat dilihat pada Gambar 3.90



Gambar 3.90 ***Class mata pelajaran model***

a. Deskripsi *attribut*

Class mata_pelajaran_control tidak memiliki atribut

b. Deskripsi operasi

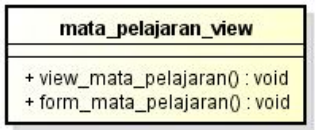
Deskripsi operasi class mata_pelajaran_control dapat dilihat pada Tabel 3.62

Tabel 3.62 ***Deskripsi mata pelajaran model***

Nama	Keterangan
Get_mata_pelajaran(): void	Prosedur mengambil mata pelajaran dari database, digunakan untuk menampilkan seluruh data mata pelajaran
get_ mata_pelajaran _by_id():void	Prosedur mengambil data mata pelajaran berdasarkan id mata pelajaran, digunakan untuk menampilkan data mata pelajaran berdasarkan id
Cari_data() : void	Prosedur untuk melakukan cari data mata pelajaran
Tambah_data() : void	Prosedur untuk melakukan tambah data mata pelajaran. Memanggil mata_pelajaran_view untuk menampilkan form tambah.
Ubah_latihan():void	Prosedur untuk memproses data masukan dari form ubah, memanggil mata_pelajaran_model untuk menyimpan data yang sudah diubah.
Hapus_latihan():void	Prosedur untuk menghapus data, akan menampilkan pesan konfirmasi apakah data mata pelajaran akan dihapus, jika iya maka akan memanggil
	mata_pelajaran_model untuk melakukan penghapusan data pada *database*

29. *Class* mata pelajaran view

Class mata_pelajaran_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* mata_pelajaran_view dapat dilihat pada Gambar 3.91



Gambar 3.91 ***Class mata pelajaran view***

a. Deskripsi *attribut*

Class mata_pelajaran_view tidak memiliki atribut

b. Deskripsi operasi

Deskripsi operasi class mata_pelajaran_view dapat dilihat pada Tabel 3.63

Tabel 3.63 ***Deskripsi class mata pelajaran view***

Nama	Keterangan
mata_pelajaran_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* mata_pelajaran_model dan mata_pelajaran_control
mata_pelajaran_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* mata_pelajaran_control

30. *Class* detail mata pelajaran

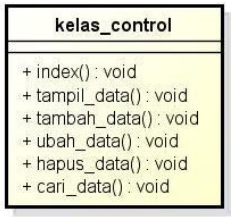
Class mengajar merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class detail mata pelajaran dapat di lihat pada Gambar 3.92



Gambar 3.92 ***Class detail mata pelajaran***

31. *Class* kelas control

Class kelas_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data mata pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* kelas_control dapat dilihat pada Gambar 3.93



Gambar 3.93 *Class kelas control*

a. Deskripsi *attribut*

Class kelas_control tidak memiliki atribut

b. Deskripsi operasi

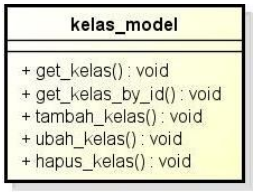
Deskripsi operasi class kelas_control dapat dilihat pada Tabel 3.64

Tabel 3.64 *Deskripsi class kelas control*

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data kelas.
Tampil_data() : void	Prosedur yang dipanggil ketika *user* meminta untuk menampilkan data, prosedur ini akan memberi perintah kepada *class* kelas_control untuk mengambil data pada *database.*
Tambah_data() : void	Prosedur untuk melakukan tambah data. Memanggil kelas_view untuk menampilkan form tambah.
Ubah_data() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil kelas_model untuk menyimpan data yang sudah diubah.
Hapus_data() : void	Prosedur untuk menghapus data kelas, akan menampilkan pesan konfirmasi apakah data kelas akan dihapus, jika iya maka akan memanggil kelas_model untuk melakukan penghapusan data pada *database*
Cari_data() : void	Prosedur untuk mencari data pada data kelas

32. *Class* kelas model

Class kelas_model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* kelas _model dapat dilihat pada Gambar 3.94



Gambar 3.94 *Class kelas model*

a. Deskripsi *attribut*

Class kelas _model tidak memiliki atribut

b. Deskripsi operasi

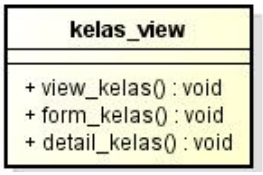
Deskripsi operasi class kelas_model dapat dilihat pada Tabel 3.65

Tabel 3.65 *Deskripsi class kelas model*

Nama	Keterangan
Get_ kelas (): void	Prosedur mengambil kelas dari database, digunakan untuk menampilkan seluruh data orang tua.
get_ kelas _by_id():void	Prosedur mengambil data kelas berdasarkan id kelas, digunakan untuk menampilkan data kelas berdasarkan id
Tambah_ kelas () : void	Prosedur untuk melakukan tambah data kelas. Memanggil kelas_view untuk menampilkan form tambah.
Ubah_ kelas ():void	Prosedur untuk memproses data masukan dari form ubah, memanggil kelas_model untuk menyimpan data yang sudah diubah.
Hapus_ kelas ():void	Prosedur untuk menghapus data, akan menampilkan pesan konfirmasi apakah data kelas akan dihapus, jika iya maka akan memanggil kelas_model untuk melakukan penghapusan data pada *database*

33. *Class* kelas view

Class kelas _view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* kelas _view dapat dilihat pada Gambar 3.95



Gambar 3.95 *Class kelas view*

c. Deskripsi *attribut*

Class kelas_view tidak memiliki atribut

d. Deskripsi operasi

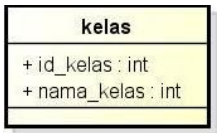
Deskripsi operasi class kelas_view dapat dilihat pada Tabel 3.66

Tabel 3.66 *Deskripsi class kelas view*

Nama	Keterangan
kelas _view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* kelas _model dan kelas _control
kelas _form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* kelas _control

34. *Class* kelas

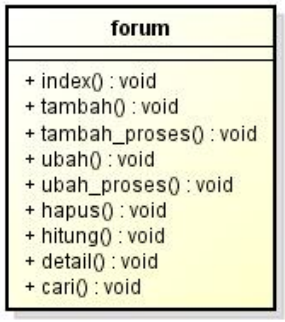
Class materi pelspelajaransrsn merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class msteri pelajaran dapat di lihat pada Gambar 3.96



Gambar 3.96 *Class kelas*

35. *Class* forum control

Class forum merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data mata pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* forum dapat dilihat pada Gambar 3.97



Gambar 3.97 ***Class Forum Komunikasi control***

e. Deskripsi *attribut*

Class forum control tidak memiliki atribut

f. Deskripsi operasi

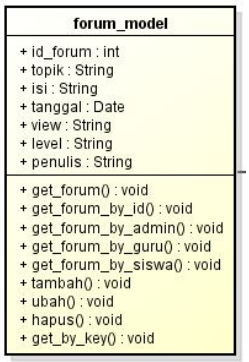
Deskripsi operasi class forum_komunikasi_control dapat dilihat pada Tabel 3.67

Tabel 3.67 ***Deskripsi class forum komunikasi control***

Nama	Keterangan
Index(): void	Merupakan prosedur yang dijalankan ketika *class* pertama kali dipanggil, prosedur ini akan memanggil prosedur tampil data untuk menampilkan data forum komunikasi.
Tampil_data() : void	Prosedur yang dipanggil ketika *user* meminta untuk menampilkan data, prosedur ini akan memberi perintah kepada *class* forum_komunikasi_control untuk mengambil data pada *database.*
Tambah_data() : void	Prosedur untuk melakukan tambah data. Memanggil forum_komunikasi_view untuk menampilkan form tambah.
Ubah_data() : void	Prosedur untuk memproses data masukan dari form ubah, memanggil forum_komunikasi_model untuk menyimpan data yang sudah diubah.
Hapus_data() : void	Prosedur untuk menghapus data materi pelaran, akan menampilkan pesan konfirmasi apakah data forum komunikasi akan dihapus, jika iya maka akan memanggil forum_komunikasi_model untuk melakukan penghapusan data pada *database*
Cari_data() : void	Prosedur untuk mencari data pada data forum komunikasi

36. *Class* forum komunikasi model

Class forum_ model merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* forum_ model dapat dilihat pada Gambar 3.98



Gambar 3.98 ***Class forum komunikasi model***

a. Deskripsi *attribut*

Class forum_model tidak memiliki atribut

b. Deskripsi operasi

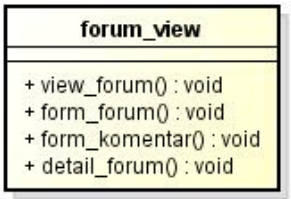
Deskripsi operasi class forum_ model dapat dilihat pada Tabel 3.68

Tabel 3.68 ***Deskripsi class forum komunikasi control***

Nama	Keterangan
Get_forum(): void	Prosedur mengambil forum komunikasi dari database, digunakan untuk menampilkan seluruh data forum komunikasi.
get_ forum_by_id():void	Prosedur mengambil data forum komunikasi berdasarkan id forum, digunakan untuk menampilkan data forum komunikasi berdasarkan id
Tambah () : void	Prosedur untuk melakukan tambah data topik. Memanggil forum_komunikasi _view untuk menampilkan form tambah.
Ubah ():void	Prosedur untuk memproses data masukan dari form ubah, memanggil forum_komunikasi_model untuk menyimpan data yang sudah diubah.
Hapus ():void	Prosedur untuk menghapus data, akan menampilkan pesan konfirmasi apakah data topik akan dihapus, jika iya maka akan memanggil forum_komunikasi_model untuk melakukan penghapusan data pada *database*

37. *Class* forum komunikasi view

Class forum_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* forum_view dapat dilihat pada Gambar 3.99



Gambar 3.99 ***Class forum komunikasi view***

a. Deskripsi *attribut*

Class forum_view tidak memiliki atribut

b. Deskripsi operasi

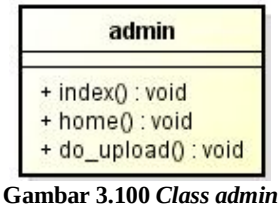
Deskripsi operasi forum_view dapat dilihat pada Tabel 3.69

Tabel 3.69 ***Deskripsi class forum komunikasi view***

Nama	Keterangan
forum_view(): void	Prosedur untuk menampilkan data kategori yang hasil prosedur dari *class* forum_komunikasi_model dan forum_komunikasi_control
forum_form() : void	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* forum_komunikasi_control

38. *Class* admin control

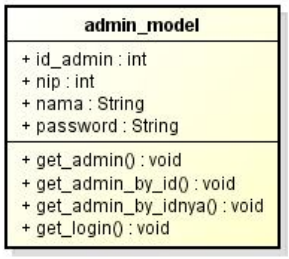
Class admin control merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class admin dapat di lihat pada Gambar 3.100



Gambar 3.100 ***Class admin***

39. Class admin model

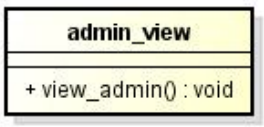
Class admin control merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class admin dapat di lihat pada Gambar 3.101



Gambar 3.101 ***class admin model***

40. Class admin view

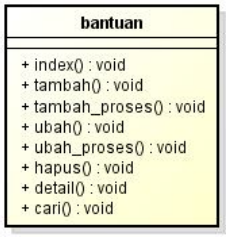
Class admin_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* admin_view dapat dilihat pada Gambar 3.102



Gambar 3.102 ***class admin view***

41. Class bantuan control

Class bantuan control merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class bantuan dapat di lihat pada Gambar 3.103



Gambar 3.103 ***class bantuan control***

42. Class bantuan model

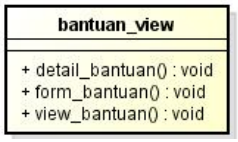
Class bantuan control merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class bantuan dapat di lihat pada Gambar 3.104



Gambar 3.104 *Class bantuan model*

43. Class bantuan view

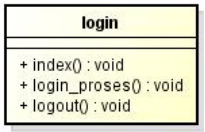
Class bantuan_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* bantuan_view dapat dilihat pada Gambar 3.105



Gambar 3.105 *Class bantuan view*

44. Class login control

Class login_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* login_control dapat dilihat pada Gambar 3.106

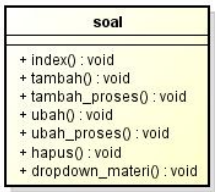


Gambar 3.106 *Class login control*

45. Class soal control

Class soal_control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk

memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* soal_control dapat dilihat pada Gambar 3.107



Gambar 3.107 *Class soal control*

46. Class soal model

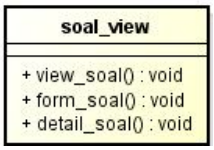
Class soal control merupakan *class* yang berisi atribut-atribut yang digunakan dalam proses pengolahan data yang dilakukan. Attribut dan operasi yang dimiliki class soal dapat di lihat pada Gambar 3.108



Gambar 3.108 *Class soal model*

47. Class soal view

Class bantuan_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* bantuan_view dapat dilihat pada Gambar 3.109

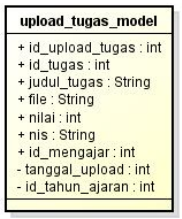


Gambar 3.109 *Class soal view*

48. Class upload tugas

Class bantuan_view merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data materi pelajaran dan bertugas untuk

memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* bantuan_view dapat dilihat pada Gambar 3.110



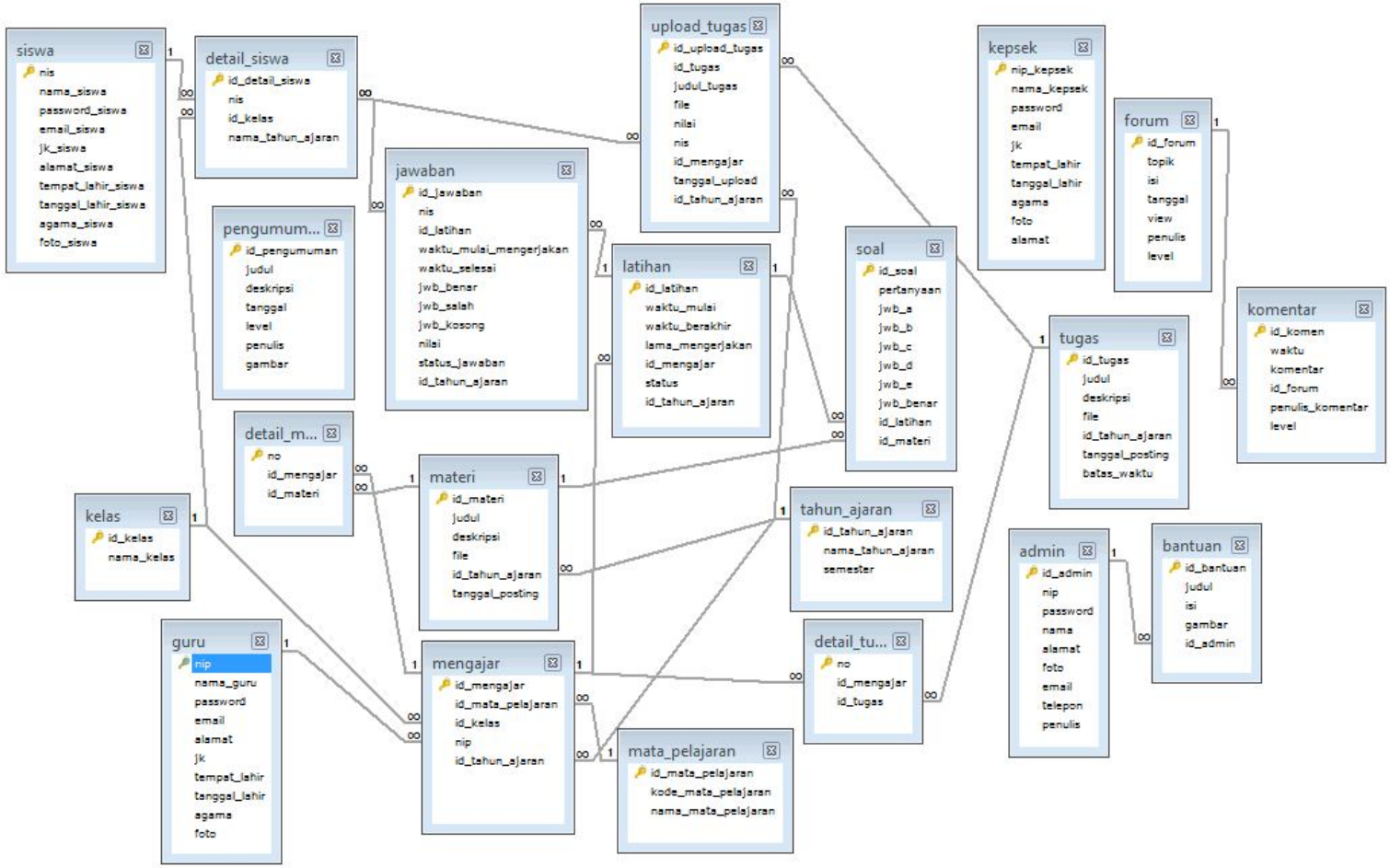
Gambar 3.110 ***Class upload tugas model***

3.3 Perancangan Sistem

Perancangan merupakan penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam suatu kesatuan yang utuh. Tahap ini meliputi konfigurasi dari komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah instalasi sistem akan mendapatkan hasil yang baik berdasarkan rancangan bangun yang telah ditetapkan pada tahap akhir analisis sistem.

3.3.1 Diagram Relasi

Diagram relasi merupakan rangkaian hubungan antar dua tabel atau lebih pada sistem basis data. Rangkain basis data pada *E-learning* SMK-PU Negeri Prov. Jawa Barat dapat dilihat pada gambar 3.111



Gambar 3.111 ***Diagram relasi***

3.3.2 Struktur Tabel

Struktur tabel adalah spesifikasi *file* dari setiap tabel yang digunakan dalam sistem yang akan dibangun. Struktur tabel dari sistem yang akan dibangun sebagai berikut :

a. Admin

Nama tabel : admin

Primary key : nip

Tabel 3.70 ***Struktur Tabel Admin***

Nama Field	Tipe	Ukuran	Keterangan
id_admin	Int	11	Primary key
nip	Varchar	20	
password	Varchar	100	
nama	Varchar	50	
alamat	Text		
foto	Varchar	100	
email	Varchar	50	
telepon	Varchar	20	

b. Bantuan

Nama tabel : bantuan

Primary key : id_bantuan

Tabel 3.71 ***Struktur Tabel Bantuan***

Nama Field	Tipe	Ukuran	Keterangan
id_bantuan	Int	11	Primary key
judul	Varchar	50	
isi	Text		
gambar	Varchar	110	
Id_admin	int	11	Foreign_key ke tabel admin

c. Mengajar

Nama tabel : mengajar

Primary key : id_mengajar

Tabel 3.72 ***Struktur Tabel Mengajar***

Nama Field	Tipe	Ukuran	Keterangan
Id_mengajar	Int	11	Primary key

Id_mata_pelajaran	Int	11	Foreign_key ke tabel mata_pelajaran
Id_kelas	Int	11	Foreign key ke tabel kelas
Nip	Varchar	20	Foreign key ke tabel guru
Id_tahun_ajaran	Int	11	Foreign key ke tabel tahun_ajaran

d. Detail materi

Nama tabel : detail_materi

Primary key : id_detail_materi

Tabel 3.73 ***Struktur Tabel Detail Materi***

Nama Field	Tipe	Ukuran	Keterangan
no	Int	11	Primary key
Id_mengajar	Int	11	Foreign key ke tabel mengajar
Id_materi	Int	11	Foreign key ke tabel materi

e. Detail tugas

Nama tabel : detail_tugas

Primary key : id_detail_tugas

Tabel 3.74 ***Struktur Tabel Detail Tugas***

Nama Field	Tipe	Ukuran	Keterangan
no	Int	11	Primary key
Id_mengajar	Int	11	Foreign key ke tabel mengajar
Id_tugas	Int	11	Foreign key ke tabel tugas

f. Jawaban

Nama tabel : jawaban

Primary key : id_jawaban

Tabel 3.75 ***Struktur Tabel Jawaban***

Nama Field	Tipe	Ukuran	Keterangan
Id_jawaban	Int	11	Primary key
nis	Varchar	20	Foreign key ke tabel siswa
Id_latihan	Int	11	Foreign key ke tabel latihan
Waktu_mulai	Date		
Waktu_selesai	Date		
Jwb_benar	Int	11	
Jwb_salah	Int	11	

Jwb_kosong	Int	11	
Nilai	Int	11	
Status	Varchar	50	
Id_tahun_ajaran	Int	11	Foreign_key ke tabel tahun_ajaran

g. Forum

Nama tabel : forum

Primary key : id_forum

Tabel 3.76 *Struktur Tabel Forum*

Nama Field	Tipe	Ukuran	Keterangan
Id_forum	Int	11	Primary key
Topik	Varchar	30	
Isi	Text		
Tanggal	Date		
penulis	Varchar	20	
View	Int	11	
Level	Varchar	20	

h. Guru

Nama 139able : guru

Primary key : nip

Tabel 3.77 *Struktur Tabel Guru*

Nama Field	Tipe	Ukuran	Keterangan
Nip	Varchar	20	Primary key
Nama_guru	Varchar	30	
Password	Varchar	100	
Email	Varchar	30	
Alamat	Varchar	100	
Jk	Varchar	15	
Tempat_lahir	Varchar	30	
Tanggal_lahir	Date		
Agama	Varchar	10	
Foto	Varchar	100	

i. Kelas

Nama tabel : kelas

Primary key : id_kelas

Tabel 3.78 ***Struktur Tabel Kelas***

Nama Field	Tipe	Ukuran	Keterangan
Id_kelas	Int	11	Primary key
Nama_kelas	Varchar	10	

j. Kepala sekolah

Nama tabel : kepsek

Primary key : nip_kepsek

Tabel 3.79 ***Struktur Tabel Kepala Sekolah***

Nama Field	Tipe	Ukuran	Keterangan
Nip_kepsek	Varchar	20	Primary key
Nama_kepsek	Varchar	30	
Password	Varchar	100	
Email	Varchar	30	
Alamat	Varchar	100	
Jk	Varchar	15	
Tempat_lahir	Varchar	30	
Tanggal_lahir	Date		
Agama	Varchar	10	
Foto_kepsek	Varchar	100	

k. Komentar

Nama tabel : komentar

Primary key : id_komen

Tabel 3.80 ***Struktur Tabel Komentar***

Nama Field	Tipe	Ukuran	Keterangan
Id_komen	Int	11	Primary key
Waktu	Date		
Komentar	Text		
Id_forum	Int	11	Foreign key ke tabel forum
Level	Varchar	50	
Penulis	Varchar	20	

l. Mata pelajaran

Nama tabel : mata_pelajaran

Primary key : id_mata_pelajaran

Tabel 3.81 ***Struktur Tabel Mata Pelajaran***

Nama Field	Tipe	Ukuran	Keterangan
Id_mata_pelajaran	Int	11	Primary key
Kode_mata_pelajaran	Varchar	20	
Nama_mata_pelajaran	Varchar	30	

m. Materi pelajaran

Nama tabel : materi_pelajaran

Primary key : id_materi_pelajaran

Tabel 3.82 ***Struktur Tabel Materi Pelajaran***

Nama Field	Tipe	Ukuran	Keterangan
Id_materi	Int	11	Primary key
Judul	Varchar	30	
Deskripsi	Text		
File	Varchar	150	
Id_tahun_ajaran	Int	11	Foreign key ke tabel tahun_ajaran
Tanggal_posting	Date		

n. Pengumuman

Nama tabel : pengumuman

Primary key : id_pengumuman

Tabel 3.83 ***Struktur Tabel Pengumuman***

Nama Field	Tipe	Ukuran	Keterangan
Id_pengumuman	Int	11	Primary key
Judul	Varchar	100	
Deskripsi	Text		
Tanggal	Date		
Penulis	Varchar	20	
level	Varchar	20	

o. Siswa

Nama tabel : siswa

Primary key : nis

Tabel 3.84 ***Struktur Tabel Siswa***

Nama Field	Tipe	Ukuran	Keterangan
Nis	Varchar	20	Primary key

Nama_siswa	Varchar	50	
Password_siswa	Varchar	100	
Jk_siswa	Varchar	20	
Alamat_siswa	Varchar	100	
Email_siswa	Varchar	30	
Tempat_lahir_siswa	Varchar	30	
Tanggal_lahir_siswa	Date		
Foto_siswa	Varchar	100	
Agama_siswa	Varchar	20	
Id_kelas	Int	11	Foreign key ke tabel kelas

p. Soal

Nama tabel : soal

Primary key : id_soal

Tabel 3.85 *Struktur Tabel Soal*

Nama Field	Tipe	Ukuran	Keterangan
Id_soal	Int	11	Primary key
No_soal	Int	11	
Pertanyaan	Varchar	150	
Jwb_a	Varchar	150	
Jwb_b	Varchar	150	
Jwb_c	Varchar	150	
Jwb_d	Varchar	150	
Jwb_e	Varchar	150	
Jwb_benar	Varchar	20	
Id_latihan	Int	11	Foreign key ke tabel latihan
Id_materi	Int	11	Foreign key ke tabel materi
Status	Varchar	10	

q. Tahun ajaran

Nama tabel : tahun_ajaran

Primary key : id_tahun_ajaran

***Tabel 3.86* Struktur Tabel Tahun Ajaran**

Nama Field	Tipe	Ukuran	Keterangan
Id_tahun_ajaran	Int	11	Primary key
Nama_tahun_ajaran	Varchar	20	
Semester	Varchar	10	

r. Tugas

Nama tabel : tugas

Primary key : id_tugas

Tabel 3.87 **Struktur Tabel Tugas**

Nama Field	Tipe	Ukuran	Keterangan
Id_tugas	Int	11	Primary key
Judul	Varchar	30	
Deskripsi	Text		
File	Varchar	150	
Id_tahun_ajaran	Int	11	
Tanggal_posting	Date		

s. Latihan

Nama tabel : latihan

Primary key : id_latihan

Tabel 3.88 ***Struktur Tabel Latihan***

Nama Field	Tipe	Ukuran	Keterangan
Id_latihan	Int	11	Primary key
Waktu_mulai	Date		
Waktu_berakhir	Date		
Lama_mengerjakan	Date		
Id_detail_mp	Int	11	Foreign key ke tabel mengajar
Status	Varchar	50	
Id_tahun_ajaran	Int	11	

t. Upload

Nama tabel : upload_tugas

Primary key : id_upload_tugas

Nama Field	Tipe	Ukuran	Keterangan
id_upload_tugas	int	11	Primary key
judul_tugas	date		
file	date		
nilai	date		
id_mengajar	int	11	Foreign key ke tabel mengajar
nis	varchar	50	Foreign_key ke tabel siswa
id_tahun_ajaran	int	11	Foreign_key ke tabel tahun_ajaran
tanggal_upload	date		

u. Detail siswa

Nama tabel : detail_siswa

Primary key : id_detail_siswa

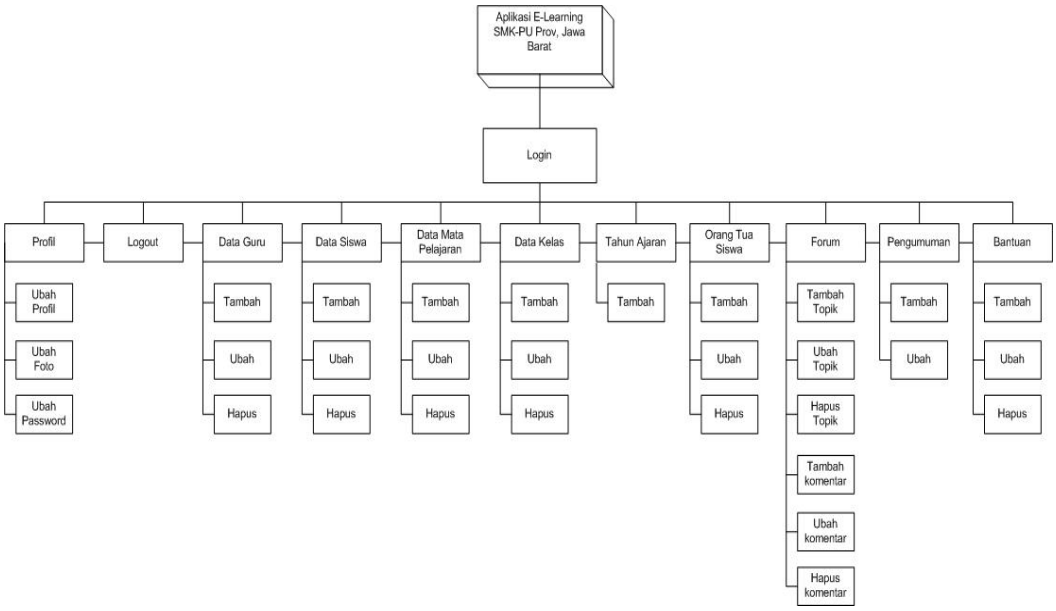
Nama Field	Tipe	Ukuran	Keterangan
id_detail_siswa	int	11	Primary key
Nis	vharchar	20	Foreign key ke tabel siswa
id_kelas	Int	11	Foreign key ke tabel kelas
nama_tahun_ajaran	Varchar		Foreign key ke tabel tahun_ajaran

3.3.3 Arsitektur Menu

Menu dirancang untuk menggambarkan segala macam bentuk kegiatan yang bisa dilakukan dalam program aplikasi ini, seperti *input* data, cari data, dan membuat laporan berdasarkan data *input.* Struktur menu yang dirancang sesuai dengan level pengguna sistem. Terdapat lima pengguna pada aplikasi *e-learning* pada SMK-PU Negeri Prov. Jawa Barat. Adapun struktur menu dari setiap level pengguna adalah sebagai berikut :

1. Perancangan struktur menu admin.

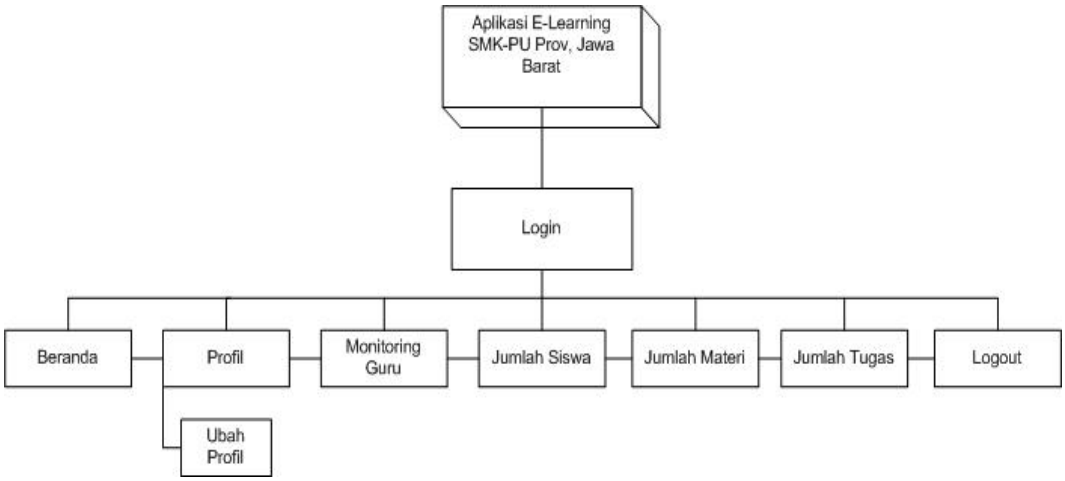
   Perancangan struktur menu admin dapat dilihat pada Gambar 3.112.



Gambar 3.112 ***Perancangan Struktur Menu Admin***

2. Perancangan struktur menu kepala sekolah.

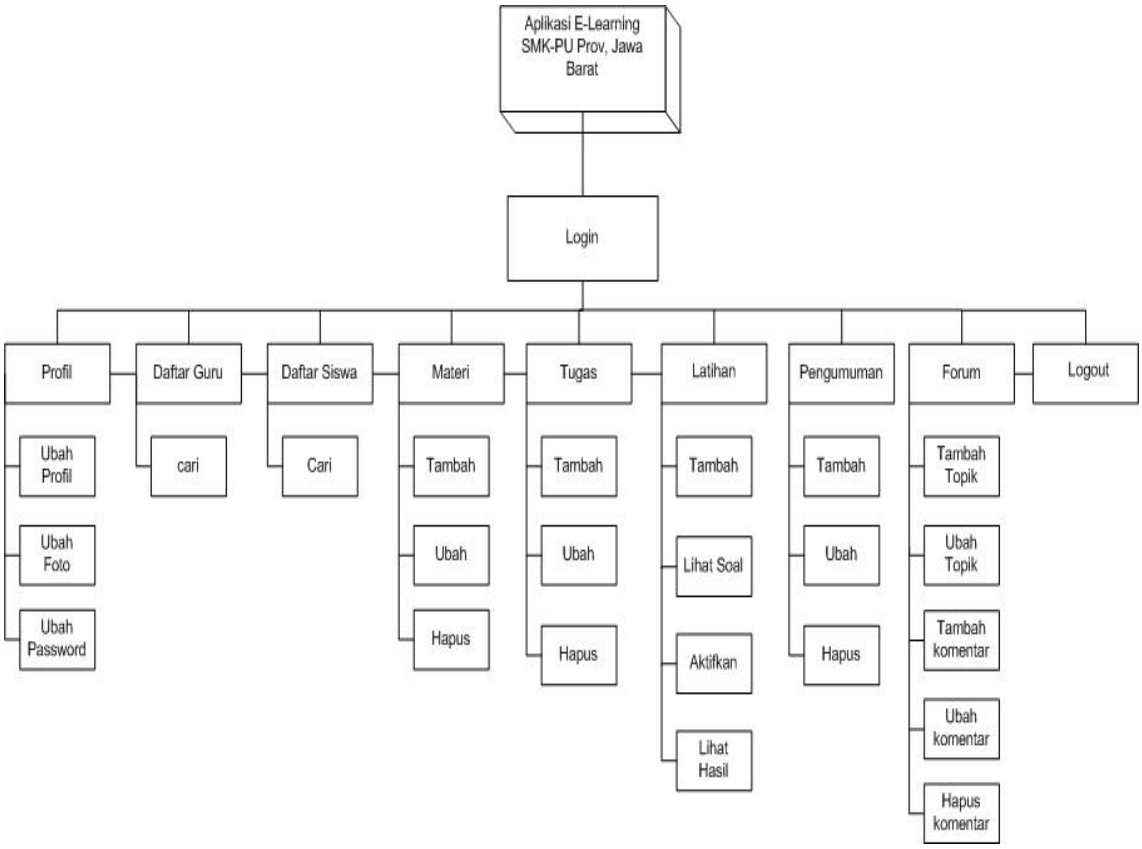
   Perancangan struktur menu admin dapat dilihat pada Gambar 3.112.



Gambar 3.113 ***Perancangan Struktur Menu Kepala Sekolah***

3. Perancangan struktur menu guru.

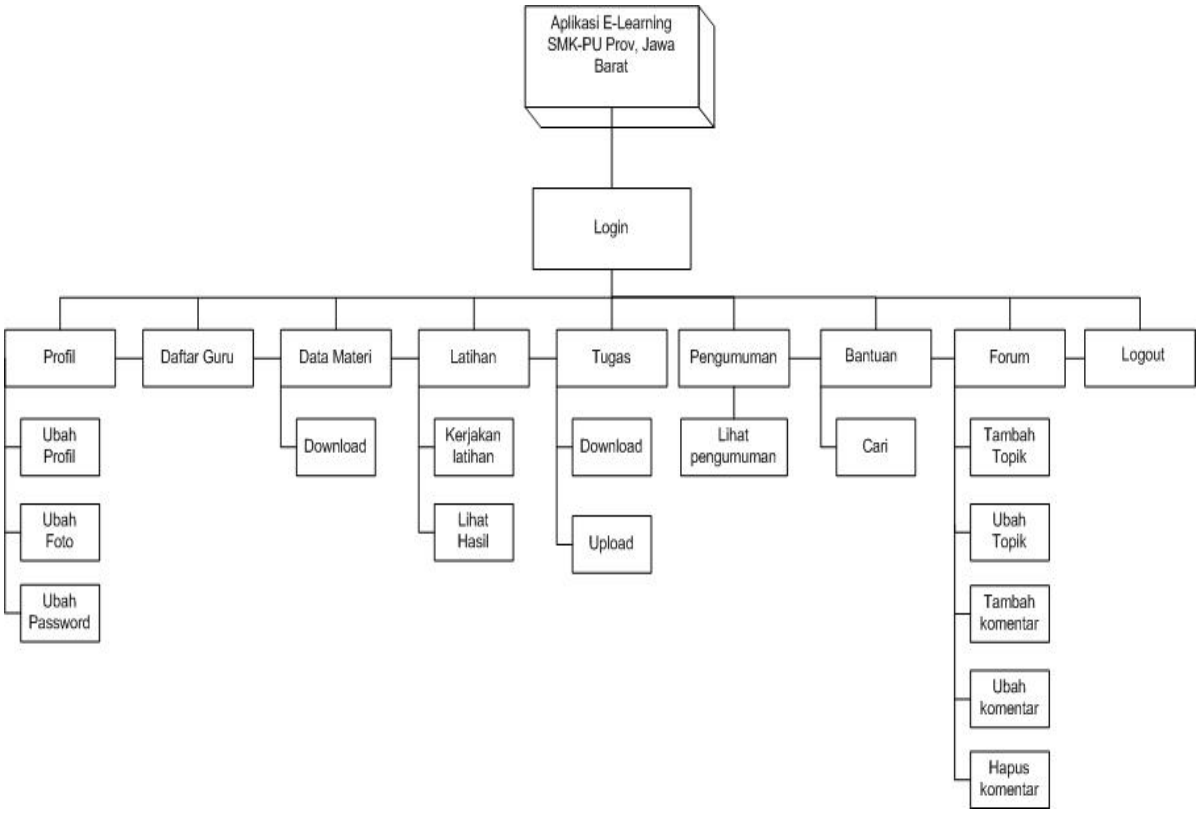
   Perancangan struktur menu admin dapat dilihat pada Gambar 3.114.



Gambar 3.114 ***Perancangan Struktur Menu Guru***

4. Perancangan struktur menu siswa.

   Perancangan struktur menu admin dapat dilihat pada Gambar 3.115.

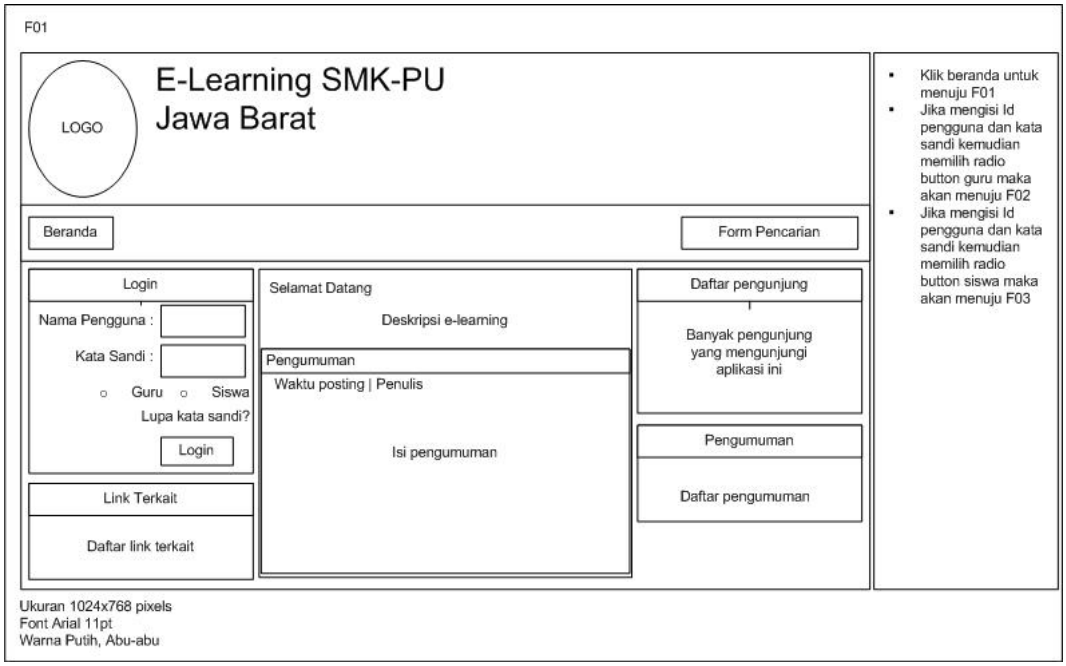


Gambar 3.115 ***Perancangan Struktur Menu Siswa***

3.3.4 Perancangan Antarmuka

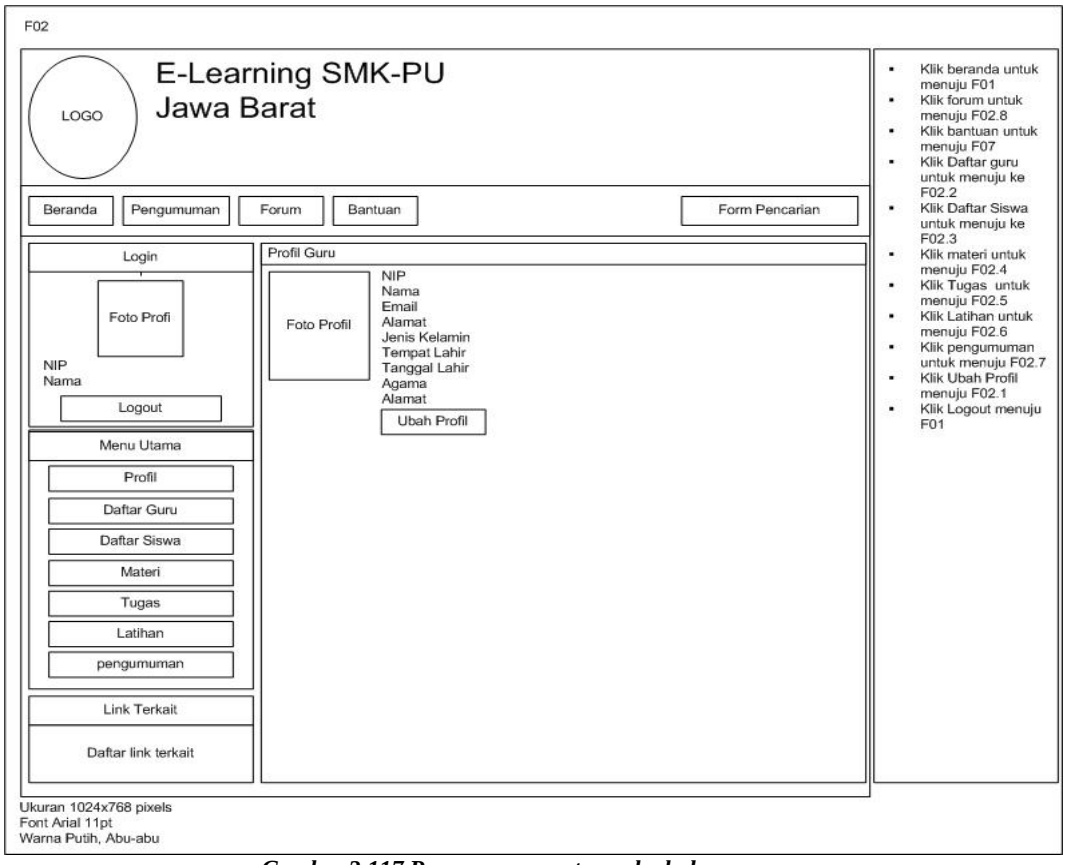
Perancangan antarmuka diperlukan pada program aplikasi ini dengan tujuan untuk mempermudah pengguna dalam menggunakan program aplikasi ini. Dengan adanya perancangan antarmuka ini berbagai pengguna baik awam maupun yang sudah berpengalaman dapan mengoperasikan program ini tanpa adanya kesulitan yang besar.

a. Perancangan halaman utama aplikasi *e-learning* SMK-PU Prov. Jawa Barat dapat dilihat pada Gambar 3.116



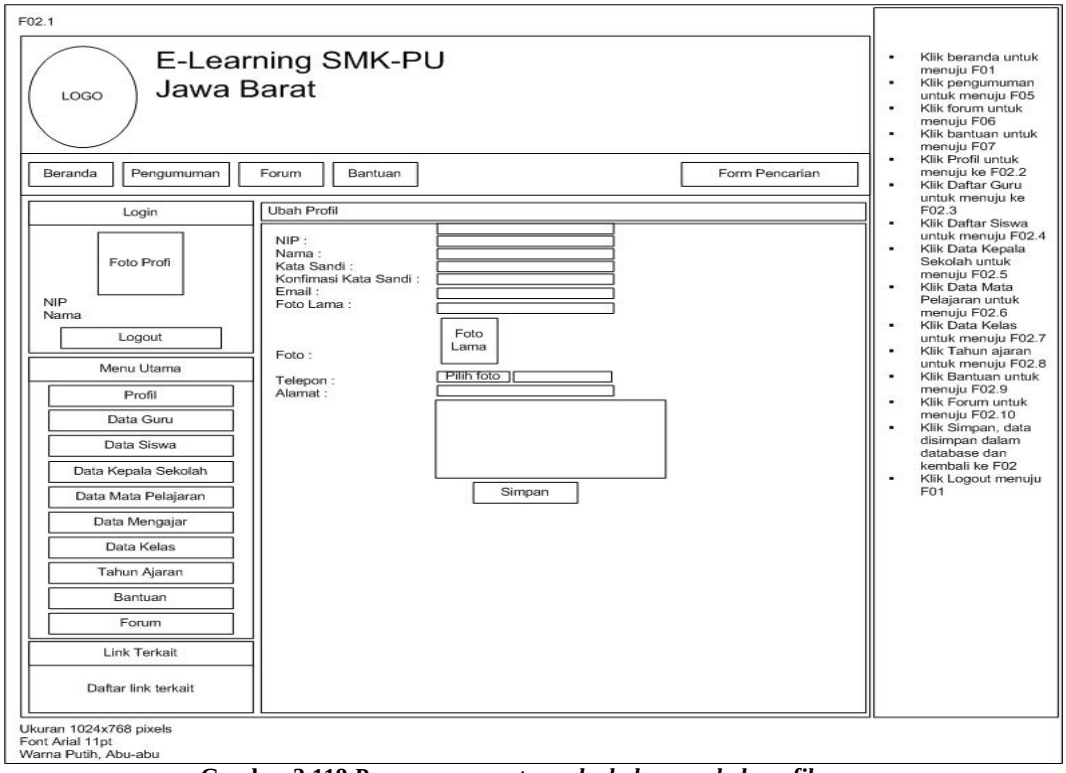
Gambar 3.116 ***Perancangan antarmuka halaman utama***

b. Perancangan halaman guru dapat dilihat pada Gambar 3.117



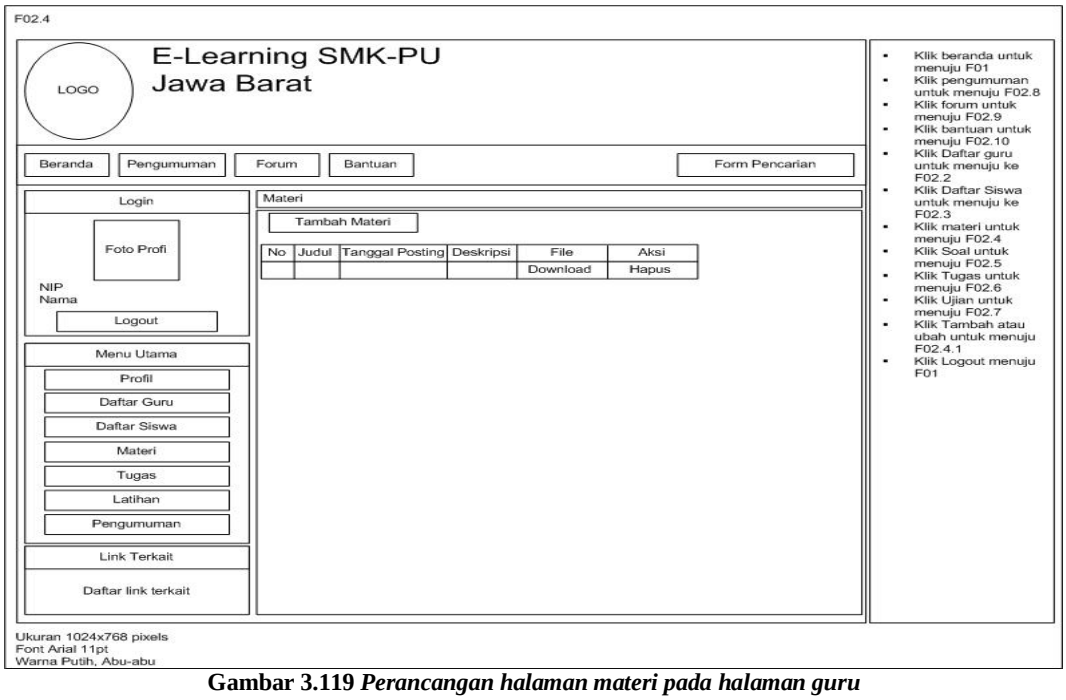
Gambar 3.117 Perancangan antarmuka halaman guru

c. Perancangan halaman ubah profil guru dapat dilihat pada Gambar 3.118



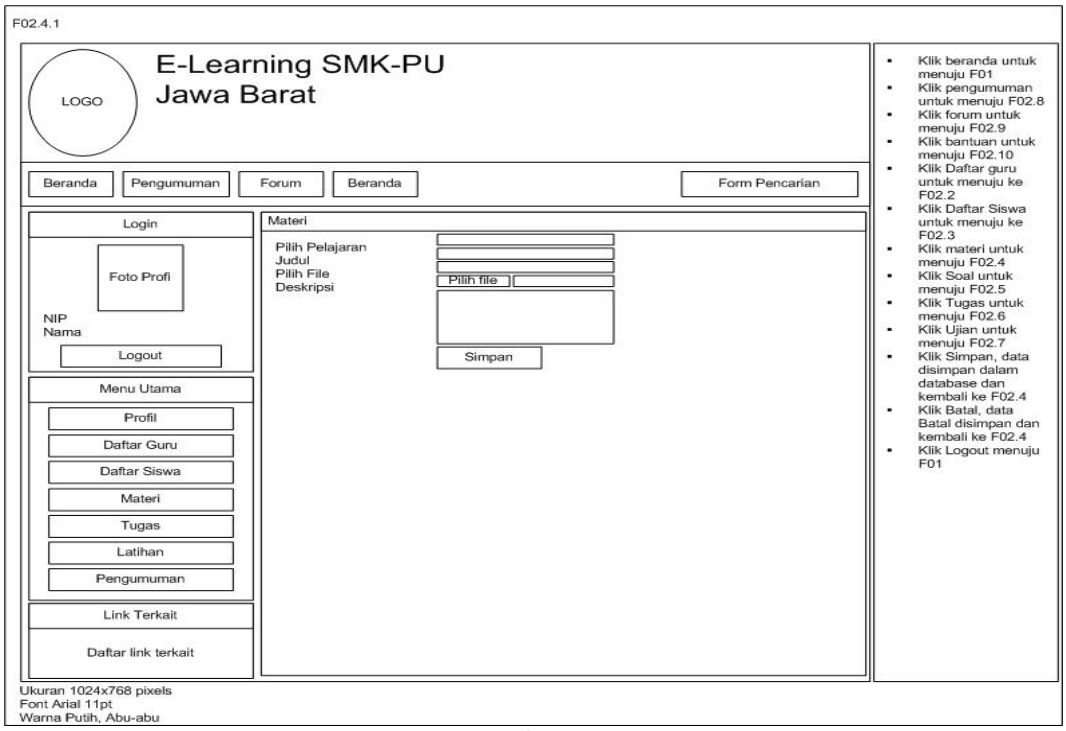
Gambar 3.118 ***Perancangan antarmuka halaman ubah profil guru***

d. Perancangan halaman materi pada halaman guru dapat dilihat pada Gambar 3.119



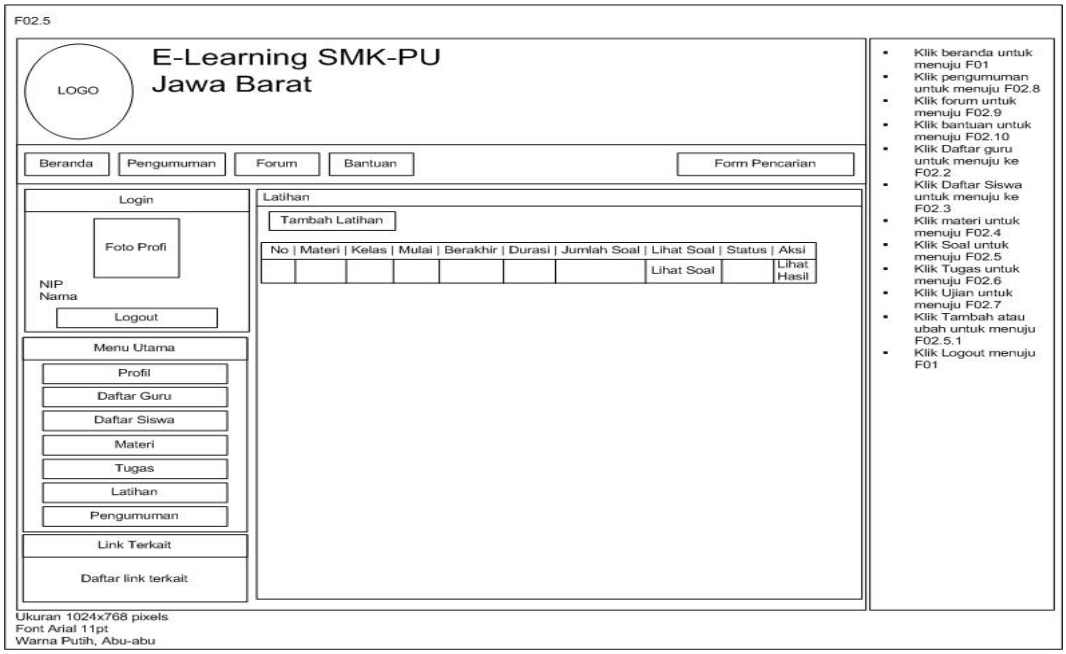
Gambar 3.119 ***Perancangan halaman materi pada halaman guru***

e. Perancangan halaman tambah dan ubah materi pada halaman guru dapat dilihat pada Gambar 3.120



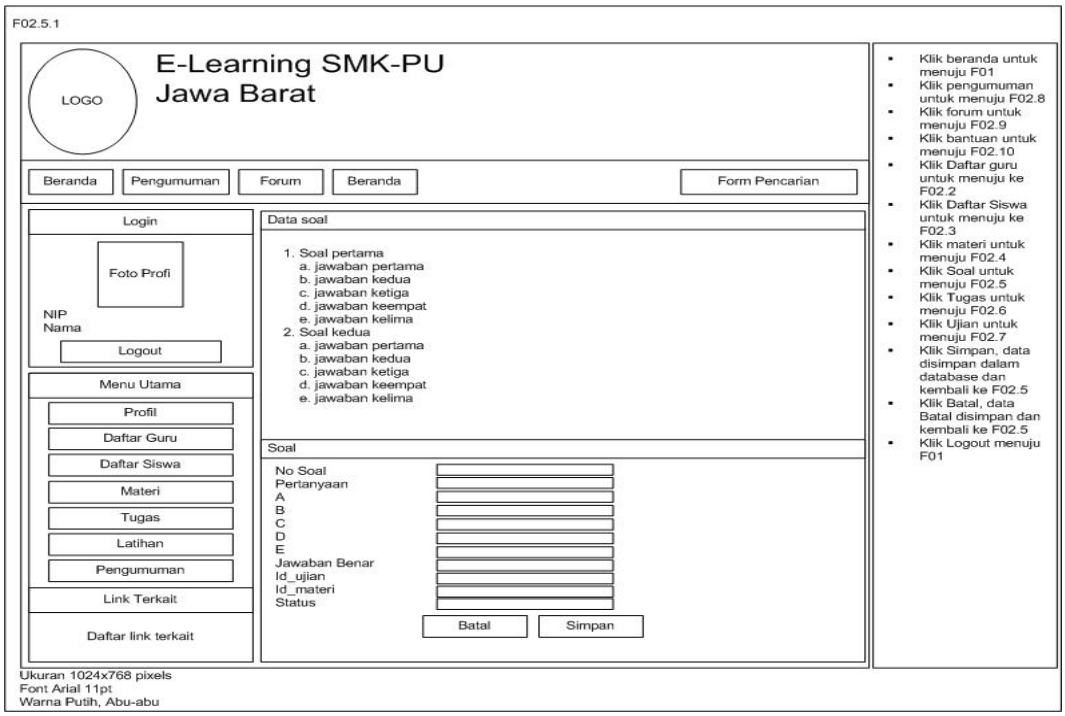
Gambar 3.120 ***Perancangan form materi pada halaman guru***

f. Perancangan halaman latihan pada halaman guru dapat dilihat pada Gambar 3.121



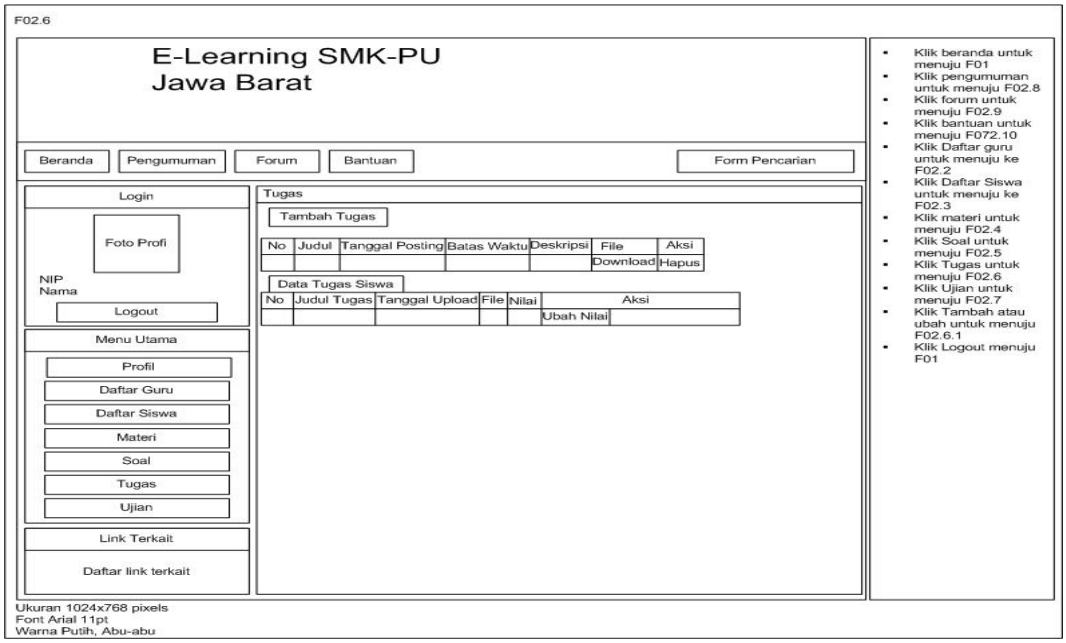
Gambar 3.121 ***Perancangan halaman latihan pada halaman guru***

g. Perancangan halaman form soal pada halaman guru dapat dilihat pada Gambar 3.122



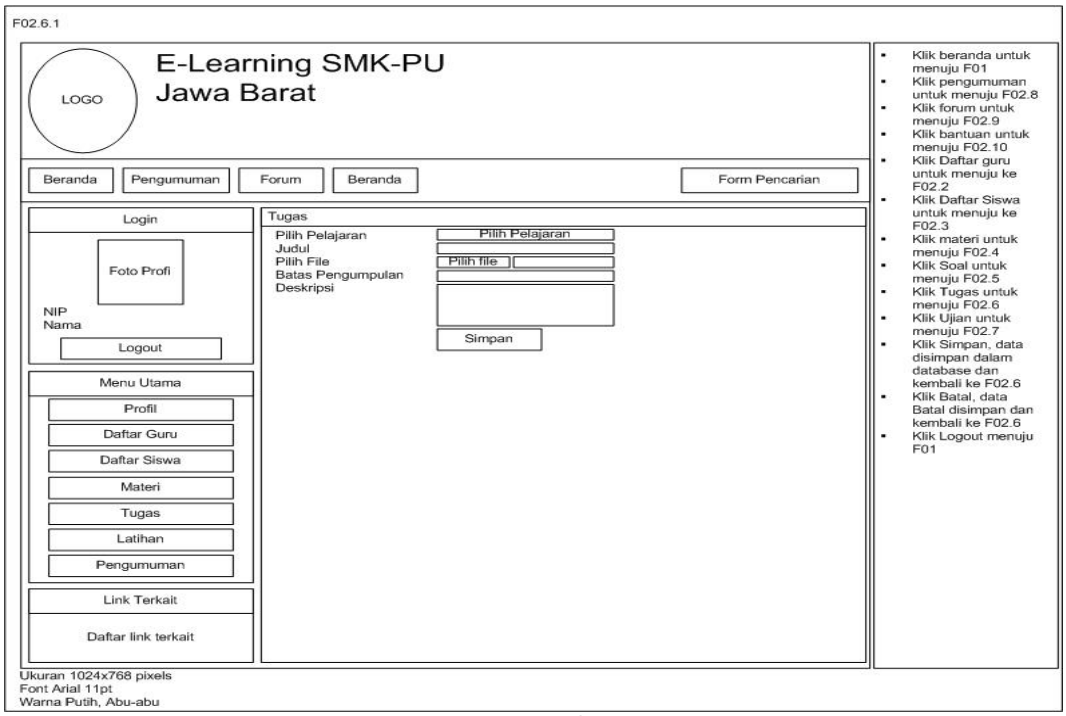
Gambar 3.122 ***Form soal pada halaman guru***

h. Perancangan halaman tugas pada halaman guru dapat dilihat pada Gambar 3.123



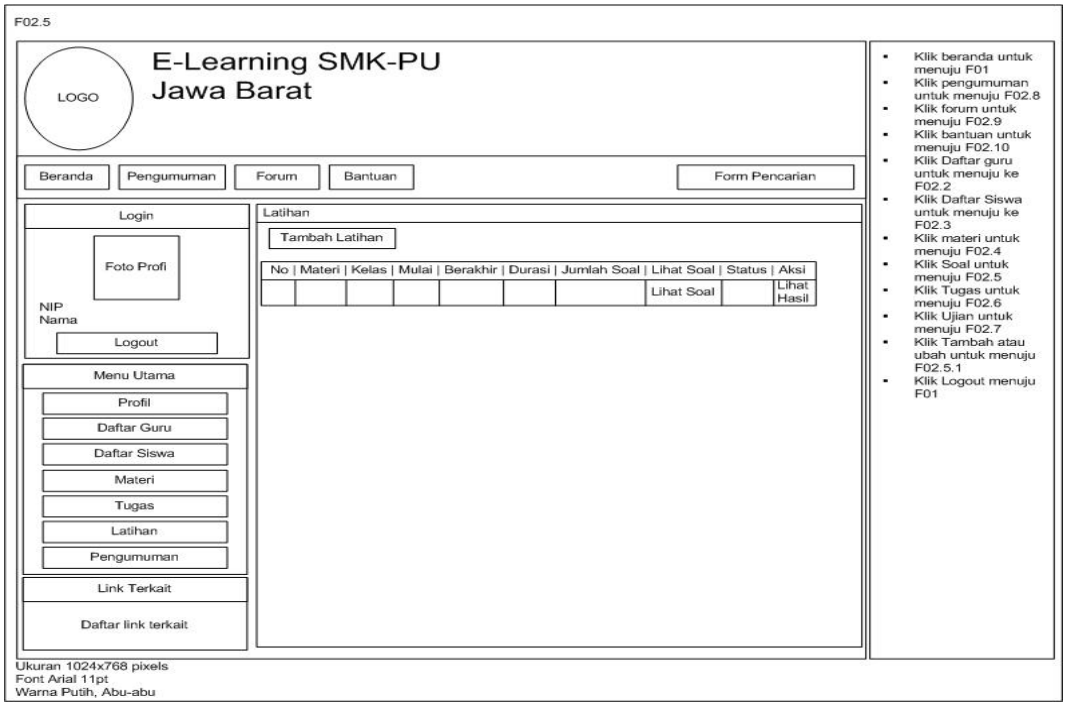
Gambar 3.123 ***Perancangan halaman tugas pada halaman guru***

i. Perancangan halaman form tugas pada halaman guru dapat dilihat pada Gambar 3.124



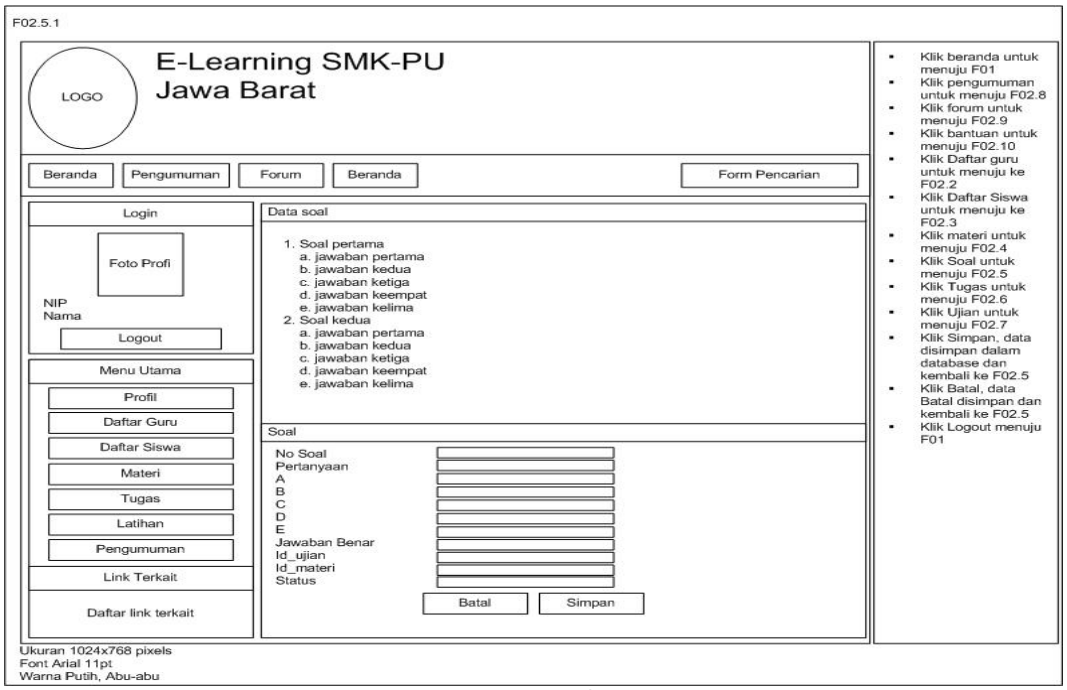
Gambar 3.124 ***Perancangan halaman form tugas pada halaman guru***

j. Perancangan halaman latihan pada halaman guru dapat dilihat pada Gambar 3.125



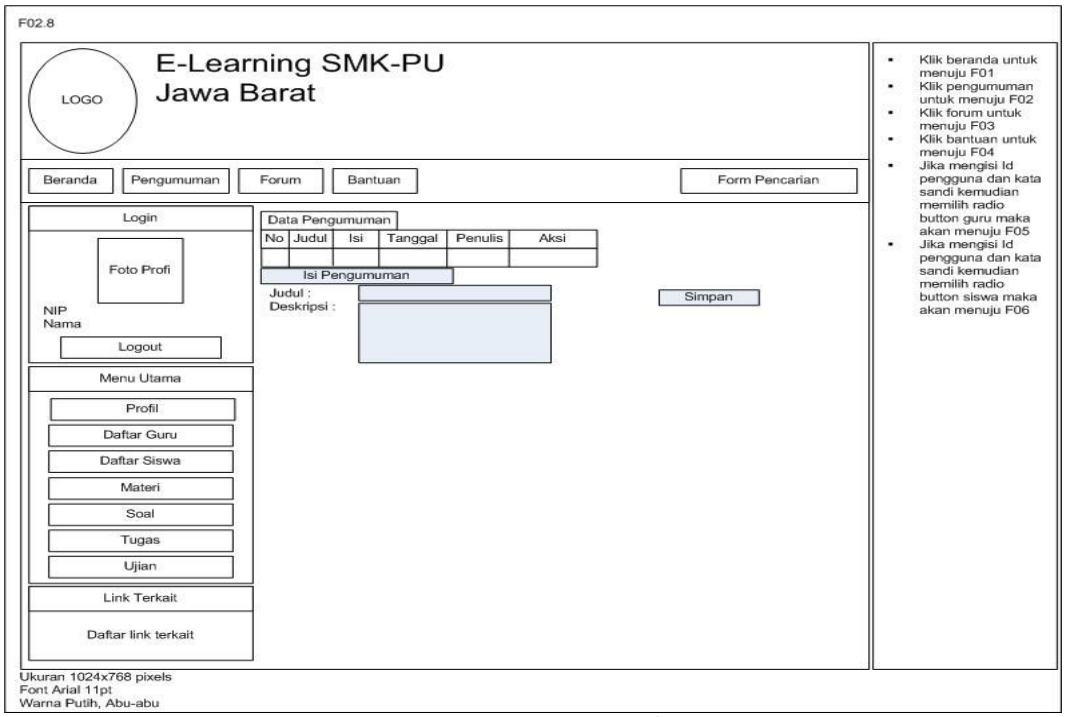
Gambar 3.125 ***Perancangan halaman latihan pada halaman guru***

k. Perancangan halaman form soal pada halaman guru dapat dilihat pada Gambar 3.126



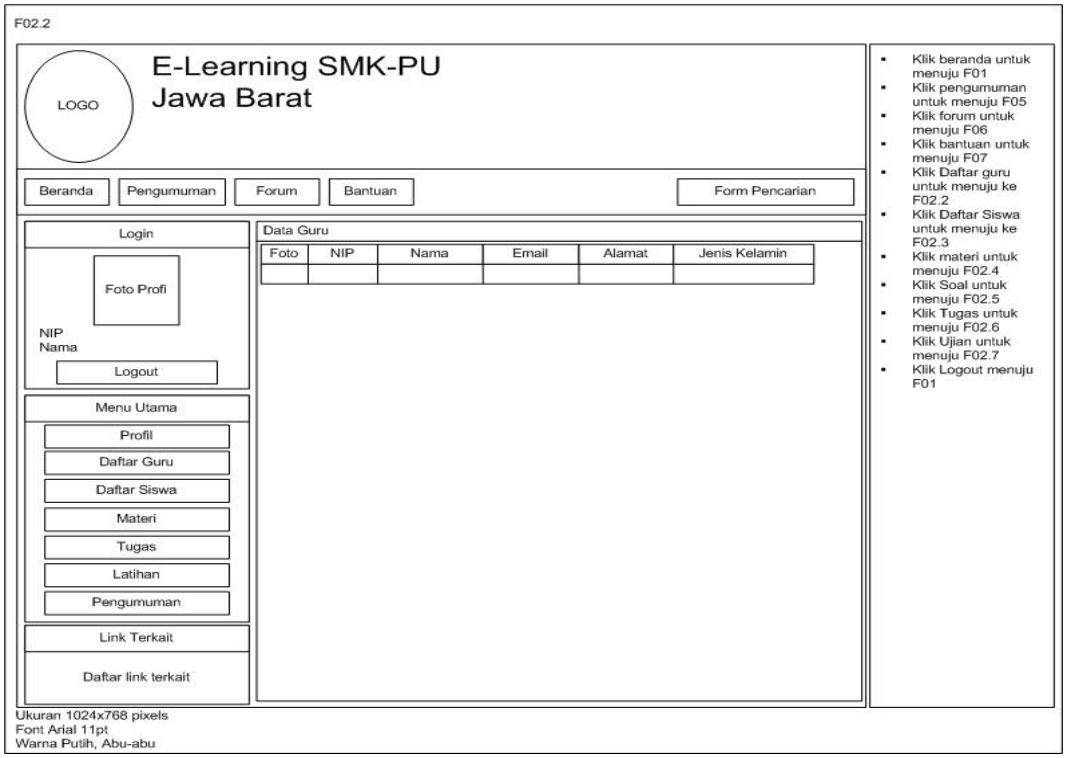
Gambar 3.126 ***Perancangan halaman form soal pada halaman guru***

l. Perancangan halaman daftar pengumuman pada halaman guru dapat dilihat pada Gambar 3.127



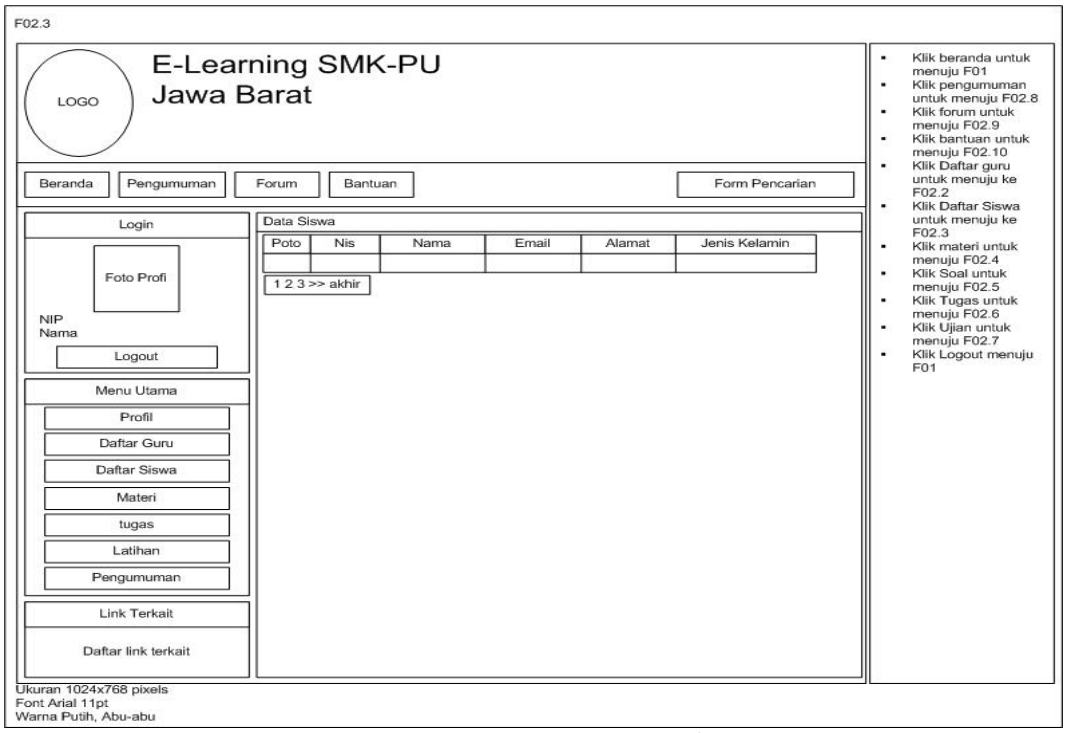
Gambar 3.127 ***Perancangan halaman daftar pengumuman***

m. Perancangan halaman daftar guru dapat dilihat pada Gambar 3.128



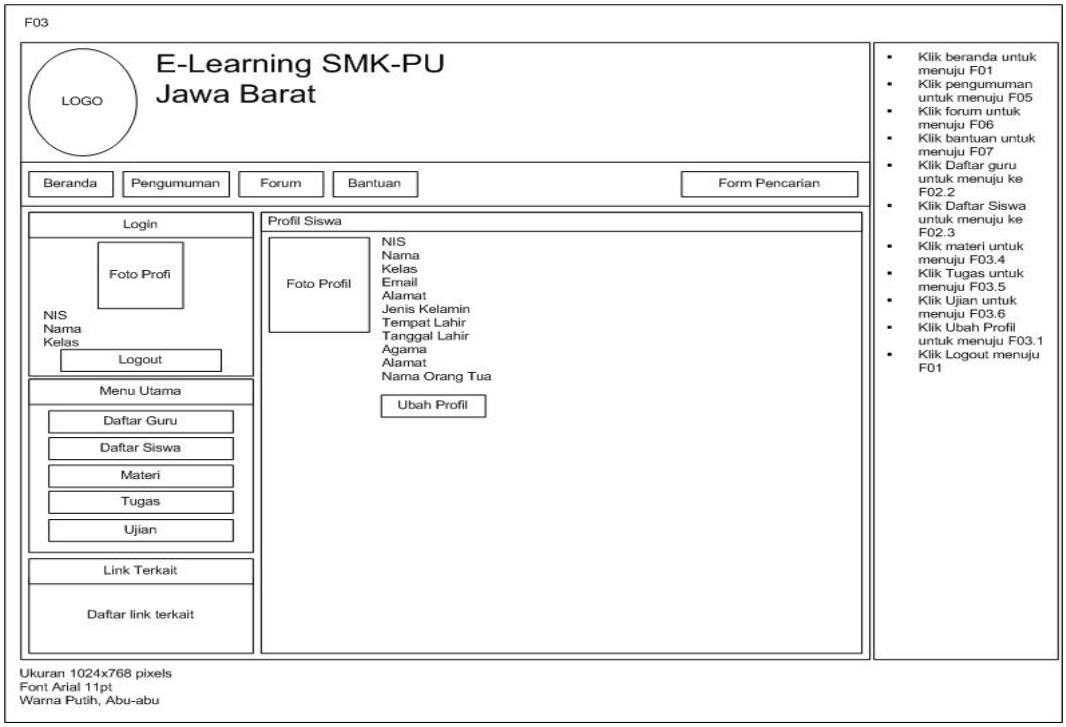
Gambar 3.128 ***Perancangan halaman daftar guru***

n. Perancangan halaman daftar siswa dapat dilihat pada Gambar 3.129



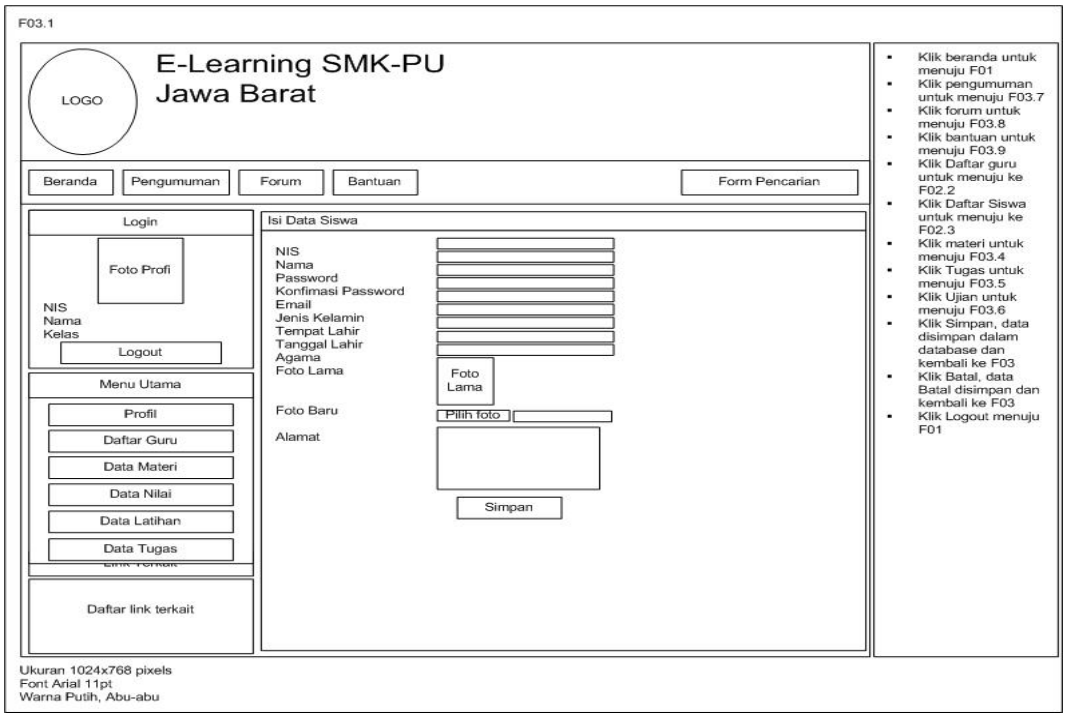
Gambar 3.129 ***Perancangan halaman daftar siswa***

o. Perancangan halaman siswa dapat dilihat pada Gambar 3.130



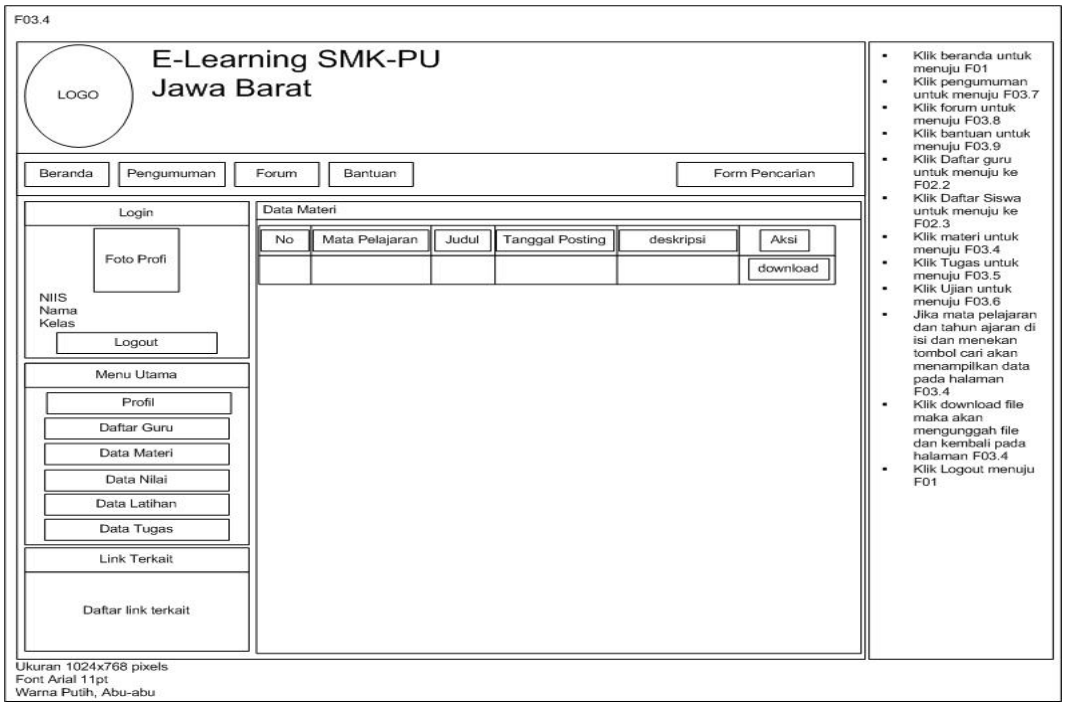
Gambar 3.130 ***Perancangan halaman siswa***

p. Perancangan halaman ubah profil siswa dapat dilihat pada Gambar 3.131



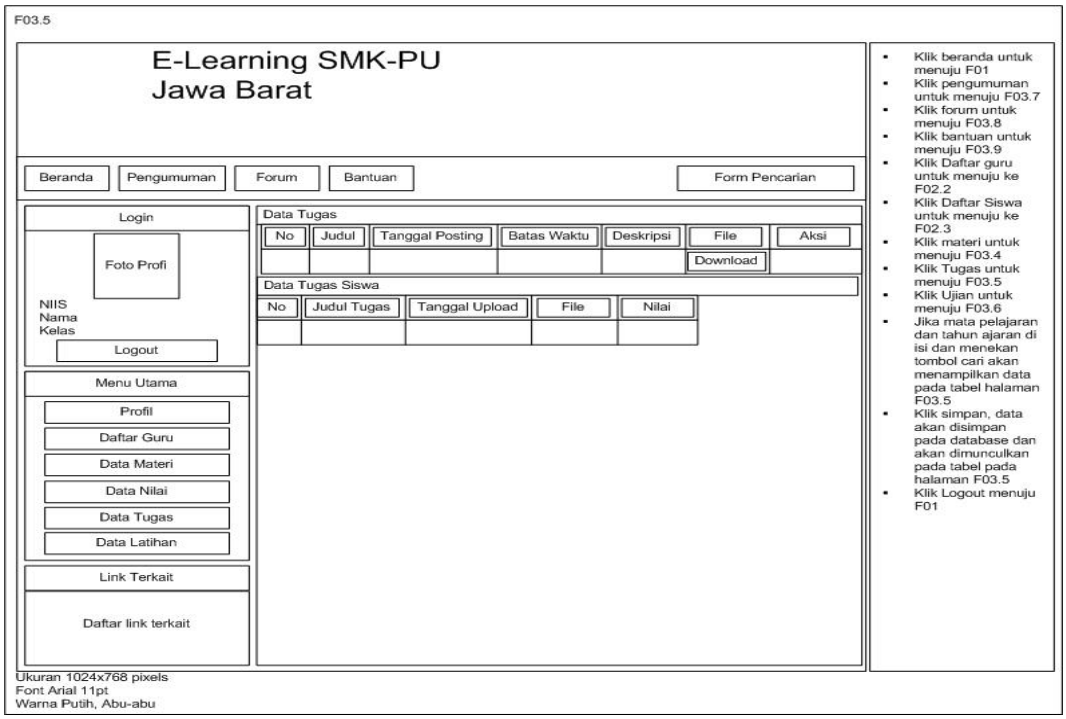
Gambar 3.131 ***Perancangan halaman ubah profil siswa***

q. Perancangan halaman materi pada halaman siswa dapat dilihat pada Gambar 3.132



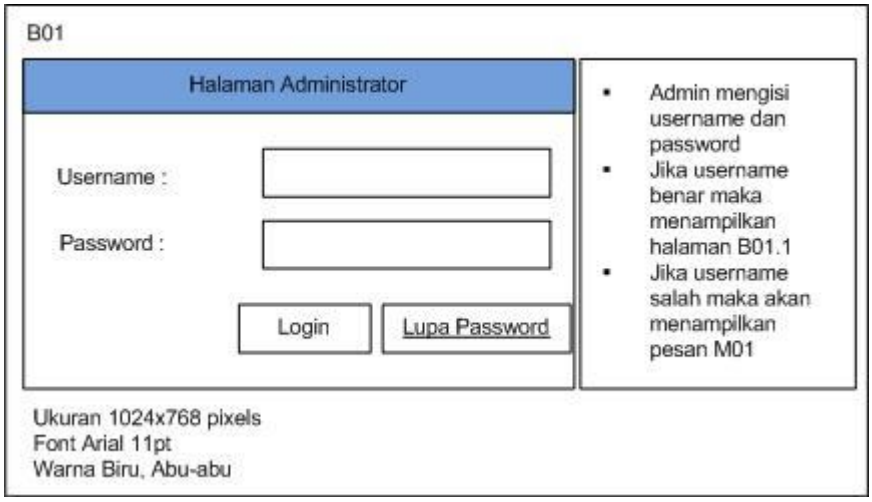
Gambar 3.132 ***Perancangan halaman materi pada halaman siswa***

r. Perancangan halaman tugas pada halaman siswa dapat dilihat pada Gambar 3.133



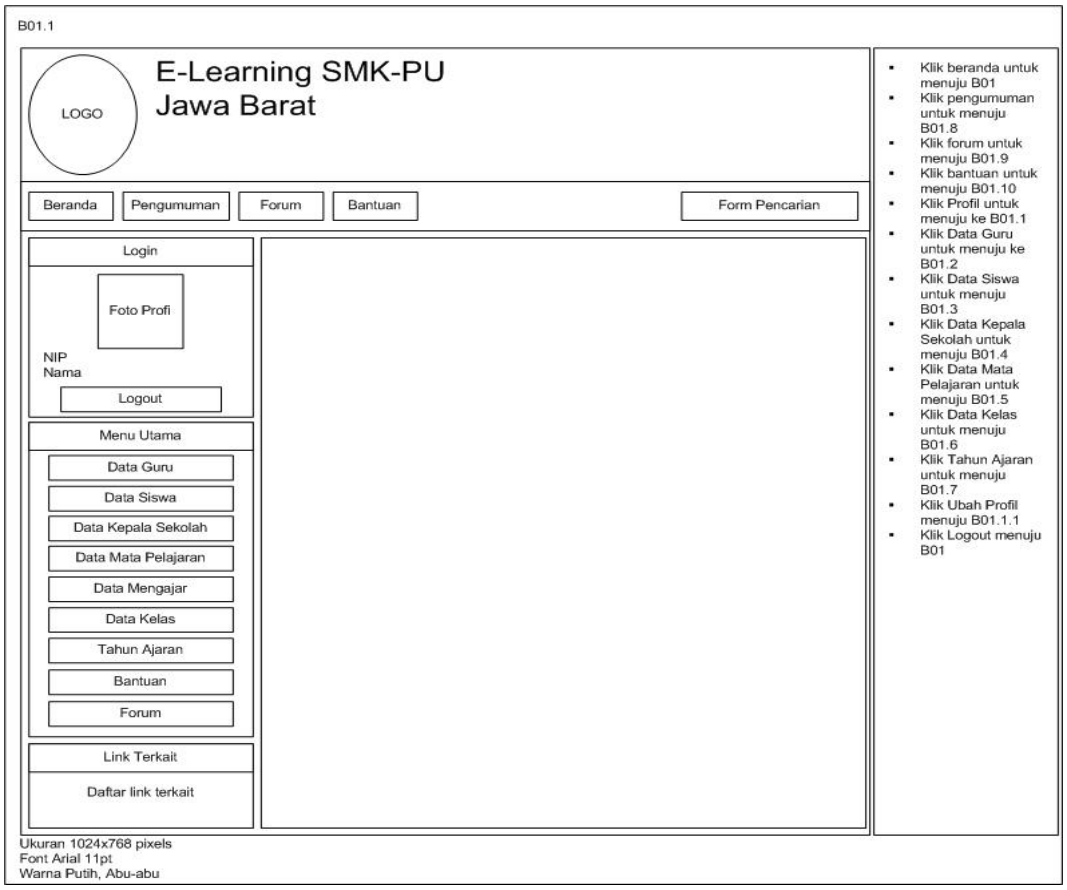
Gambar 3.133 ***Perancangan halaman tugas pada halaman siswa***

s. Perancangan halaman *back* dapat dilihat pada Gambar 3.134



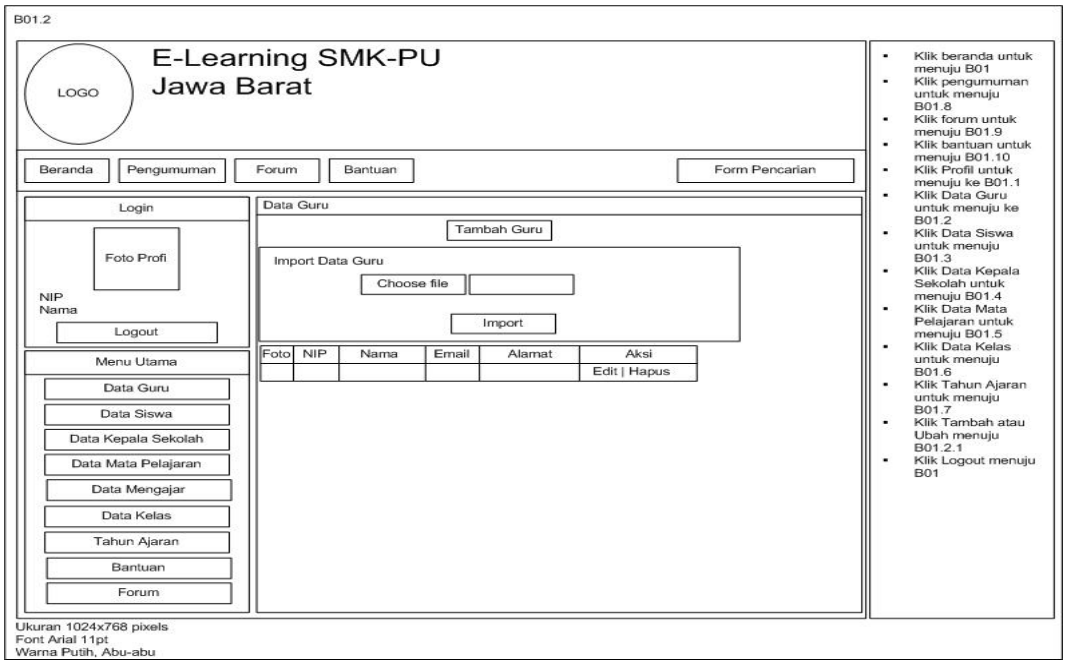
Gambar 3.134 ***Perancangan halaman back***

t. Perancangan halaman admin dapat dilihat pada Gambar 3.135



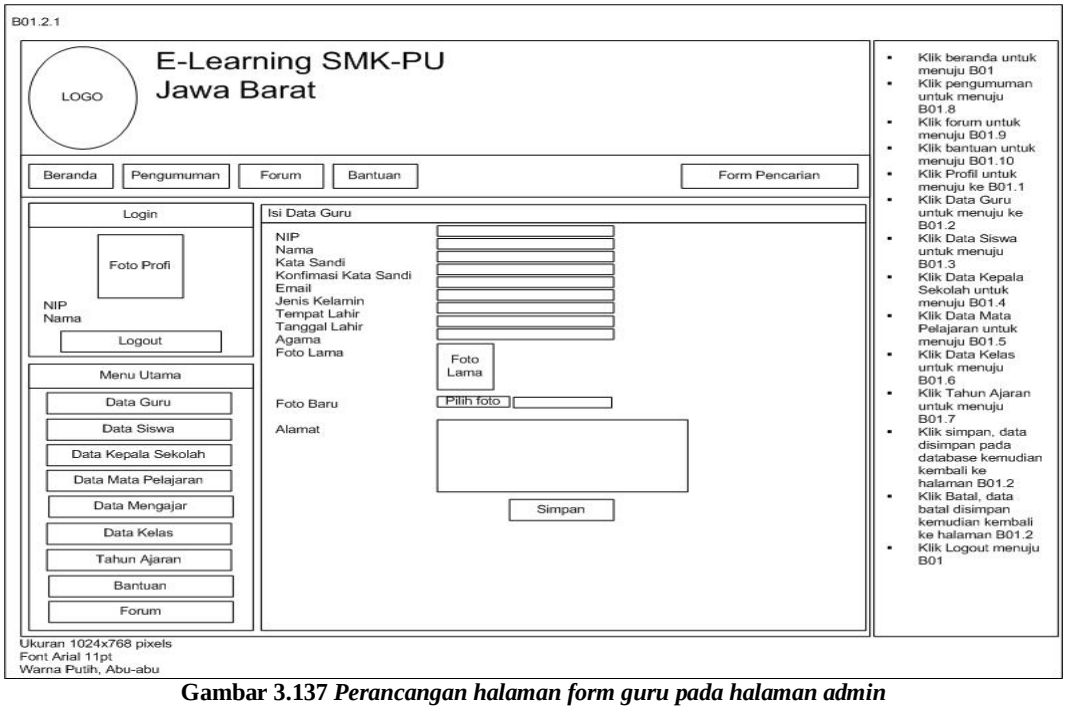
Gambar 3.135 ***Perancangan halaman Admin***

u. Perancangan halaman data guru pada halaman admin dapat dilihat pada Gambar 3.136



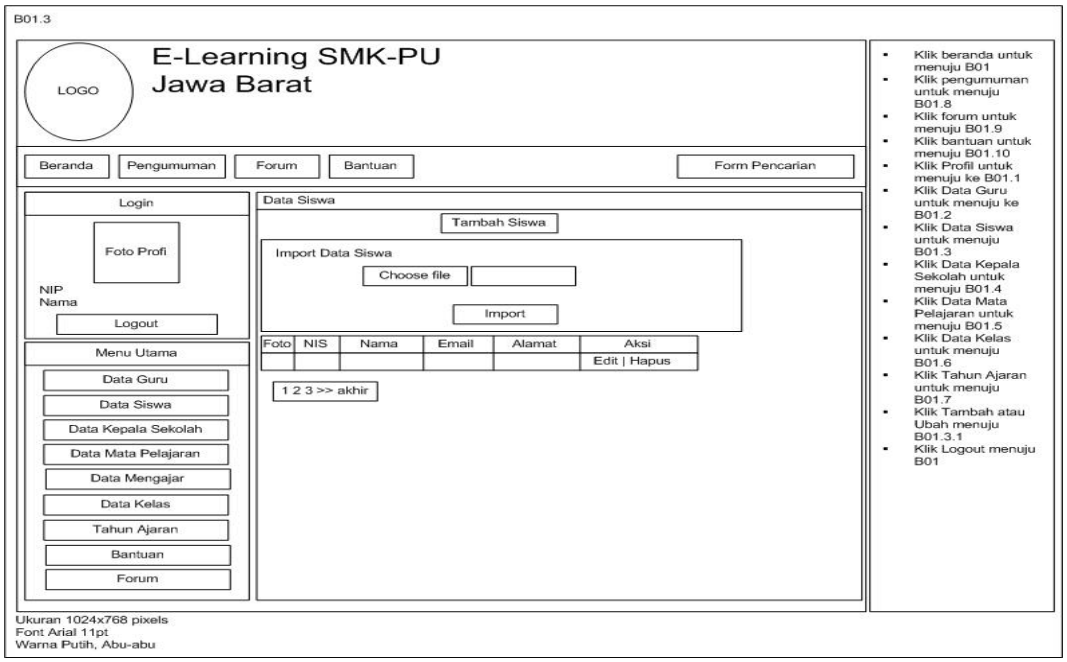
Gambar 3.136 ***Perancangan halaman data guru pada halaman admin***

v. Perancangan halaman form guru pada halaman admin dapat dilihat pada Gambar 3.137



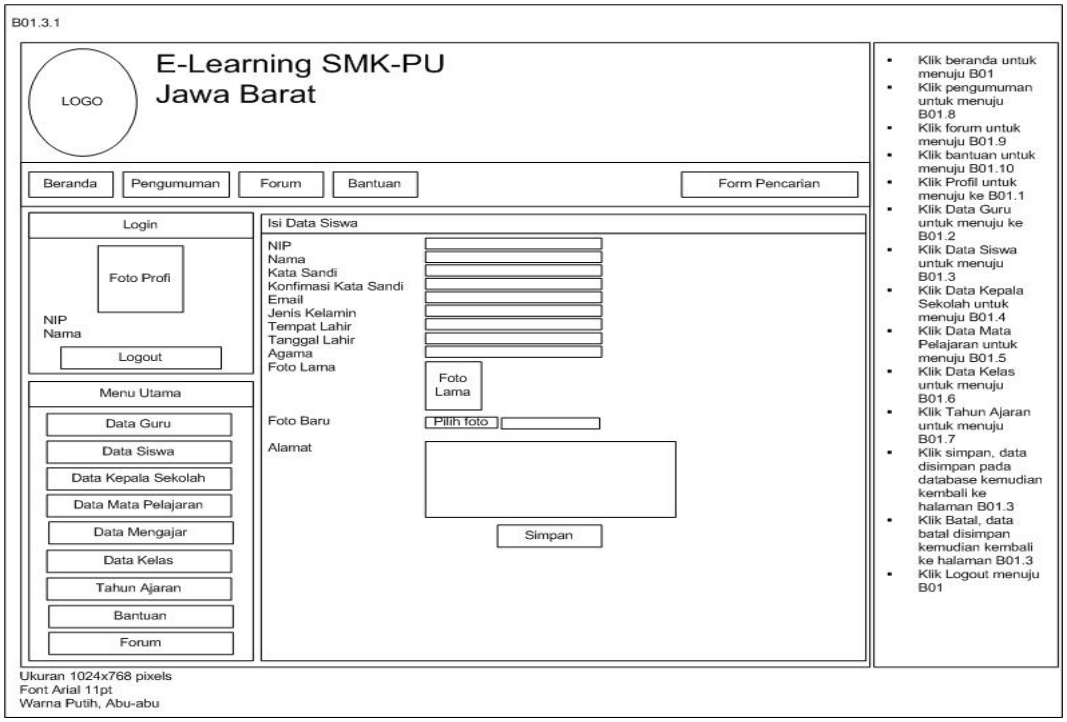
Gambar 3.137 ***Perancangan halaman form guru pada halaman admin***

w. Perancangan halaman data siswa pada halaman admin dapat dilihat pada Gambar 3.138



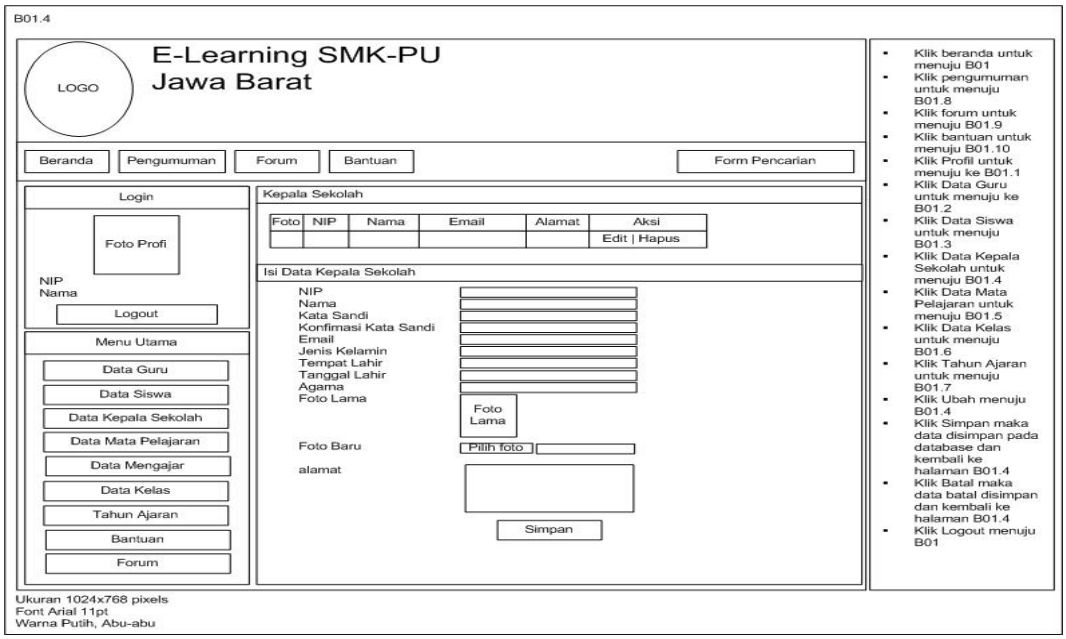
Gambar 3.138 ***Perancangan halaman data siswa pada halaman admin***

x. Perancangan halaman form siswa pada halaman admin dapat dilihat pada Gambar 3.139



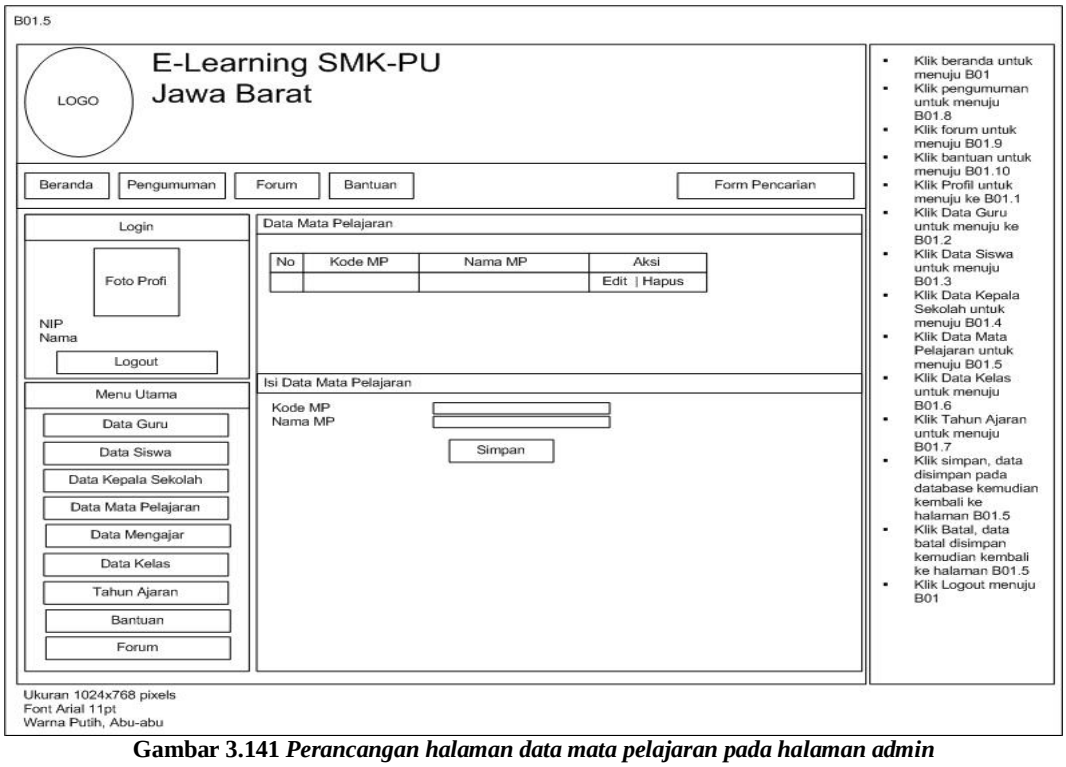
Gambar 3.139 ***Perancangan halaman form siswa pada halaman admin***

y. Perancangan halaman kepala sekolah pada halaman admin dapat dilihat pada Gambar 3.140



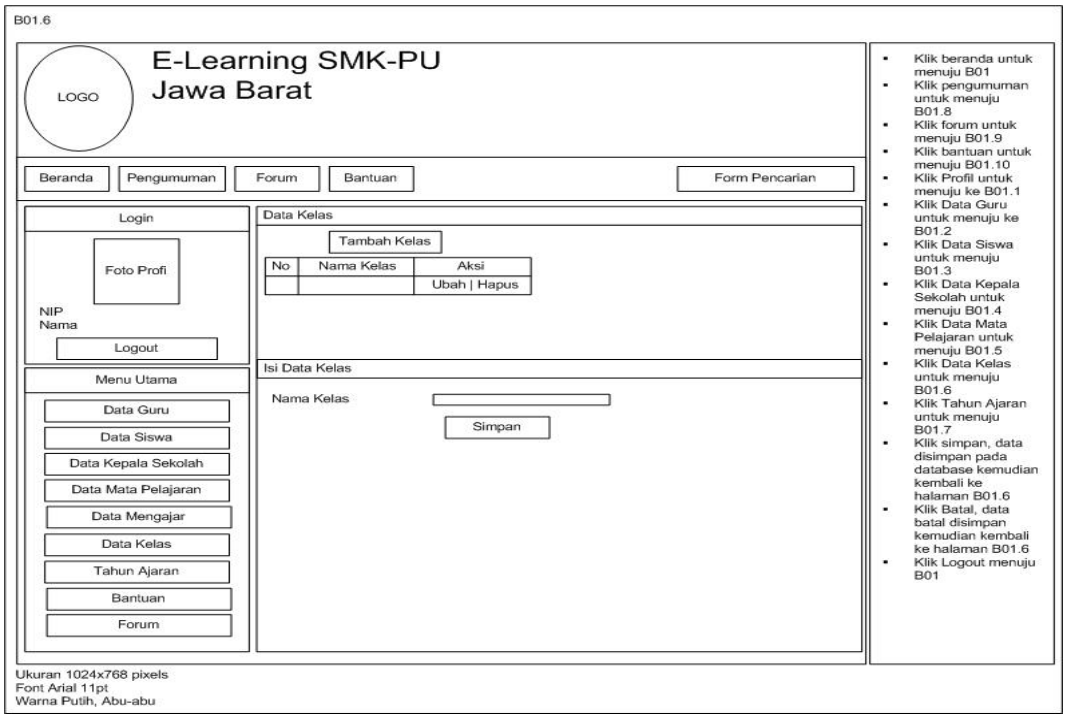
Gambar 3.140 ***Perancangan halaman data kepala sekolah pada halaman admin***

z. Perancangan halaman data mata pelajaran dapat dilihat pada Gambar 3.141



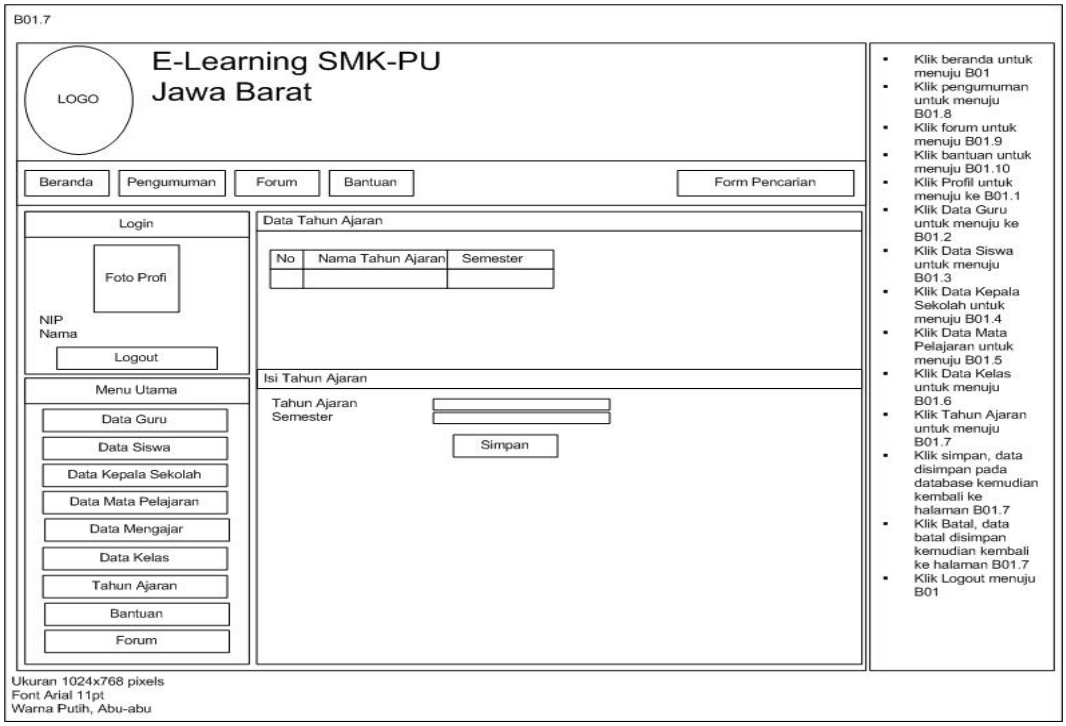
Gambar 3.141 ***Perancangan halaman data mata pelajaran pada halaman admin***

aa. Perancangan halaman kelas pada halaman admin dapat dilihat pada Gambar 3.142



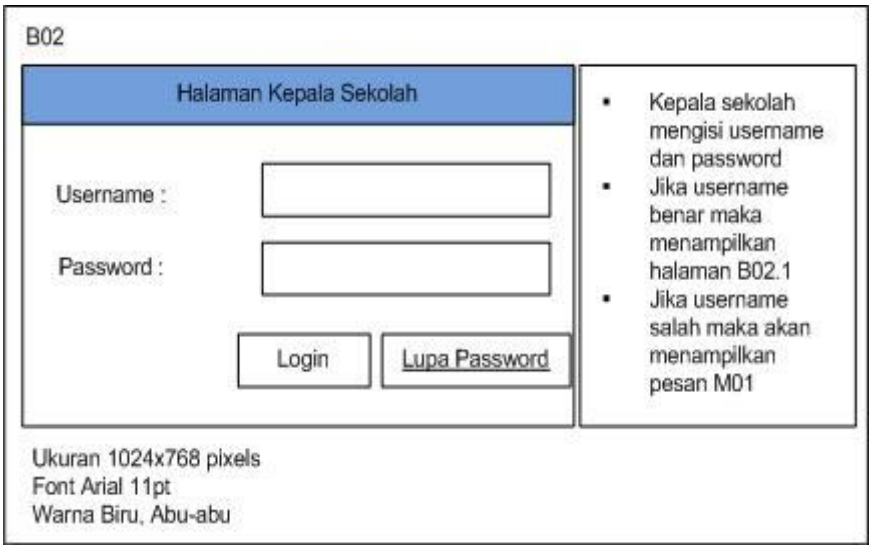
Gambar 3.142 ***Perancangan halaman data kelas pada halaman admin***

bb. Perancangan halaman tahun ajaran dapat dilihat pada Gambar 3.143



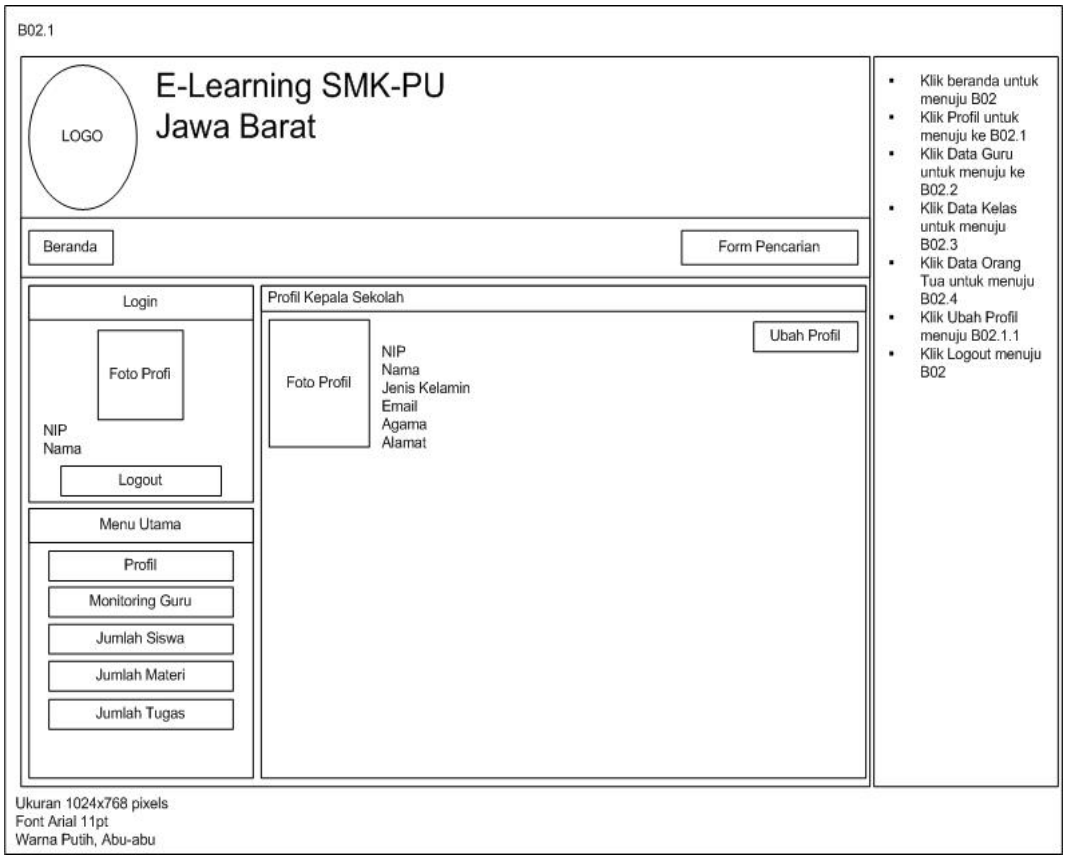
Gambar 3.143 ***Perancangan halaman tahun ajaran pada halaman admin***

cc. Perancangan halaman login Kepala sekolah dapat dilihat pada Gambar 3.144



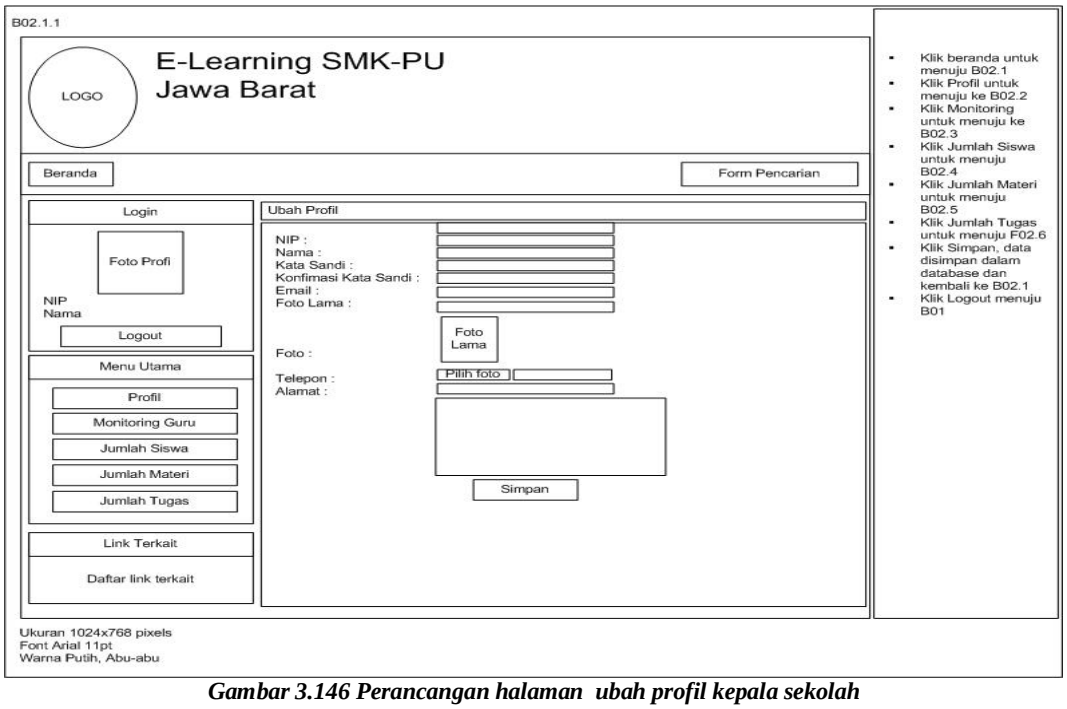
Gambar 3.144 ***Perancangan halaman login kepala sekolah***

dd. Perancangan halaman Kepala Sekolah dapat dilihat pada Gambar 3.145



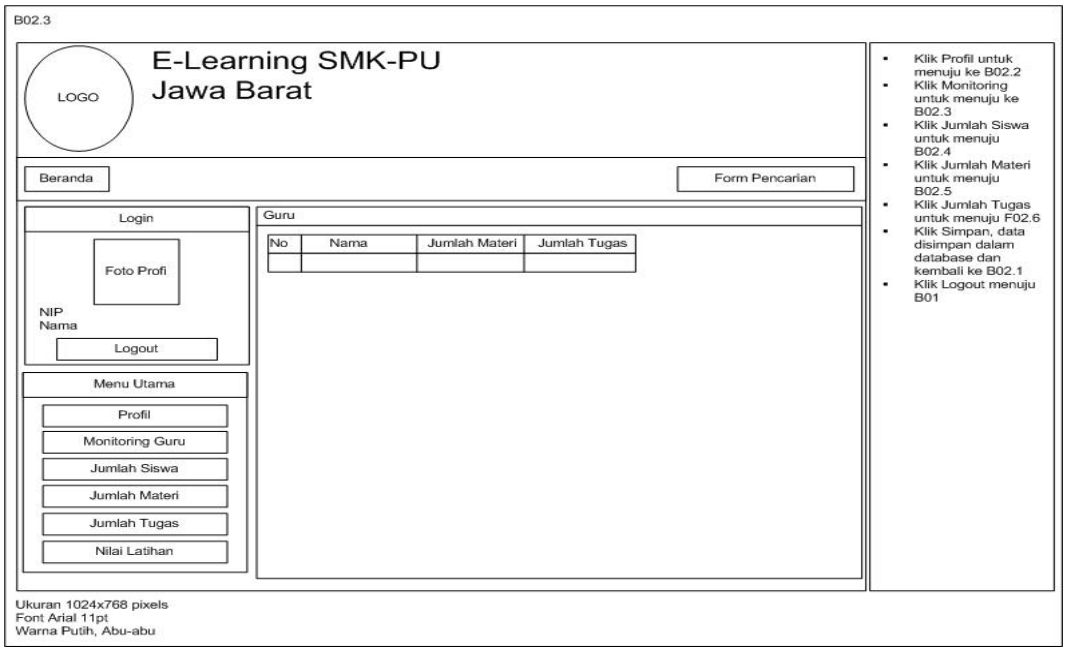
Gambar 3.145 ***Perancangan halaman kepala sekolah***

ee. Perancangan halaman ubah profil kepala sekolah dapat dilihat pada Gambar 3.146



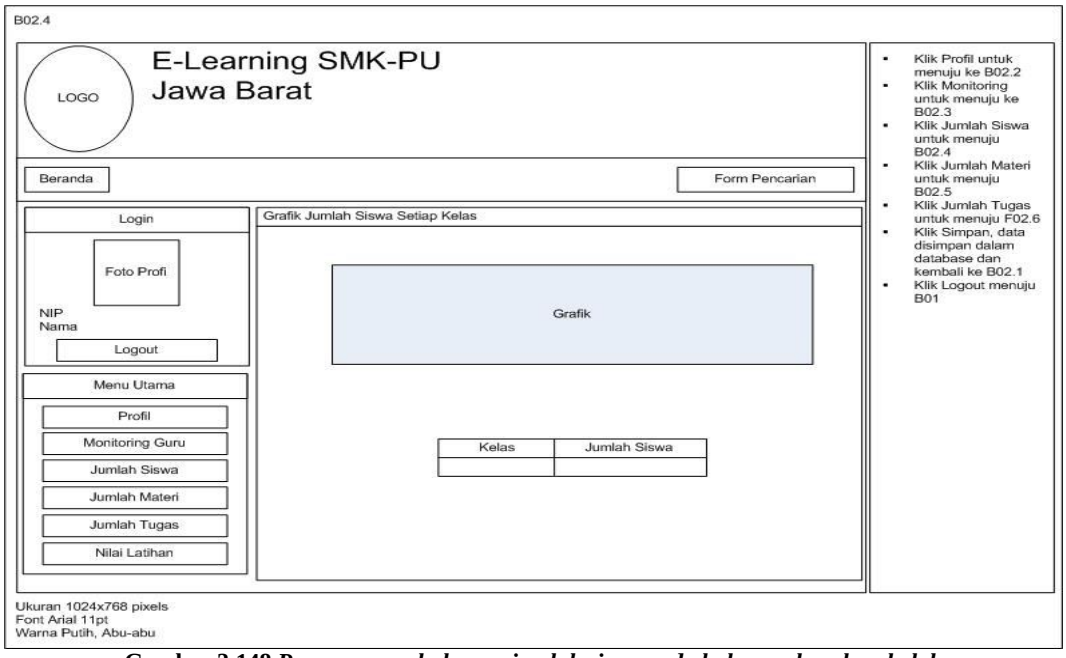
Gambar 3.146 Perancangan halaman ubah profil kepala sekolah

ff. Perancangan halaman monitoring guru pada halaman kepala sekolah dapat dilihat pada Gambar 3.147



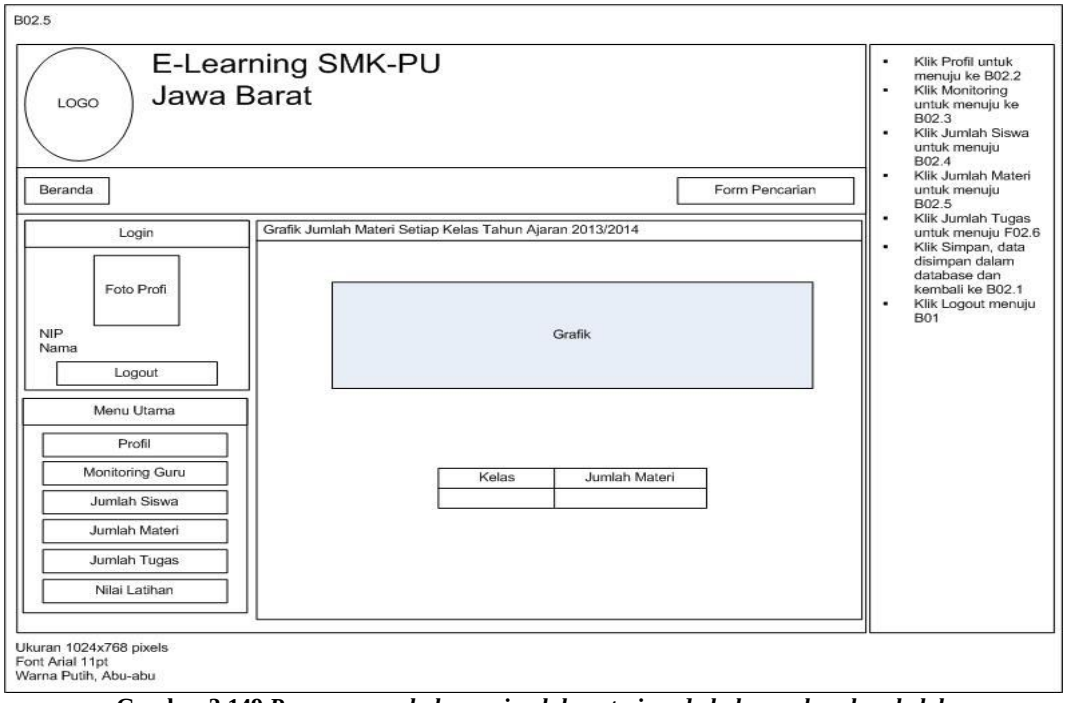
Gambar 3.147 ***Perancangan halaman monitoring guru pada halaman kepala sekolah***

gg. Perancangan halaman jumlah siswa pada kepala sekolah dapat dilihat pada Gambar 3.148



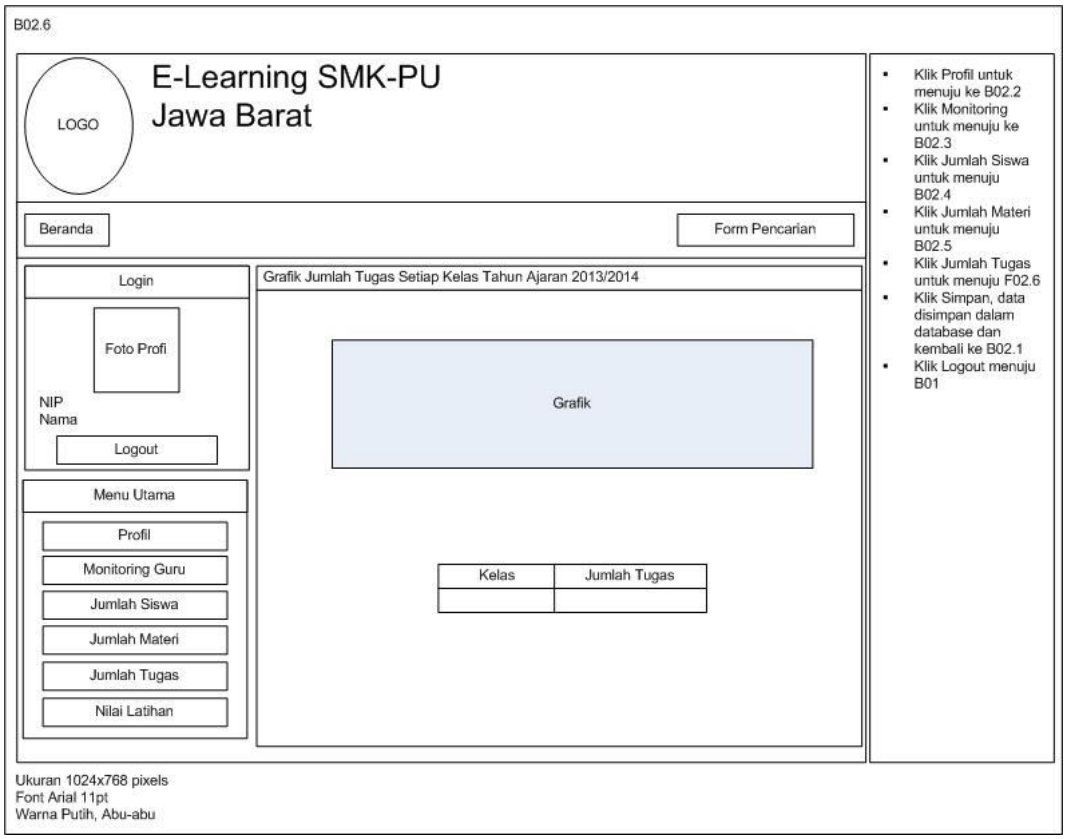
Gambar 3.148 ***Perancangan halaman jumlah siswa pada halaman kepala sekolah***

hh. Perancangan halaman jumlah materi pada halaman kepala sekolah dapat dilihat pada Gambar 3.149



Gambar 3.149 ***Perancangan halaman jumlah materi pada halaman kepala sekolah***

ii. Perancangan halaman jumlah tugas pada halaman kepala sekolah dapat dilihat pada Gambar 3.150



Gambar 3.150 ***Perancangan halaman jumlah tugas pada halaman kepala sekolah***

3.3.5 Perancangan Pesan

Perancangan pesan merupakan bahasa yang disampaikan oleh sistem kepada pengguna untuk menyampaikan informasi sesuai yang dilakukan oleh pengguna. Perancangan pesan yang ada pada aplikasi *e-learning* pada SMK-PU Prov. Jawa Barat dapat dilihat pada Tabel 3.89.

Tabel 3.89 Daftar Perancangan Pesan

No	Kode Pesan	Isi Pesan	Jenis Pesan	Terdapat Pada Form
1	M01	a. Bagian username harus diisi b. Bagian password harus diisi	Informasi	F01
2	M02	Data telah disimpan	Informasi	B01.2, B01.3, B01.4, B01.5,

				B01.6, B01.7, B01.8, B019, F02.3, F02.4, F02.5, F02.6, F02.7, F02.7.2, F03.3, F03.4, F03.5
3	M03	Data berhasil diubah	Informasi	B01.2, B01.3, B01.4, B01.5, B01.6, B01.9, F02.6
4	M04	Data berhasil dihapus	Informasi	B01.2, B01.3, B01.4, B01.5, B01.6, B01.7, B01.9, B01.10, F02.3, F.02,.4, F02.6
5	M05	Apakah anda yakin ingin menghapus data guru : *nama guru* ?	Konfirmasi	B01.2
6	M06	Apakah anda yakin ingin menghapus data siswa : *nama siswa* ?	Konfirmasi	B01.3
7	M07	Apakah anda yakin ingin menghapus data kepala sekolah : *nama kepala sekolah* ?	Konfirmasi	B01.4
8	M08	a. Bagian NIP harus diisi b. Bagian Nama Guru harus diisi c. Bagian Password harus diisi d. Bagian Konfirmasi Password harus diisi e. Bagian Email harus diisi f. Bagian Tempat lahir harus diisi g. Bagian tanggal lahir harus diisi	Informasi	B01.2.1
9	M09	a. Bagian NIS harus diisi b. Bagian Nama Siswa harus diisi c. Bagian Password	Informasi	B01.3.1

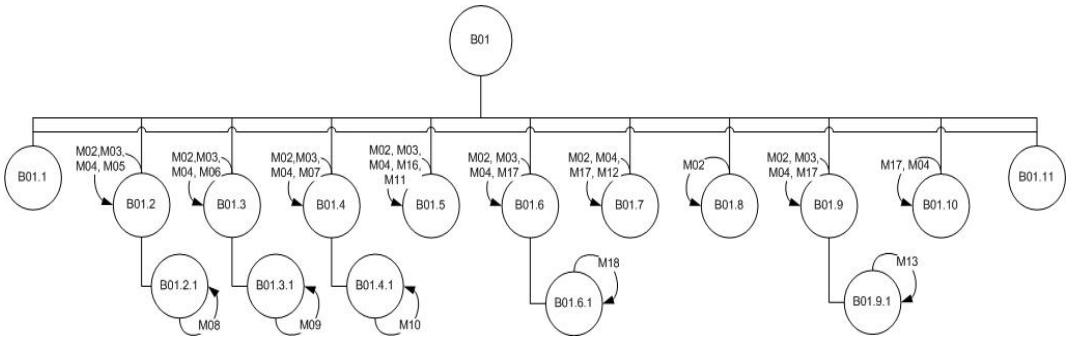
		harus diisi d. Bagian Konfirmasi harus diisi e. Bagian Email harus diisi f. Bagian Tempat Lahir harus diisi g. Bagian Tanggal Lahir		
10	M10	a. Bagian NIP harus diisi b. Bagian Nama Kepala Sekolah harus diisi c. Bagian Password harus diisi d. Bagian Konfirmasi Password harus diisi e. Bagian Email harus diisi f. Bagian Tempat Lahir harus diisi g. Bagian Tanggal Lahir harus diisi	Informasi	B01.4.1
11	M11	a. Bagian Kode Mata Pelajaran harus diisi b. Bagian Nama Mata Pelajaran harus diisi	Informasi	B01.5
12	M12	Bagian Kelas harus diisi	Informasi	B01.7
13	M13	a. Bagian Judul harus diisi b. Bagian Isi harus diisi	Informasi	B01.9.1, F.02.3.1, F02.6, F02.7.1
14	M14	a. Bagian Judul harus diisi b. Bagian Batas Mengumpulkan harus diisi c. Bagian Isi harus diisi	Informasi	F02.4.1
15	M15	a. Bagian Waktu Mulai harus diisi b. Bagian Waktu Berakhir harus diisi c. Bagian Lama mengerjakan harus diisi	Informasi	F02.5.1,
16	M16	Apakah anda yakin ingin menghapus data mata pelajaran : *nama mata*	Konfirmasi	B01.5

		pelajaran ?		
17	M17	Apakah anda yakin ingin menghapus data ini?	Konfirmasi	B01.6, B01.7, B01.9, B01.10, F02.6
18	M18	a. Bagian mata pelajaran harus diisi b. Bagian Kelas harus diisi c. Bagian Guru harus diisi	Informasi	B01.6.1
19	M19	a. Bagian pertanyaan harus diisi b. Bagian Jawab A harus diisi c. Bagian Jawab B harus diisi d. Bagian Jawab C harus diisi e. Bagian Jawab E harus diisi f. Bagian Jawab Benar harus diisi g. Bagian Materi harus diisi	Informasi	F02.5.2

3.3.6 Jaringan Semantik

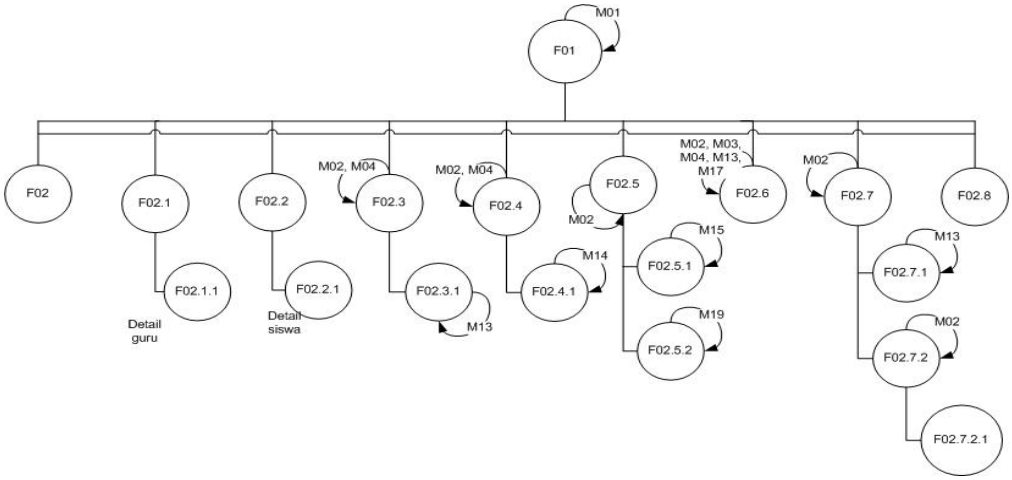
Struktur aplikasi ini dibuat secara moduler, yaitu program dipecah menjadi modul-modul kecil yang mudah dibuat, mudah dites, dan mudah dimodifikasi. Dalam pembuatan aplikasi *e-learning* ini menggunakan alat bantu bagan jaringan semantik.

1. Jaringan semantik aplikasi *e-learning* untuk admin.



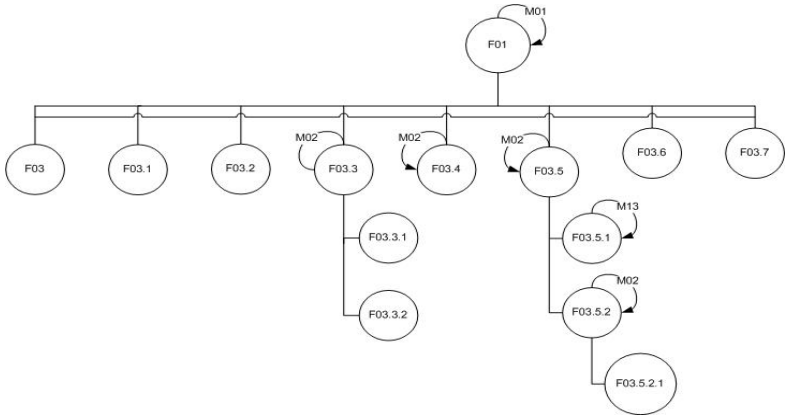
Gambar 3.151 Jaringan Semantik Admin

2. Jaringan semantik aplikasi *e-learning* untuk guru.



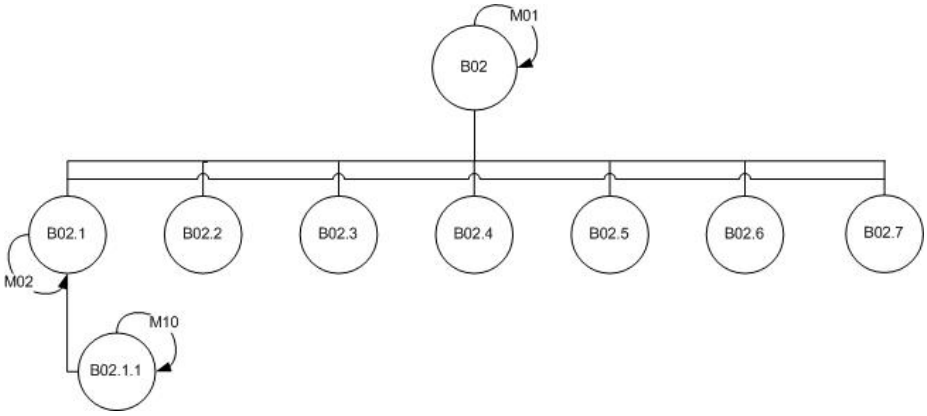
Gambar 3.152 ***Jaringan Semantik Untuk Guru***

3. Jaringan semantik aplikasi *e-learning* untuk siswa.



Gambar 3.153 ***Jaringan Semantik Untuk Siswa***

4. Jaringan semantik aplikasi *e-learning* untuk kepala sekolah.



Gambar 3.154 ***Jaringan Semantik Untuk Kepala Sekolah***