# BAB III

# ANALISIS DAN RANCANGAN SISTEM

* 1. **Perencanaan Sistem**

Perencanaan sistem adalah proses membuat sebuah laporan perencanaan sistem yang menggunakan sumber sistem informasi yang digunakan untuk membuat aplikasi *e-learning*. Hal-hal yang diperlukan yaitu membuat aplikasi *e-learning*, seperti menganalisa alur proses sistem yang akan berjalan dengan sistem baru dan bagaimana pengaruhnya, sehingga dapat berfungsi secara maksimal**.**

Membuat perencanaan harus menentukan batasan ruang lingkup sistem yang akan dibangun. Penulis mengumpulkan data, menentukan masalah dan memilih narasumber untuk mendiskusikan kebutuhan sistem dalam pembuatan sebuah aplikasi *e-learning* yang berada pada SMK Pelita Nusantara 1 Semarang.

* 1. **Analisa Sistem Yang Sedang Berjalan**

Setelah penulis melakukan wawancara dan pengamatan sistem yang sedang berjalan pada SMK Pelita Nusantara 1 Semarang, proses pembelajaran masih menerapkan sistem manual yang berarti proses belajar mengajar masih terikat oleh waktu mata pelajaran yang tersedia dan harus bertatap muka dengan guru pengajar.

* 1. **Analisa Sistem Yang Akan Dikembangkan**

Perkembangan sistem tidak lepas dari kebutuhan yang ada pada suatu organisasi ataupun perusahaan. Adapun hal yang mendorong perkembangan dari Aplikasi *E-Learning* yaitu:

1. Dengan adanya aplikasi *e-learning*, kegiatan pembelajaran memudahkan antara guru dengan siswa untuk pembelajaran secara *online* tanpa bertatap muka secara langsung.
2. Dengan adanya aplikasi *e-learning*, siswa dapat mendownload materi yang diberikan oleh guru, siswa dapat mengerjakan soal ujian maupun tugas yang diberikan oleh guru.
   1. **Analisa Kebutuhan Sistem**

Analisa kebutuhan sistem merupakan hal yang penting untuk mengetahui kebutuhan-kebutuhan yang nantinya akan digunakan untuk mendukung proses pembuatan sistem baru.

* 1. **Perancangan Sistem**

Perancangan sistem yang disusun bertujuan untuk menggambarkan secara jelas proses – proses sistem aplikasi yang dibentuk dengan metode pendekatan yang digunakan menggunakan *object oriented* maka model yang dipakai adalah *Unfield Modeling Language.*

* + 1. **Analisa Kebutuhan User**

Dalam sistem ini penulis menganalisa siapa saja yang akan menggunakan sistem dan apa yang dilakukan dalam sistem ini. Sistem ini akan digunakan oleh:

1. *Administrator* atau bisa disebut admin, yaitu seseorang yangmemiliki hak akses penuh dalam penggunaan aplikasi *e-learning*, selain itu mampu mengoperasikan komputer. Meliputi: mengelola data siswa, data guru, data mapel, data materi, jurusan, kelas, soal, hasil ujian, dan profil sekolah.
2. Guru, yaitu seseorang yang memiliki hak akses yang hanya diberikan kepada guru. Meliputi: data materi, soal, ujian, dan hasil ujian.
3. Siswa, yaitu seseorang yang memiliki hak akses yang hanya diberikan kepada siswa, selain itu mampu mengoperasikan komputer. Meliputi: data materi, data tugas, dan ujian.
   * 1. **Analisa Kebutuhan Hardware**

Dalam pembuatan aplikasi *e-learning* penulis menggunakan beberapa perangkat keras, yaitu :

1. Laptop *Processor Core i3*
2. *RAM* minimum 2GB
3. *Keyboard*
4. *Mouse*
   * 1. **Analisa Kebutuhan Software**

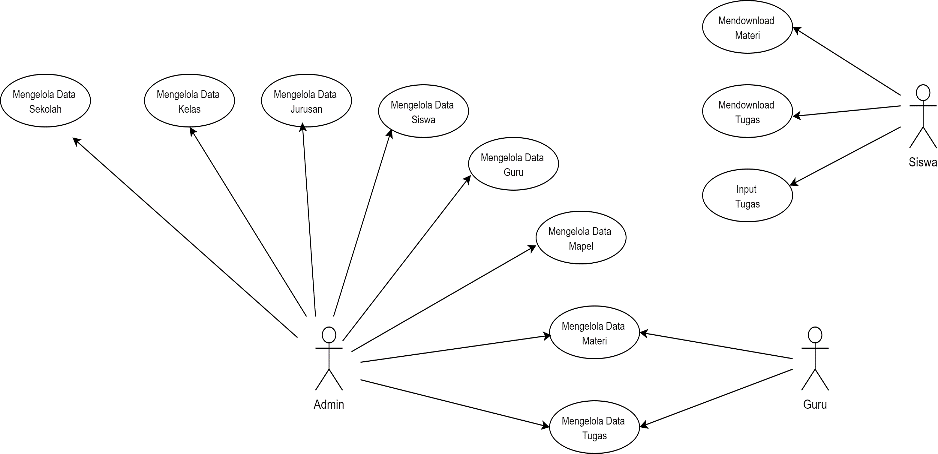
Beberapa *software* yang digunakan penulis untuk mendukung pembuatan sistem, yaitu :

1. Sistem Operasi Windows 10
2. Bahasa Pemrograman *PHP*
3. Vscode untuk penulisan kode program
4. *Framework CI*
5. *Database MySQL*
6. *Rational Rose Enterprise* untuk desain UML
7. Figma untuk desain *interface*.
   * 1. **Analisa Perancangan Sistem**

Dalam merancang Aplikasi *E-Learning*, penulis menggunakan alat bantu perancangan sistem yaitu menggunakan UML (*Unified Modelling* *System*). Di dalam UML, terdapat gambaran sistem berupa diagramdiantaranya *use case diagram*, *class diagram*, *activity diagram*, dan *sequence* *diagram.*

* + 1. **Use Case Diagram**

*Use case diagram* merupakan gambaran dari kegiatan yangdilakukan oleh sistem. Use case diagram berinteraksi antara satu aktor atau lebih. *Use case diagram* dari Aplikasi *E-Learning* dapat di lihat pada gambar 3.1.



Gambar 3.1 *Use Case Diagram.*

* + 1. **Skenario Sistem**

Detail dari masing-masing *use case* diatas dapat dilihat pada uraian berikut :

1. Skenario *Use Case* Mengelola Data Siswa

Pada Tabel 3.1 mencatat skenario use case untuk mengelola data siswa.

Tabel 3.1. Tabel Skenario *Use Case* Mengelola Data Siswa

|  |  |  |
| --- | --- | --- |
| Nama *Use Case* | Mengelola Data Siswa | |
| Aktor | Admin | |
| Tujuan | *Input*, edit, hapus data siswa | |
| Deskripsi | *Use Case* ini merupakan proses untuk melakukan pengelolaan data siswa. Dalam proses dapat dilakukan proses simpan, edit, dan hapus data. | |
| Prakondisi | Admin telah login | |
| Bidang Khas Event : | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih form data siswa. | Langkah 2 :  Sistem merespon dengan menampilkan menu data. |
|  | Langkah 3 :  Admin menginputkan data siswa. |  |
|  | Langkah 4 :  Simpan data. | Langkah 5 :  Sistem menyimpan data tersebut. |
| Bidang Alternatif 1 | Langkah 1.1 :  Admin memilih data siswa. | Langklah 1.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 1.3 :  Admin merubah data siswa. | Langkah 1.4 :  Sistem mengupdate data tersebut. |
| Bidang Alternatif 2 | Langkah 2.1 :  Admin memilih data siswa. | Langklah 2.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 2.3 :  Admin menghapus data siswa. | Langkah 2.4 :  Sistem menghapus data tersebut. |
| Kesimpulan | Untuk merubah data yang ada dalam form data siswa dibutuhkan administrator untuk mengelola data tersebut. | |
| Post Kondisi | Perubahan data siswa dalam *database.* | |

1. Skenario *Use Case* Mengelola Data Guru

Pada Tabel 3.2 mencatat skenario use case untuk mengelola data guru.

Tabel 3.2. Tabel Skenario *Use Case* Mengelola Data Guru

|  |  |  |
| --- | --- | --- |
| Nama *Use Case* | Mengelola Data Guru | |
| Aktor | Admin | |
| Tujuan | *Input,* edit, hapus data guru | |
| Deskripsi | *Use Case* ini merupakan proses untuk melakukan pengelolaan data guru. Dalam proses dapat dilakukan proses simpan, edit, dan hapus data. | |
| Prakondisi | Admin telah login. | |
| Bidang Khas Event : | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih form data guru. | Langkah 2 :  Sistem merespon dengan menampilkan menu data. |
|  | Langkah 3 :  Admin menginputkan data guru. |  |
|  | Langkah 4 :  Simpan data. | Langkah 5 :  Sistem menyimpan data tersebut. |
| Bidang Alternatif 1 | Langkah 1.1 :  Admin memilih data guru. | Langklah 1.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 1.3 :  Admin merubah data guru. | Langkah 1.4 :  Sistem mengupdate data tersebut. |
| Bidang Alternatif 2 | Langkah 2.1 :  Admin memilih data guru. | Langklah 2.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 2.3 :  Admin menghapus data guru. | Langkah 2.4 :  Sistem menghapus data tersebut. |
| Kesimpulan | Untuk merubah data yang ada dalam form data guru dibutuhkan administrator untuk mengelola data tersebut. | |
| Post Kondisi | Perubahan data guru dalam *database.* | |

1. Skenario *Use Case* Mengelola Data Mapel

Pada Tabel 3.3 mencatat skenario use case untuk mengelola data mapel.

Tabel 3.3. Tabel Skenario *Use Case* Mengelola Data Mapel

|  |  |  |
| --- | --- | --- |
| Nama *Use Case* | Mengelola Data Mapel | |
| Aktor | Admin | |
| Tujuan | *Input*, edit, hapus data mapel | |
| Deskripsi | *Use Case* ini merupakan proses untuk melakukan pengelolaan data mapel. Dalam proses dapat dilakukan proses simpan, edit, dan hapus data. | |
| Prakondisi | Admin telah login | |
| Bidang Khas Event : | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin memilih form data mapel. | Langkah 2 :  Sistem merespon dengan menampilkan menu data. |
|  | Langkah 3 :  Admin menginputkan data mapel. |  |
|  | Langkah 4 :  Simpan data. | Langkah 5 :  Sistem menyimpan data tersebut. |
| Bidang Alternatif 1 | Langkah 1.1 :  Admin memilih data mapel. | Langklah 1.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 1.3 :  Admin merubah data mapel. | Langkah 1.4 :  Sistem mengupdate data tersebut. |
| Bidang Alternatif 2 | Langkah 2.1 :  Admin memilih data mapel. | Langklah 2.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 2.3 :  Admin menghapus data mapel. | Langkah 2.4 :  Sistem menghapus data tersebut. |
| Kesimpulan | Untuk merubah data yang ada dalam form data mapel dibutuhkan administrator untuk mengelola data tersebut. | |
| Post Kondisi | Perubahan data mapel dalam *database.* | |

1. Skenario *Use Case* Mengelola Data Materi

Pada Tabel 3.4 mencatat skenario use case untuk mengelola data materi.

Tabel 3.4. Tabel Skenario *Use Case* Mengelola Data Materi

|  |  |  |
| --- | --- | --- |
| Nama *Use Case* | Mengelola Data Materi | |
| Aktor | Admin/ Guru | |
| Tujuan | *Input*, edit, hapus data materi | |
| Deskripsi | *Use Case* ini merupakan proses untuk melakukan pengelolaan data materi. Dalam proses dapat dilakukan proses simpan, edit, dan hapus data. | |
| Prakondisi | Admin/ Guru telah login | |
| Bidang Khas Event : | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin/ Guru memilih form data materi. | Langkah 2 :  Sistem merespon dengan menampilkan menu data. |
|  | Langkah 3 :  Admin/ Guru menginputkan data materi. |  |
|  | Langkah 4 :  Simpan data. | Langkah 5 :  Sistem menyimpan data tersebut. |
| Bidang Alternatif 1 | Langkah 1.1 :  Admin/ Guru memilih data materi. | Langklah 1.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 1.3 :  Admin/ Guru merubah data materi. | Langkah 1.4 :  Sistem mengupdate data tersebut. |
| Bidang Alternatif 2 | Langkah 2.1 :  Admin/ Guru memilih data materi. | Langklah 2.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 2.3 :  Admin/ Guru menghapus data materi. | Langkah 2.4 :  Sistem menghapus data tersebut. |
| Kesimpulan | Untuk merubah data yang ada dalam form data materi dibutuhkan administrator untuk mengelola data tersebut. | |
| Post Kondisi | Perubahan data materi dalam *database.* | |

1. Skenario *Use Case* Mengelola Data Tugas

Pada Tabel 3.5 mencatat skenario use case untuk mengelola data tugas.

Tabel 3.5. Tabel Skenario *Use Case* Mengelola Data Tugas

|  |  |  |
| --- | --- | --- |
| Nama *Use Case* | Mengelola Data Tugas | |
| Aktor | Admin/ Guru | |
| Tujuan | *Input*, edit, hapus data tugas | |
| Deskripsi | *Use Case* ini merupakan proses untuk melakukan pengelolaan data soal. Dalam proses dapat dilakukan proses simpan, edit, dan hapus data. | |
| Prakondisi | Admin/ Guru telah login | |
| Bidang Khas Event : | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Admin/ Guru memilih form data tugas. | Langkah 2 :  Sistem merespon dengan menampilkan menu data. |
|  | Langkah 3 :  Admin/ Guru menginputkan data tugas. |  |
|  | Langkah 4 :  Simpan data. | Langkah 5 :  Sistem menyimpan data tersebut. |
| Bidang Alternatif 1 | Langkah 1.1 :  Admin/ Guru memilih data tugas. | Langklah 1.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 1.3 :  Admin/ Guru merubah data tugas. | Langkah 1.4 :  Sistem mengupdate data tersebut. |
| Bidang Alternatif 2 | Langkah 2.1 :  Admin/ Guru memilih data tugas. | Langklah 2.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 2.3 :  Admin/ Guru menghapus data tugas. | Langkah 2.4 :  Sistem menghapus data tersebut. |
| Bidang Alternatif 3 | Langkah 3.1 :  Admin/ Guru memilih data tugas. | Langklah 3.2 :  Sistem menampilkan data tersebut. |
|  | Langlah 3.3 :  Admin/ Guru mencetak data tugas. | Langkah 3.4 :  Sistem menampilkan data tersebut. |
| Kesimpulan | Untuk merubah data yang ada dalam form data tugas dibutuhkan administrator untuk mengelola data tersebut. | |
| Post Kondisi | Perubahan data soal dalam *database.* | |

1. Skenario *Use Case* Mengelola Data Materi

Pada Tabel 3.6 mencatat skenario use case untuk mengelola data materi.

Tabel 3.6. Tabel Skenario *Use Case* Download Materi

|  |  |  |
| --- | --- | --- |
| Nama *Use Case* | Download Materi | |
| Aktor | Siswa | |
| Tujuan | *Download* materi | |
| Deskripsi | *Use Case* ini merupakan proses untuk melakukan download materi yang telah diinputkan admin/ guru. | |
| Prakondisi | Siswa telah login | |
| Bidang Khas Event : | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Siswa memilih form data materi. | Langkah 2 :  Sistem merespon dengan menampilkan menu data. |
|  | Langkah 3 :  Siswa memilih data materi. |  |
|  | Langkah 4 :  Download data. | Langkah 5 :  Sistem mendownload data tersebut. |
| Kesimpulan | Untuk mendownload data materi, admin/ guru sudah mengupload data. | |
| Post Kondisi | Materi berhasil didownload*.* | |

1. Skenario *Use Case* Ujian

Pada Tabel 3.7 mencatat skenario use case untuk mengelola data materi.

Tabel 3.7. Tabel Skenario *Use Case*Ujian

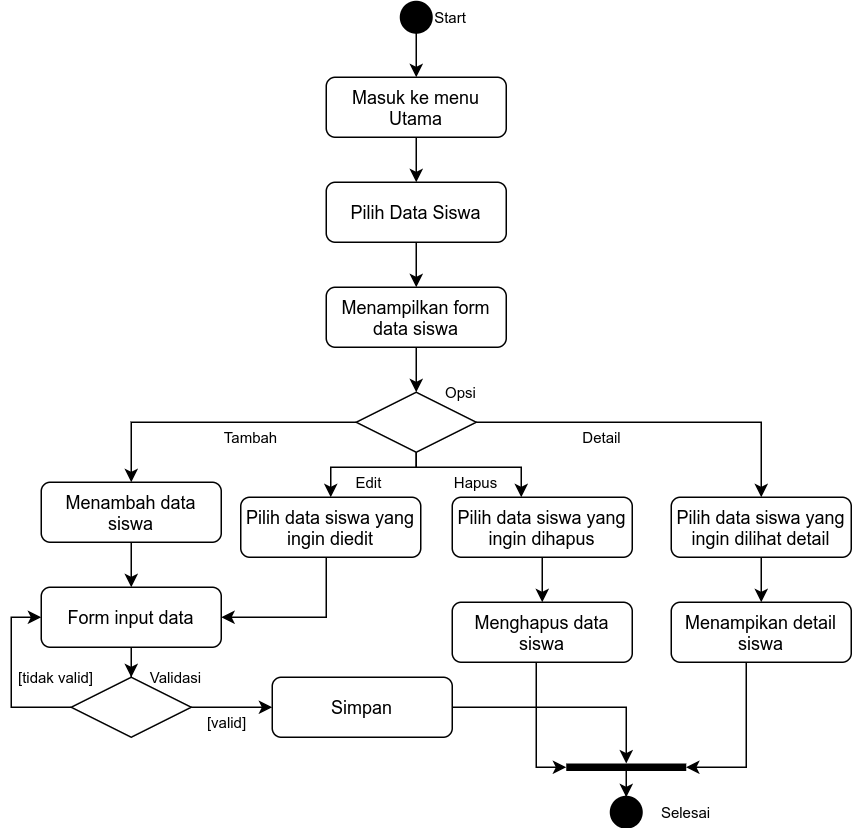
|  |  |  |
| --- | --- | --- |
| Nama *Use Case* | Ujian | |
| Aktor | Siswa | |
| Tujuan | Mengikuti Ujian | |
| Deskripsi | *Use Case* ini merupakan proses untuk melakukan ujian yang telah diinputkan admin/ guru. | |
| Prakondisi | Siswa telah login | |
| Bidang Khas Event : | Kegiatan Aktor | Respon Sistem |
|  | Langkah 1 :  Siswa memilih form data ujian. | Langkah 2 :  Sistem merespon dengan menampilkan menu data. |
|  | Langkah 3 :  Siswa memilih data ujian. |  |
|  | Langkah 4 :  Melakukan ujian. | Langkah 5 :  Selesai. |
| Kesimpulan | Untuk mendownload data ujian, admin/ guru sudah mengupload data. | |
| Post Kondisi | Soal Ujian berhasil didownload*.* | |

* + 1. **Activity Diagram**

*Activity Diagram* menggambarkan aliran akitivitas dalam sistem yangsedang dirancang, bagaimana masing-masing aliran berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. Berikut gambar *activity* *diagram:*

1. *Activity Diagram* Mengelola Data Siswa

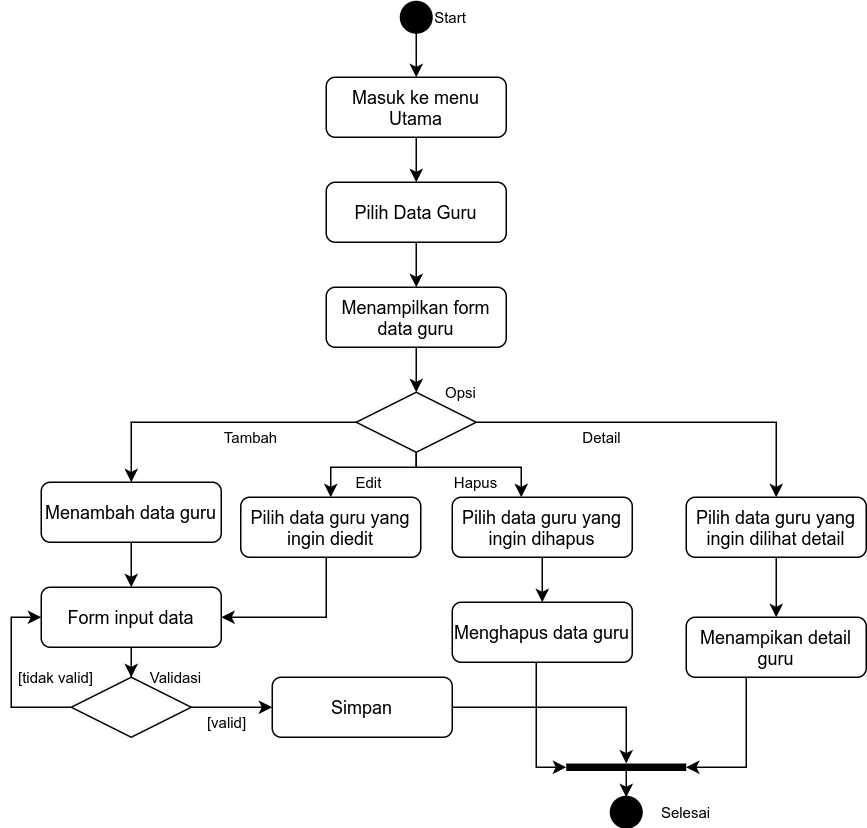
*Activity diagram* mengelola data siswa yaitu admin menginputkan data-data siswa.



Gambar 3.2. *Activity Diagram* Mengelola Data Siswa

1. *Activity Diagram* Mengelola Data Guru

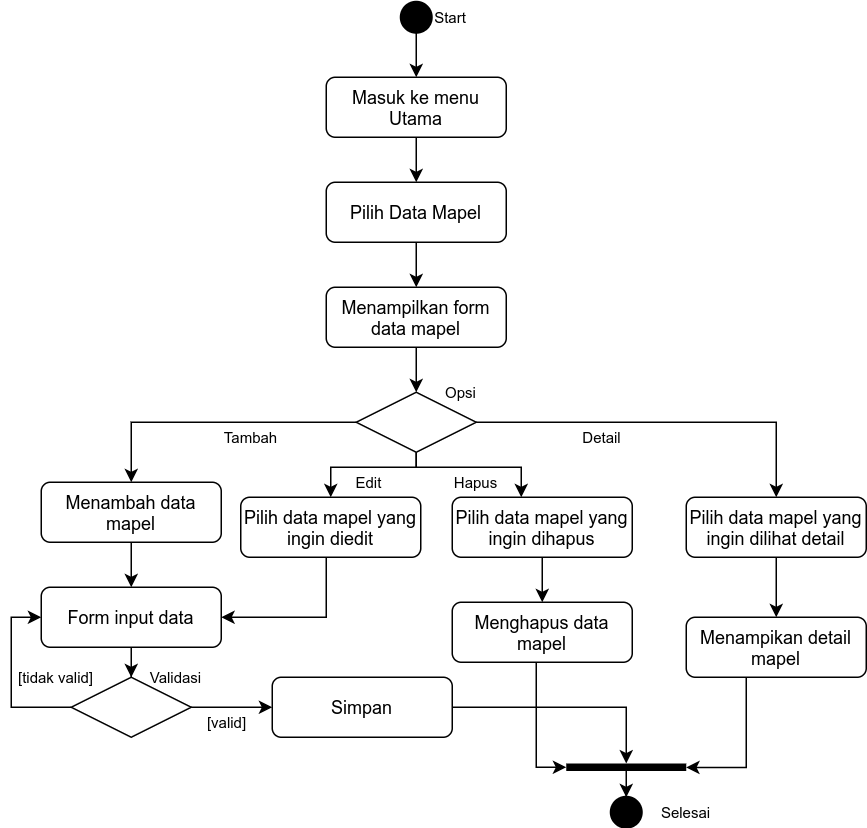
*Activity diagram* mengelola data guru yaitu admin menginputkan data-data guru.



Gambar 3.3. *Activity Diagram* Mengelola Data Guru

1. *Activity Diagram* Mengelola Data Mapel

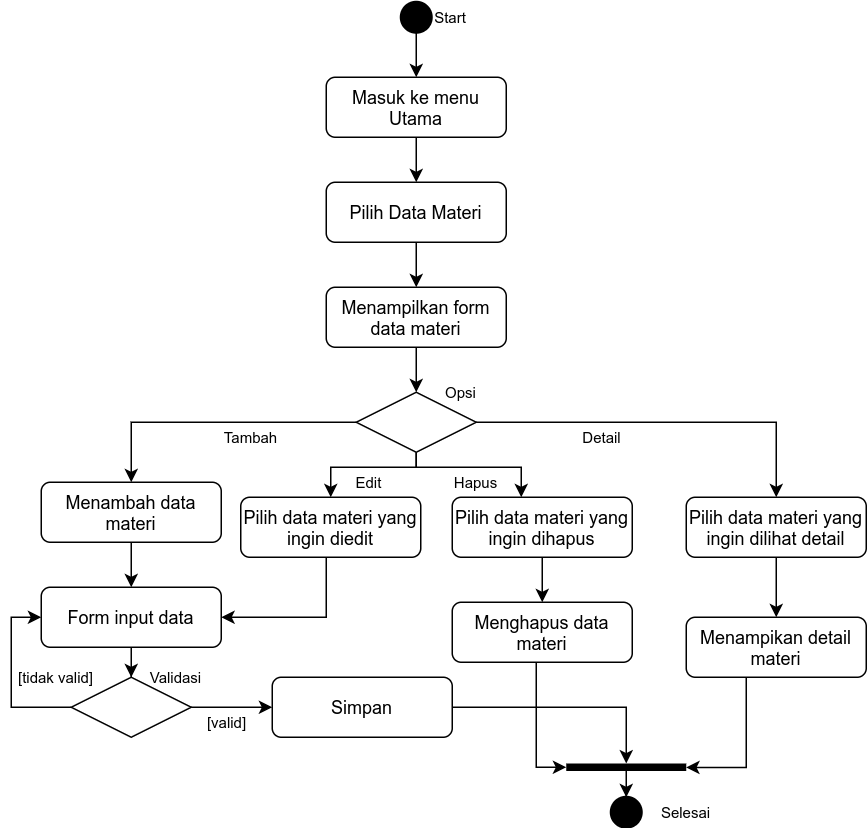
*Activity diagram* mengelola data mapel yaitu admin menginputkandata-data mata pelajaran.



Gambar 3.4. *Activity Diagram* Mengelola Data Mapel

1. *Activity Diagram* Mengelola Data Materi

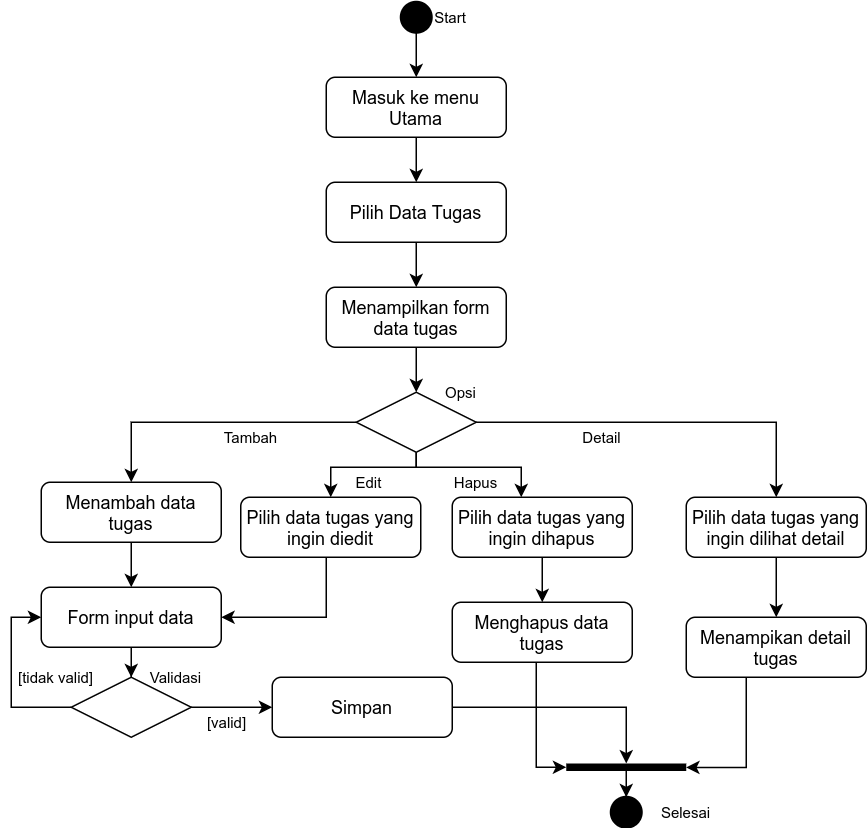
*Activity diagram* mengelola data materi yaitu admin atau gurumenginputkan data materi.



Gambar 3.5. *Activity Diagram* Mengelola Data Materi

1. *Activity Diagram* Mengelola Data Tugas

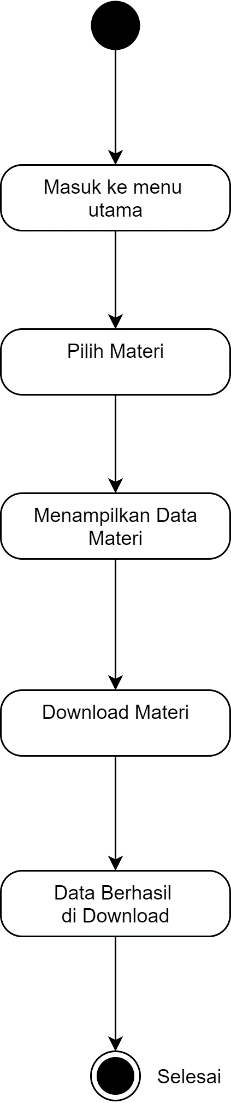
*Activity diagram* mengelola data tugas yaitu admin atau gurumenginputkan data soal.



Gambar 3.6. *Activity Diagram* Mengelola Data Tugas

1. *Activity Diagram* Download Materi

*Activity diagram* d**o**wnload materi merupakan gambaran siswamendwonload materi file yang diberikan oleh guru.



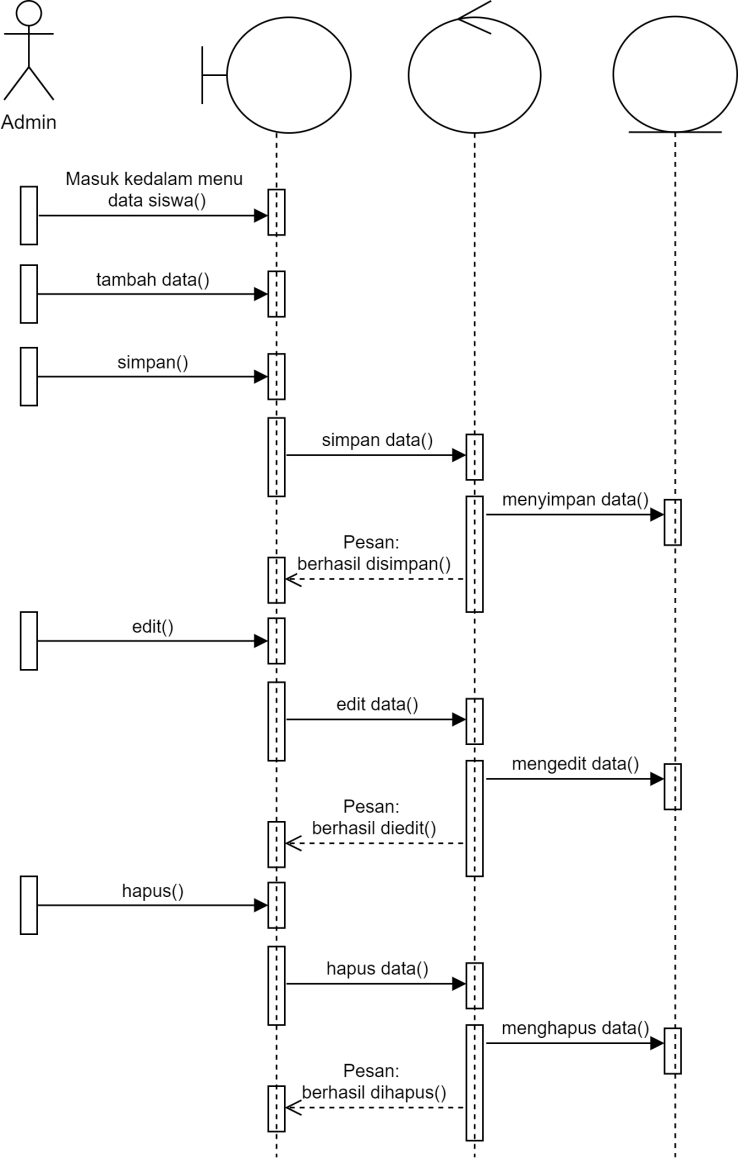
Gambar 3.7. *Activity Diagram* Download Materi

* + 1. **Sequence Diagram**

*Sequence diagram* memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam suatu sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, *display*, dan sebagainya berupa pesan. Diagram ini secara khususberasosiasi dengan *use case* diagram. *Sequence* diagram juga memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu didalam *use case.* Berikut ini adalah *Sequence Diagram* dari Aplikasi *E-Learning.*

1. *Sequence Diagram* Mengelola Data Siswa

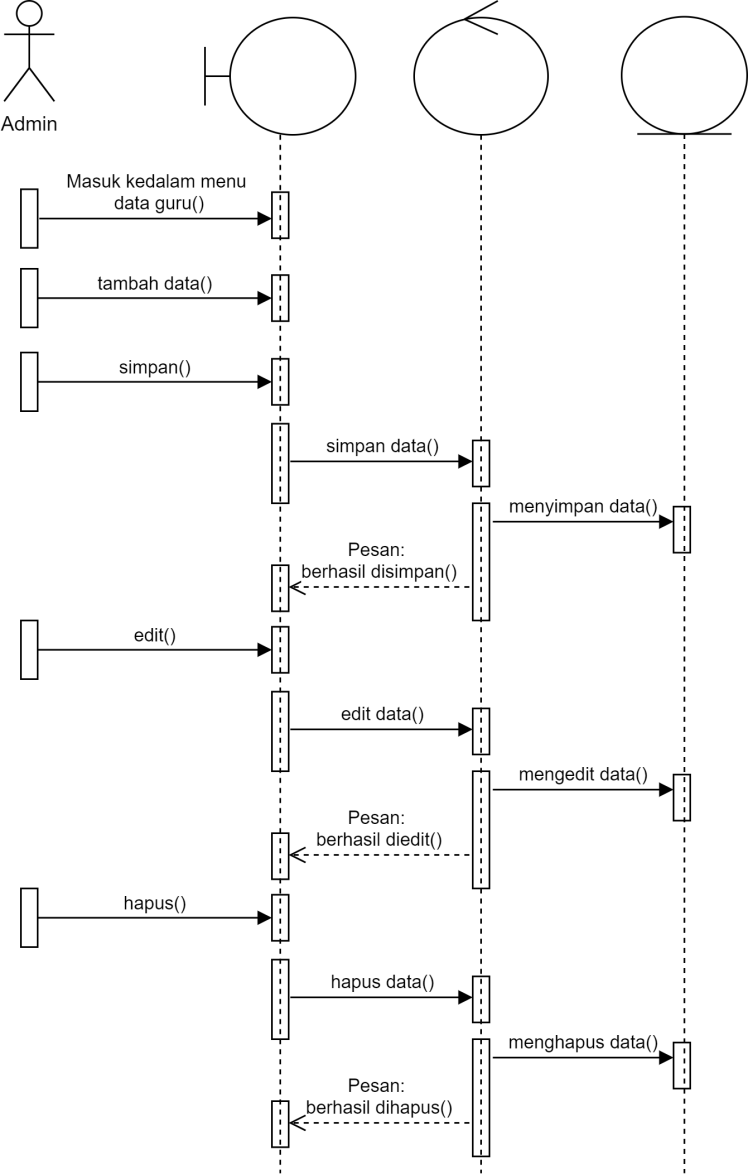
*Sequence diagram* mengelola data siswa berfungsi untuk mengetahuialur dalam admin menginputkan, mengedit, atau menghapus data siswa.



Gambar 3.8. *Sequence Diagram* Mengelola Data Siswa

1. *Sequence Diagram* Mengelola Data Guru

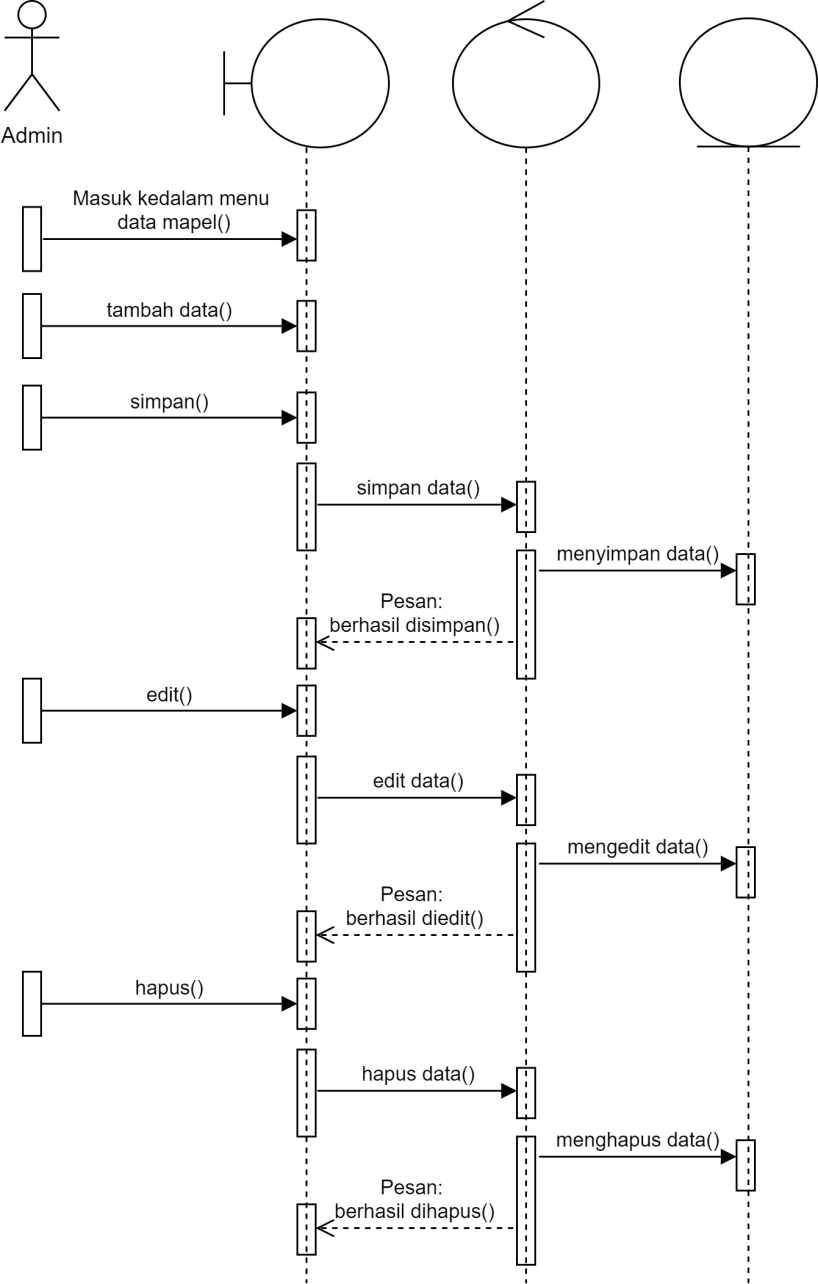
*Sequence diagram* mengelola data guru berfungsi untuk mengetahuialur dalam admin menginputkan, mengedit, atau menghapus data guru.



Gambar 3.9. *Sequence Diagram* Mengelola Data Guru

1. *Sequence Diagram* Mengelola Data Mapel

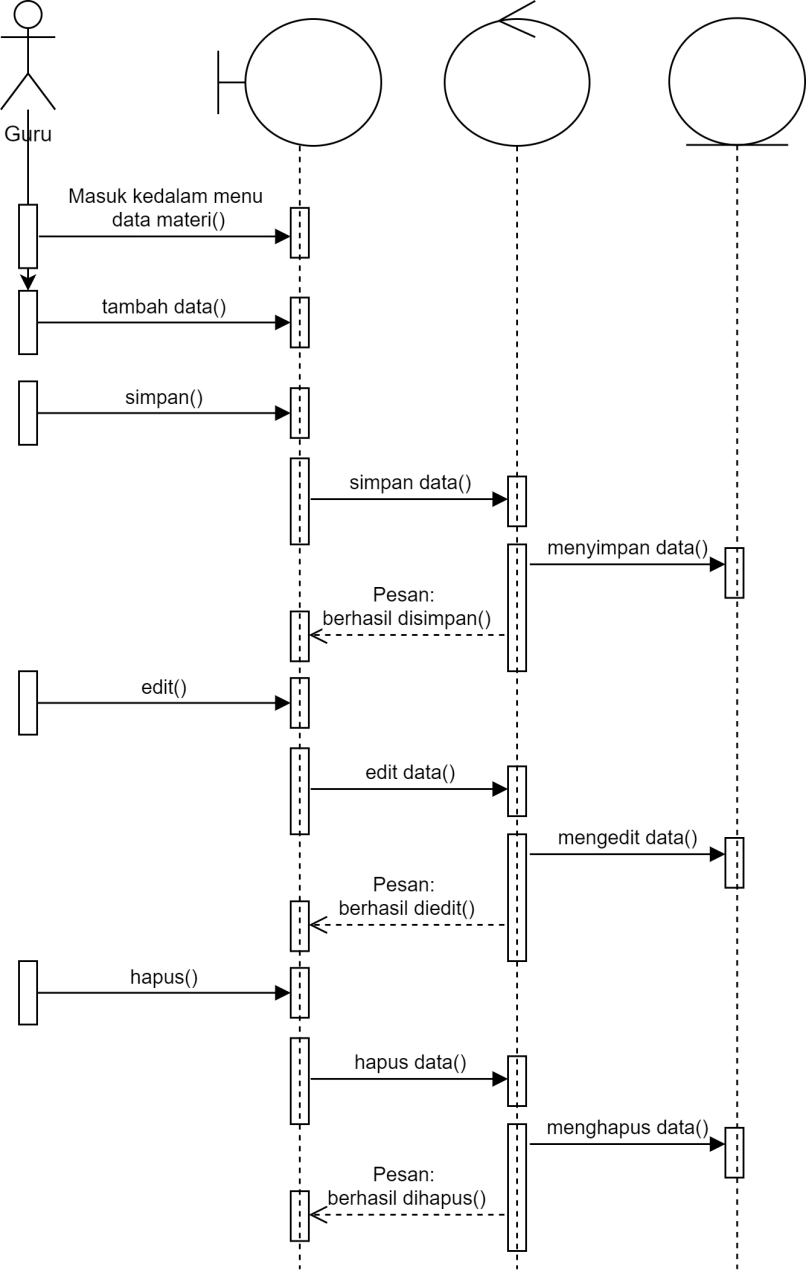
*Sequence diagram* mengelola data mapel berfungsi untuk mengetahuialur dalam admin menginputkan, mengedit atau menghapus data mata pelajaran.



Gambar 3.10. *Sequence Diagram* Mengelola Data Mapel

1. *Sequence Diagram* Mengelola Data Materi

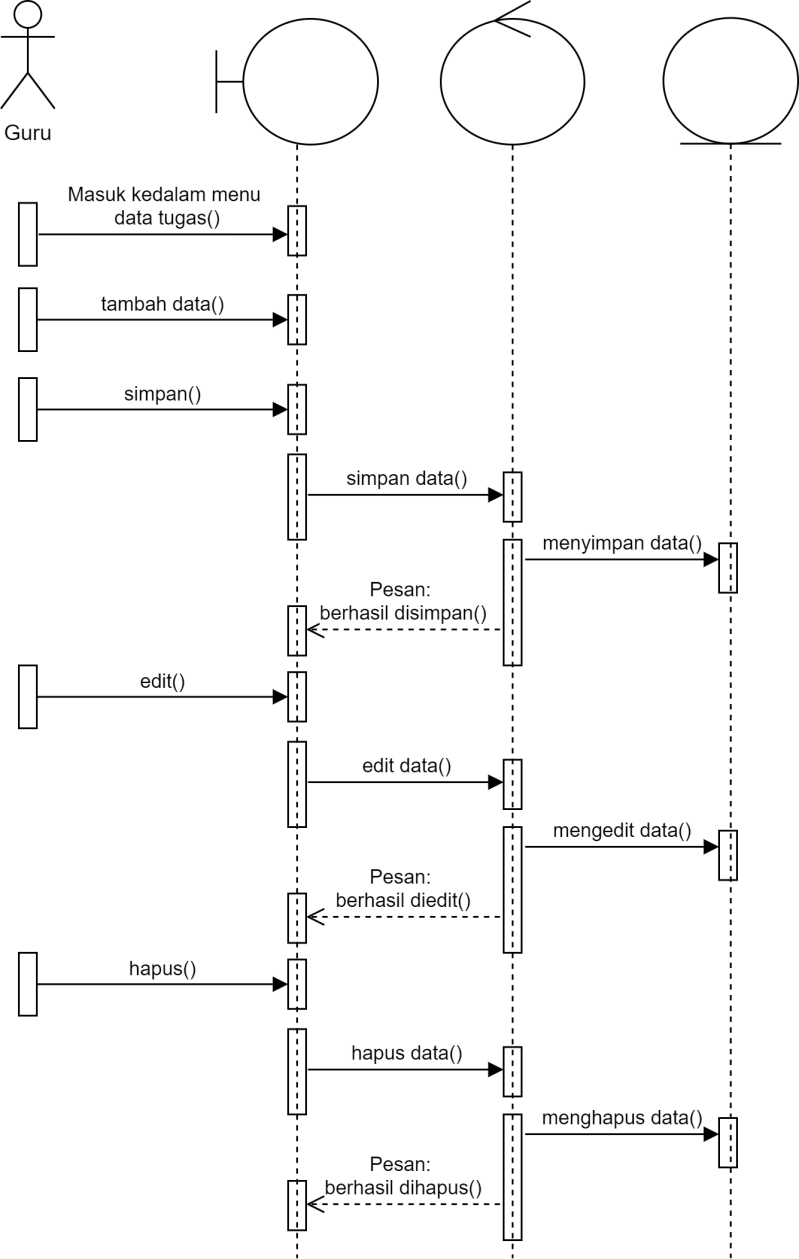
*Sequence diagram* mengelola data materi berfungsi untuk mengetahui alur dalam admin menginputkan, mengedit atau menghapus data siswa.



Gambar 3.11. *Sequence Diagram* Mengelola Data Materi

1. *Sequence Diagram* Mengelola Data Tugas

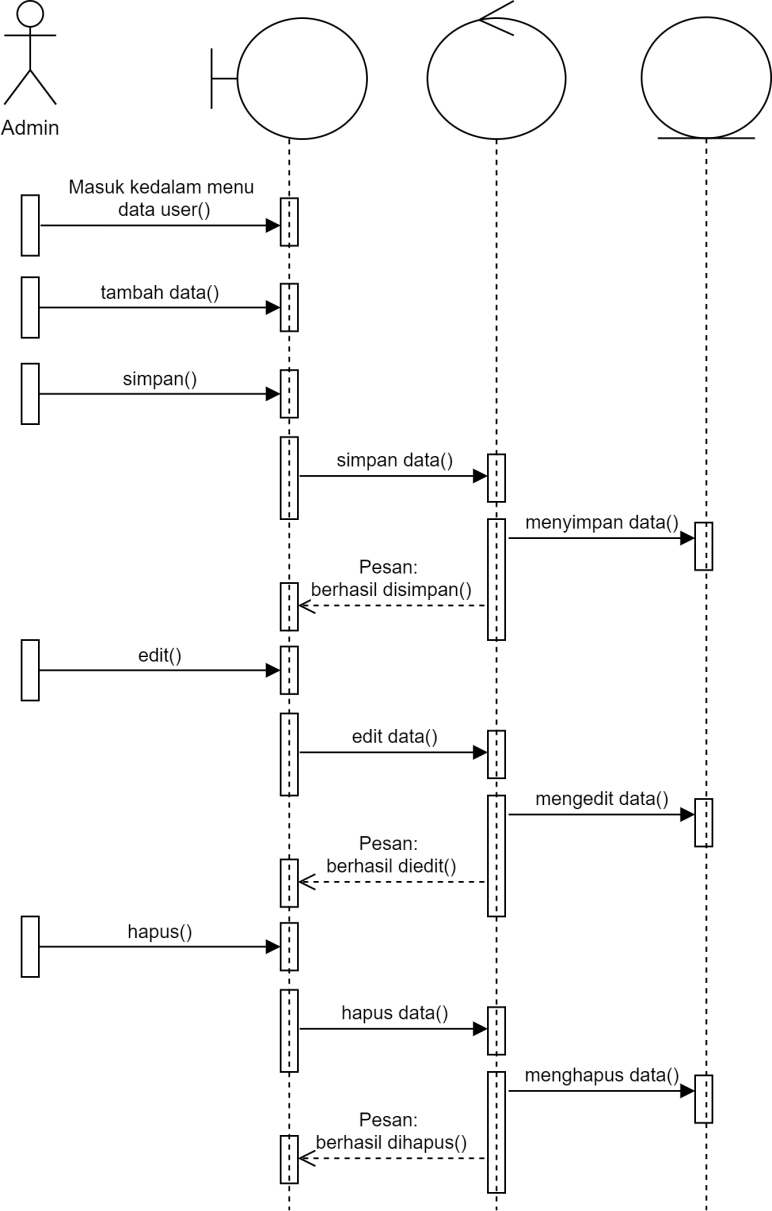
*Sequence diagram* mengelola data tugas berfungsi untuk mengetahui alurdalam admin atau guru dalam menginputkan, mengedit atau menghapus data tugas.



Gambar 3.12. *Sequence Diagram* Mengelola Data Tugas

1. *Sequence Diagram* User

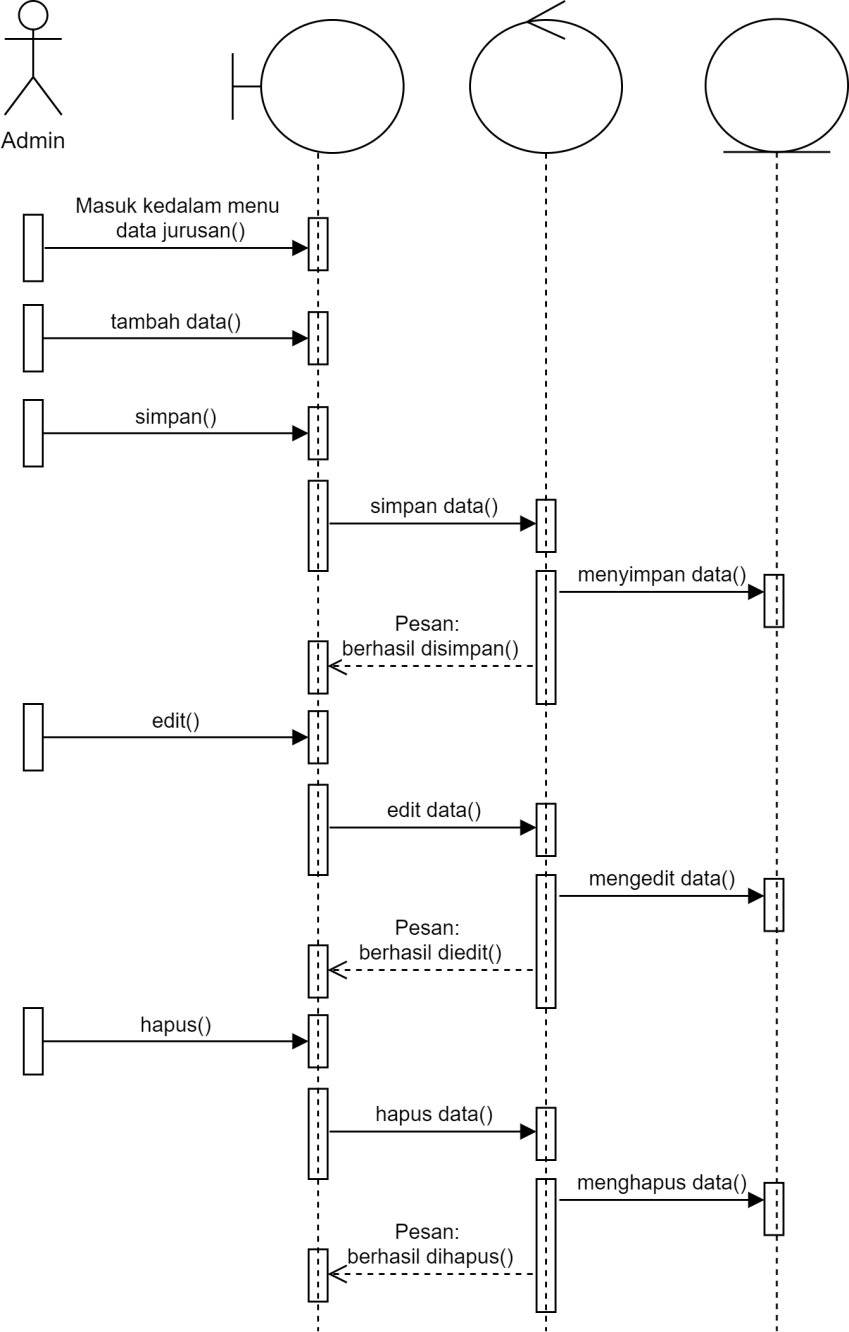
*Sequence diagram* mengelola ujian berfungsi untuk mengetahui alur dalam admin menginputkan, mengedit atau menghapus daftar user.



Gambar 3.13. *Sequence Diagram* Mengelola User

1. *Sequence Diagram* Mengelola Jurusan

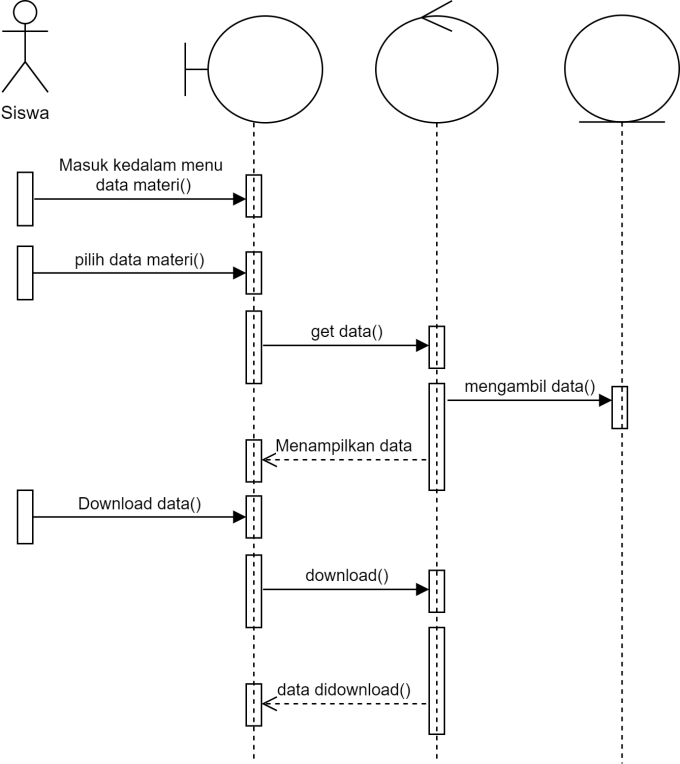
*Sequence diagram* mengelola hasil ujian berfungsi untuk mengetahui alur dalam admin menginputkan, mengedit atau menghapus daftar jurusan.



Gambar 3.14. *Sequence Diagram* Mengelola Data Jurusan

1. *Sequence Diagram* Download Materi

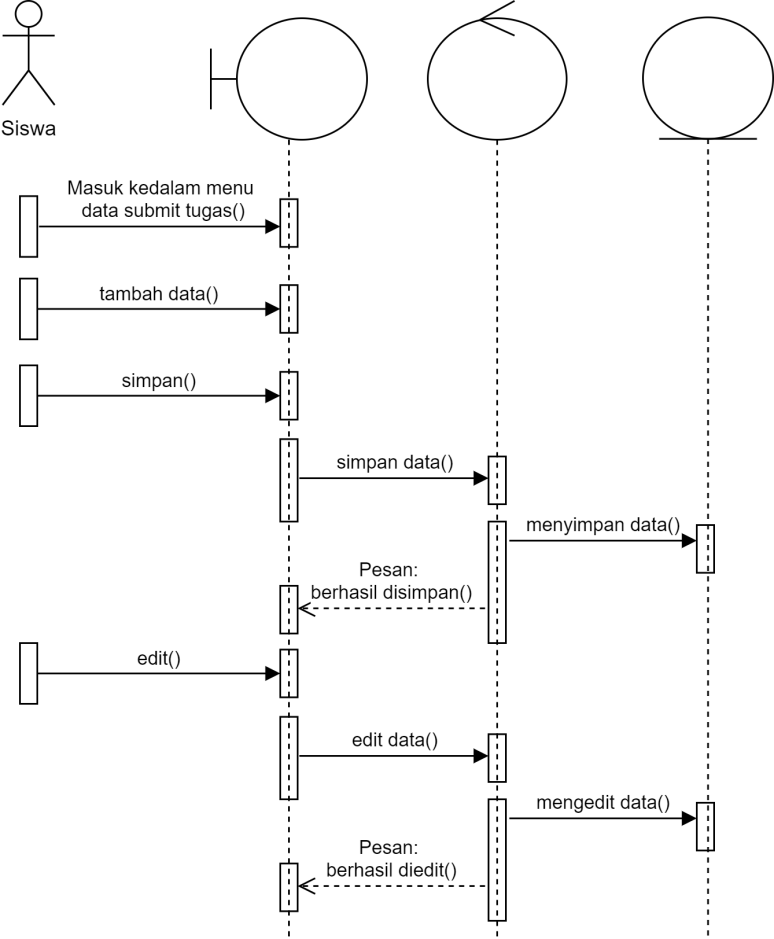
*Sequence diagram* download materi berfungsi untuk mengetahui alurdalam admin menginputkan data siswa.



Gambar 3.15. *Sequence Diagram* Download Materi

1. *Sequence Diagram* Submit Tugas

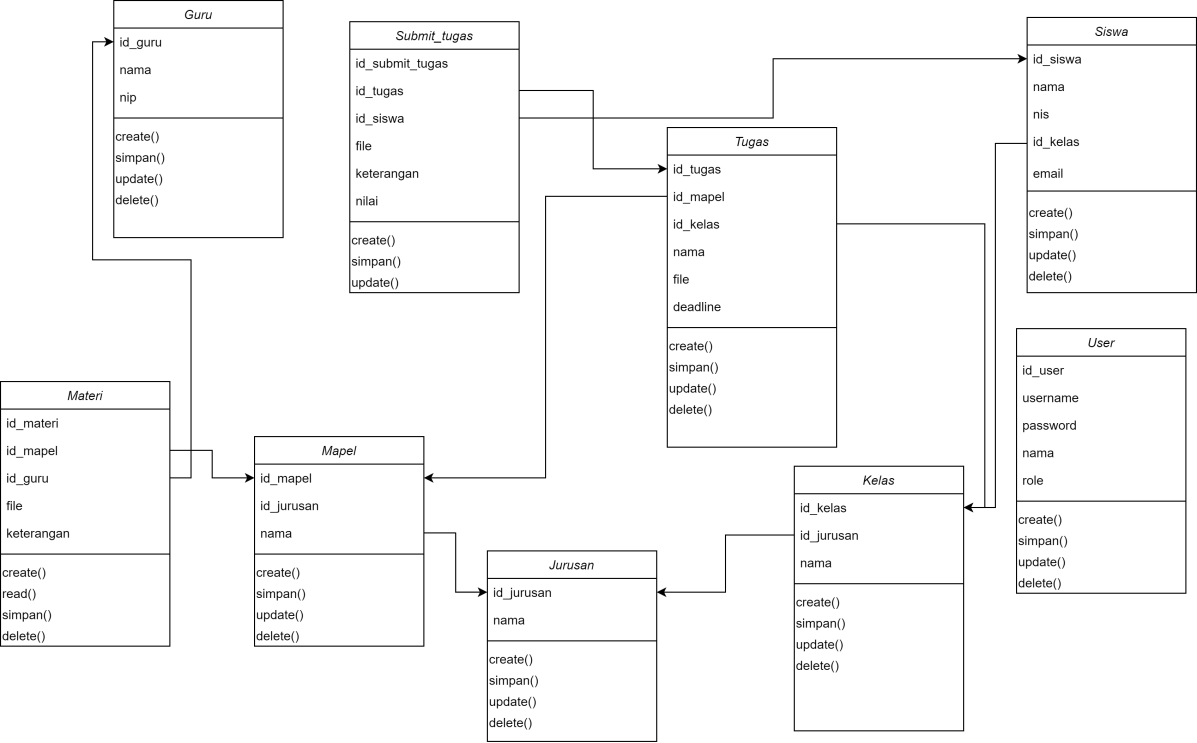
*Sequence diagram* ujian berfungsi untuk mengetahui alur dalam siswamengerjakan/ mensubmit tugas.



Gambar 3.16. *Sequence Diagram* Submit Tugas.

* + 1. **Class Diagram**

*Class diagram* membahas mengenai rancangan secara keseluruhanberupa entitas-entitas yang digunakan dalam sistem beserta relasinya terhadap entitas lain. Berikut *class diagram* pada aplikasi *e-learning.*



Gambar 3.17. *Classs Diagram*

* 1. **Perancangan *Database***

Perancangan *database* merupakan gambaran untuk pembuatan tabel didalam *database* beserta atributnya. Perancangan *database* memberikan gambaran mengenai relasi antar tabel didalam aplikasi. Berikut merupakan perancangan *database* aplikasi *e-learning.*

1. *Tabel User*

Kunci Utama : id\_user

Fungsi : *Untuk login ke halaman menu admin*

Tabel 3.8. *Database* Tabel Admin

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Nama Field | | | Tipe Data | Size | Keterangan |
|  |  |  | |  |  |  |
| 1. | Id\_user | |  | Int | 6 | *Primary Key* |
|  |  | | |  |  |  |
| 2. | Username | | | Varchar | 100 |  |
|  |  | | |  |  |  |
| 3. | Password | | | Varchar | 100 |  |
|  |  |  | |  |  |  |
| 4. | Role |  | | Int |  |  |
|  |  |  | |  |  |  |
| 5. | Nama |  | | Varchar | 50 |  |
| 6. | Foto |  | | Varchar | 100 |  |

1. *Tabel Guru*

Kunci Utama : id\_guru

Fungsi : *Untuk mengelola data guru*

Tabel 3.9. *Database* Tabel Guru

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Nama Field | | Tipe Data | Size | Keterangan |
|  |  |  |  |  |  |
| 1. | Id\_guru |  | Int | 6 | *Primary Key* |
|  |  |  |  |  |  |
| 2. | Nama |  | Varchar | 100 |  |
|  |  |  |  |  |  |
| 3. | Nip |  | Varchar | 100 |  |
|  |  |  |  |  |  |
| 4. | Email |  | Varchar | 200 |  |

1. *Tabel Siswa*

Kunci Utama : id\_siswa

Fungsi : *Untuk mengelola data siswa*

Tabel 3.10. *Database* Tabel Siswa

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Nama Field | | Tipe Data | Size | Keterangan |
|  |  |  |  |  |  |
| 1. | Id\_siswa |  | Int | 6 | *Primary Key* |
|  |  |  |  |  |  |
| 2. | Nama |  | Varchar | 100 |  |
|  |  |  |  |  |  |
| 3. | Nis |  | Varchar | 50 |  |
|  |  |  |  |  |  |
| 4. | Kelas |  | Varchar | 50 |  |
|  |  |  |  |  |  |
| 5. | Email |  | Varchar | 50 |  |

1. *Tabel Mapel*

Kunci Utama : id\_mapel

Fungsi : *Untuk mengelola data mapel*

Tabel 3.11. *Database* Tabel Mapel

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Nama Field | Tipe Data | Size | Keterangan |
| 1. | Id\_mapel | Int | 6 | *Primary Key* |
| 2. | Id\_jurusan | Int | 6 |  |
| 3. | Nama | Varchar | 100 |  |

1. *Tabel Materi*

Kunci Utama : id\_materi

Fungsi : *Untuk mengelola data materi*

Tabel 3.12. *Database* Tabel Materi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Nama Field | | Tipe Data | Size | Keterangan |
|  |  |  |  |  |  |
| 1. | Id\_materi |  | Int | 11 | *Primary Key* |
|  |  |  |  |  |  |
| 2. | Id\_guru |  | Int | 11 | *Foreign Key* |
|  |  | |  |  |  |
| 3. | Id\_mapel | | Int | 11 | *Foreign Key* |
|  |  |  |  |  |  |
| 4. | File |  | Varchar | 200 |  |
| 5. | Keterangan | | Text |  |  |

1. *Tabel Jurusan*

Kunci Utama : id\_jurusan

Fungsi : *Untuk mengelola data jurusan*

Tabel 3.13. *Database* Tabel Jurusan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Field | *Tipe Data* | *Size* | *Keterangan* |
| *1.* | Id\_jurusan | Int | 11 | *Primary Key* |
| *2.* | Nama | Varchar | 50 |  |

1. *Tabel Kelas*

Kunci Utama : id\_kelas

Fungsi : *Untuk mengelola data kelas*

Tabel 3.14. *Database* Tabel Kelas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Field | *Tipe Data* | *Size* | *Keterangan* |
| *1.* | Id\_kelas | Int | 11 | *Primary Key* |
| *2.* | Nama | Varchar | 50 |  |
| *3.* | Id\_jurusan | Int | 11 |  |

1. *Tabel Submit Tugas*

Kunci Utama : id\_submit\_tugas

Fungsi : *Untuk mengelola data submit tugas*

Tabel 3.14. *Database* Tabel Submit Tugas

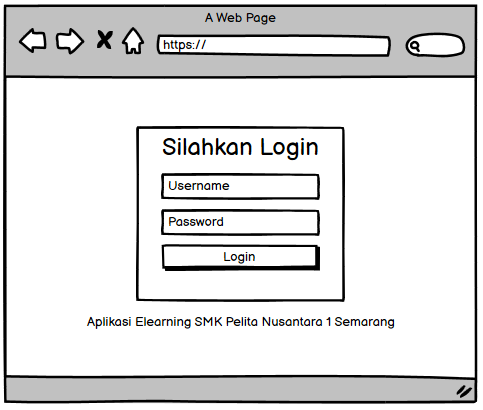
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Nama Field | *Tipe Data* | *Size* | *Keterangan* |
| *1.* | Id\_submit\_tugas | Int | 11 | *Primary Key* |
| *2.* | Id\_tugas | Int | 11 |  |
| *3.* | Id\_siswa | Int | 11 |  |
| *4.* | file | Varchar | 200 |  |
| *5.* | keterangan | Text |  |  |
| *6.* | nilai | Int | 11 |  |

* 1. **Perancangan Antarmuka**

Perancangan antarmuka atau *interface* dari suatu sistem berfungsi untuk memberikan gambaran yang akan ditampilkan dari program. *Interface* yang menarik juga akan memberikan kemudahan bagi penggunanya. Berikut ini rancangan antar muka aplikasi *e-learning :*

1. Tampilan Halaman Login

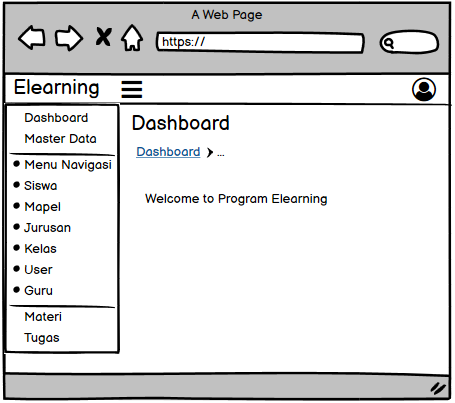
Halaman Login berisi *form input username* dan *password* sebelum admin masuk ke dalam sistem.



Gambar 3.18.Perancangan Antar Muka Halaman Login.

1. Tampilan Halaman Dashboard Admin

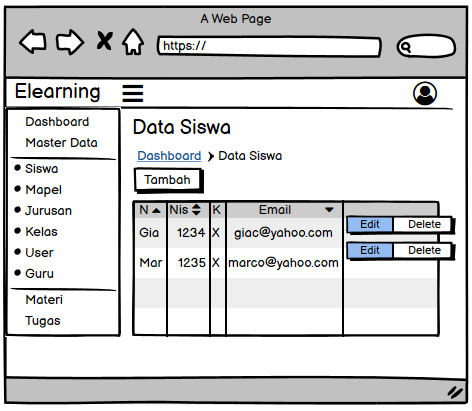
Halaman Dashboard berisi menu-menu yang dapat diakses oleh administrator, guru, dan siswa.



Gambar 3.19. Perancangan Antar Muka Halaman Dashboard.

1. Tampilan Halaman Data Siswa

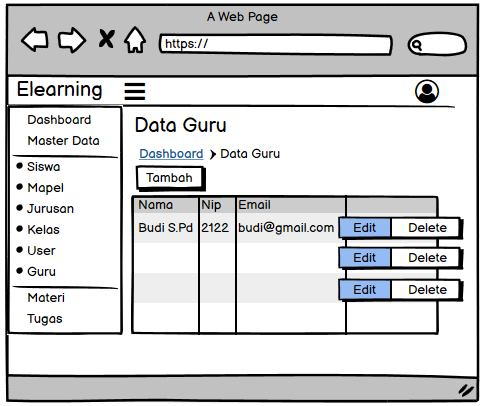
Halaman data siswa memiliki fungsi untuk mengelola data siswa. Pada halaman ini terdapat tambah, simpan, edit, hapus.



Gambar 3.20. Perancangan Antar Muka Halaman Data Siswa.

1. Tampilan Halaman Data Guru

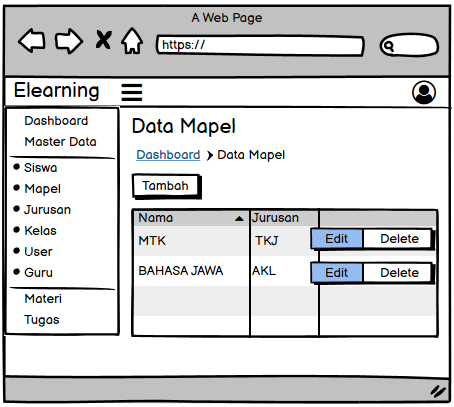
Halaman data guru memiliki fungsi untuk mengelola data guru. Pada halaman ini terdapat tambah, simpan, edit, hapus.



Gambar 3.21. Perancangan Antar Muka Halaman Data Guru.

1. Tampilan Halaman Data Mapel

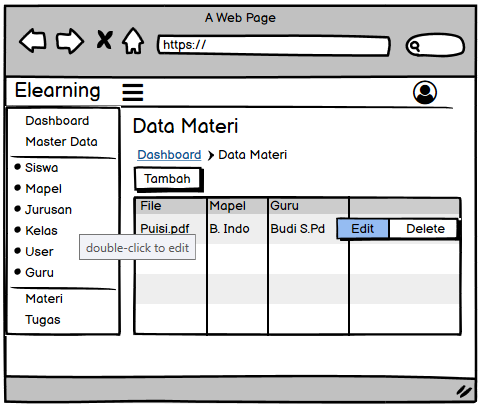
Halaman data mapel memiliki fungsi untuk mengelola data mapel. Pada halaman ini terdapat tambah, simpan, edit, hapus.



Gambar 3.22. Perancangan Antar Muka Halaman Data Mapel.

1. Tampilan Halaman Data Materi

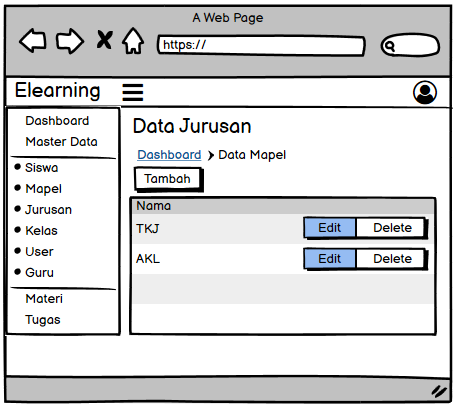
Halaman data materi memiliki fungsi untuk mengelola data materi. Pada halaman ini terdapat tambah, simpan, edit, hapus.



Gambar 3.23. Perancangan Antar Muka Halaman Data Materi.

1. Tampilan Halaman Data Jurusan

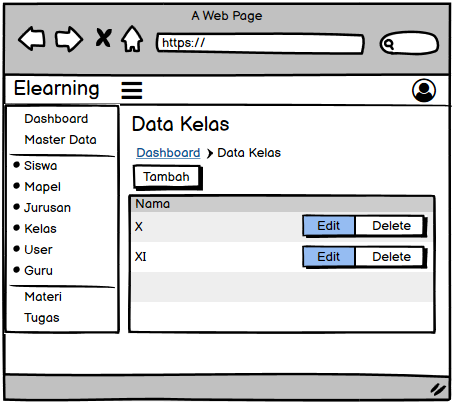
Halaman data jurusan memiliki fungsi untuk mengelola data jurusan. Pada halaman ini terdapat tambah, simpan, edit, hapus.



Gambar 3.24. Perancangan Antar Muka Halaman Data Jurusan.

1. Tampilan Halaman Data Kelas

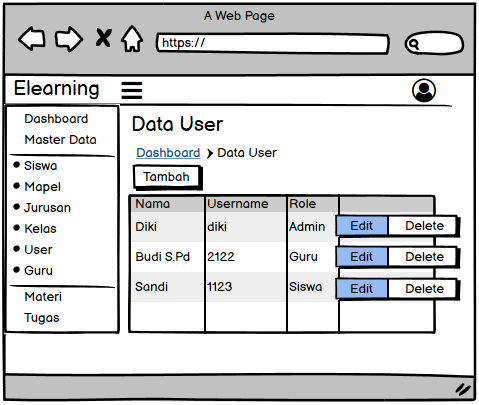
Halaman data kelas memiliki fungsi untuk mengelola data jurusan. Pada halaman ini terdapat tambah, simpan, edit, hapus.



Gambar 3.25. Perancangan Antar Muka Halaman Data Kelas.

1. Tampilan Halaman Data User

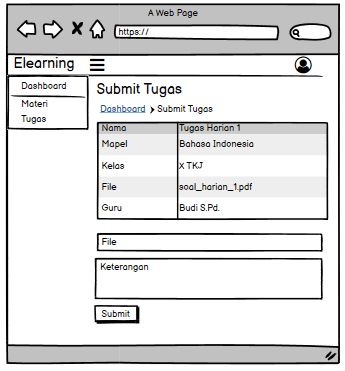
Halaman data kelas memiliki fungsi untuk mengelola data jurusan. Pada halaman ini terdapat tambah, simpan, edit, hapus.



Gambar 3.26. Perancangan Antar Muka Halaman Data User.

1. Tampilan Halaman Submit Tugas

Halaman data kelas memiliki fungsi untuk mengelola data jurusan. Pada halaman ini terdapat tambah, simpan, edit.



Gambar 3.27. Perancangan Antar Muka Halaman Submit Tugas.