BAB 3
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Pada bagian ini akan dilakukan analisis terhadap sistem yang sedang berjalan. Hasilnya digunakan untuk mengidentifikasi dan mengevaluasi berbagai kelemahan, permasalahan, kebutuhan dan hambatan yang terdapat pada sistem pembelajaran di SMA Negeri 3 Sumedang. Dari proses analisis tadi akan dapat dihasilkan berbagai saran perbaikan terhadap sistem yang dapat dijadikan dasar dalam merancang *e-learning* yang akan dibangun.

3.1.1 Analisis Masalah

Permasalahan yang sering kali timbul tanpa diduga, berawal dari hal-hal kecil yang sering kali terlupakan hingga akhirnya bila terus menerus terjadi akan menjadi masalah yang bisa berakibat besar.

Berdasarkan hasil penelitian yang dilakukan di SMA Negeri 3 Sumedang, adapun masalah–masalah yang timbul adalah sebagai berikut :

1. Kurang tersedianya fasilitas tambahan materi yang berisi tentang materi-materi pelajaran berdasarkan kurikulum yang berlaku
2. Kurangnya sarana informasi dan komunikasi antara siswa dan guru baik di sekolah ataupun di luar sekolah.
3. Analisis Kurang tersedianya media penyimpanan materi atau tugas dari siswa-siswi dan guru yang dapat diakses setiap saat di setiap tempat baik di sekolah ataupun di luar sekolah.

3.1.2 Prosedur yang Sedang Berjalan

Analisis sistem yang sedang berjalan bertujuan agar dapat diuraikan prosedur- prosedur yang masih berlangsung di SMA Negeri 3 Sumedang. Maka dari itu akan ditemukan kekurangan atau kelemahan dari sistem yang sedang berjalan. Analisis terhadap sistem yang sedang berjalan dibuat untuk menguraikan bagaimana langkah-langkah prosedural yang

sedang berjalan, dari analisis ini dapat ditemukan bagaimana kekurangan ataupun kelemahan dari sistem yang sedang berjalan.

3.1.2.1 Prosedur Pembelajaran di SMA Negeri 3 Sumedang

Prosedur pembelajaran yang berjalan di SMA Negeri 3 Sumedang adalah sebagai berikut :

1. Prosedur pemberian materi.
2. Prosedur tugas
3. Prosedur ujian

Prosedur-prosedur yang terlibat dapat dijelaskan sebagai berikut :

1. Prosedur Pemberian Materi

Prosedur pemberian materi merupakan kegiatan inti dari pembelajaran dimana guru memberikan penjelasan materi pelajaran secara langsung kepada siswa-siswinya: Adapupn alur prosedur nya seperti dibawah ini :

a. Guru menyiapkan materi pelajaran berdasarkan kurikulum yang akan diajarkan kepada siswa.
b. Setelah materi pelajarannya siap, guru memberikan materi pelajaran tersebut dan mengajarkan siswa di kelas sesuai dengan kurikulum.
c. Siswa mempelajari materi pelajaran dan mengikuti pelajaran di kelas.
d. Apabila materi pelajaran yang diajarkan guru belum dimengerti siswa, siswa dapat menanyakan materi pelajaran tersebut kepada guru, dan guru dapat menjelaskannya kembali.

2. Prosedur Tugas

Prosedur tugas adalah tahap untuk mengevaluasi kemampuan siswa-siswi terhadap materi pelajaran yang sudah disampaikan oleh guru. Adapaun alur prosedurnya seperti dibawah ini

a. Guru memberikan tugas kepada siswa.
b. Siswa mengerjakan tugas yang diberikan oleh guru.
c. Guru mengoreksi jawaban tugas yang telah dikerjakan siswa, dan memasukan nilainya ke daftar nilai siswa.

d. Guru memberikan jawaban tugas yang telah dimasukan ke daftar nilai kepada siswa.

3. Prosedur Ujian

Prosedur ujian adalah tahap untuk menguji kemampuan siswa-siswi terhadap materi pelajaran yang sudah disampaikan oleh guru. Adapaun alur prosedurnya seperti dibawah ini :

a. Guru memberikan soal ujian kepada siswa.
b. Siswa-siswi mengerjakan soal ujian yang diberikan oleh guru.
c. Siswa-siswi mengumpulkan jawaban soal ujian kepada guru.
d. Guru memeriksa hasil jawaban soal ujian para siswa.
e. Guru memasukan nilai kedalam buku nilai.
f. Guru mengembalikan hasil soal ujian yang sudah diperiksa dan dinilai kepada siswa.

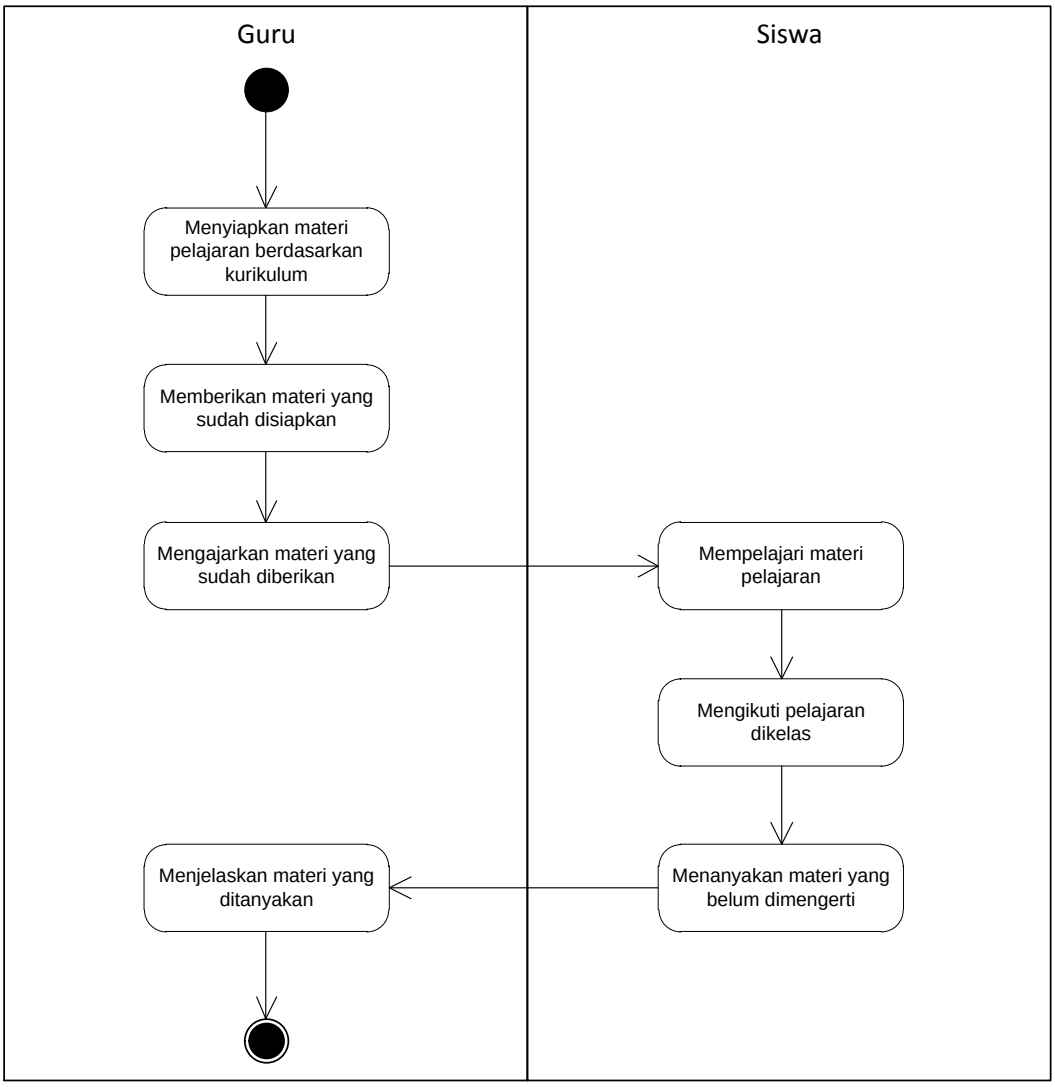
3.1.2.2 Model Activity Diagram

Activity diagram (diagram aktivitas) adalah diagram yang menggambarkan aliran fungsionalitas dari sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business work flow*). Dapat juga digunakan untuk menggambarkan aliran kejadian (*flow of events*).Berikut ini adalah diagram aktivitas pembelajaran yang sedang berjalan di SMA Negeri 3 Sumedang

1. *Activity Diagram* Prosedur Pemberian Materi

Deskripsi : Aktifitas yang dapat dilakukan dalam proses belajar mengajar antara guru dan siswa di sekolah

Activity diagram prosedur pemberian materi di SMA NEGERI 3 Sumedang dapat dilihat pada Gambar 3.1.

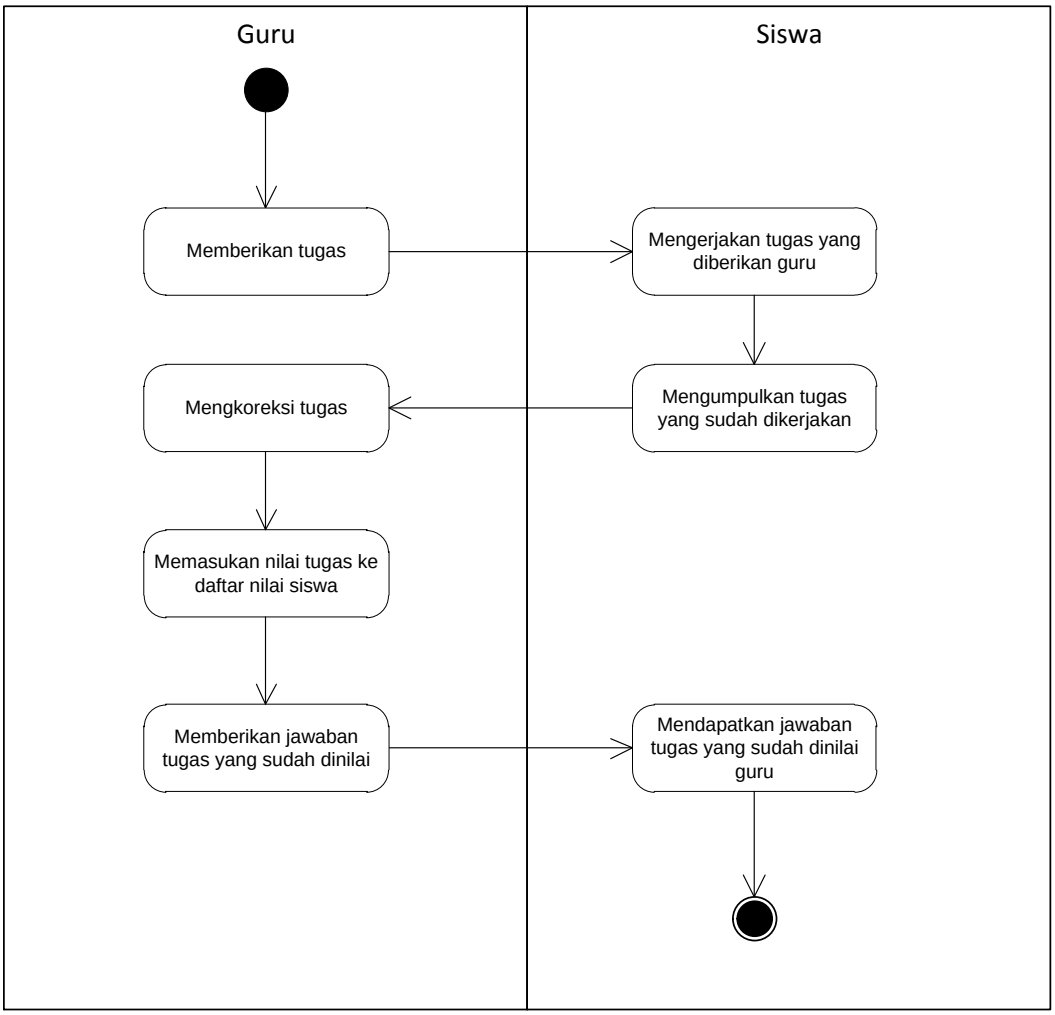


Gambar 3. 1 Activity Diagram Prosedur Pemberian Materi

2. *Activity Diagram* Prosedur Tugas

Deskripsi : Aktifitas yang dapat dilakukan ketika tugas sekolah diberikan dari guru ke siswa

Activity diagram prosedur tugas di SMA NEGERI 3 Sumedang dapat dilihat pada Gambar 3.2.

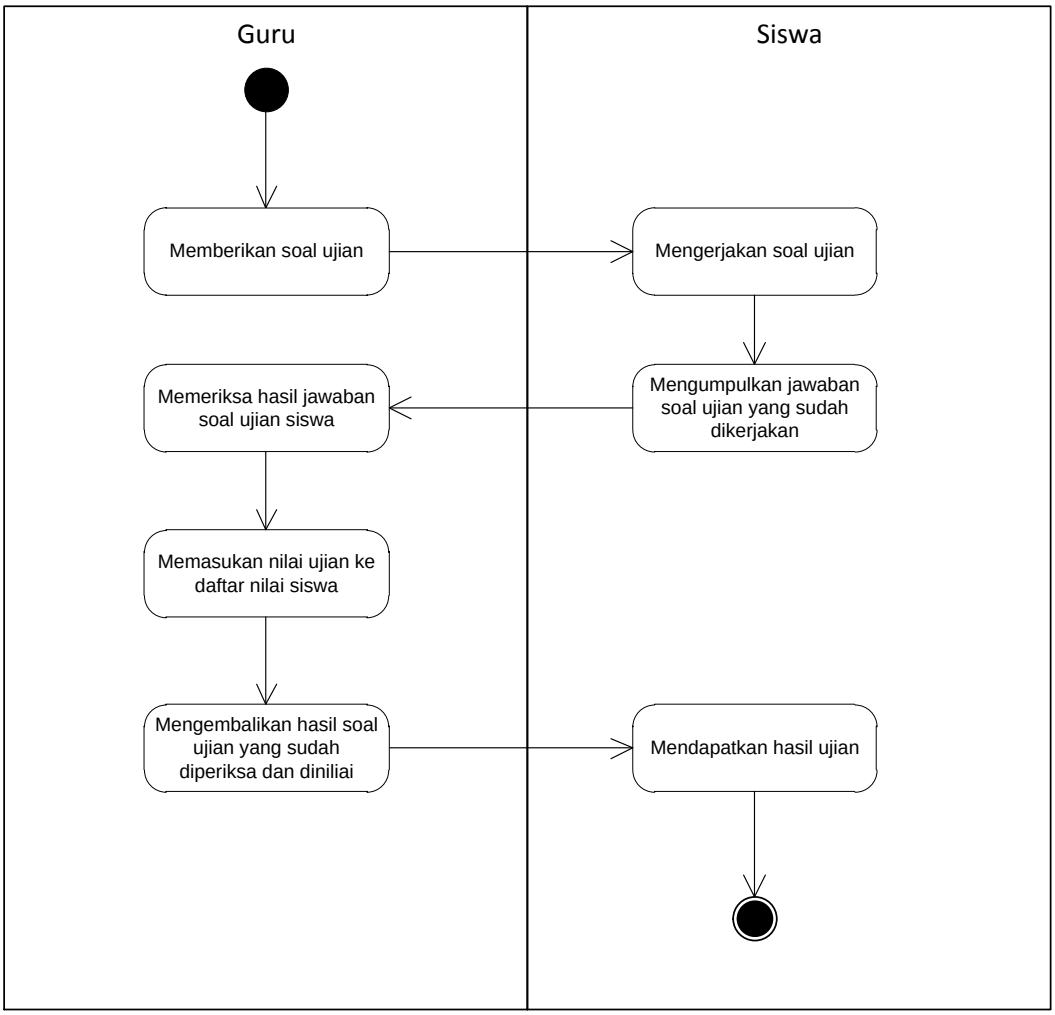


Gambar 3. 2 Activity Diagram Prosedur Tugas

3. *Activity Diagram* Prosedur Ujian

Deskripsi : Guru meringkas isi materi pelajaran dari kurikulum yang berlaku guna memudahkan siswa untuk memahami isi dari materi tersebut

Activity diagram prosedur ujian di SMA NEGERI 3 Sumedang dapat dilihat pada Gambar 3.3.



Gambar 3. 3 Activity Diagram Prosedur Ujian

3.1.3 Metode E-learning

Metode yang dipakai dalam pembangunan *e-learning* di SMA Negeri 3 Sumedang adalah *Blended learning* karena *Blended learning* metode pembelajaran yang menggabungkan antara sistem *e-learning* dengan metode konvensional atau tata muka

3.1.4 Aturan Bisnis E-learning di SMA Negeri 3 Sumedang

Aturan bisnis *e-learning* di SMA Negeri 3 Sumedang sebagai berikut:

1. Username dan Password

   username untuk admin dan guru dapat diganti sesuai dengan kemauan, sedangkan untuk siswa menggunakan nis yang berisi 9 karakter dan tidak bisa diganti. Password harus lebih dari 6 karakter.

2. Pesan

   Fitur pesan adalah alat untuk berkomunikasi antar guru dengan siswa, guru dengan guru, atau siswa dengan siswa. Admin tidak bisa mengunakan fitur pesan ini.

3. Forum

   Siswa atau guru dapat membuat toik forum dan mengomentari topik forum tersebut.

4. Mengajar

   Guru dapat mengajar lebih dari satu pelajaran. Tidak boleh ada dua guru yang mengajar pelajaran yang sama dalam satu kelas.

5. Materi dan tugas

   Siswa dapat mengunduh materi jika guru menggunggah materi ke dalam kelas. Siswa dapat mengirim tugas jika guru membuat tugas untuk siswa

6. Ujian Online

   Siswa dapat mengerjakan ujian online yang sudah dibuat oleh guru. Siswa harus menyelesaikan ujian online sebuelum batas waktu yang ditentukan guru, jika batas waktu sudah habis sebelum menyelesaikan ujian online maka ujian online akan dihentikan oleh sistem dan nilai ujian dihitung berdasarkan jumlah jawaban terakhir yang dikerjakan. Ujian online dilakukan di lab. komputer sekolah dengan pengawasan guru sesuai dengan jadwal pelajaran.

3.1.5 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional sendiri merupakan analisis yang dibutuhkan untuk menentukan spesifikasi sistem yang nantinya akan diterapkan pada aplikasi *e-learning*. Analisis kebutuhan non fungsional juga menentukan spesifikasi masukan (*input*) yang diperlukan sistem, keluaran (*output*) yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan (*input*) dan akhirnya didapatkan suatu keluaran (*output*) yang dikehendaki. Kebutuhan non fungsional terbagi menjadi beberapa analisis yaitu analisis pengguna sistem, analisis pengkodean, analisis perangkat keras, analisis perangkat lunak, analisis jaringan

3.1.5.1 Analisis Pengguna Sistem

Sistem yang akan dibangun digunakan oleh tiga jenis pengguna, yaitu

1. Admin

   Admin adalah petugas sekolah yang mengelola jaringan komputer disekolah SMA Negeri 3 Sumedang yang ditempatkan di ruang perpustakaan

2. Guru

   Guru adalah pendidik atau pengajar yang berada di SMA Negeri 3 Sumedang

3. Siswa.

   Siswa adalah peserta didik yang ada di SMA Negeri 3 Sumedang

Adapun karakteristik pengguna di SMA Negeri 3 Sumedang dapat dilihat pada Tabel 3.1.

Tabel 3. 1 Karakteristik Pengguna SMA Negeri 3 Sumedang

Karakteristik Pengguna	
Pengguna	Admin
Hak Akses	a. Pengelolaan data Siswa-siswi. b. Pengkategorian Siswa-siswi (Siswa-siswi kelas x, xi, xii) c. Pengelolahan data Guru d. Pengkategorian Guru (Guru fisika, Guru geografi dan lain-lain) e. Pengelolaan berita f. Pengelolaan ilmu

Karakteristik Pengguna	
	g. Pengelolaan pengumuman h. Pengelolaan polling i. Pengelolaan forum diskusi (topik) j. Pengelolaan matapelajaran k. Pengelolaan kelas (berdasarkan nama guru dan pelajaran) l. Pengelolaan tahun ajaran (semester).
Tingkat Pendidikan	S1 atau setara
Tingkat Keterampilan	Menguasai komputer, mengerti cara kerja sistem e-learning, dapat menggunakan internet untuk browsing dan mengerti cara unduh unggah file
Pengguna	Guru
Hak Akses	a. Unggah berita yang berkaitan dengan pendidikan b. Unggah ilmu c. Unggah pengumuman d. Unggah *ebook* yang berkaitan dengan pelajaran di sekolah e. Mengikuti forum diskusi f. Unggah konten/materi. g. Pengelolaan kelas *online* h. Pengelolaan soal ujian *online* i. Pengelolaan tugas. j. Mengikuti polling
Tingkat Pendidikan	S1 atau setara
Tingkat Keterampilan	Menguasai komputer, mengerti cara penggunaan untuk web, dapat menggunakan internet untuk browsing dan mengerti cara unduh unggah file
Pengguna	Siswa
Hak Akses	a. Lihat berita b. Lihat ilmu c. Lihat pengumuman d. Mengikuti polling e. Mengikuti forum diskusi f. Unduh konten/materi, *ebook* g. Pemililihan kelas (berdasarkan nama guru dan pelajaran)

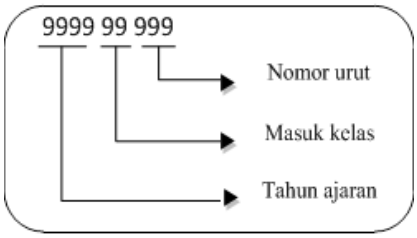
Karakteristik Pengguna	
	h. Mengikuti ujian *online* (berdasarkan nama guru dan pelajaran) b. *Unduh*/unggah tugas. c. Mengikuti polling
Tingkat Keterampilan	Menguasai komputer, mengerti cara penggunaan web, dapat menggunakan internet untuk browsing dan mengerti cara unduh unggah file

3.1.5.2 Analisis Pengkodean

Pada bagian ini akan dibahas tentang pengkodean yang ada di SMA Negeri 3 Sumedang. Pengkodean di SMA Negeri 3 Sumedang terdiri dari pengkodean Nomor Induk Siswa (NIS), kelas, dan pelajaran.

1. Pengkodean NIS (Nomor Induk Siswa)

   Pengkodean NIS disekolah tersebut hanya menggunakan nomor urut, hal tersebut dapat mengganggu system yang akan dibangun, untuk mengatasinya maka akan diusulkan pengkodean NIS baru. Pengkodean NIS terdiri dari 9 karakter, dengan format dapat dilihat pada gambar 3.5.



Gambar 3. 4 Format Pengkodean NIS

   Contoh : 121310254 berarti siswa tersebut terdaftar di sekolah pada tahun ajaran 2012/2013, kode 10 berarti siswa masuk ke kelas 10, kode 254 berarti no urut siswa.

2. Pengkodean Kelas

   Pengkodean kelas terdiri dari 10 karakter, dengan format seperti Gambar 3.5



Gambar 3. 5 Format Pengkodean Kelas

Jurusan : 00 = Umum

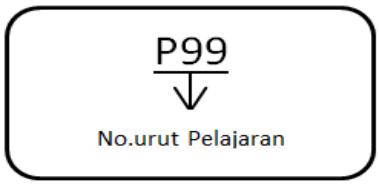
01= IPA

02= IPS

Contoh : 2012110102 menunjukkan kelas tahun 2012 tingkat XI jurusan IPA urutan kelas nomor 2.

3. Pengkodean Pelajaran

   Pengkodean kelas terdiri dari 3 karakter, dengan format seperti Gambar 3.6



Gambar 3. 6 Format Pengkodean Pelajaran

P = Pelajaran

99 = No. urut pelajaran

Contoh : P05 menunjukkan pelajaran matematika.

3.1.5.3 Analisis Perangkat Keras

Sistem yang akan dibangun akan diimplementasikan langsung dalam jaringan *internet*. Perangkat keras yang digunakan di SMA Negeri 3 Sumedang saat ini adalah komputer dengan spesifikasi sebagai berikut:

1. Prosesor setara dengan kecepatan 1.4 GHz
2. RAM 500 MB – 2 GB
3. *Harddisk* dengan kapasitas 100 Gb
4. Monitor dengan resolusi sebesar 1024 x 768 *pixels*

Sedangkan untuk menjalankan aplikasi yang akan dibangun membutuhkan perangkat keras dengan spesifikasi minimum , yaitu:

1. Prosesor kecepatan 800 MHz atau lebih.
2. RAM 256 Mb atau lebih
3. *Harddisk* dengan ruang kosong 10 Gb atau lebih
4. VGA Card *shared* 64 Mb atau lebih
5. Monitor dengan resolusi 1024 x 768 *pixels* atau lebih
6. Kartu jaringan, HUB/Switch, Router atau modem.

Berdasarkan data tersebut, maka perangkat keras di sekolah telah mencukupi untuk dikembangkannya aplikasi *e-learning* yang akan dibangun.

3.1.5.4 Analisis Perangkat Lunak

Perangkat lunak yang tersedia saat ini di SMA Negeri 3 Sumedang:

1. Sistem Operasi *Windows7 32-bit Operating System.*
2. *Browser* : *Mozilla Firefox* dan *Google Chrome*.

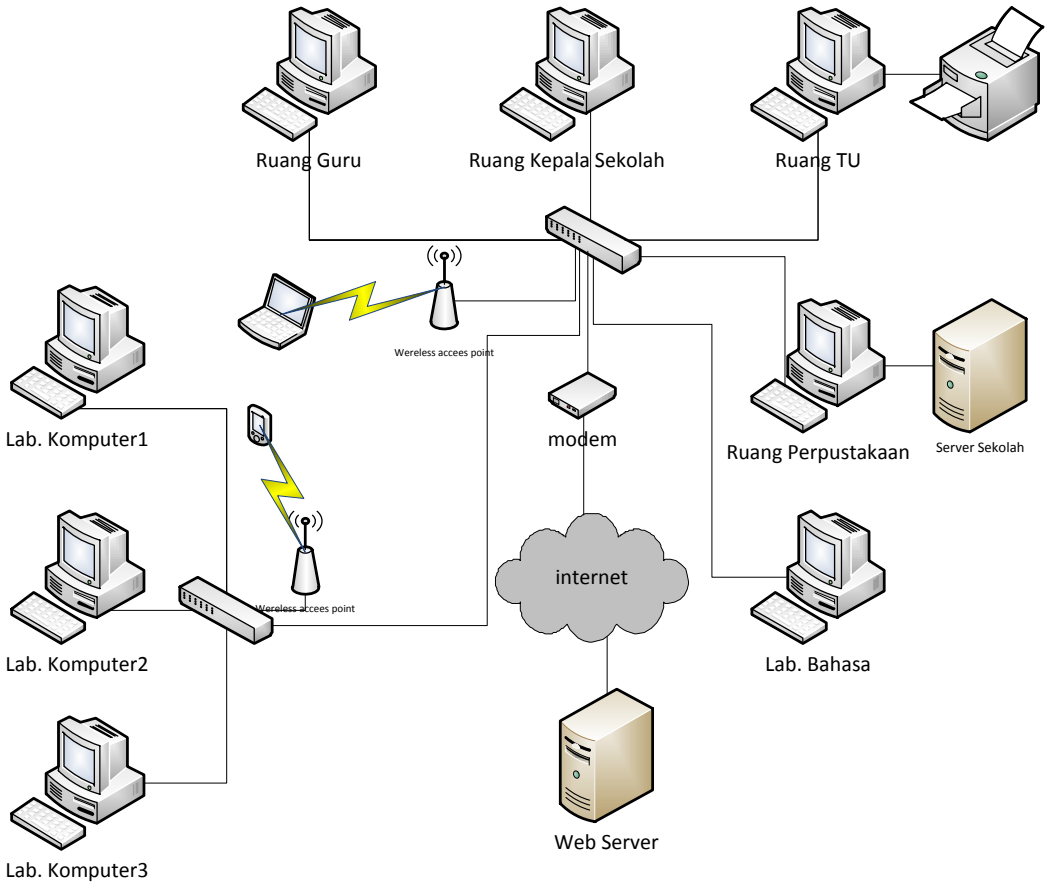
Perangkat lunak yang digunakan untuk membangun aplikasi *e-learning* adalah sebagai berikut :

1. Sistem Operasi *Windows* 7 sebagai Sistem Operasi.
2. *PHP* sebagai bahasa pemrograman
3. *MySql* Sebagai *Database.*
4. *PhpMyAdmin* Sebagai *Interface Database.*
5. *Adobe Dreamweaver CS3* Sebagai *code Ubahor.*
6. *Mozile Firefox dan Google Chrome* sebagai *web browser.*
7. *Microsoft Visio 2010* untuk memodelkan *UML*.

3.1.5.5 Analisis Jaringan

Pada dasarnya dalam analisis jaringan ini adalah menganalisa jaringan yang akan digunakan dalam aplikasi *e-learning* tersebut dan jaringan yang terdapat pada instansi yang bersangkutan. Jaringan yang ada pada SMA Negeri 3 Sumedang sudah terhubung oleh internet. Namun dalam pemanfaatan jaringan internet tersebut belum maksimal. Untuk dapat beroperasi secara *online*, aplikasi *e-learning* yang telah selesai dibangun

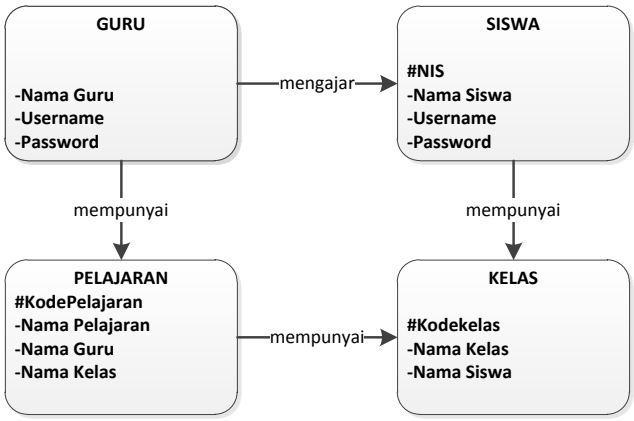
harus di*hosting* terlebih dahulu ke *Web Server*. *Hosting* aplikasi dilakukan di *website* komersial. Setiap *browser* baik itu admin, guru ataupun siswa harus terhubung ke internet untuk dapat mengakses aplikasi *e-learning*. Adapun gambar antar muka dari aplikasi *e-learning* terdapat pada Gambar 3.7.



Gambar 3. 7 Jaringan Antar Muka SMA Negeri 3 Sumedang

3.1.6 Analisis Data

Data yang digunakan pada aplikasi ini adalah data yang ada di SMA Negeri 3 Sumedang dapat dilihat pada Gambar 3.8.



Gambar 3. 8 Data yang ada di SMA Negeri 3 Sumedang

3.1.7 Analisis Kebutuhan Fungsional

Kebutuhan dalam pembangunan aplikasi ini menyangkup beberapa diagram UML, dimana diagram-diagram tersebut masing-masing menggambarkan proses-proses yang berjalan pada aplikasi e-Learning ini. Berikut diagram-diagram yang dipakai dalam menganalisis program :

1. *Use case* diagram
2. *Class* diagram
3. *Sequence* diagram
4. *Package* diagram
5. *Deployment* diagram

3.1.7.1 Deskripsi Umum Sistem

Sistem yang dibuat merupakan aplikasi *e-learning* yang digunakan di SMA Negeri 3 Sumedang. Sistem yang dibangun dapat membantu siswa yang hendak mencari referensi tentang materi, membantu guru dalam proses pembelajaran, Sistem yang dibangun akan diimplementasikan pada SMA Negeri 3 Sumedang yang terhubung langsung kedalam *database* learning.

3.1.7.2 Fungsi E-learning SMA Negeri 3 Sumedang

Secara umum, *e-learning* SMA Negeri 3 Sumedang sebagai media untuk melakukan proses pembelajaran secara online. Fungsi utama perangkat lunak yang dibangun dibagi menjadi beberapa fasilitas untuk setiap aktor yang meliputi menu-menu apa saja yang ada pada aplikasi yang dibagun.

1. Fasilitas yang disediakan untuk aktor admin yaitu :
   a. Admin dapat mengolah data siswa meliputi *import*, tambah, ubah, hapus, cari data siswa.
   b. Admin dapat mengolah data guru meliputi tambah, ubah, hapus, cari data guru.
   c. Admin dapat mengolah data kelas meliputi tambah, ubah, hapus, cari data kelas.
   d. Admin dapat mengolah data mengajar meliputi tambah, ubah, hapus, cari data mengajar.
   e. Admin dapat mengolah polling meliputi tambah, ubah, hapus.
   f. Admin dapat mengolah data pelajaran meliputi tambah, ubah, hapus, cari data pelajaran.
   g. Admin dapat mengelola tahun ajaran
   h. Admin dapat mengelola berita, ilmu, *ebook*, pengumuman meliputi ubah dan hapus.
2. Fasilitas yang disediakan untuk guru yaitu :
   a. Guru dapat lihat berita, lihat ilmu, lihat pengumuman.
   b. Guru dapat tambah, ubah, hapus berita
   c. Guru dapat tambah, ubah, hapus ilmu
   d. Guru dapat tambah, ubah, hapus *ebook*
   e. Guru dapat tambah, ubah, hapus pengumuman.
   f. Guru dapat tambah, ubah, hapus materi dan tugas
   g. Guru dapat tambah, ubah, hapus soal ujian, dan lihat nilai ujian yang sudah dilakukan siswa.
   h. Guru dapat mengirim pesan ke guru lain atau ke siswa.
   i. Guru dapat membuat dan mengomentari topik forum.
   j. Logout untuk keluar dari sistem.
3. Fasilitas yang disediakan untuk siswa yaitu :
   a. Siswa dapat lihat berita, lihat ilmu, lihat pengumuman.
   b. Siswa dapat lihat dan unduh *ebook*.

c. Siswa dapat masuk kelas berdasarkan nama guru dan pelajaran, unggah unduh materi dan tugas.
d. Siswa dapat mengikuti ujian online dan melihat nilai ujian online yang sudah dilakukan.
e. Siswa dapat mengirim pesan ke siswa lain atau ke guru
f. Siswa dapat mengikuti polling.
g. Siswa dapat membuat dan mengomentari topik forum.
h. Logout untuk keluar dari sistem.

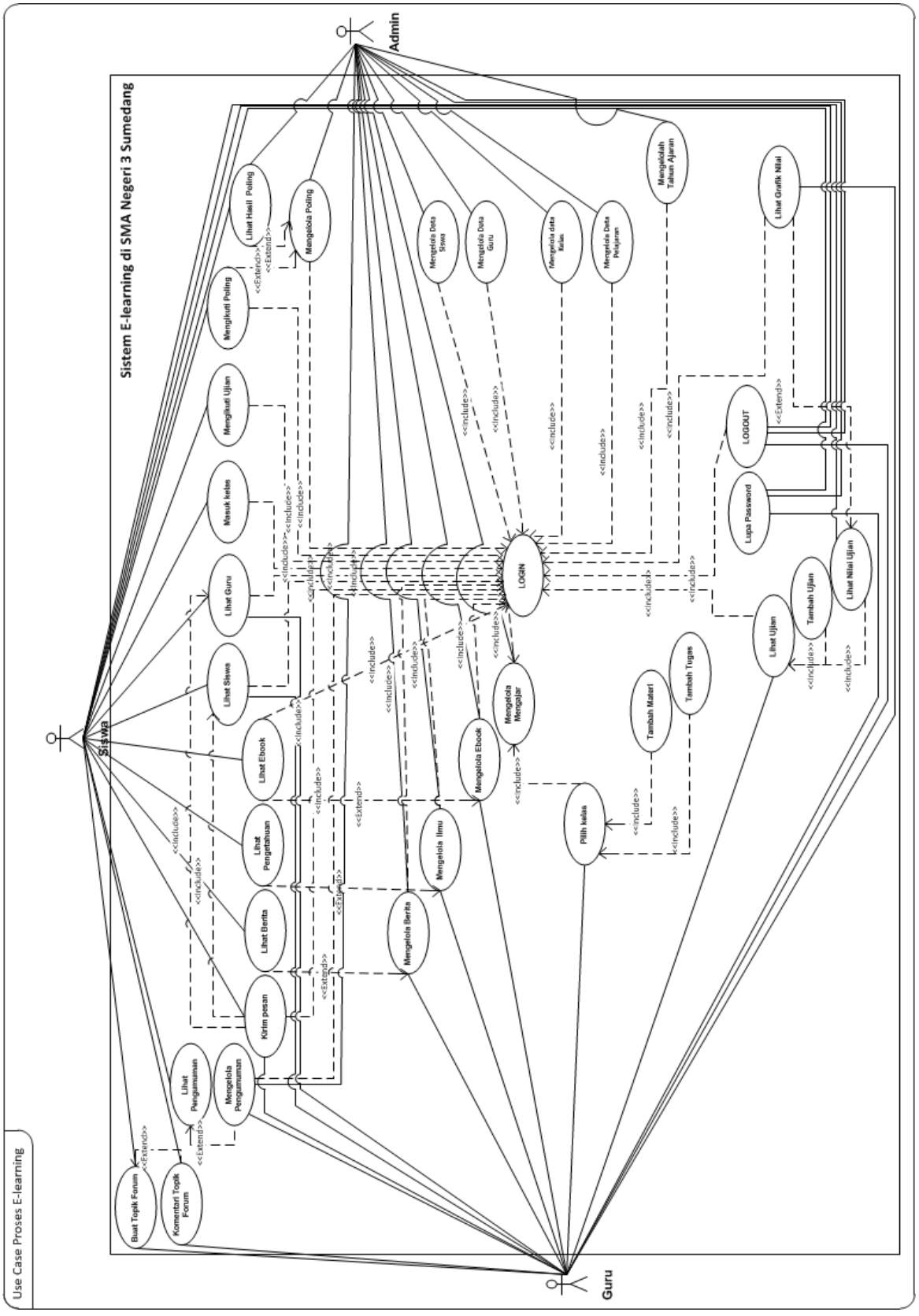
3.1.7.3 Identifikasi Aktor

Aplikasi *e-learning* yang dibangun akan digunakan oleh tiga orang aktor yaitu *admin*, guru, dan murid. Deskripsi dari masing-masing aktor adalah sebagai berikut :

1. Aktor pertama adalah admin sebagai pengelola yang semua hak akses yang ada pada sistem meliputi *import*, tambah, ubah, hapus, cari data.
2. Aktor kedua adalah guru yang memiliki hak akses lihat berita , proses mengolah berita, ilmu, ebook, pengumuman, materi dan tugas, ujian dan lihat nilai ujian, mengirim pesan, membuat dan mengomentari topik forum.
3. Aktor ketiga adalah siswa yang memiliki hak akses lihat berita, lihat ilmu, lihat pengumuman, unduh *ebook*, unggah unduh materi dan tugas, ikut ujian online dan lihat nilai ujian, mengirim pesan, membuat dan mengomentari topik forum, mengikuti polling, lihat hasil polling.

3.1.7.4 *Use* Case Diagram

Pemodelan *use case* adalah pemodelan sistem dari perspektif pandangan pemakai akhir (*end user*). Model *use case* adalah pandangan dari luar sistem. *Use case* diagram dari aplikasi elearning di SMA Negeri 3 Sumedang dapat dilihat pada Gambar 3.9.



Gambar 3. 9 Use Case Elearning SMA Negeri 3 Sumedang

3.1.7.5 *Scenario Diagram*

Scenario diagram berfungsi untuk menjelaskan lebih detail mengenai modul-modul yang terdapat dalam website *e-learning* yang dibangun. *Scenario diagram* dari use case *e-learning* di SMA Negeri 3 Sumedang adalah sebagai berikut:

1. Login

   Interaksi antara aktor dengan use case login dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.2.

Tabel 3. 2 Skenario Use Case login

<table>
<tr><th colspan="5">Skenario Use Case</th></tr>
<tr><th colspan="5">Identifikasi</th></tr>
<tr><td colspan="2">Nama</td><td colspan="3">Login</td></tr>
<tr><td colspan="2">Tujuan</td><td colspan="3">Masuk ke dalam sistem</td></tr>
<tr><td colspan="2">Deskripsi</td><td colspan="3">Proses login merupakan proses untuk autentikasi untuk menggunakan hak akses baik sebagai admin,, guru, dan siswa dalam sister</td></tr>
<tr><td colspan="2">Aktor</td><td colspan="3">Admin, guru, siswa</td></tr>
<tr><td colspan="2">Use Case yang Berkaitan</td><td colspan="3">Lihat kelas, lihat ujian, kirim pesan, lihat siswa, lihat guru, pilih kelas, mengikuti ujian, mengikuti polling, mengelola berita, mengelola ilmu, mengelola ebook, mengelola kelas, mengelola polling, lihat data siswa, lihat data guru, lihat data kelas, lihat data pelajaran, mengelola tahun ajaran, mengelola pengumuman,</td></tr>
<tr><th colspan="5">Skenario Utama</th></tr>
<tr><td colspan="2">Kondisi Awal</td><td colspan="3">Form login belum diisi</td></tr>
<tr><th colspan="2">Aksi Aktor</th><th colspan="3">Reaksi Sistem</th></tr>
<tr><td>1</td><td>Mengisi form login dengan username dan password</td><td>2</td><td colspan="2">Mengecek data pada database, bila data ada dan cocok maka akan masuk ke dalam sistem dan menampilkan menu utama sesuai hak akses. Jika data tidak cocok maka menampilkan pesan bahwa data login salah dan kembali ke form login</td></tr>
<tr><td>3</td><td>Lihat menu utama sesuai hak akses jika validasi berhasil</td><td></td><td colspan="2"></td></tr>
<tr><td colspan="2">Kondisi Akhir</td><td colspan="3">Berhasil masuk ke dalam sistem dan menampilkan menu utama sesuai dengan hak akses.</td></tr>
</table>

2. Lihat Berita

Interaksi antara aktor dengan use case lihat berita dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.3.

Tabel 3. 3 Skenario Use Case Lihat Berita

Skenario *Use Case*			
Identifikasi			
Nama	Lihat Berita		
Tujuan	Melihat berita tentang pendidikan dan memberikan komentar		
Deskripsi	Proses Lihat berita merupakan proses untuk mengetahui berita tentang pendidikan dan mengomentari berita tersebut.		
Aktor	Siswa		
Use Case yang Berkaitan	Tambah berita		
Skenario Utama			
Kondisi Awal	Koneksi ke web server berhasil dan menampilkan list berita		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih list berita	2	Menampilkan detail berita yang dipilih
3	Melihat detail berita		
Kondisi Akhir	Berhasil melihat berita		

3. Lihat Ilmu

Interaksi antara aktor dengan use case lihat ilmu dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.4.

Tabel 3. 4 Skenario Use Case Lihat Ilmu

Skenario *Use Case*	
Identifikasi	
Nama	Lihat Ilmu
Tujuan	Melihat ilmu (*sains)* yang ditulis oleh guru
Deskripsi	Proses lihat ilmu merupakan proses untuk mengetahui ilmu (*sains*) yang ada dalam sistem
Aktor	Siswa
Use Case yang Berkaitan	Tambah ilmu
Skenario Utama	
Kondisi Awal	Koneksi ke web server berhasil dan menampilkan list ilmu

Skenario *Use Case*			
Aksi Aktor		**Reaksi Sistem**	
1	Memilih list ilmu	2	Menampilkan detail ilmu ilmu yang dipilih.
3	melihat detail ilmu		
Kondisi Akhir	Berhasil melihat ilmu		

4. Lihat *Ebook*

Interaksi antara aktor dengan use case *lihat ebook* dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.5.

Tabel 3. 5 Skenario Use Case Lihat Ebook

Skenario *Use Case*			
Identifikasi			
Nama	Lihat *Ebook*		
Tujuan	Melihat dan unduh *ebook* yang ada dalam sistem		
Deskripsi	Proses lihat *ebook* merupakan proses untuk melihat *ebook* yang ada dalam sistem		
Aktor	Siswa		
Use Case yang Berkaitan	Tambah *ebook*, mengelola *ebook*, login		
Skenario Utama			
Kondisi Awal	Koneksi ke web server berhasil, menjalankan menu ebook dan menampilkan kategori *ebook*		
Aksi Aktor		**Reaksi Sistem**	
1	memilih kategori *ebook*	2	Menampilkan daftar *ebook* berdasarkan kategori yang dipilih
3	Memilih daftar *ebook* dan memilih detail *ebook*	4	Menampilkan isi dari *ebook*
Skenario (Unduh *Ebook*)			
1	Memilih daftar *ebook* dan memilih unduh *ebook*	2	Menampilkan data ebook yang diunduh
Kondisi Akhir	Berhasil melihat dan unggah unduh *ebook* yang ada dalam sistem		

5. Pilih Kelas

Interaksi antara aktor dengan use case pilih kelas dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.6.

Tabel 3. 6 Skenario Use Case Pilih Kelas

Skenario *Use Case*			
Identifikasi			
Nama	Pilih Kelas		
Tujuan	Memilih daftar kelas berdasarkan pelajaran		
Deskripsi	Proses pilih kelas merupakan proses untuk bisa memilih kelas sesuai dengan kelas mengajar		
Aktor	Guru		
Use Case yang Berkaitan	Mengelola mengajar, tambah materi, tambah tugas, login		
Skenario Utama			
Kondisi Awal	Login berhasil, kelas mengajar telah ditetapkan dan menampilkan daftar kelas mengajar		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih tambah materi atau tambah tugas	2	Menampilkan detail kelas yang yang dipilih
3	Pilih kelas		
Kondisi Akhir	Berhasil memilih kelas		

6. Tambah Materi

Interaksi antara aktor dengan use case tambah materi dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.7.

Tabel 3. 7 Skenario Use Case Tambah Materi

Skenario *Use Case*	
Identifikasi	
Nama	Tambah Materi
Tujuan	Menambahkan materi ke dalam sistem
Deskripsi	Proses tambah materi merupakan proses penambahan materi ke dalam sistem
Aktor	Guru
Use Case yang Berkaitan	Pilih kelas
Skenario Utama	
Kondisi Awal	Login berhasil dan menjalankan menu tambah materi, kelas mengajar sudah dipilih

Skenario *Use Case*			
Aksi Aktor		**Reaksi Sistem**	
1	Memilih kelas yang akan diberikan materi, tambahkan file, tambahkan deskripsi	2	Mengecek kelengkapan data pada form tambah materi, apabila form tambah data materi sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* materi, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data materi
3	Melihat tugas yang telah ditambahkan pada list tugas jika validasi berhasil		
Kondisi Akhir	Berhasil menambah materi ke dalam sistem		

7. Tambah Tugas

Interaksi antara aktor dengan use case tambah tugas dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.8.

Tabel 3. 8 Skenario Use Case Tambah Tugas

Skenario *Use Case*			
Identifikasi			
Nama	Tambah Tugas		
Tujuan	Menambahkan tugas ke dalam sistem		
Deskripsi	Proses tambah tugas merupakan proses penambahan tugas ke dalam sistem		
Aktor	Guru		
Use Case yang Berkaitan	Mengelola mengajar		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, kelas mengajar sudah dipilih		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih kelas yang akan diberikan tugas, tambahkan file, tambahkan deskripsi	2	Mengecek kelengkapan data pada form tambah tugas, apabila form tambah data tugas sudah lengkap dan benar maka data akan

Skenario *Use Case*			
			ditambahkan dan disimpan dalam *database* tugas, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data tugas
3	Melihat tugas yang telah ditambahkan pada list tugas jika validasi berhasil		
Kondisi Akhir	Berhasil menambah tugas ke dalam sistem		

8. Lihat Ujian

Interaksi antara aktor dengan use case lihat ujian dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.9.

Tabel 3. 9 Skenario Use Case Lihat Ujian

Skenario *Use Case*			
Identifikasi			
Nama	Lihat Ujian		
Tujuan	Melihat detail ujian di dalam sistem		
Deskripsi	Proses lihat ujian merupakan proses menampilkan daftar ujian yang ada di dalam sistem		
Aktor	Guru		
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan daftar ujian		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih daftar ujian	2	Menampilkan detail ujian
3	Melihat detail ujian		
Kondisi Akhir	Berhasil melihat detail ujian di dalam sistem		

9. Tambah Ujian

Interaksi antara aktor dengan use case tambah ujian dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.10.

Tabel 3. 10 Skenario Use Case Tambah Ujian

Skenario *Use Case*			
Identifikasi			
Nama	Tambah ujian		
Tujuan	Menambahkan ujian ke dalam sistem		
Deskripsi	Proses tambah ujian merupakan proses menambahkan sola ujian ke dalam sistem		
Aktor	Guru		
Use Case yang Berkaitan	Lihat ujian		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama		
Aksi Aktor		**Reaksi Sistem**	
1	mengisi form tambah ujian	2	Mengecek kelengkapan data pada form tambah data ujian, apabila form tambah data ujian sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* ujian, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data ujian
3	Melihat data ujian yang telah ditambahkan jika validasi berhasil		
Kondisi Akhir	Berhasil menambahkan ujian ke dalam sistem		

10. Lihat Nilai Ujian

Interaksi antara aktor dengan use case lihat nilai ujian dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.11.

Tabel 3. 11 Skenario Use Case Lihat Nilai Ujian

Skenario *Use Case*	
Identifikasi	
Nama	Lihat nilai ujian
Tujuan	Melihat nilai ujian yang sudah dikerjakan siswa
Deskripsi	Proses lihat nilai ujian merupakan proses menampilkan nilai ujian yang sudah dikerjakan oleh siswa
Aktor	Guru

Skenario *Use Case*			
Use Case yang Berkaitan	Lihat ujian, *dashboard*		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan daftar ujian		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih daftar ujian	2	Menampilkan nilai ujian siswa yang sudah dikerjakan
3	Melihat nilai ujian siswa yang sudah mengerjakan		
Kondisi Akhir	Berhasil melihat nilai ujian yang dikerjakan siswa		

11. Mengelola Berita

Interaksi antara aktor dengan use case mengelola berita dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.12.

Tabel 3. 12 Skenario Use Case Mengelola Berita

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola Berita		
Tujuan	Mengelola semua data yang berkaitan dengan berita		
Deskripsi	Proses mengelola berita merupakan proses mengelola semua data yang berkaitan dengan berita		
Aktor	Admin, guru		
Use Case yang Berkaitan	Login, Lihat Berita		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu berita, menampilkan semua daftar berita		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih menu tambah berita	2	Menampilkan form tambah data berita
3	Mengisi form tambah berita	4	Mengecek kelengkapan data pada form tambah data berita, apabila form tambah data berita sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* berita, jika data tidak lengkap maka

Skenario *Use Case*			
			tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data berita
5	Melihat data berita yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Berita)			
1	Memilih ubah berita	2	Menampilkan form ubah data berita
3	Mengisi form ubah berita	4	Mengecek kelengkapan data pada form ubah data berita, apabila form ubah data ilmu sudah lengkap dan benar maka data akan disimpan dalam *database* berita, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form ubah data berita
5	Melihat data berita yang telah diubah jika validasi berhasil		
Skenario (Hapus Berita)			
1	Memilih hapus berita dari daftar berita	2	Menampilkan pemberitahuan hapus berita
3	Memilih ya	4	Menghapus data ilmu yang tersimpan dalam *database* berita
Skenario (Cari Berita)			
1	Cari berdasarkan judul berita dan isi dalam form cari data berita	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan berita yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data berita tidak ditemukan dan kembali ke form cari data berita
3	Melihat detail berita yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil mengelola semua data ilmu		

12. Mengelola Ilmu

Interaksi antara aktor dengan use case mengelola ilmu dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.13.

Tabel 3. 13 Skenario Use Case Ilmu

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola Ilmu		
Tujuan	Mengelola semua data yang berkaitan dengan Ilmu		
Deskripsi	Proses mengelola ilmu merupakan proses mengelola semua data yang berkaitan dengan Ilmu		
Aktor	Admin, guru		
Use Case yang Berkaitan	Login, Lihat Ilmu		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu ilmu , menampilkan semua daftar ilmu		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih menu tambah ilmu	2	Menampilkan form tambah data ilmu
3	Mengisi form tambah ilmu	4	Mengecek kelengkapan data pada form tambah data ilmu, apabila form tambah data ilmu sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* ilmu, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data ilmu
5	Melihat data ilmu yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Ilmu)			
1	Memilih ubah ilmu	2	Menampilkan form ubah data ilmu
3	Mengisi form ubah ilmu	4	Mengecek kelengkapan data pada form ubah data ilmu, apabila form ubah data ilmu sudah lengkap dan benar maka data akan disimpan dalam *database* ilmu, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form ubah data ilmu
5	Melihat data ilmu yang telah diubah jika validasi berhasil		
Skenario (Hapus Ilmu)			

Skenario *Use Case*			
1	Memilih hapus ilmu dari daftar ilmu	2	Menampilkan pemberitahuan hapus ilmu
3	Memilih ya	4	Menghapus data ilmu yang tersimpan dalam *database* ilmu
Skenario (Cari Ilmu)			
1	Cari berdasarkan judul ilmu dan isi dalam form cari data ilmu	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan ilmu yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data ilmu tidak ditemukan dan kembali ke form cari data ilmu
3	Melihat detail ilmu yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil mengelola semua data ilmu		

13. Mengelola *Ebook*

Interaksi antara aktor dengan use case mengelola *ebook* dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.14.

Tabel 3. 14 Skenario Use Case Mengelola Ebook

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola *ebook*		
Tujuan	Mengelola semua data yang berkaitan dengan *ebook*		
Deskripsi	Proses mengelola *ebook* merupakan proses mengelola semua data yang berkaitan dengan *ebook*		
Aktor	Admin, guru		
Use Case yang Berkaitan	Login, lihat *ebook*		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan semua daftar *ebook*		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih menu tambah *ebook*	2	Menampilkan form tambah data *ebook*
3	Mengisi form tambah *ebook*	4	Mengecek kelengkapan data pada form tambah data *ebook*, apabila form tambah data *ebook* sudah lengkap dan benar maka data

Skenario *Use Case*			
			akan ditambahkan dan disimpan dalam *database ebook*, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data *ebook*
5	Melihat data *ebook* yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah *Ebook*)			
1	Memilih ubah *ebook*	2	Menampilkan form ubah data *ebook*
3	Mengisi form ubah *ebook*	4	Mengecek kelengkapan data pada form ubah data *ebook*, apabila form ubah data *ebook* sudah lengkap dan benar maka data akan disimpan dalam *database ebook*, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form ubah data *ebook*
5	Melihat data *ebook* yang telah diubah jika validasi berhasil		
Skenario (Hapus *Ebook*)			
1	Memilih hapus *ebook* dari daftar *ebook*	2	Menampilkan pemberitahuan hapus *ebook*
3	Memilih ya	4	Menghapus data *ebook* yang tersimpan dalam *database ebook*
Skenario (Cari *Ebook*)			
1	Cari berdasarkan judul *ebook* da nisi dalam form cari data ebook	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan *ebook* yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data *ebook* tidak ditemukan dan kembali ke form cari data *ebook*
3	Melihat detail *ebook* yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil mengelola semua data *ebook*		

14. Mengelola Mengajar

Interaksi antara aktor dengan use case mengelola mengajar dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.15.

Tabel 3. 15 Skenario Use Case Mengelola Kelas

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola Mengajar		
Tujuan	Mengelola semua data yang berkaitan dengan mengajar		
Deskripsi	Proses mengelola kelas merupakan proses mengelola semua data yang berkaitan dengan kelas		
Aktor	Admin		
Use Case yang Berkaitan	Login, pilih kelas		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan daftar mengajar guru yang sudah ada		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih menu tambah mengajar	2	Menampilkan form tambah data mengajar
3	Mengisi form tambah mengajar	4	Mengecek kelengkapan data pada form tambah data mengajar, apabila form tambah data mengaja sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* mengajar, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data mengajar
5	Melihat data mengajar yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Mengajar)			
1	Memilih ubah mengajar	2	Menampilkan form ubah data mengajar
3	Mengisi form ubah mengajar	4	Mengecek kelengkapan data pada form ubah data mengajar, apabila form ubah data mengajar sudah lengkap dan benar maka data akan disimpan dalam *database* mengajar, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap

Skenario *Use Case*			
			di form ubah data mengajar
5	Melihat data mengajar yang telah diubah jika validasi berhasil		
Skenario (Hapus Mengajar)			
1	Memilih hapus mengajar dari daftar mengajar	2	Menampilkan pemberitahuan hapus mengajar
3	Memilih ya	4	Menghapus data mengajar yang tersimpan dalam *database* mengajar
Skenario (Cari Mengajar)			
1	Cari berdasarkan nama guru dan isi dalam form cari data mengajar	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan mengajar yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data mengajar tidak ditemukan dan kembali ke form cari data mengajar
3	Melihat detail mengajar yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil mengelola data mengajar		

15. Mengelola Data Siswa

Interaksi antara aktor dengan use case mengelola data siswa dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.16.

Tabel 3. 16 Skenario Use Case Mengelola Data Siswa

Skenario *Use Case*	
Identifikasi	
Nama	Mengelola Data Siswa
Tujuan	Mengelola semua data yang berkaitan dengan data siswa
Deskripsi	Proses mengelola data siswa merupakan proses mengelola semua data yang berkaitan dengan siswa
Aktor	Admin
Use Case yang Berkaitan	Login
Skenario Utama	
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan semua

Skenario *Use Case*			
	daftar siswa		
Aksi Aktor		**Reaksi Sistem**	
1	pilih *import* data siswa	2	menampilkan pemberitahuan *import* data siswa
3	Pilih ya	4	Menyimpan data siswa ke dalam *database* dan menampilkannya
5	Melihat data siswa yang sudah di *import*		
Skenario (Tambah Siswa)			
1	Memilih menu tambah siswa	2	Menampilkan form tambah data siswa
3	Mengisi form tambah siswa	4	Mengecek kelengkapan data pada form tambah data siswa, apabila form tambah data siswa sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* siswa, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data siswa
5	Melihat data siswa yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Siswa)			
1	Memilih ubah siswa dari daftar siswa	2	Menampilkan form ubah data siswa
3	Mengisi form ubah siswa	4	Mengecek kelengkapan data pada form ubah data siswa, apabila form ubah data siswa sudah lengkap dan benar maka data telah diubah dan disimpan dalam *database* siswa
5	Melihat data siswa yang telah diubah		
Skenario (Hapus Siswa)			
1	Memilih hapus siswa dari daftar siswa	2	Menampilkan pemberitahuan hapus siswa
3	Memilih ya	4	Menghapus data siswa yang tersimpan dalam *database* siswa
Skenario (Cari Siswa)			
1	Cari berdasarkan nama siswa dan isi dalam form cari data siswa	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan siswa yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data siswa tidak

Skenario *Use Case*			
			ditemukan dan kembali ke form cari data siswa
3	Melihat detail siswa yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil mengelola data siswa		

16. Mengelola Data Guru

Interaksi antara aktor dengan use case mengelola data guru dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.17.

Tabel 3. 17 Skenario Use Case Mengelola Data Guru

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola Data Guru		
Tujuan	Mengelola semua data yang berkaitan dengan data guru		
Deskripsi	Proses mengelola data guru merupakan proses mengelola semua data yang berkaitan dengan guru		
Aktor	Admin		
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan semua daftar guru		
Aksi Aktor		**Reaksi Sistem**	
1	pilih tambah guru	2	Menampilkan form tambah data guru
3	Mengisi form tambah guru	4	Mengecek kelengkapan data pada form tambah data guru, apabila form tambah data guru sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* guru, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data guru
5	Melihat data guru yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Guru)			

Skenario *Use Case*			
1	Memilih ubah guru dari daftar guru	2	Menampilkan form ubah data guru
3	Mengisi form ubah guru	4	Mengecek kelengkapan data pada form ubah data guru, apabila form ubah data guru sudah lengkap dan benar maka data telah diubah dan disimpan dalam *database* guru
5	Melihat data guru yang telah diubah		
Skenario (Hapus Guru)			
1	Memilih hapus guru dari daftar guru	2	Menampilkan pemberitahuan hapus guru
3	Memilih ya	4	Menghapus data guru yang tersimpan dalam *database* guru
Skenario (Cari Guru)			
1	Cari berdasarkan nama guru dan isi dalam form cari data guru	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan guru yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data guru tidak ditemukan dan kembali ke form cari data guru
3	Melihat detail guru yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil mengelola data guru		

17. Mengelola Data Kelas

Interaksi antara aktor dengan use case mengelola data kelas dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.18.

Tabel 3. 18 Skenario Use Case Mengelola Data Kelas

Skenario *Use Case*	
Identifikasi	
Nama	Mengelola Data Kelas
Tujuan	Mengelola semua data yang berkaitan dengan data kelas
Deskripsi	Proses mengelola data kelas merupakan proses mengelola semua data yang berkaitan dengan kelas
Aktor	Admin
Use Case yang Berkaitan	Login
Skenario Utama	

Skenario *Use Case*			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan semua daftar kelas		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih tambah kelas	2	Menampilkan form tambah data kelas
3	Mengisi form tambah kelas	4	Mengecek kelengkapan data pada form tambah data kelas, apabila form tambah data kelas sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* kelas, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data kelas
5	Melihat data kelas yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Kelas)			
1	Memilih ubah kelas dari daftar kelas	2	Menampilkan form ubah data kelas
3	Mengisi form ubah kelas	4	Mengecek kelengkapan data pada form ubah data kelas, apabila form ubah data kelas sudah lengkap dan benar maka data telah diubah dan disimpan dalam *database* kelas
5	Melihat data kelas yang telah diubah		
Skenario (Hapus Kelas)			
1	Memilih hapus kelas dari daftar kelas	2	Menampilkan pemberitahuan hapus kelas
3	Memilih ya	4	Menghapus data kelas yang tersimpan dalam *database* kelas
Skenario (Cari Kelas)			
1	Cari berdasarkan nama kelas dan isi dalam form cari data kelas	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan kelas yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data kelas tidak ditemukan dan kembali ke form cari data kelas
3	Melihat detail kelas yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil mengelola data kelas		

18. Mengelola Data Pelajaran

Interaksi antara aktor dengan use case mengelola data pelajaran dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.19.

Tabel 3. 19 Skenario Use Case Mengelola Data Pelajaran

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola Data Pelajaran		
Tujuan	Mengelola semua data yang berkaitan dengan data pelajaran		
Deskripsi	Proses mengelola data pelajaran merupakan proses mengelola semua data yang berkaitan dengan pelajaran		
Aktor	Admin		
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan semua daftar pelajaran		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih tambah kelas	2	Menampilkan form tambah data pelajaran
3	Mengisi form tambah pelajaran	4	Mengecek kelengkapan data pada form tambah data pelajaran, apabila form tambah data pelajaran sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* pelajaran, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data pelajaran
5	Melihat data pelajaran yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Pelajaran)			
1	Memilih ubah pelajaran dari daftar pelajaran	2	Menampilkan form ubah data pelajaran
3	Mengisi form ubah pelajaran	4	Mengecek kelengkapan data pada form ubah data pelajaran, apabila form ubah data pelajaran sudah lengkap dan benar maka data telah diubah dan disimpan dalam *database* pelajaran

Skenario *Use Case*			
5	Melihat data pelajaran yang telah diubah		
Skenario (Hapus Pelajaran)			
1	Memilih hapus pelajaran dari daftar pelajaran	2	Menampilkan pemberitahuan hapus pelajaran
3	Memilih ya	4	Menghapus data pelajaran yang tersimpan dalam *database* pelajaran
Kondisi Akhir	Berhasil mengelola data pelajaran		

19. Mengelola Tahun Ajaran

Interaksi antara aktor dengan use case tahun ajaran dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.20.

Tabel 3. 20 Skenario Use Case Mengelola Tahun Ajaran

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola Tahun Ajaran		
Tujuan	Mengelola data yang berkaitan dengan data tahun ajaran		
Deskripsi	Proses mengelola data tahun ajaran merupakan proses mengelola semua data yang berkaitan dengan tahun ajaran		
Aktor	Admin		
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih tambah data tahun ajaran	2	Menampilkan form tambah data tahun ajaran
3	Menambah tahun ajaran	4	Mengecek kelengkapan data pada form tambah data tahun ajaran, apabila form tambah data tahun ajaran sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* tahun ajaran, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data tahun ajaran

Skenario *Use Case*			
5	Melihat data tahun ajaran yang telah ditambahkan jika validasi berhasil		
Skenario (Hapus Tahun Ajaran)			
1	Memilih hapus tahun ajaran dari daftar tahun ajaran	2	Menampilkan pemberitahuan hapus tahun ajaran
3	Memilih ya	4	Menghapus data tahun ajaran yang tersimpan dalam *database* tahun ajaran
Kondisi Akhir	Berhasil mengelola data tahun ajaran		

20. Kirim Pesan

Interaksi antara aktor dengan use case kirim pesan dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.21.

Tabel 3. 21 Skenario Use Case Kirim Pesan

Skenario *Use Case*			
Identifikasi			
Nama	Kirim pesan		
Tujuan	Mengelola data pesan antara guru dan siswa		
Deskripsi	Proses kirim pesan merupakan proses mengelola semua data yang berkaitan dengan pesan		
Aktor	Guru , siswa		
Use Case yang Berkaitan	Lihat siswa, lihat guru, login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih daftar guru atau siswa	2	Menampilkan menampilkan form pesan
3	Melakukan kirim pesan kepada guru atau siswa	4	Menampilkan percakapan
5	Melihat percakapan pesan	6	Percakapan pesan disimpan dalam *database* pesan
Skenario (Hapus Pesan)			
1	Memilih hapus pesan dari daftar pesan	2	Menampilkan pemberitahuan hapus pesan
3	Memilih ya	4	Menghapus data pesan yang tersimpan dalam *database* pesan

Skenario *Use Case*	
Kondisi Akhir	Berhasil mengelola data pesan antara guru dan siswa

21. Lihat Siswa

Interaksi antara aktor dengan use case lihat siswa dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.22.

Tabel 3. 22 Skenario Use Case Lihat Siswa

Skenario *Use Case*			
Identifikasi			
Nama	Lihat siswa		
Tujuan	Melihat semua data siswa yang ada di dalam sistem		
Deskripsi	Proses lihat data siswa merupakan proses untuk bisa melihat semua data siswa yang ada di dalam sistem		
Aktor	Guru, siswa		
Use Case yang Berkaitan	Kirim pesan, login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan daftar siswa		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih daftar siswa	2	Menampilkan profil siswa
3	melihat profil siswa		
Skenario (Cari Siswa)			
1	Cari berdasarkan nama siswa dan isi dalam form cari data siswa	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan siswa yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data siswa tidak ditemukan dan kembali ke form cari data siswa
3	Melihat profil siswa yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil melihat semua data siswa yang ada di dalam sistem		

22. Lihat Guru

Interaksi antara aktor dengan use case lihat guru dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.23.

Tabel 3. 23 Skenario Use Case Lihat Guru

Skenario *Use Case*			
Identifikasi			
Nama	Lihat guru		
Tujuan	Melihat semua data guru yang ada di dalam sistem		
Deskripsi	Proses lihat data guru merupakan proses untuk bisa melihat semua data guru yang ada di dalam sistem		
Aktor	Guru, siswa		
Use Case yang Berkaitan	Kirim pesan, login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan daftar guru		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih daftar guru	2	Menampilkan profil guru
3	Melihat profil guru		
Skenario (Cari Guru)			
1	Cari berdasarkan nama guru dan isi dalam form cari data guru	2	Mengecek ketersediaan data, apabila data tersedia maka menampilkan guru yang dicari, jika data tidak tersedia maka tampilkan pesan bahwa data guru tidak ditemukan dan kembali ke form cari data guru
3	Melihat profil guru yang dicari jika data ditemukan		
Kondisi Akhir	Berhasil melihat semua data guru yang ada di dalam sistem		

23. Masuk Kelas

Interaksi antara aktor dengan use case masuk kelas dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.24.

Tabel 3. 24 Skenario Use Case Masuk Kelas

Skenario *Use Case*	
Identifikasi	
Nama	Masuk kelas

Skenario *Use Case*			
Tujuan	Melakukan semua proses yang ada di dalam kelas		
Deskripsi	Proses masuk kelas merupakan proses untuk bisa melakukan kegiatan pembelajaran yang ada di dalam sistem		
Aktor	Siswa		
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan daftar kelas		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih kelas	2	Menampilkan detail kelas yang dipilih siswa
3	Masuk ke dalam kelas		
Skenario (Melihat Materi)			
1	Memilih menu materi	2	Menampilkan daftar materi
3	Lihat daftar materi dan pilih detail materi	4	Menampilkan *unduh* materi
5	*Unduh* materi		
Skenario (Melihat Tugas)			
1	Memilih menu tugas	2	Menampilkan daftar tugas
3	Lihat daftar tugas dan pilih detail tugas	4	Menampilkan detail tugas
5	Mimilih *unggah* tugas dan mengisi form *unggah* tugas	6	Mengecek kelengkapan data pada form tambah data *unggah* tugas, apabila form tambah data *unggah* tugas sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* tugas, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data *unggah* tugas
7	Melihat tugas yang ter*unggah* jika validasi berhasil		
Kondisi Akhir	Berhasil melakukan semua proses yang ada di dalam kelas		

24. Mengikuti Ujian

Interaksi antara aktor dengan use case mengikuti ujian dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.25.

Tabel 3. 25 Skenario Use Case Mengikuti Ujian

Skenario *Use Case*			
Identifikasi			
Nama	Mengikuti Ujian		
Tujuan	Melakukan semua proses ujian		
Deskripsi	Proses mengikuti ujian merupakan proses untuk bisa melakukan kegiatan ujian		
Aktor	Siswa		
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan daftar ujian		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih nama ujian	2	Menampilkan detail ujian yang dipilih siswa
3	Melihat detail ujian		
Skenario (Mengisi Ujian)			
1	Melihat soal ujian	2	Menampilkan form soal ujian
3	Mengisi mengisi soal ujian dan menyelesaikan soal ujian	4	Pemeriksaan jawaban soal ujian yang diselesaikan siswa dan penilaian jawaban soal ujian
5	Melihat nilai jawaban soal ujian yang telah dikerjakan	6	Menampilkan nilai jawaban soal ujian
Kondisi Akhir	Berhasil melakukan semua proses ujian		

25. Mengikuti *Polling*

Interaksi antara aktor dengan use case mengikuti *polling* dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.26.

Tabel 3. 26 Skenario Use Case Mengikuti Polling

Skenario *Use Case*	
Identifikasi	
Nama	Mengikuti *polling*
Tujuan	Melakukan *polling* berdasarkan topic
Deskripsi	Proses mengikuti *polling* merupakan proses untuk bisa meberikan pendapat tentang topik
Aktor	Siswa, guru

Skenario *Use Case*			
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	login berhasil menampilkan menu utama		
Aksi Aktor		**Reaksi Sistem**	
1	Memberikan pendapat	2	Menampilkan pilihan pendapat
3	Memilih pendapat	4	Menampilkan hasil *polling*
5	Melihat hasil Polling		
Kondisi Akhir	Berhasil melakukan *polling* berdasarkan topik		

26. Mengelola *Polling*

Interaksi antara aktor dengan use case mengelola *polling* dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.27.

Tabel 3. 27 Skenario Use Case Mengelola Polling

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola polling		
Tujuan	Mengelola semua data yang berkaitan dengan polling		
Deskripsi	Proses mengelola polling merupakan proses mengelola semua data yang berkaitan dengan polling		
Aktor	Admin		
Use Case yang Berkaitan	Mengikuti polling, login		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan semua daftar polling		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih menu tambah polling	2	Menampilkan form tambah data polling
3	Mengisi form tambah polling	4	Mengecek kelengkapan data pada form tambah data polling, apabila form tambah data polling sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* polling, apabila belum lengkap tampilkan pesan dan tetap berada di form

Skenario *Use Case*			
			tambah data polling
5	Melihat data polling yang telah ditambahkan		
Skenario (Ubah Polling)			
1	Memilih ubah polling dari daftar polling	2	Menampilkan form ubah data polling
3	Mengisi form ubah polling	4	Mengecek kelengkapan data pada form ubah data polling, apabila form ubah data polling sudah lengkap dan benar maka data telah diubah dan disimpan dalam *database* polling, apabila belum lengkap tampilkan pesan dan tetap berada di form tambah data polling
5	Melihat data polling yang telah diubah		
Skenario (Hapus Polling)			
1	Memilih hapus kelas dari daftar polling	2	Menampilkan pemberitahuan hapus polling
3	Memilih ya	4	Menghapus data pelajaran yang tersimpan dalam *database* polling
Kondisi Akhir	Berhasil mengelola semua data pelajaran		

27. Lihat Grafik Nilai

Interaksi antara aktor dengan use case lihat grafik nilai dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.28.

Tabel 3. 28 Skenario Use Case Lihat Grafik Nilai

Skenario *Use Case*	
Identifikasi	
Nama	Lihat Grafik Nilai
Tujuan	Melihat grafik nilai rata-rata ujian siswa
Deskripsi	Proses lihat grafik nilai merupakan proses untuk melihat grafik nilai rata-rata ujian siswa
Aktor	guru
Use Case yang Berkaitan	Login, lihat nilai ujian
Skenario Utama	

Skenario *Use Case*			
Kondisi Awal	Login berhasil		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih memilih menu kelas mengajar	2	Menampilkan list kelas mengajar dan menu lihat grafik ujian.
4	Pilih menu lihat grafik nilai	3	Menampilkan grafik nilai rata-rata ujian
5	Lihat grafik nilai rata-rata ujian siswa		
Kondisi Akhir	Berhasil melihat grafik nilai rata-rata ujian siswa		

28. Logout

Interaksi antara aktor dengan use case logout dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.29.

Tabel 3. 29 Skenario Use Case Logout

Skenario *Use Case*			
Identifikasi			
Nama	Logout		
Tujuan	Keluar dari sistem		
Deskripsi	Proses logout merupakan proses untuk bisa keluar dari sistem		
Aktor	Admin, guru, murid		
Use Case yang Berkaitan	Login		
Skenario Utama			
Kondisi Awal	Login berhasil		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih tombol logout	2	Mengeluarkan aktor dari sistem
4	Keluar dari sistem	3	Menampilkan beranda diluar sistem
Kondisi Akhir	Aktor keluar dari sistem		

29. Mengelola Pengumuman

Interaksi antara aktor dengan use case mengelola pengumuman dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.30.

Tabel 3. 30 Skenario Use Case Pengumuman

Skenario *Use Case*			
Identifikasi			
Nama	Mengelola Pengumuman		
Tujuan	Mengelola semua data yang berkaitan dengan pengumuman		
Deskripsi	Proses mengelola pengumuman merupakan proses mengelola semua data yang berkaitan dengan pengumuman		
Aktor	Admin, guru		
Use Case yang Berkaitan	Login, Lihat Pengumuman		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama, menampilkan semua daftar pengumuman		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih menu tambah pengumuman	2	Menampilkan form tambah data pengumuman
3	Mengisi form tambah pengumuman	4	Mengecek kelengkapan data pada form tambah data pengumuman, apabila form tambah data pengumuman sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* pengumuman, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah data pengumuman
5	Melihat data pengumuman yang telah ditambahkan jika validasi berhasil		
Skenario (Ubah Pengumuman)			
1	Memilih ubah pengumuman dari daftar pengumuman	2	Menampilkan form ubah data pengumuman
3	Mengisi form ubah pengumuman	4	Mengecek kelengkapan data pada form ubah data pengumuman, apabila form ubah data pengumuman sudah lengkap dan benar maka data akan disimpan dalam *database* pengumuman, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form ubah data

Skenario *Use Case*			
			pengumuman
5	Melihat data pengumuman yang telah diubah jika validasi berhasil		
Skenario (Hapus Pengumuman)			
2	Memilih hapus pengumuman dari daftar pengumuman	3	Menampilkan pemberitahuan hapus pengumuman
4	Memilih ya	5	Menghapus data pengumuman yang tersimpan dalam *database* pengumuman
Kondisi Akhir	Berhasil mengelola semua data pengumuman		

30. Lihat Pengumuman

Interaksi antara aktor dengan use case lihat pengumuman dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.31.

Tabel 3. 31 Skenario Use Case Lihat Pengumuman

Skenario *Use Case*			
Identifikasi			
Nama	Lihat Pengumuman		
Tujuan	Melihat semua data guru yang ada di dalam sistem		
Deskripsi	Proses lihat data guru merupakan proses untuk bisa melihat semua data guru yang ada di dalam sistem		
Aktor	siswa		
Use Case yang Berkaitan	-		
Skenario Utama			
Kondisi Awal	Login berhasil dan menjalankan menu utama		
Aksi Aktor		**Reaksi Sistem**	
		1	Menampilkan daftar pengumuman
2	Memilih pengumuman	3	Menampilkan detail pengumuman
4	lihat detail data pengumuman		
Kondisi Akhir	Berhasil melihat semua data guru yang ada di dalam sistem		

31. Komentar Topik Forum

Interaksi antara aktor dengan use case komentar topik forum dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.32.

Tabel 3. 32 Skenario Use Case Komentar Forum

Skenario *Use Case*			
Identifikasi			
Nama	Komentar Topik Forum		
Tujuan	Mengelola semua data yang berkaitan dengan forum		
Deskripsi	Proses buat topik forum merupakan proses mengelola tambah data yang berkaitan dengan forum		
Aktor	Guru, siswa		
Use Case yang Berkaitan	Login, BuatTopik Forum		
Skenario Utama			
Kondisi Awal	Sudah login, menampilkan data forum		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih daftar forum	2	Menampilkan detail forum
3	Memilih komentari topik forum	4	Menampilkan form komentari topik forum
5	Melakukan komentari topik forum dan mengisi form komentari topik forum	6	Mengecek kelengkapan data pada form komentari topik forum, apabila form komentari topik forum sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* forum, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form komentari topik forum
7	Melihat forum yang telah dikomentari jika validasi berhasil		
Skenario (Hapus Forum)			
1	Pilih hapus data komentari topik forum	2	Konfirmasi hapus data komentari topik forum
3	Pilih ya	4	Hapus data komentar topik forum
Kondisi Akhir	Berhasil mengelola semua forum		

32. Buat Topik Forum

Interaksi antara aktor dengan use case buat topik forum dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.33.

Tabel 3. 33 Skenario Use Case Buat Topik Forum

Skenario *Use Case*			
Identifikasi			
Nama	Buat Topik Forum		
Tujuan	Mengelola tambah data topik forum		
Deskripsi	Proses buat topik forum merupakan proses mengelola tambah data yang berkaitan dengan forum		
Aktor	Guru, siswa		
Use Case yang Berkaitan	Login, BuatTopik Forum		
Skenario Utama			
Kondisi Awal	Sudah login, menampilkan data forum		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih tambah topik forum	2	Menampilkan form tambah topik forum
3	Melakukan tambah topik forum dan mengisi form tambah topik forum	4	Mengecek kelengkapan data pada form tambah topik forum, apabila form tambah topik forum sudah lengkap dan benar maka data akan ditambahkan dan disimpan dalam *database* forum, jika data tidak lengkap maka tampilkan pesan bahwa data salah atau tidak lengkap dan tetap di form tambah topik forum
5	Melihat forum yang telah ditambah jika validasi berhasil		
Skenario (Hapus Forum)			
1	Pilih hapus data tambah topik forum	2	Konfirmasi hapus data tambah topik forum
3	Pilih ya	4	Hapus data tambah topik forum
Kondisi Akhir	Berhasil mengelola tambah topik forum		

33. Lihat Hasil Polling

Interaksi antara aktor dengan use case lihat hasil polling dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.34.

Tabel 3. 34 Skenario Use Case Lihat Hasil polling

Skenario *Use Case*			
Identifikasi			
Nama	Lihat Hasil Polling		
Tujuan	Melihat hasil polling		
Deskripsi	Proses lihat hasil polling merupakan proses untuk bisa melihat hasil polling dari daftar polling yang ada		
Aktor	Guru, Siswa		
Use Case yang Berkaitan			
Skenario Utama			
Kondisi Awal	Pertanyaan polling sudah dijawab		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih menu lihat hasil polling	2	Menampilkan hasil Polling
3	Lihat hasil polling		
Kondisi Akhir	Berhasil melihat hasil polling		

34. Lupa Password

Interaksi antara aktor dengan use case lupa password dijelaskan dalam *use case scenario* yang dapat dilihat pada Tabel 3.35.

Tabel 3. 35 Skenario Use Case Lupa Password

Skenario *Use Case*			
Identifikasi			
Nama	Lupa Password		
Tujuan	Mengirim tautan ganti password dan mengganti password dengan yang baru		
Deskripsi	Proses lupa password merupakan proses untuk bisa mengirim tautan ganti password yang dikirimkan lewat email pengguna		
Aktor	Admin, Guru, Siswa		
Use Case yang Berkaitan			
Skenario Utama			
Kondisi Awal	Pengguna lupa password untuk login, menampilkan form lupa password		
Aksi Aktor		**Reaksi Sistem**	
1	Memilih jenis user dan mengisi	2	Validasi ketersediaan email. Apabila tersedia

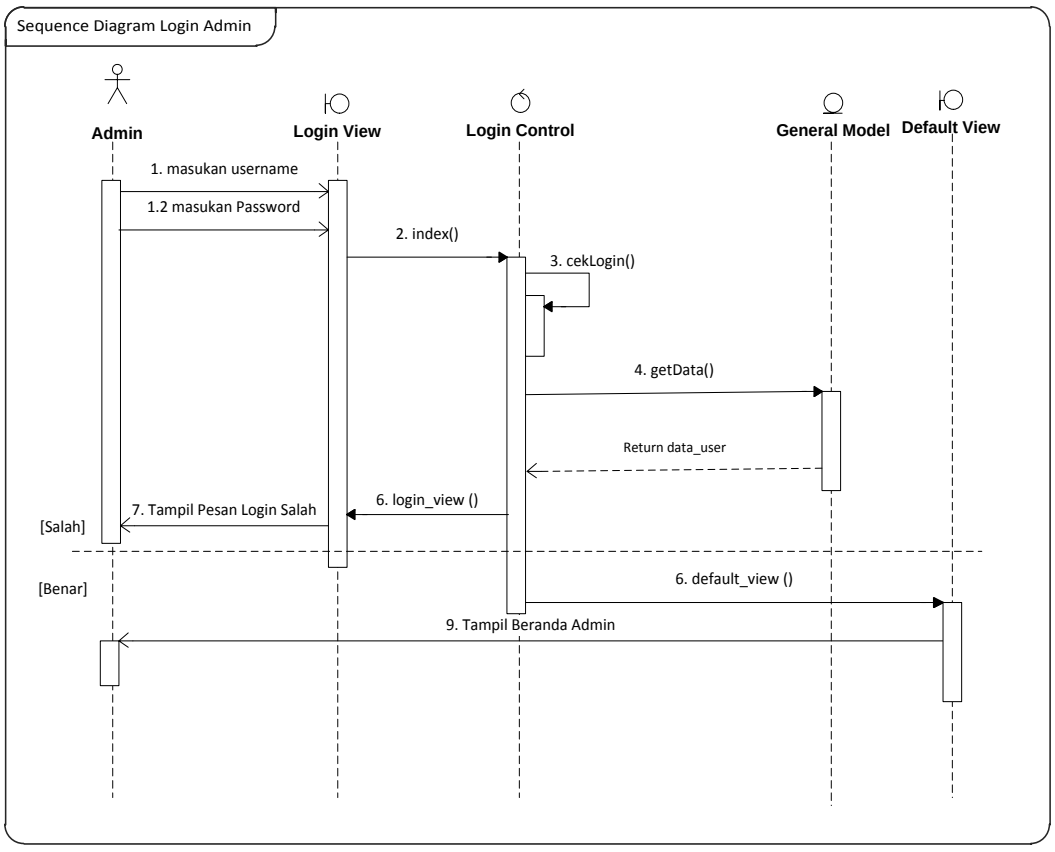
Skenario *Use Case*			
	username		maka kirim tautan ganti password ke email yang terdaftar.
3	Menerima tautan ganti password melakukan penggantian password.	4	Masukan password baru ke database pengguna
Kondisi Akhir	Berhasil mengirim tautan ganti password dan mengganti password dengan yang baru		

3.1.7.6 *Sequence* Diagram

Sequence diagram adalah grafik dua dimensi dimana objek ditunjukkan dalam dimensi horizontal, sedangkan *lifeline* ditunjukkan dalam dimensi vertikal. *Sequence diagram* dari use case *e-learning* di SMA Negeri 3 Sumedang adalah sebagai berikut:

1. Sequence Diagram Login Admin

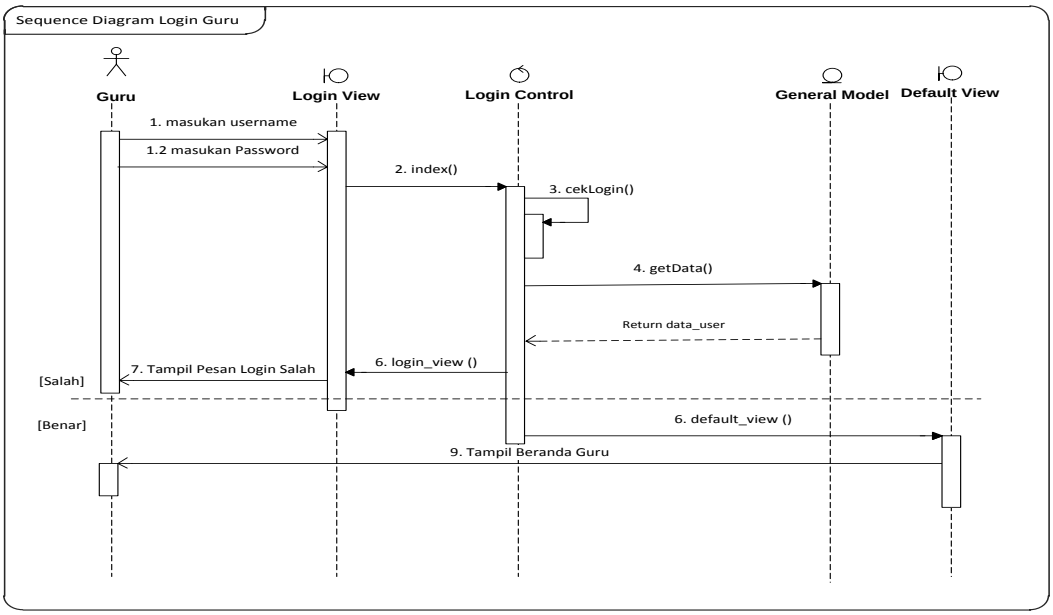
Sequence diagram login admin dapat dilihat pada Gambar 3.10.



Gambar 3. 10 Sequence Diagram Login Admin

2. Sequence Diagram Login Guru

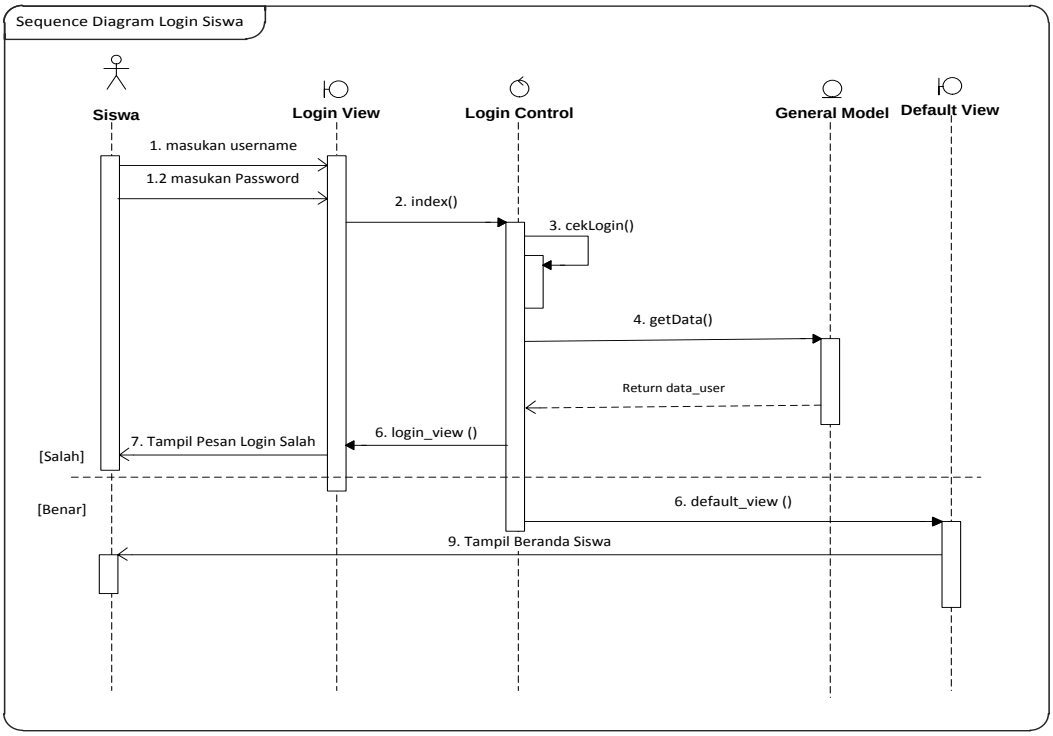
Sequence diagram guru dapat dilihat pada Gambar 3.11.



Gambar 3. 11 Sequence Diagram Login Guru

3. Sequence Diagram Login Siswa

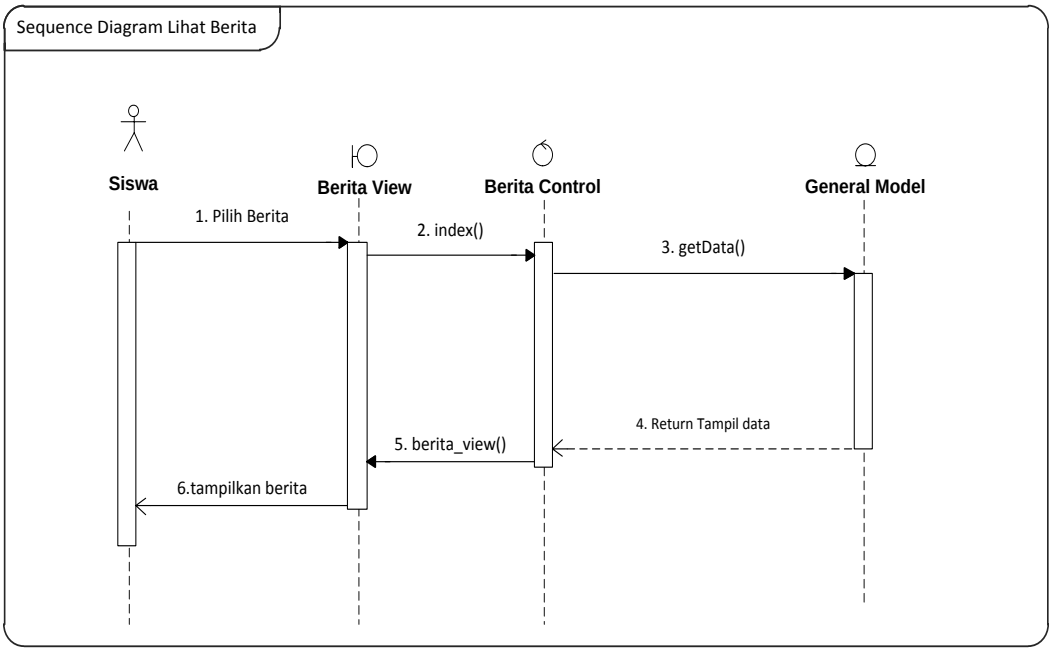
Sequence diagram siswa dapat dilihat pada Gambar 3.12.



Gambar 3. 12 Sequence Diagram Login Siswa

4. Sequence Diagram Lihat Berita

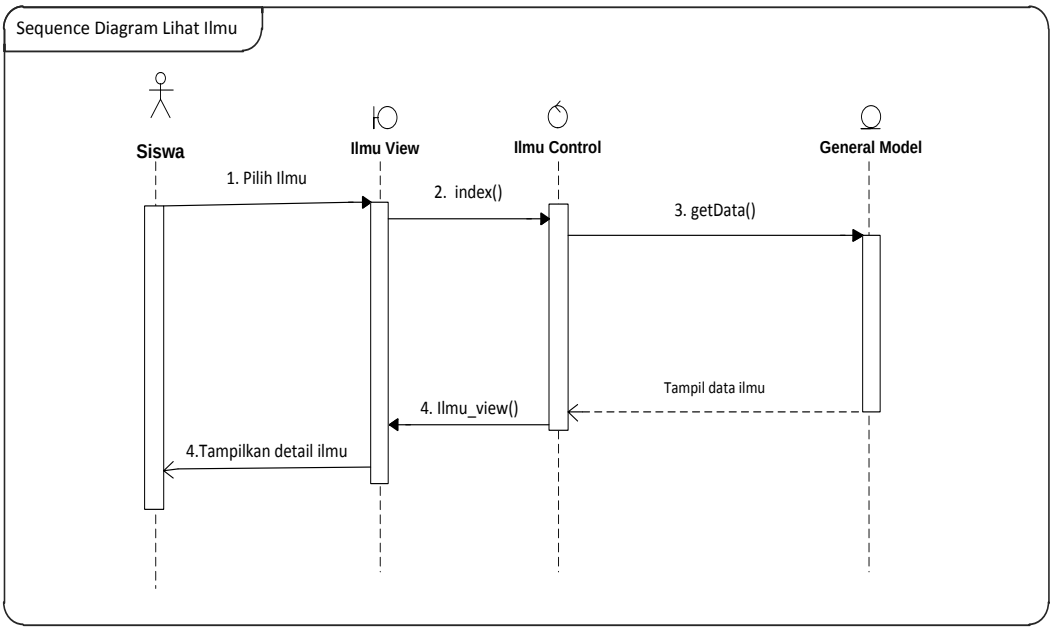
Sequence diagram lihat berita dapat dilihat pada Gambar 3.13.



Gambar 3. 13 Sequence Diagram Lihat Berita

5. Sequence Diagram Lihat Ilmu

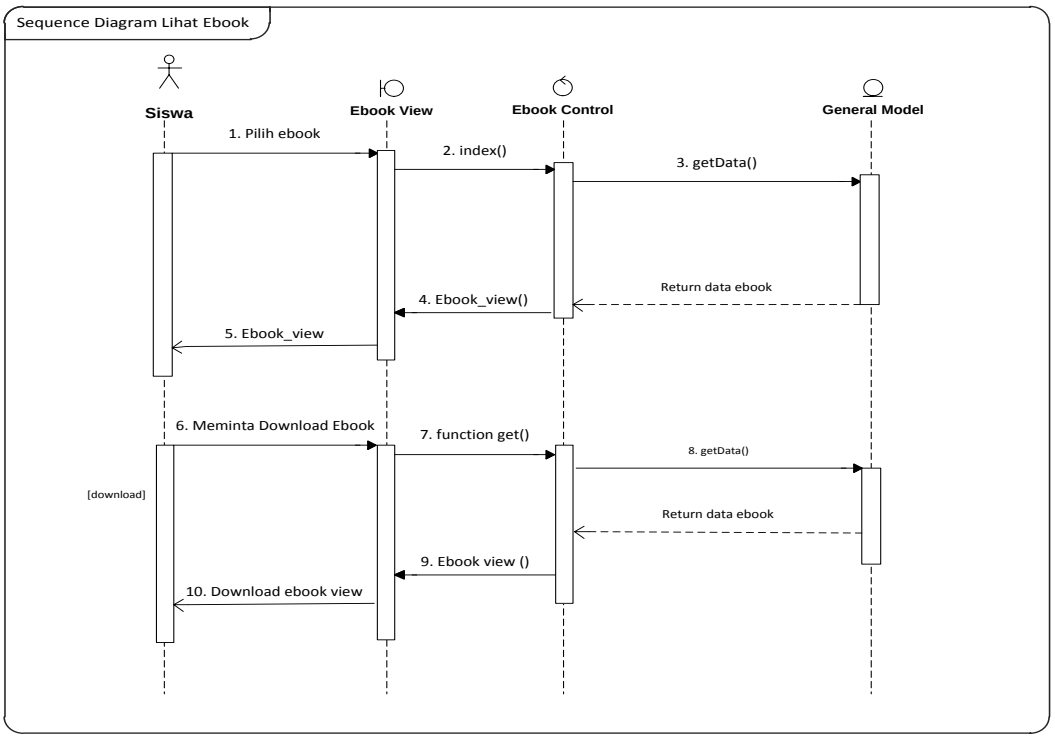
Sequence diagram lihat ilmu dapat dilihat pada Gambar 3.14.



Gambar 3. 14 Sequence Diagram Lihat Ilmu

6. Sequence Diagram Lihat Ebook

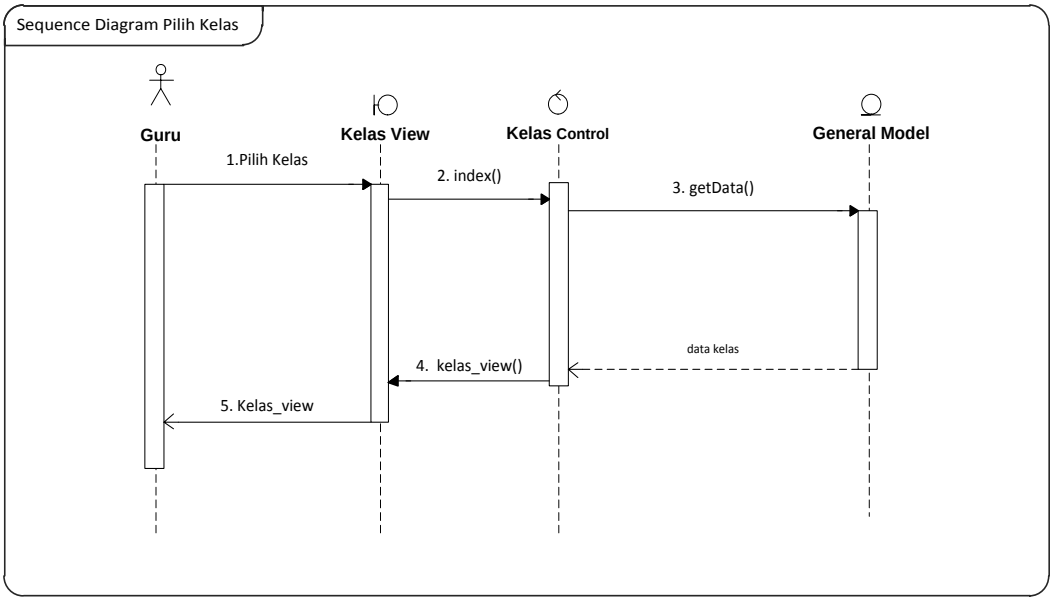
Sequence diagram lihat ebook dapat dilihat pada Gambar 3.15.



Gambar 3. 15 Sequence Diagram Lihat Ebook

7. Sequence Diagram Pilih Kelas

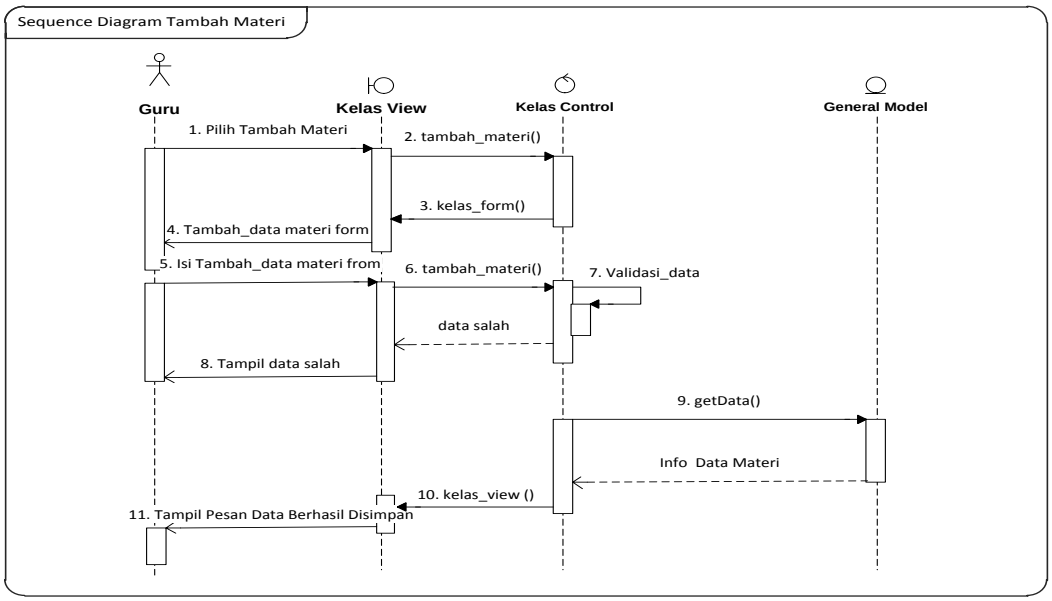
Sequence diagram lihat kelas dapat dilihat pada Gambar 3.16.



Gambar 3. 16 Sequence Diagram Pilih Kelas

8. Sequence Diagram Tambah Materi

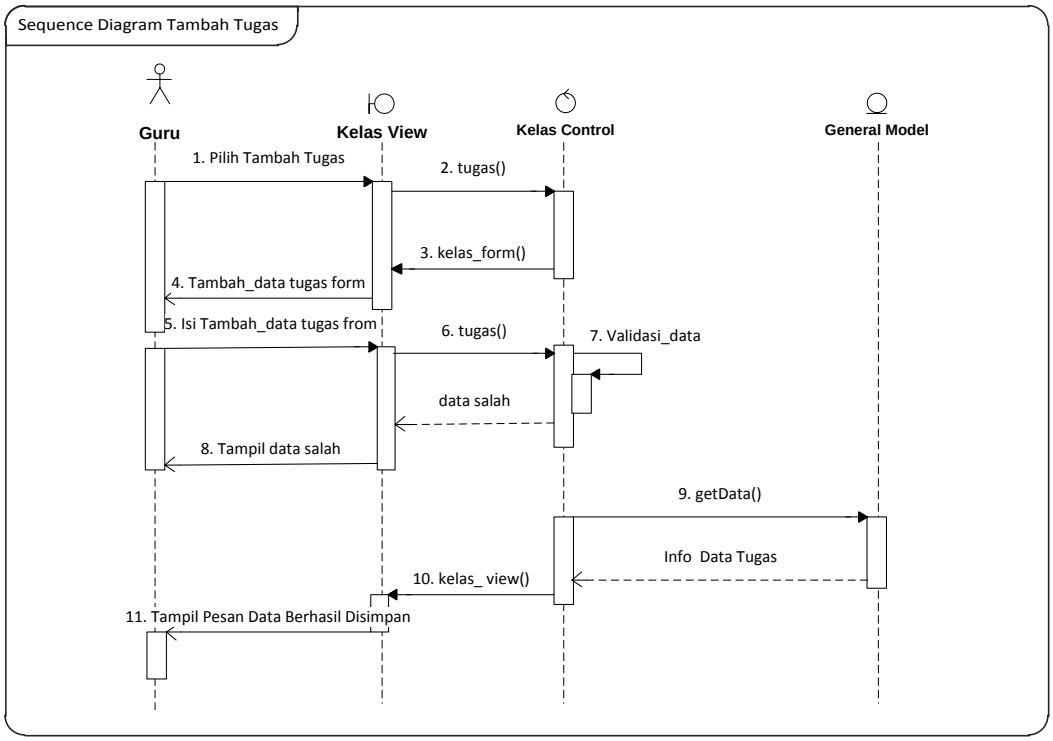
Sequence diagram tambah materi dapat dilihat pada Gambar 3.17.



Gambar 3. 17 Sequence Diagram Tambah Materi

9. Sequence Diagram Tambah Tugas

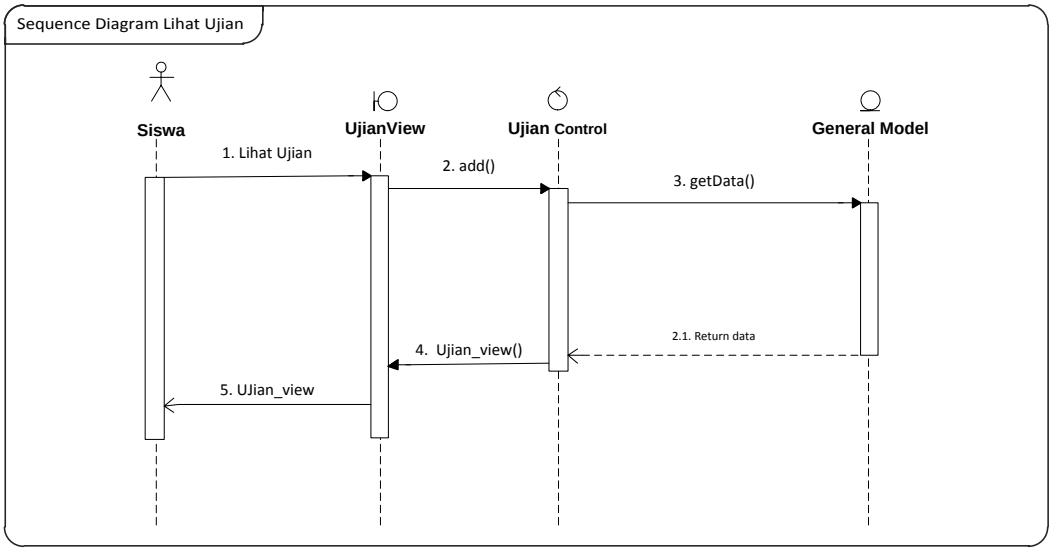
Sequence diagram tambah tugas dapat dilihat pada Gambar 3.18.



Gambar 3. 18 Sequence Diagram Tambah Tugas

10. Sequence Diagram Lihat Ujian

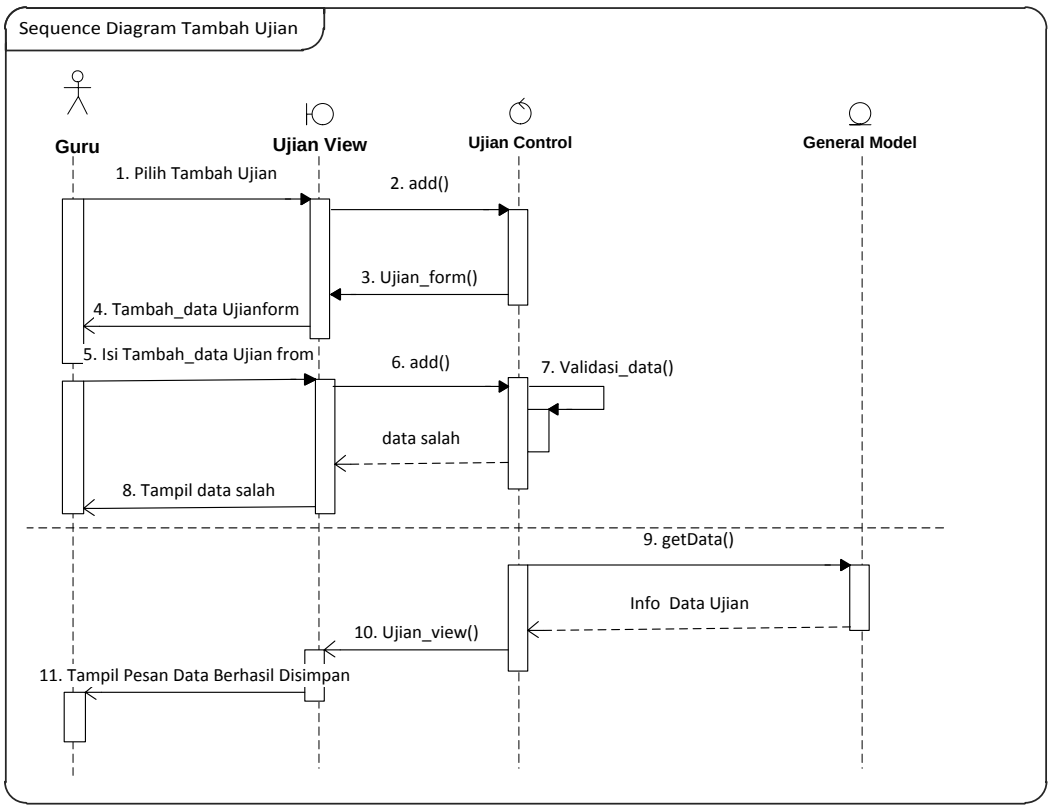
Sequence diagram lihat ujian dapat dilihat pada Gambar 3.19.



Gambar 3. 19 Sequence Diagram Lihat Ujian

11. Sequence Diagram Tambah Ujian

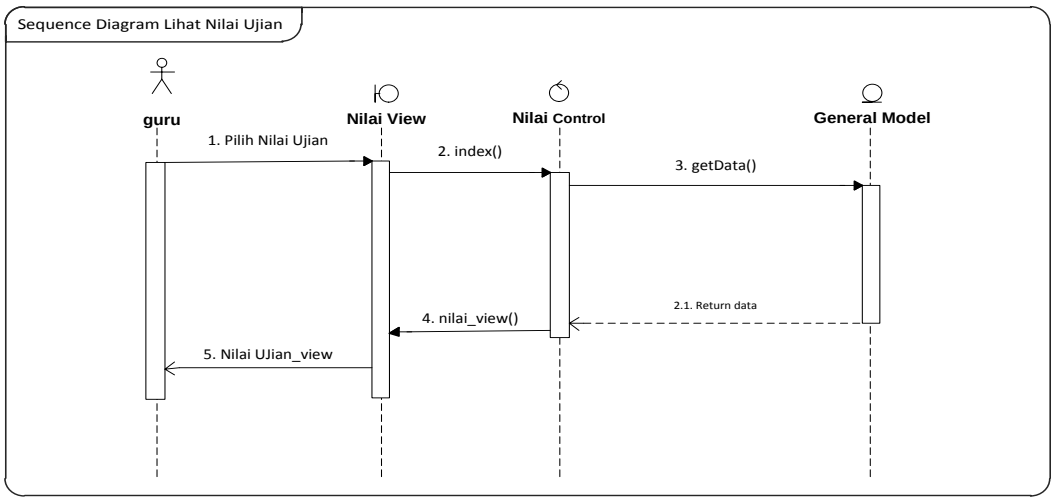
Sequence diagram tambah ujian dapat dilihat pada Gambar 3.20.



Gambar 3. 20 Sequence Diagram Tambah Ujian

12. Sequence Diagram Lihat Nilai Ujian

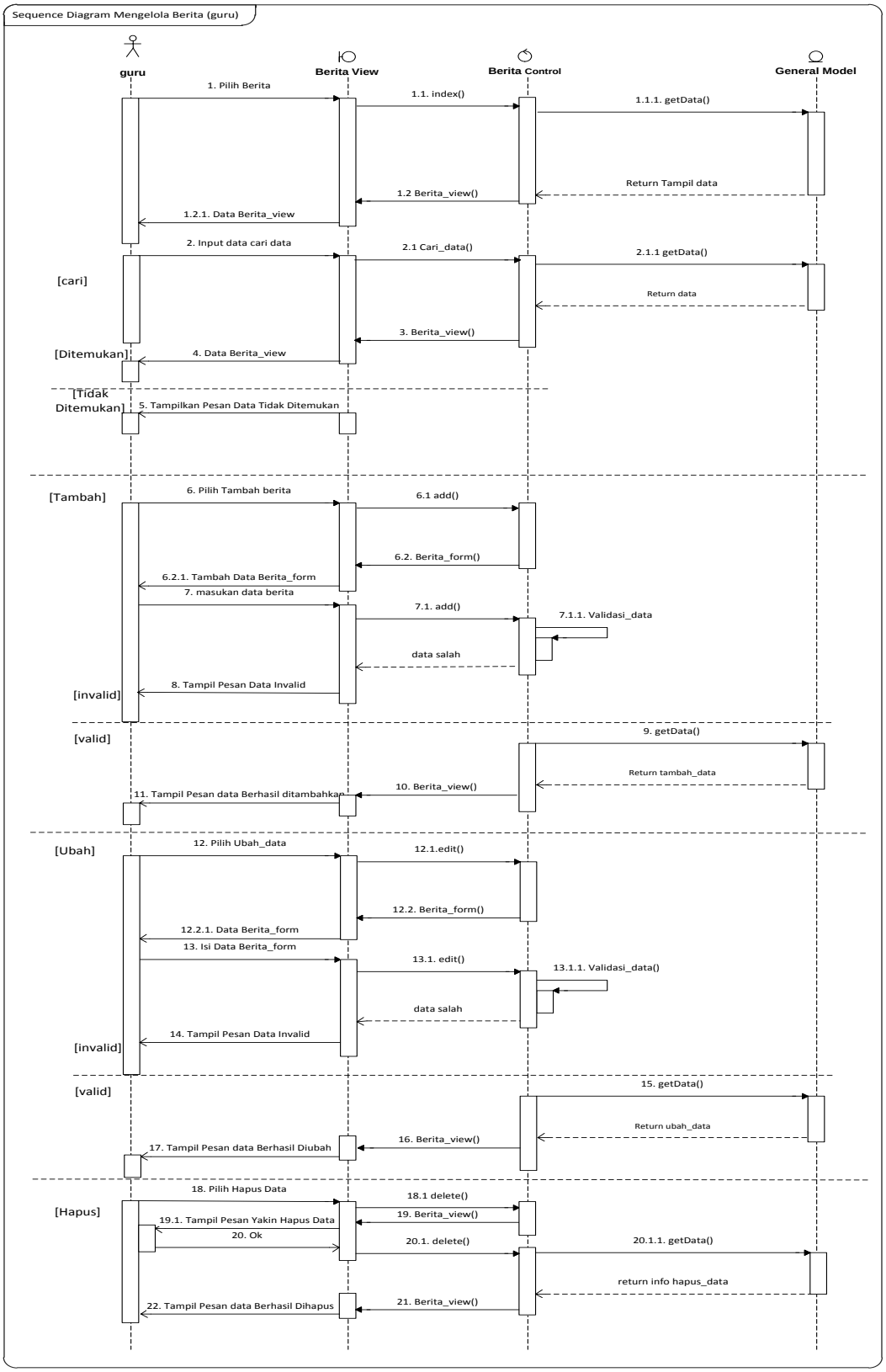
Sequence diagram lihat nilai ujian dapat dilihat pada Gambar 3.21.



Gambar 3. 21 Sequence Diagram Lihat Nilai Ujian

13. Sequence Diagram Mengelola Berita (guru)

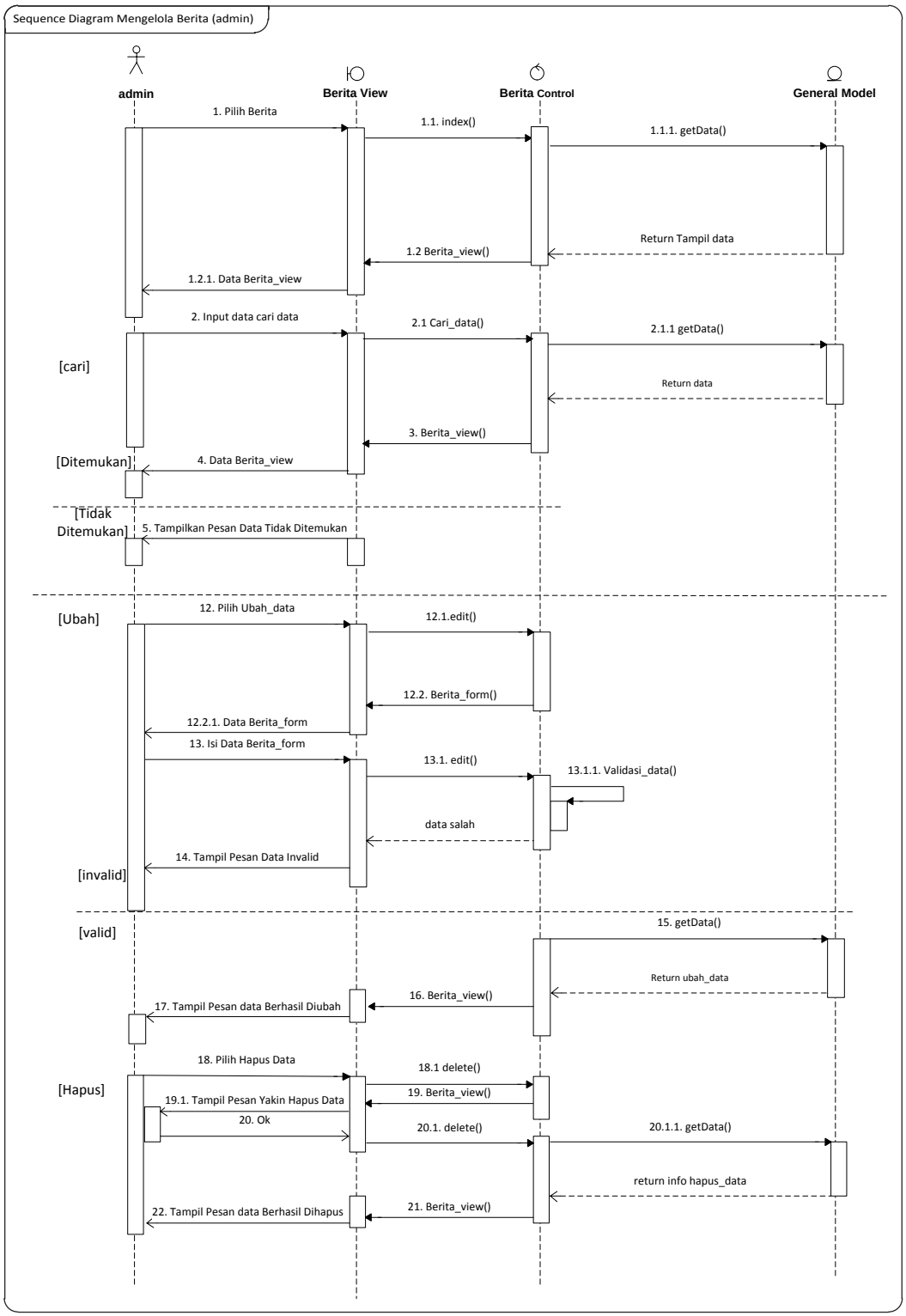
Sequence diagram mengelola berita (guru) dapat dilihat pada Gambar 3.22.



Gambar 3. 22 Sequence Diagram Mengelola Berita (guru)

14. Sequence Diagram Mengelola Berita (admin)

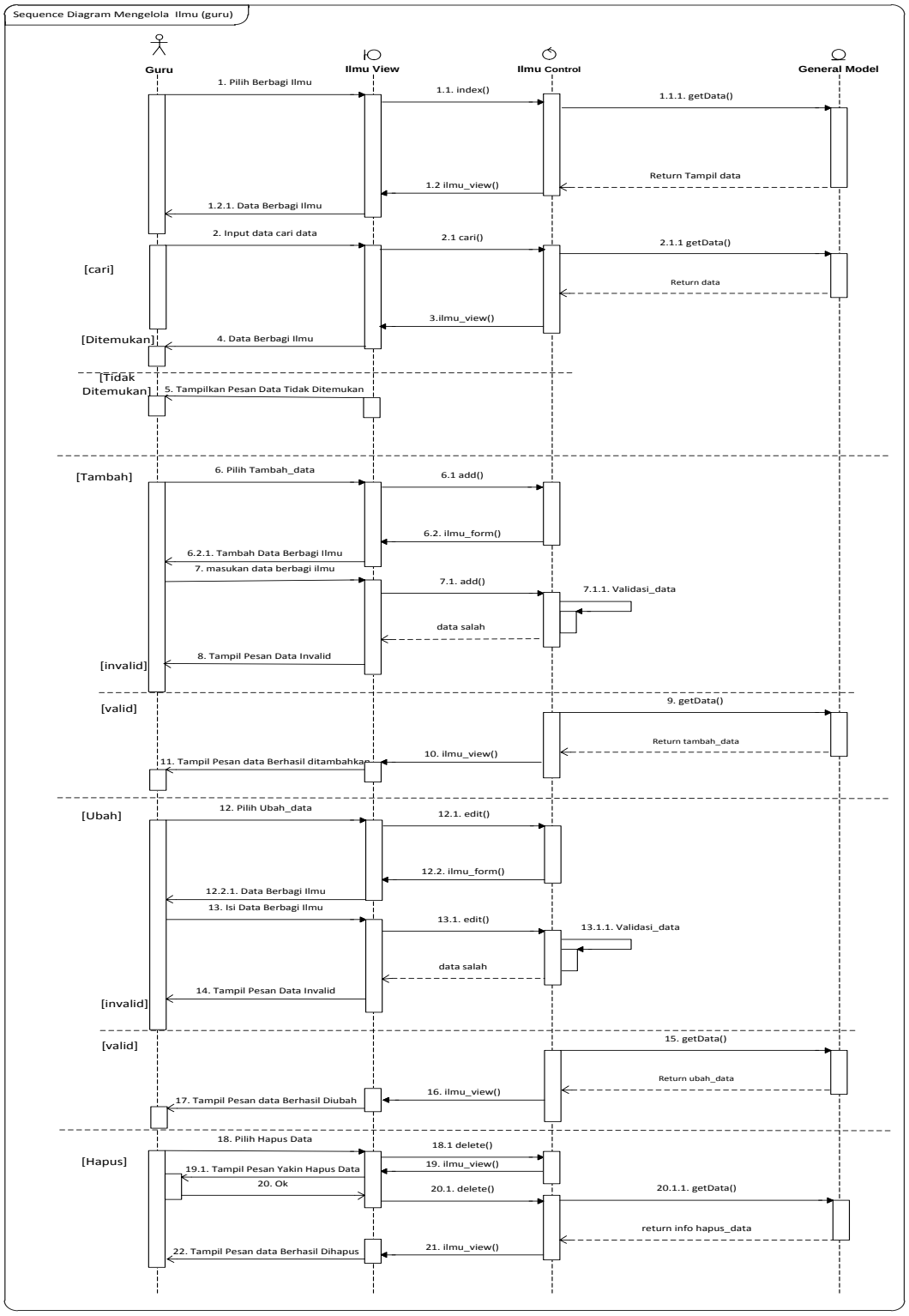
Sequence diagram mengelola berita (admin) dapat dilihat pada Gambar 3.23.



Gambar 3. 23 Sequence Diagram Mengelola Berita (admin)

15. Sequence Diagram Mengelola Ilmu (guru)

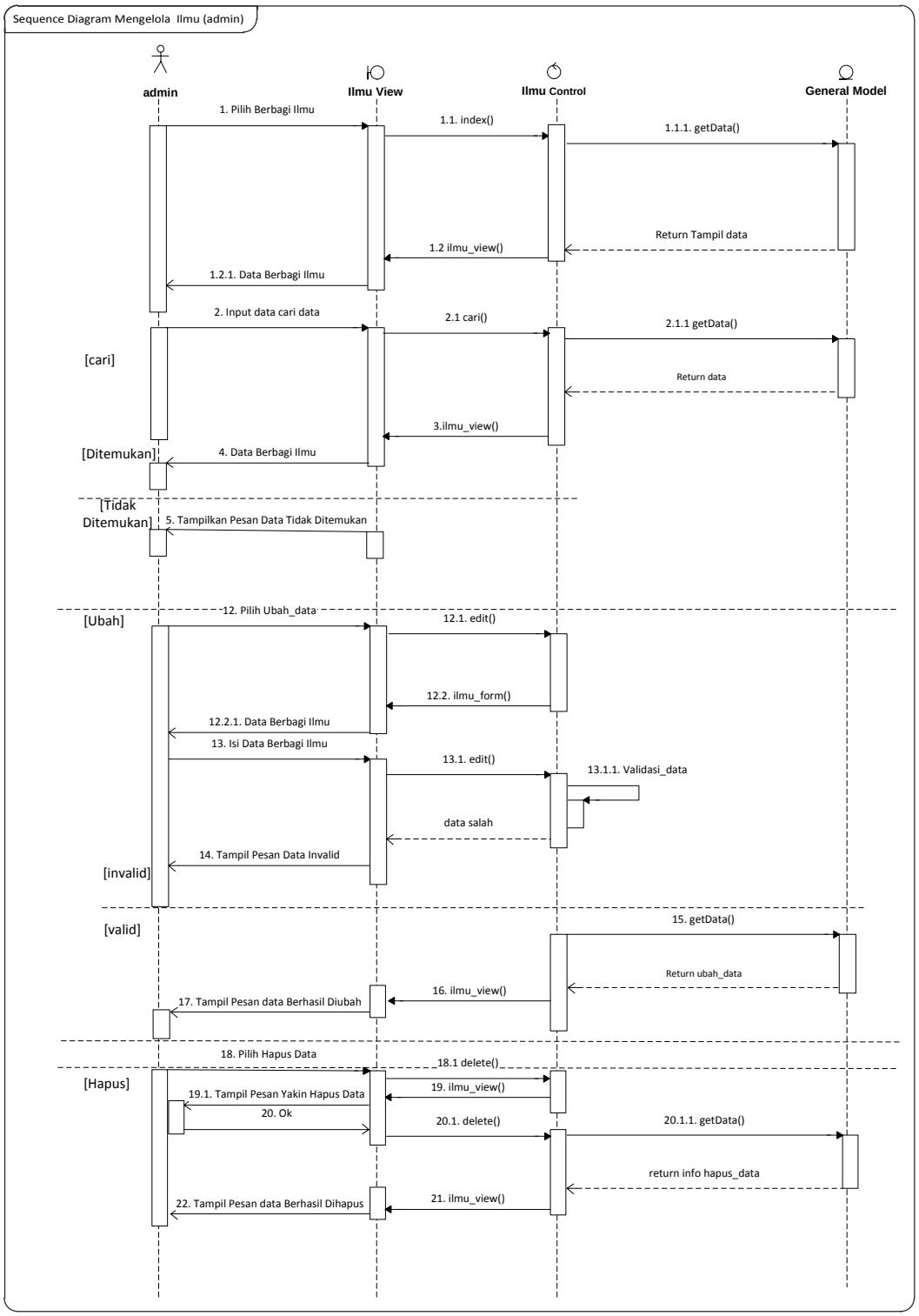
Sequence diagram mengelola ilmu (guru) dapat dilihat pada Gambar 3.24.



Gambar 3. 24 Sequence Diagram Mengelola Ilmu (guru)

16. Sequence Diagram Mengelola Ilmu (admin)

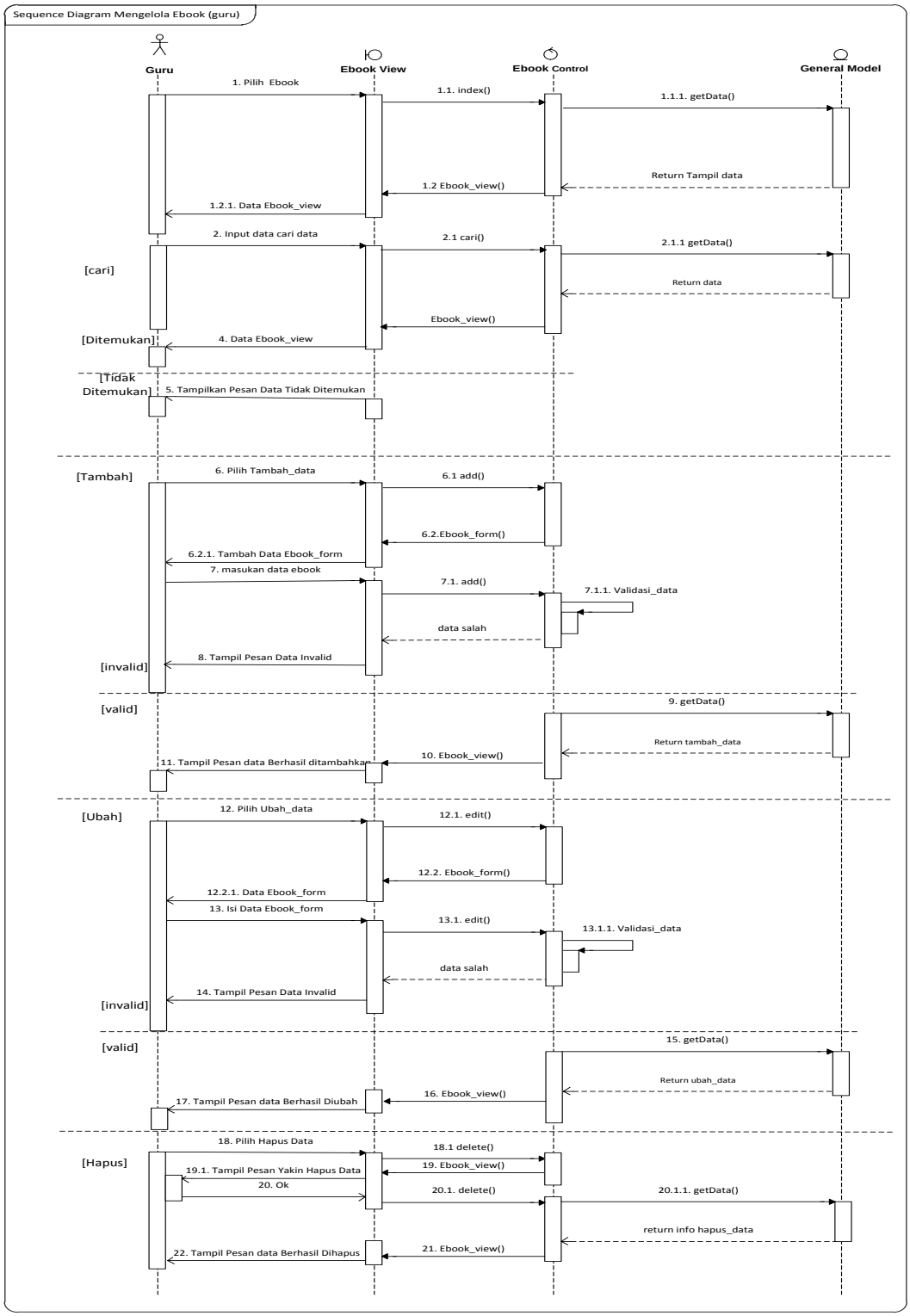
Sequence diagram mengelola ilmu (admin) dapat dilihat pada Gambar 3.25.



Gambar 3. 25 Sequence Diagram Mengelola Ilmu (admin)

17. Sequence Diagram Mengelola Ebook (guru)

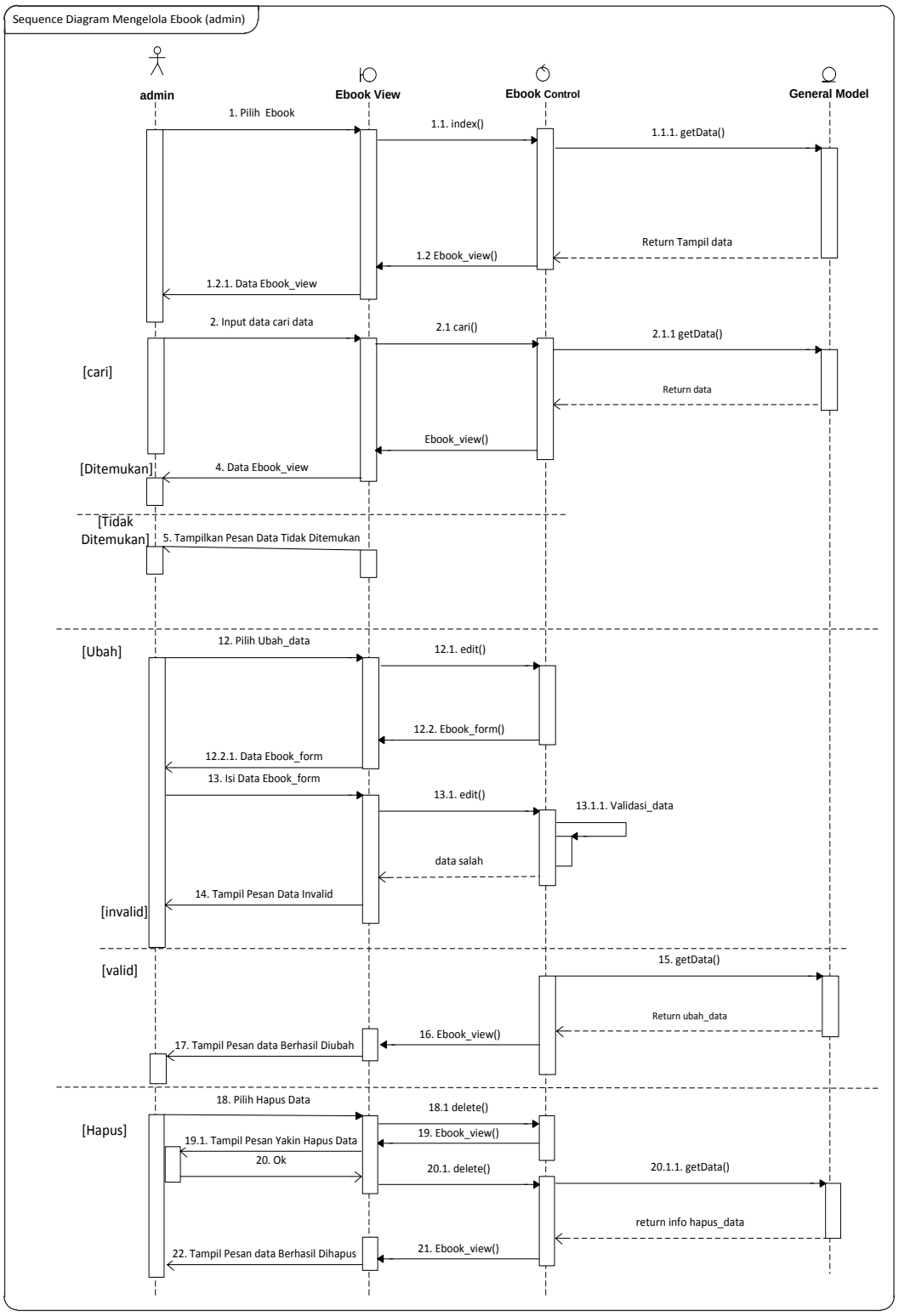
Sequence diagram mengelola ebook (guru) dapat dilihat pada Gambar 3.26.



Gambar 3. 26 Sequence Diagram Mengelola Ebook (guru)

18. Sequence Diagram Mengelola Ebook (admin)

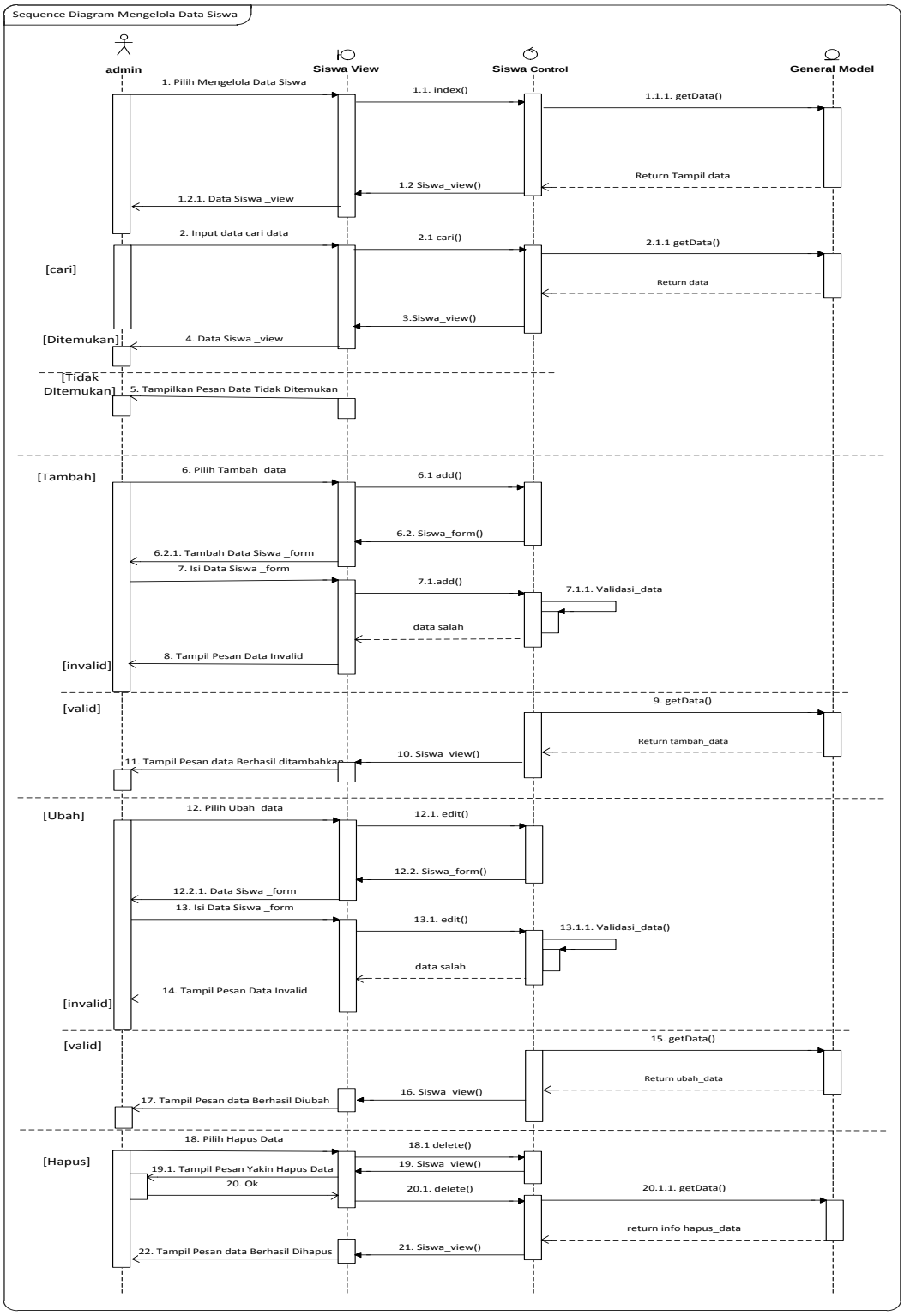
Sequence diagram mengelola ebook (admin) dapat dilihat pada Gambar 3.27.



Gambar 3. 27 Sequence Diagram Mengelola Ebook (admin)

19. Sequence Diagram Mengelola Data Siswa

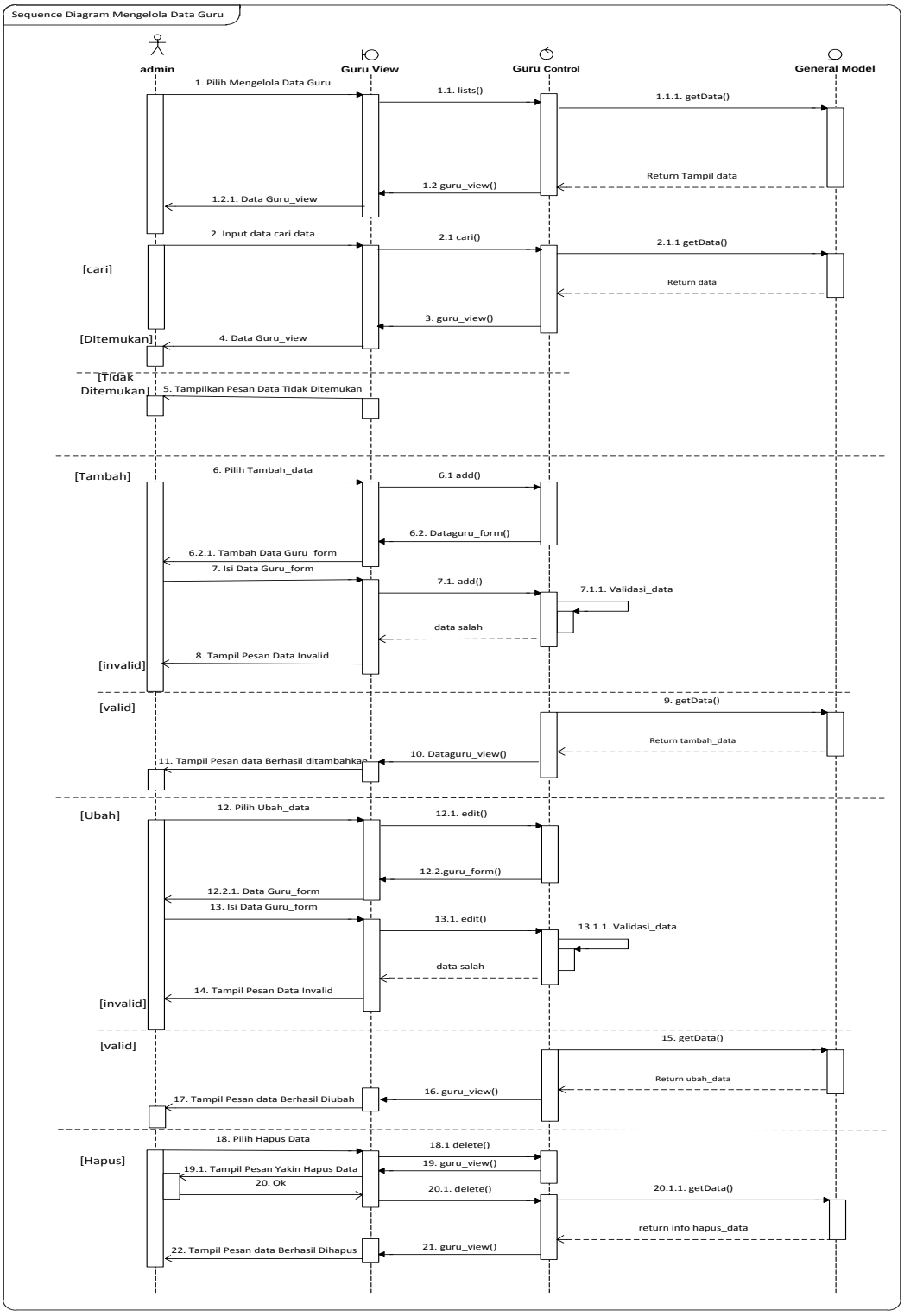
Sequence diagram mengelola data siswa dapat dilihat pada Gambar 3.28.



Gambar 3. 28 Sequence Diagram Menglola Data Siswa

20. Sequence Diagram Mengelola Data Guru

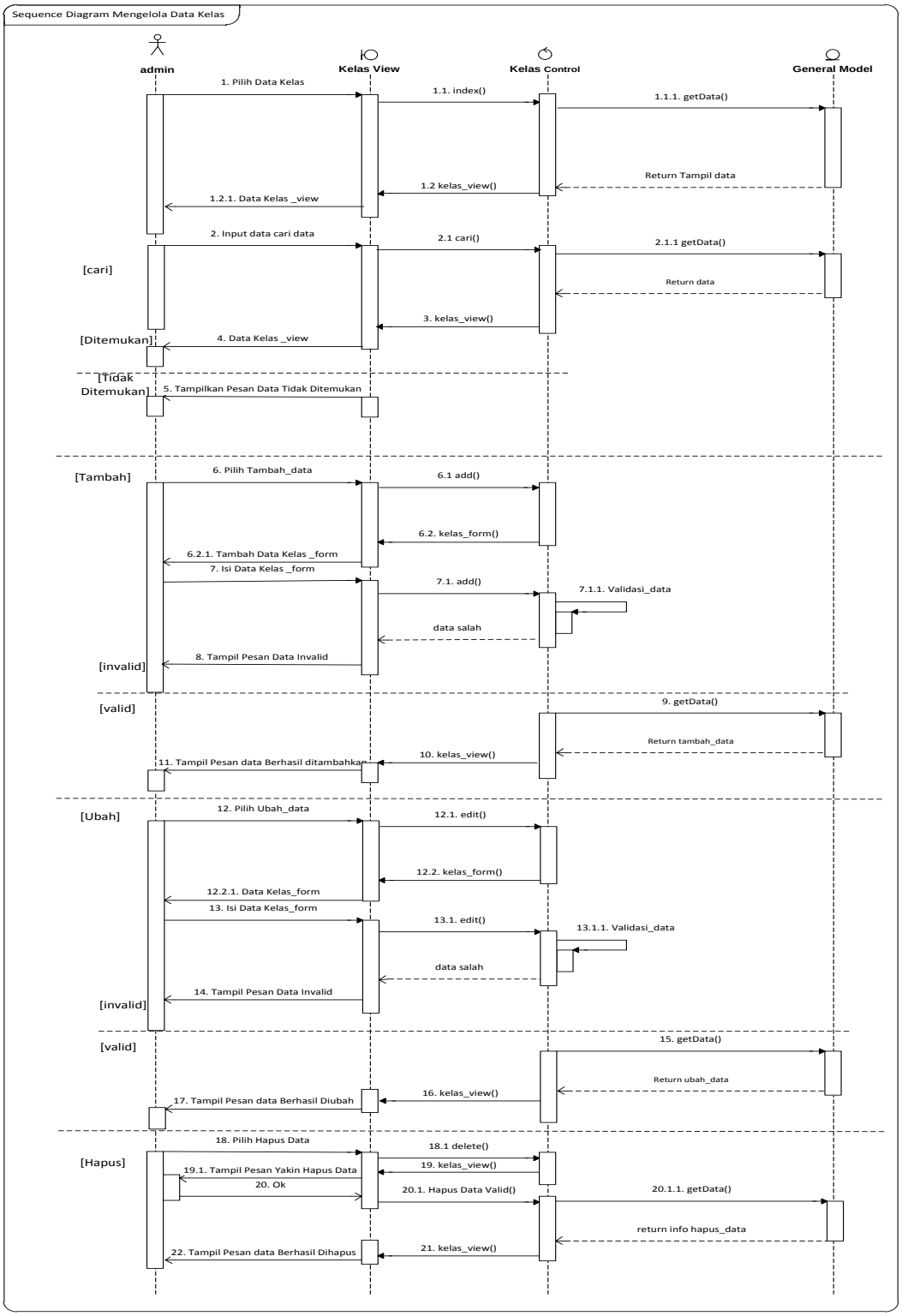
Sequence diagram mengelola data guru dapat dilihat pada Gambar 3.29.



Gambar 3. 29 Sequence Diagram Mengelola Data Guru

21. Sequence Diagram Mengelola Data Kelas

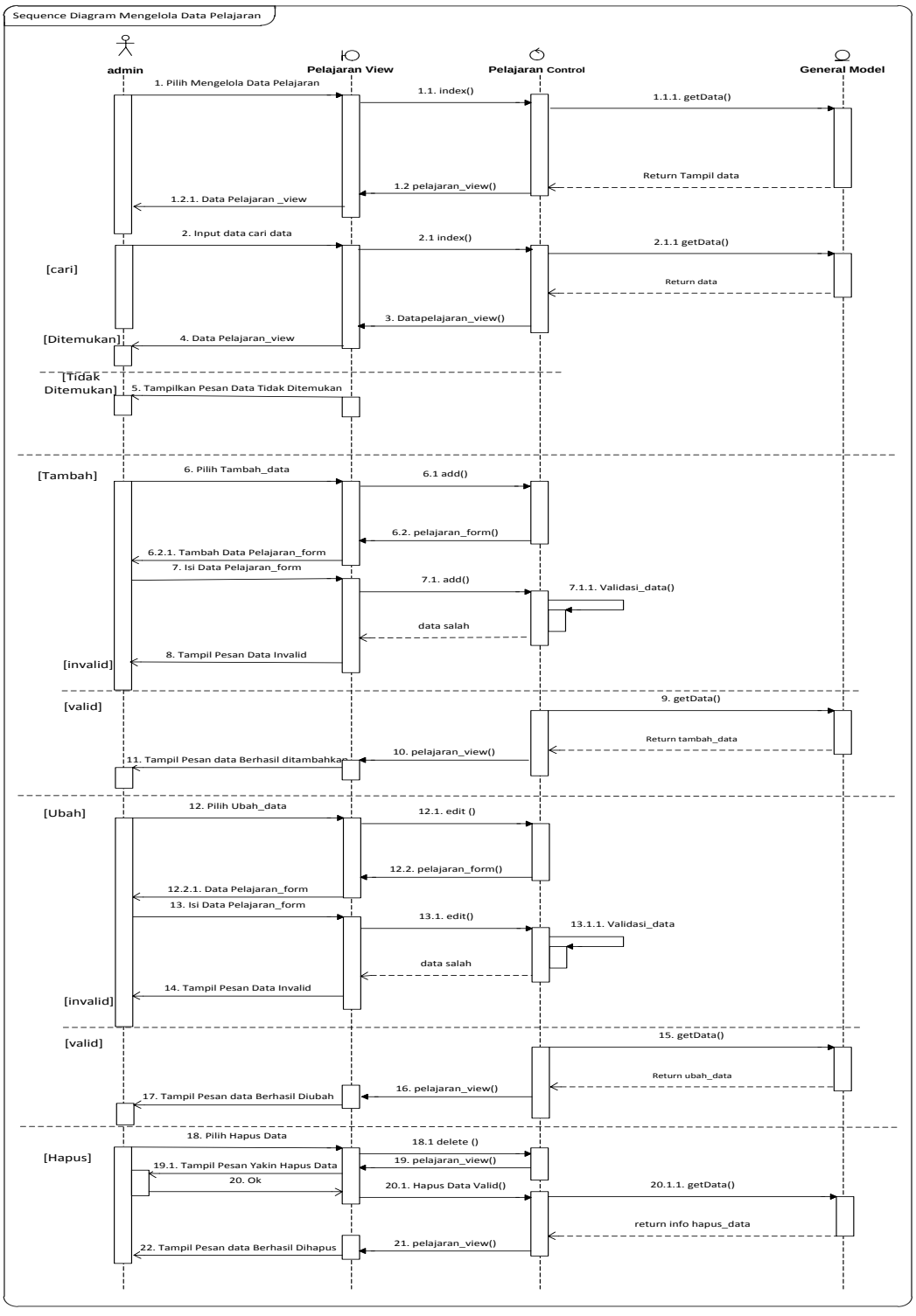
Sequence diagram mengelola data kelas dapat dilihat pada Gambar 3.30.



Gambar 3. 30 Sequence Diagram Mengelola Data Kelas

22. Sequence Diagram Mengelola Data Pelajaran

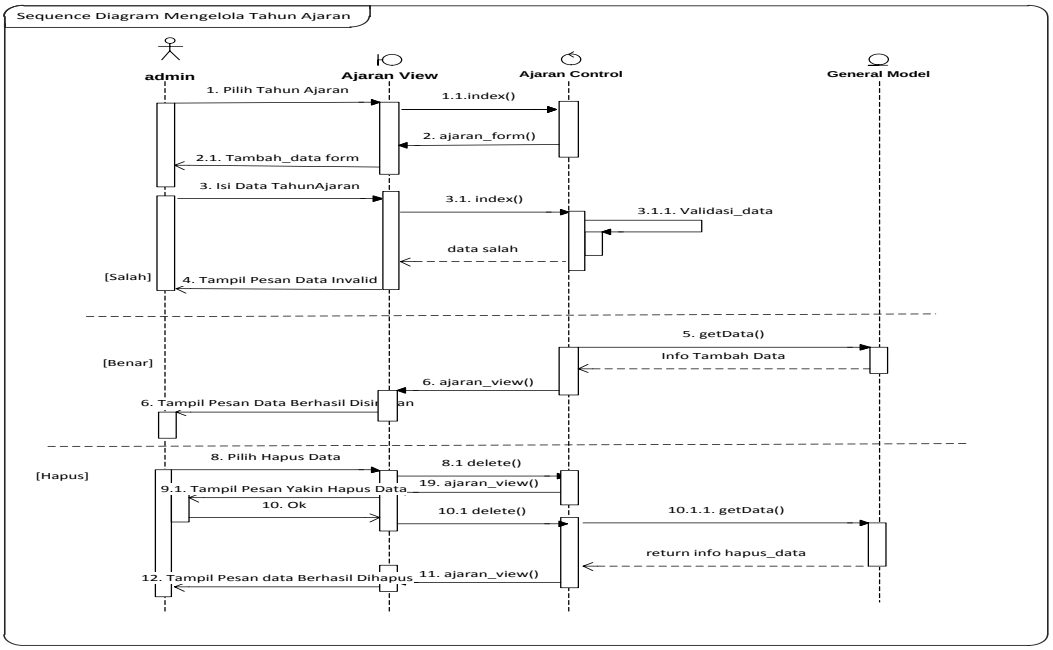
Sequence diagram mengelola data pelajaran dapat dilihat pada Gambar 3.31.



Gambar 3. 31 Sequence Diagram Mengelola Data pelajaran

23. Sequence Diagram Mengelola Tahun Ajaran

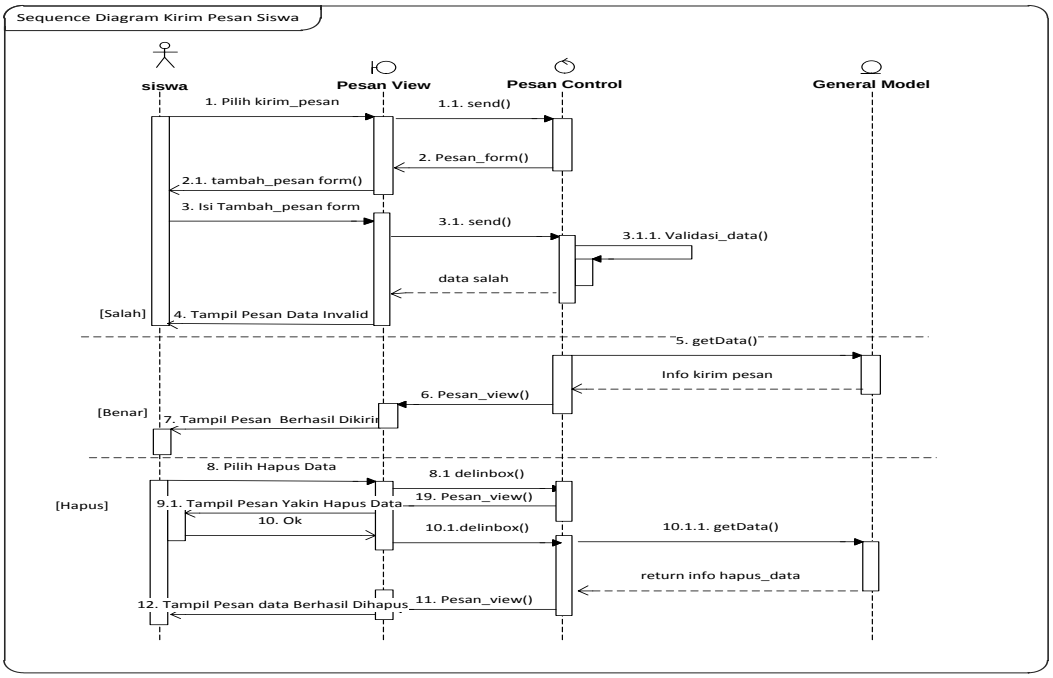
Sequence diagram mengelola tahun ajaran dapat dilihat pada Gambar 3.32.



Gambar 3. 32 Sequence Diagram Mengelola Tahun Ajaran

24. Sequence Diagram Kirim Pesan Siswa

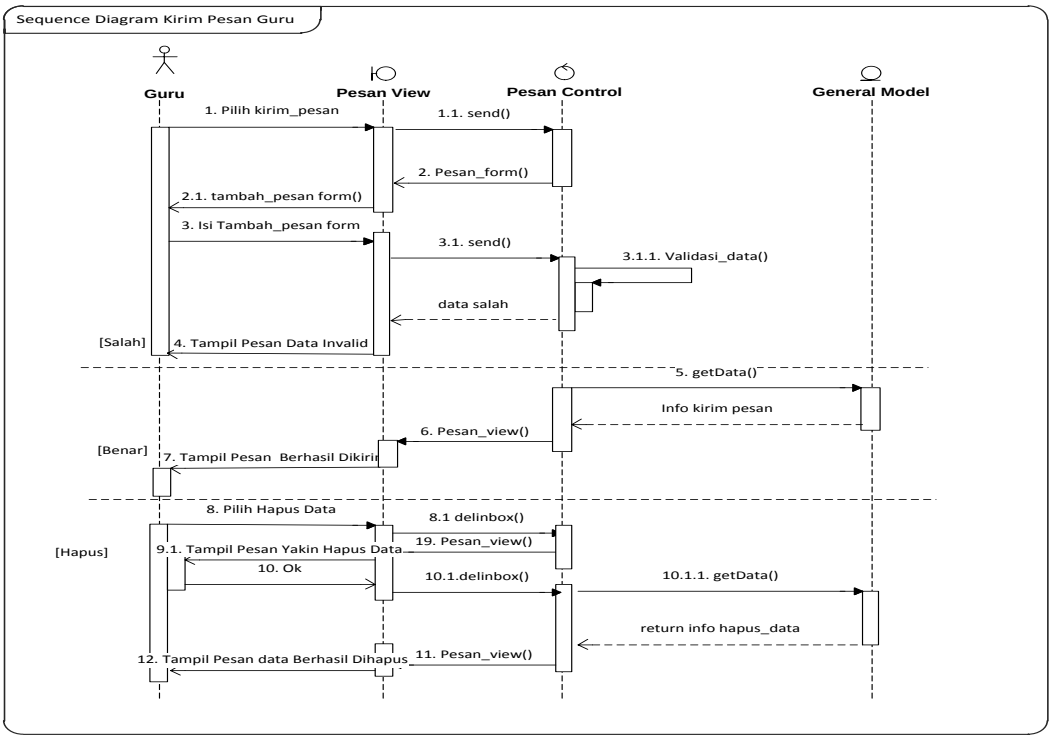
Sequence diagram kirim pesan siswa dapat dilihat pada Gambar 3.33.



Gambar 3. 33 Sequence Diagram Kirim Pesan Siswa

25. Sequence Diagram Kirim Pesan Guru

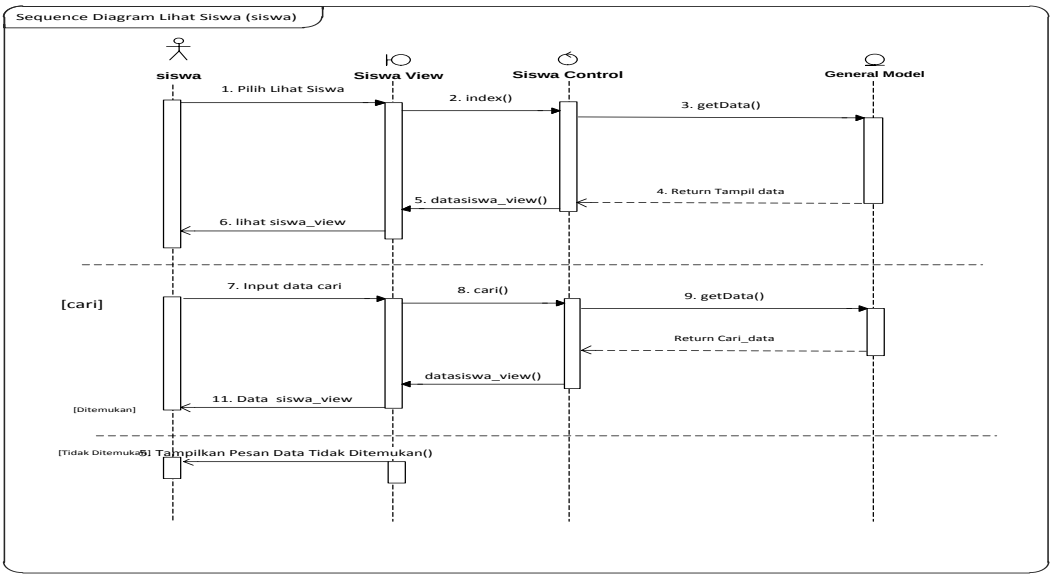
Sequence diagram kirim pesan guru dapat dilihat pada Gambar 3.34.



Gambar 3. 34 Sequence Diagram Kirim Pesan Guru

26. Sequence Diagram Lihat Siswa (siswa)

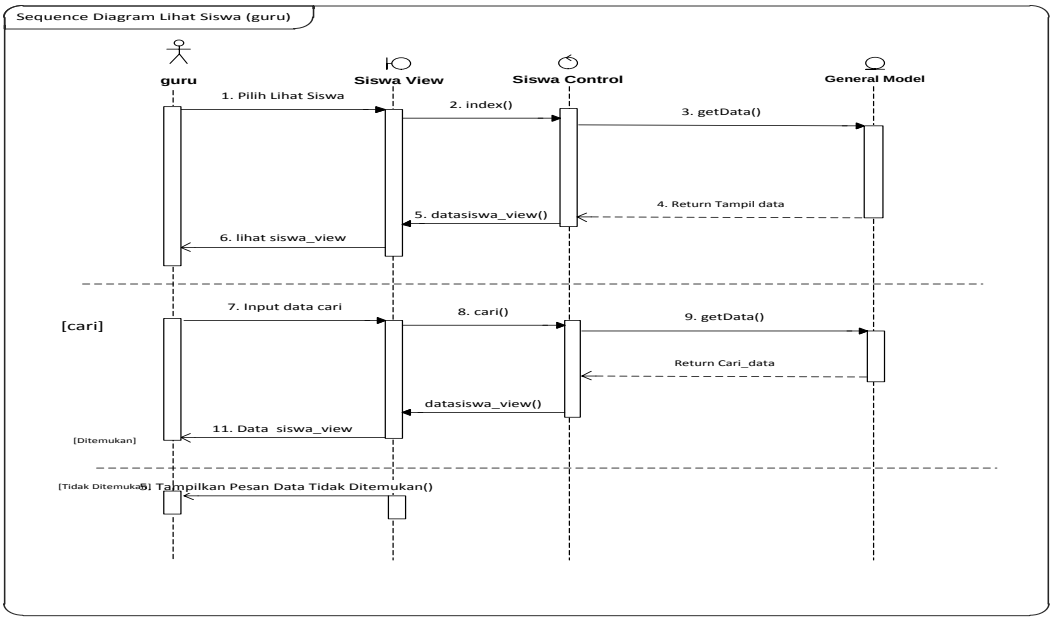
Sequence diagram lihat siswa (siswa) dapat dilihat pada Gambar 3.35.



Gambar 3. 35 Sequence Diagram Lihat Siswa (siswa)

27. Sequence Diagram Lihat Siswa (guru)

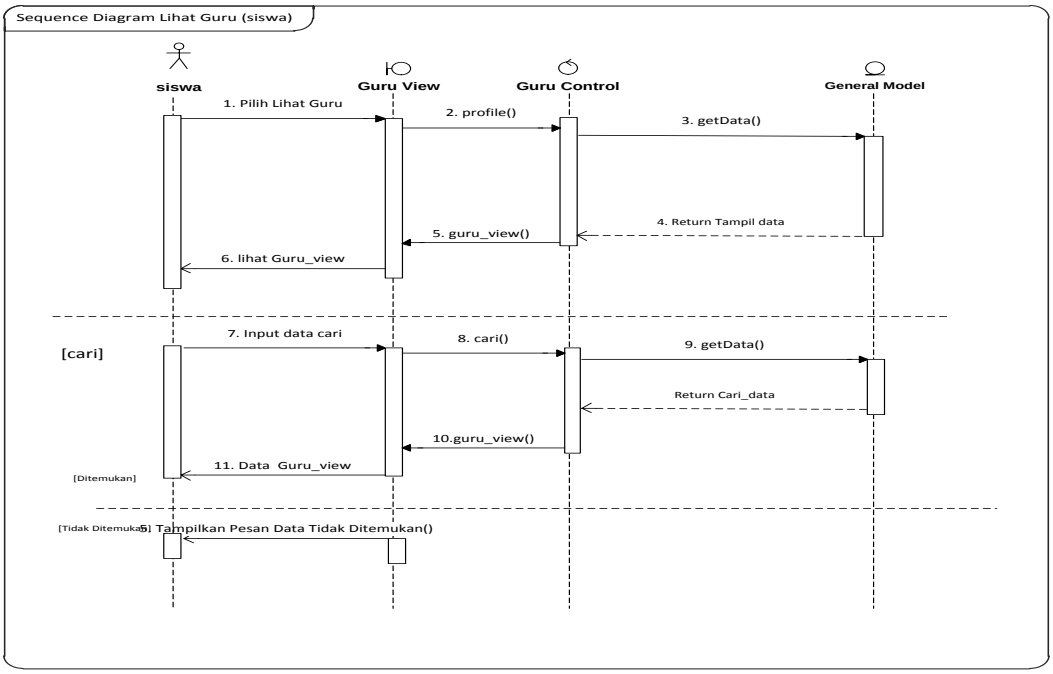
Sequence diagram lihat siswa (guru) dapat dilihat pada Gambar 3.36.



Gambar 3. 36 Sequence Diagram Lihat Siswa (guru)

28. Sequence Diagram Lihat Guru (siswa)

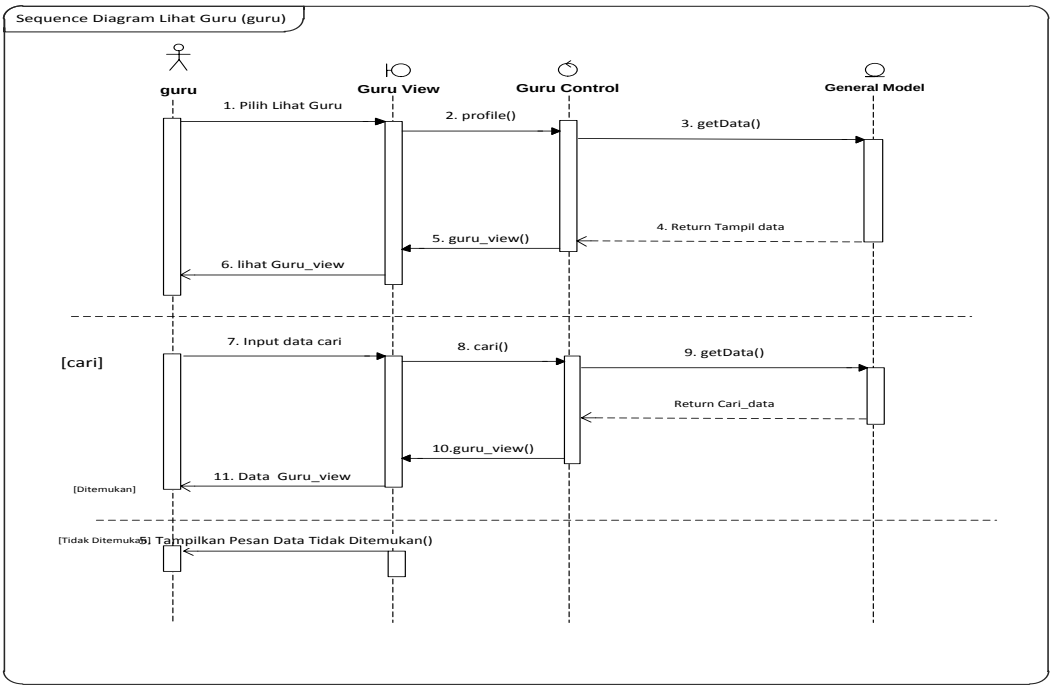
Sequence diagram lihat guru dapat dilihat pada Gambar 3.37.



Gambar 3. 37 Sequence Diagram Lihat Guru (siswa)

29. Sequence Diagram Lihat Guru (guru)

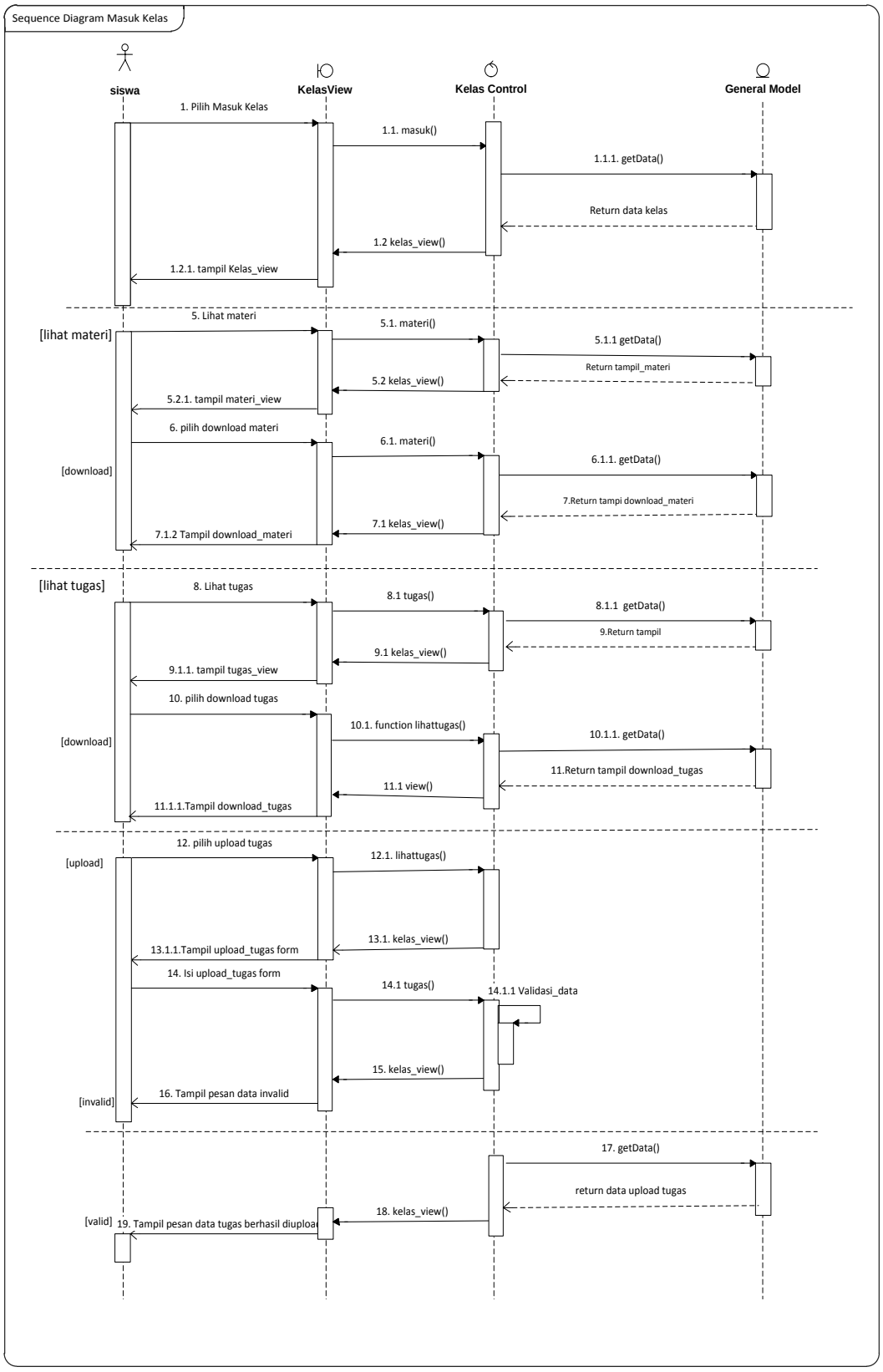
Sequence diagram lihat guru dapat dilihat pada Gambar 3.38.



Gambar 3. 38 Sequence Diagram Lihat Guru (guru)

30. Sequence Diagram Masuk Kelas

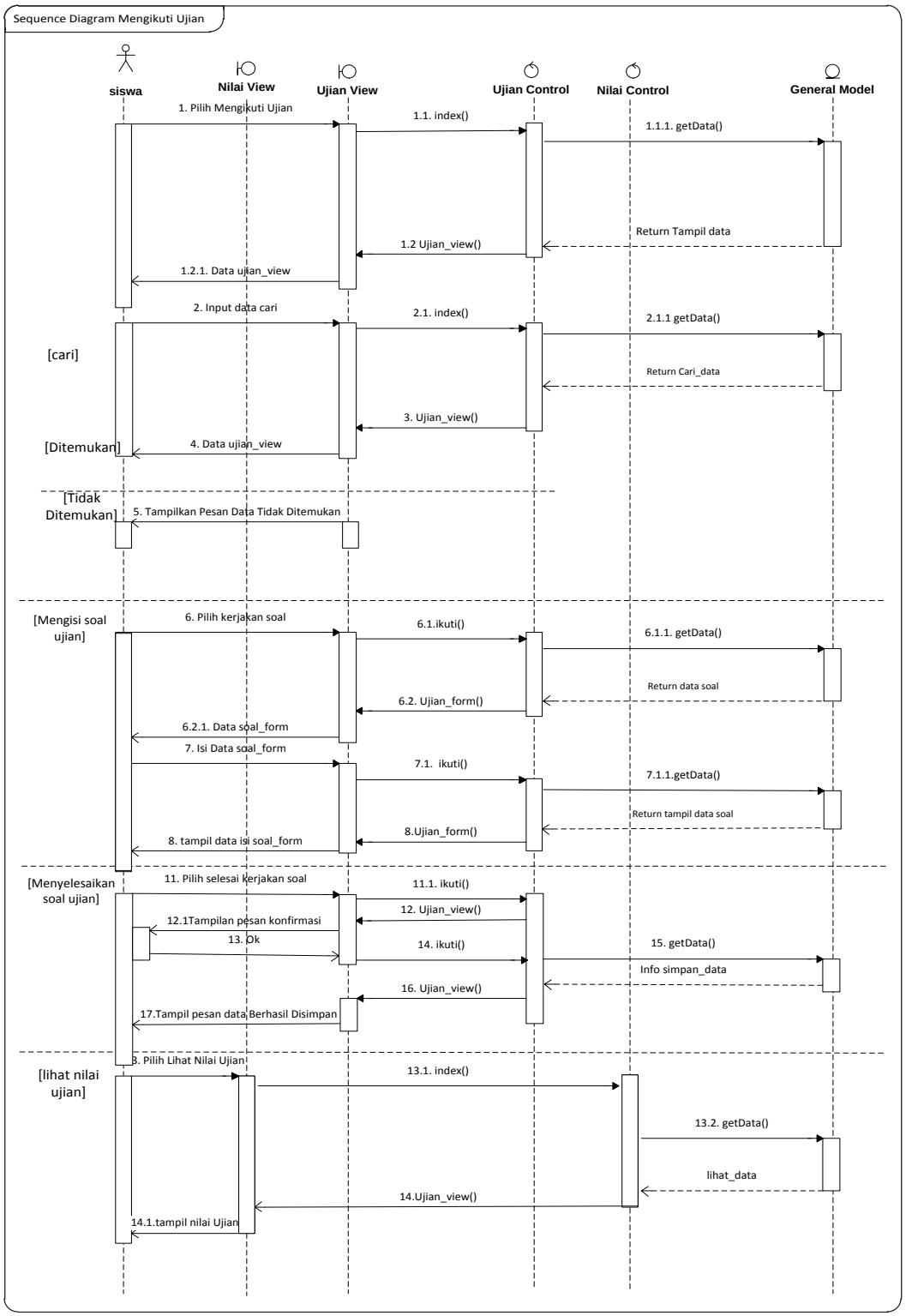
Sequence diagram masuk kelas dapat dilihat pada Gambar 3.39.



Gambar 3. 39 Sequence Diagram Masuk Kelas

31. Sequence Diagram Mengikuti Ujian

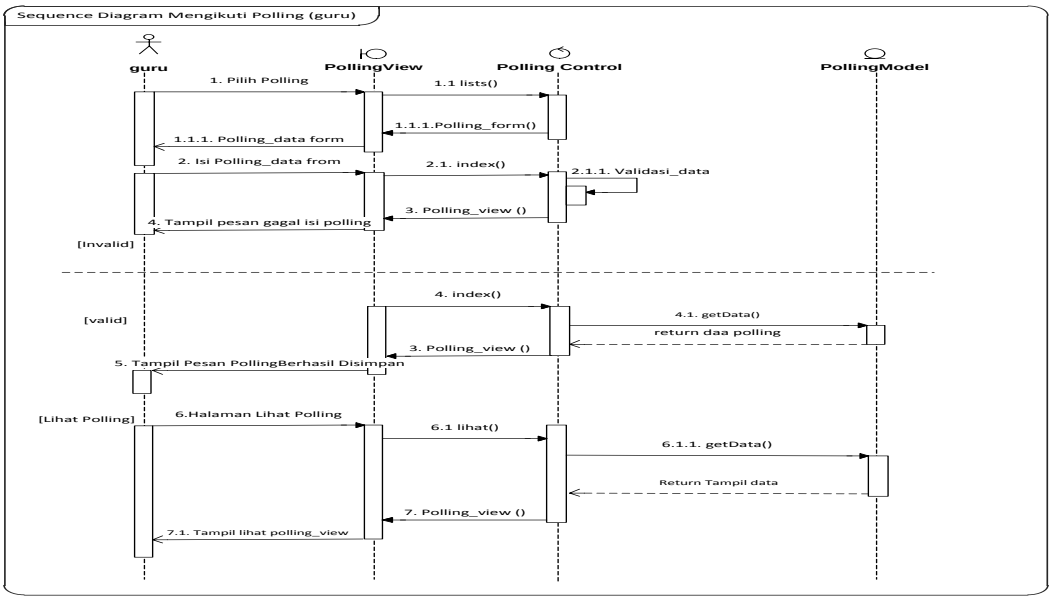
Sequence diagram mengikuti ujian dapat dilihat pada Gambar 3.40.



Gambar 3. 40 Sequence Diagram Mengikuti Ujian

32. Sequence Diagram Mengikuti *Polling* (guru)

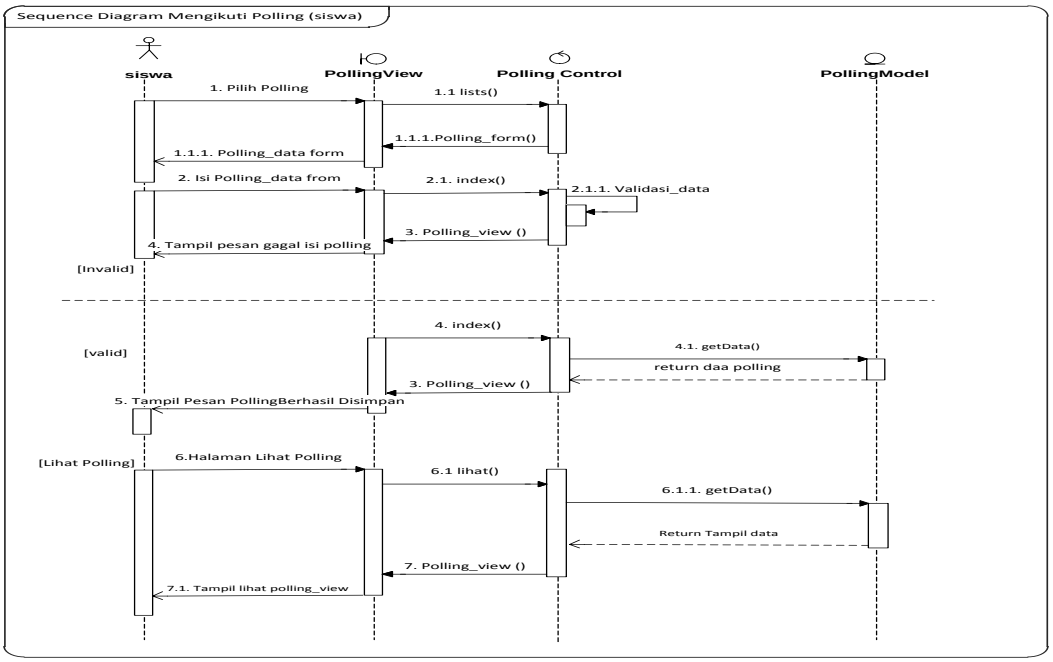
Sequence diagram mengikuti *polling* (guru) dapat dilihat pada Gambar 3.41.



Gambar 3. 41 Sequence Diagram Mengikuti Polling (guru)

33. Sequence Diagram Mengikuti *Polling* (siswa)

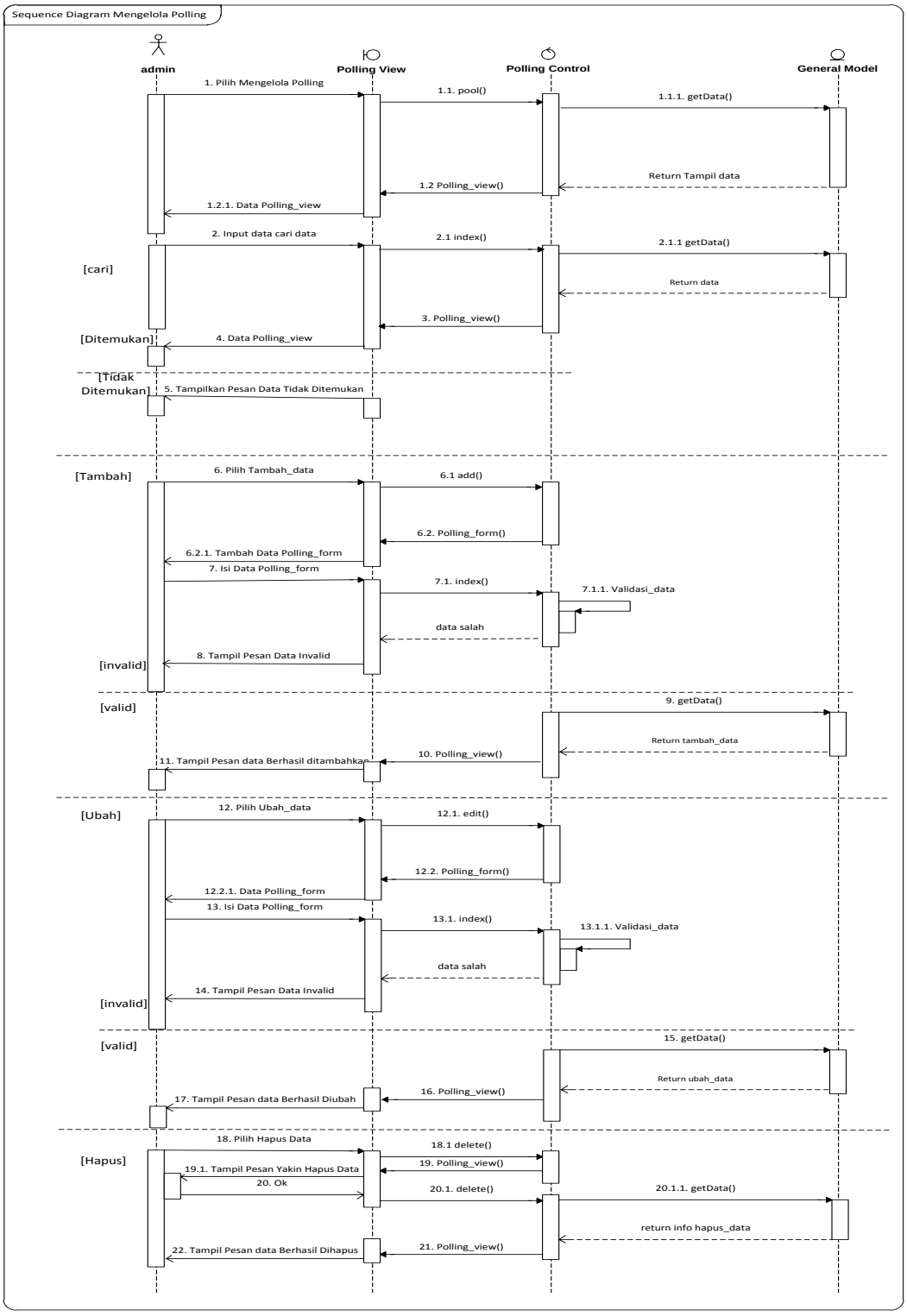
Sequence diagram mengikuti *polling* (siswa) dapat dilihat pada Gambar 3.42.



Gambar 3. 42 Sequence Diagram Mengikuti Polling (siswa)

34. Sequence Diagram Mengelola *Polling*

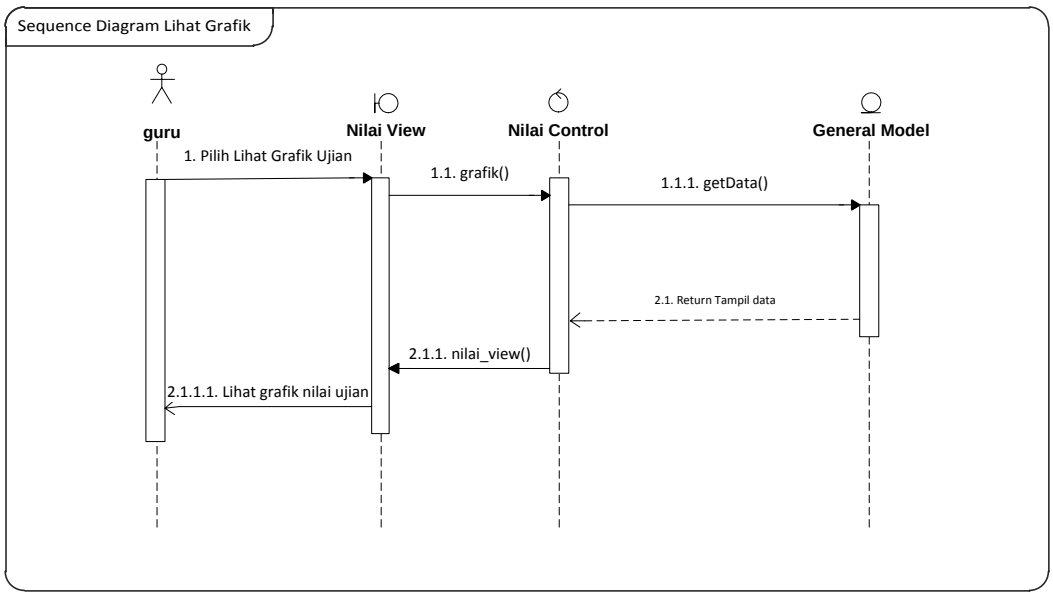
Sequence diagram mengelola *polling* dapat dilihat pada Gambar 3.43.



Gambar 3. 43 Sequence Diagram Mengelola Polling

35. Sequence Diagram Lihat Grafik Nilai

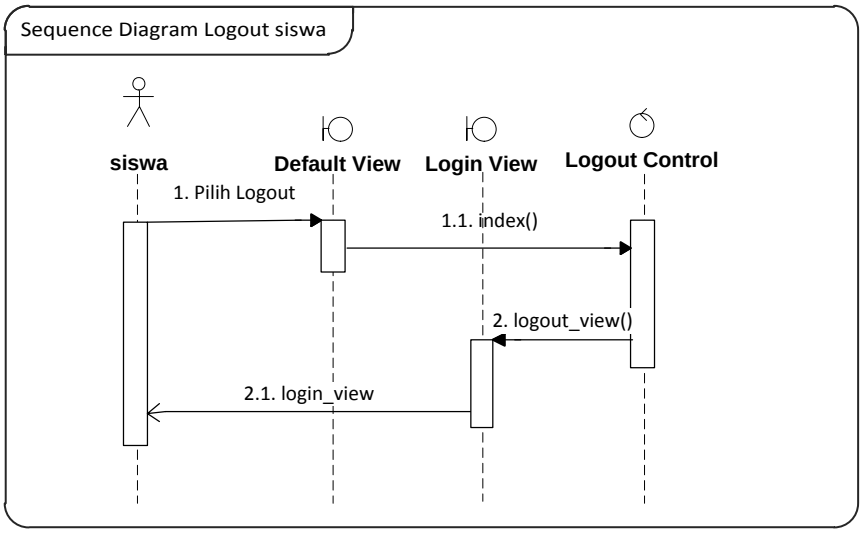
Sequence diagram lihat grafik nilai dapat dilihat pada Gambar 3.44.



Gambar 3. 44 Sequence Diagram Lihat Grafik Nilai

36. Sequence Diagram *Logout* Siswa

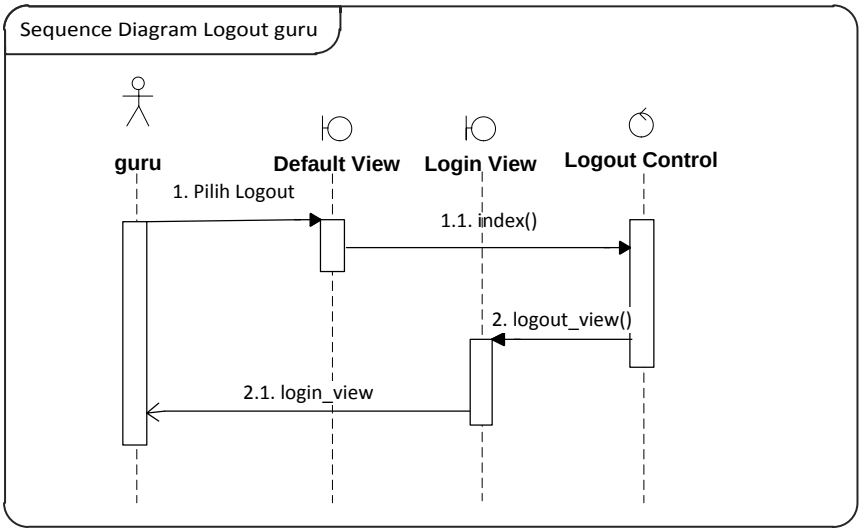
Sequence diagram *logout* siswa dapat dilihat pada Gambar 3.45.



Gambar 3. 45 Sequence Diagram Logout Siswa

37. Sequence Diagram *Logout* Guru

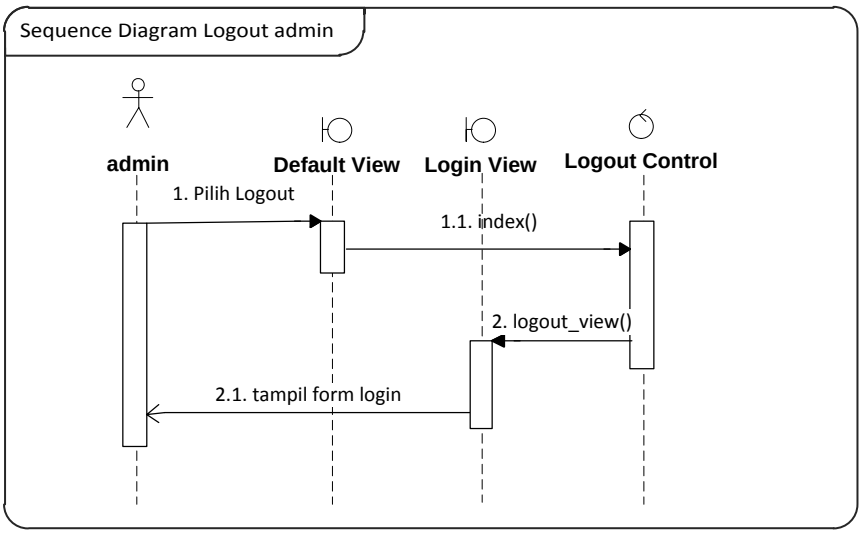
Sequence diagram *logout* guru dapat dilihat pada Gambar 3.46.



Gambar 3. 46 Sequence Diagram Logout Guru

38. Sequence Diagram *Logout* Admin

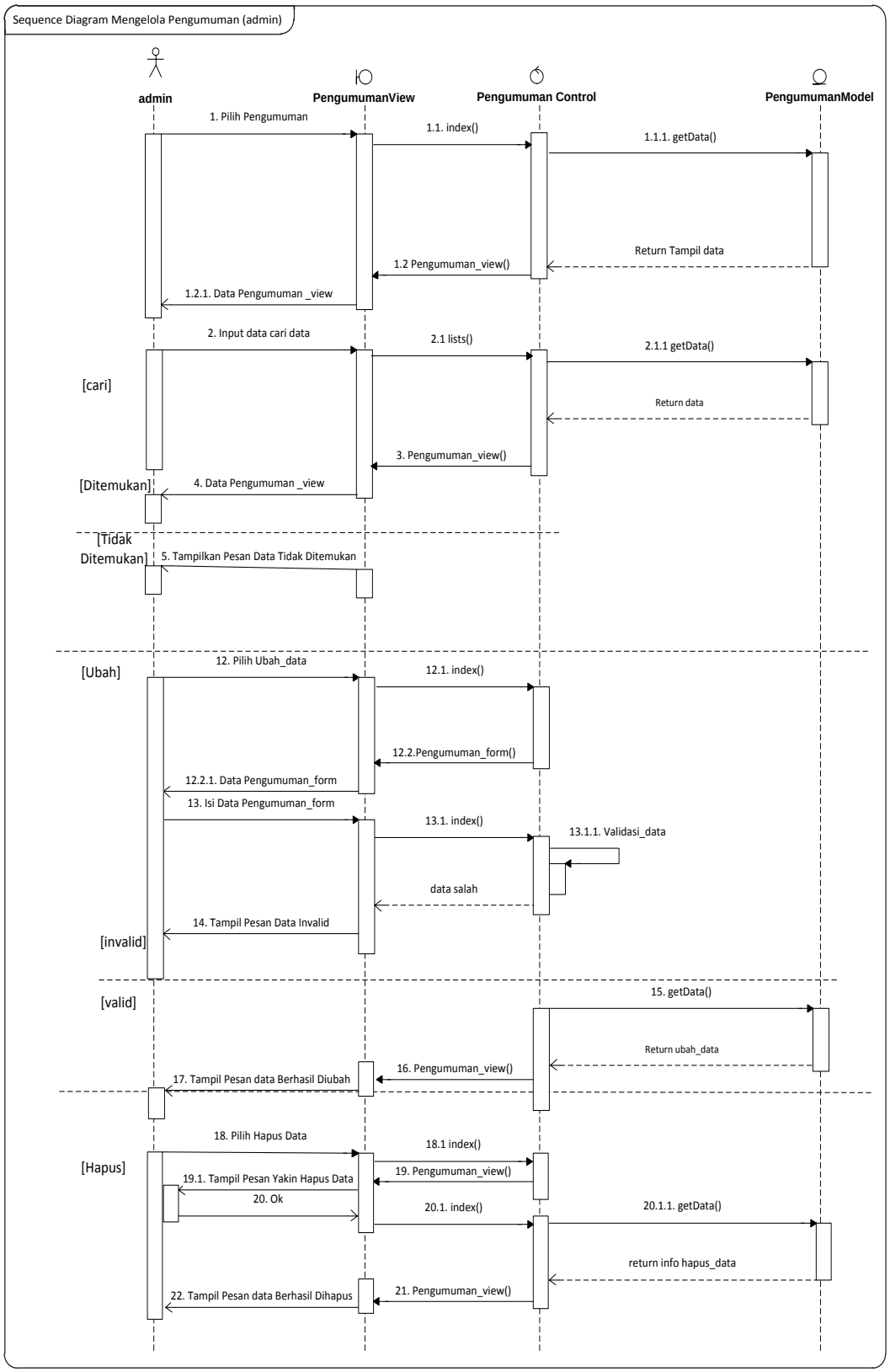
Sequence diagram *logout* admin dapat dilihat pada Gambar 3.47.



Gambar 3. 47 Sequence Diagram Logout Admin

39. Sequence Diagram Mengelola Pengumuman (admin)

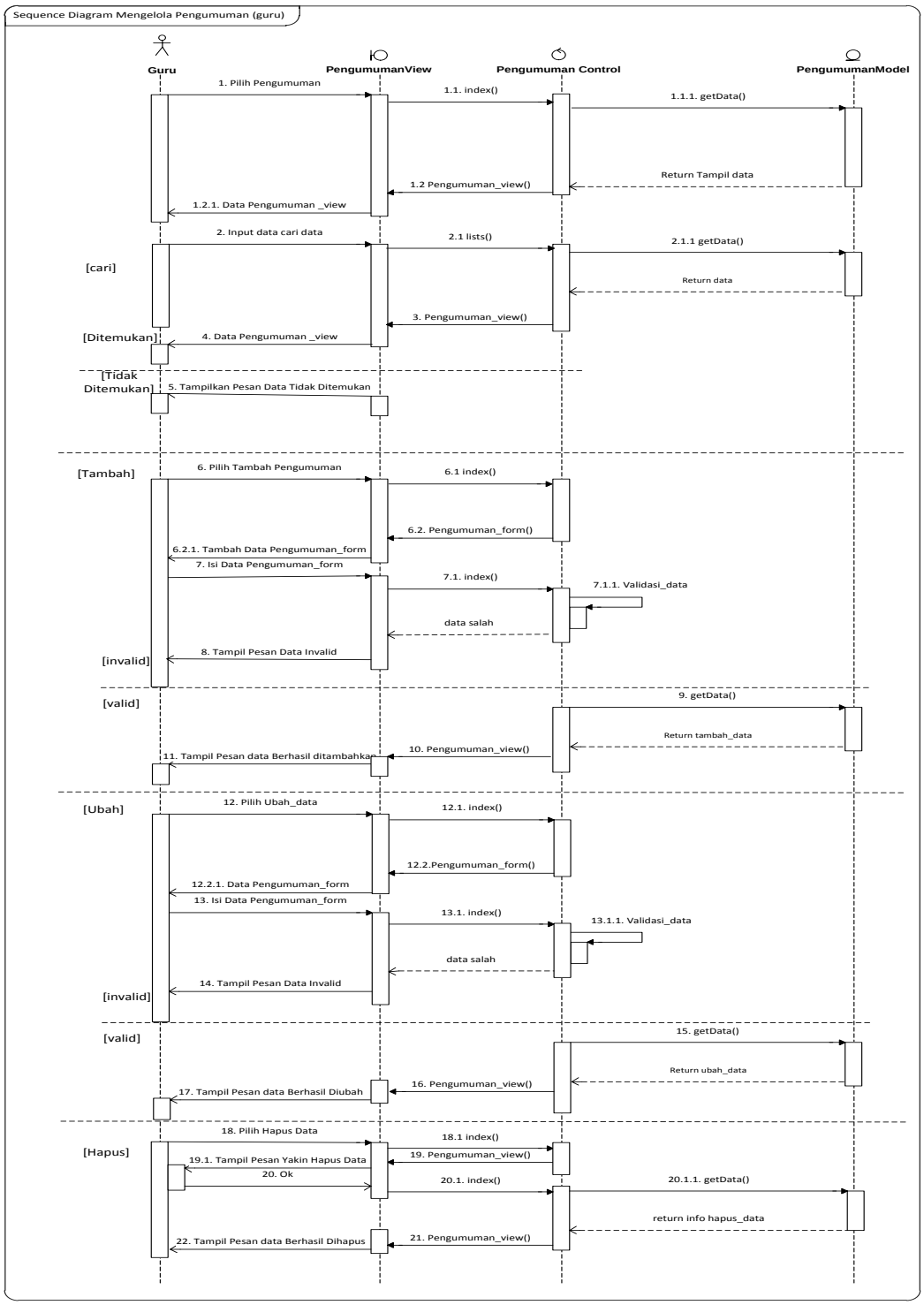
Sequence diagram mengelola pengumuman (admin) dapat dilihat pada Gambar 3.48.



Gambar 3. 48 Sequence Diagram Mengelola Pengumuman (admin)

40. Sequence Diagram Mengelola Pengumuman (guru)

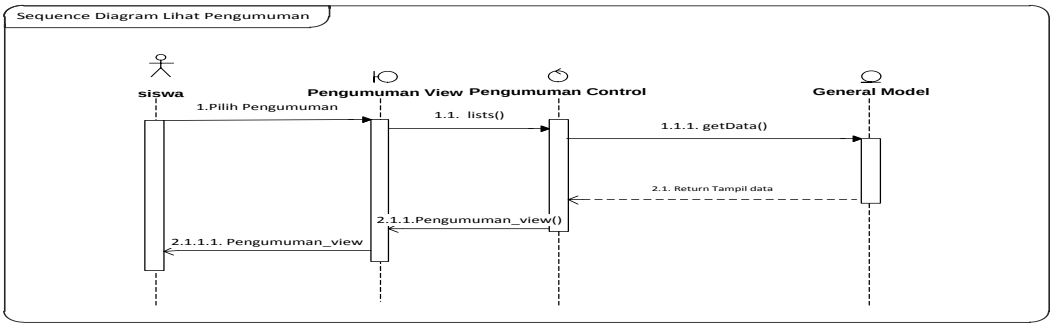
Sequence diagram mengelola pengumuman (guru) dapat dilihat pada Gambar 3.49.



Gambar 3. 49 Sequence Diagram Mengelola Pengumuman (guru)

41. Sequence Diagram Lihat Pengumuman

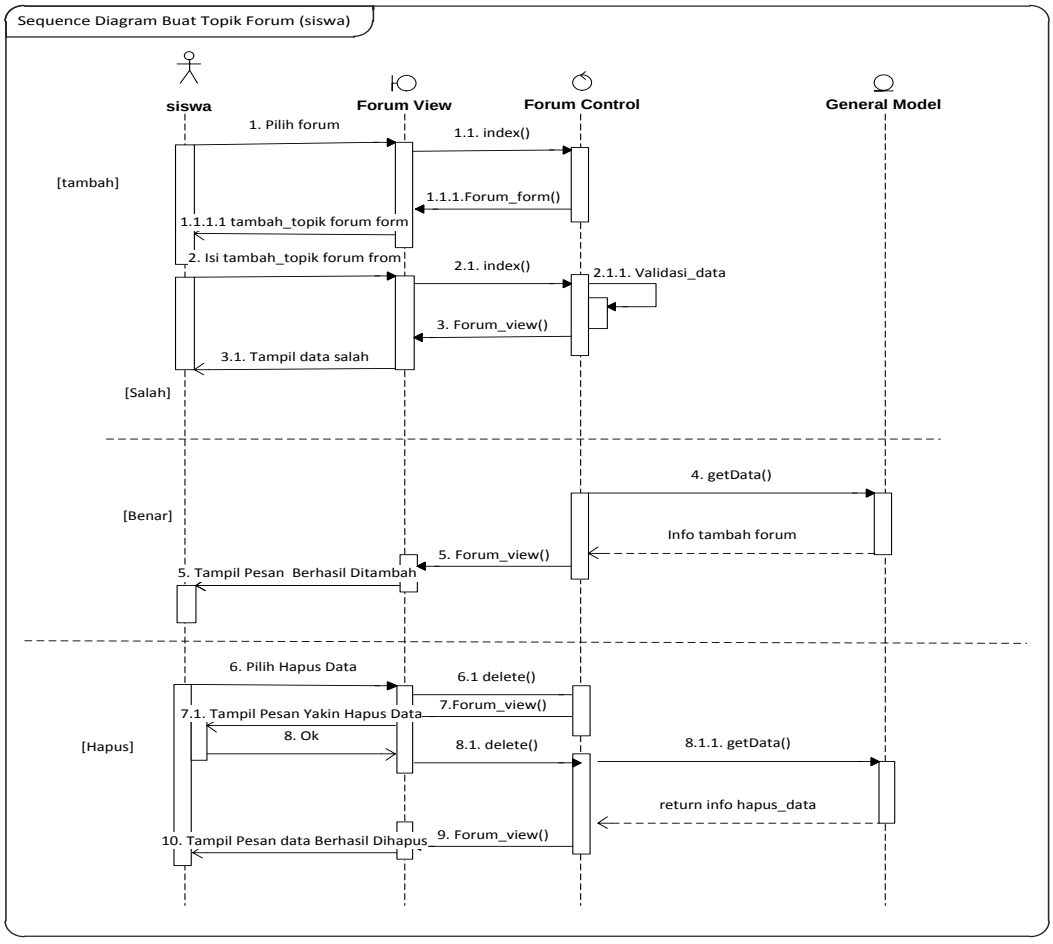
Sequence diagram lihat pengumuman dapat dilihat pada Gambar 3.50.



Gambar 3. 50 Sequence Diagram Lihat Pengumuman

42. Sequence Diagram Buat Topik Forum (siswa)

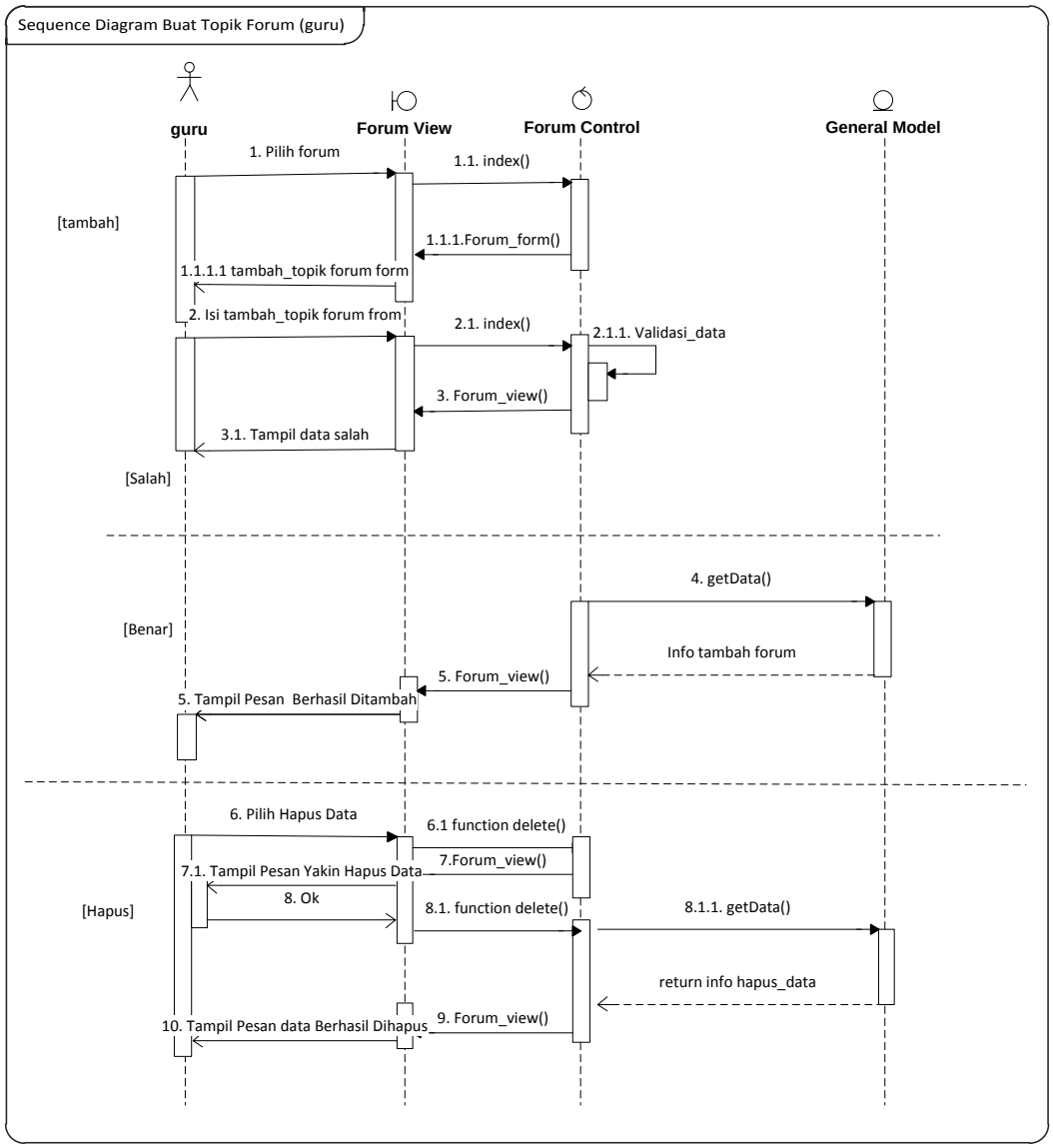
Sequence diagram buat topik forum (siswa) dapat dilihat pada Gambar 3.51.



Gambar 3. 51 Sequence Diagram Buat Topik Forum (siswa)

43. Sequence Diagram Buat Topik Forum (guru)

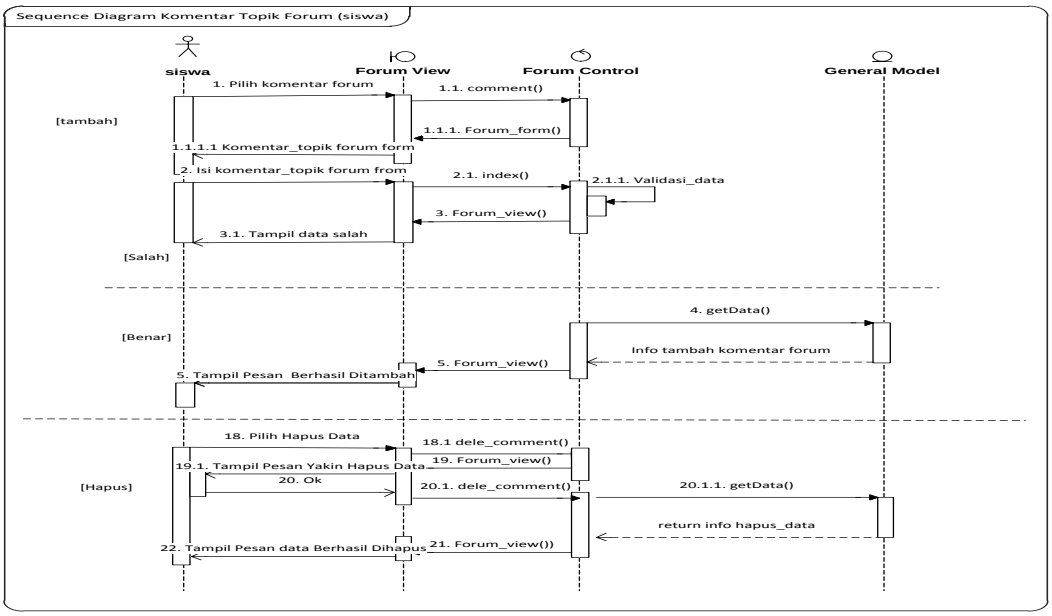
Sequence diagram buat topik forum (guru) dapat dilihat pada Gambar 3.52.



Gambar 3. 52 Sequence Diagram Buat Topik Forum (guru)

44. Sequence Diagram Komentar Topik Forum (siswa)

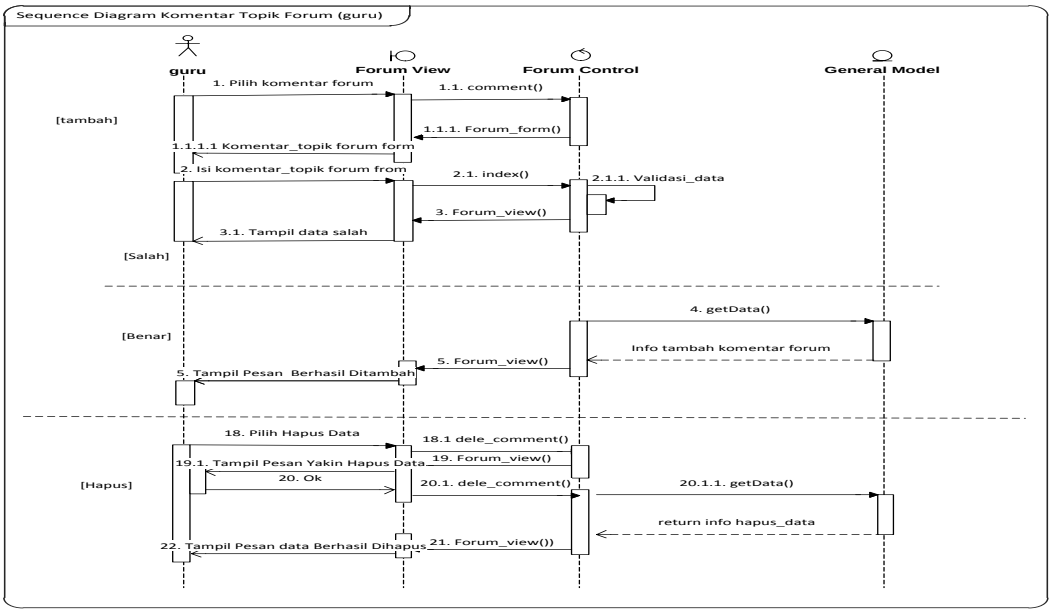
Sequence diagram komentar topik forum (siswa) dapat dilihat pada Gambar 3.53.



Gambar 3. 53 Sequence Diagram Komentar Topik Forum (siswa)

45. Sequence Diagram Komentar Topik Forum (guru)

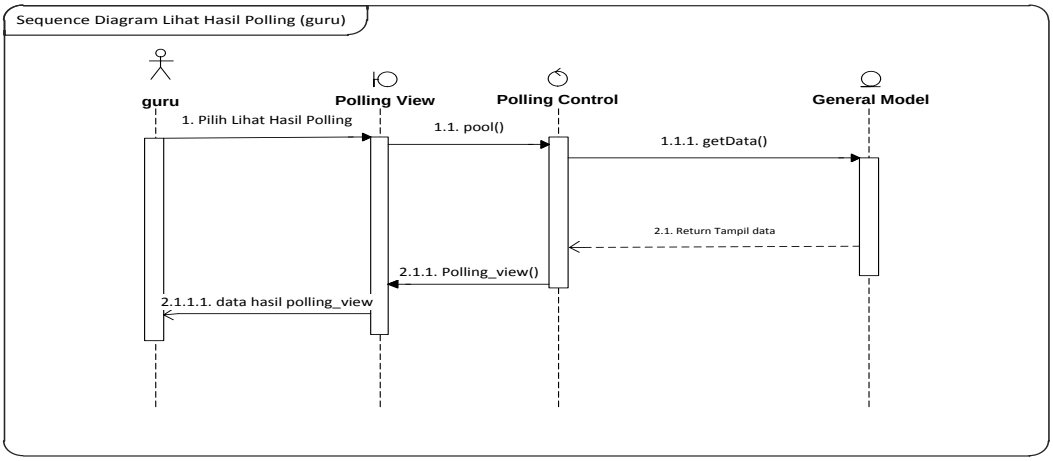
Sequence diagram komentar topik forum (guru) dapat dilihat pada Gambar 3.54.



Gambar 3. 54 Sequence Diagram Komentar Topik Forum (guru)

46. Sequence Diagram Lihat Hasil Polling (guru)

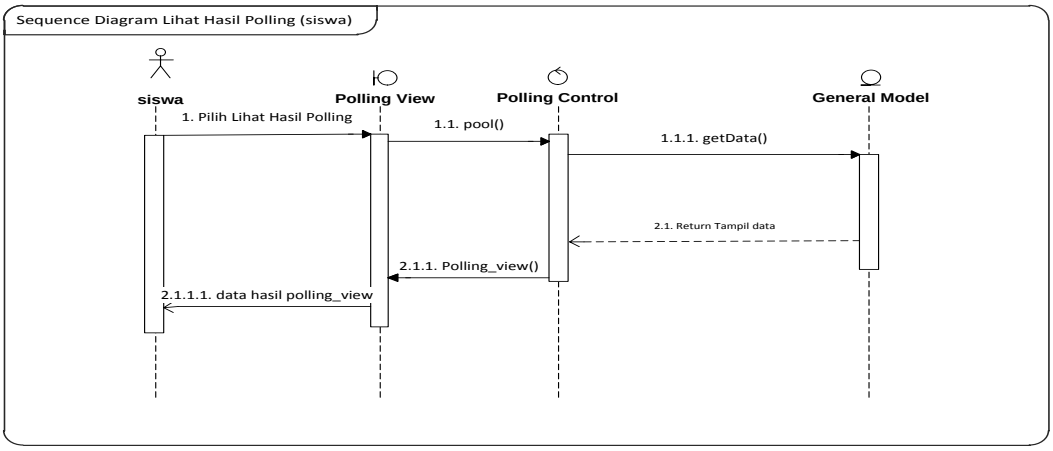
    Sequence diagram lihat hasil polling (guru) dapat dilihat pada Gambar 3.55.



Gambar 3. 55 Sequence Diagram Lihat Hasil Polling (guru)

47. Sequence Diagram Lihat Hasil Polling (siswa)

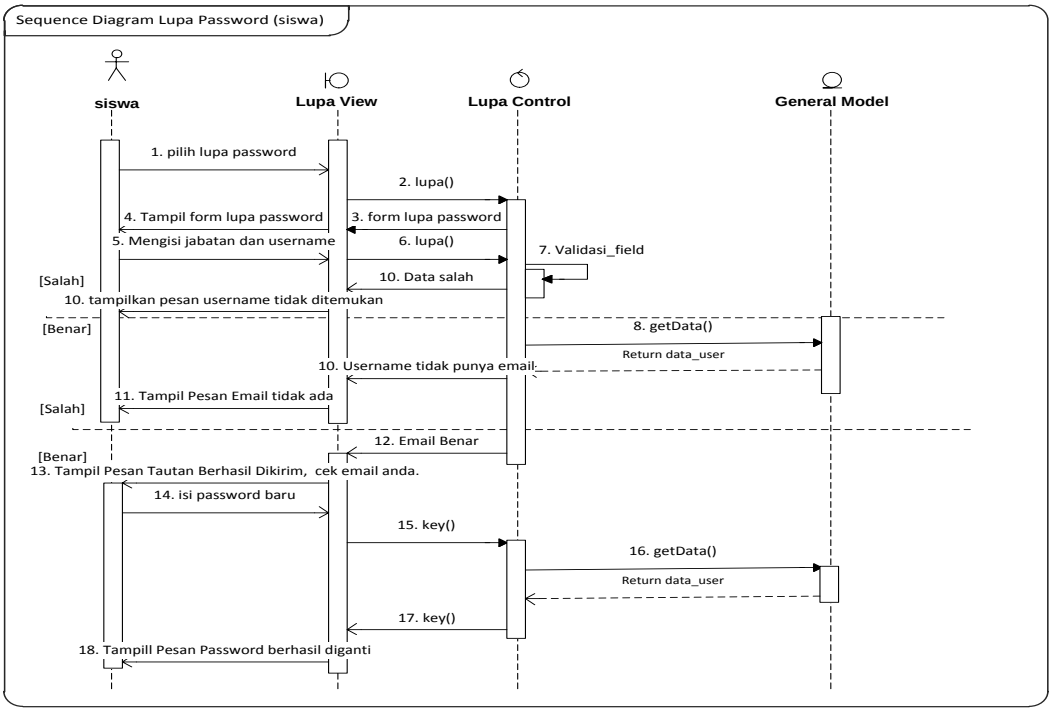
    Sequence diagram lihat hasil polling (siswa) dapat dilihat pada Gambar 3.56.



Gambar 3. 56 Sequence Diagram Lihat Hasil Polling (siswa)

48. Sequence Diagram Lupa Password (siswa)

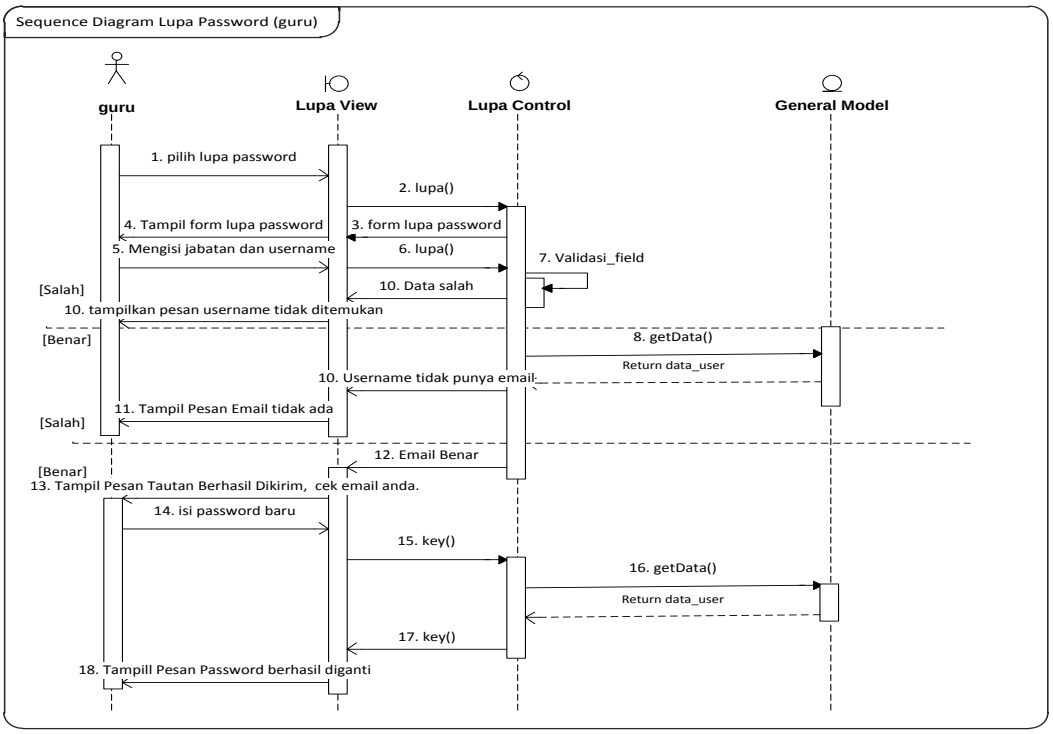
    Sequence diagram lupa password (guru) dapat dilihat pada Gambar 3.57.



Gambar 3. 57 Sequence Diagram Lupa Password (siswa)

49. Sequence Diagram Lupa Password (guru)

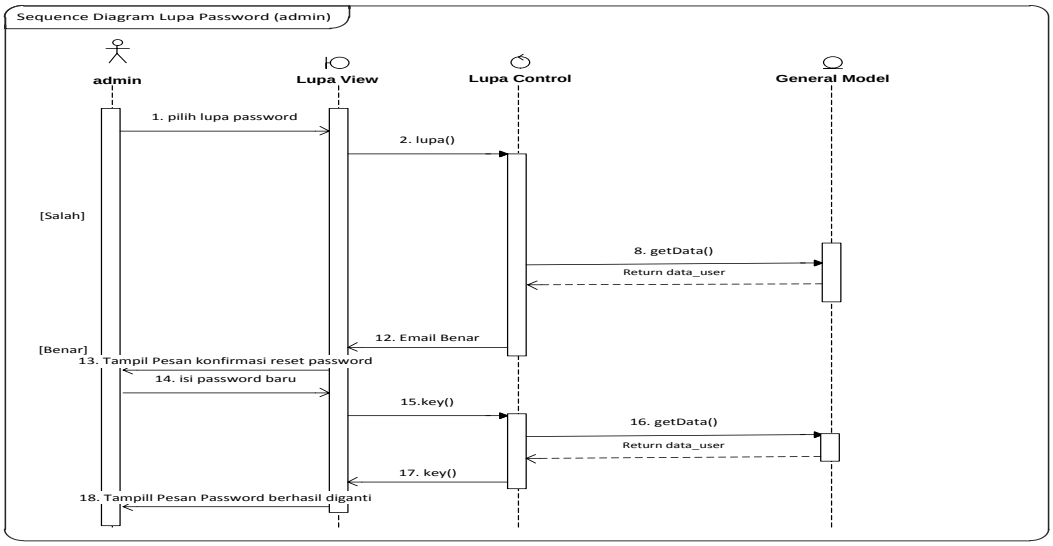
Sequence diagram lupa password (guru) dapat dilihat pada Gambar 3.58



Gambar 3. 58 Sequence Diagram Lupa Password (guru)

50. Sequence Diagram Lupa Password (admin)

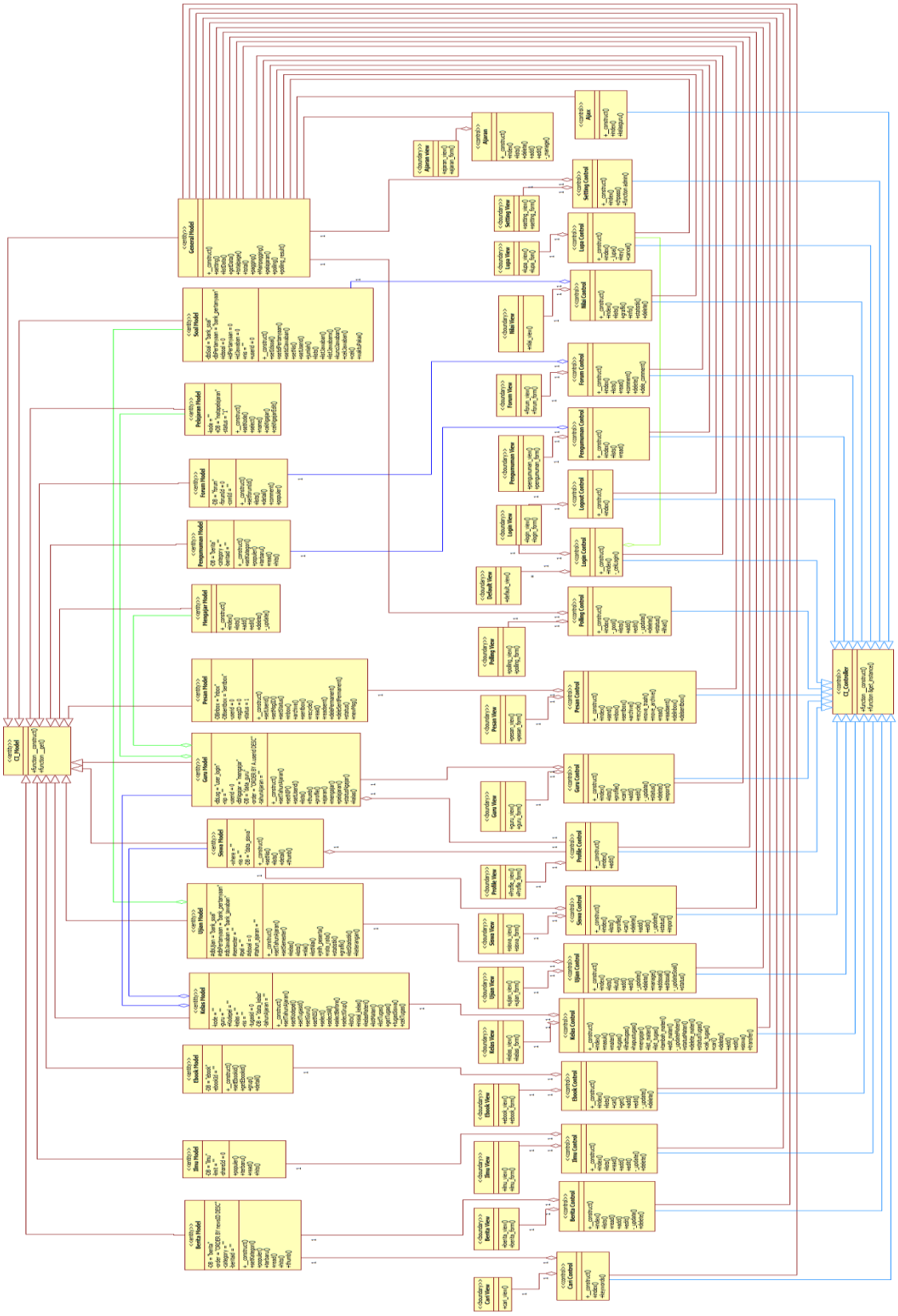
Sequence diagram lupa password (admin) dapat dilihat pada Gambar 3.59.



Gambar 3. 59 Sequence Diagram Lupa Password (admin)

3.1.7.7 *Class* Diagram

Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat di antara mereka. *Class* diagram juga menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan objek tersebut. *Class* diagram *elearning* SMA Negeri 3 Sumedang dapat dilihat pada Gambar 3.60.



Gambar 3. 60 Class Diagram Elearning SMA Negeri 3 Sumedang

Setiap kelas pada Gambar 3.60 memiliki jenis kelasnya masing-masing seperti yang terlihat pada Tabel 3.36.

Tabel 3. 36 Deskripsi Class Diagram

Deskripsi Class		
No	**Nama**	**Jenis**
1.	Berita Control	Control
2.	Ilmu Control	Control
3.	Ebook Control	Control
4.	Kelas Control	Control
5.	Ujian Control	Control
6.	Siswa Control	Control
7.	Profile Control	Control
8.	Guru Control	Control
9.	Pesan Control	Control
10.	Polling Control	Control
11.	Login Control	Control
12.	Logout Control	Control
13.	Pengumuman Control	Control
14.	Forum Control	Control
15.	Nilai Control	Control
16.	Lupa Control	Control
17.	Setting Control	Control
18.	Cari Control	Control
19.	Ajax Control	Control
20.	Ajaran	Control
21.	Berita View	Boundary
22.	Ilmu View	Boundary
23.	Ebook View	Boundary
24.	Kelas View	Boundary

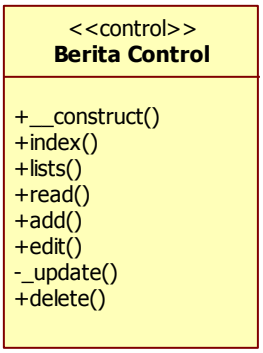
Deskripsi Class		
25.	Ujian View	Boundary
26.	Siswa View	Boundary
27.	Profile View	Boundary
28.	Guru View	Boundary
29.	Pesan View	Boundary
30.	Polling View	Boundary
31.	Login View	Boundary
32.	Logout View	Boundary
33.	Pengumuman View	Boundary
34.	Forum View	Boundary
35.	Nilai View	Boundary
36.	Lupa View	Boundary
37.	Setting View	Boundary
38.	Ajaran View	Boundary
39.	Berita Model	Entity
40.	Ilmu Model	Entity
41.	Ebook Model	Entity
42.	Kelas Model	Entity
43.	Ujian Model	Entity
44.	Siswa Model	Entity
45.	Guru Model	Entity
46.	Pesan Model	Entity
47.	Mengajar Model	Entity
48.	Pengumuman Model	Entity
49.	Forum Model	Entity
50.	Pelajaran Model	Entity

Deskripsi Class		
51.	Soal Model	Entity
52.	General Model	Entity

Deskripsi dari setiap class yang ada dijelaskan sebagai berikut :

1. *Class* Berita Control

Class Berita Control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data berita dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* Berita Control dapat dilihat pada Gambar 3.61.



Gambar 3. 61 Deskripsi Class Berita Control

a. Deskripsi Atribut

Class Berita Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Berita Control dapat dilihat pada Tabel 3.37

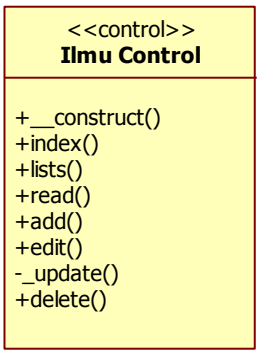
Tabel 3. 37 Deskripsi Operasi Class Berita Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil

Deskripsi Operasi Class	
lists()	Prosedur yang dipanggil untuk menampilkan data berita. Memanggil *class* General Model untuk menampilkan list data
read()	Prosedur untuk untuk melihat berita. Memanggil *class* Berita Model dan *class* Berita View untuk menampilkan berita.
add()	Prosedur untuk melakukan proses penambahan data berita berdasarkan data masukan dari form tambah dan akan memanggil *class* Berita Model untuk melakukan proses tambah data kategori dan menyimpannya pada *database*.
edit()	Prosedur untuk melakukan perubahan data berita, akan memanggil *class* Berita Model untuk mengambil data kemudian memanggil *class* Berita View untuk menampilkan form ubah.
_update()	Prosedur untuk memproses data masukan dari form, memanggil *class* Berita Model untuk menyimpan data yang sudah diubah.
delete()	Prosedur untuk menghapus data berita, akan memanggil *class* Berita Model untuk melakukan penghapusan data pada *database*.

2. *Class* Ilmu Control

Class Ilmu Control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data ilmu dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* Ilmu Control dapat dilihat pada Gambar 3.62.



Gambar 3. 62 Deskripsi Class Ilmu Control

a. Deskripsi Atribut

Class Ilmu Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Ilmu Control dapat dilihat pada Tabel 3.38

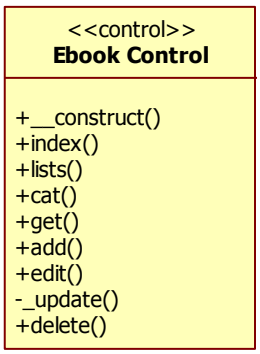
Tabel 3. 38 Deskripsi Operasi Class Ilmu Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data ilmu. Memanggil *class* General Model untuk menampilkan list data
read()	Prosedur untuk untuk melihat ilmu. Memanggil *class* Ilmu Model dan *class* Ilmu View untuk menampilkan ilmu.
add()	Prosedur untuk melakukan proses penambahan data ilmu berdasarkan data masukan dari form tambah dan akan memanggil *class* Ilmu Model untuk melakukan proses tambah data kategori dan menyimpannya pada *database*.

Deskripsi Operasi Class	
edit()	Prosedur untuk melakukan perubahan data ilmu, akan memanggil *class* Ilmu Model untuk mengambil data kemudian memanggil *class* Ilmu View untuk menampilkan form ubah.
_update()	Prosedur untuk memproses data masukan dari form, memanggil Ilmu Model untuk menyimpan data yang sudah diubah.
delete()	Prosedur untuk menghapus data ilmu, akan memanggil Ilmu Model untuk melakukan penghapusan data pada *database*.

3. *Class* Ebook Control

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



Gambar 3. 63 Deskripsi Class Ebook Control

a. Deskripsi Atribut

Class Ebook Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Ebook Control dapat dilihat pada Tabel 3.39

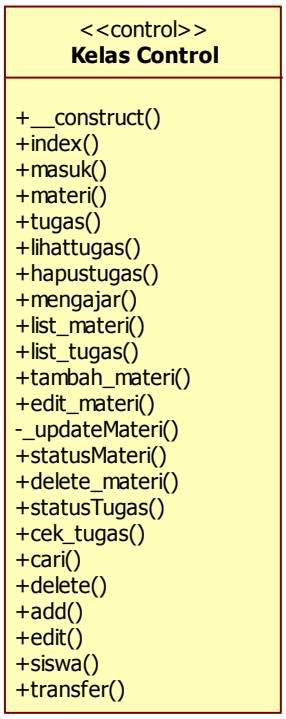
Tabel 3. 39 Deskripsi Operasi Class Ebook Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data ilmu. Memanggil *class* General Model untuk menampilkan list data
cat()	Prosedur yang dipanggil untuk menampilkan data ilmu. Memanggil *class* General Model untuk menampilkan kategori ilmu
get()	Prosedur untuk melakukan proses pengambilan data ebook dan akan memanggil *class* Ebook Model untuk melakukan proses pengambilan pada *database*.
add()	Prosedur untuk melakukan tambah data. Memanggil *class* Ebook View untuk menampilkan form tambah.
edit()	Prosedur untuk melakukan perubahan data ebook, akan memanggil *class* Ebook Model untuk mengambil data kemudian memanggil *class* Ebook View untuk menampilkan form ubah.
_update()	Prosedur untuk memproses data masukan dari form, memanggil *class* Ebook Model untuk menyimpan data yang sudah dimasukan.
delete()	Prosedur untuk menghapus data ebook, akan memanggil *class* Ebook Model untuk

Deskripsi Operasi Class	
	melakukan penghapusan data pada *database*.

4. *Class* Kelas Control

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



Gambar 3. 64 Deskripsi Class Kelas Control

a. Deskripsi Atribut

Class Kelas Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Kelas Control dapat dilihat pada Tabel 3.40

Tabel 3. 40 Deskripsi Operasi Class Kelas Control

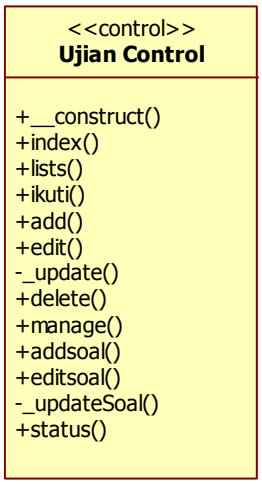
Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
masuk()	Prosedur yang dipanggil untuk masuk kelas, memanggil *class* Kelas Model
materi()	Prosedur untuk mengambil data materi dan memanggil *class* Kelas Model melakukan proses pengambilan pada *database*.
tugas()	Prosedur untuk mengambil data tugas dan memanggil *class* Kelas Model melakukan proses pengambilan pada *database*.
lihattugas()	Prosedur untuk melihat data tugas, akan memanggil *class* General Model untuk menampilkan data tugas
hapustugas()	Prosedur untuk menghapus data tugas, akan memanggil *class* Kelas Model untuk melakukan penghapusan data pada *database*.
mengajar()	Prosedur untuk menampilkan data mengajar akan memanggil class Guru Model untuk mengambil data mengajar
list_materi()	Prosedur yang dipanggil untuk menampilkan data materi. Memanggil *class* General Model untuk menampilkan list data
list_tugas()	Prosedur yang dipanggil untuk menampilkan data tugas. Memanggil *class* General Model untuk menampilkan list data

Deskripsi Operasi Class	
tambah_materi()	Prosedur untuk melakukan tambah materi. Memanggil *class* Kelas View untuk menampilkan form tambah materi.
edit_materi()	Prosedur untuk melakukan ubah materi. Memanggil *class* Kelas View untuk menampilkan form ubah materi.
_updateMateri()	Prosedur untuk memproses data masukan dari form, memanggil *class* Guru Model untuk menyimpan data yang sudah dimasukan.
statusMateri()	Prosedur untuk meaktifkan atau menonaktifkan materi
delete_materi()	Prosedur untuk menghapus data materi, akan memanggil *class* Kelas Model untuk melakukan penghapusan data pada *database*.
statusTugas()	Prosedur untuk meaktifkan atau menonaktifkan tugas
cek_tugas()	Prosedur untuk melihat tugas yang dikirim oleh siswa, akan memanggil *class* Kelas Model untuk mengambil data tugas.
cari()	Prosedur untuk mencari data kelas, akan memanggil *class* General Model untuk menampilkan data kelas.
delete()	Prosedur untuk menghapus data kelas, akan memanggil *class* Kelas Model untuk melakukan penghapusan data pada *database*.
add()	Prosedur untuk melakukan proses penambahan data kelas berdasarkan data masukan dari form tambah dan akan memanggil *class* Kelas Model untuk melakukan proses tambah data kategori

Deskripsi Operasi Class	
	dan menyimpannya pada *database*.
edit()	Prosedur untuk melakukan perubahan data kelas, akan memanggil *class* Kelas Model untuk mengambil data kemudian memanggil *class* Kelas View untuk menampilkan form ubah.
siswa()	Prosedur untuk menampilkan siswa di dalam kelas, akan memanggil *class* Kelas Model untuk melakukan mengambil data
transfer()	Prosedur untuk proses pindah siswa dari kelas, akan memanggil *class* Kelas Model dan Siswa Model untuk melakukan mengambil data

5. *Class* Ujian Control

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



Gambar 3. 65 Deskripsi Class Ujian Control

a. Deskripsi Atribut

Class Ujian Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Ujian Control dapat dilihat pada Tabel 3.41

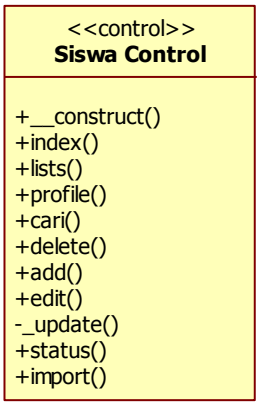
Tabel 3. 41 Deskripsi Operasi Class Ujian Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data Ujian. Memanggil *class* General Model untuk menampilkan list data
ikuti()	Prosedur untuk mengikuti ujian. Memanggil *class* Ujian Model untuk mengambil data ujian
add()	Prosedur untuk melakukan tambah data. Memanggil *class* Ujian View untuk menampilkan form tambah.
edit()	Prosedur untuk melakukan perubahan data ujian, akan memanggil *class* Ujian Model untuk mengambil data kemudian memanggil *class* Ujian View untuk menampilkan form ubah.
_update()	Prosedur untuk memproses data masukan dari form, memanggil *class* Ujian Model untuk menyimpan data yang sudah dimasukan.
delete()	Prosedur untuk menghapus data ujian, akan memanggil *class* Ujian Model untuk melakukan penghapusan data pada *database*.
manage()	Prosedur untuk mengelola ujian, akan memanggil *class* Soal Model untuk melakukan pengambilan data soal ujian pada *database*.

Deskripsi Operasi Class	
addsoal()	Prosedur untuk menambah soal ujian, akan memanggil *class* Soal Model untuk menyimpan data yang sudah dimasukan.
editsoal()	Prosedur untuk ubah soal ujian, akan memanggil *class* Soal Model untuk menyimpan data yang sudah diubah.
_updateSoal()	Prosedur untuk memproses data masukan dari form, memanggil *class* Soal Model untuk menyimpan data soal yang sudah dimasukan.
status()	Prosedur untuk meaktifkan atau menonaktifkan ujian

6. *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 Deskripsi Class Siswa Control

a. Deskripsi Atribut

Class Siswa Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Siswa Control dapat dilihat pada Tabel 3.42

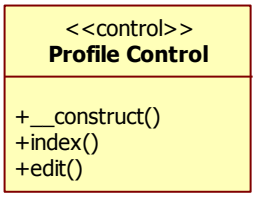
Tabel 3. 42 Deskripsi Operasi Class Siswa Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data Ujian. Memanggil *class* General Model untuk menampilkan list data
profile()	Prosedur untuk menampilkan profil siswa. Memanggil *class* Siswa Model untuk mengambil data siswa dari *database*
cari()	Prosedur untuk mencari data siswa dan memanggil *class* Siswa Model untuk mengambil data dan menampilkannya.
delete()	Prosedur untuk menghapus data siswa, akan memanggil *class* Siswa Model untuk melakukan penghapusan data pada *database*.
add()	Prosedur untuk melakukan tambah data. Memanggil *class* Siswa View untuk menampilkan form tambah.
edit()	Prosedur untuk melakukan perubahan data siswa, akan memanggil *class* Siswa Model untuk mengambil data kemudian memanggil *class* Siswa View untuk menampilkan form ubah.
_update()	Prosedur untuk memproses data masukan dari form, memanggil *class* Siswa Model untuk

Deskripsi Operasi Class	
	menyimpan data yang sudah dimasukan.
status()	Prosedur untuk meaktifkan atau menonaktifkan siswa
import()	Prosedur untuk memasukan data siswa, akan memanggil *class* General Model untuk menampilkan data.

7. *Class* Profile Control

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



Gambar 3. 67 Deskripsi Class Profile Control

a. Deskripsi Atribut

Class Profile Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Profile Control dapat dilihat pada Tabel 3.43

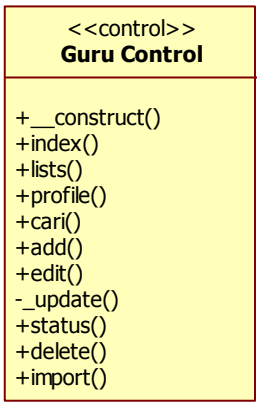
Tabel 3. 43 Deskripsi Operasi Class Profile Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil

Deskripsi Operasi Class	
edit()	Prosedur yang dipanggil untuk mengubah profil pengguna akan memanggil *class* Siswa Model untuk mengambil data siswa atau *class* Guru Model untuk mengambil data guru dari *database*

8. *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 3.68.



Gambar 3. 68 Deskripsi Class Guru Control

a. Deskripsi Atribut

Class Guru Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Guru Control dapat dilihat pada Tabel 3.44

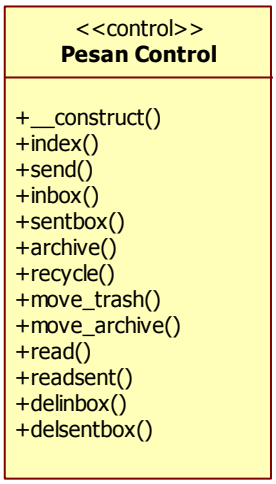
Tabel 3. 44 Deskripsi Operasi Class Guru Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika

Deskripsi Operasi Class	
	class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data Guru. Memanggil *class* General Model untuk menampilkan list data
profile()	Prosedur untuk menampilkan profil guru. Memanggil *class* Guru Model untuk mengambil data siswa dari *database*
cari()	Prosedur untuk mencari data guru dan memanggil *class* Guru Model untuk mengambil data dan menampilkannya.
add()	Prosedur untuk melakukan tambah data. Memanggil *class* Guru View untuk menampilkan form tambah.
edit()	Prosedur untuk melakukan perubahan data guru, akan memanggil *class* Guru Model untuk mengambil data kemudian memanggil *class* Guru View untuk menampilkan form ubah.
_update()	Prosedur untuk memproses data masukan dari form, memanggil *class* Guru Model untuk menyimpan data yang sudah dimasukan.
status()	Prosedur untuk meaktifkan atau menonaktifkan guru
delete()	Prosedur untuk menghapus data guru, akan memanggil *class* Guru Model untuk melakukan penghapusan data pada *database*.
import()	Prosedur untuk memasukan data guru, akan memanggil *class* General Model untuk menampilkan data.

9. *Class* Pesan Control

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



Gambar 3. 69 Deskripsi Class Pesan Control

a. Deskripsi Atribut

Class Pesan Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Pesan Control dapat dilihat pada Tabel 3.45

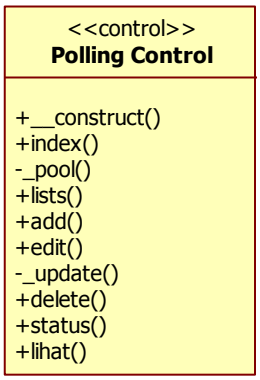
Tabel 3. 45 Deskripsi Operasi Class Pesan Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
send()	Prosedur yang dipanggil untuk mengirim pesan.
inbox()	Prosedur untuk mellihat pesan masuk dan memanggil *class* Pesan Model untuk mengambil

Deskripsi Operasi Class	
	data pesan dari *database*.
sentbox()	Prosedur untuk melihat pesan keluar dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
archive()	Prosedur untuk melihat pesan yang diarsipkan dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
recycle()	Prosedur untuk melihat pesan di tong sampah dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
move_trash()	Prosedur untuk memindahkan pesan ke tong sampah dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
move_archive()	Prosedur untuk memindahkan pesan ke arsip dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
read()	Prosedur untuk membaca pesan dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
readsent()	Prosedur untuk membaca pesan keluar dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
delinbox()	Prosedur untuk menghapus pesan masuk dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.
delsentbox()	Prosedur untuk menghapus pesan keluar dan memanggil *class* Pesan Model untuk mengambil data pesan dari *database*.

10. *Class* Polling Control

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



Gambar 3. 70 Deskripsi Class Polling Control

a. Deskripsi Atribut

Class Polling Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Polling Control dapat dilihat pada Tabel 3.46

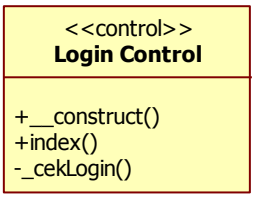
Tabel 3. 46 Deskripsi Operasi Class Polling Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
_pool()	Prosedur untuk mengikuti polling dan memanggil *class* General Model untuk mengambil data polling dari *database*.
lists()	Prosedur yang dipanggil untuk menampilkan data polling. Memanggil *class* General Model

Deskripsi Operasi Class	
	untuk menampilkan list data
add()	Prosedur untuk melakukan tambah data. Memanggil *class* Polling View untuk menampilkan form tambah.
edit()	Prosedur untuk melakukan perubahan data polling, akan memanggil *class* General Model untuk mengambil data kemudian memanggil *class* Polling View untuk menampilkan form ubah.
_update()	Prosedur untuk memproses data masukan dari form, memanggil *class* General Model untuk menyimpan data yang sudah dimasukan.
delete()	Prosedur untuk menghapus data guru, akan memanggil *class* General Model untuk melakukan penghapusan data pada *database*.
status()	Prosedur untuk meaktifkan atau menonaktifkan polling
lihat()	Prosedur untuk melihat hasil polling, memanggil *class* General Model untuk menampilkan hasil polling.

11. *Class* Login Control

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



Gambar 3. 71 Deskripsi Class Login Control

a. Deskripsi Atribut

Class Login Control tidak memiliki atribut.

b. Deskripsi Operasi

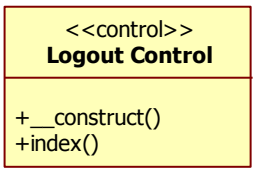
Deskripsi operasi class Login Control dapat dilihat pada Tabel 3.47

Tabel 3. 47 Deskripsi Operasi Class Login Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
_cekLogin()	Prosedur yang dipanggil untuk memvalidasi data pengguna yang dimasukan pada form Login, akan menampilkan kesalahan jika data yang dimasukan tidak valid.

12. *Class* Logout Control

Class Logout Control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada logout dan bertugas untuk memanggil *class-class* yang lainnya. Atribut dan operasi yang dimiliki *class* Logout Control dapat dilihat pada Gambar 3.72.



Gambar 3. 72 Deskripsi Class Logout Control

a. Deskripsi Atribut

Class Logout Control tidak memiliki atribut.

b. Deskripsi Operasi

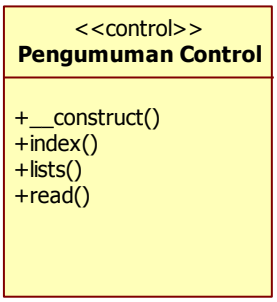
Deskripsi operasi class Logout Control dapat dilihat pada Tabel 3.48

Tabel 3. 48 Deskripsi Operasi Class Logout Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil

13. *Class* Pengumuman Control

Class Pengumuman Control 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 Control dapat dilihat pada Gambar 3.73.



Gambar 3. 73 Deskripsi Class Pengumuman Control

a. Deskripsi Atribut

Class Pengumuman Control tidak memiliki atribut.

b. Deskripsi Operasi

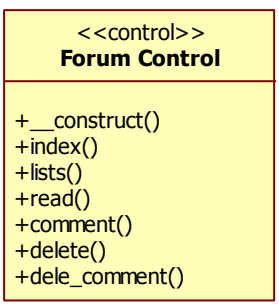
Deskripsi operasi class Pengumuman Control dapat dilihat pada Tabel 3.49

Tabel 3. 49 Deskripsi Operasi Class Pengumuman Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data pengumuman. memanggil *class* General Model untuk menampilkan list data
read()	Prosedur untuk membaca pengumuman dan memanggil *class* Pengumuman Model untuk mengambil data pengumuman dari *database*.

14. *Class* Forum Control

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



Gambar 3. 74 Deskripsi Class Forum Control

a. Deskripsi Atribut

Class Forum Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Forum Control dapat dilihat pada Tabel 3.50

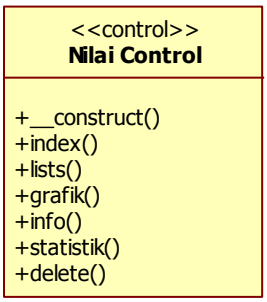
Tabel 3. 50 Deskripsi Operasi Class Forum Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data forum. memanggil *class* General Model untuk menampilkan list data
read()	Prosedur untuk membuat atau melihat topik forum dan memanggil *class* Forum Model untuk mengambil data pengumuman dari *database*.
comment()	Prosedur untuk komentar topik forum dan memanggil *class* Forum Model untuk mengambil data pengumuman dari *database*.
delete()	Prosedur untuk menghapus data forum, akan memanggil *class* Forum Model untuk melakukan penghapusan data pada *database*.
dele_comment()	Prosedur untuk menghapus data komentar topik forum, akan memanggil *class* Forum Model untuk melakukan penghapusan data pada *database*.

15. *Class* Nilai Control

Class Nilai Control merupakan *class* yang digunakan untuk mengatur proses pengolahan pada data nilai dan bertugas untuk memanggil *class-class* yang

lainnya. Atribut dan operasi yang dimiliki *class* Nilai Control dapat dilihat pada Gambar 3.75.



Gambar 3. 75 Deskripsi Class Nilai Control

a. Deskripsi Atribut

Class Nilai Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Nilai Control dapat dilihat pada Tabel 3.51

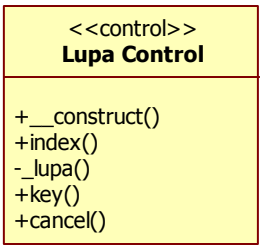
Tabel 3. 51 Deskripsi Operasi Class Nilai Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data. memanggil *class* General Model untuk menampilkan list data
grafik()	Prosedur yang dipanggil untuk menampilkan grafik nilai. memanggil *class* Ujian Model untuk mengambil data nilai rata-rata ujian
info()	Prosedur yang dipanggil untuk menampilkan data nilai ujian siswa. memanggil *class* Ujian Model untuk mengambil data nilai ujian siswa

Deskripsi Operasi Class	
statistik()	Prosedur yang dipanggil untuk menampilkan data statistik nilai ujian siswa. memanggil *class* Ujian Model untuk mengambil data nilai ujian siswa
delete()	Prosedur untuk menghapus data nilai ujian siswa

16. *Class* Lupa Control

Class Lupa Control merupakan *class* yang digunakan untuk mengatur proses pada lupa password. Atribut dan operasi yang dimiliki *class* Lupa Control dapat dilihat pada Gambar 3.76.



Gambar 3. 76 Deskripsi Class Lupa Control

a. Deskripsi Atribut

Class Lupa Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Lupa Control dapat dilihat pada Tabel 3.52

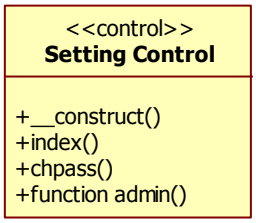
Tabel 3. 52 Deskripsi Operasi Class Lupa Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
_lupa()	Prosedur yang dipanggil untuk lupa password
key()	Prosedur yang dipanggil untuk mengganti

Deskripsi Operasi Class	
	password baru
cancel()	Prosedur yang dipanggil untuk membatalkan lupa password

17. *Class* Setting Control

Class Setting Control merupakan *class* yang digunakan untuk mengatur proses pada mengelola pengaturan umum. Atribut dan operasi yang dimiliki *class* Setting Control dapat dilihat pada Gambar 3.77.



Gambar 3. 77 Deskripsi Class Setting Control

a. Deskripsi Atribut

Class Setting Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Setting Control dapat dilihat pada Tabel 3.53

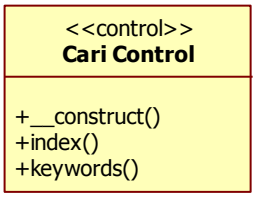
Tabel 3. 53 Deskripsi Operasi Class Setting Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil, mengatur tahun ajaran dan semester.
chpass()	Prosedur yang dipanggil untuk *reset* username dan password login pengguna
admin()	Prosedur yang dipanggil untuk *reset* username

Deskripsi Operasi Class	
	dan password login admin

18. *Class* Cari Control

Class Cari Control merupakan *class* yang digunakan untuk mengatur proses pada pencarian berita dan ilmu. Atribut dan operasi yang dimiliki *class* Cari Control dapat dilihat pada Gambar 3.78.



Gambar 3. 78 Deskripsi Class Cari Control

a. Deskripsi Atribut

Class Cari Control tidak memiliki atribut.

b. Deskripsi Operasi

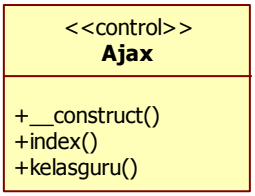
Deskripsi operasi class Cari Control dapat dilihat pada Tabel 3.54

Tabel 3. 54 Deskripsi Operasi Class Cari Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
keywords()	Prosedur yang dipanggil untuk mencari berita atau ilmu.

19. *Class* Ajax Control

Class Ajax Control merupakan *class* yang digunakan untuk mengatur tentang pengolahan respon ajax. Atribut dan operasi yang dimiliki *class* Ajax Control dapat dilihat pada Gambar 3.79.



Gambar 3. 79 Deskripsi Class Ajax Control

a. Deskripsi Atribut

Class Ajax Control tidak memiliki atribut.

b. Deskripsi Operasi

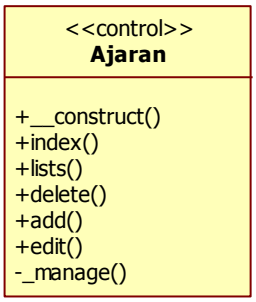
Deskripsi operasi class Ajax Control dapat dilihat pada Tabel 3.55

Tabel 3. 55 Deskripsi Operasi Class Ajax Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
kelasguru()	Prosedur yang dipanggil untuk menangani proses mengelola kelas guru

20. *Class* Ajaran Control

Class Ajaran Control merupakan *class* yang digunakan untuk mengatur tentang pengolahan tahun ajaran. Atribut dan operasi yang dimiliki *class* Ajaran Control dapat dilihat pada Gambar 3.80.



Gambar 3. 80 Deskripsi Class Ajaran Control

a. Deskripsi Atribut

Class Ajaran Control tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Ajaran Control dapat dilihat pada Tabel 3.56

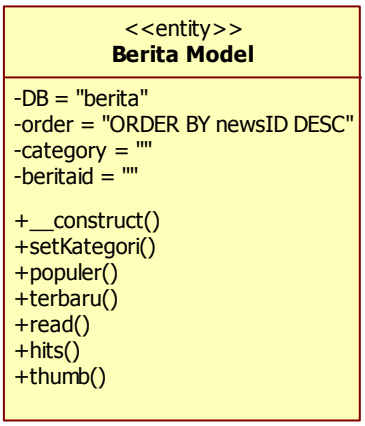
Tabel 3. 56 Deskripsi Operasi Class Ajaran Control

Deskripsi Operasi Class	
Nama	**Keterangan**
__construct()	Merupakan Prosedur yang dijalankan ketika *class* pertama kali diakses.
index()	Merupakan Prosedur yang dijalankan ketika class pertama kali dipanggil
lists()	Prosedur yang dipanggil untuk menampilkan data polling. Memanggil *class* General Model untuk menampilkan list data
delete()	Prosedur untuk menghapus data tahun ajaran, akan memanggil *class* General Model untuk melakukan penghapusan data pada *database*.
add()	Prosedur untuk melakukan tambah data. Memanggil *class* Polling View untuk menampilkan form tambah.
edit()	Prosedur untuk melakukan perubahan data tahun ajaran, akan memanggil *class* General Model

Deskripsi Operasi Class	
	untuk mengambil data kemudian memanggil *class* Ajaran View untuk menampilkan form ubah.
_manage()	Prosedur untuk mengatur tahun ajaran. Memanggil *class* General Model untuk menampilkan data

21. *Class* Berita Model

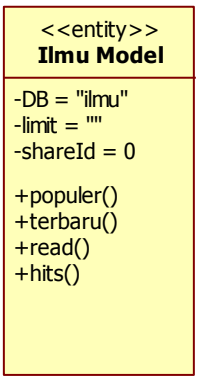
Class Berita Model merupakan class yang berisi prosedur-prosedur untuk mengolah data berita. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Berita Model dapat dilihat pada Gambar 3.81.



Gambar 3. 81 Deskripsi Class Berita Model

22. *Class* Ilmu Model

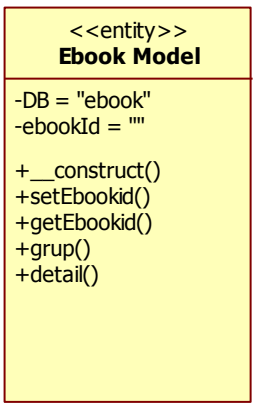
Class Ilmu Model merupakan class yang berisi prosedur-prosedur untuk mengolah data ilmu. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Ilmu Model dapat dilihat pada Gambar 3.82.



Gambar 3. 82 Deskripsi Class Ilmu Model

23. *Class* Ebook Model

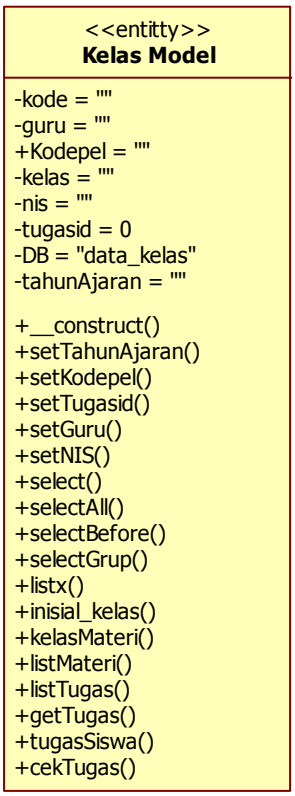
Class Ebook Model merupakan class yang berisi prosedur-prosedur untuk mengolah data *ebook*. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Ebook Model dapat dilihat pada Gambar 3.83.



Gambar 3. 83 Deskripsi Class Ebook Model

24. *Class* Kelas Model

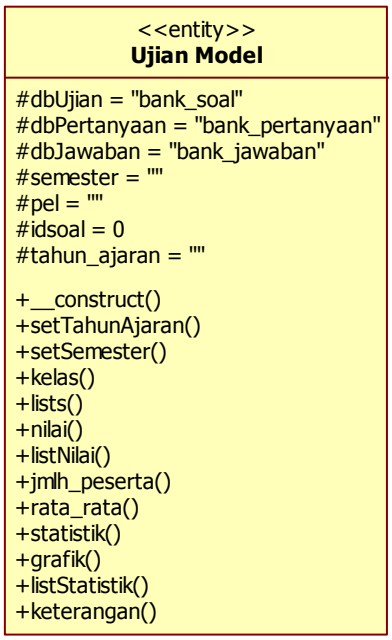
Class Kelas Model merupakan class yang berisi prosedur-prosedur untuk mengolah data kelas. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Kelas Model dapat dilihat pada Gambar 3.84.



Gambar 3. 84 Deskripsi Class Kelas Model

25. *Class* Ujian Model

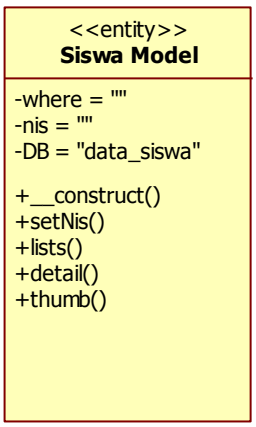
Class Ujian Model merupakan class yang berisi prosedur-prosedur untuk mengolah data ujian. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Ujian Model dapat dilihat pada Gambar 3.85.



Gambar 3. 85 Deskripsi Class Ujian Model

26. *Class* Siswa Model

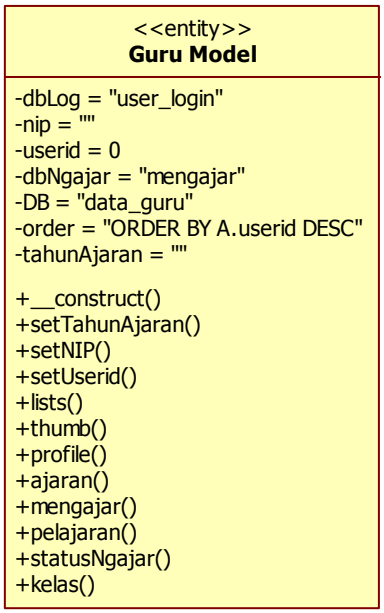
Class Siswa Model merupakan class yang berisi prosedur-prosedur untuk mengolah data Siswa. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Siswa Model dapat dilihat pada Gambar 3.86.



Gambar 3. 86 Deskripsi Class Siswa Model

27. *Class* Guru Model

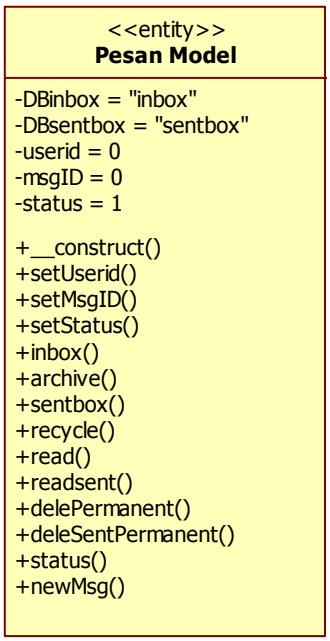
Class Guru Model merupakan class yang berisi prosedur-prosedur untuk mengolah data guru. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Guru Model dapat dilihat pada Gambar 3.87.



Gambar 3. 87 Deskripsi Class Guru Model

28. *Class* Pesan Model

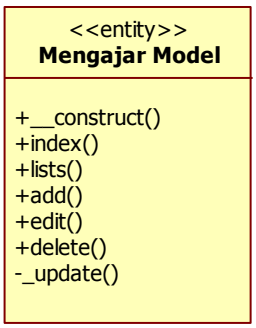
Class Pesan Model merupakan class yang berisi prosedur-prosedur untuk mengolah data pesan. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Pesan Model dapat dilihat pada Gambar 3.87.



Gambar 3. 88 Deskripsi Class Pesan Model

29. *Class* Mengajar Model

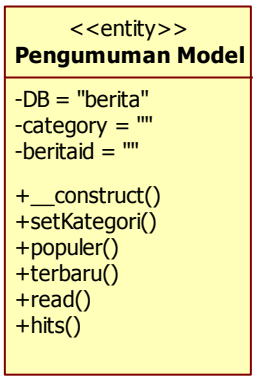
Class Mengajar Model merupakan class yang berisi prosedur-prosedur untuk mengolah data kelas. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Mengajar Model dapat dilihat pada Gambar 3.89.



Gambar 3. 89 Deskripsi Class Mengajar Model

30. *Class* Pengumuman Model

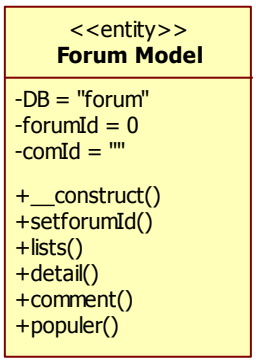
Class Pengumuman Model merupakan class yang berisi prosedur-prosedur untuk mengolah data pengumuman. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Pengumuman Model dapat dilihat pada Gambar 3.90.



Gambar 3. 90 Deskripsi Class Pengumuman Model

31. *Class* Forum Model

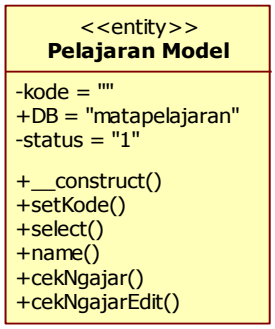
Class Forum Model merupakan class yang berisi prosedur-prosedur untuk mengolah data forum. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Forum Model dapat dilihat pada Gambar 3.91.



Gambar 3. 91 Deskripsi Class Forum Model

32. *Class* Pelajaran Model

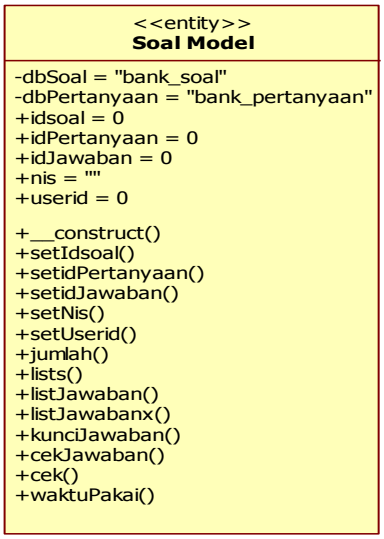
Class Pelajaran Model merupakan class yang berisi prosedur-prosedur untuk mengolah data pelajaran. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Pelajaran Model dapat dilihat pada Gambar 3.92.



Gambar 3. 92 Deskripsi Class Pelajaran Model

33. *Class* Soal Model

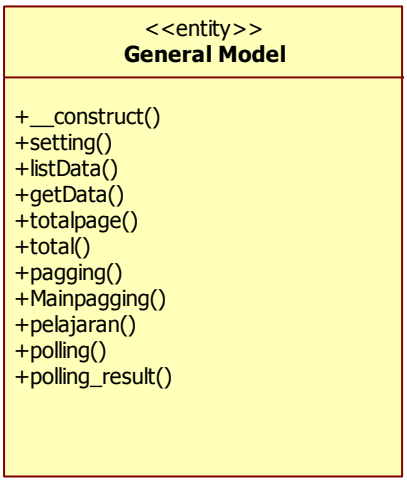
Class Soal Model merupakan class yang berisi prosedur-prosedur untuk mengolah data soal. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class Soal Model dapat dilihat pada Gambar 3.93.



Gambar 3. 93 Deskripsi Class Soal Model

34. *Class* General Model

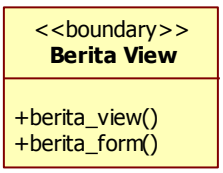
Class General Model merupakan class yang berisi prosedur-prosedur untuk mengolah data. Class ini berhubungan langsung dengan database untuk proses mengambil, menyimpan dan menghapus data. Atribut dan operasi yang dimiliki class General Model dapat dilihat pada Gambar 3.94.



Gambar 3. 94 Deskripsi Class General Model

35. *Class* Berita View

Class Berita View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Berita View dapat dilihat pada Gambar 3.95.



Gambar 3. 95 Deskripsi Class Berita View

a. Deskripsi Atribut Gambar

Class Berita View tidak memiliki atribut.

b. Deskripsi Operasi

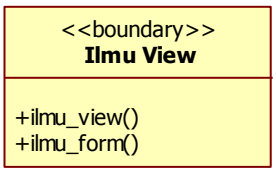
Deskripsi operasi class Berita View dapat dilihat pada Tabel 3.57

Tabel 3. 57 Deskripsi Operasi Class Berita View

Deskripsi Operasi Class	
Nama	**Keterangan**
berita_view()	Prosedur untuk menampilkan data berita yang hasil prosedur dari class Berita Control
berita_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Berita Control

36. *Class* Ilmu View

Class Ilmu View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Ilmu View dapat dilihat pada Gambar 3.96.



Gambar 3. 96 Deskripsi Class Ilmu View

a. Deskripsi Atribut Gambar

Class Ilmu View tidak memiliki atribut.

b. Deskripsi Operasi

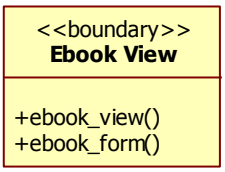
Deskripsi operasi class Ilmu View dapat dilihat pada Tabel 3.58

Tabel 3. 58 Deskripsi Operasi Class Ilmu View

Deskripsi Operasi Class	
Nama	**Keterangan**
ilmu_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Ilmu Control
ilmu_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Ilmu Control

37. *Class* EbookView

Class Ebook View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Ebook View dapat dilihat pada Gambar 3.97.



Gambar 3. 97 Deskripsi Class Ebook View

a. Deskripsi Atribut Gambar

Class Ebook View tidak memiliki atribut.

b. Deskripsi Operasi

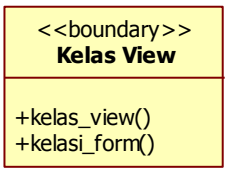
Deskripsi operasi class Ebook View dapat dilihat pada Tabel 3.59

Tabel 3. 59 Deskripsi Operasi Class Ebook View

Deskripsi Operasi Class	
Nama	**Keterangan**
ebook_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Ebook Control
ebook_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Ebook Control

38. *Class* Kelas View

Class Kelas View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Kelas View dapat dilihat pada Gambar 3.98.



Gambar 3. 98 Deskripsi Class Kelas View

a. Deskripsi Atribut Gambar

Class Kelas View tidak memiliki atribut.

b. Deskripsi Operasi

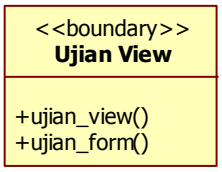
Deskripsi operasi class Kelas View dapat dilihat pada Tabel 3.60.

Tabel 3. 60 Deskripsi Operasi Class Kelas View

Deskripsi Operasi Class	
Nama	**Keterangan**
kelas_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Kelas Control
kelas_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Kelas Control

39. *Class* Ujian View

Class Ujian View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Ujian View dapat dilihat pada Gambar 3.99.



Gambar 3. 99 Deskripsi Class Ujian View

a. Deskripsi Atribut Gambar

Class Ujian View tidak memiliki atribut.

b. Deskripsi Operasi

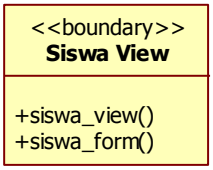
Deskripsi operasi class Ujian View dapat dilihat pada Tabel 3.61

Tabel 3. 61 Deskripsi Operasi Class Ujian View

Deskripsi Operasi Class	
Nama	**Keterangan**
ujian_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Ujian Control
ujian_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Ujian Control

40. *Class* Siswa View

Class Siswa View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Siswa View dapat dilihat pada Gambar 3.100.



Gambar 3. 100 Deskripsi Class Siswa View

a. Deskripsi Atribut Gambar

Class Siswa View tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Siswa View dapat dilihat pada Tabel 3.62

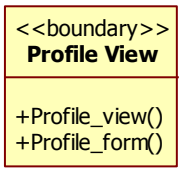
Tabel 3. 62 Deskripsi Operasi Class Siswa View

Deskripsi Operasi Class	
Nama	**Keterangan**
siswa_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Siswa Control
siswa_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data

Deskripsi Operasi Class	
	pada *class* Siswa Control

41. *Class* Profile View

Class Profile View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Profile View dapat dilihat pada Gambar 3.101.



Gambar 3. 101 Deskripsi Class Profile View

a. Deskripsi Atribut Gambar

Class Profile View tidak memiliki atribut.

b. Deskripsi Operasi

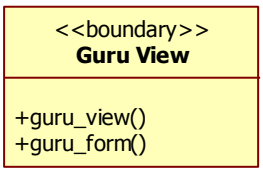
Deskripsi operasi class Profile View dapat dilihat pada Tabel 3.63

Tabel 3. 63 Deskripsi Operasi Class Profile View

Deskripsi Operasi Class	
Nama	**Keterangan**
profile_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Profile Control
profile_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Profile Control

42. *Class* Guru View

Class Guru View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Guru View dapat dilihat pada Gambar 3.102.



Gambar 3. 102 Deskripsi Class Guru View

a. Deskripsi Atribut Gambar

Class Guru View tidak memiliki atribut.

b. Deskripsi Operasi

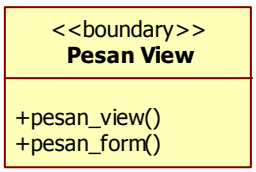
Deskripsi operasi class Guru View dapat dilihat pada Tabel 3.64

Tabel 3. 64 Deskripsi Operasi Class Guru View

Deskripsi Operasi Class	
Nama	**Keterangan**
guru_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Guru Control
guru_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Guru Control

43. *Class* Pesan View

Class Pesan View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Pesan View dapat dilihat pada Gambar 3.103.



Gambar 3. 103 Deskripsi Class Pesan View

a. Deskripsi Atribut Gambar

Class Pesan View tidak memiliki atribut.

b. Deskripsi Operasi

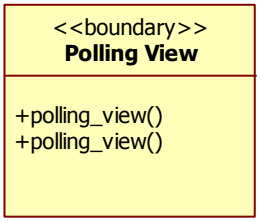
Deskripsi operasi class Pesan View dapat dilihat pada Tabel 3.65

Tabel 3. 65 Deskripsi Operasi Class Pesan View

Deskripsi Operasi Class	
Nama	**Keterangan**
pesan_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Pesan Control
pesan_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Pesan Control

44. *Class* Polling View

Class Polling View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Polling View dapat dilihat pada Gambar 3.104.



Gambar 3. 104 Deskripsi Class Polling View

a. Deskripsi Atribut Gambar

Class Polling View tidak memiliki atribut.

b. Deskripsi Operasi

Deskripsi operasi class Polling View dapat dilihat pada Tabel 3.66

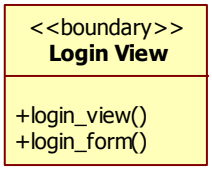
Tabel 3. 66 Deskripsi Operasi Class Polling View

Deskripsi Operasi Class	
Nama	**Keterangan**
polling_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Polling Control
polling_form()	Prosedur untuk menampilkan form ketika

Deskripsi Operasi Class	
	memanggil prosedur tambah data dan ubah data pada *class* Polling Control

45. *Class* Login View

Class Login View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Login View dapat dilihat pada Gambar 3.105.



Gambar 3. 105 Deskripsi Class Login View

a. Deskripsi Atribut Gambar

Class Login View tidak memiliki atribut.

b. Deskripsi Operasi

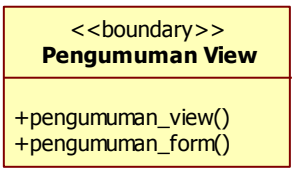
Deskripsi operasi class Login View dapat dilihat pada Tabel 3.67

Tabel 3. 67 Deskripsi Operasi Class Login View

Deskripsi Operasi Class	
Nama	**Keterangan**
login_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Login Control
login_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Login Control

46. *Class* Pengumuman View

Class Pengumuman View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Pengumuman View dapat dilihat pada Gambar 3.106.



Gambar 3. 106 Deskripsi Class Pengumuman View

a. Deskripsi Atribut Gambar

Class Pengumuman View tidak memiliki atribut.

b. Deskripsi Operasi

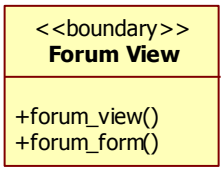
Deskripsi operasi class Pengumuman View dapat dilihat pada Tabel 3.68

Tabel 3. 68 Deskripsi Operasi Class Pengumuman View

Deskripsi Operasi Class	
Nama	**Keterangan**
pengumuman_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Pengumuman Control
pengumuman _form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Pengumuman Control

47. *Class* Forum View

Class Forum View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Forum View dapat dilihat pada Gambar 3.107.



Gambar 3. 107 Deskripsi Class Polling Forum View

a. Deskripsi Atribut Gambar

Class Forum View tidak memiliki atribut.

b. Deskripsi Operasi

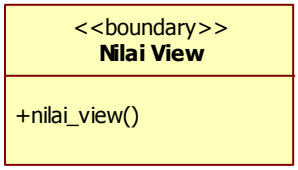
Deskripsi operasi class Forum View dapat dilihat pada Tabel 3.69

Tabel 3. 69 Deskripsi Operasi Class Forum View

Deskripsi Operasi Class	
Nama	**Keterangan**
forum_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Forum Control
forum_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Forum Control

48. *Class* Nilai View

Class Nilai View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Nilai View dapat dilihat pada Gambar 3.108.



Gambar 3. 108 Deskripsi Class Nilai View

a. Deskripsi Atribut Gambar

Class Nilai View tidak memiliki atribut.

b. Deskripsi Operasi

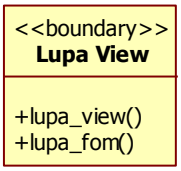
Deskripsi operasi class Nilai View dapat dilihat pada Tabel 370

Tabel 3. 70 Deskripsi Operasi Class Nilai View

Deskripsi Operasi Class	
Nama	**Keterangan**
nilai_view()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Forum Control

49. *Class* Lupa View

Class Lupa View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Lupa View dapat dilihat pada Gambar 3.109.



Gambar 3. 109 Deskripsi Class Lupa View

a. Deskripsi Atribut Gambar

Class Lupa View tidak memiliki atribut.

b. Deskripsi Operasi

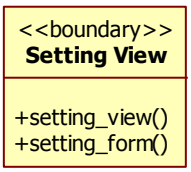
Deskripsi operasi class Lupa View dapat dilihat pada Tabel 3.71.

Tabel 3. 71 Deskripsi Operasi Class Lupa View

Deskripsi Operasi Class	
Nama	**Keterangan**
login_form()	Prosedur untuk menampilkan data ilmu yang hasil prosedur dari *class* Lupa Control
lupa_view()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Lupa Control

50. *Class* Setting View

Class Setting View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Setting View dapat dilihat pada Gambar 3.110.



Gambar 3. 110 Deskripsi Class Setting View

a. Deskripsi Atribut Gambar

Class Setting View tidak memiliki atribut.

b. Deskripsi Operasi

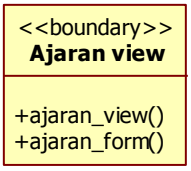
Deskripsi operasi class Setting View dapat dilihat pada Tabel 3.72

Tabel 3. 72 Deskripsi Operasi Class Setting View

Deskripsi Operasi Class	
Nama	**Keterangan**
setting_view()	Prosedur untuk menampilkan data pengaturan yang hasil prosedur dari *class* Setting Control
setting_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Setting Control

51. *Class* Ajaran View

Class Ajaran View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Ajaran View dapat dilihat pada Gambar 3.111.



Gambar 3. 111 Deskripsi Class Ajaran View

a. Deskripsi Atribut Gambar

Class Ajaran View tidak memiliki atribut.

b. Deskripsi Operasi

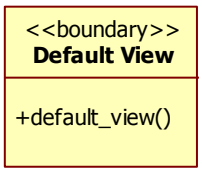
Deskripsi operasi class Ajaran View dapat dilihat pada Tabel 3.73

Tabel 3. 73 Deskripsi Operasi Class Ajaran View

Deskripsi Operasi Class	
Nama	**Keterangan**
ajaran_view()	Prosedur untuk menampilkan data tahun ajaran yang hasil prosedur dari *class* Ajaran Control
ajaran_form()	Prosedur untuk menampilkan form ketika memanggil prosedur tambah data dan ubah data pada *class* Ajaran Control

52. *Class* Default View

Class Default View merupakan *class* yang berisi prosedur-prosedur untuk mengatur tampilan yang berinteraksi langsung dengan user. Atribut dan operasi yang dimiliki *class* Default View dapat dilihat pada Gambar 3.112.



Gambar 3. 112 Deskripsi Class Default View

a. Deskripsi Atribut Gambar

Class Default View tidak memiliki atribut.

b. Deskripsi Operasi

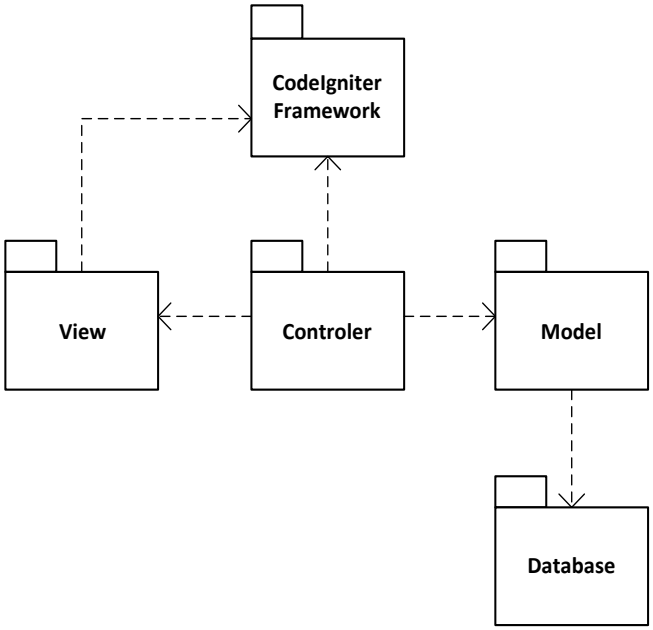
Deskripsi operasi class Default View dapat dilihat pada Tabel 3.73

Tabel 3. 74 Deskripsi Operasi Class Default View

Deskripsi Operasi Class	
Nama	**Keterangan**
default_view()	Prosedur untuk m menampilkan halaman beranda hasil prosedur dari *class* Login Control

3.1.7.8 *Package* Diagram

Package diagram adalah sebuah bentuk pengelompokkan yang memungkinkan untuk mengambil setiap bentuk pada UML dan mengelompokkan elemen-elemennya dalam tingkatan unit yang lebih tinggi. *Package* diagram untuk pengelompokkan class aplikasi e-learning di SMA Negeri 3 Sumedang dapat dilihat pada Gambar 3.113.

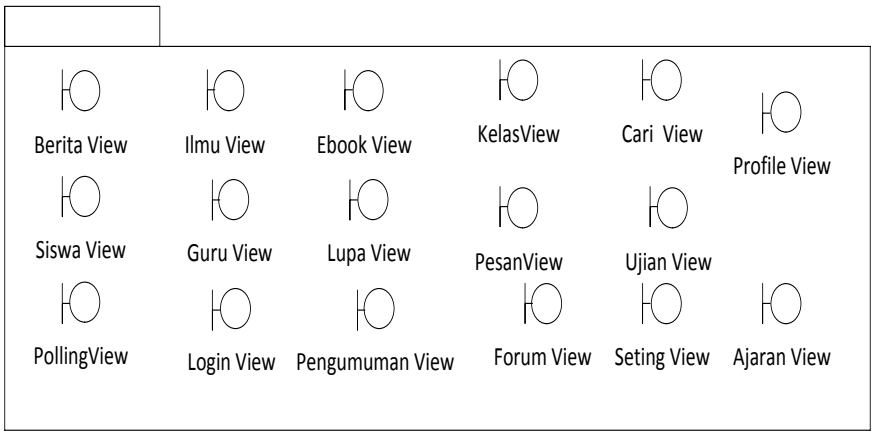


Gambar 3. 113 *Package* Diagram e-learning di SMA Negeri 3 Sumedang

Deskripsi untuk *package* diagram pada Gambar 3.113 adalah sebagai berikut :

1. *Package View*

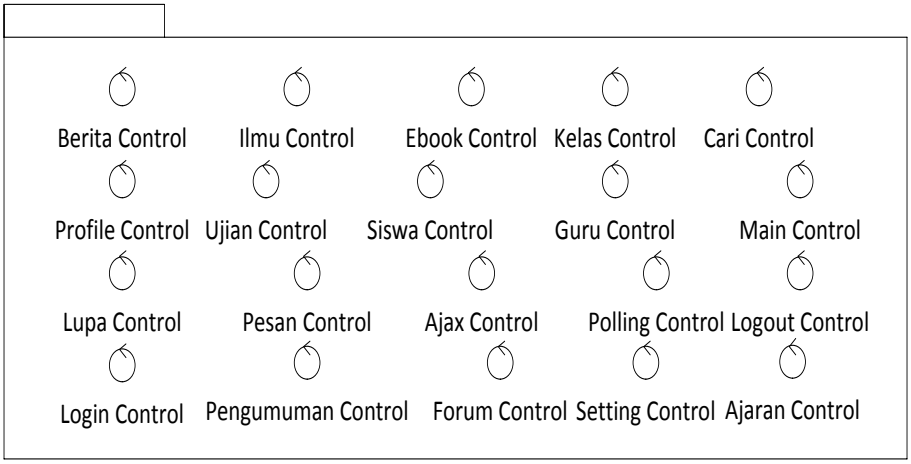
   Package View berisi class-class view yang digunakan untuk menampilkan interface, dapat berupa tabel atau form. Isi dari *Package View* dapat dilihat pada Gambar 3.114.



Gambar 3. 114 Komponen Package View

2. *Package Controler*

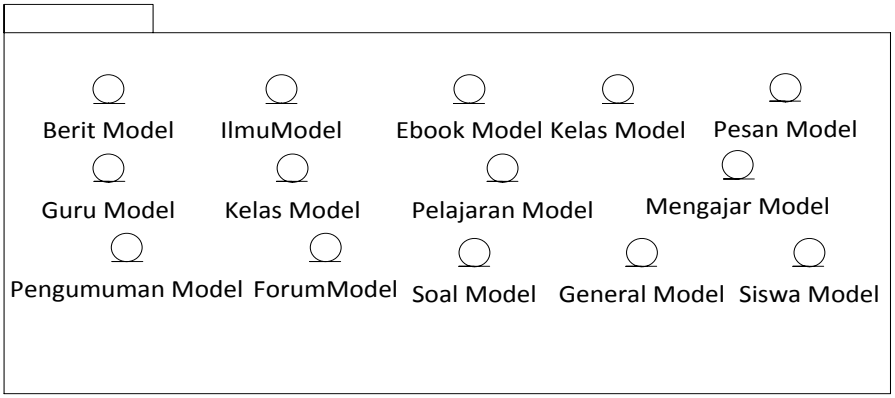
Package Controller berisi class-class control yang merupakan pengatur dalam setiap proses yang ada. Isi dari *Package Controller* dapat dilihat pada Gambar 3.115.



Gambar 3. 115 Komponen Package Controler

3. *Package Model*

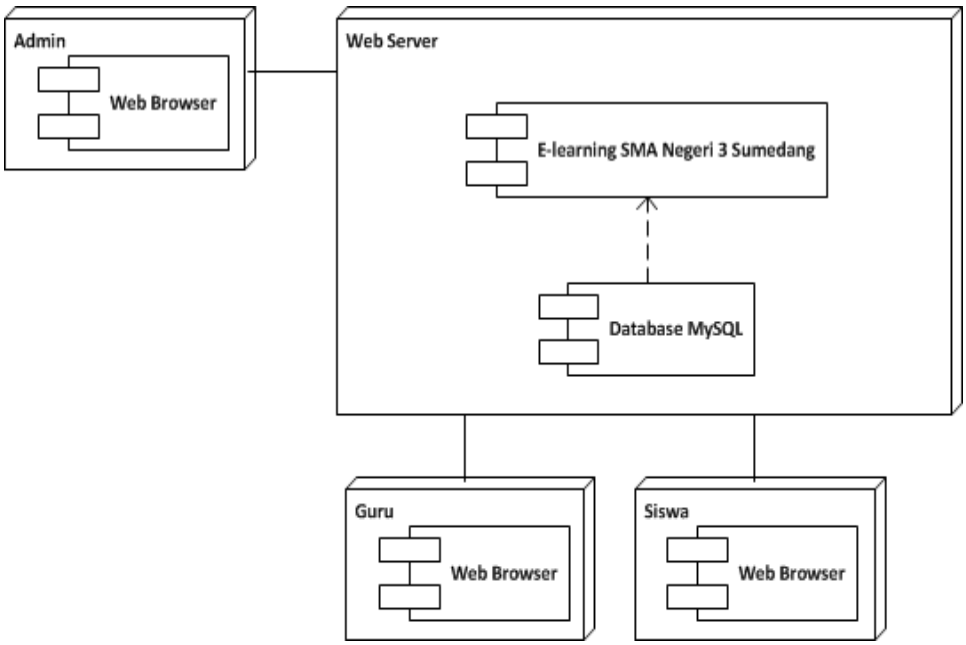
Package Model berisi class-class model yang mengakses langsung pada database untuk dapat mengolah data. Isi dari *Package Model* dapat dilihat pada Gambar 3.116.



Gambar 3. 116 Komponen Package Model

3.1.7.9 *Deployment* Diagram

Deployment diagram menunjukkan susunan fisik sebuah sistem, menunjukkan bagian perangkat lunak mana yang berjalan pada perangkat keras mana. *Deployment* diagram e-learning SMA Negeri 3 Sumedang dapat dilihat pada Gambar 3.117.



Gambar 3. 117 Deployment Diagram E-Learning SMA Negeri 3 Sumedang

3.2 Perancangan Sistem

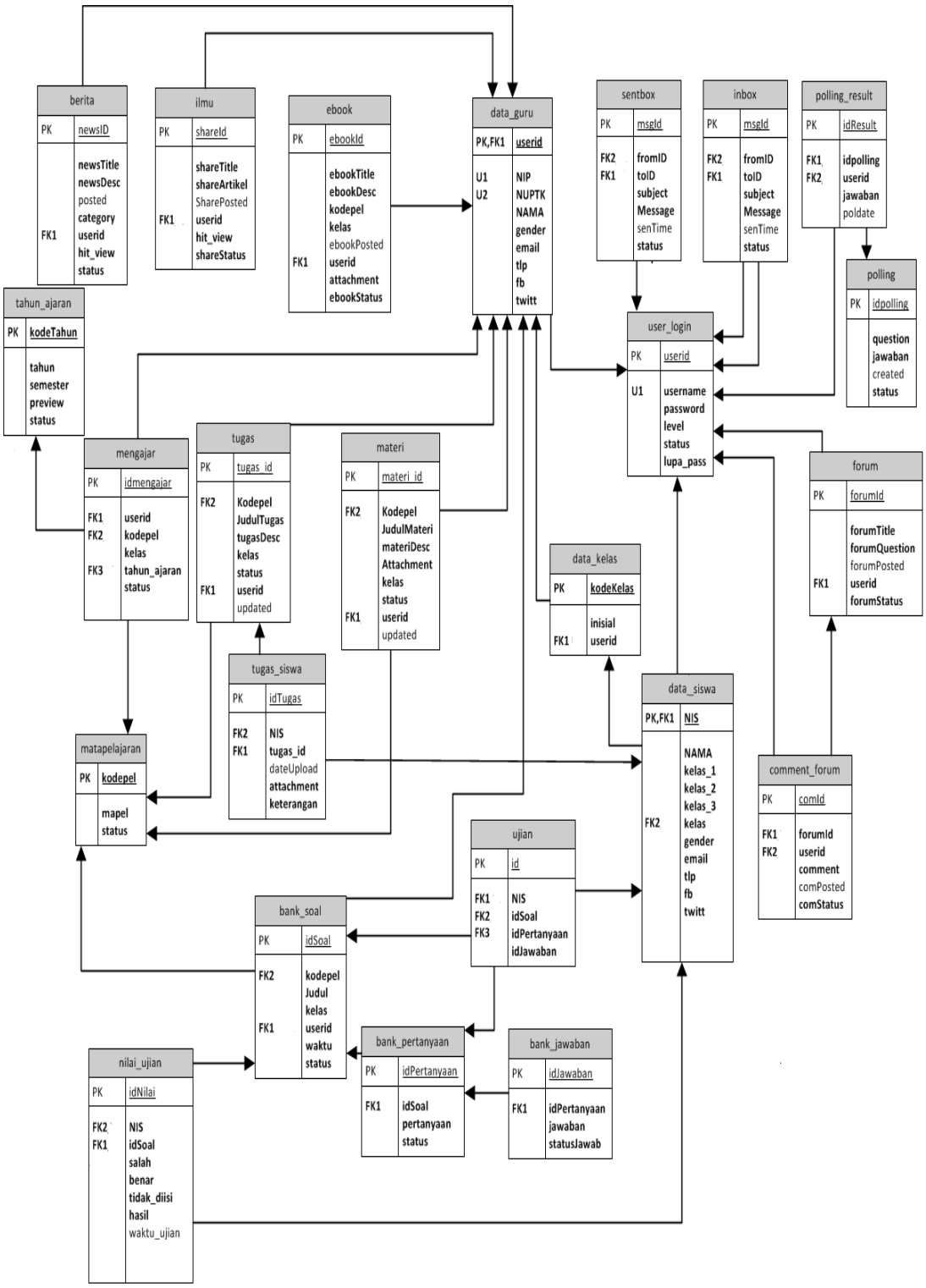
Perancangan merupakan bagian dari metodologi pembangunan suatu perangkat lunak yang harus dilakukan setelah melalui tahapan analisis. Pada bagian ini akan dijelaskan perancangan yang diusulkan.

3.2.1 Perancangan Data

Perancangan data atau lebih dikenal dengan perancangan basis data yaitu menciptakan atau merancang data yang terhubung dan disimpan secara bersama-sama. Untuk menggambarkannya digunakanlah skema relasi dan struktur *file*. Dari dua hasil tersebut, implementasi basis data akan bisa dikerjakan.

3.2.1.1 Diagram Relasi

Diagram relasi merupakan penggambaran hubungan diantara dua tabel atau lebih pada sistem *database*. Tabel yang ada pada aplikasi yang akan dibangun dapat dilihat pada Gambar 3.118.



Gambar 3. 118 Diagram Relasi

3.2.2 Perancangan Stuktur Tabel

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

1. Tabel Berita

   Tabel ini berfungsi untuk menyimpan data berita. Struktur tabel berita dapat dilihat pada Tabel 3.75.

Tabel 3. 75 Stuktur Berita

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
newsID	int	(11)	Primary key	AUTO_INCREMENT
newsTitle	varchar	(255)		
newsDesc	text			
posted	timestamp			
category	int	(11)		
hit_view	int	(11)		
status	int	(1)		1=aktif , 0=tidak aktif
userid	int	(11)	Foreign key	FK *reference* table data_guru (userid)

2. Tabel Ilmu

   Tabel ini berfungsi untuk menyimpan data ilmu. Struktur tabel ilmu dapat dilihat pada Tabel 3.76.

Tabel 3. 76 Stuktur Ilmu

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
shareId	Int	(11)	Primary key	AUTO_INCREMENT
shareTitle	varchar	(255)		
shareArtikel	text			
SharePosted	timestamp			
hit_view	int	(11)		

Stuktur Tabel				
status	int	(1)		1=aktif , 0=tidak aktif
userid	int	(11)	Foreign key	FK *reference* table data_guru (userid)

3. Tabel Ebook

   Tabel ini berfungsi untuk menyimpan data ebook. Struktur tabel ebook dapat dilihat pada Tabel 3.77.

Tabel 3. 77 Stuktur Ebook

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
ebookId	int	(11)	Primary key,	AUTO_INCREMENT
ebookTitle	varchar	(255)		
ebookDesc	text			
kodepel	varchar	(30)		
kelas	int	(11)		
ebookPosted	timestamp			
attachment	varchar	(255)		
ebookStatus	int	(1)		1=aktif , 0=tidak aktif
userid	int	(11)	Foreign key	FK reference tabel data_guru (userid)

4. Tabel User_login

   Tabel ini berfungsi untuk menyimpan data user. Struktur tabel user_login dapat dilihat pada Tabel 3.78.

Tabel 3. 78 Stuktur User_login

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
userid	int	(11)	Primary key	AUTO_INCREMENT
username	Varchar	(255)	Unique	

Stuktur Tabel				
password	Varchar	(255)		
level	Varchar	(10)		Admin, guru, siswa
status	int	(1)		1=aktif , 0=tidak aktif
lupa_pass	Varchar	(255)		

5. Tabel Sentbox

   Tabel ini berfungsi untuk menyimpan data pesan terkirim. Struktur tabel sentbox dapat dilihat pada Tabel 3.79.

Tabel 3. 79 Stuktur Sentbox

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
msgId	int	(11)	Primary key	AUTO_INCREMENT
fromID	int	(11)	Foreign key	FK2 reference tabel user_login (userid)
toID	int	(11)	Foreign key	FK1 reference tabel user_login (userid)
Message	Varchar	(255)		
senTime	Timestamp			
status	int	(1)		1=aktif , 0=tidak aktif

6. Tabel Inbox

   Tabel ini berfungsi untuk menyimpan data pesan masuk. Struktur tabel inbox dapat dilihat pada Tabel 3.80.

Tabel 3. 80 Stuktur Inbox

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
msgId	int	(11)	Primary key	AUTO_INCREMENT
fromID	Varchar	(255)	Foreign key	FK1 reference tabel user_login (userid)

Stuktur Tabel				
toID	Varchar	(255)	Foreign key	FK2 reference tabel user_login (userid)
Message	Int	(11)		
senTime	Timestamp			
status	int	(1)		1=dibaca , 0=belum dibaca

7. Tabel Forum

   Tabel ini berfungsi untuk menyimpan data forum. Struktur tabel forum dapat dilihat pada Tabel 3.81.

Tabel 3. 81 Stuktur Forum

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
forumId	int	(11)	Primary key	AUTO_INCREMENT
forumTitle	varchar	(255)		
forumQuestion	text			
forumPosted	timestamp			
forumStatus	int	(1)		1=aktif , 0=tidak aktif
userid	int	(11)	Foreign key	FK reference tabel user_login (userid)

8. Tabel Comment_forum

   Tabel ini berfungsi untuk menyimpan data komentar forum. Struktur tabel comment_forum dapat dilihat pada Tabel 3.82.

Tabel 3. 82 Stuktur Comment_forum

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
comId	int	(11)	Primary key	AUTO_INCREMENT
forumId	int	(11)	Foreign key	FK *reference* tabel

Stuktur Tabel				
				forum (forumId)
parentId	int	(11)		
comment	text			
comPosted	timestamp			
comStatus	int	(1)		1=aktif , 0=tidak aktif
userid	int	(11)	Foreign key	FK reference tabel user_login (userid)

9. Tabel Data_guru

   Tabel ini berfungsi untuk menyimpan data guru. Struktur tabel data_guru dapat dilihat pada Tabel 3.83.

Tabel 3. 83 Stuktur Data_guru

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
userid	int	(11)	Primary key Foreign key	FK reference tabel user_login (userid)
NIP	varchar	(255)	Unique	
NUPTK	varchar	(255)	Unique	
NAMA	varchar	(255)		
gender	enum	('L', 'P')		Laki_laki, Perempuan
tlp	varchar	(255)		
email	varchar	(255)		
fb	varchar	(255)		
twitt	varchar	(255)		

10. Tabel Data_siswa

Tabel ini berfungsi untuk menyimpan data siswa. Struktur tabel data_siswa dapat dilihat pada Tabel 3.84.

Tabel 3. 84 Stuktur Data_siswa

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
NIS	varchar	(255)	Primary key	
NAMA	varchar	(255)		
kelas_1	varchar	(30)		
kelas_2	varchar	(30)		
kelas_3	varchar	(30)		
kelas	varchar	(30)	Foreign key	FK1 reference tabel data_kelas(kodekelas)
gender	enum	('L', 'P')		Laki_laki, Perempuan
tlp	int	(30)		
email	varchar	(255)		
fb	varchar	(255)		
twitt	varchar	(255)		

11. Tabel Data_kelas

Tabel ini berfungsi untuk menyimpan data kelas. Struktur tabel data_kelas dapat dilihat pada Tabel 3.85.

Tabel 3. 85 Stuktur Data_kelas

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idKelas	int	(11)	Primay Key	AUTO_INCREMENT
kodeKelas	varchar	(30)		
inisial	varchar	(10)		
userid	int	(11)	Foreign key	FK1 reference tabel data_guru (userid)

12. Tabel Mengajar

Tabel ini berfungsi untuk menyimpan data mengajar guru. Struktur tabel megajar dapat dilihat pada Tabel 3.86.

Tabel 3. 86 Stuktur Mengajar

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idmengajar	int	(11)	Primary key	AUTO_INCREMENT
userid	int	(11)	Foreign key	FK1 reference tabel data_guru (userid)
kodepel	varchar	(255)	Foreign key	FK2 reference tabel matapelajaran(kodepel)
kelas	varchar	(255)		
tahun_ajaran	varchar	(30)	Foreign key	FK3 reference tabel tahun_ajaran(tahun)
status	int	(1)		1=aktif , 0=tidak aktif

13. Tabel Materi

Tabel ini berfungsi untuk menyimpan data materi. Struktur tabel materi dapat dilihat pada Tabel 3.87.

Tabel 3. 87 Stuktur Materi

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
materi_id	int	(11)	Primary Key	AUTO_INCREMENT
Kodepel	varchar	(30)	Foreign key	FK2 *reference* tabel matapelajaran (Kodepel)
JudulMateri	varchar	(255)		
materiDesc	text			
Attachment	varchar	(255)		
kelas	text			
status	int	(1)		1=aktif , 0=tidak aktif

Stuktur Tabel				
updated	timestamp			
userid	int	(11)	Foreign key	FK1 reference tabel data_guru (userid)

14. Tabel Tugas

Tabel ini berfungsi untuk menyimpan data tugas. Struktur tabel tugas dapat dilihat pada Tabel 3.88.

Tabel 3. 88 Stuktur Tugas

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
tugas_id	int	(11)	Primary Key	AUTO_INCREMENT
Kodepel	varchar	(30)	Foreign key	FK2 *reference* tabel matapelajaran (Kodepel)
JudulTugas	varchar	(255)		
tugasDesc	text			
kelas	text			
status	int	(1)		
updated	timestamp			
userid	int	(11)	Foreign key	FK1 reference tabel data_guru (userid)

15. Tabel Tugas_siswa

Tabel ini berfungsi untuk menyimpan data tugas siswa. Struktur tabel tugas_siswa dapat dilihat pada Tabel 3.89.

Tabel 3. 89 Stuktur Tugas_siswa

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
id_Tugas	int	(11)	Primary key	AUTO_INCREMENT

Stuktur Tabel				
NIS	int	(11)	Foreign key	FK2 reference tabel data_siswa(NIS)
tugas_id	int	(11)	Foreign key	FK1 reference tabel tugas (tugas_id)
attachment	varchar	(255)		
keterangan	varchar	(255)		

16. Tabel Matapelajaran

Tabel ini berfungsi untuk menyimpan data matapelajaran. Struktur tabel matapelajaran dapat dilihat pada Tabel 3.90.

Tabel 3. 90 Stuktur Matapelajaran

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
Kodepel	varchar	(30)	Primay Key	AUTO_INCREMENT
Maple	varchar	(255)		
Status	Int	(1)		

17. Tabel Ujian

Tabel ini berfungsi untuk menyimpan data ujian. Struktur tabel ujian dapat dilihat pada Tabel 3.91.

Tabel 3. 91 Stuktur Ujian

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
Id	int	(11)	Primary key	AUTO_INCREMENT
NIS	varchar	(30)	Foreign key	FK1 reference tabel data_siswa(NIS)
idSoal	int	(11)	Foreign key	FK2 reference tabel bank_soal(idSoal)
idPertanyaan	int	(11)	Foreign key	FK3 reference tabel bank_soal

Stuktur Tabel				
				(idPertanyaan)
idJawaban	varchar	(255)		

18. Tabel Bank_soal

Tabel ini berfungsi untuk menyimpan data soal. Struktur tabel bank_soal dapat dilihat pada Tabel 3.92.

Tabel 3. 92 Stuktur Bank_soal

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idSoal	int	(11)	Primary Key	AUTO_INCREMENT
Kodepel	varchar	(30)	Foreign key	FK1 reference tabel matapelajaran(kodepel)
JudulMateri	varchar	(255)		
Kelas	text			
Userid	varchar	(255)	Foreign key	FK2 reference tabel data_guru(userid)
Waktu	int	(3)		
Status	int	(1)		

19. Tabel Bank_pertanyaan

Tabel ini berfungsi untuk menyimpan data pertanyaan. Struktur tabel bank_pertanyaan dapat dilihat pada Tabel 3.93.

Tabel 3. 93 Stuktur Bank_pertanyaan

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idPertanyaan	int	(11)	Primay Key	AUTO_INCREMENT
idSoal	int	(11)	Foreign key	FK *reference* tabel bank_soal (idSoal)
pertanyaan	text			

Stuktur Tabel				
Status	int	(1)		

20. Tabel Bank_jawaban

Tabel ini berfungsi untuk menyimpan data jawaban. Struktur tabel bank_jawaban dapat dilihat pada Tabel 3.94.

Tabel 3. 94 Stuktur Bank_jawaban

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idJawaban	int	(11)	Primay Key	AUTO_INCREMENT
idPertanyaan	int	(11)	Foreign key	FK *reference* tabel bank_pertanyaan (idPertanyaan)
Jawaban	text			
statusJawab	int	(1)		

21. Tabel Nilai_ujian

Tabel ini berfungsi untuk menyimpan data nilai. Struktur tabel Nilai_ujian dapat dilihat pada Tabel 3.95.

Tabel 3. 95 Stuktur Nilai_ujian

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idNilai	int	(11)	Primay Key	AUTO_INCREMENT
NIS	varchar	(255)	Foreign key	FK2 *reference* tabel Data_siswa (NIS)
idSoal	int	(11)	Foreign key	FK1*reference* tabel bank_soal (idSoal)
salah	int	(11)		
Benar	int	(11)		
tidak_diisi				

Stuktur Tabel				
Hasil	float			
waktu_ujian	timestamp	(11)		

22. Tabel Polling

Tabel ini berfungsi untuk menyimpan data polling. Struktur tabel polling dapat dilihat pada Tabel 3.96.

Tabel 3. 96 Stuktur Polling

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idpolling	int	(11)	Primay Key	AUTO_INCREMENT
question	varchar	(255)		
jawaban	text			
created	timestamp			
status	enum	('0', '1')		0=tidak aktif, 1=aktif

23. Tabel Polling_result

Tabel ini berfungsi untuk menyimpan data jawaban polling. Struktur tabel polling_result dapat dilihat pada Tabel 3.97.

Tabel 3. 97 Stuktur Polling_result

Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
idResult	int	(11)	Primay Key	AUTO_INCREMENT
idpolling	int	(11)		
userid	int	(11)	Foreign key	FK1 *reference* tabel User_login (userid)
jawaban	varchar	(255)		
poldate	timestamp			

24. Tabel Tahun_Ajaran

Tabel ini berfungsi untuk menyimpan data jawaban tahun ajaran. Struktur tabel tahun_ajaran dapat dilihat pada Tabel 3.98.

Tabel 3. 98 Stuktur Tahun_Ajaran

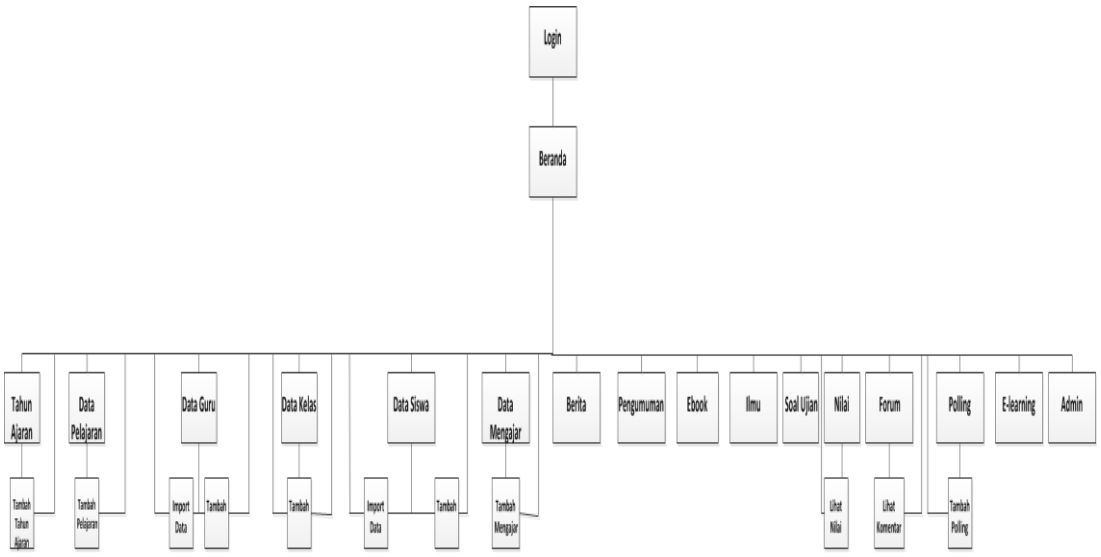
Stuktur Tabel				
Nama Field	**Tipe**	**Panjang**	**Kunci**	**Keterangan**
kodeTahun	varchar	(30)	Primay Key	AUTO_INCREMENT
tahun	int	(11)		
semester	enum			
preview	varchar	(255)		
status	int	(1)		

3.2.3 Perancangan Struktur Menu

Perancangan sruktur menu berisikan menu dan sub menu yang berfungsi untuk memudahkan user di dalam menggunakan sistem. Struktur menu Elearning di SMA Negeri 3 Sumedang adalah sebagai berikut:

3.2.3.1 Stuktur Menu Admin

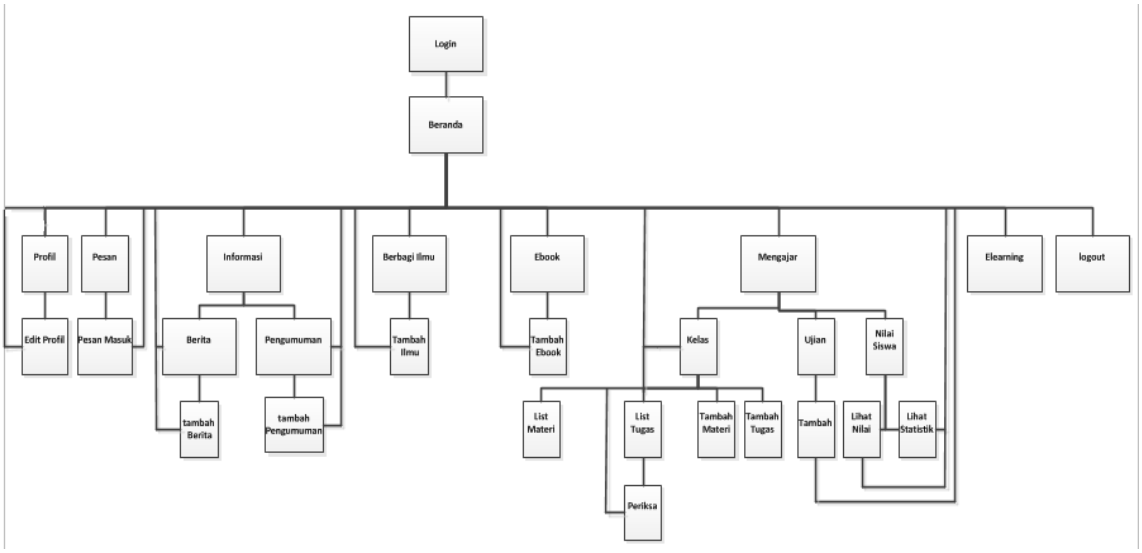
Struktur menu yang berisikan menu dan submenu yang berfungsi untuk memudahkan admin dalam menggunakan aplikasi *e-learning*.



Gambar 3. 119 Stuktur Menu Admin

3.2.3.2 Stuktur Menu Guru

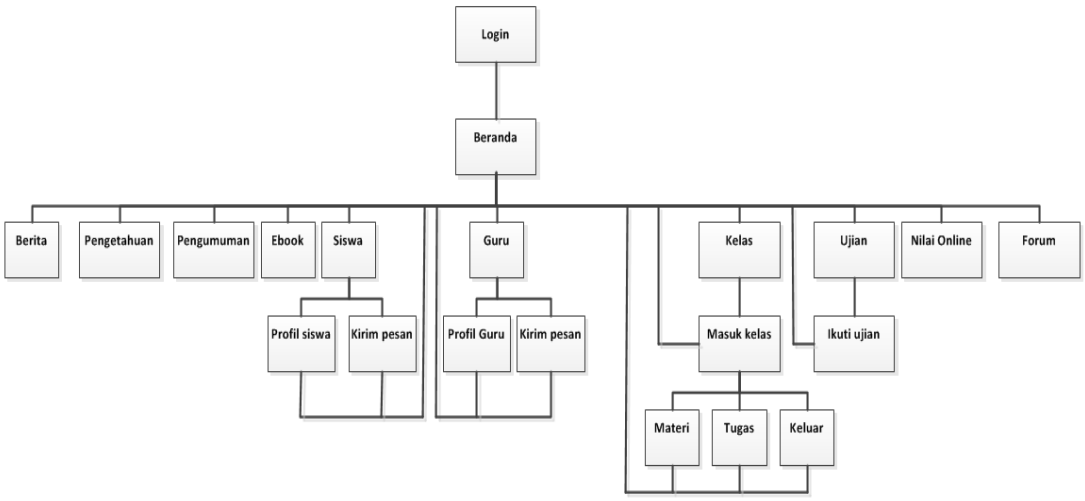
Struktur menu yang berisikan menu dan submenu yang berfungsi untuk memudahkan guru dalam menggunakan aplikasi *e-learning*.



Gambar 3. 120 Stuktur Menu Guru

3.2.3.3 Stuktur Menu Siswa

Struktur menu yang berisikan menu dan submenu yang berfungsi untuk memudahkan siswa dalam menggunakan aplikasi *e-learning*.



Gambar 3. 121 Stuktur Menu Siswa

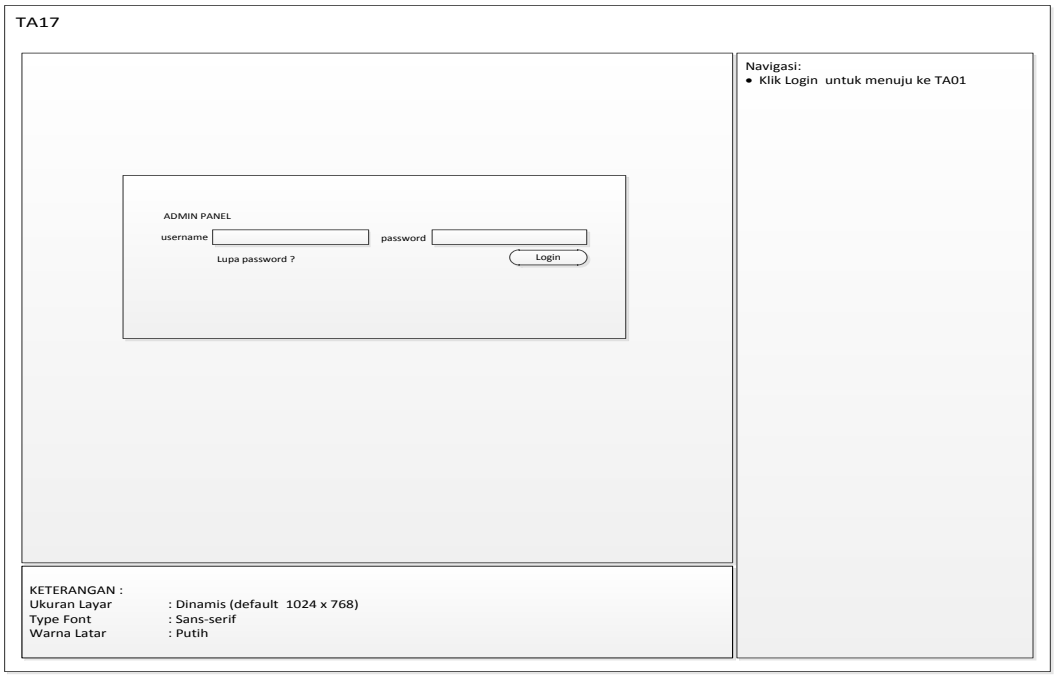
3.2.4 Perancangan Antar Muka

Perancangan antar muka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun sehingga akan mempermudah dalam mengimplementasikan serta akan memudahkan dalam pembuatan aplikasi yang mudah digunakan pengguna.

3.2.4.1 Tampilan Antar Muka Admin

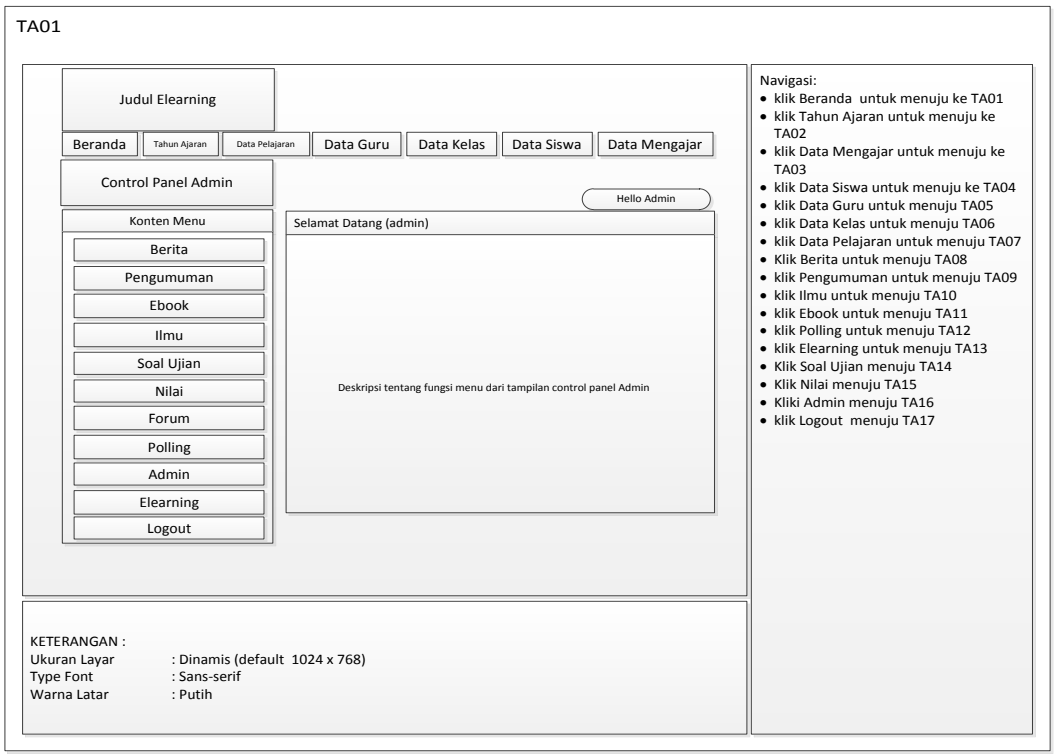
Antar muka admin bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun.

1. Halaman Login (Admin)



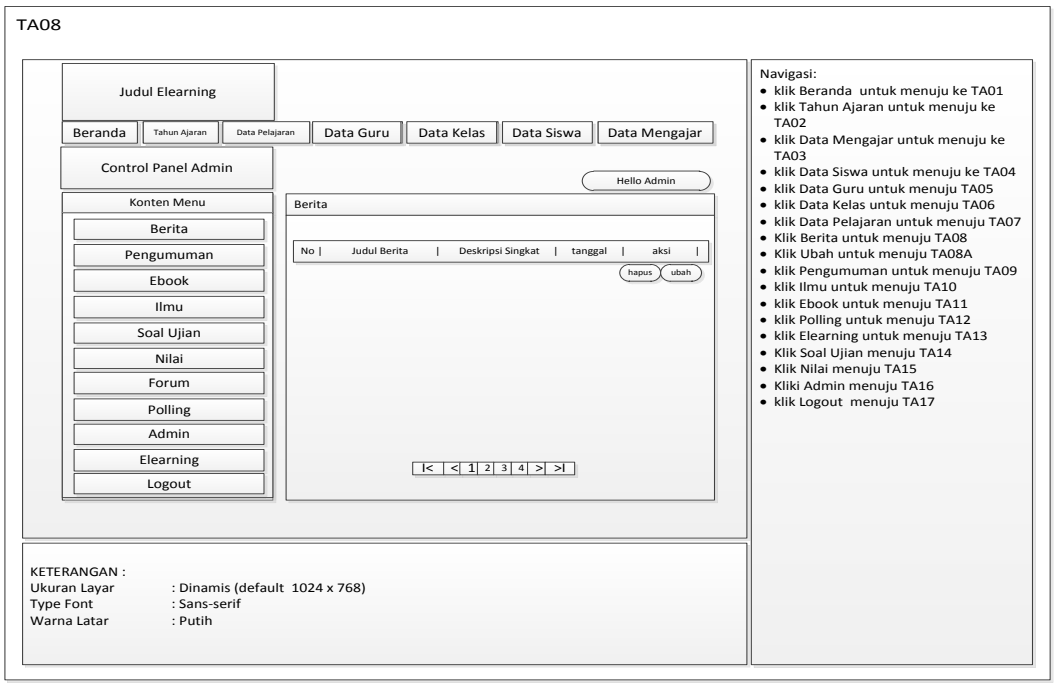
Gambar 3. 122 Tampilan Halaman Login (Admin)

2. Halaman Beranda (Admin)



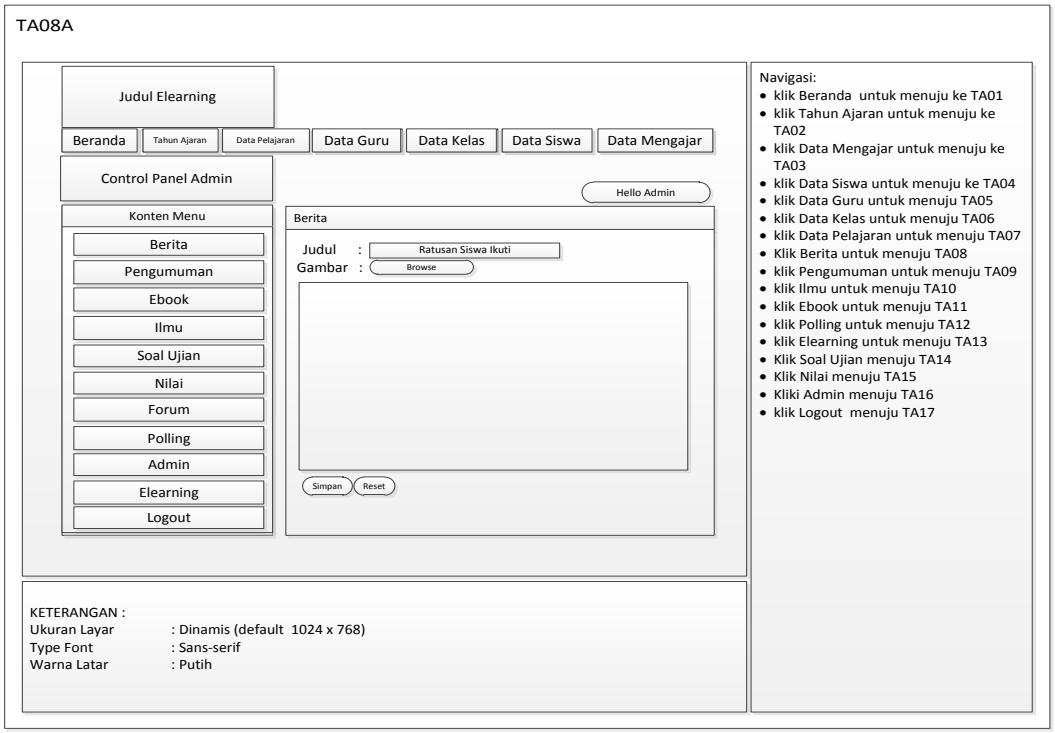
Gambar 3. 123 Tampilan Halaman Beranda (Admin)

3. Halaman Berita (Admin)



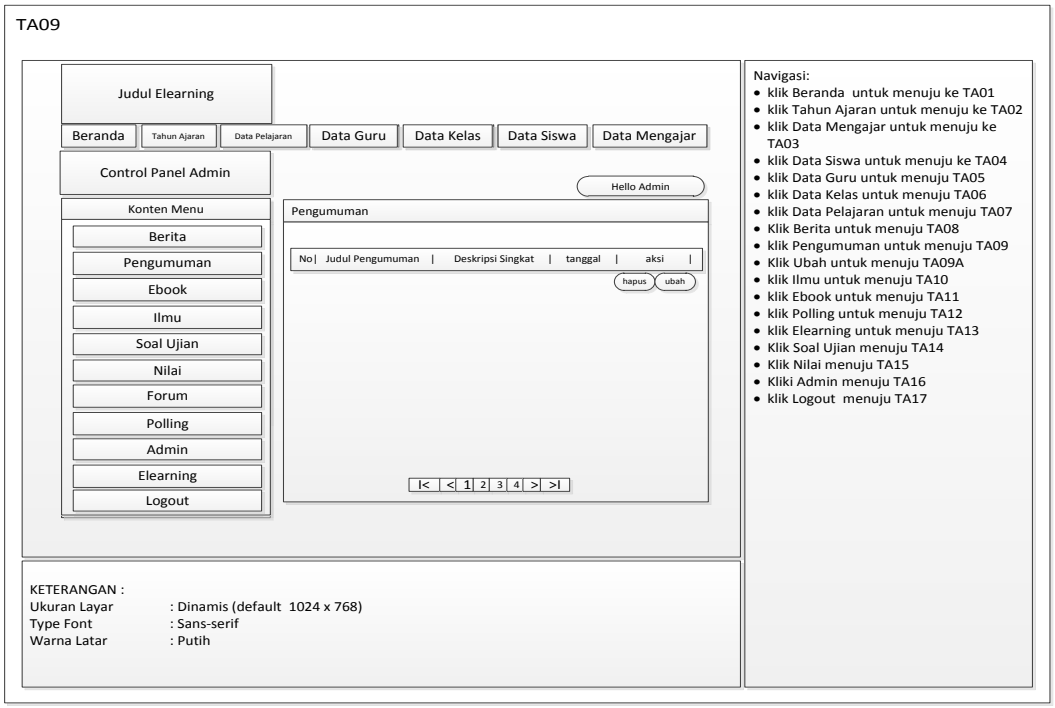
Gambar 3. 124 Tampilan Halaman Berita (Admin)

4. Halaman Ubah Berita (Admin)



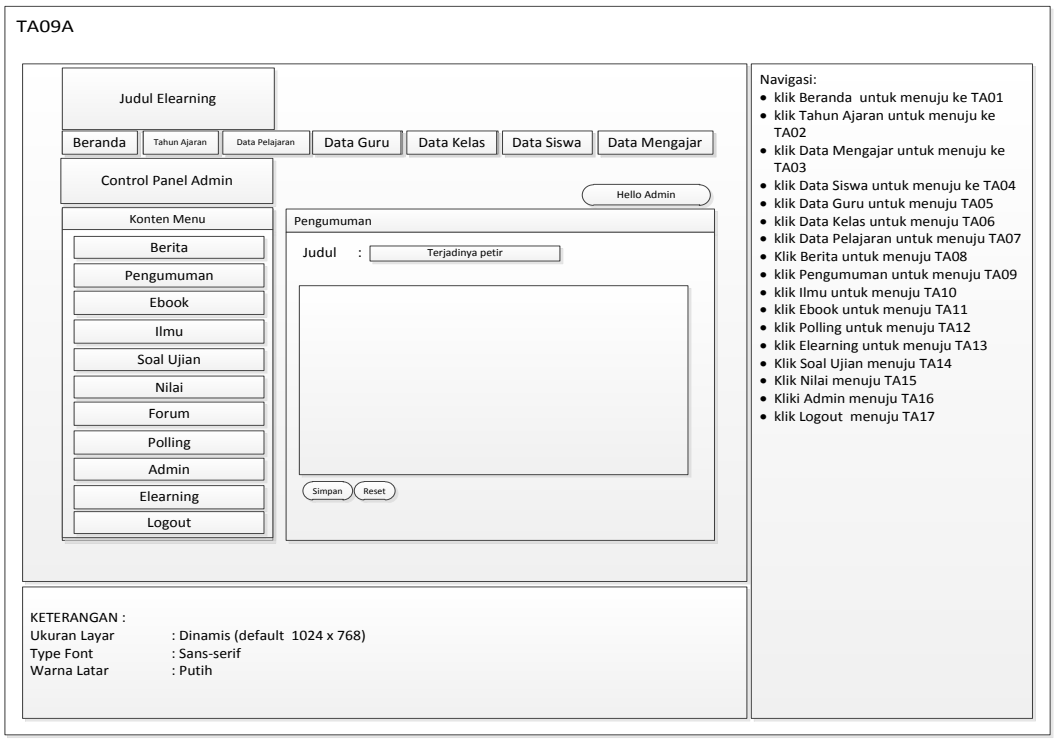
Gambar 3. 125 Tampilan Halaman Ubah Berita (Admin)

5. Halaman Pengumuman (Admin)



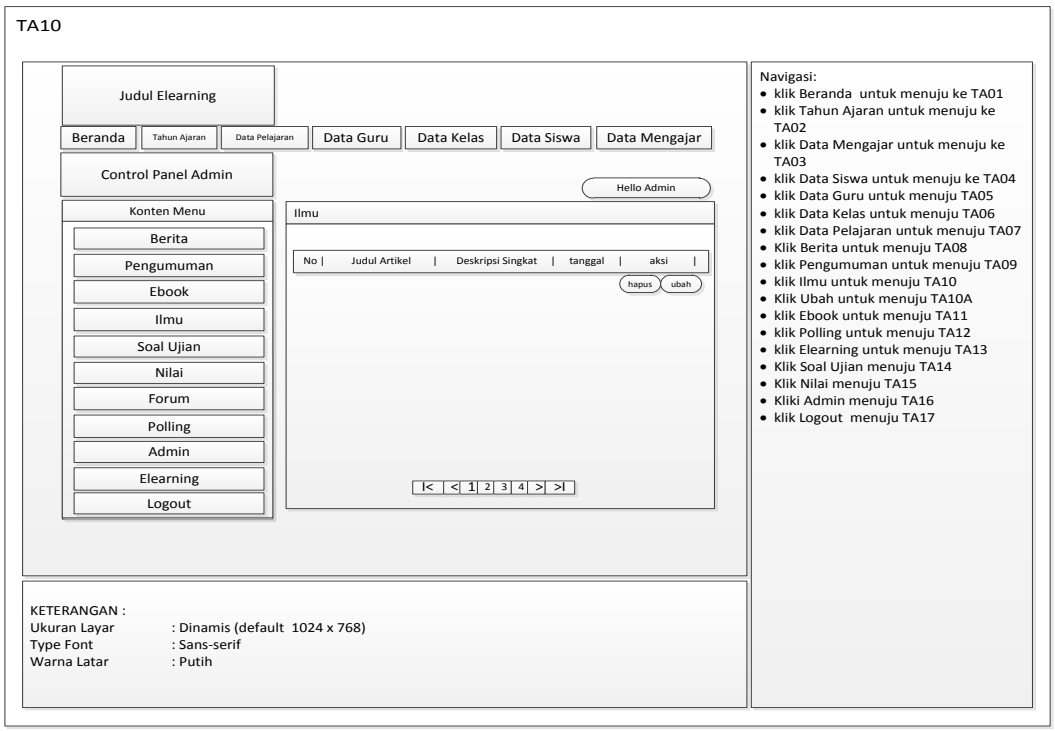
Gambar 3. 126 Tampilan Halaman Pengumuman (Admin)

6. Halaman Ubah Pengumuman (Admin)



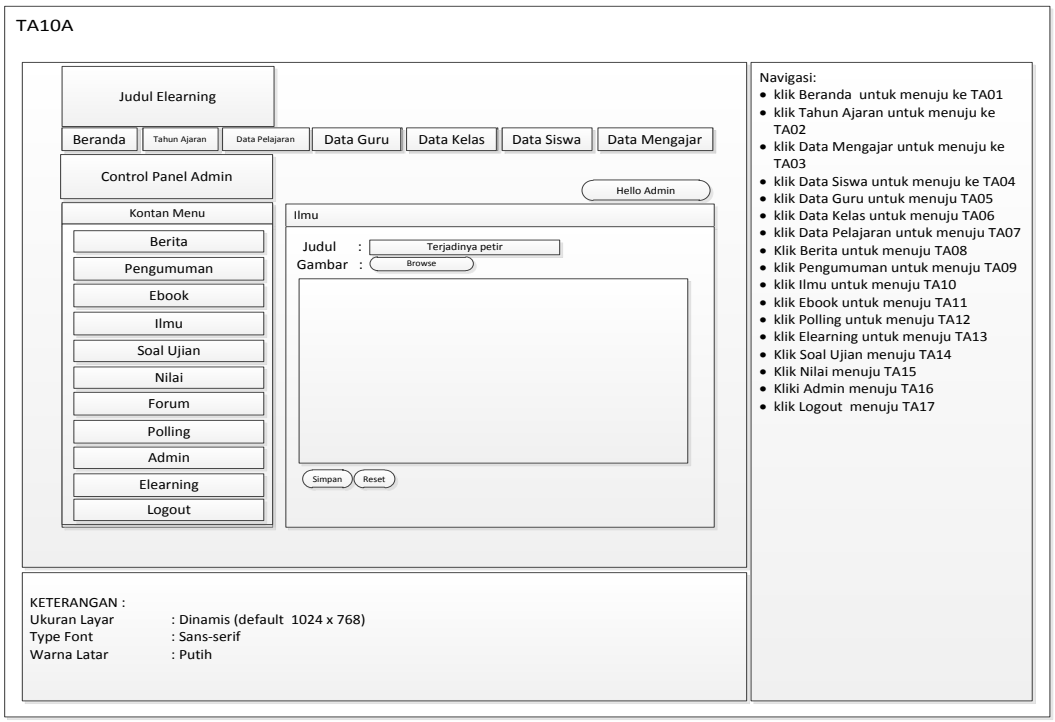
Gambar 3. 127 Tampilan Halaman Ubah Pengumuman (Admin)

7. Halaman Ilmu (Admin)



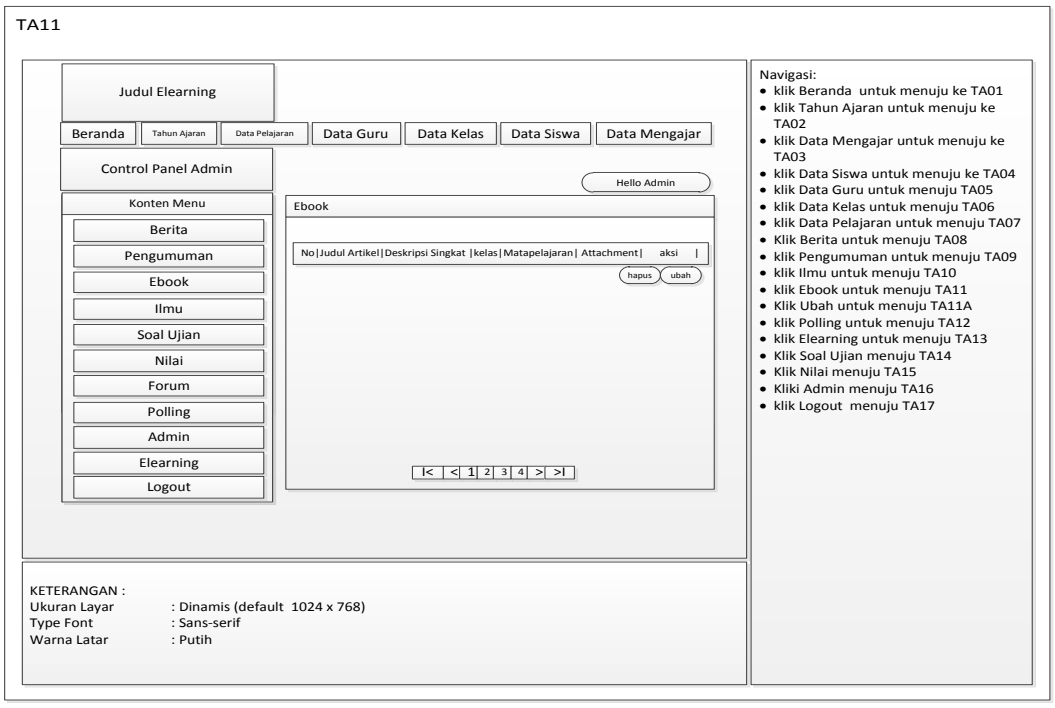
Gambar 3. 128 Tampilan Halaman Ilmu (Admin)

8. Halaman Ubah Ilmu (Admin)



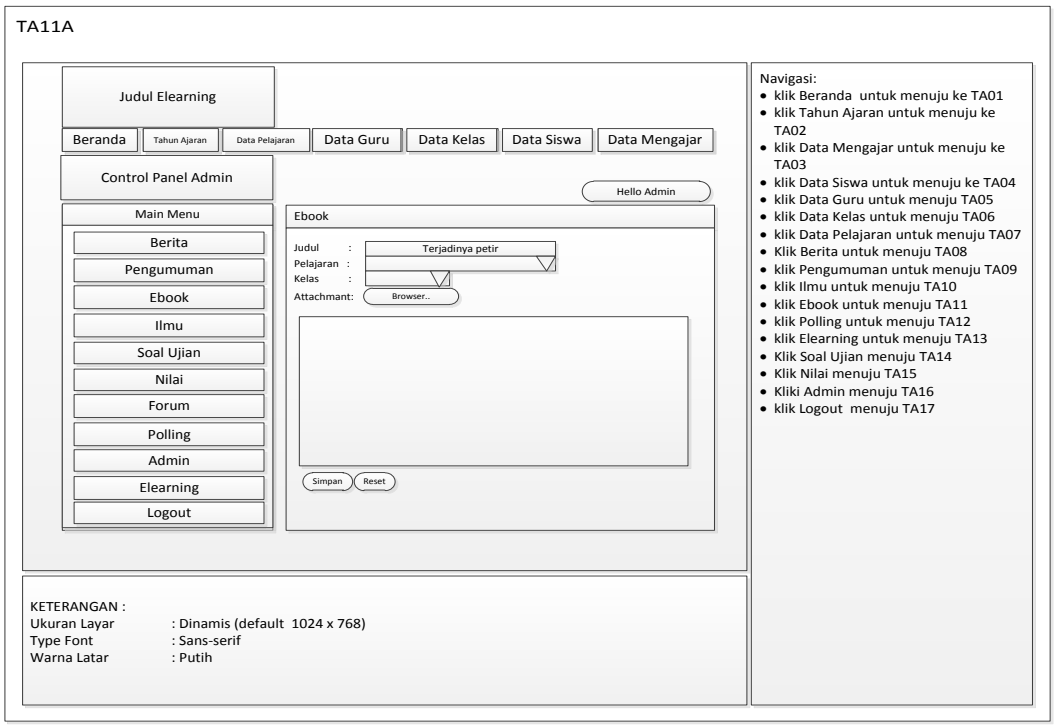
Gambar 3. 129 Tampilan Halaman Ubah Ilmu (Admin)

9. Halaman Ebook (Admin)



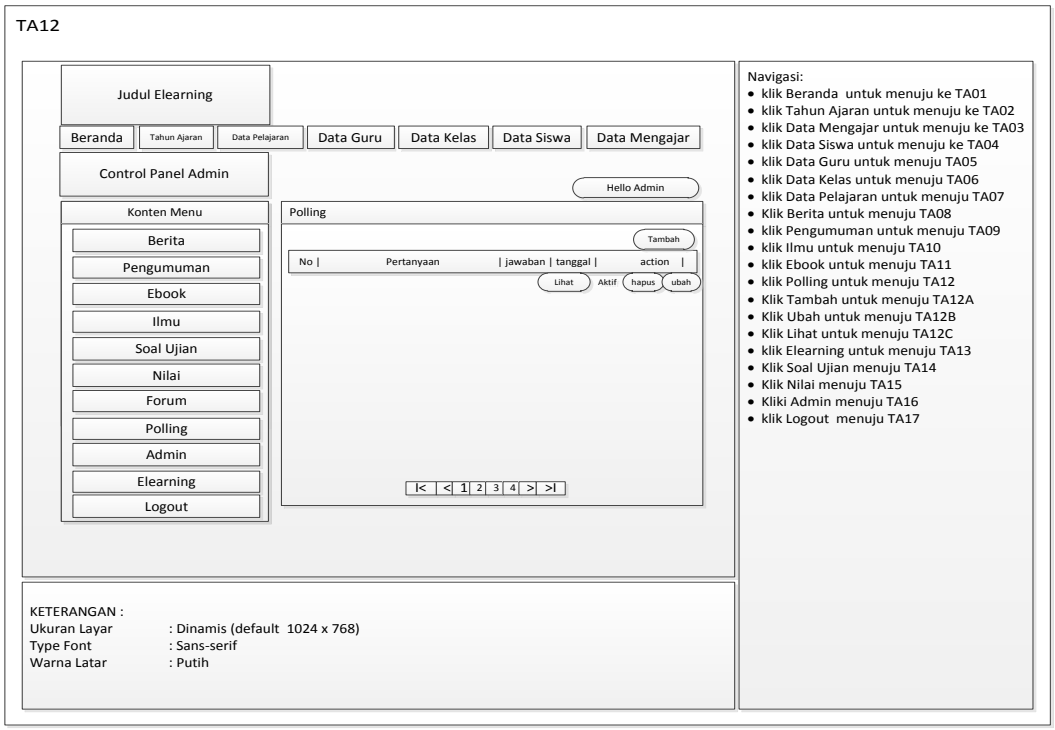
Gambar 3. 130 Tampilan Halaman Ebook (Admin)

10. Halaman Ubah Ebook (Admin)



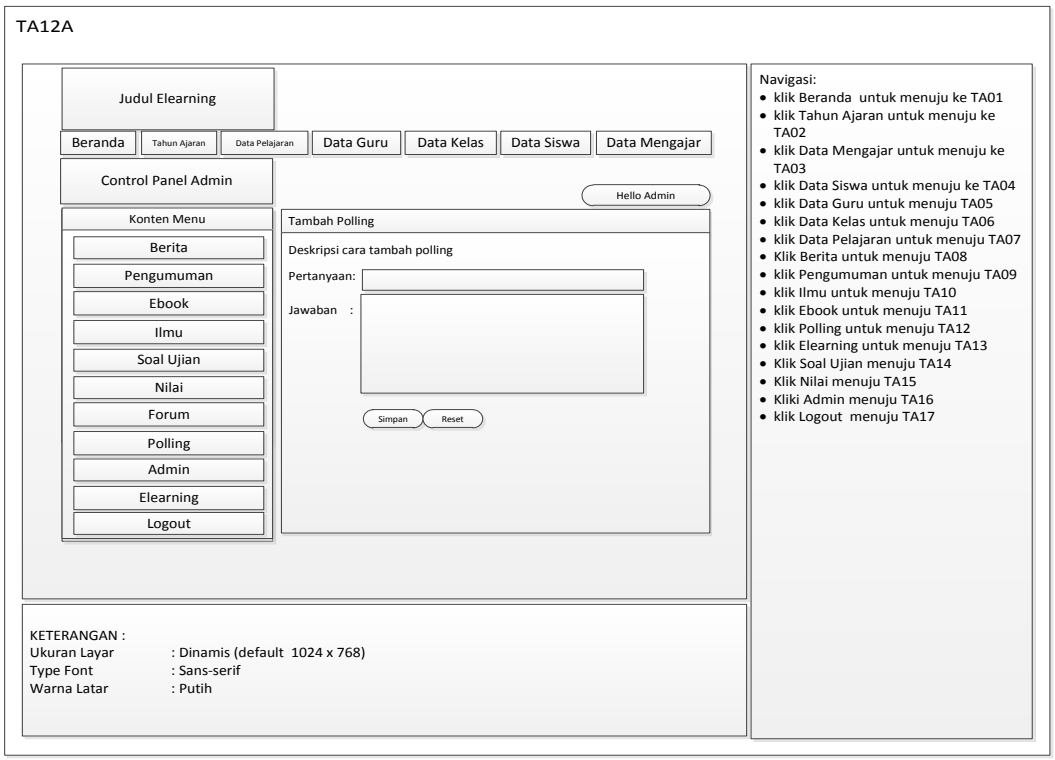
Gambar 3. 131 Tampilan Halaman Ubah Ebook (Admin)

11. Halaman Polling (Admin)



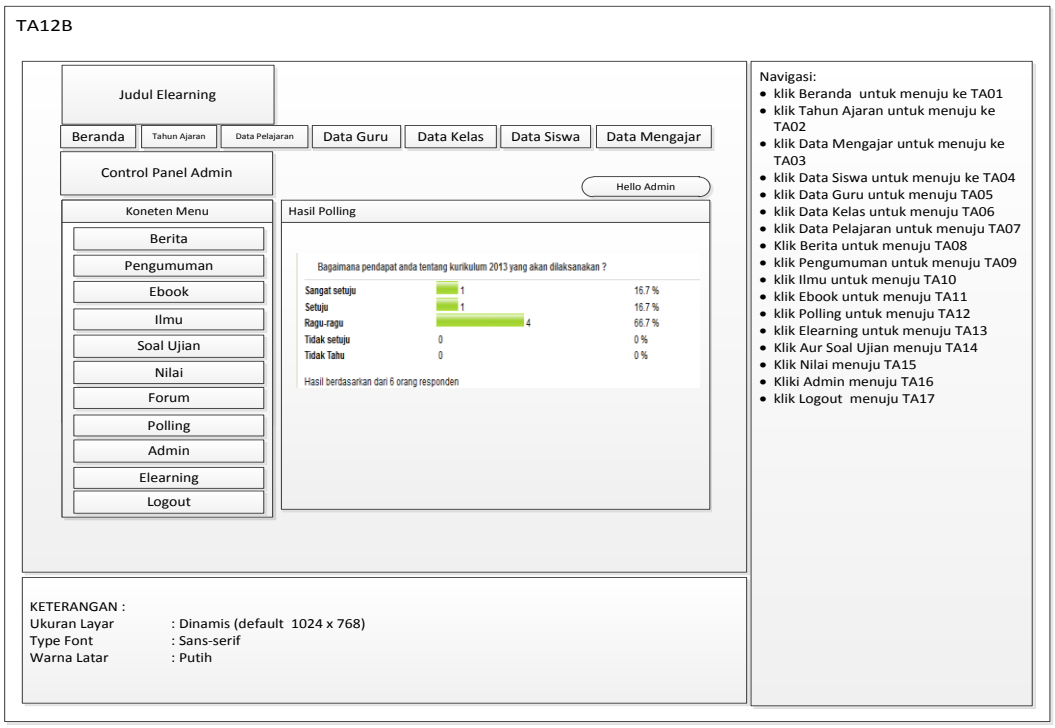
Gambar 3. 132 Tampilan Halaman Polling (Admin)

12. Halaman Tambah Polling (Admin)



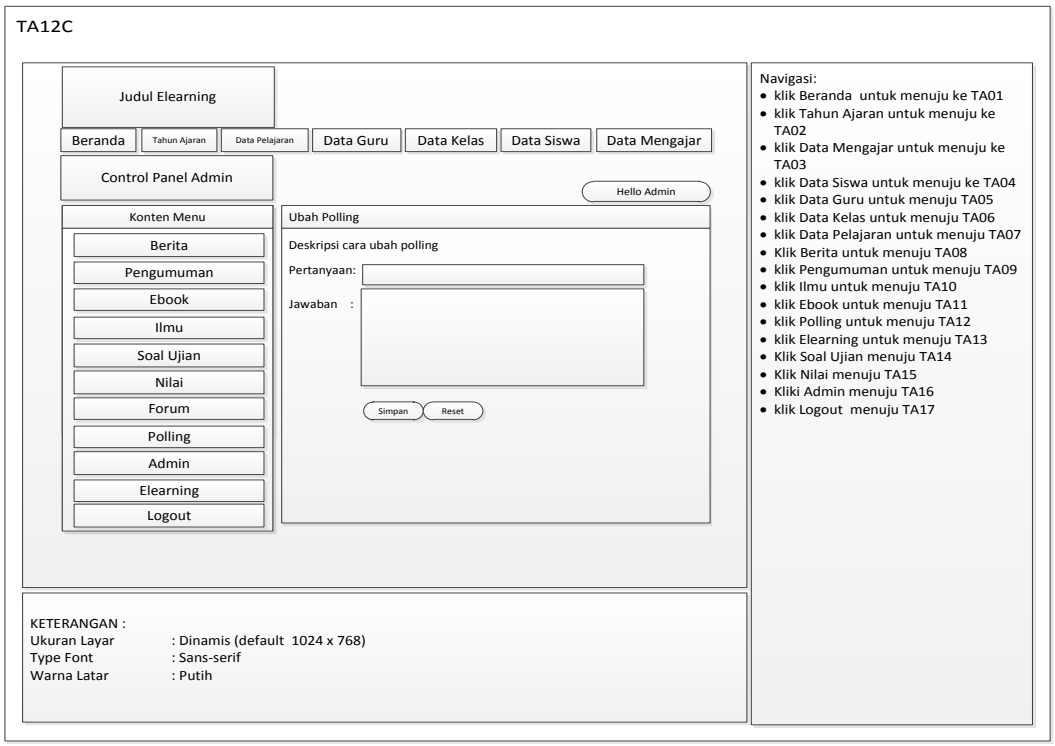
Gambar 3. 133 Tampilan Halaman Tambah Polling (Admin)

13. Halaman Lihat Hasil Polling (Admin)



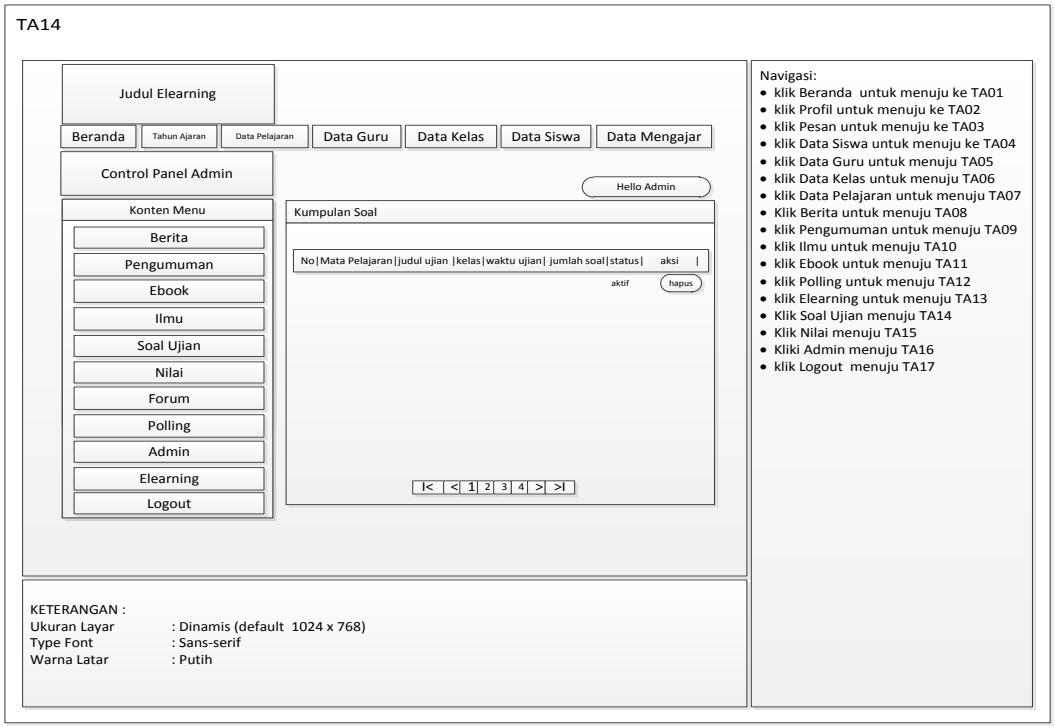
Gambar 3. 134 Tampilan Halaman Lihat Hasil Polling (Admin)

14. Halaman Ubah Polling (Admin)



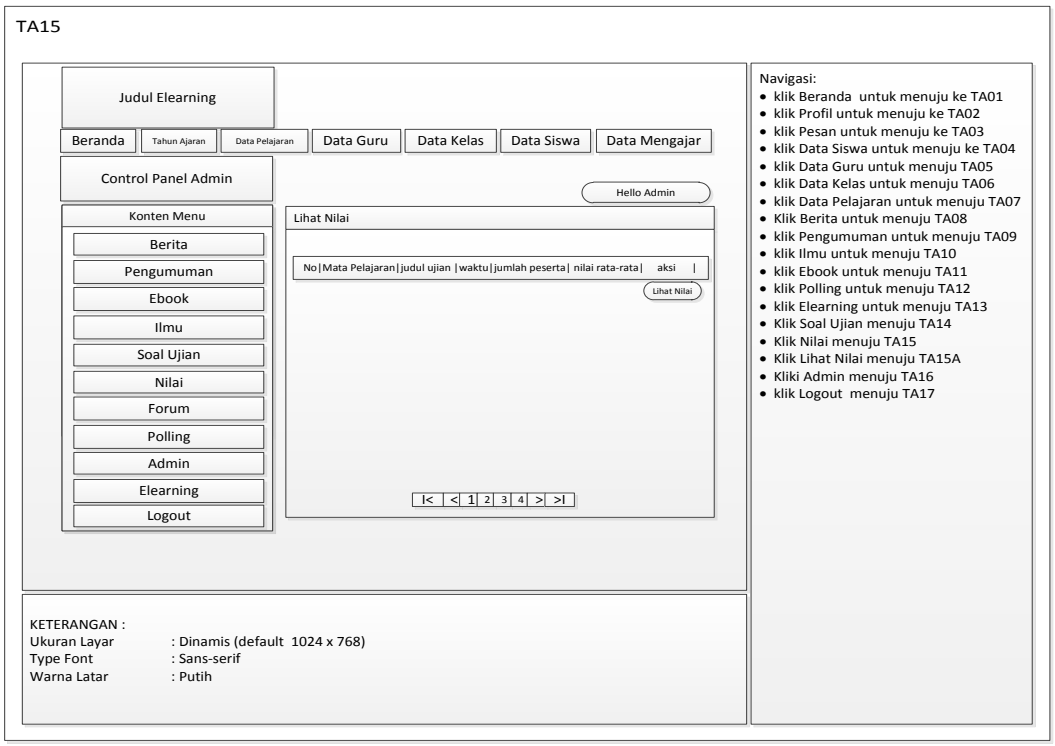
Gambar 3. 135 Tampilan Halaman Ubah Polling (Admin)

15. Halaman Soal Ujian (Admin)



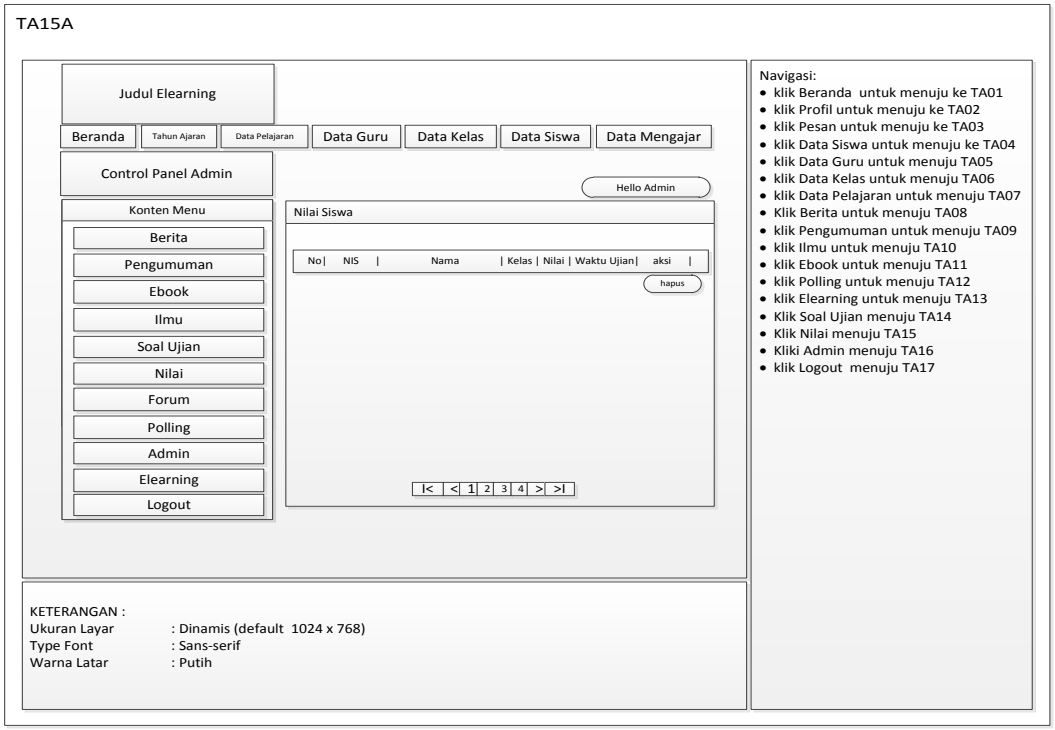
Gambar 3. 136 Tampilan Halaman Soal Ujian (Admin)

16. Halaman Nilai (Admin)



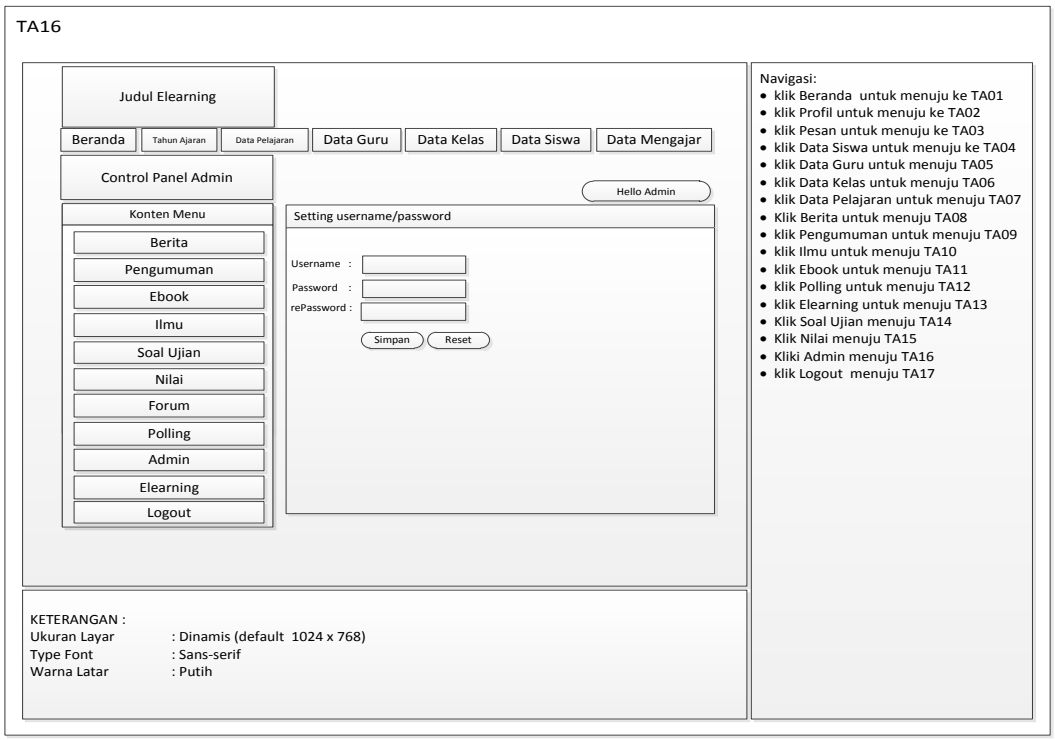
Gambar 3. 137 Tampilan Halaman Nilai (Admin)

17. Halaman Lihat Nilai (Admin)



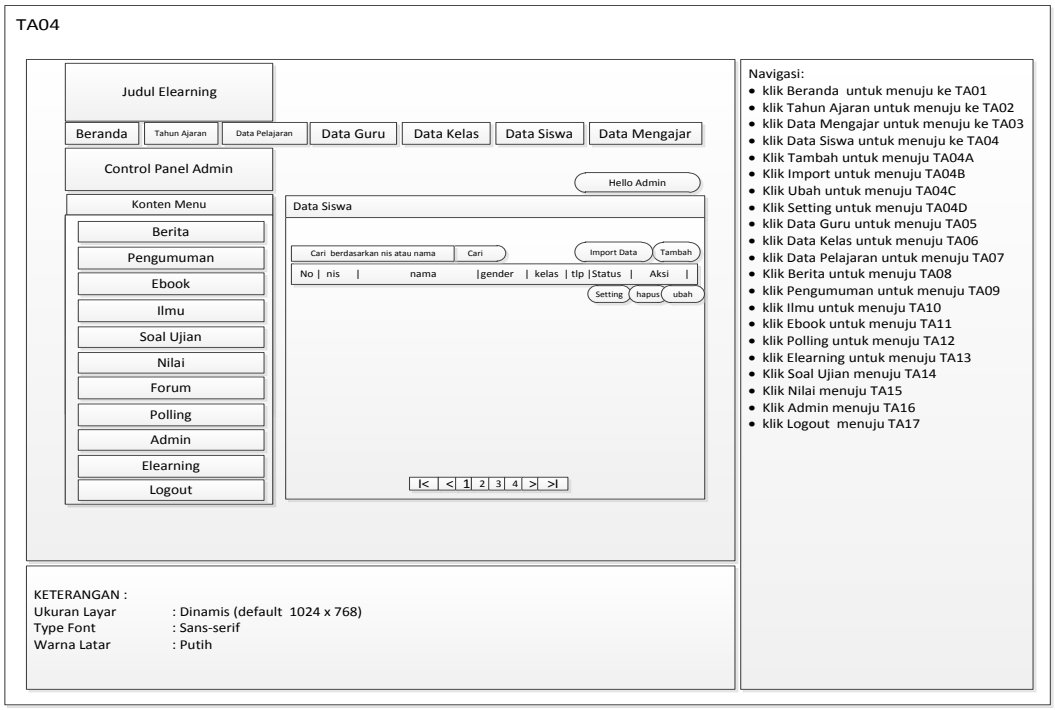
Gambar 3. 138 Tampilan Halaman Lihat Nilai (Admin)

18. Halaman Menu Admin (Admin)



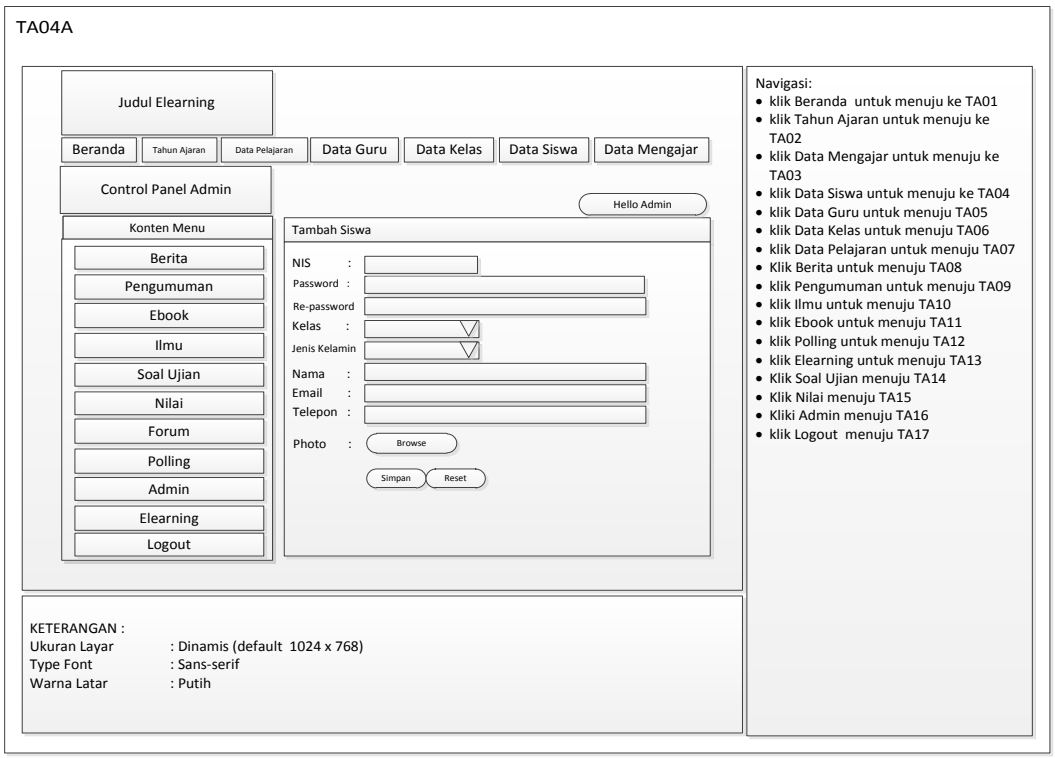
Gambar 3. 139 Tampilan Halaman Menu Admin (Admin)

19. Halaman Data Siswa (Admin)



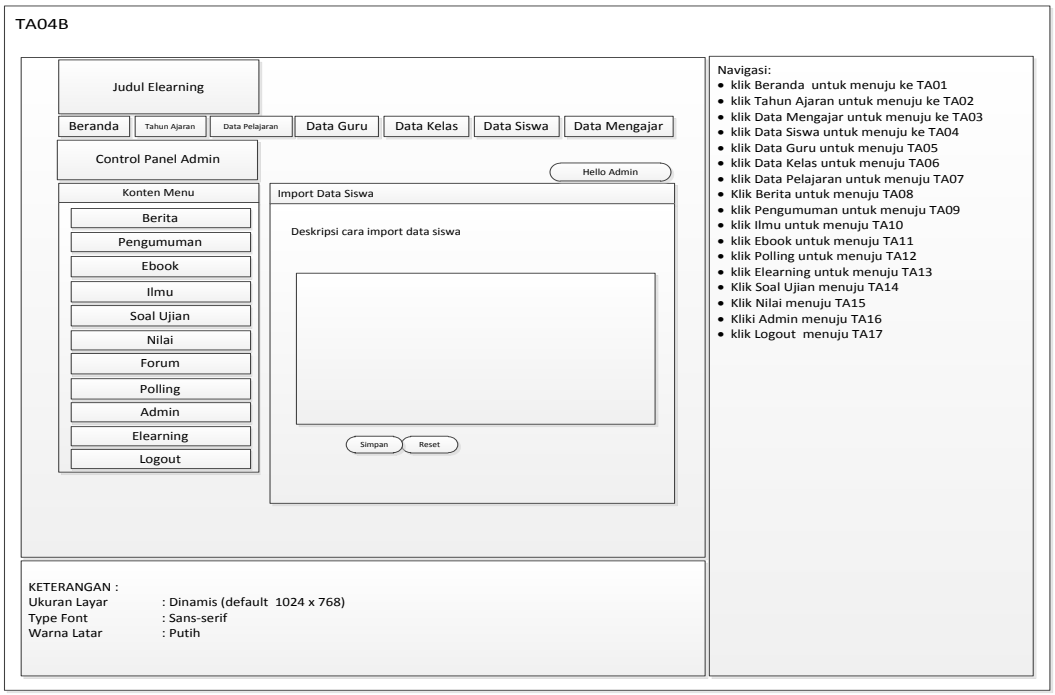
Gambar 3. 140 Tampilan Halaman Data Siswa (Admin)

20. Halaman Tambah Siswa (Admin)



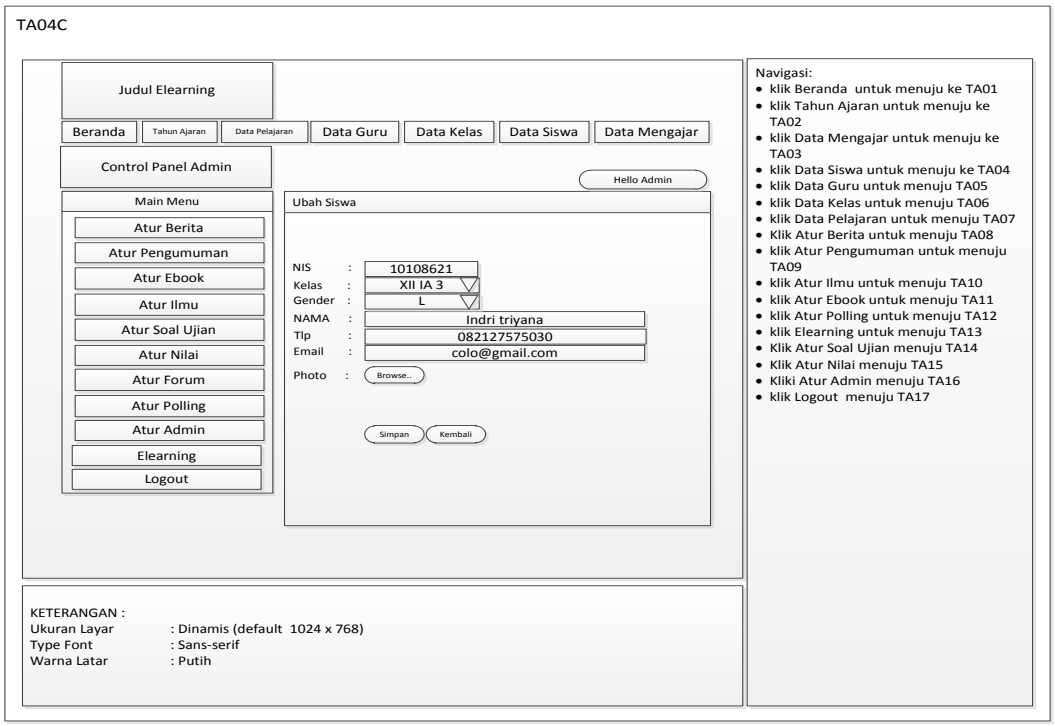
Gambar 3. 141 Tampilan Halaman Tambah Siswa (Admin)

21. Halaman Import Siswa (Admin)



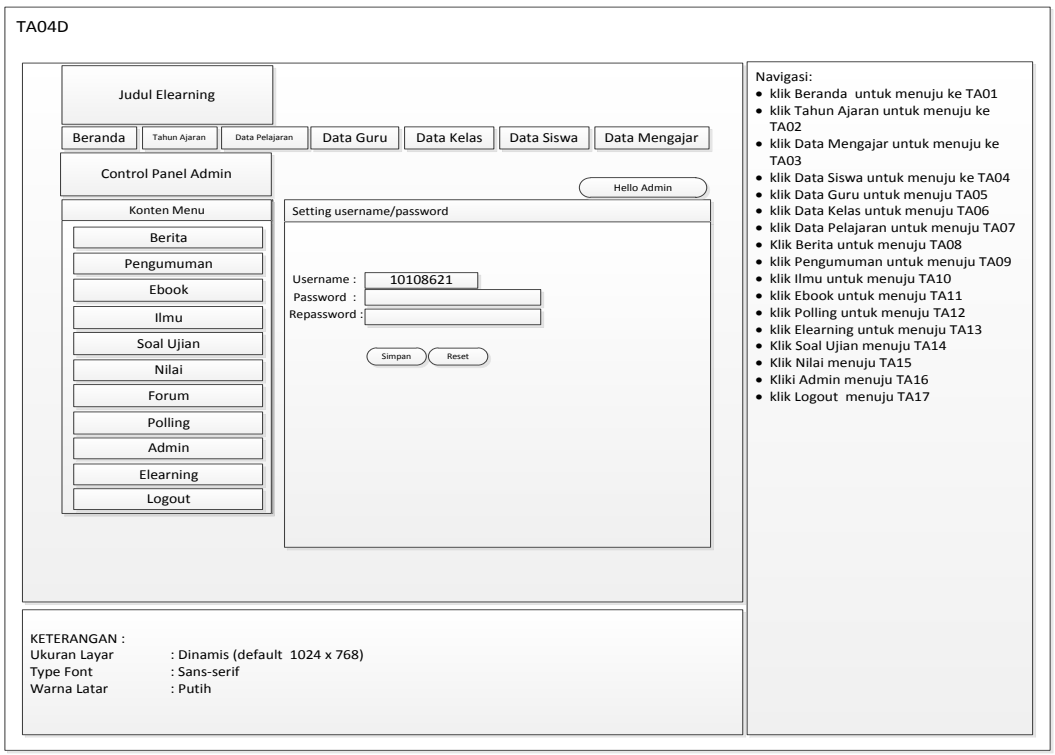
Gambar 3. 142 Tampilan Halaman Import Siswa (Admin)

22. Halaman Ubah Siswa (Admin)



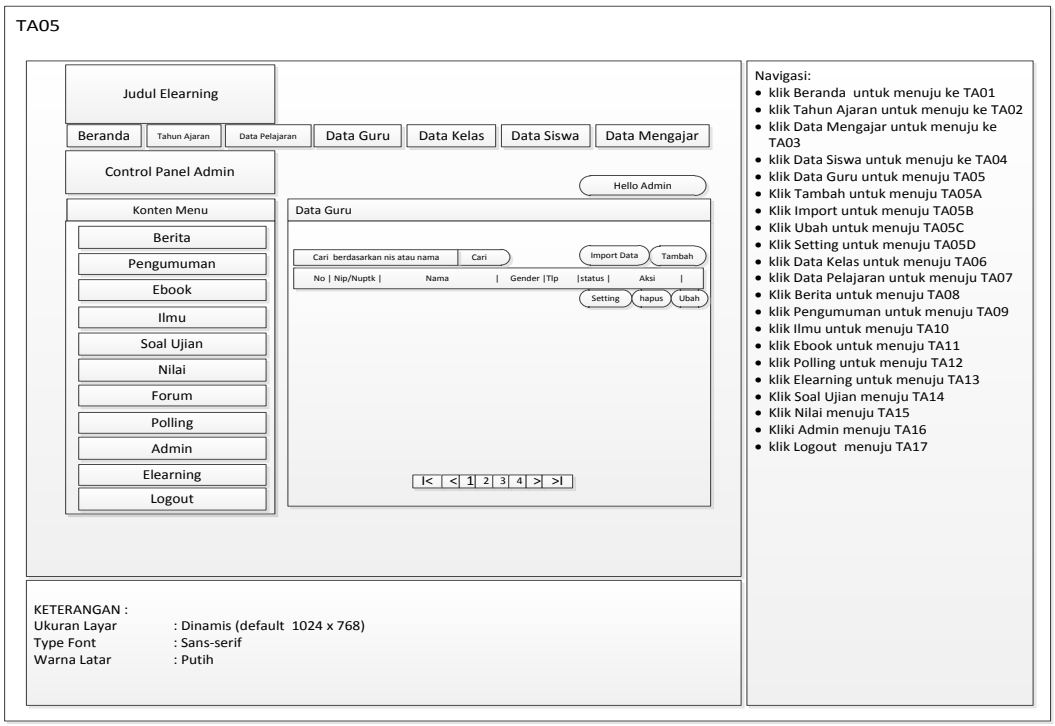
Gambar 3. 143 Tampilan Halaman Ubah Siswa (Admin)

23. Halaman Setting Siswa (Admin)



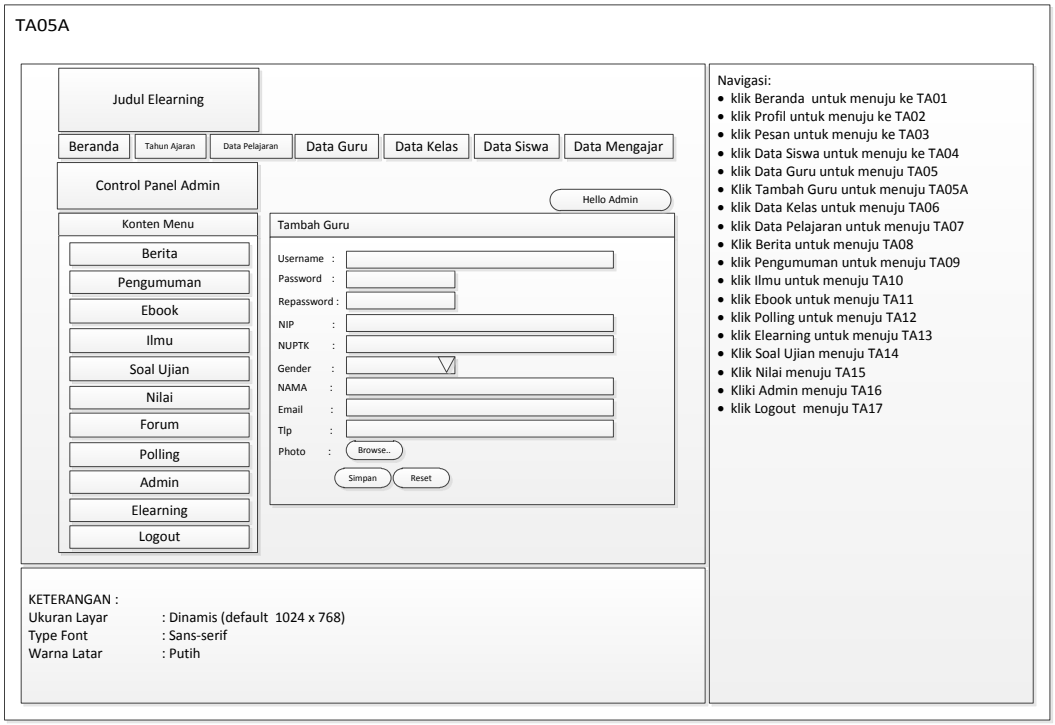
Gambar 3. 144 Tampilan Halaman Setting Siswa (Admin)

24. Halaman Data Guru (Admin)



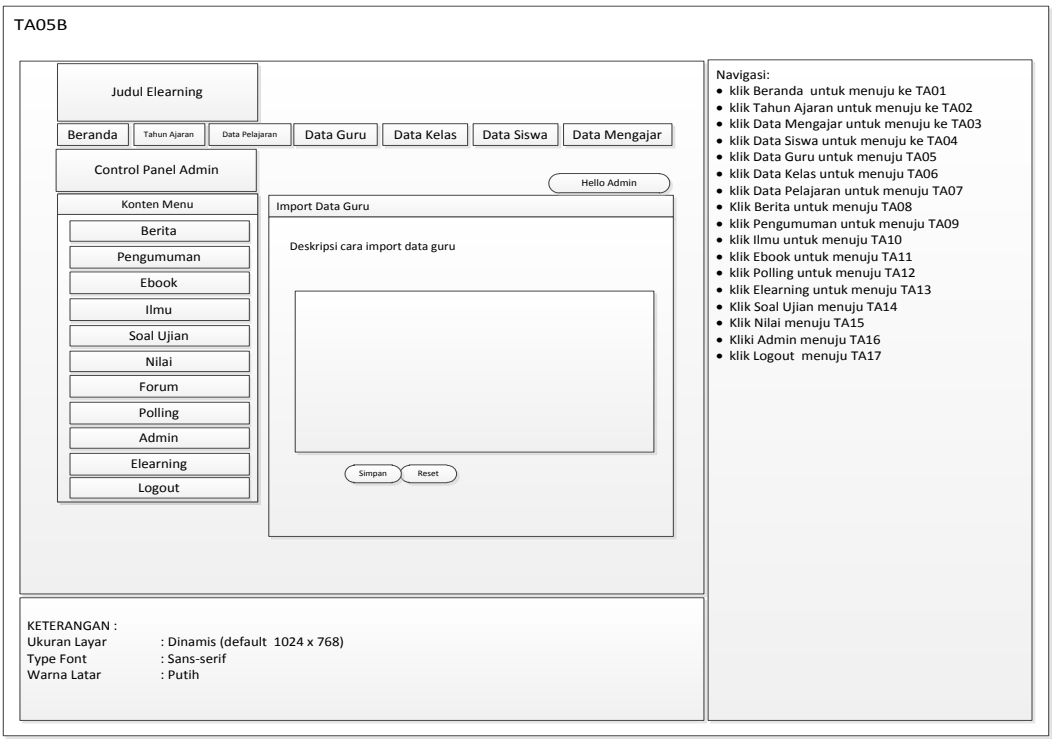
Gambar 3. 145 Tampilan Halaman Data Guru (Admin)

25. Halaman Tambah Guru (Admin)



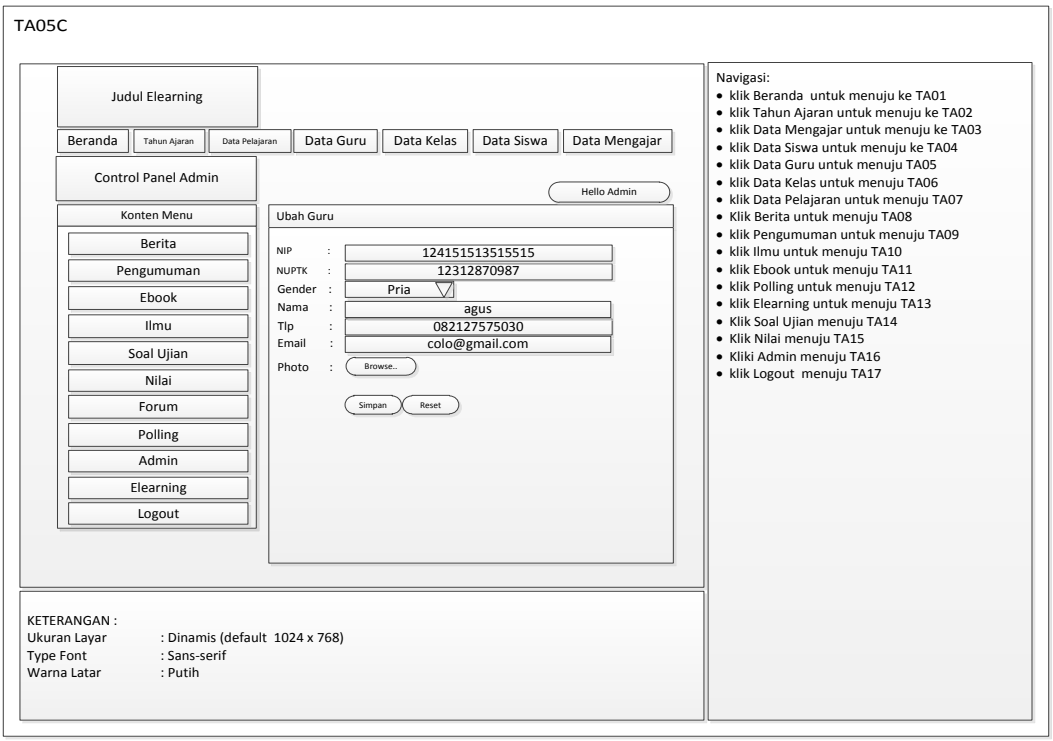
Gambar 3. 146 Tampilan Halaman Tambah Guru (Admin)

26. Halaman Import Guru (Admin)



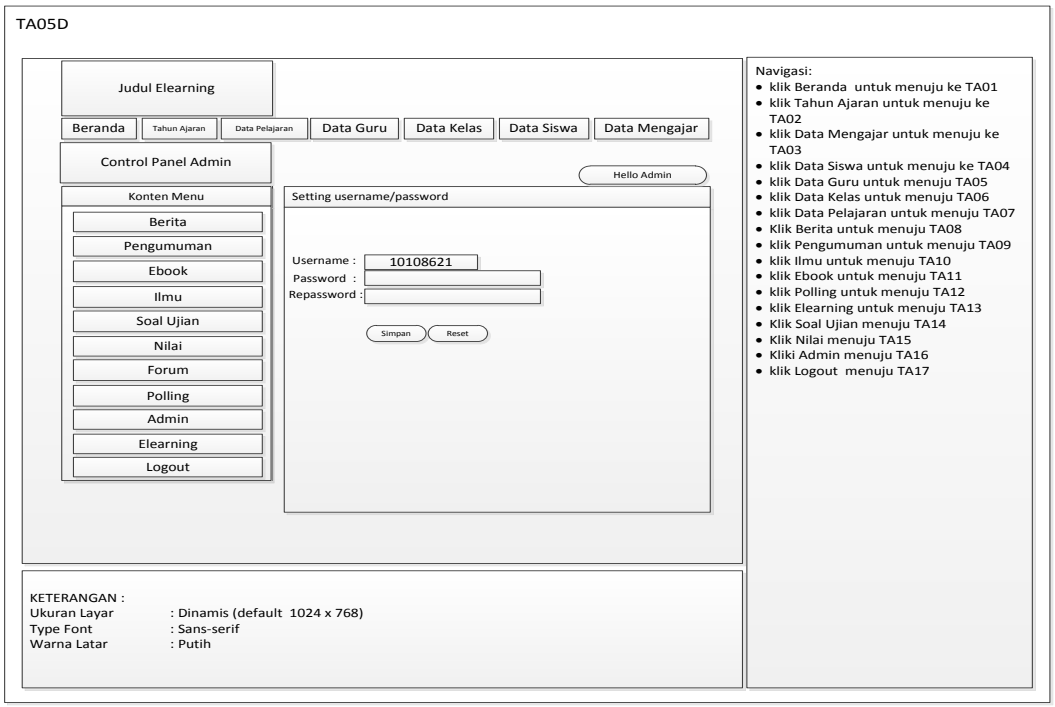
Gambar 3. 147 Tampilan Halaman Import Guru (Admin)

27. Halaman Ubah Guru (Admin)



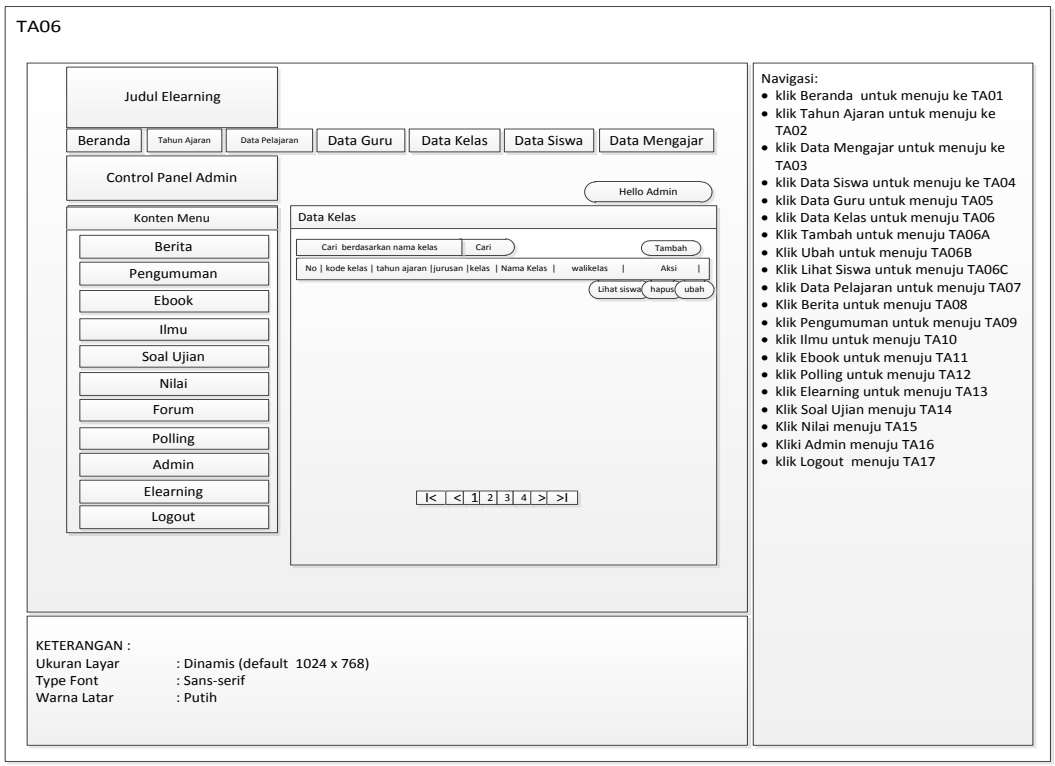
Gambar 3. 148 Tampilan Halaman Ubah Guru (Admin)

28. Halaman Setting Guru (Admin)



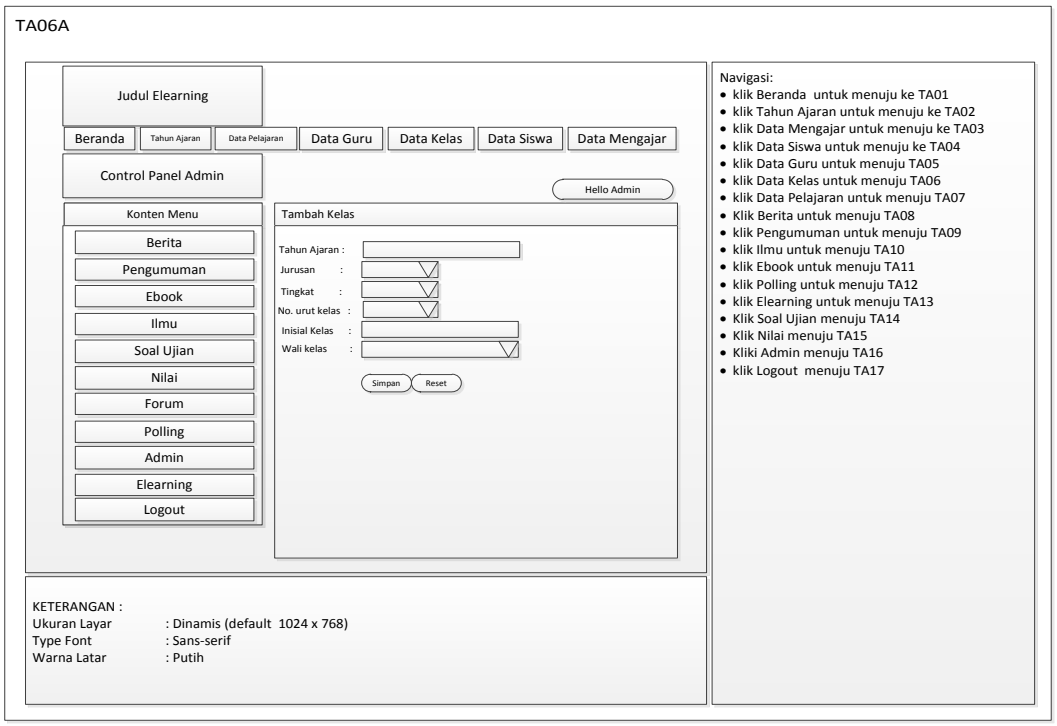
Gambar 3. 149 Tampilan Halaman Setting Guru (Admin)

29. Halaman Data Kelas (Admin)



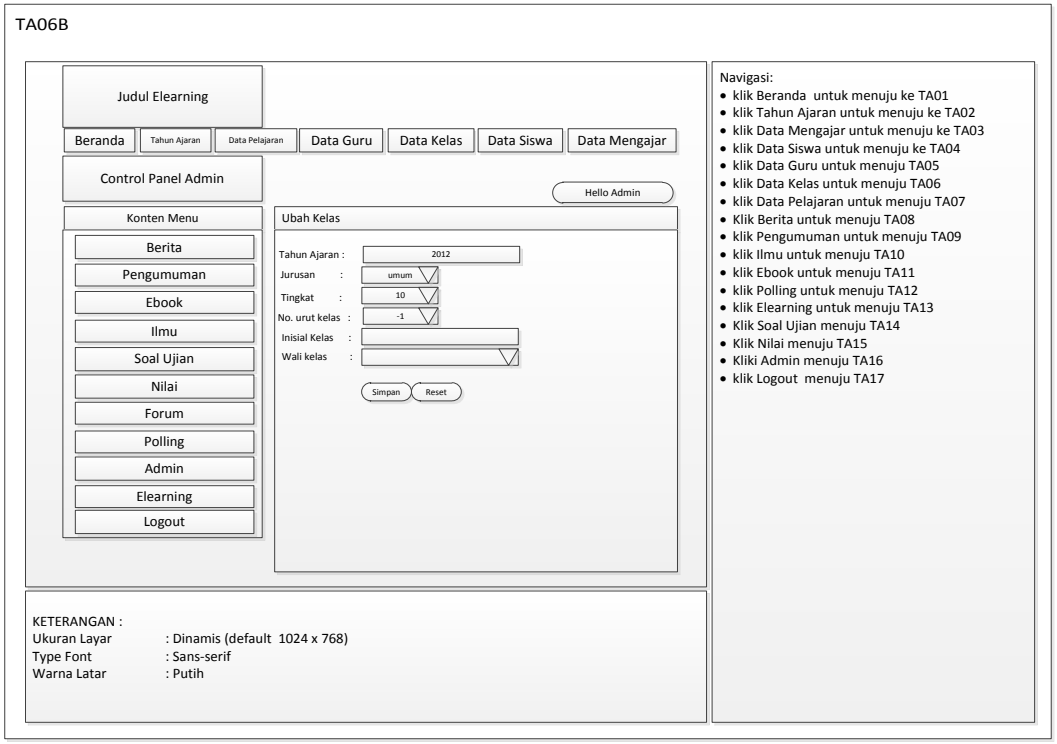
Gambar 3. 150 Tampilan Halaman Data Kelas (Admin)

30. Halaman Tambah Kelas (Admin)



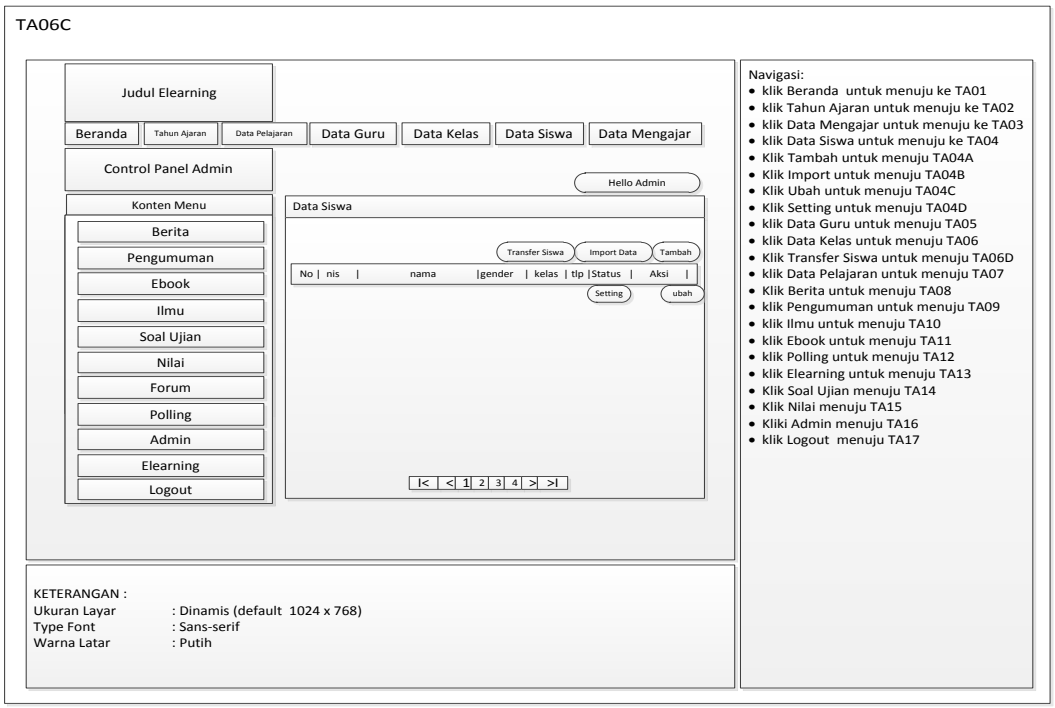
Gambar 3. 151 Tampilan Halaman Tambah Kelas (Admin)

31. Halaman Ubah Kelas (Admin)



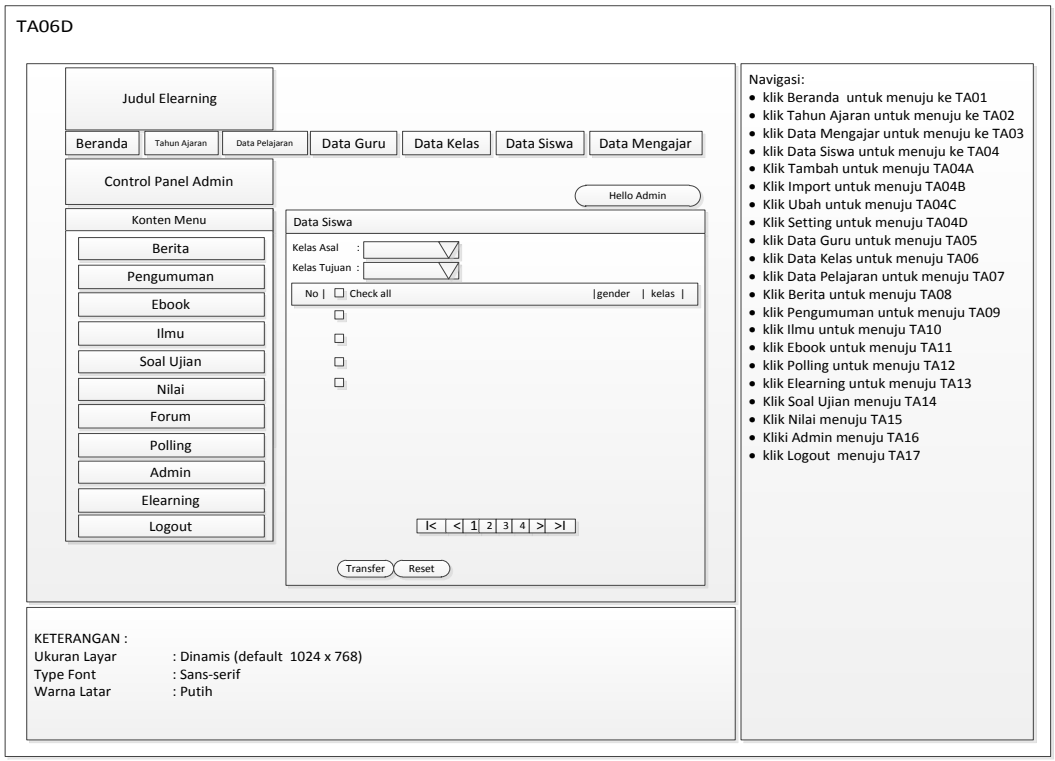
Gambar 3. 152 Tampilan Halaman Ubah Kelas (Admin)

32. Halaman Lihas Siswa Kelas (Admin)



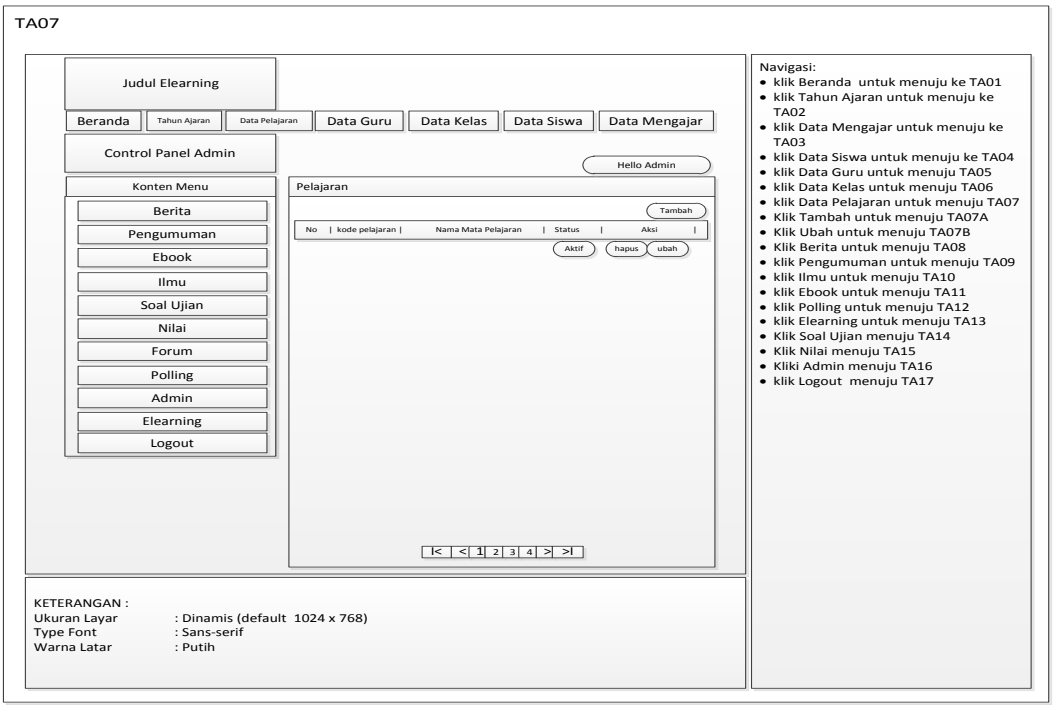
Gambar 3. 153 Tampilan Halaman Lihat Siswa Kelas (Admin)

33. Halaman Transfer Siswa (Admin)



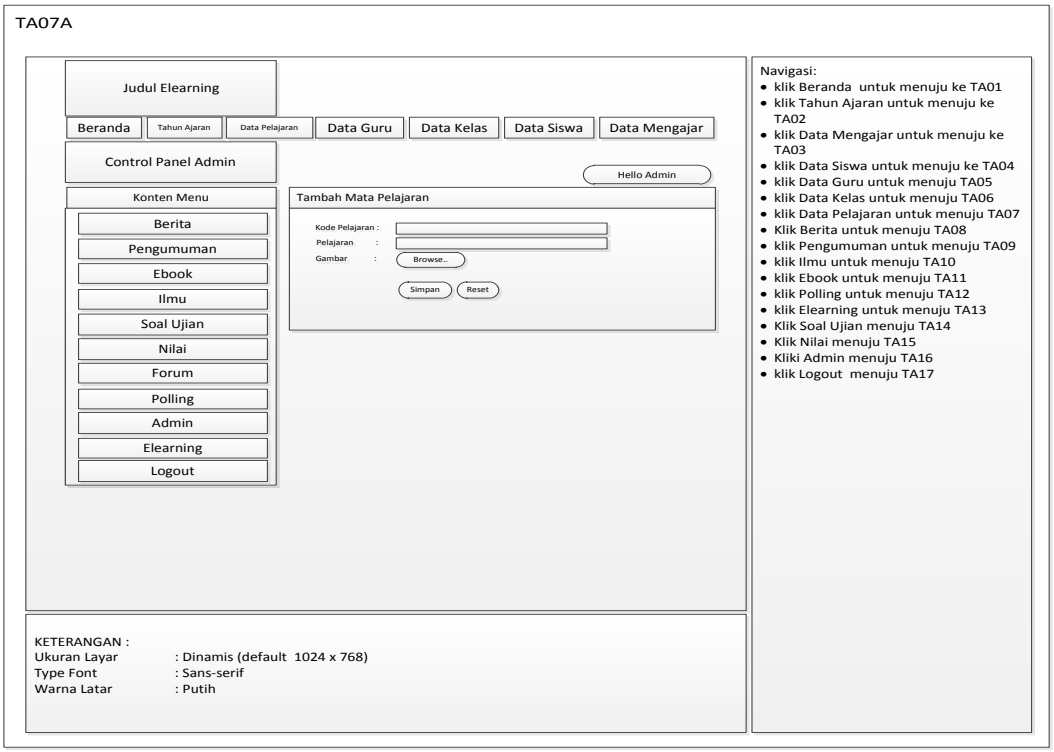
Gambar 3. 154 Tampilan Halaman Transfer Siswa (Admin)

34. Halaman Data Pelajaran (Admin)



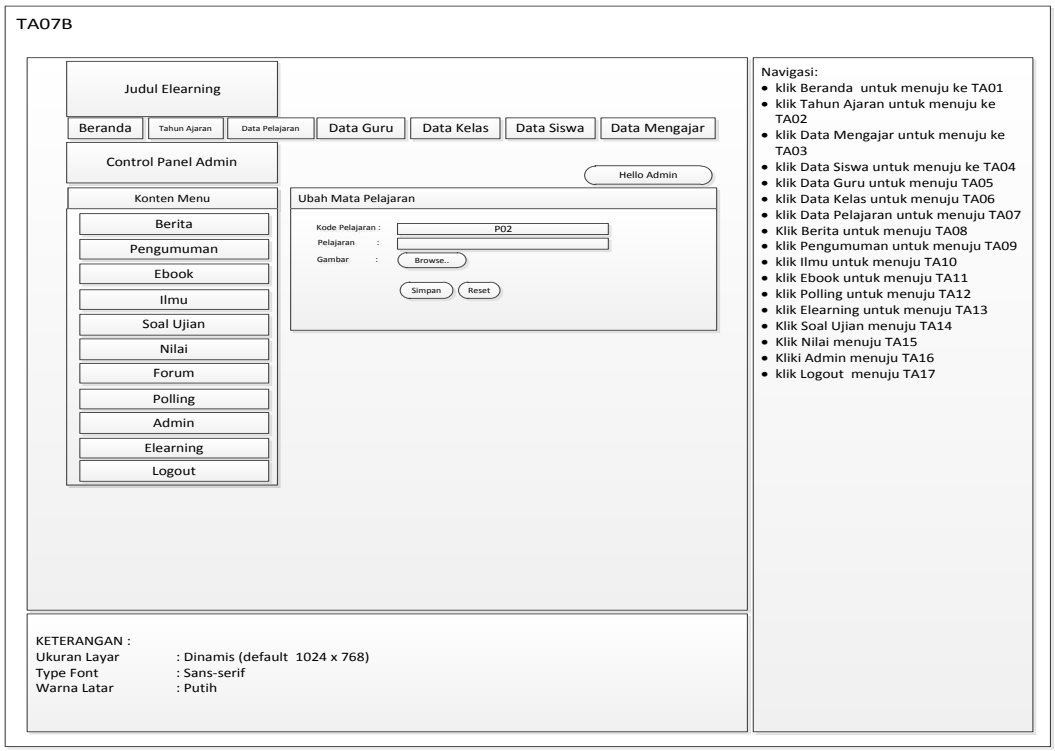
Gambar 3. 155 Tampilan Halaman Data Pelajaran (Admin)

35. Halaman Tambah Pelajaran (Admin)



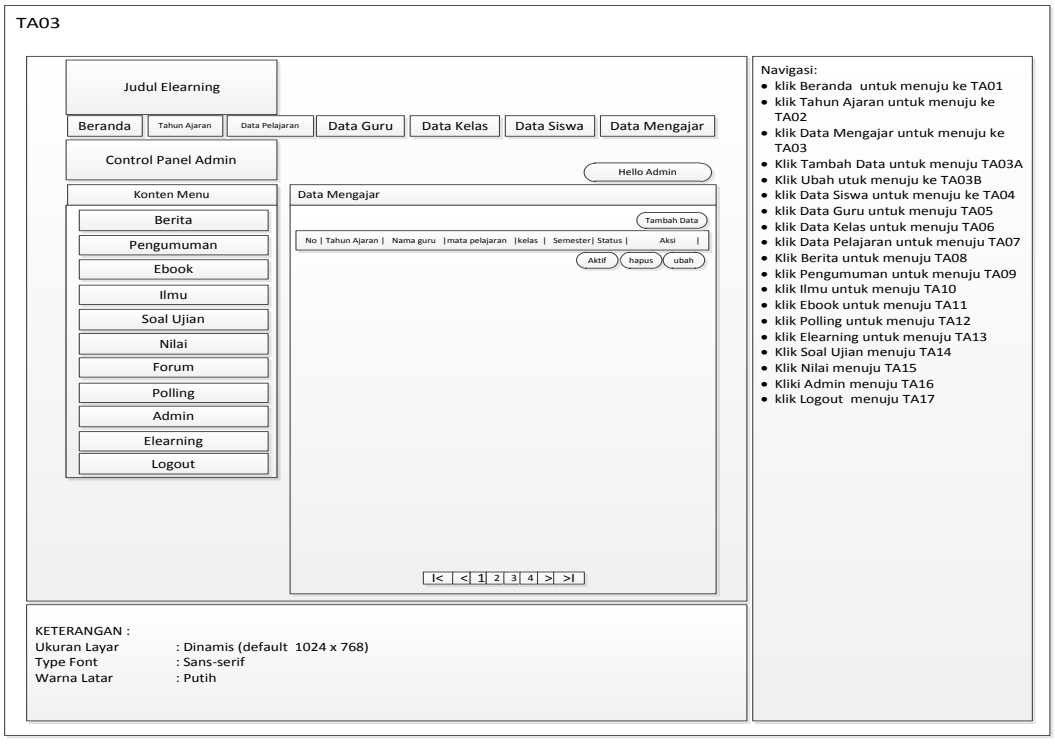
Gambar 3. 156 Tampilan Halaman Tambah Pelajaran (Admin)

36. Halaman Ubah Pelajaran (Admin)



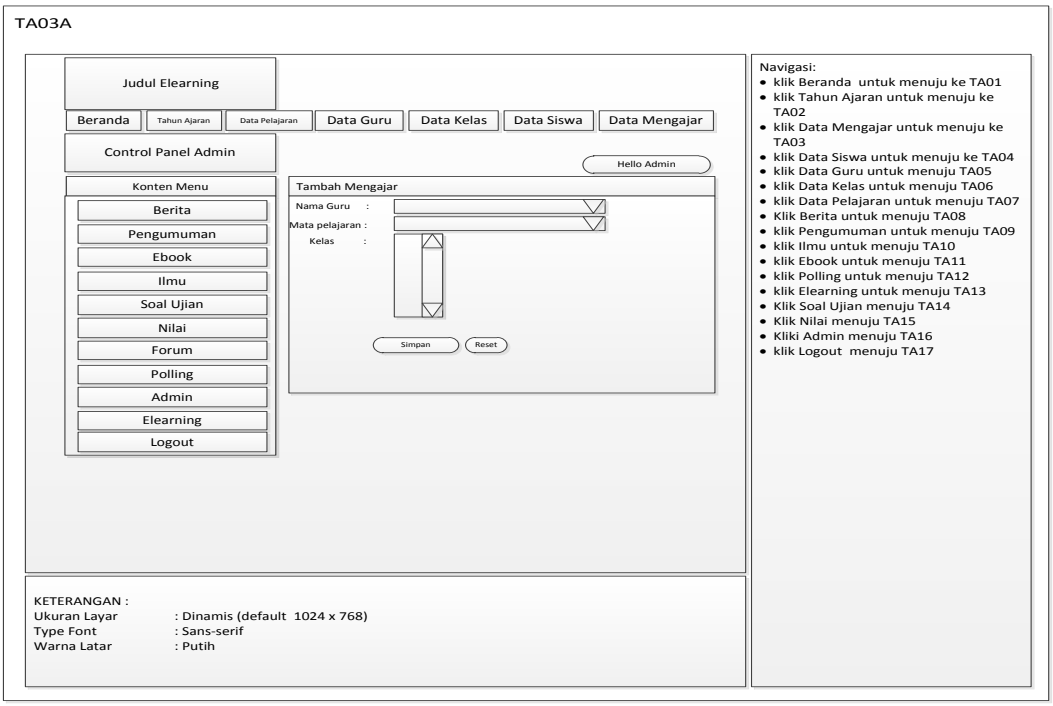
Gambar 3. 157 Tampilan Halaman Ubah Pelajaran (Admin)

37. Halaman Data Mengajar (Admin)



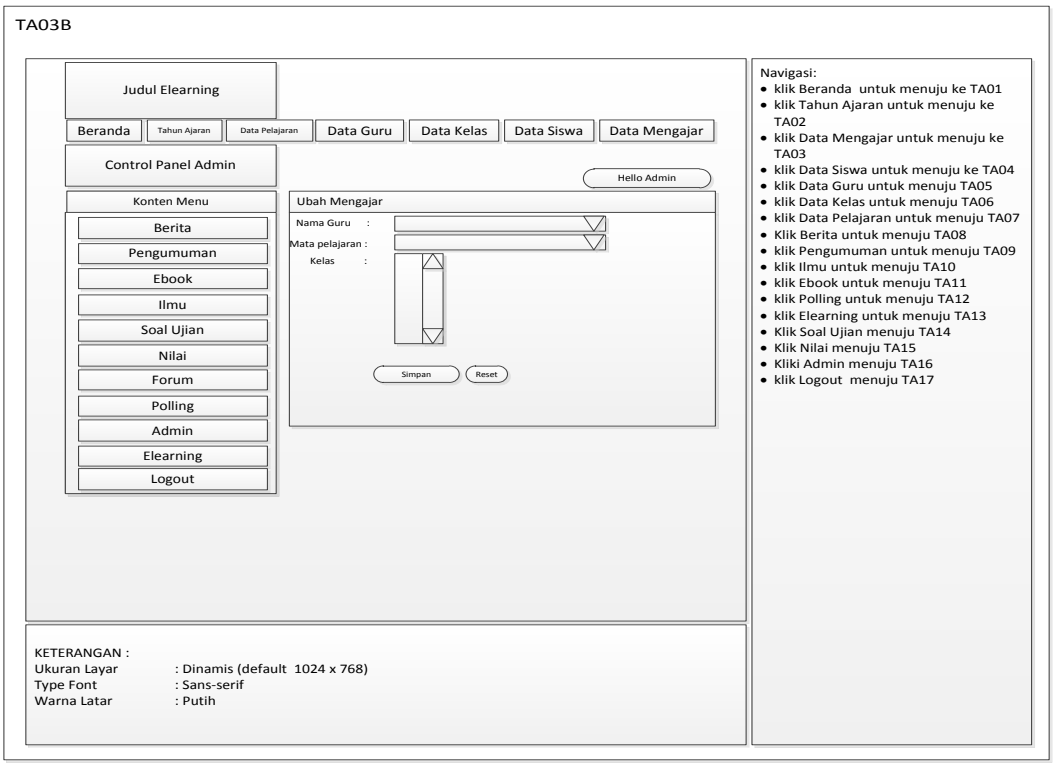
Gambar 3. 158 Tampilan Halaman Data Mengajar (Admin)

38. Halaman Tambah Mengajar (Admin)



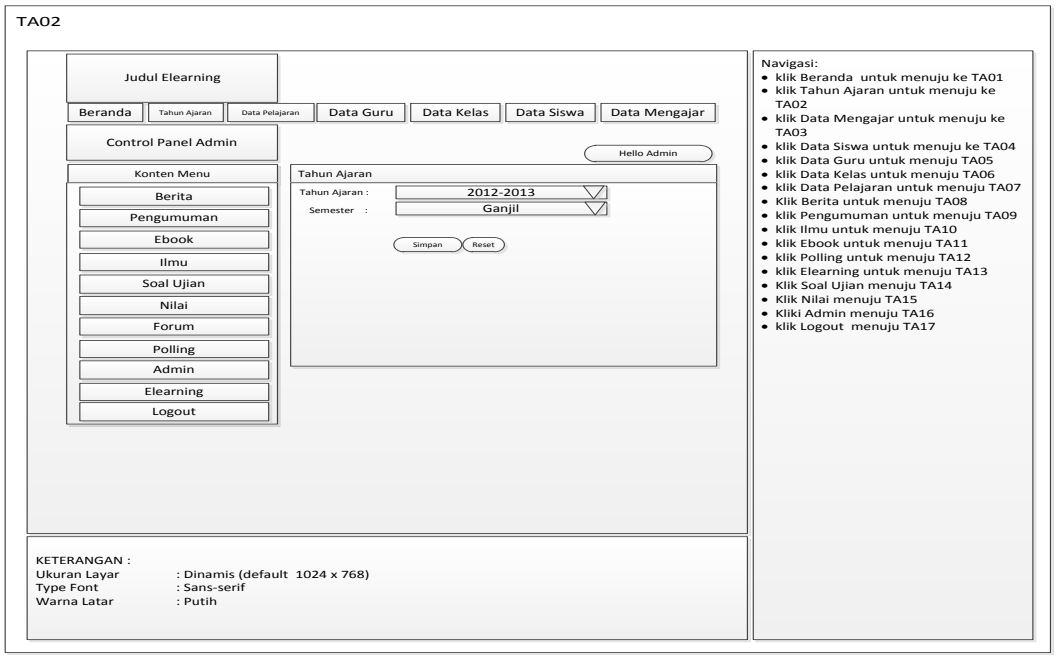
Gambar 3. 159 Tampilan Halaman Tambah Mengajar (Admin)

39. Halaman Ubah Mengajar (Admin)



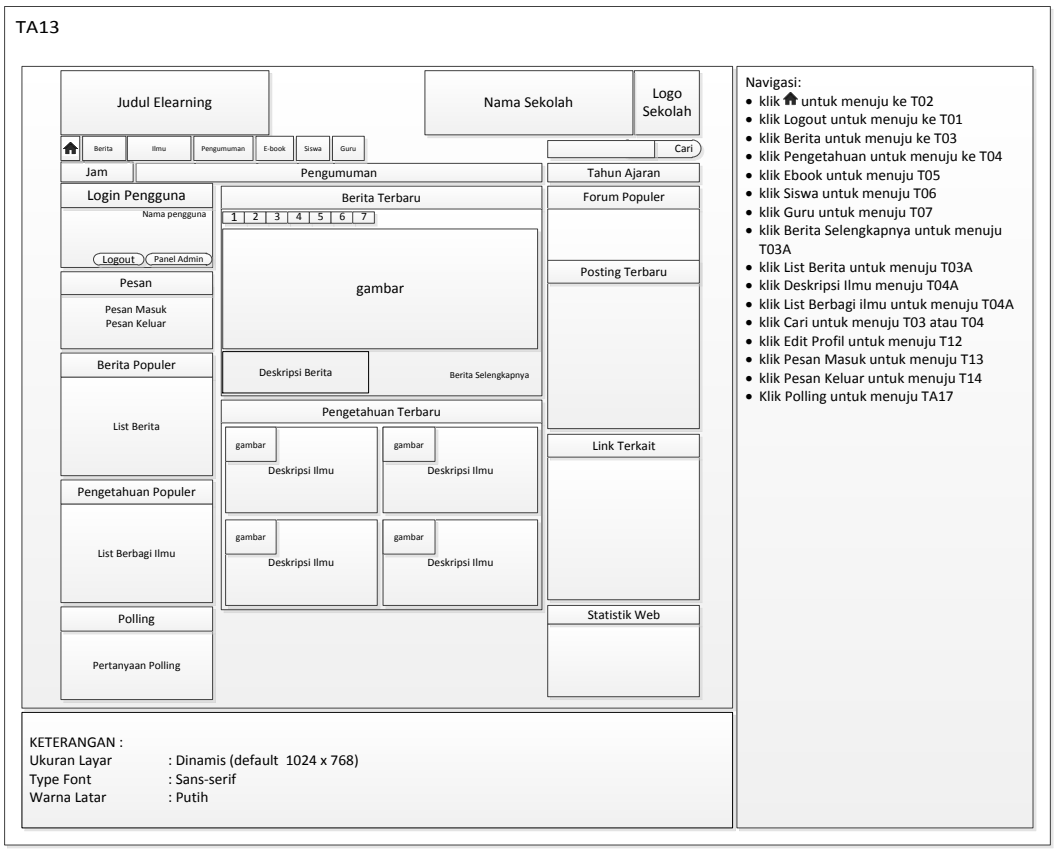
Gambar 3. 160 Tampilan Halaman Ubah Mengajar (Admin)

40. Halaman Tahun Ajaran (Admin)



Gambar 3. 161 Tampilan Halaman Tahun Ajaran (Admin)

41. Halaman Beranda Admin (sesudah login)

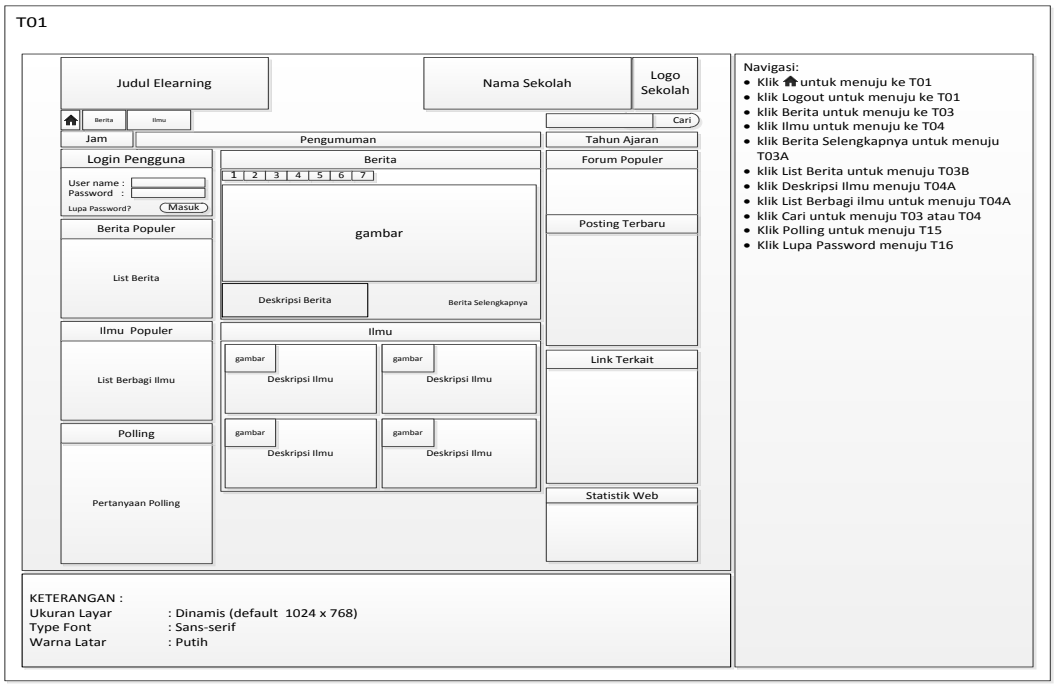


Gambar 3. 162 Tampilan Halaman Beranda Admin (sesudah login)

3.2.4.2 Tampilan Antar Muka Guru

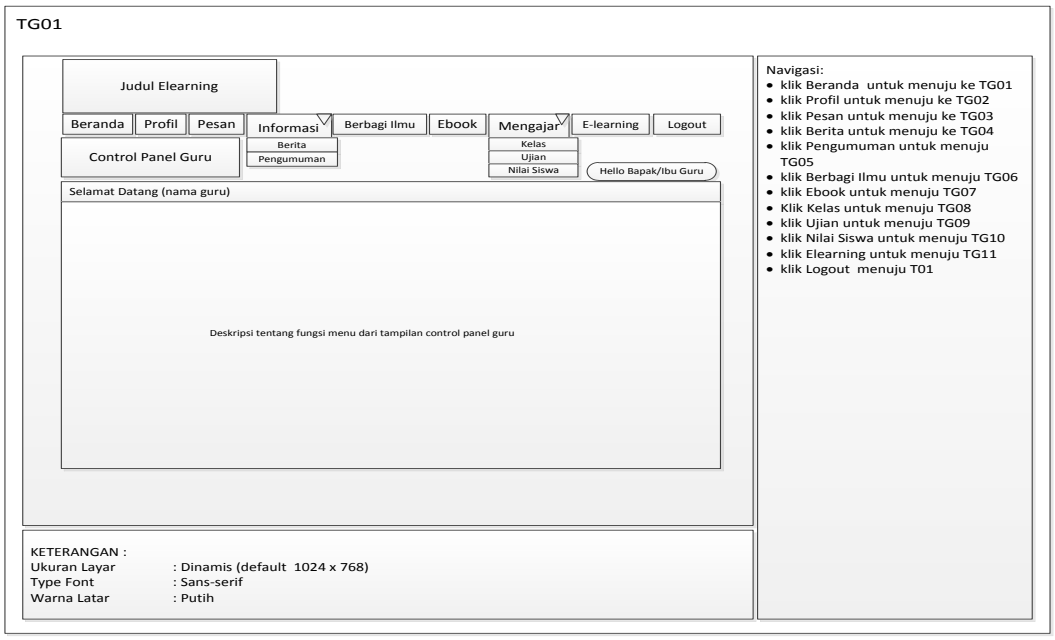
Antar muka guru bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun.

1. Halaman Beranda Guru (sebelum login)



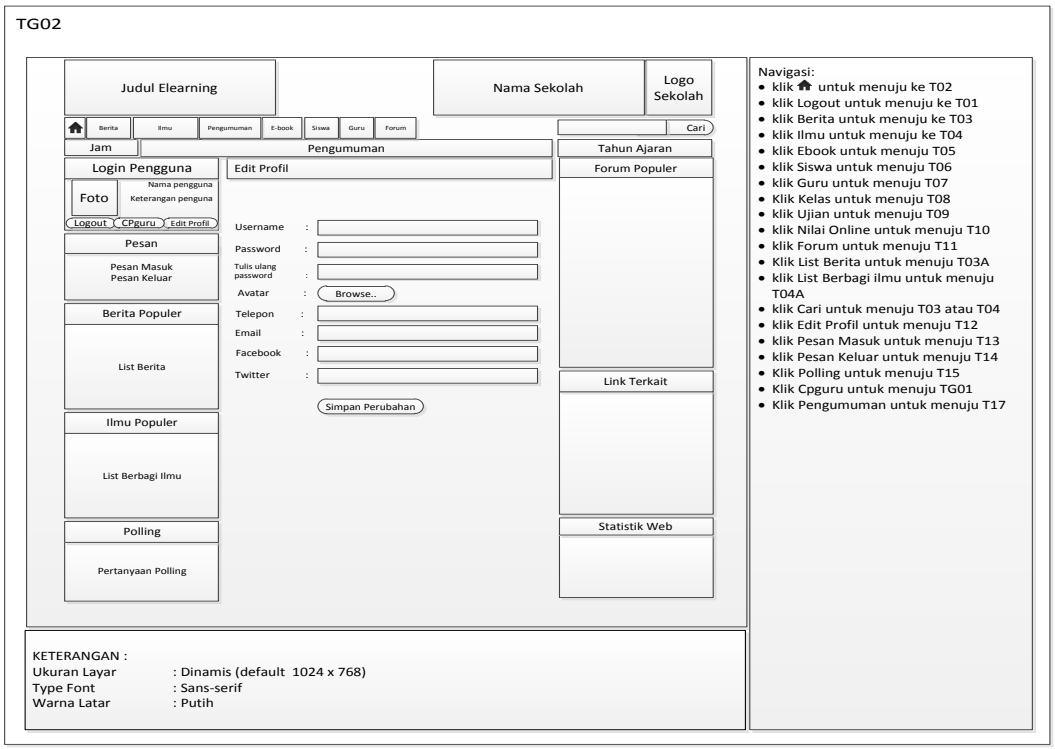
Gambar 3. 163Tampilan Halaman Beranda Guru (sebelum login)

2. Halaman Beranda (Guru)



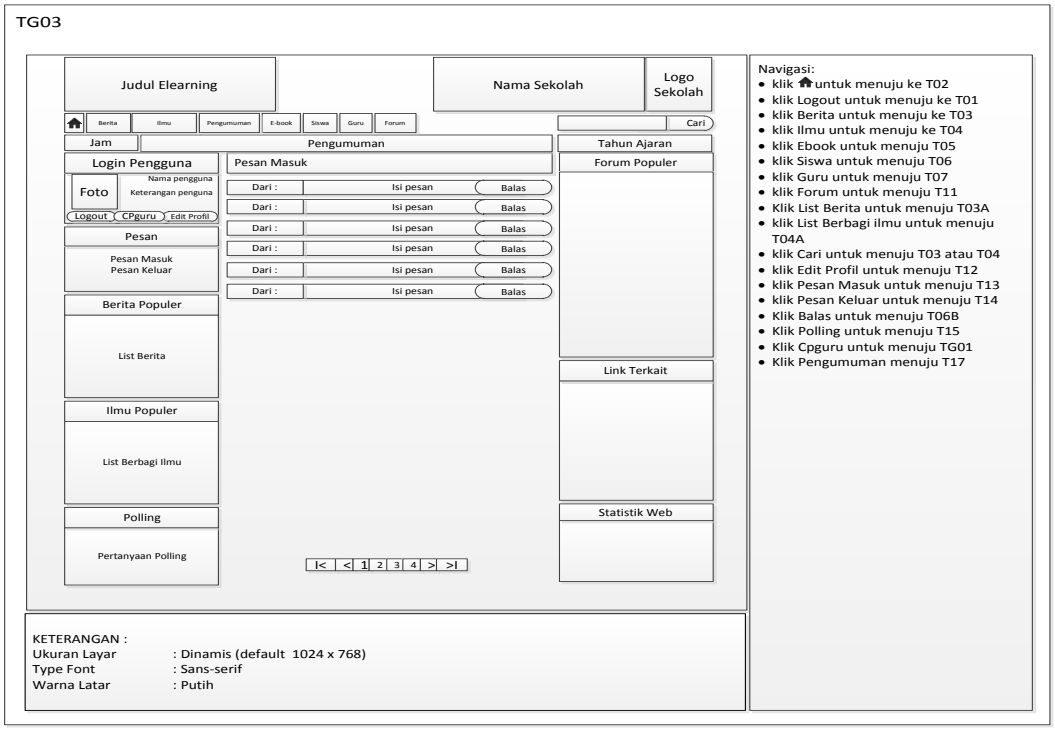
Gambar 3. 164 Tampilan Halaman Beranda (Guru)

3. Halaman Edit Profil (Guru)



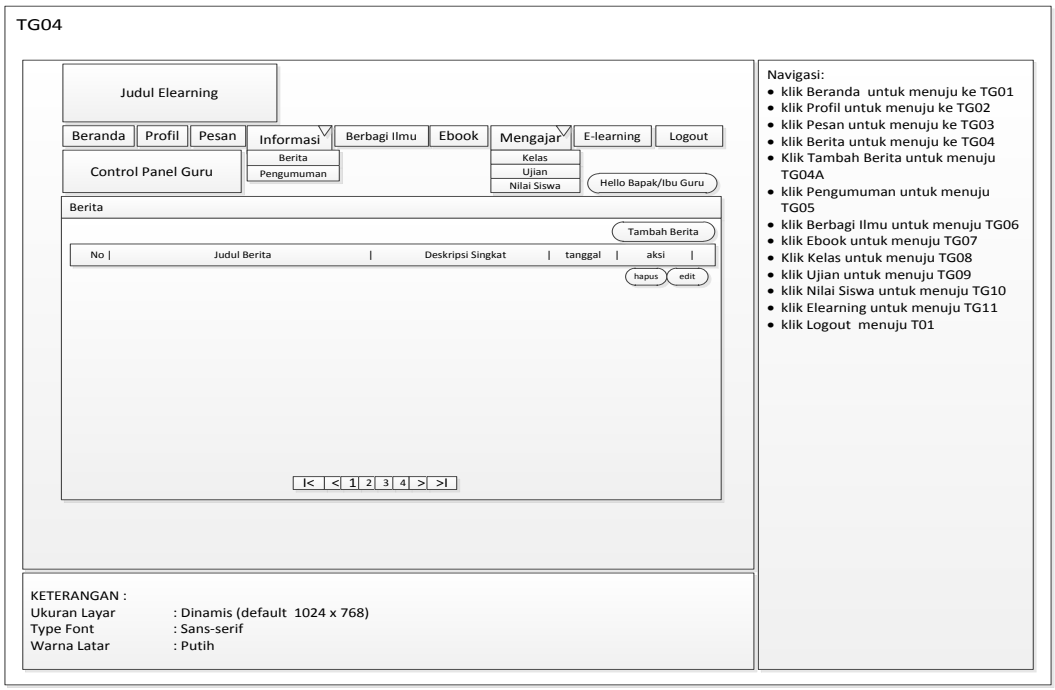
Gambar 3. 165 Halaman Edit Profil (Guru)

4. Halaman Pesan (Guru)



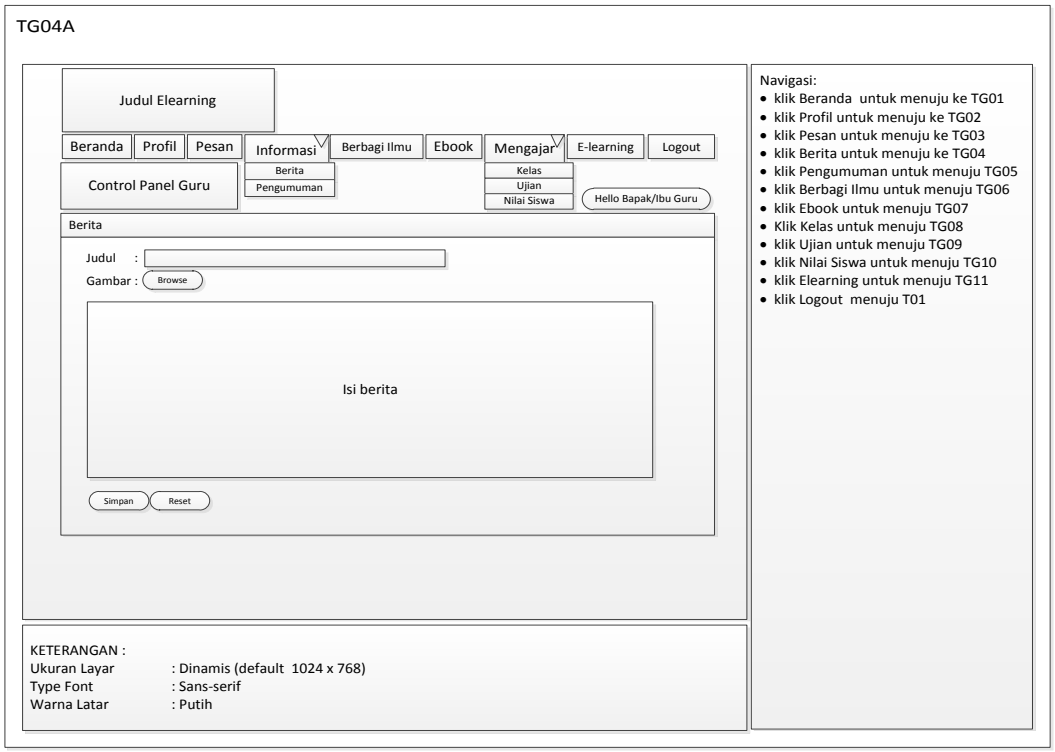
Gambar 3. 166 Halaman Pesan (Guru)

5. Halaman Berita (Guru)



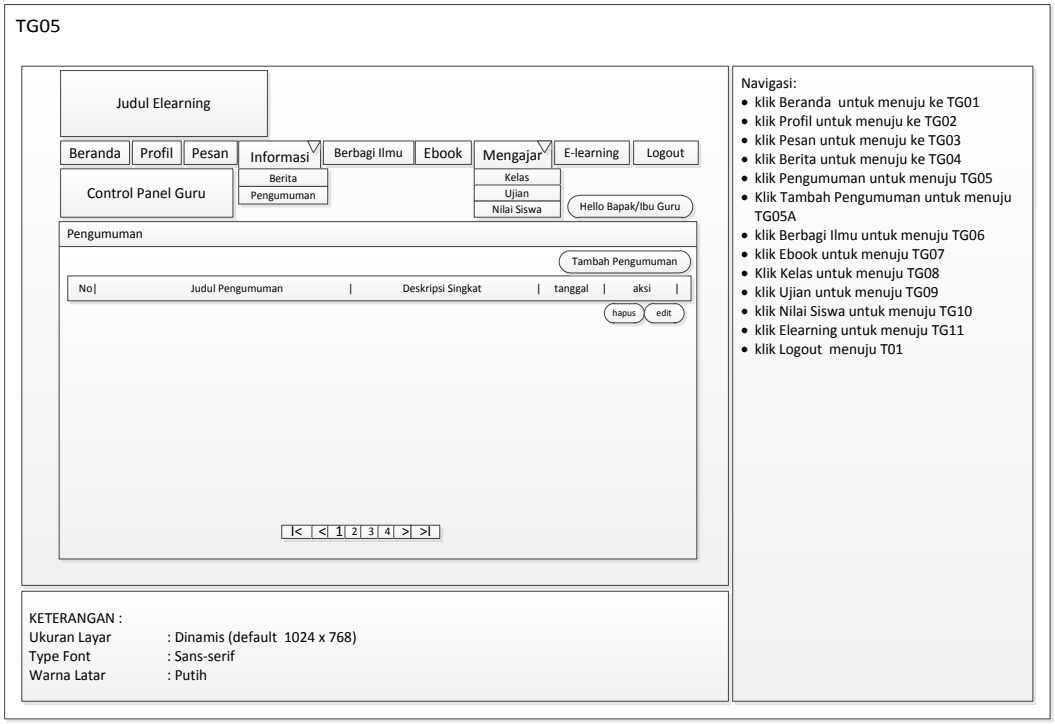
Gambar 3. 167 Tampiilan Halaman Berita (Guru)

6. Halaman Tambah Berita (Guru)



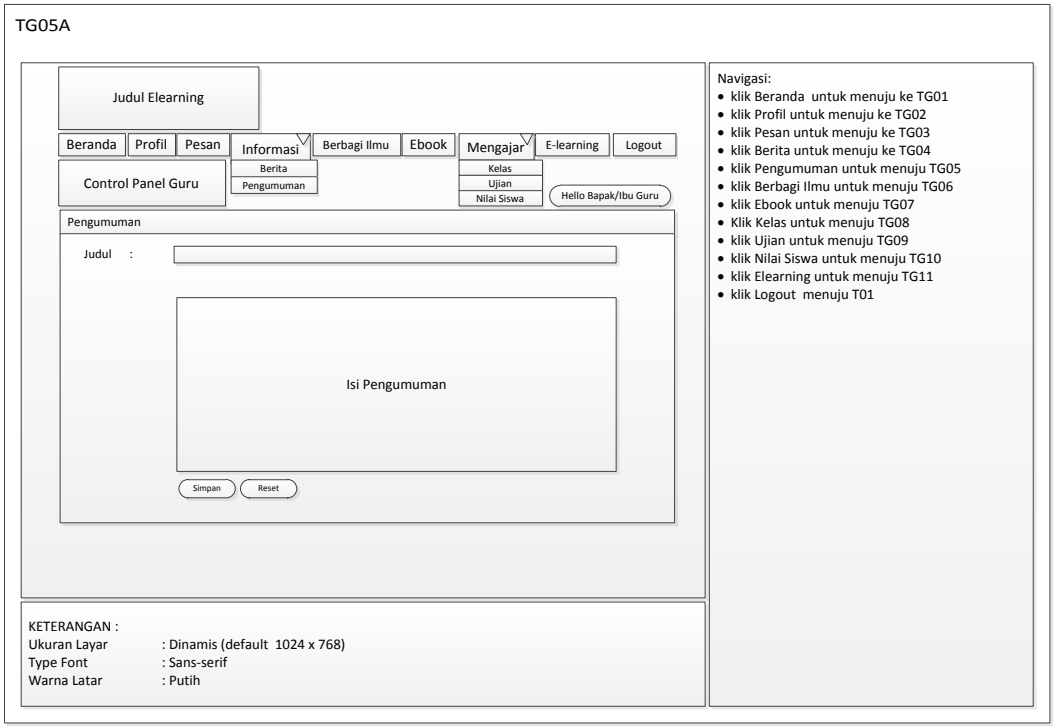
Gambar 3. 168 Tampilan Halaman Tambah Berita (Guru)

7. Halaman Pengumuman (Guru)



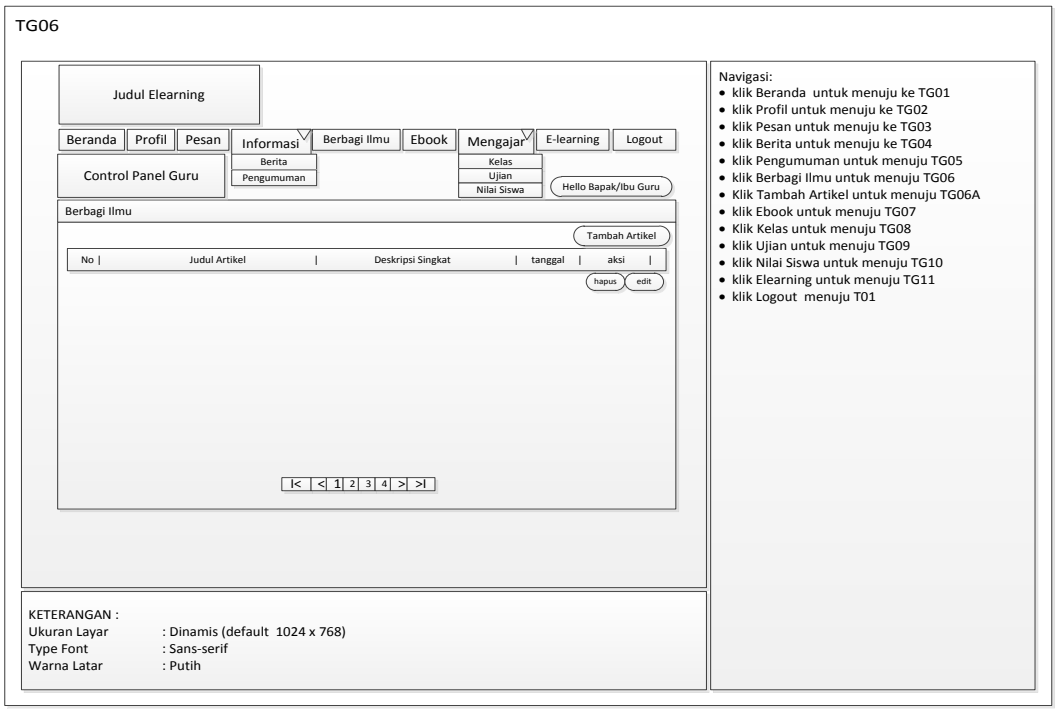
Gambar 3. 169 Tampilan Halaman Pengumuman (Guru)

8. Halaman Tambah Pengumuman (Guru)



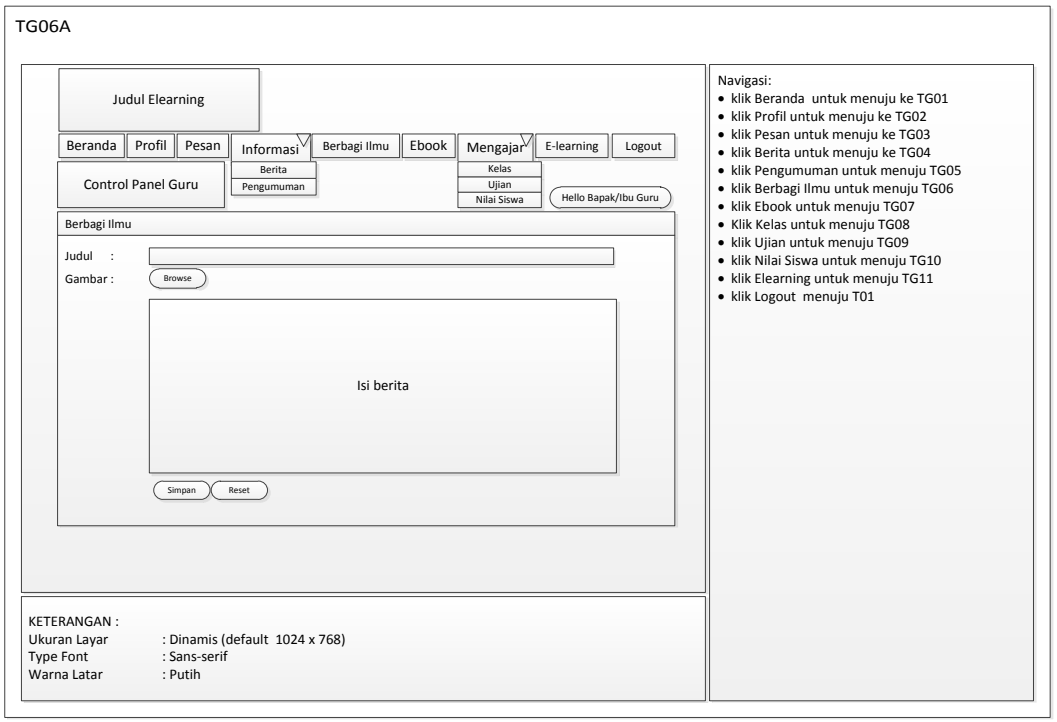
Gambar 3. 170 Tampilan Halaman Tambah Pengumuman (Guru)

9. Halaman Ilmu (Guru)



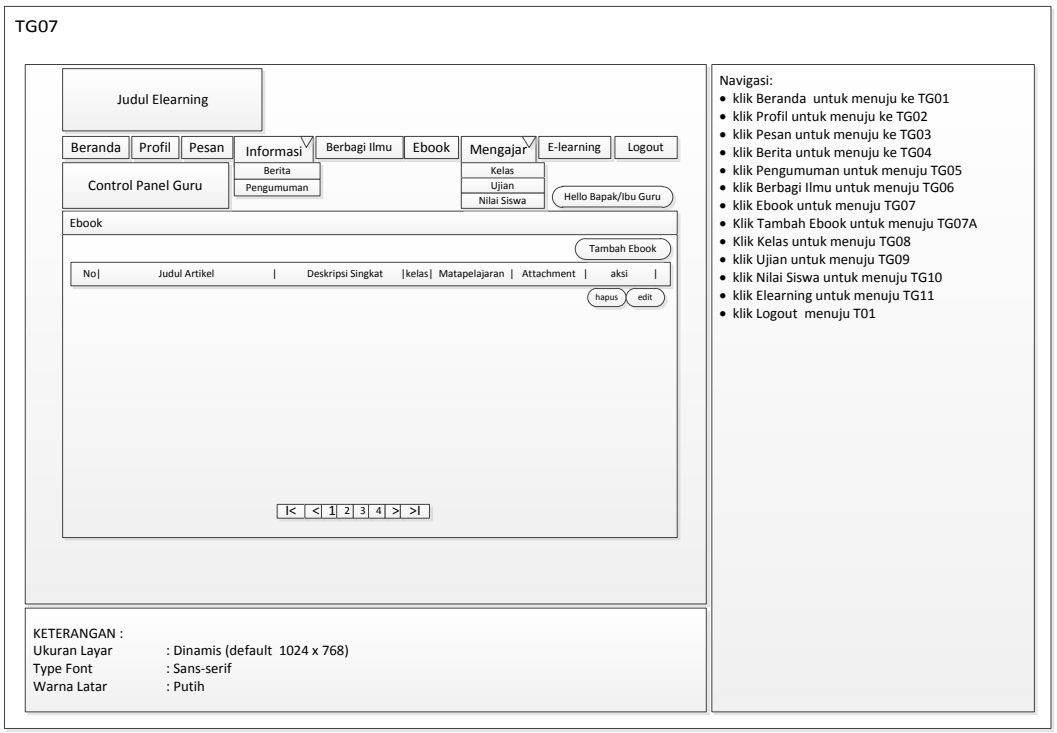
Gambar 3. 171 Tampilan Halaman Ilmu (Guru)

10. Halaman Tambah Ilmu (Guru)



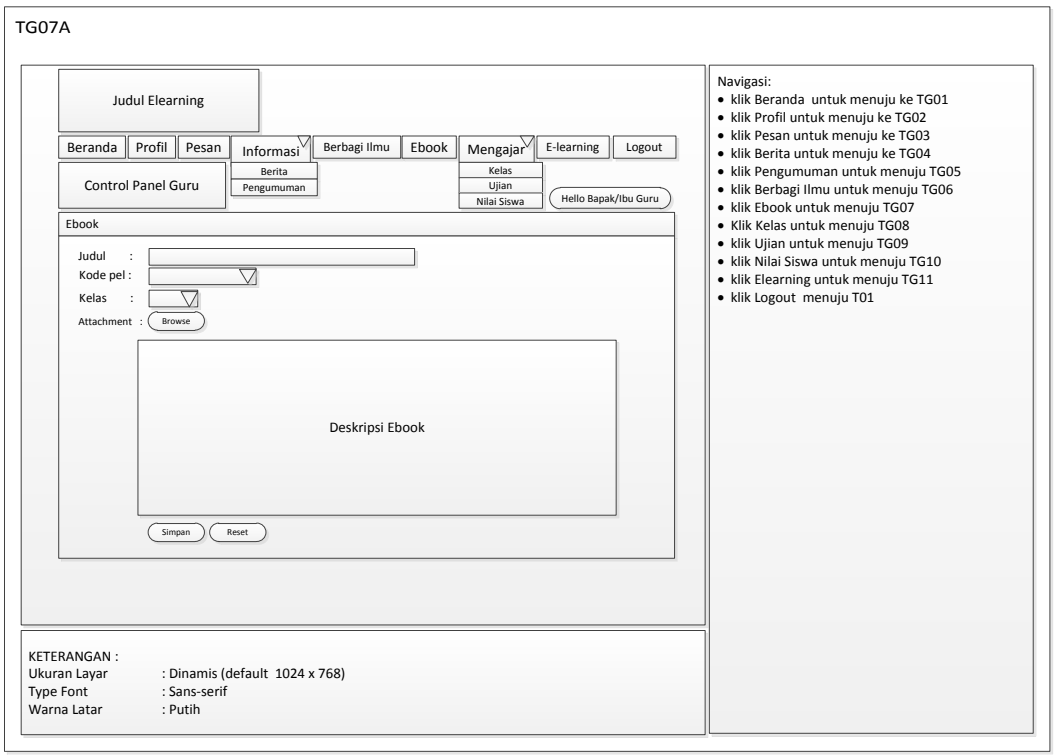
Gambar 3. 172 Tampilan Halaman Tambah Ilmu (Guru)

11. Halaman Ebook (Guru)



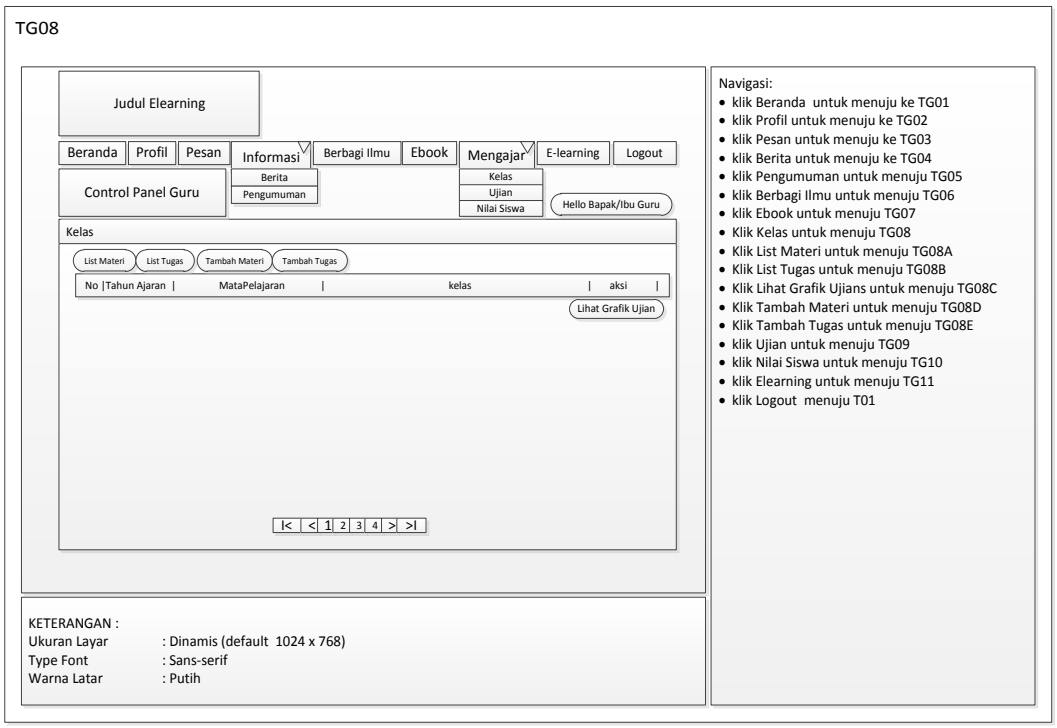
Gambar 3. 173 Tampilan Halaman Ebook (Guru)

12. Halaman Tambah Ebook (Guru)



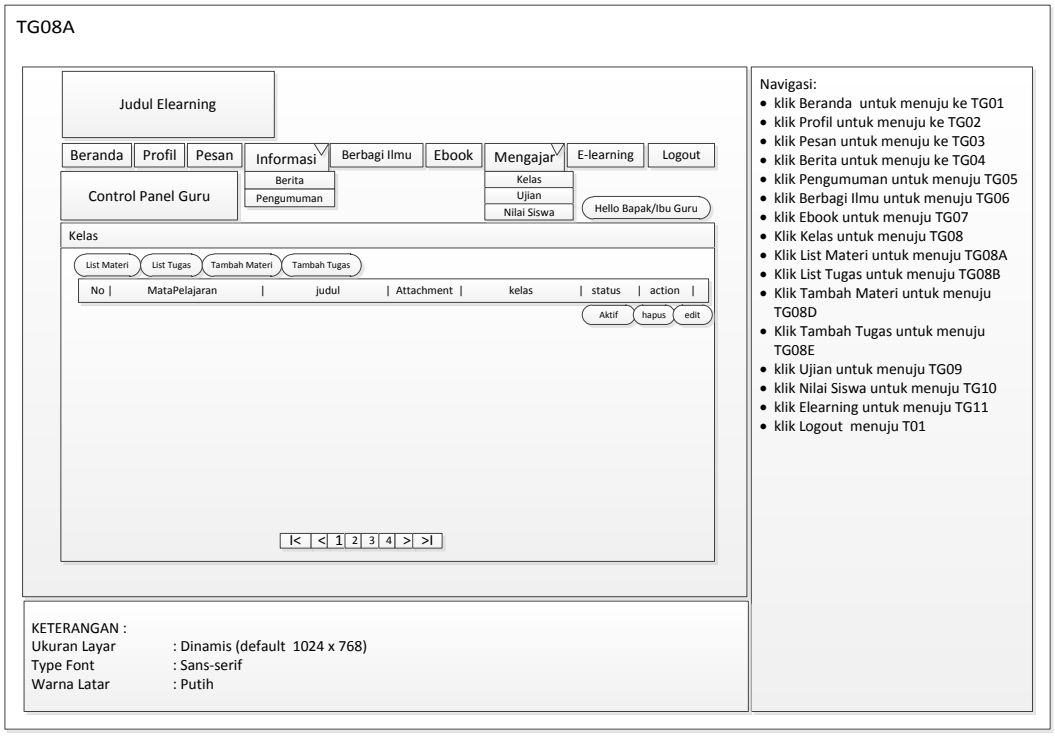
Gambar 3. 174 Tampilan Halaman Tambah Ebook (Guru)

13. Halaman Kelas (Guru)



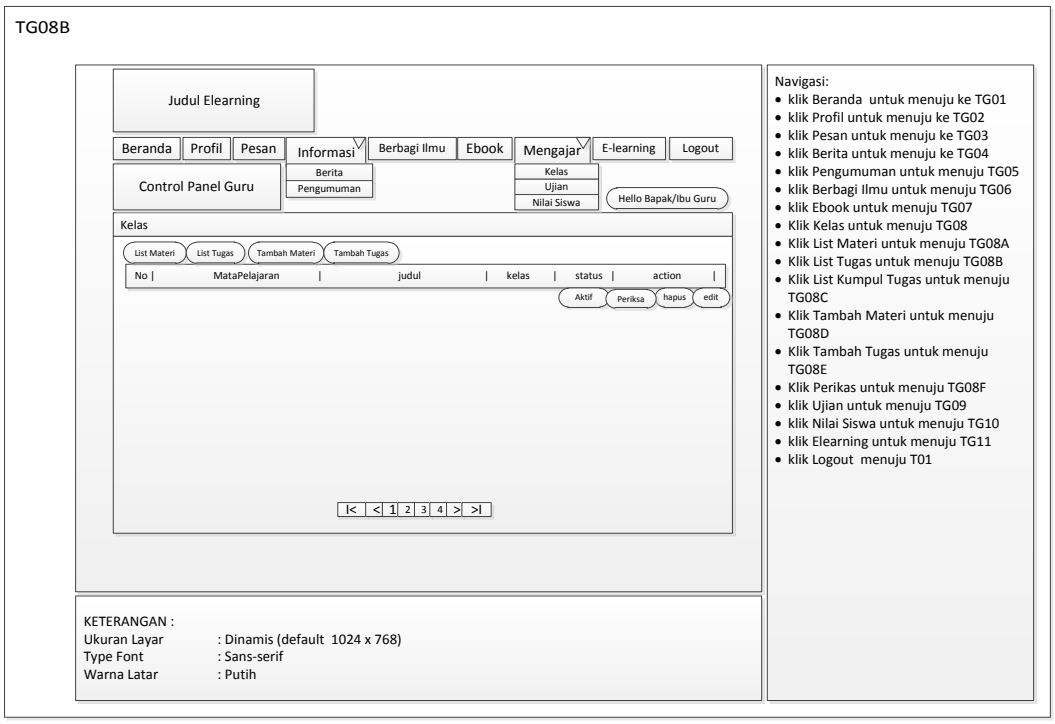
Gambar 3. 175 Tampilan Halaman Kelas (Guru)

14. Halaman List Materi (Guru)



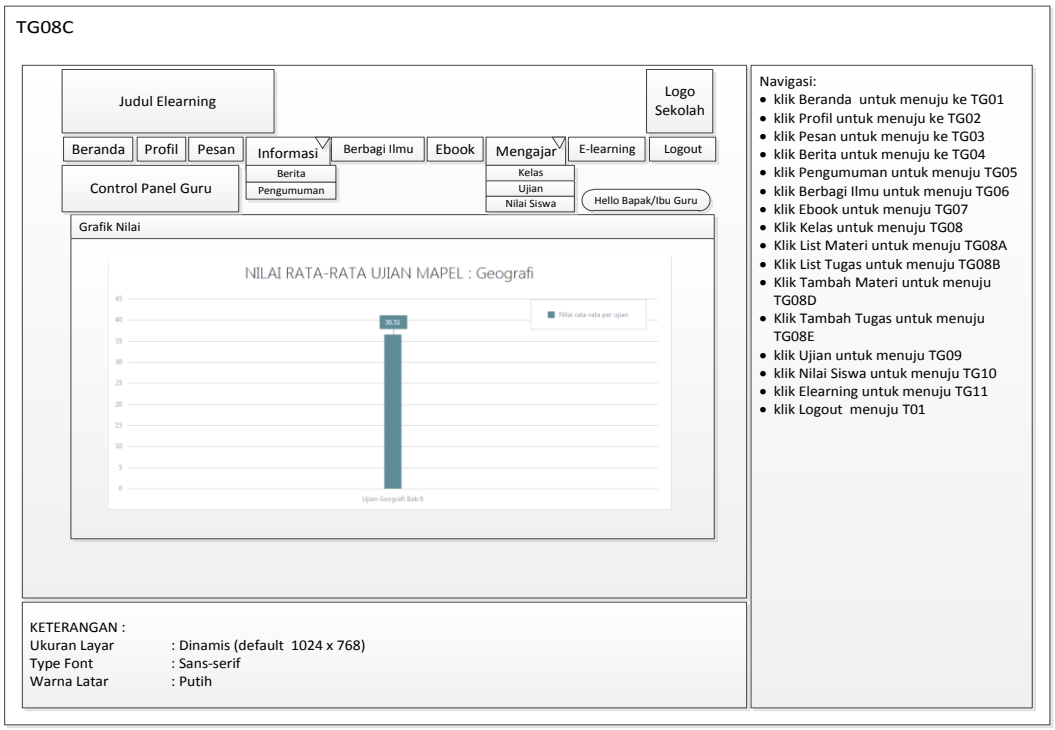
Gambar 3. 176 Tampilan Halaman List Materi (Guru)

15. Halaman List Tugas (Guru)



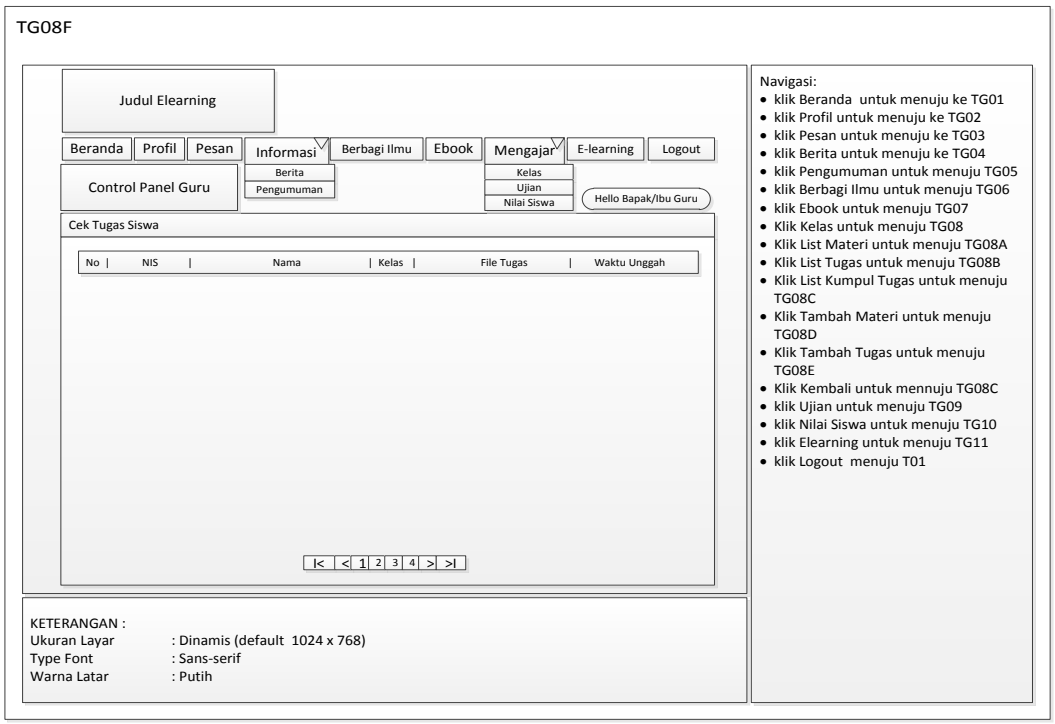
Gambar 3. 177 Tampilan Halaman List Tugas (Guru)

16. Halaman Lihat Grafik Nilai (Guru)



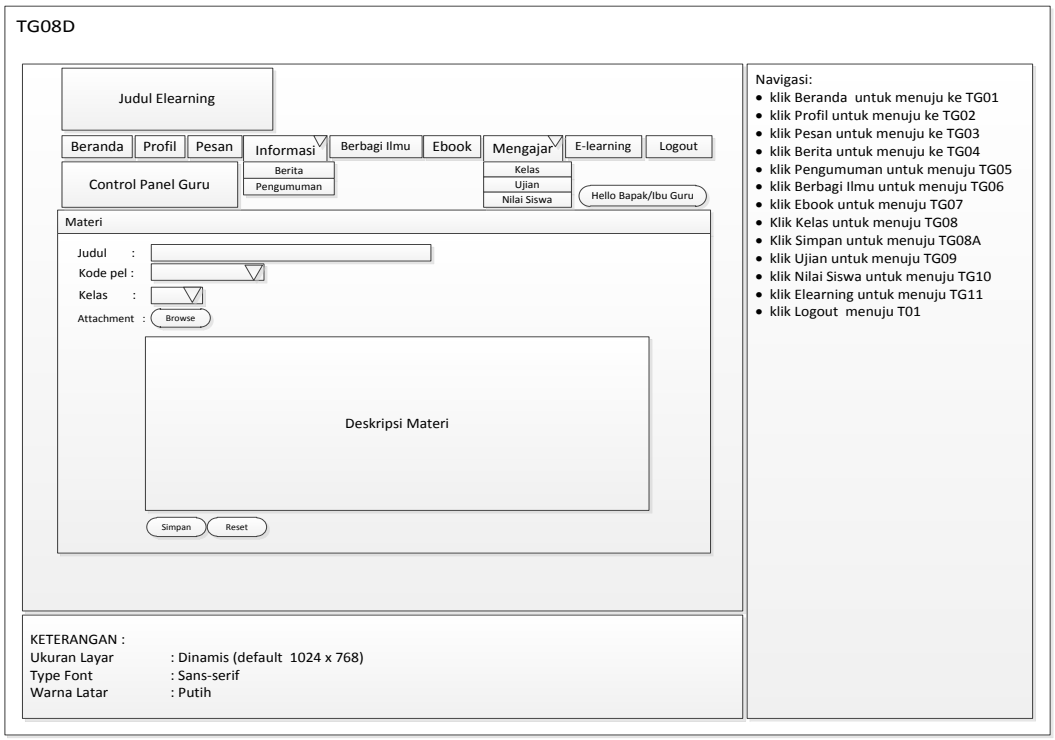
Gambar 3. 178 Tampilan Halaman Lihat Grafik Nilai (Guru)

17. Halaman Periksa Tugas Siswa (Guru)



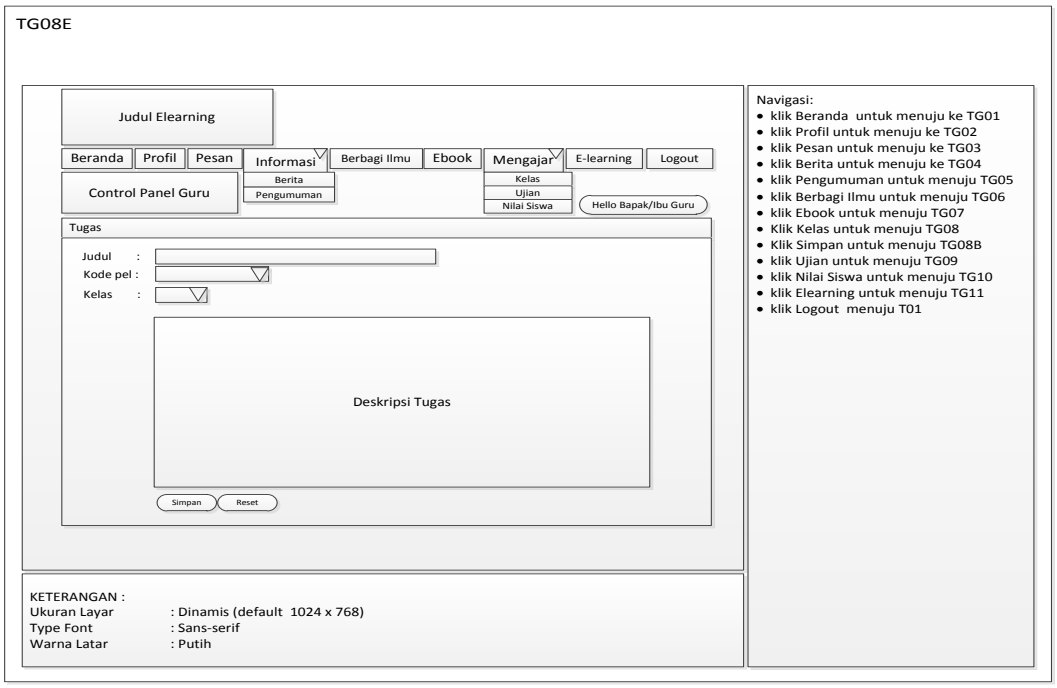
Gambar 3. 179 Tampilan Halaman Periksa Tugas Siswa (Guru)

18. Halaman Tambah Materi (Guru)



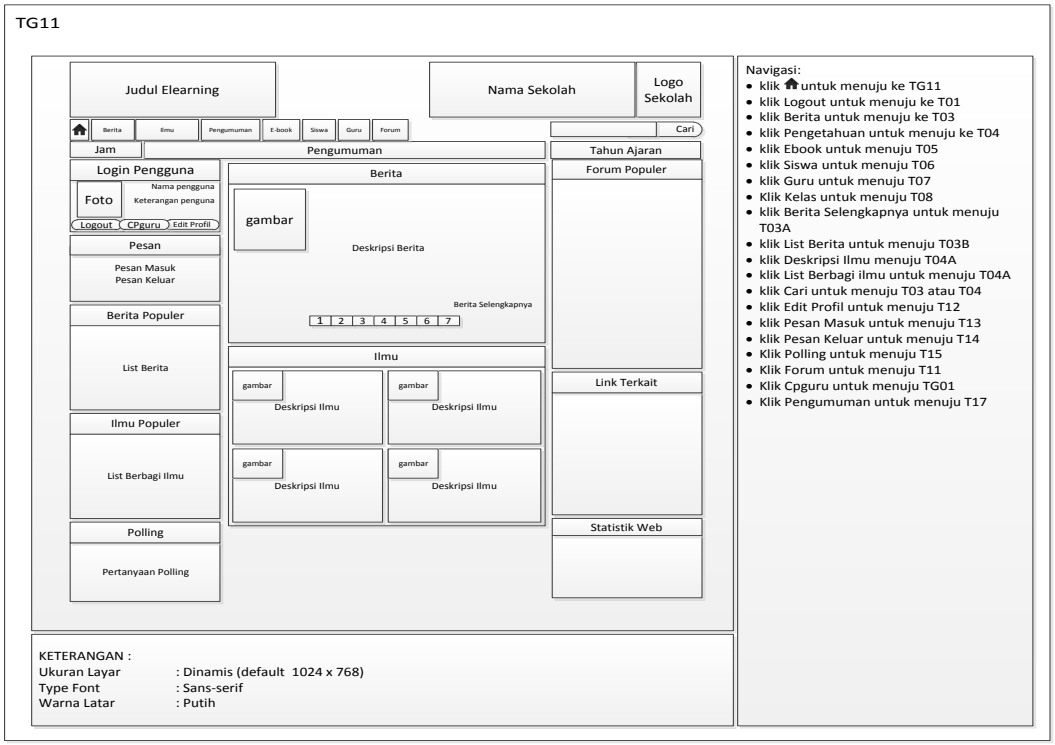
Gambar 3. 180 Tampilan Halaman Tambah Materi (Guru)

19. Halaman Tambah Tugas (Guru)



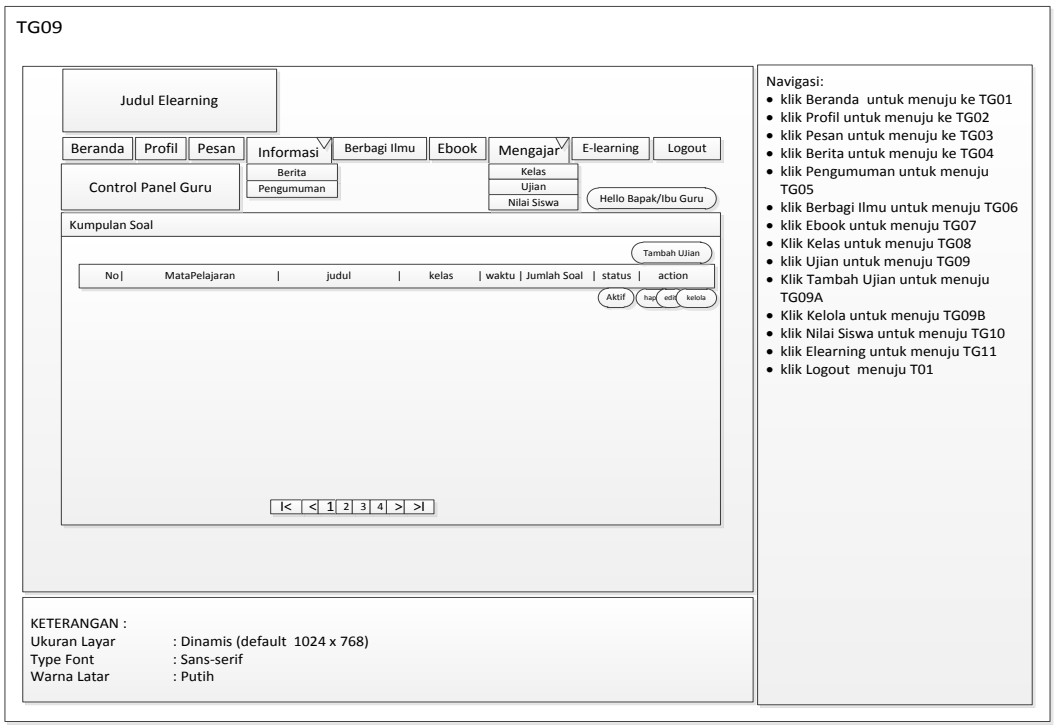
Gambar 3. 181 Tampilan Halaman Tambah Tugas (Guru)

20. Halaman Beranda Guru (Sesudah login)



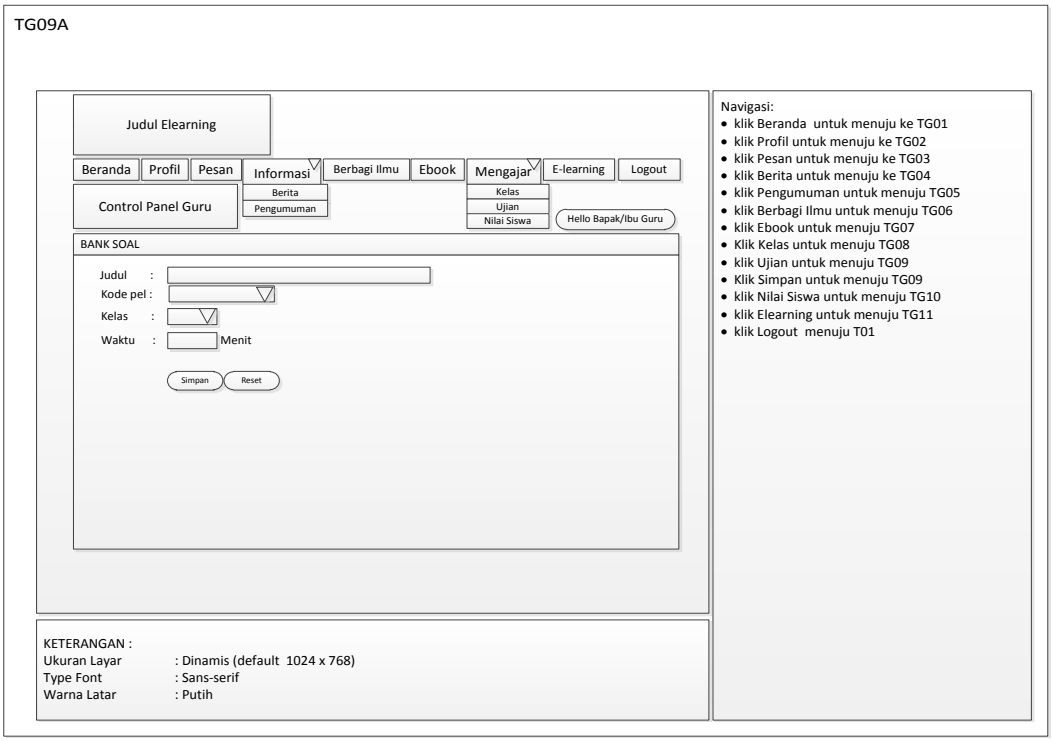
Gambar 3. 182 Tampilan Halaman Beranda Guru (sesudah login)

21. Halaman Ujian (Guru)



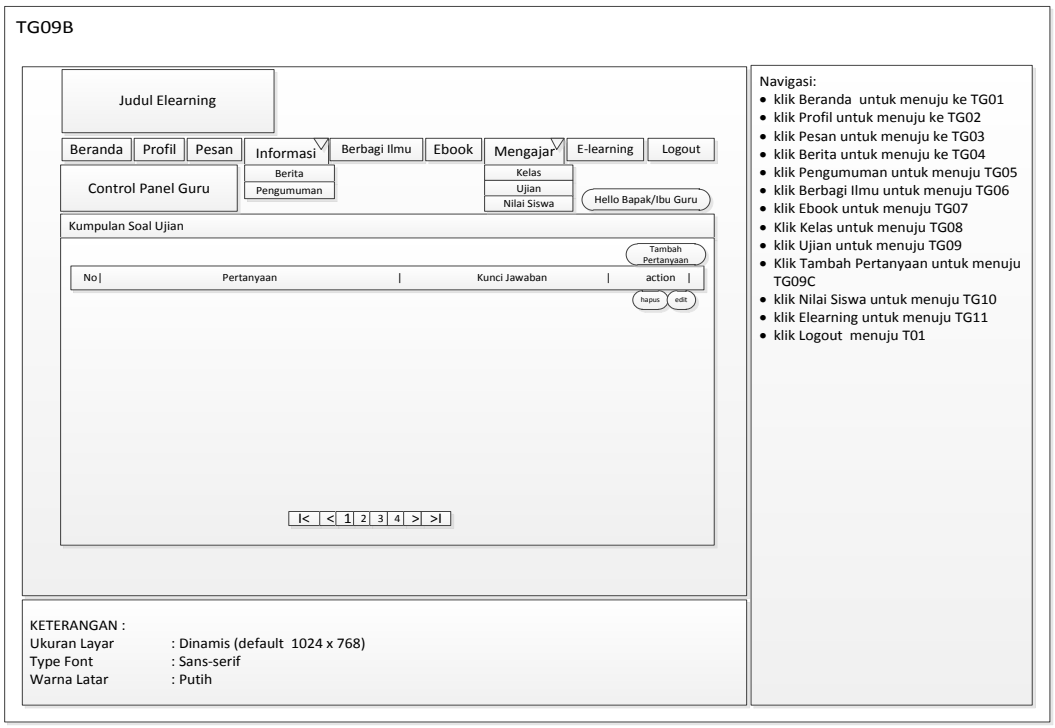
Gambar 3. 183 Tampilan Halaman Ujian Guru

22. Halaman Tambah Ujian (Guru)



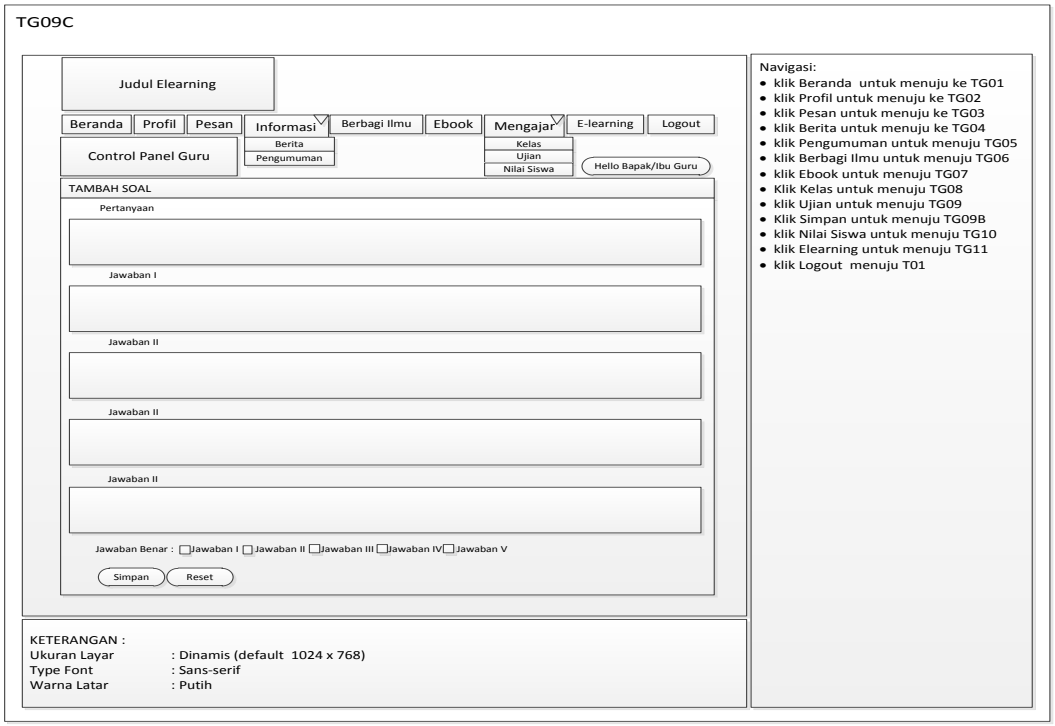
Gambar 3. 184 Tampilan Halaman Tambah Ujian (Guru)

23. Halaman Pertanyaan Ujian (Guru)



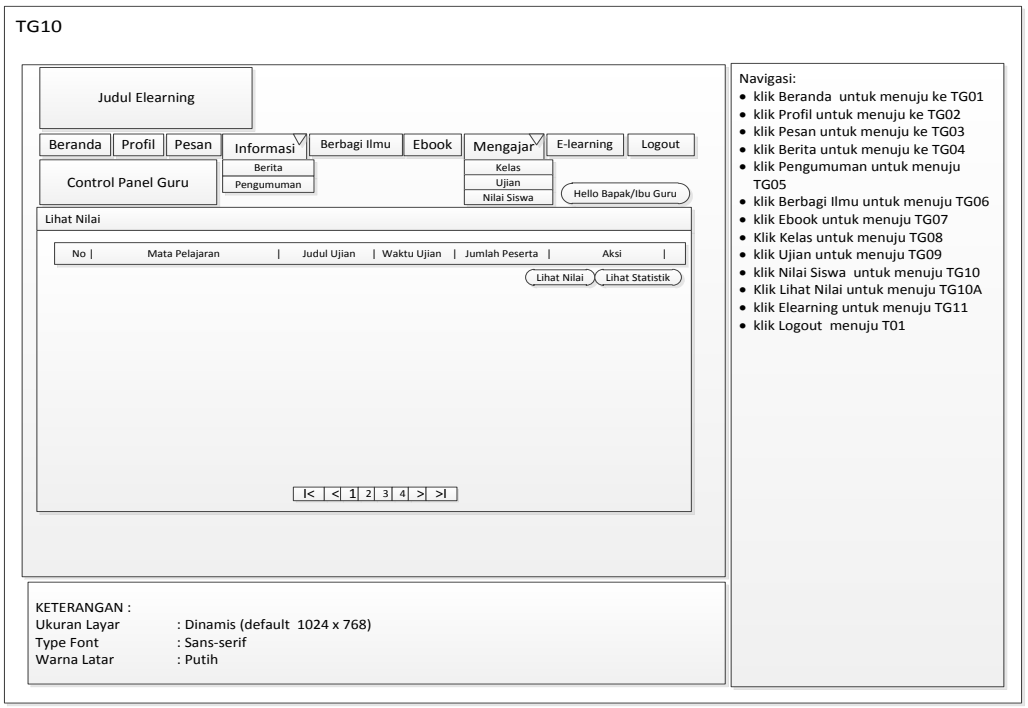
Gambar 3. 185 Tampilan Halaman Pertanyaan Ujian (Guru)

24. Halaman Tambah Pertanyaan Ujian (Guru)



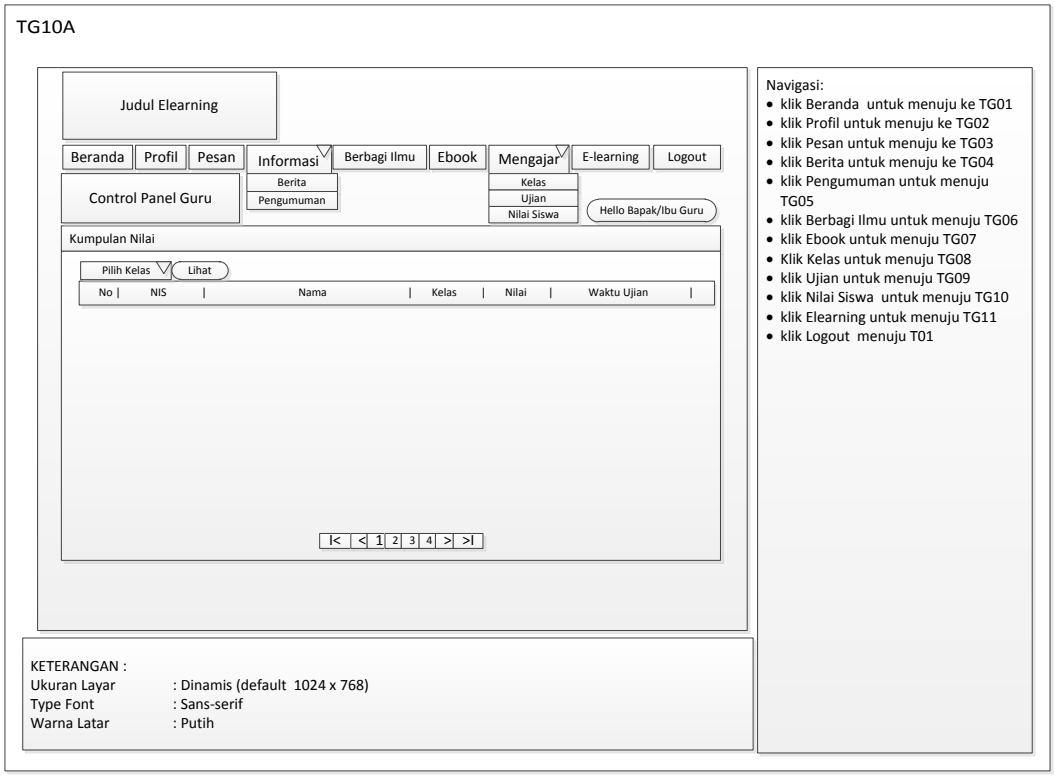
Gambar 3. 186 Tampilan Halaman Tambah Pertanyaan Ujian (Guru)

25. Halaman Nilai Siswa (Guru)



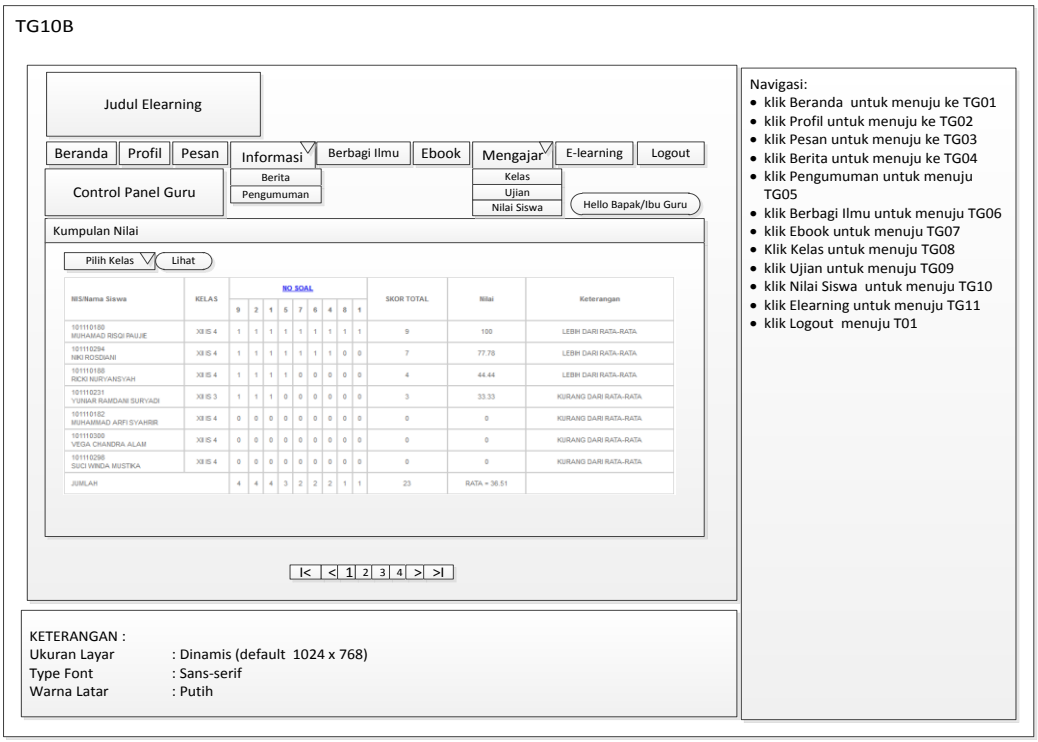
Gambar 3. 187 Tampilan Halaman Nilai Online Guru

26. Halaman Lihat Nilai (Guru)



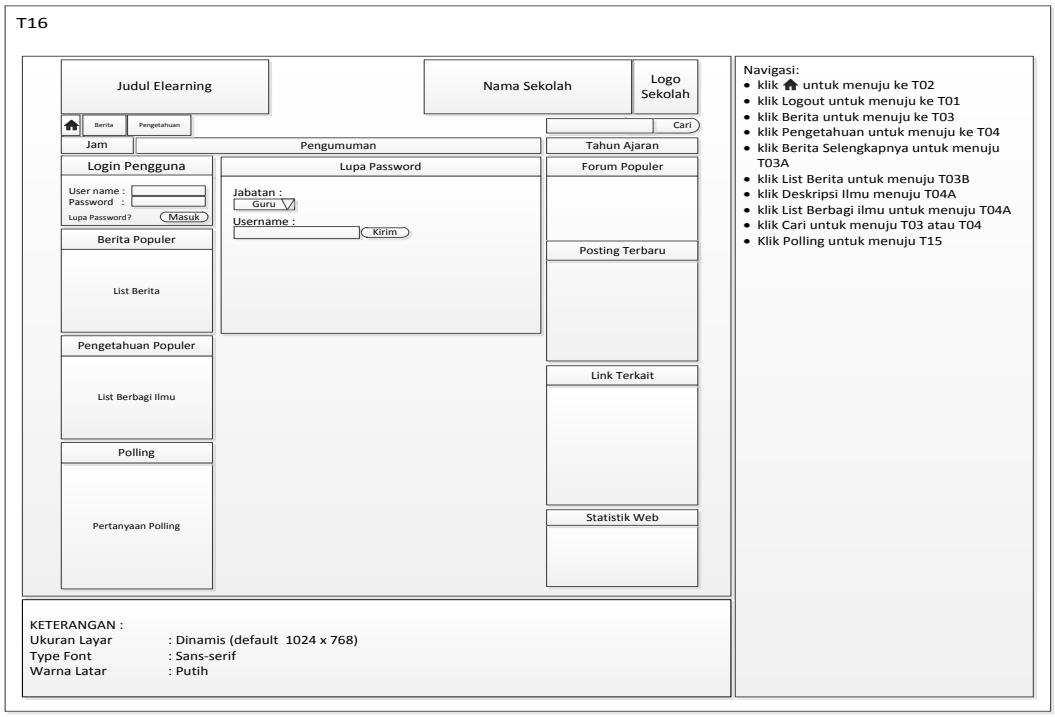
Gambar 3. 188 Tampilan Halaman Lihat Nilai (Guru)

27. Halaman Lihat Statistik (Guru)



Gambar 3. 189 Halaman Lihat Statistik (Guru)

28. Halaman Lupa Password Guru

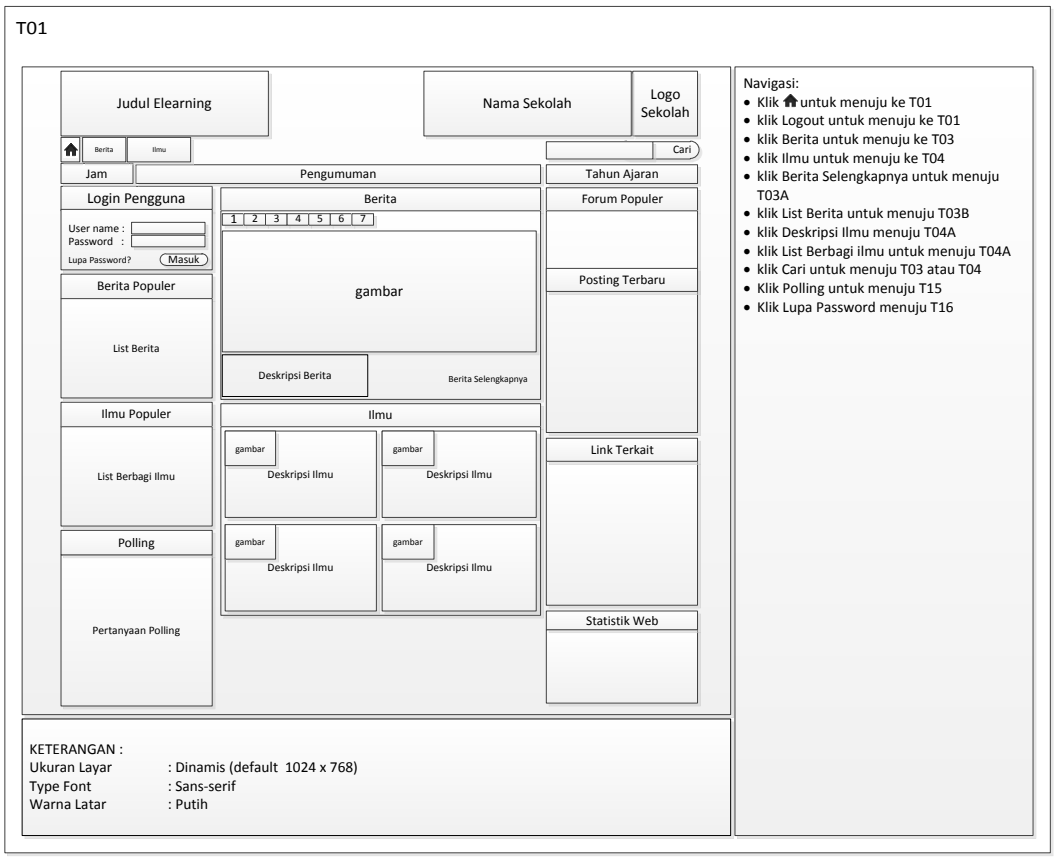


Gambar 3. 190 Tampilan Halaman Lupa Password Guru

3.2.4.3 Tampilan Antar Muka Siswa

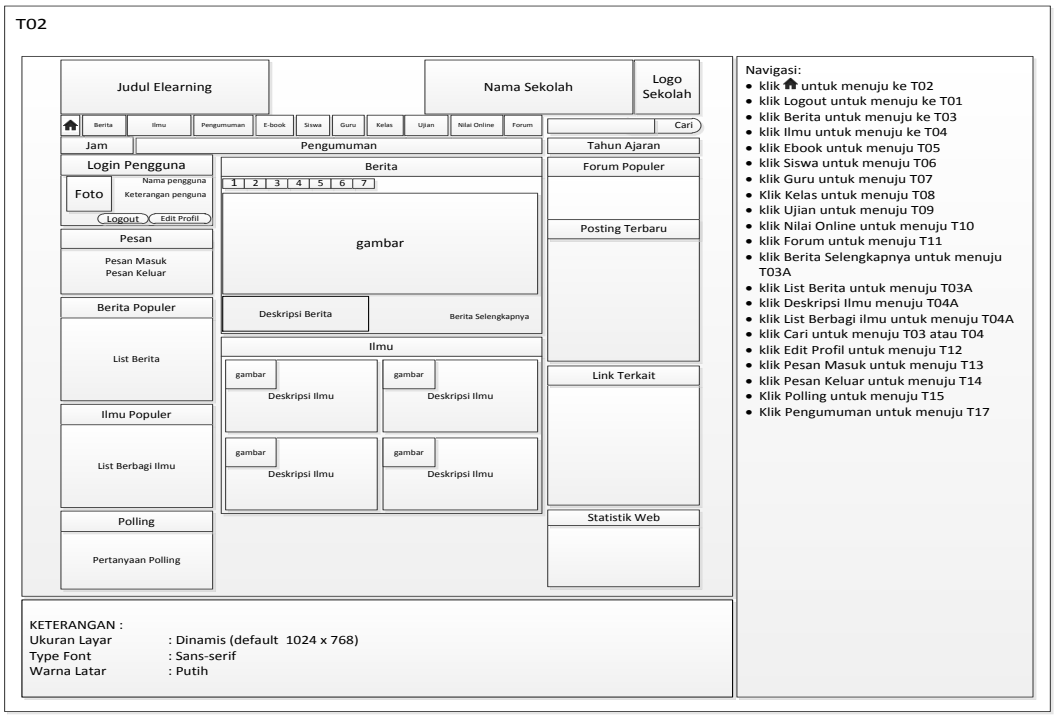
Antar muka siswa bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun.

1. Halaman Beranda (sebelum login)



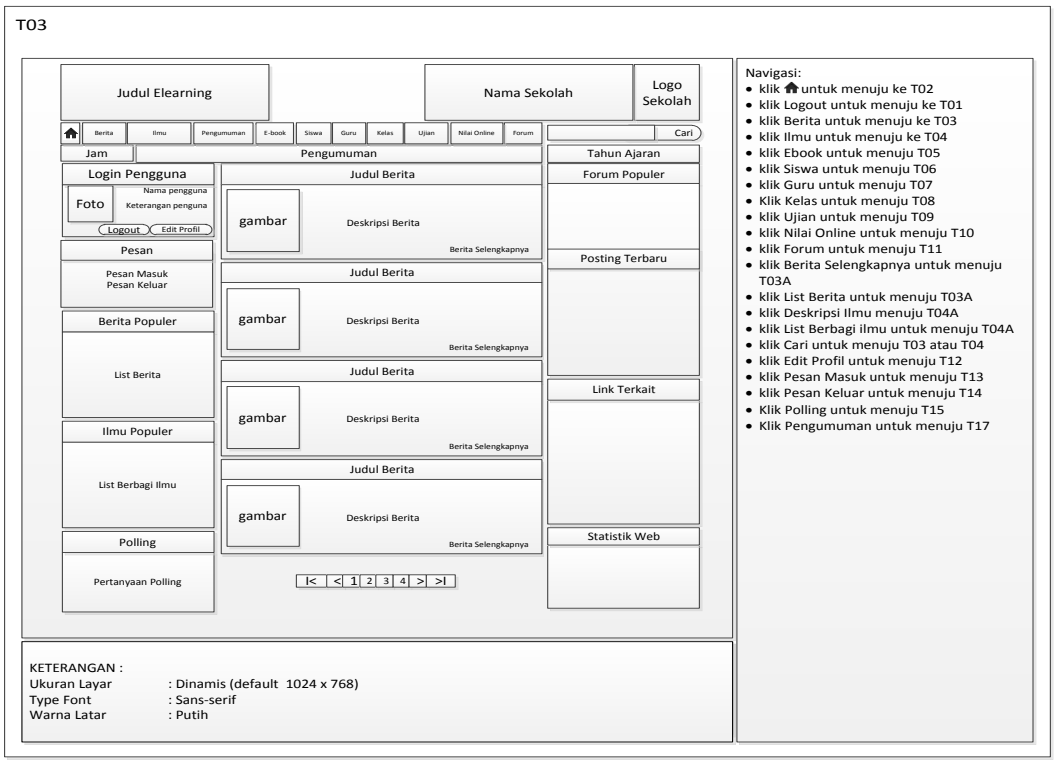
Gambar 3. 191 Tampilan Halaman Beranda (sebelum login)

2. Halaman Beranda (sesudah login)



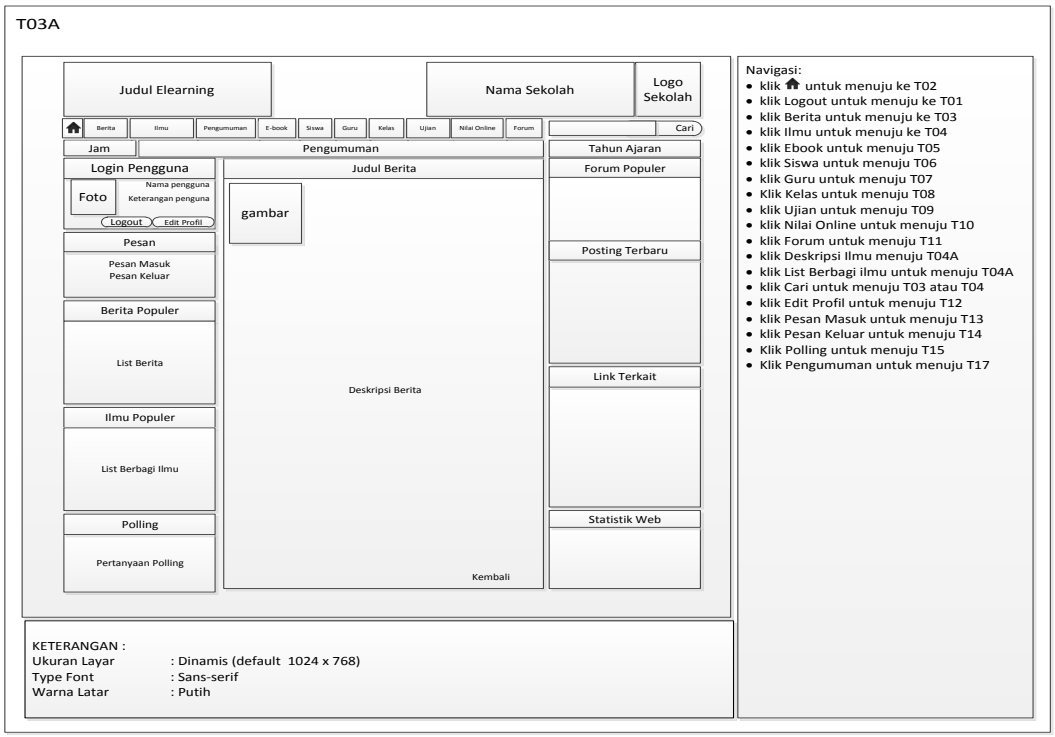
Gambar 3. 192 Tampilan Halaman Beranda (sesudah login)

3. Halaman Berita



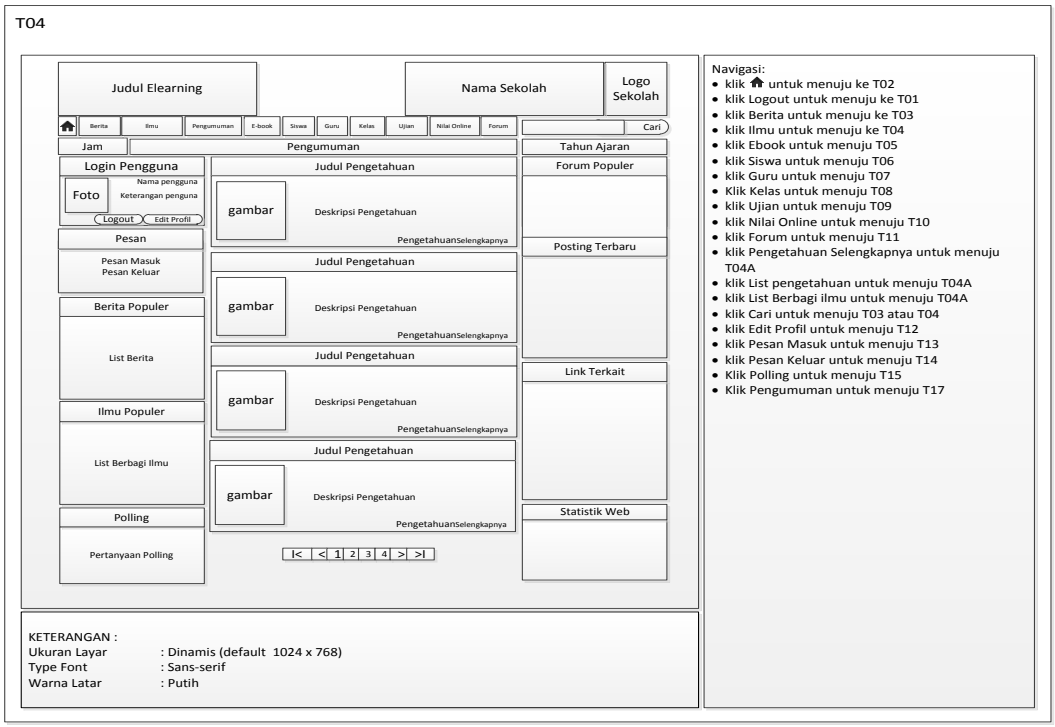
Gambar 3. 193 Tampilan Halaman Berita

4. Halaman Berita Lengkap



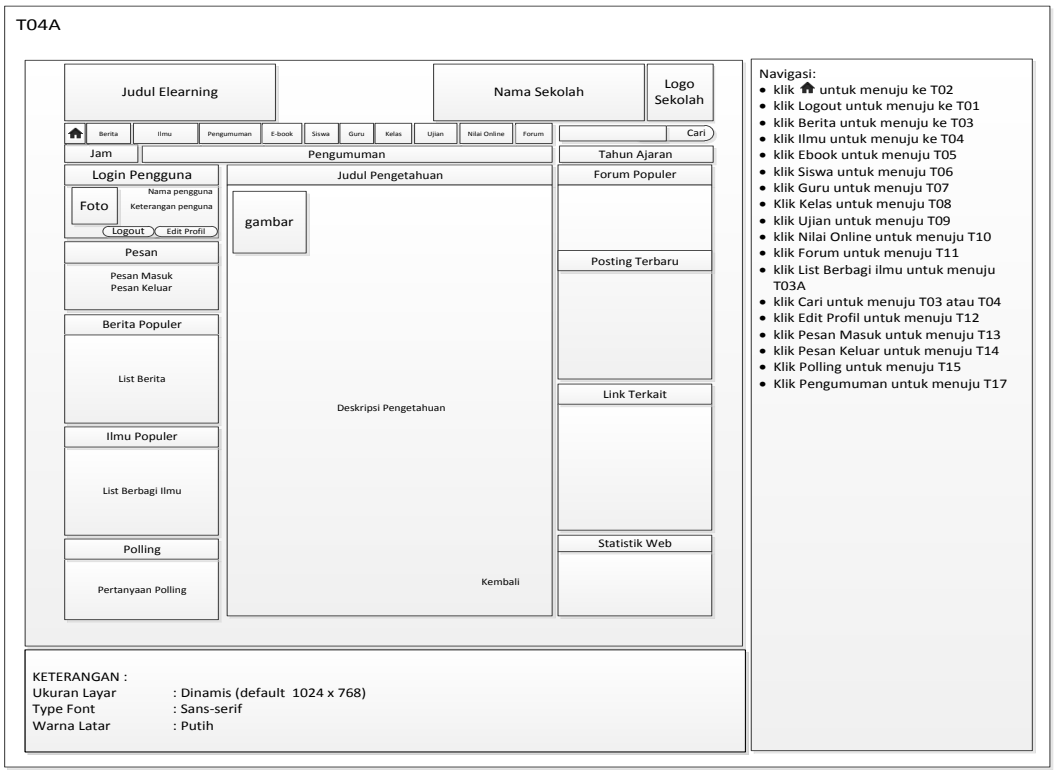
Gambar 3. 194 Tampilan Halaman Berita Lengkap

5. Halaman Ilmu



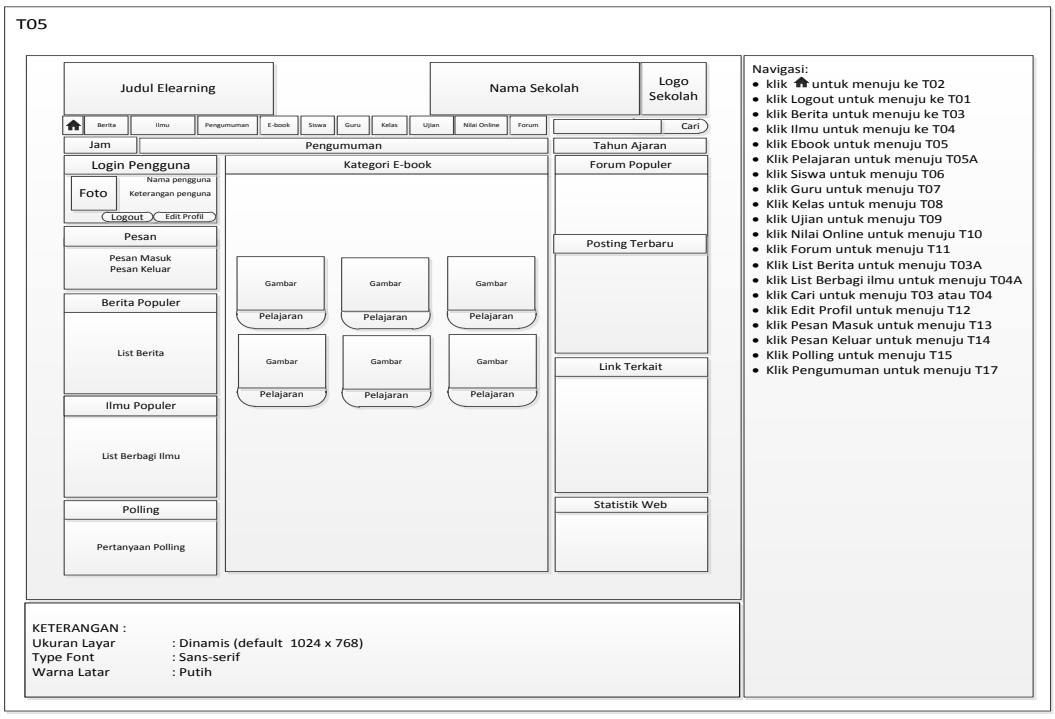
Gambar 3. 195 Tampilan Halaman Ilmu

6. Halaman Ilmu Lengkap



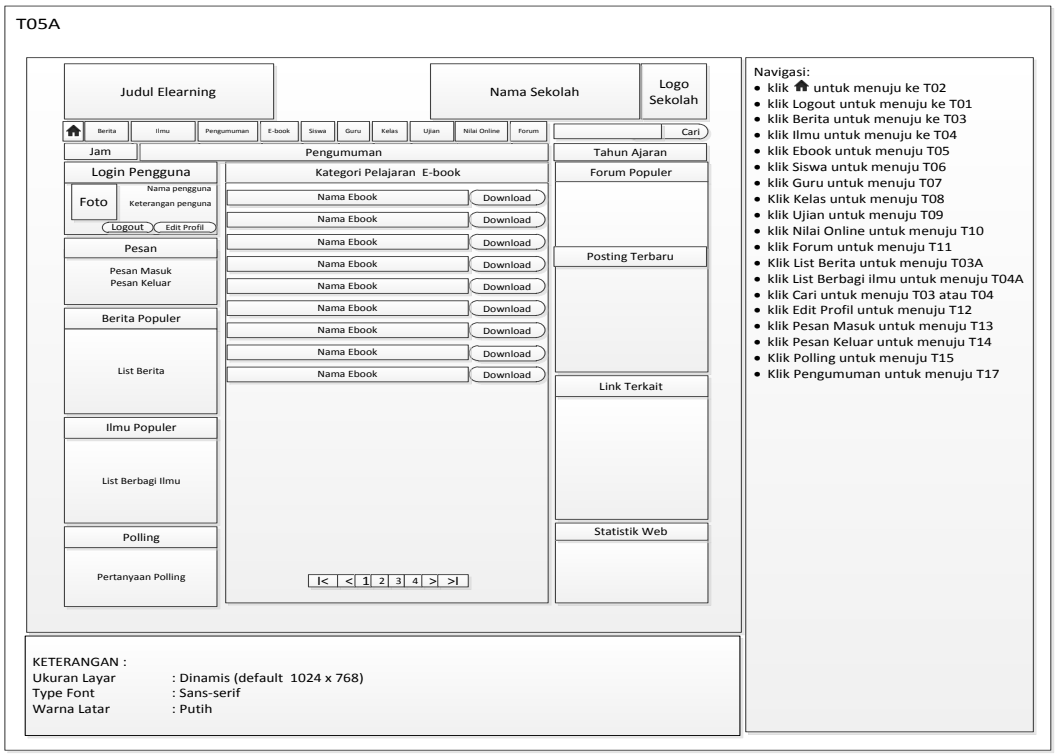
Gambar 3. 196 Tampilan Halaman Ilmu Lengkap

7. Halaman Ebook



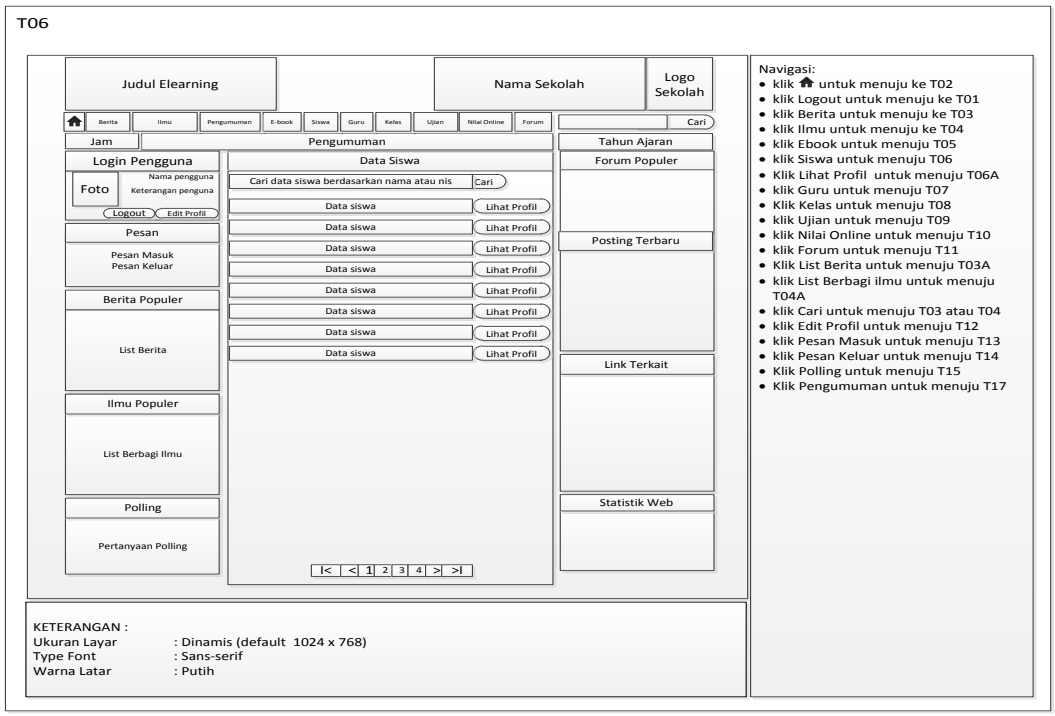
Gambar 3. 197 Tampilan Halaman Ebook

8. Halaman Ebook Kategori



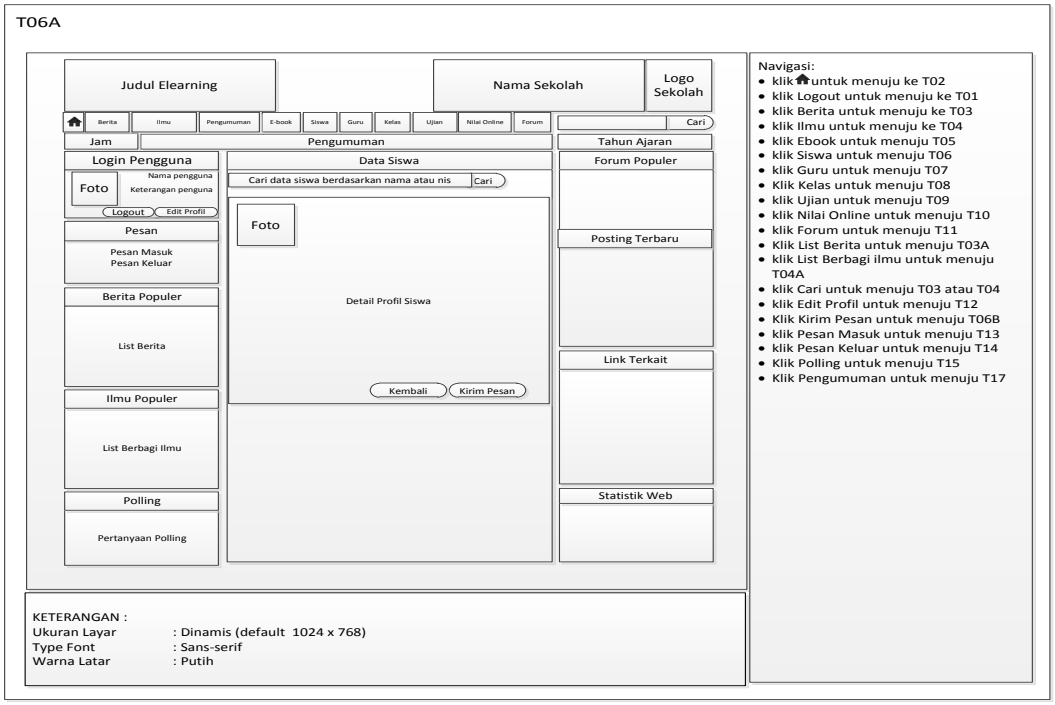
Gambar 3. 198 Tampilan Halaman Ebook Kategori

9. Halaman Siswa



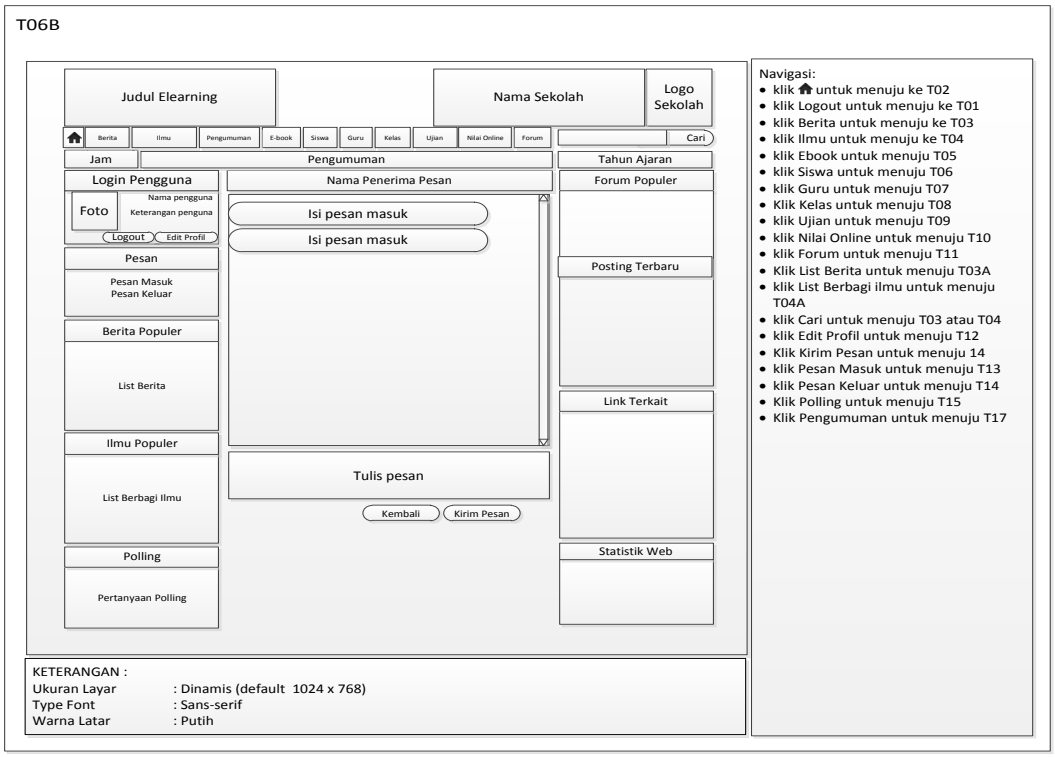
Gambar 3. 199 Tampilan Halaman Siswa

10. Halaman Siswa Profil



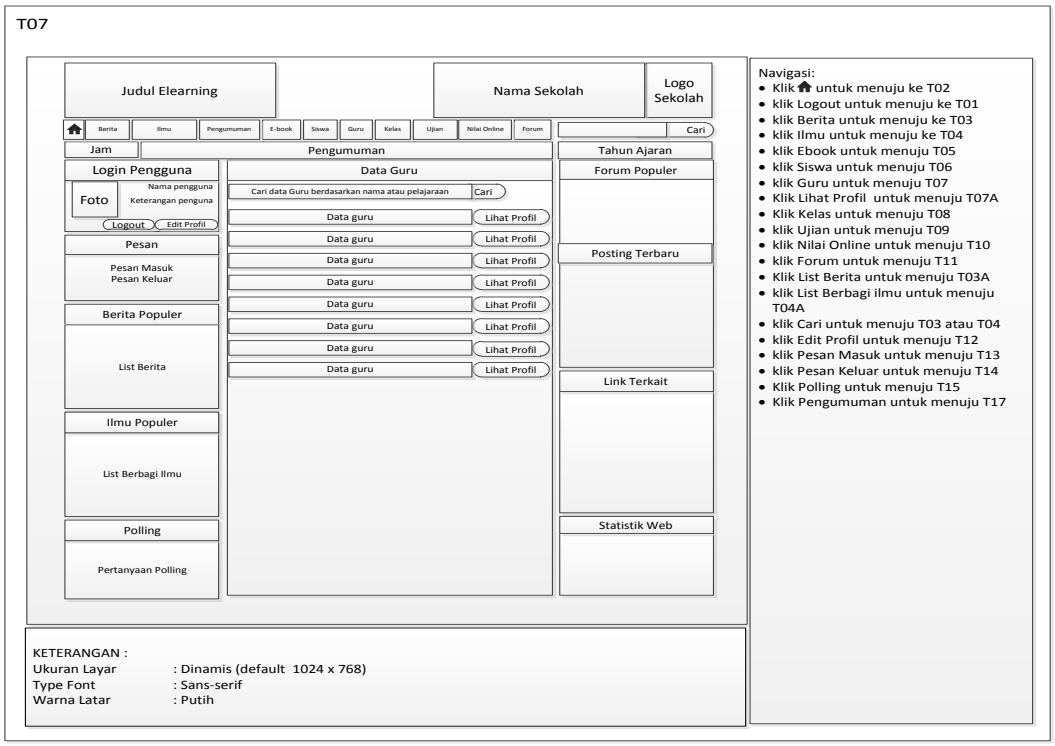
Gambar 3. 200 Tampilan Halaman Siswa Profil

11. Halaman Pesan



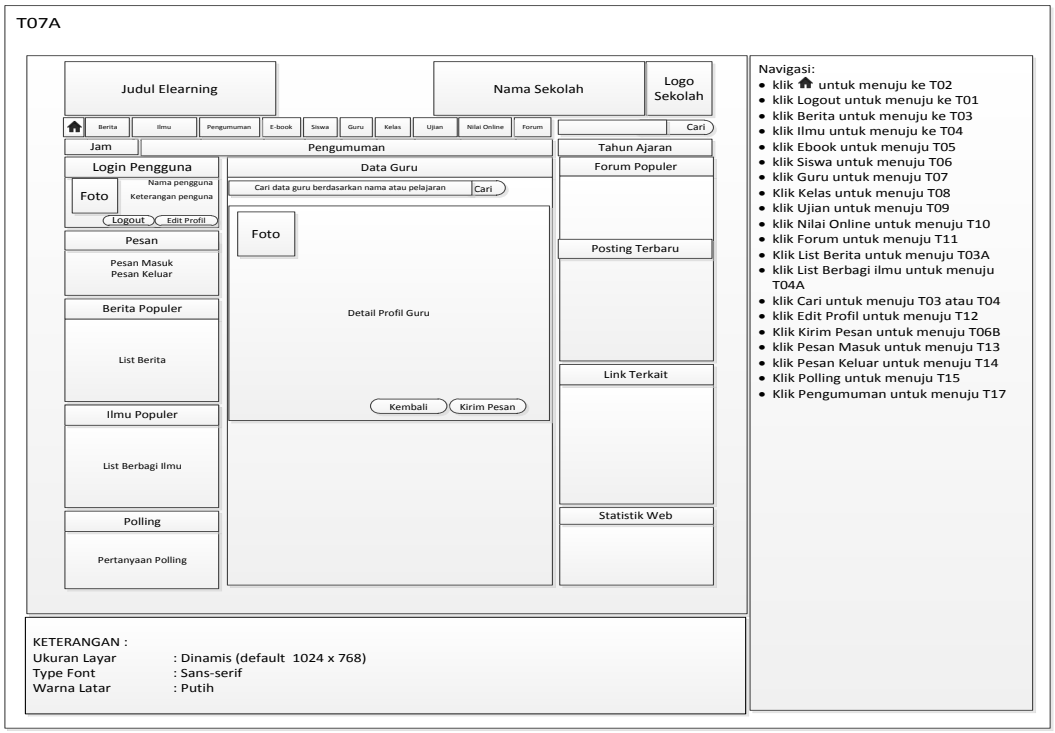
Gambar 3. 201 Tampilan Halaman Pesan

12. Halaman Guru



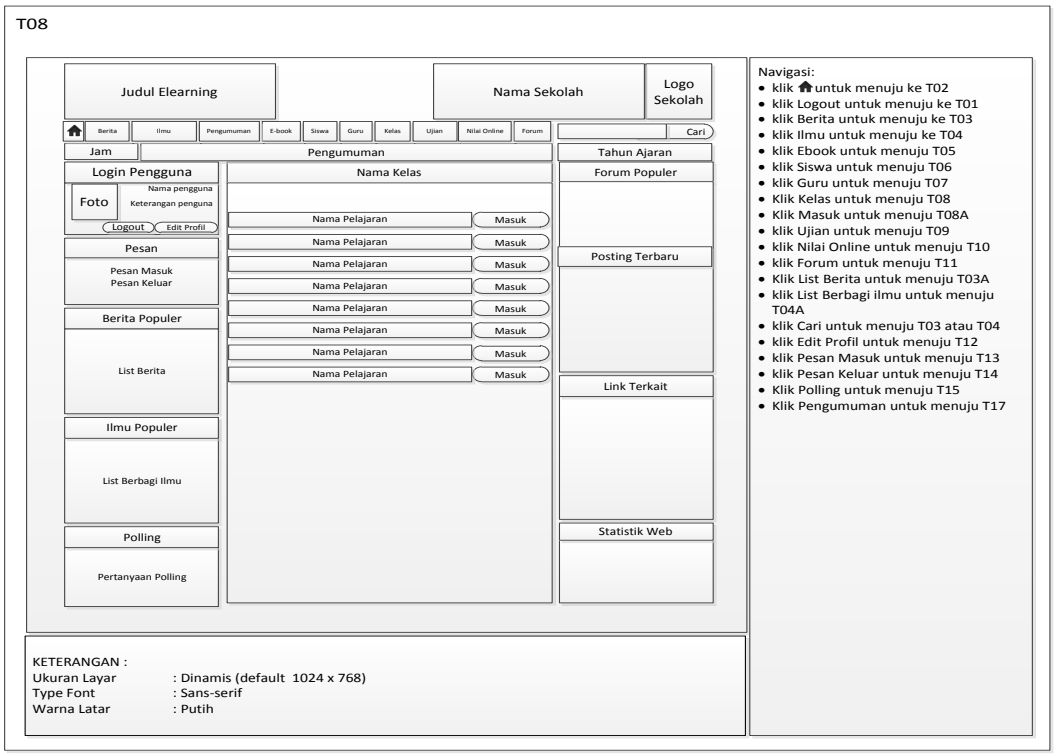
Gambar 3. 202 Tampilan Halaman Guru

13. Halaman Guru Profil



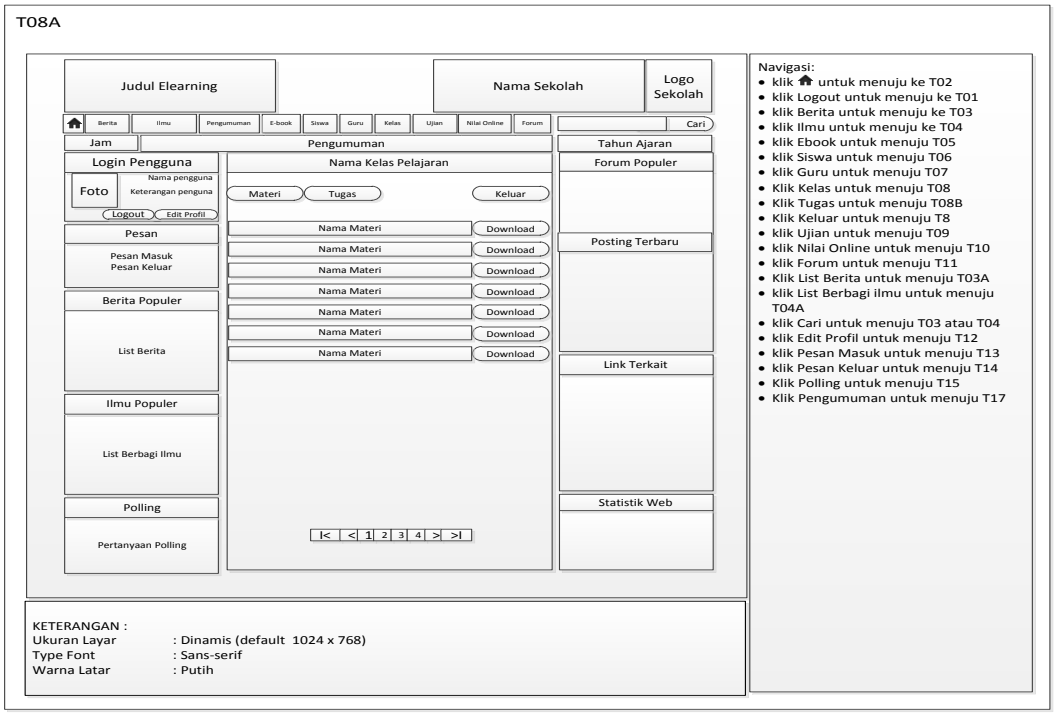
Gambar 3. 203 Tampilan Halaman Guru Profil

14. Halaman Kelas



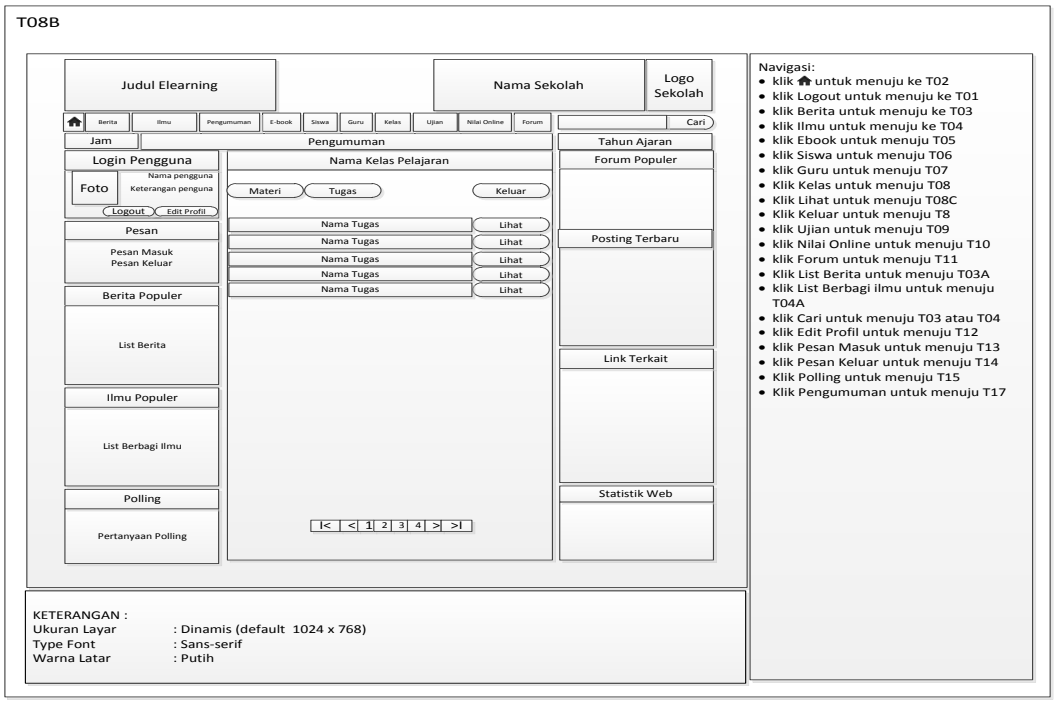
Gambar 3. 204 Tampilan Halaman Kelas

15. Halaman Kelas Lihat Materi



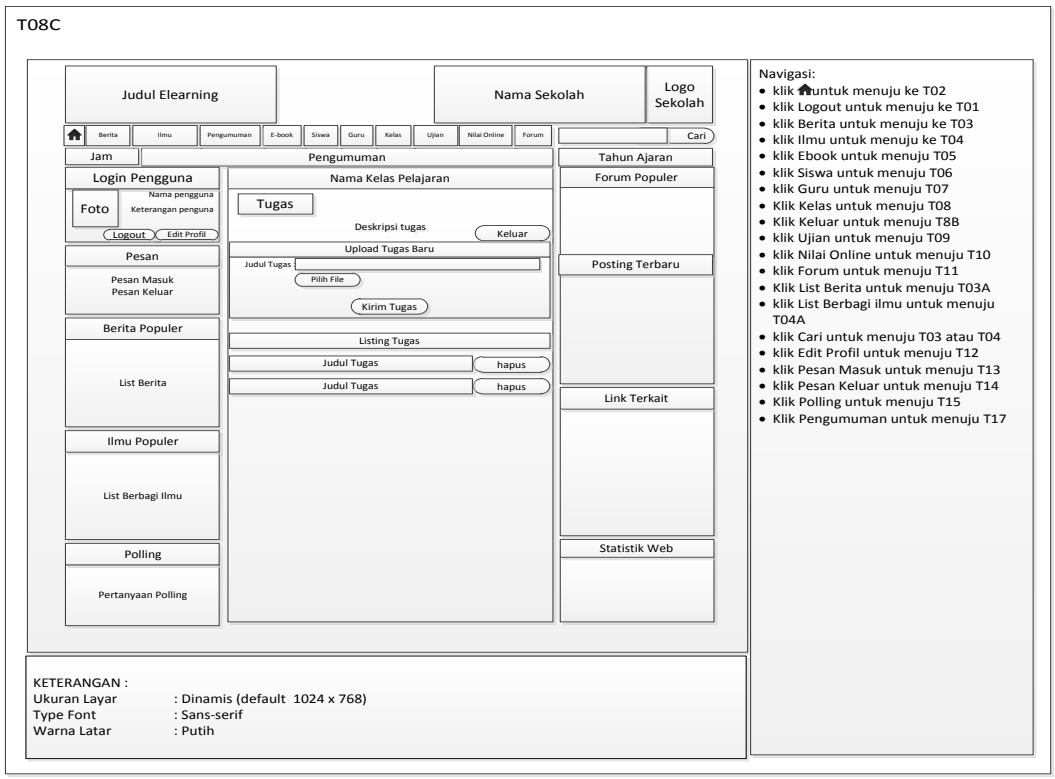
Gambar 3. 205 Tampilan Halaman Kelas Lihat Materi

16. Halaman Kelas Lihat Tugas



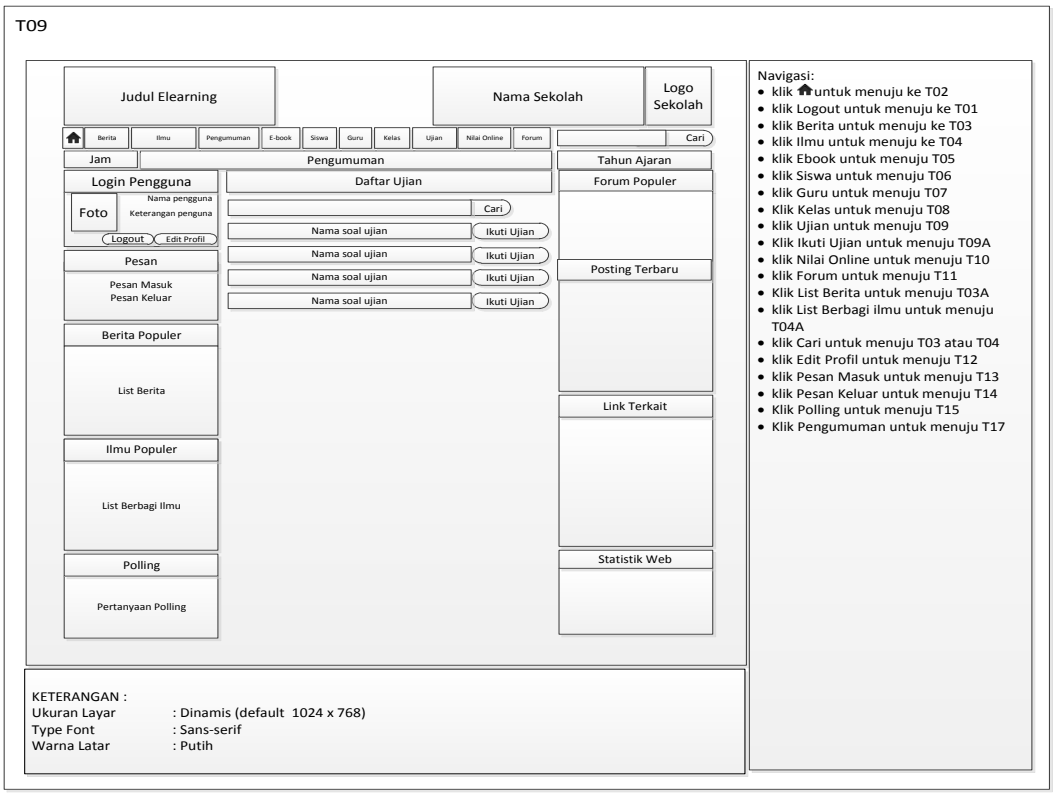
Gambar 3. 206 Tampilan Halaman Kelas Lihat Tugas

17. Halaman Kelas Kirim Tugas



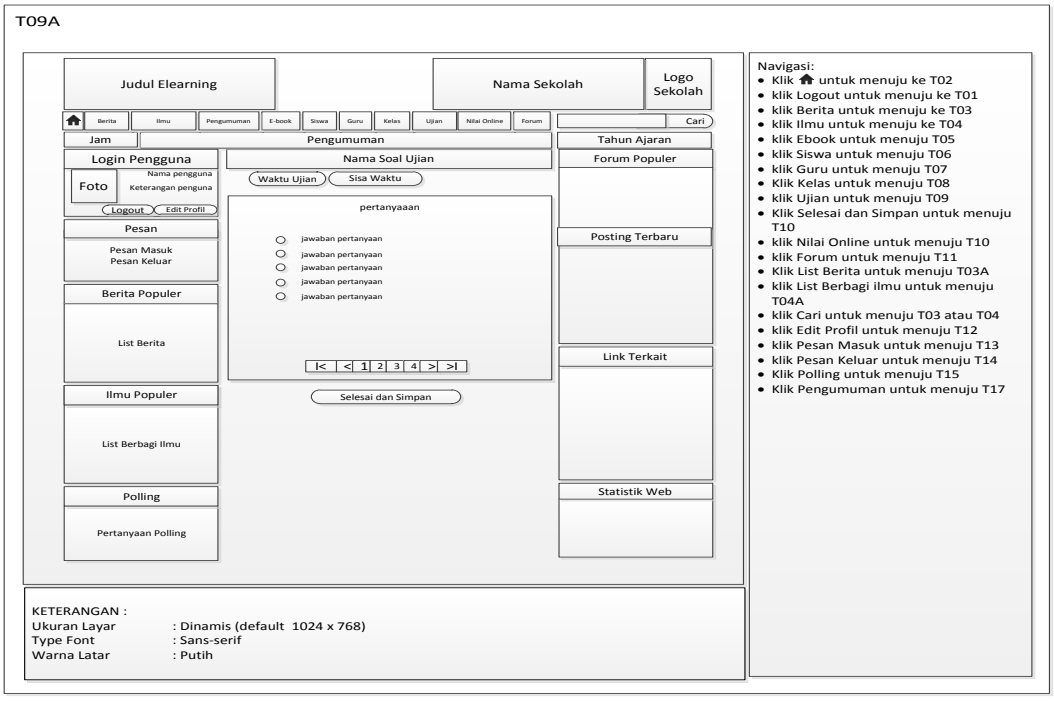
Gambar 3. 207 Tampilan Halaman Kelas Kirim Tugas

18. Halaman Ujian



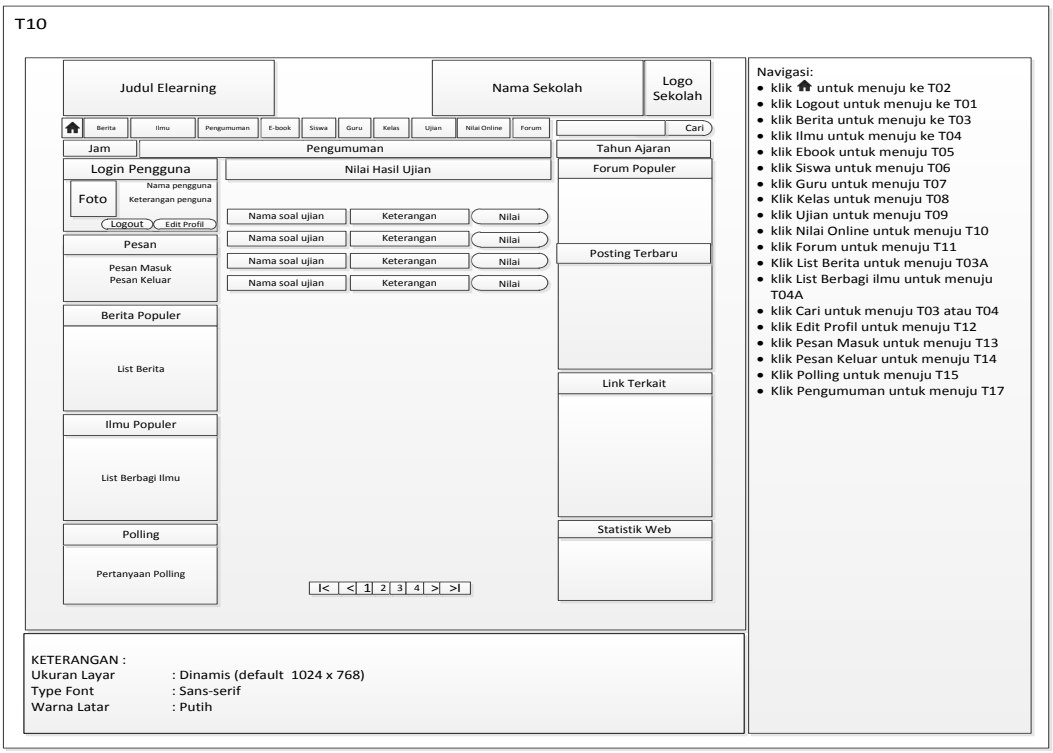
Gambar 3. 208 Tampilan Halaman Ujian

19. Halaman Soal Ujian



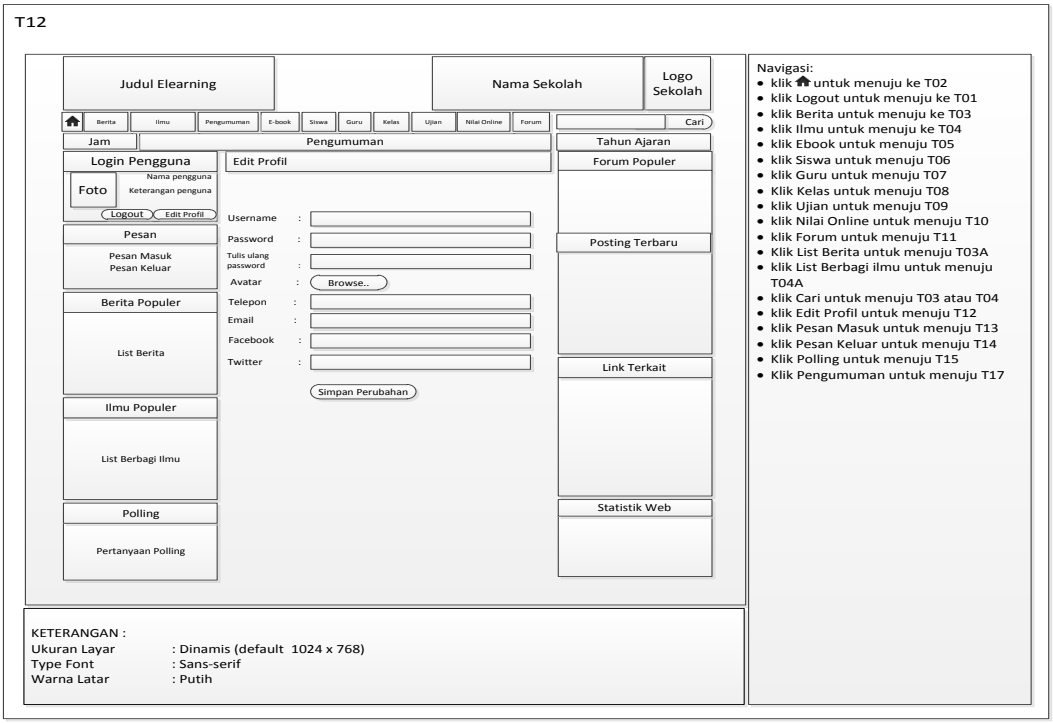
Gambar 3. 209 Tampilan Halaman Soal Ujian

20. Halaman Nilai Online



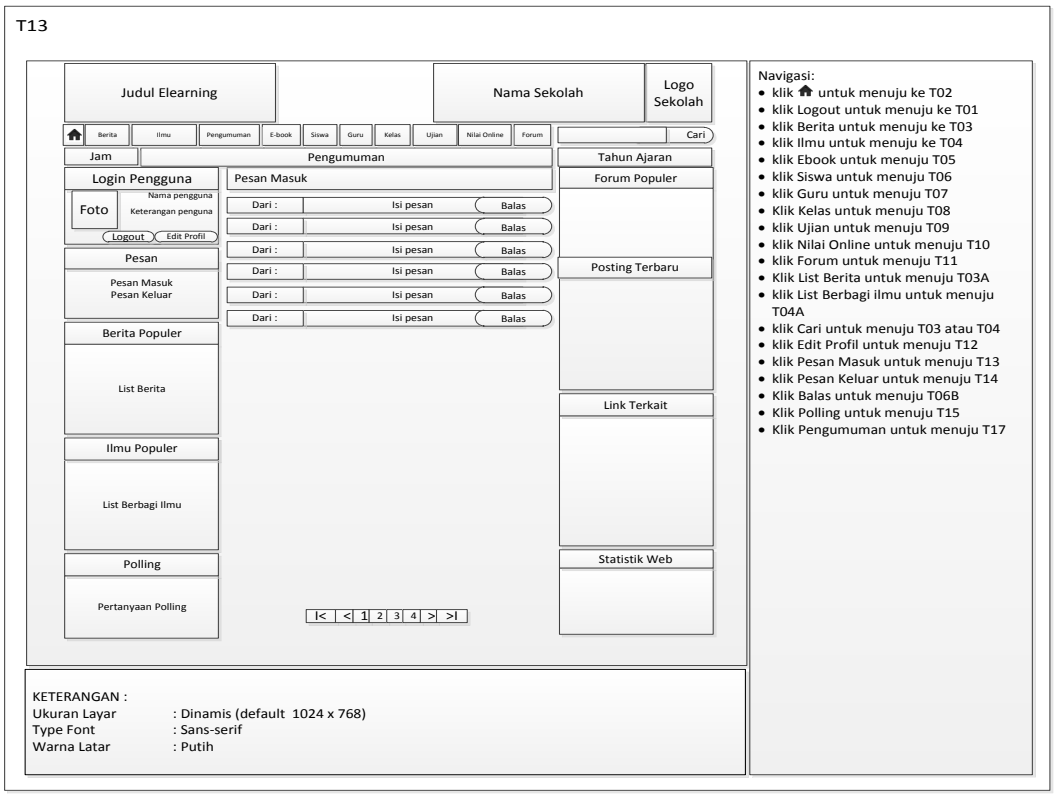
Gambar 3. 210 Tampilan Halaman Nilai Online

21. Halaman Edit Profil



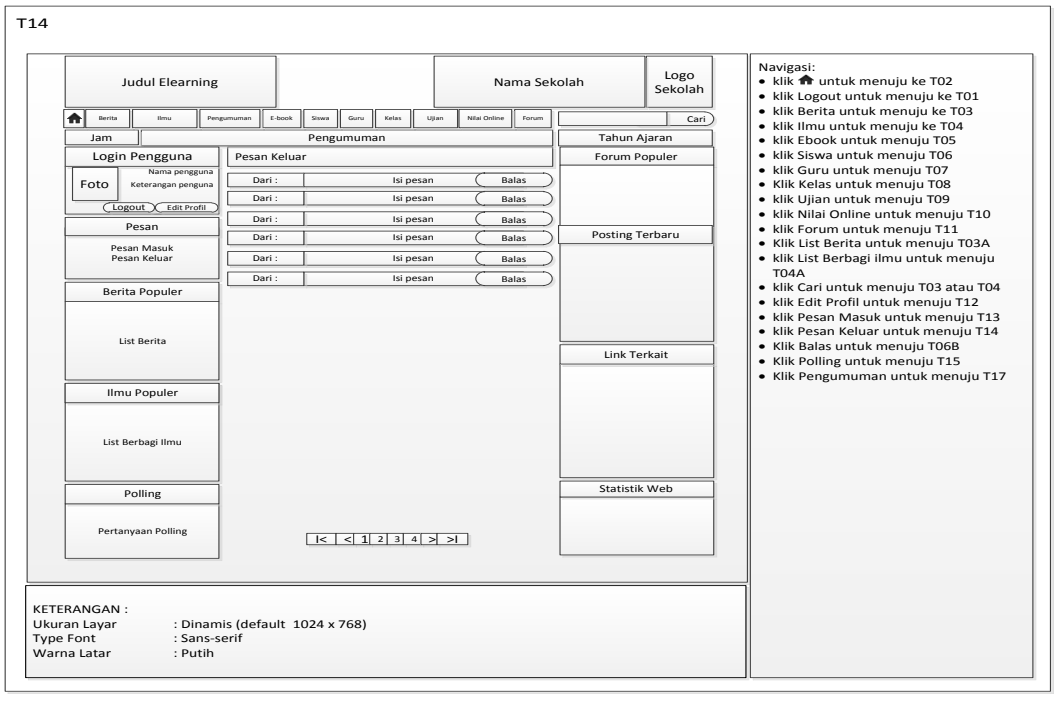
Gambar 3. 211 Tampilan Halaman Edit Profil

22. Halaman Pesan Masuk



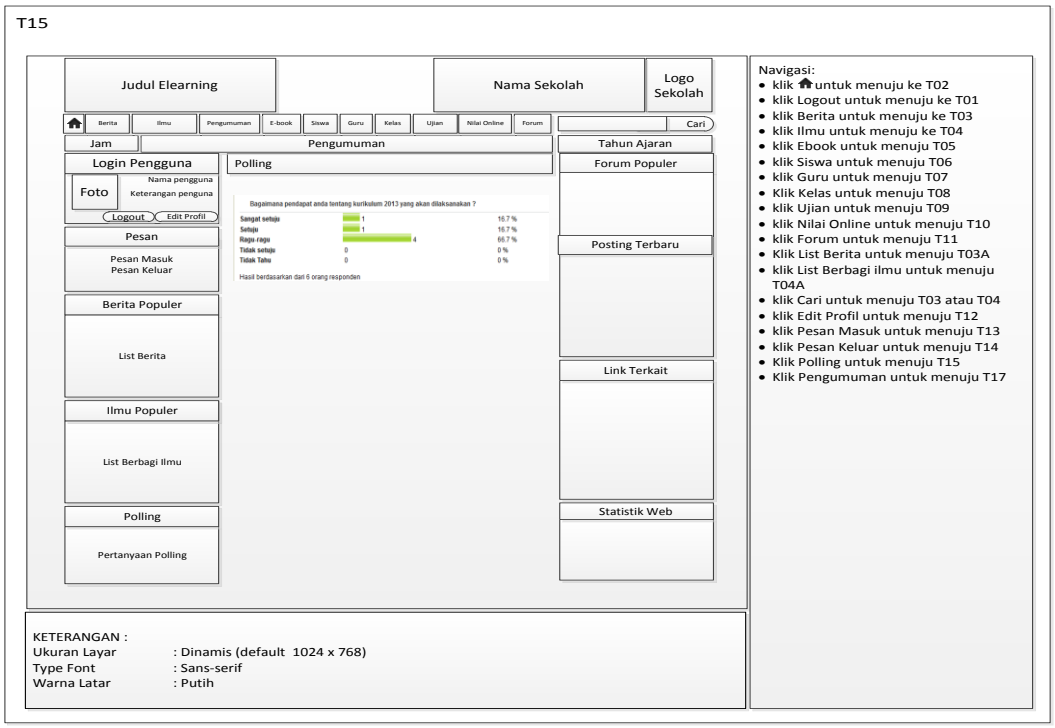
Gambar 3. 212 Tampilan Halaman Pesan Masuk

23. Halaman Pesan Masuk



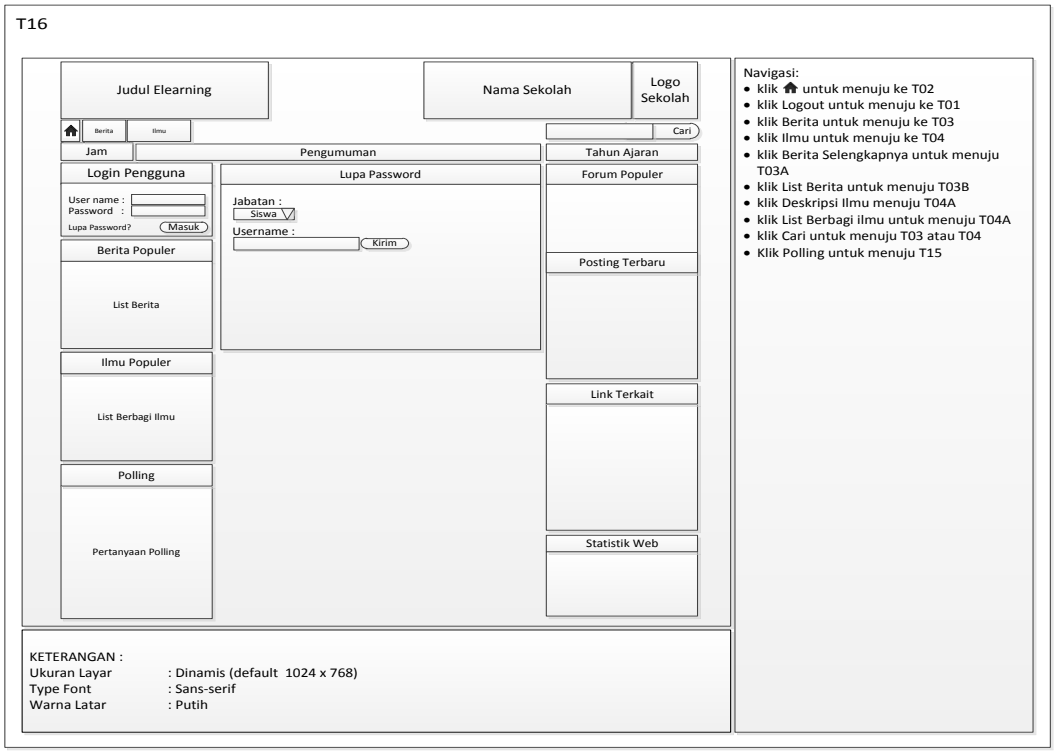
Gambar 3. 213 Tampilan Halaman Pesan Masuk

24. Halaman Polling



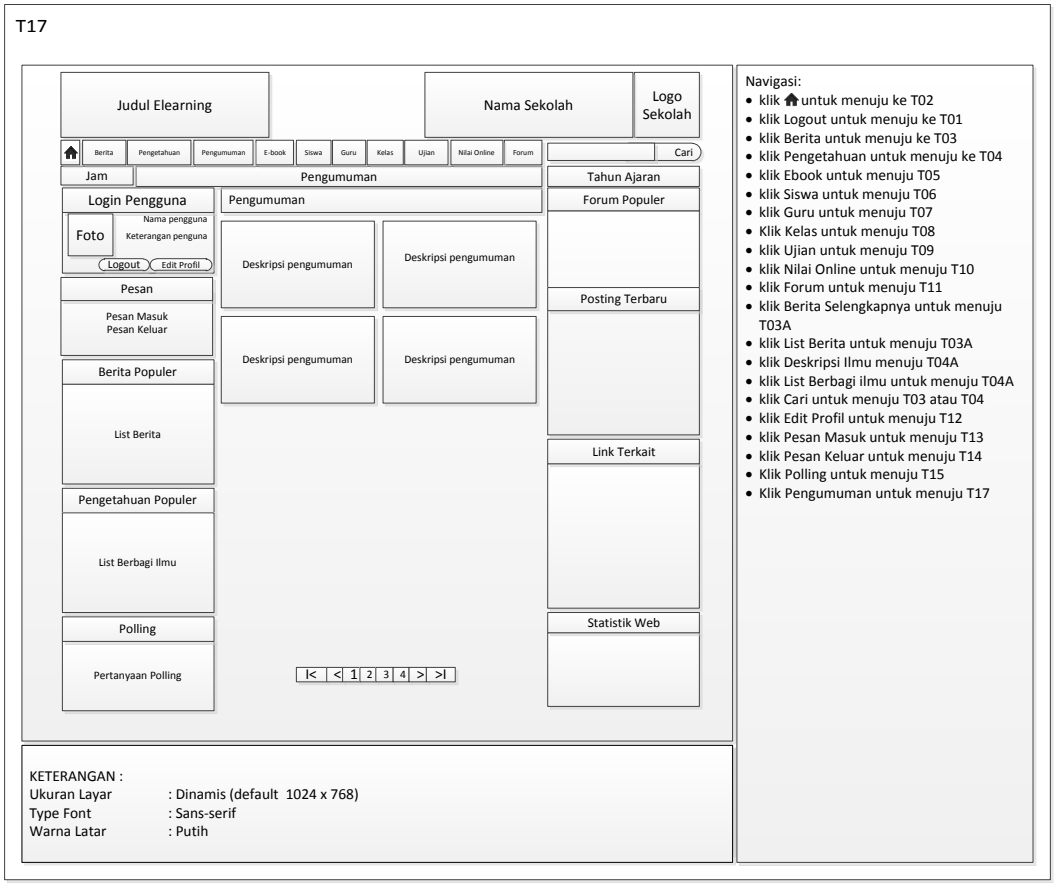
Gambar 3. 214 Tampilan Halaman Polling

25. Halaman Lupa Password Siswa



Gambar 3. 215 Tampilan Halaman Lupa Password Siswa

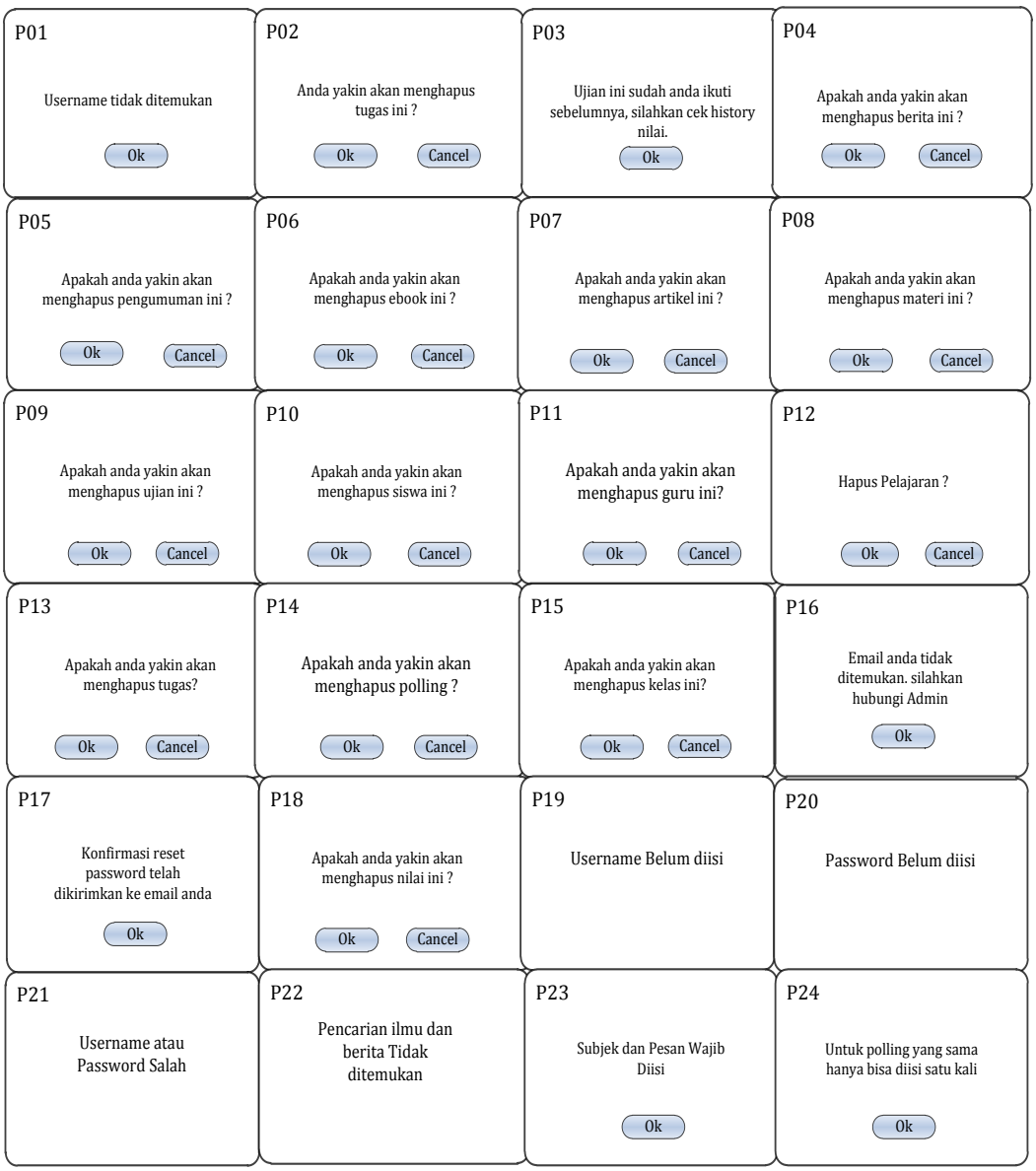
26. Halaman Pengumuman



Gambar 3. 216 Tampilan Halaman Pengumuman

3.2.5 Perancangan Pesan

Pada gambar dibawah ini merupakan perancangan pesan yang terdapat pada aplikasi *elearning* SMA NEGERI 3 Sumedang. Adapun tampilan pesan yang ada dapat dilihat pada Gambar 3.215.



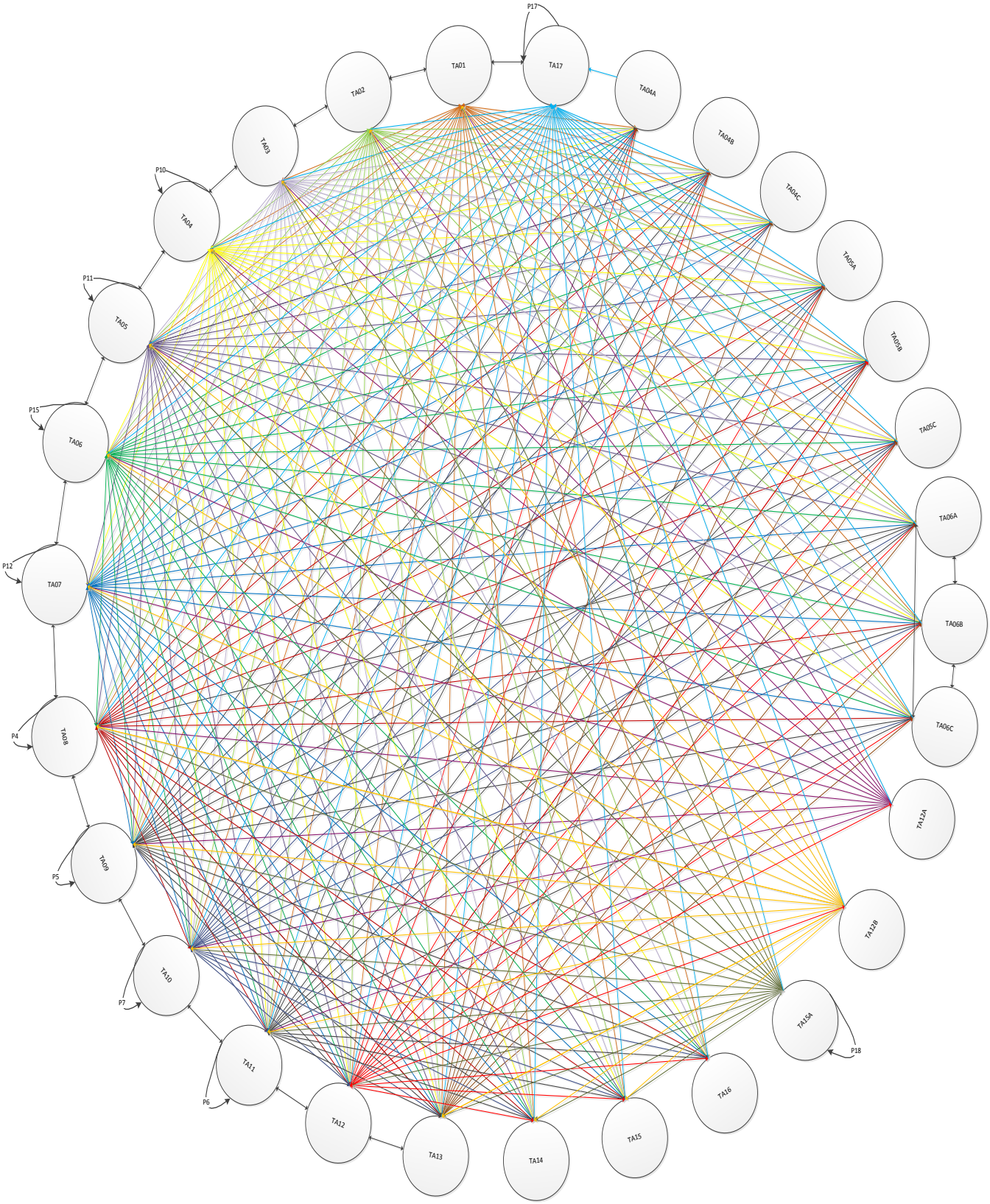
Gambar 3. 217 Perancangan Pesan

3.2.6 Jaringan Semantik

Jaringan semantik digunakan untuk menghubungkan setiap tampilan dan pesan pada sebuah aplikasi yang dirancang. Jaringan semantik sistem yang akan dibangun dibedakan sesuai jenis penggunanya admin, guru, dan siswa.

1. Jaringan Semantik Admin

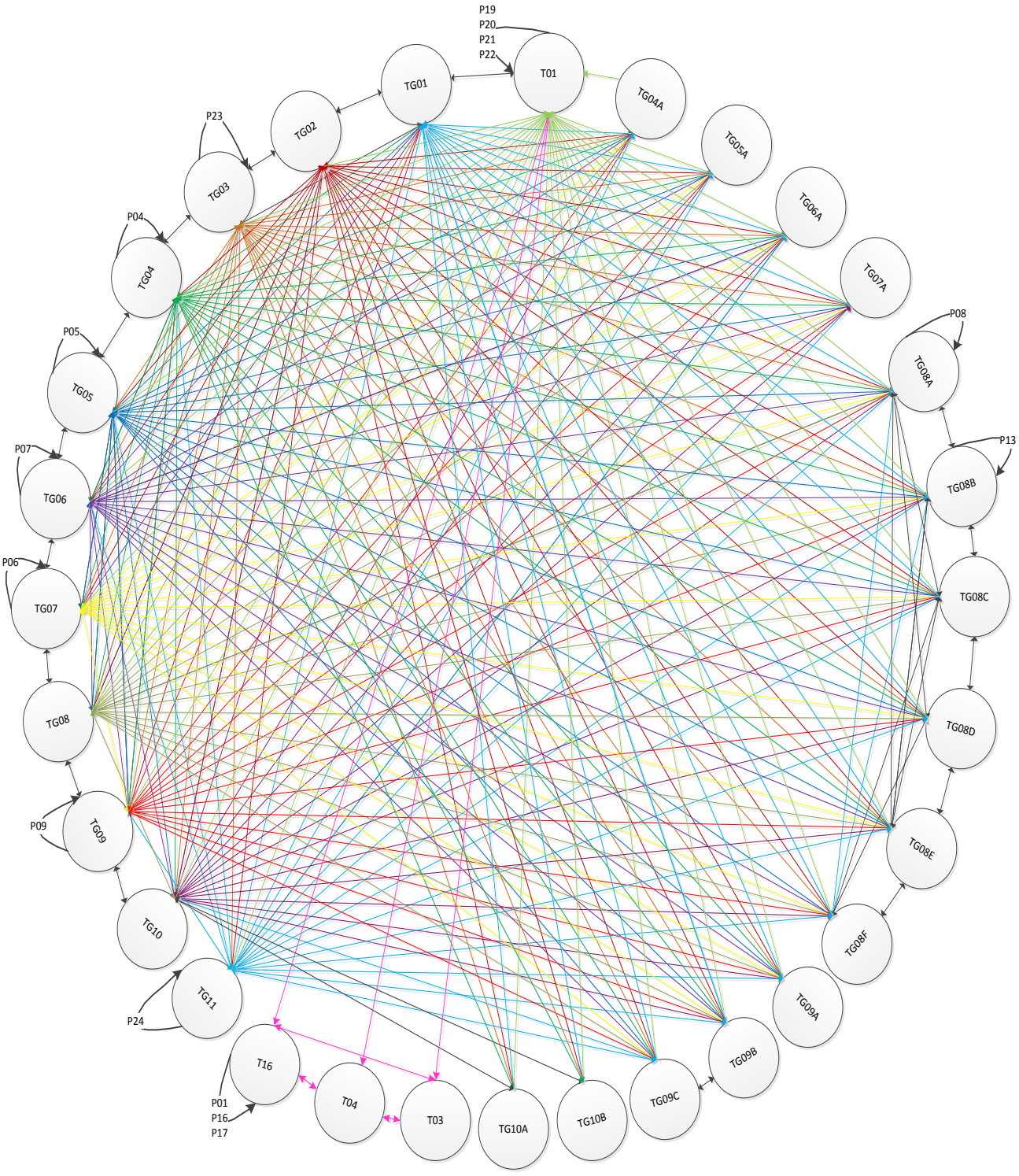
   Berikut dapat dilihat jaringan semantik admin dari aplikasi *e-learning* di SMA Negeri 3 Sumedang.



Gambar 3. 218 Jaringan Semantik Admin

2. Jaringan Semantik Guru

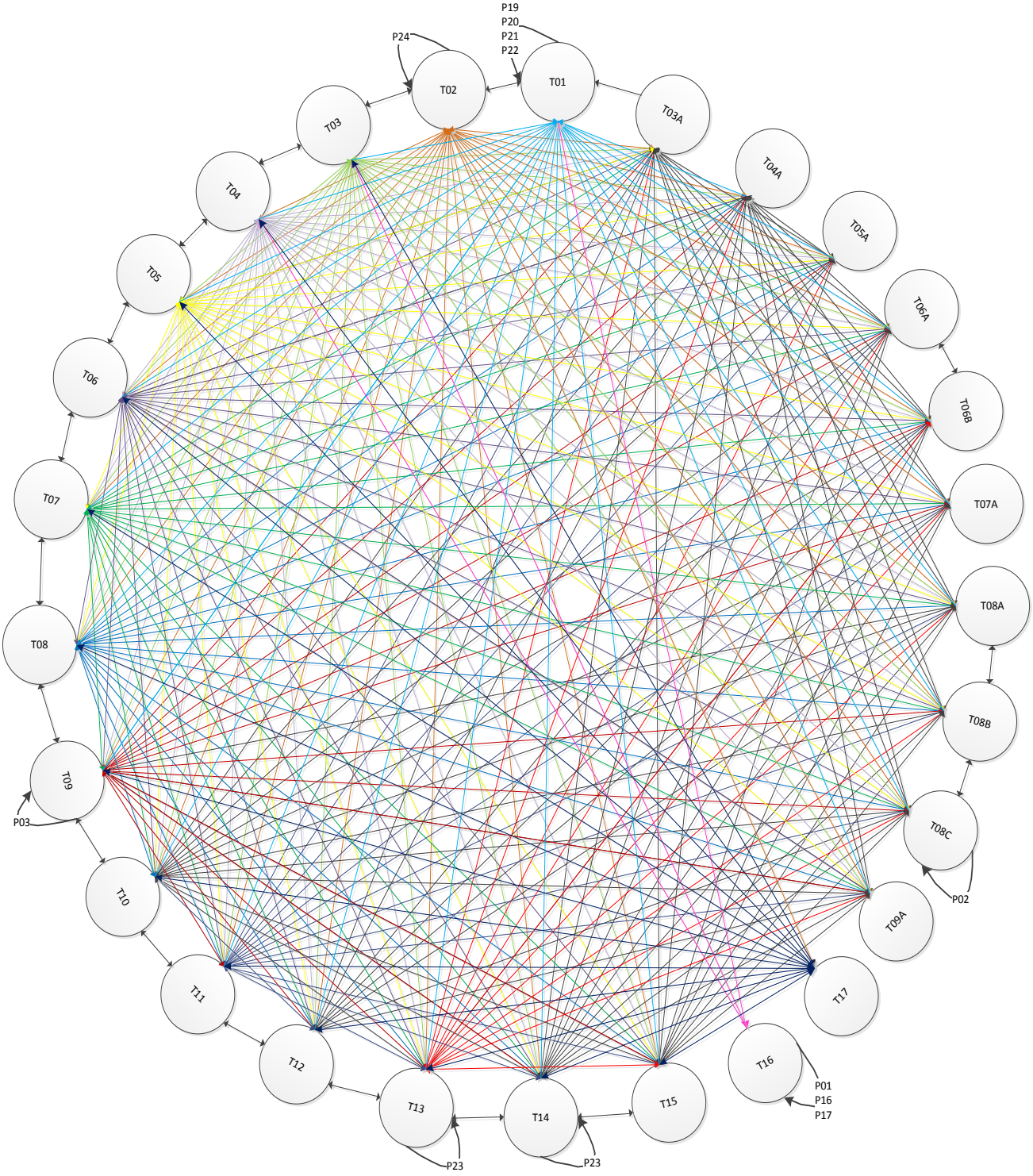
   Berikut dapat dilihat jaringan semantik guru dari aplikasi *e-learning* di SMA Negeri 3 Sumedang.



Gambar 3. 219 Jaringan Semantik Guru

3. Jaringan Semantik Siswa

Berikut dapat dilihat jaringan semantik siswa dari aplikasi *e-learning* di SMA Negeri 3 Sumedang.

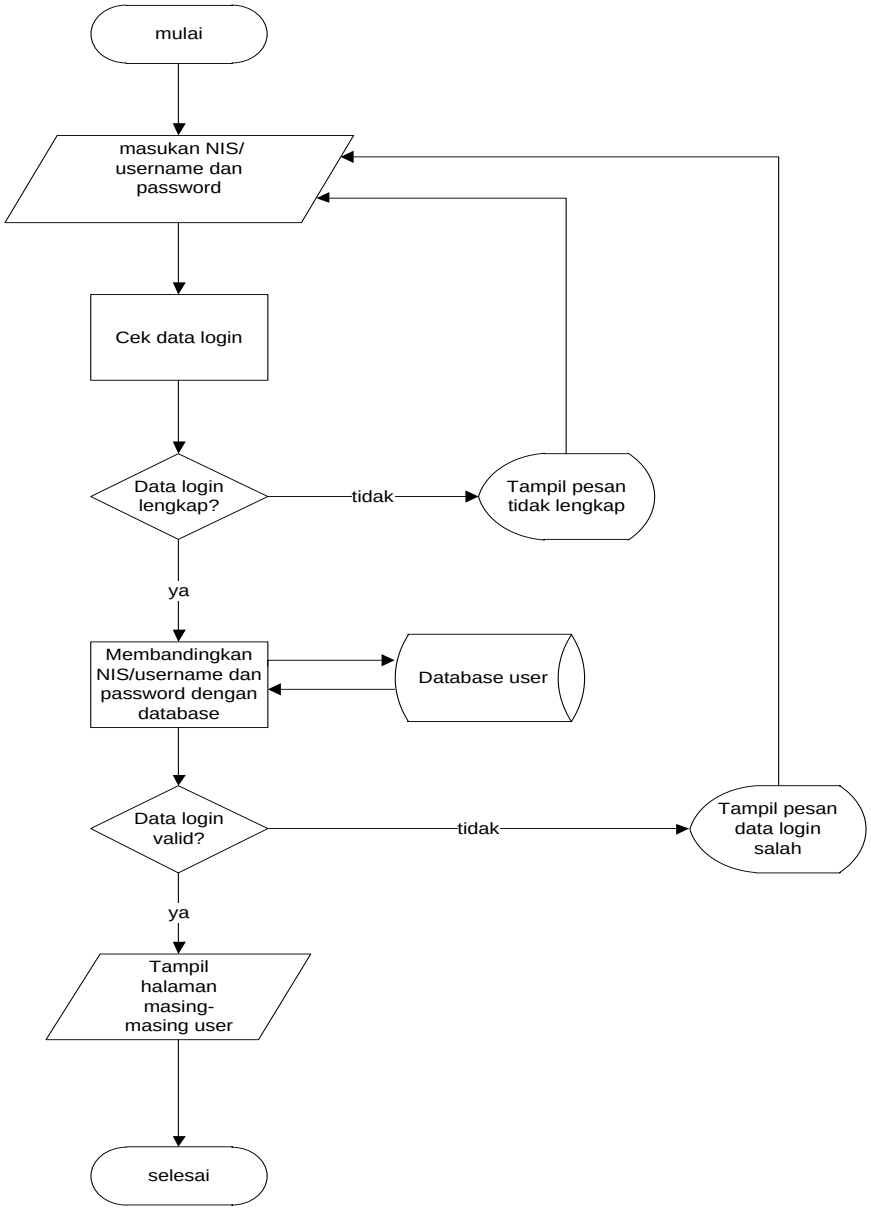


Gambar 3. 220 Jaringan Semantik Siswa

3.2.7 Perancangan Prosedural

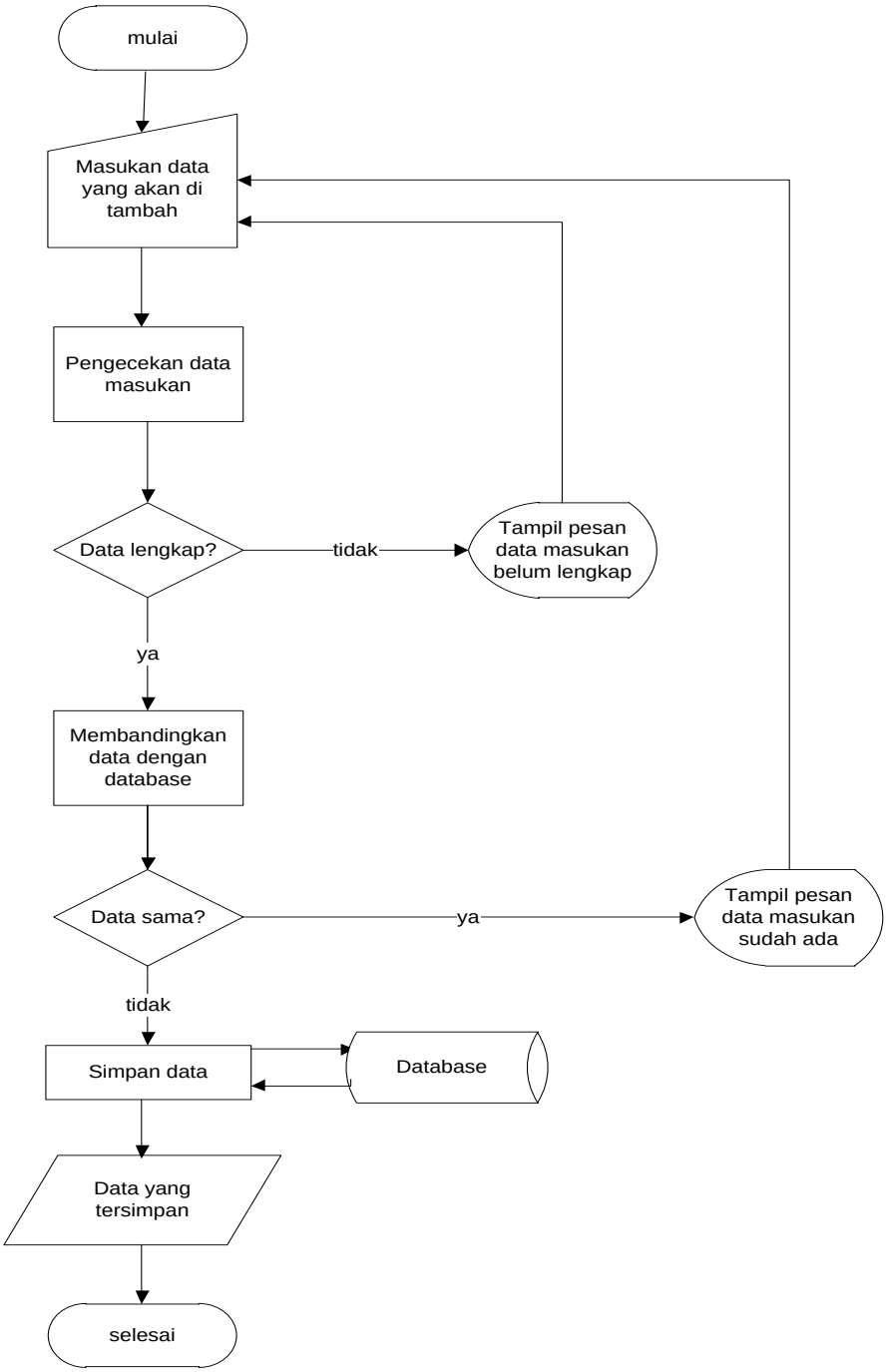
Perancangan prosedural mentransformasi elemen-elemen struktural dari arsitektur program ke dalam suatu deskripsi prosedural dari komponen-komponen perangkat lunak. Adapun perancangan prosedural untuk aplikasi yang akan dibangun adalah sebagai berikut:

1. Prosedural login, merupakan prosedur yang terjadi ketika user akan mengakses aplikasi. Prosedural login dapat dilihat pada Gambar 3. 221.



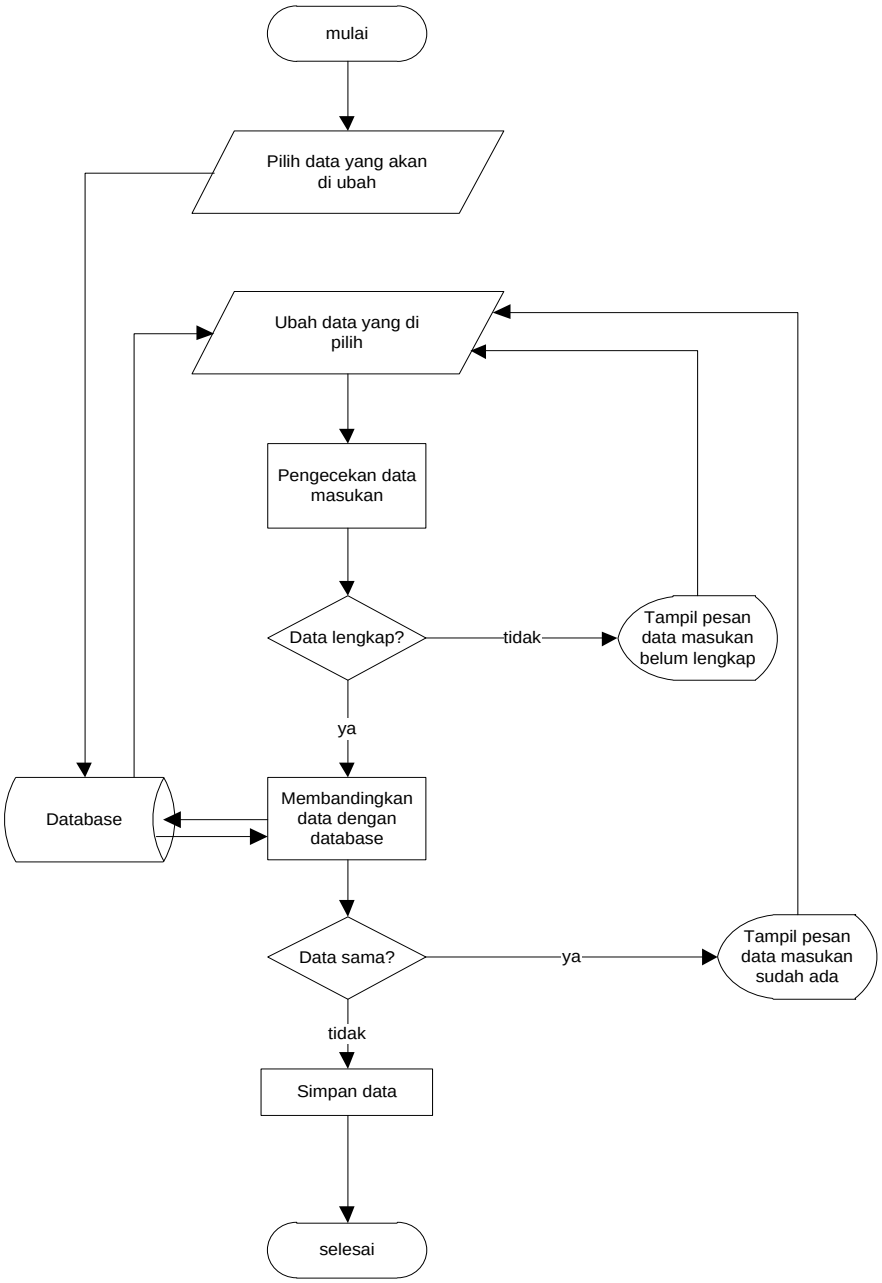
Gambar 3. 221 Prosedural Login

2. Prosedural tambah data, prosedur ini dilakukan ketika pengguna akan melakukan penambahan data. Prosedural tambah data dapat dilihat pada Gambar 3. 222.



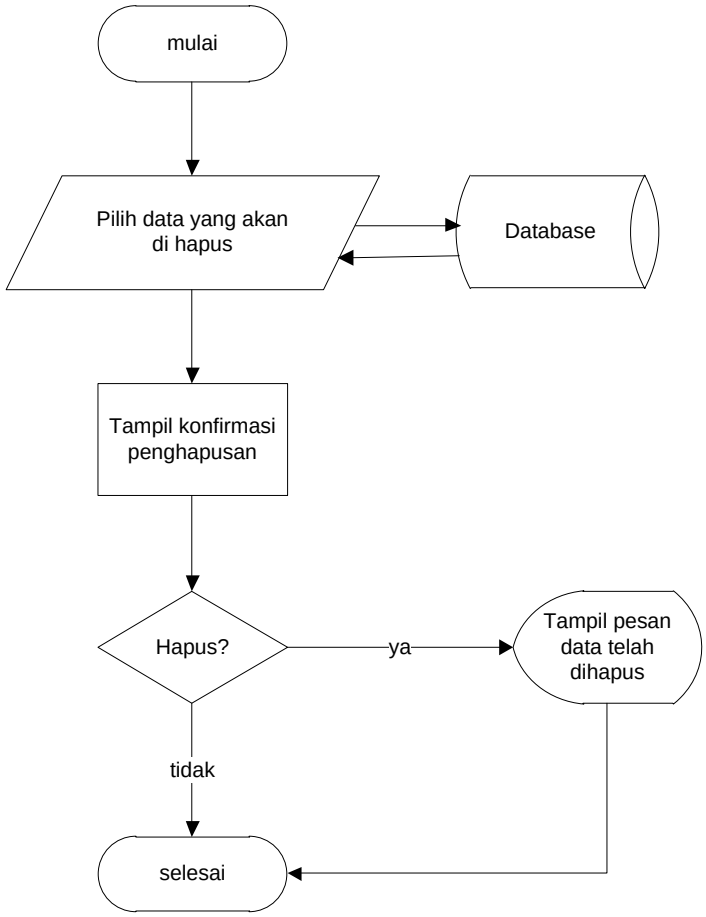
Gambar 3. 222 Prosedural Tambah Data

3. Prosedural ubah data, prosedur ini dilakukan ketika pengguna akan melakukan perubahan data. Prosedural ubah dapat dilihat pada Gambar 3.223.



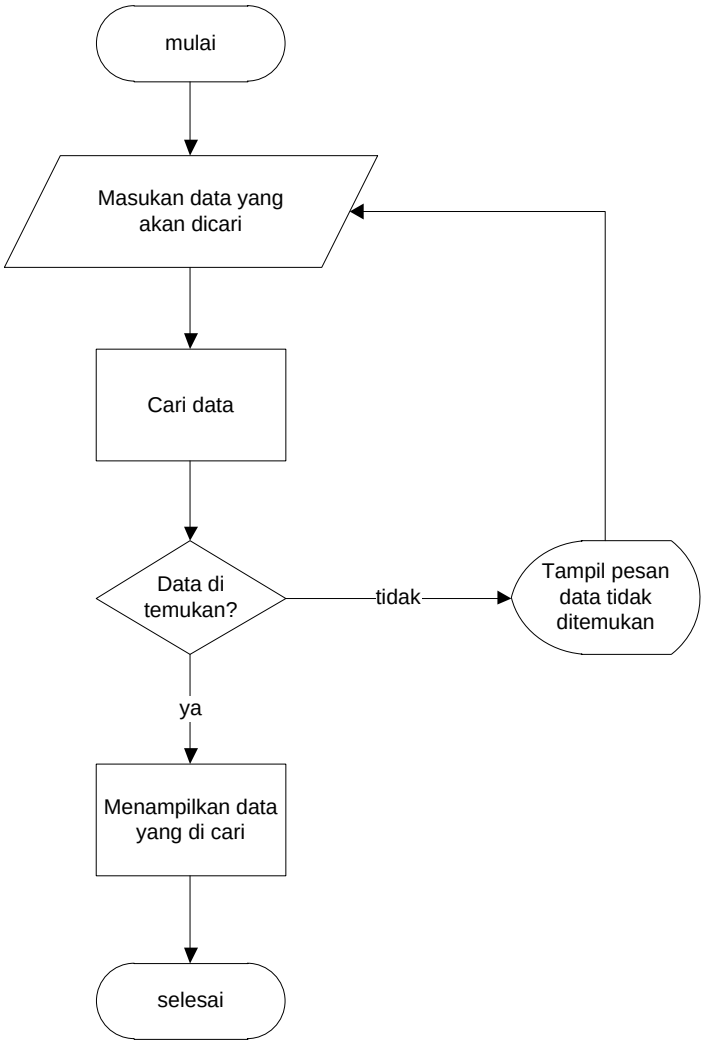
Gambar 3. 223 Prosedural Ubah Data

4. Prosedural hapus data, prosedur ini dilakukan ketika *user* akan melakukan penghapusan data. Prosedural hapus data dapat dilihat pada Gambar 3.224.



Gambar 3. 224 Prosedural Hapus Data

5. Prosedural cari, prosedural ini dilakukan ketika *user* akan melakukan pencarian data. Prosedural cari dapat dilihat pada Gambar 3.225.



Gambar 3. 225 Prosedural Cari