

BAB IV ANALISIS DAN PERANCANGAN

4.1. Analisis

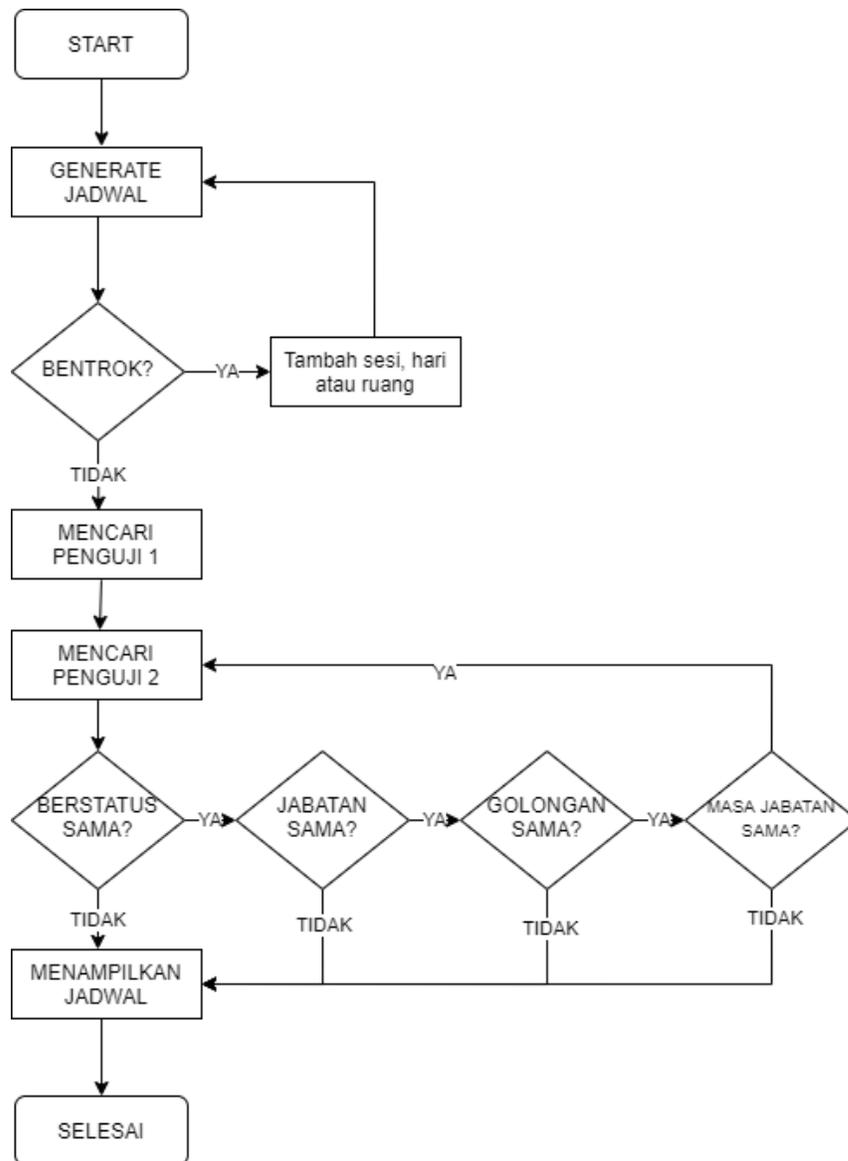
Analisis merupakan hal yang sangat penting saat membangun sebuah sistem. Dengan proses analisis yang tepat akan lebih memudahkan pengembang saat memulai implementasi pembuatan sistem, sekaligus akan mengurangi *trial and error* saat melakukan pengujian sebuah program.

4.1.1. Analisis *rule*/aturan

Pada sistem penjadwalan yang akan dibangun terdapat *rule*/aturan yang dibutuhkan untuk men-*generate* jadwal, adapun *rule* yang dibutuhkan yaitu :

- a. *Generate* jadwal terlebih dahulu dimana tidak terjadi bentrok pada hari, jam dan ruang yang sama.
- b. Penguji 1 wajib dosen PNS dan pada *group* riset berstatus Major.
- c. Penguji 2 wajib dosen PNS dan pada *group* riset berstatus Major/Minor.
- d. Jika penguji 1 dan 2 memiliki status *group* riset sama, maka perlu dipertimbangkan jabatan fungsionalnya, dapat diartikan penguji 1 jabatan fungsionalnya lebih tinggi dari penguji 2, jika masih terdapat status yang sama, maka dapat dilihat dari golongan dosen, atau masa jabatannya.
- e. Jumlah pengujian tiap dosen harus proporsional.

Berikut merupakan flowchart dari *rule* yang akan diberlakukan pada pembuatan jadwal yang terdapat pada Gambar 4.1.



Gambar 4. 1 *Flowchart Generate Jadwal*

4.1.2. Analisis Pengguna

Pada sistem yang dibangun terdapat dua jenis pengguna. Pengguna yang pertama adalah admin sistem. Admin sistem merupakan pengguna yang dapat mengelola master data. Untuk menjadi Admin sistem dibutuhkan *login* terlebih dahulu pada sistem.

Tabel 4. 1 Level Pengguna

No	Pengguna	Deskripsi
1	Admin	Dapat mengelola data master yang dibutuhkan di dalam sistem. Data tersebut adalah data dosen, status dosen, golongan, jabatan, masa jabatan, sesi, jadwal.
2	Panitia	Dapat membuat jadwal, serta mengelola jadwal

4.1.3. Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang berisi fitur atau fungsi apa saja yang harus disediakan oleh sistem, mencakup bagaimana sistem bekerja pada *Input* tertentu dan bagaimana perilaku sistem pada situasi tertentu. Kebutuhan ini juga yang merepresentasikan tujuan dari sistem ini di bangun. Berikut adalah kebutuhan fungsional setelah menganalisa sistem.

Tabel 4. 2 Kebutuhan Fungsional

No	Aktifitas	Deskripsi
1	Melakukan <i>login</i>	Fitur keamanan dengan username dan <i>password</i> yang di gunakan untuk mengakses sistem.
2	Kelola data dosen	Fitur untuk menambah mengedit dan menghapus data dosen dalam hal ini adalah admin.
4	Kelola status	Fitur untuk menambah mengedit dan menghapus data status dosen dalam hal ini adalah admin.
5	Kelola golongan	Fitur untuk menambah mengedit dan menghapus data golongan dosen dalam hal ini adalah admin.

6	Kelola jabatan	Fitur untuk menambah mengedit dan menghapus data jabatan dosen dalam hal ini adalah admin.
7	Kelola sesi	Fitur untuk menambah mengedit dan menghapus data sesi dalam hal ini adalah admin.
8.	Kelola jadwal	Fitur untuk menambah mengedit dan menghapus data jadwal dalam hal ini adalah admin dan panitia.
9.	Kelola Ruang	Fitur untuk menambah mengedit dan menghapus data Ruang dalam hal ini adalah admin dan panitia.

4.1.4. Kebutuhan Non-Fungsional

Kebutuhan *non-fungsional* merupakan kebutuhan yang mendeskripsikan sekumpulan batasan, karakteristik dan properti pada sistem, baik dalam lingkungan pengembangan maupun operasional, atau atribut kualitas yang harus dipenuhi oleh sistem. Dalam kebutuhan ini terdapat kebutuhan perangkat lunak dan perangkat keras yang dapat menunjang proses pengembangan dan operasional.

a. Kebutuhan Perangkat Lunak

Kondisi atau kriteria perangkat lunak yang digunakan dalam penelitian ini terdapat pada Tabel 4.3

Tabel 4. 3 Kebutuhan Perangkat Lunak

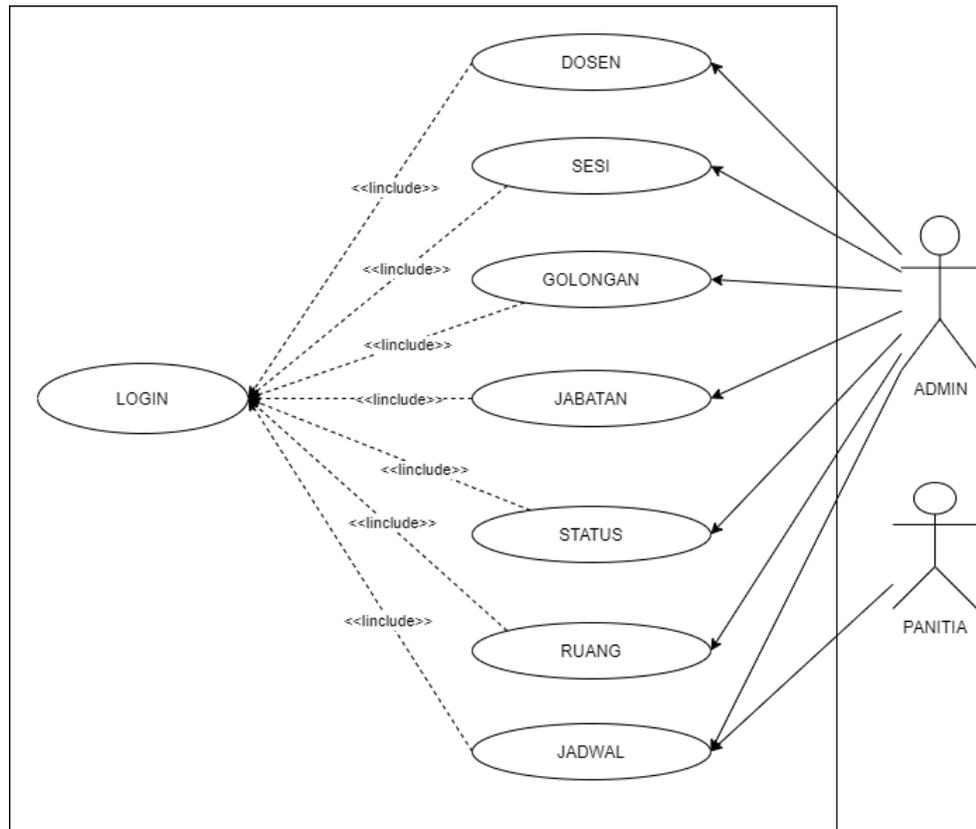
No	Nama	Keterangan
1	<i>Text editor</i>	<i>Sublim Text 3</i>
2	<i>Framework</i>	<i>CodeIgniter 3.4</i>
3	<i>Database</i>	<i>PhpMyadmin 7.2</i>
4	<i>Web browser</i>	<i>Google Chrome</i>

Tabel 4. 4 Kebutuhan Perangkat Keras

Perangkat Keras	Keterangan
<i>Processor</i>	Intel ® Core™ i3
Memori RAM	2 gb RAM
Memori VGA	Minimum 512 Gb ke atas
<i>Hard disk space</i>	Di butuhkan sekitar 1 Gb

4.1.5. Use Case Diagram

Use Case diagram menggambarkan alur sistem yang dirancang untuk memenuhi kebutuhan *funksional*. Berdasarkan analisis di atas terdapat 2 pengguna dan terdapat 6 (enam) kegiatan di dalam sistem, berikut adalah alur sistem *Use Case* sistem yang akan di gambarkan pada gambar 4.2.



Gambar 4. 2 Use Case Diagram

4.1.6. Skenario Use Case Diagram

Skenario *Use Case* bertujuan untuk mendeskripsikan aktifitas yang terdapat pada *Use Case diagram*, berikut adalah skenario *Use Case* dalam melakukan aktifitas-aktifitas kegiatan.

a. Melakukan *login*

Skenario ketika admin dan panitia melakukan *login* akan dijelaskan pada Tabel 4.5.

Tabel 4. 5 Melakukan *login*

Melakukan <i>Login</i>	
Pengguna	Admin sistem
Tujuan	Untuk keamanan sistem dengan melakukan otorisasi menggunakan <i>username</i> dan <i>password</i>
Kondisi awal	Terkoneksi dengan Internet
Skenario utama	Memasukan <i>username</i> dan <i>password</i>
Kondisi alternatif	<ol style="list-style-type: none"> 1. Jika <i>username</i> dan atau <i>password</i> belum diisi, muncul notifikasi error pada laman <i>login</i> 2. Jika <i>username</i> atau <i>password</i> tidak sesuai, muncul notifikasi <i>login</i> gagal pada laman <i>login</i>
Kondisi akhir	Menampilkan halaman utama sistem

b. Melakukan pengelolaan data Dosen

Skenario ketika admin melakukan pengelolaan data dosen akan dijelaskan pada Tabel 4.6.

Tabel 4. 6 Mengelola data dosen

Kelola Data Dosen	
Pengguna	Admin sistem
Tujuan	Untuk menambahkan data dosen, dalam hal ini adalah admin dapat menambah, menghapus dan mengedit data dosen yang sudah ada
Kondisi awal	<ol style="list-style-type: none"> 1. <i>Login</i> sebagai admin sistem 2. Tersambung dengan Internet

Skenario utama	<ol style="list-style-type: none"> 1. Admin memilih menu Data Dosen 2. Sistem menampilkan data Dosen
Kondisi alternatif	<ol style="list-style-type: none"> 1. Jika admin ingin menambah data dosen, admin menekan tombol <i>icon</i> tambah, jika berhasil maka akan langsung tersimpan kedalam database 2. Jika admin ingin menghapus data dosen, admin menekan tombol <i>icon</i> sampah. 3. Jika admin ingin mengedit data dosen, admin menekan tombol <i>icon</i> edit
Kondisi akhir	Sistem menampilkan data dosen sesuai yang berada di <i>database</i>

c. Melakukan pengolahan data status dosen

Skenario ketika admin melakukan pengelolaan data status akan dijelaskan pada Tabel 4.7.

Tabel 4. 7 Mengelola data status

Kelola Data Status	
Pengguna	Admin sistem
Tujuan	Untuk menambahkan data status, dalam hal ini adalah admin dapat menambah, menghapus dan mengedit data status yang sudah ada
Kondisi awal	<ol style="list-style-type: none"> 3. <i>Login</i> sebagai admin sistem 4. Tersambung dengan Internet
Skenario utama	<ol style="list-style-type: none"> 3. Admin memilih menu Data Status 4. Sistem menampilkan data status

Kondisi alternatif	<p>4. Jika admin ingin menambah data status, admin menekan tombol <i>icon</i> tambah, jika berhasil maka akan langsung tersimpan kedalam database</p> <p>5. Jika admin ingin menghapus data status, admin menekan tombol <i>icon</i> sampah.</p> <p>6. Jika admin ingin mengedit data status, admin menekan tombol <i>icon</i> edit</p>
Kondisi akhir	Sistem menampilkan data status sesuai yang berada di <i>database</i>

d. Skenario mengelola data sesi

Skenario ketika admin melakukan pengelolaan data sesi akan dijelaskan pada Tabel 4.8.

Tabel 4. 8 Mengelola data sesi

Kelola Data Sesi	
Pengguna	Admin sistem
Tujuan	Untuk menambahkan data sesi, dalam hal ini adalah admin dapat menambah, menghapus dan mengedit data sesi yang sudah ada
Kondisi awal	<p>5. <i>Login</i> sebagai admin sistem</p> <p>6. Tersambung dengan Internet</p>
Skenario utama	<p>5. Admin memilih menu Data Sesi</p> <p>6. Sistem menampilkan data sesi</p>

Kondisi alternatif	<p>7. Jika admin ingin menambah data sesi, admin menekan tombol <i>icon</i> tambah, jika berhasil maka akan langsung tersimpan kedalam database</p> <p>8. Jika admin ingin menghapus data sesi, admin menekan tombol <i>icon</i> sampah.</p> <p>9. Jika admin ingin mengedit data sesi, admin menekan tombol <i>icon</i> edit</p>
Kondisi akhir	Sistem menampilkan data sesi sesuai yang berada di <i>database</i>

e. Skenario mengelola data golongan

Skenario ketika admin melakukan pengelolaan data golongan akan dijelaskan pada Tabel 4.9.

Tabel 4. 9 Mengelola data golongan

Kelola Data Golongan	
Pengguna	Admin sistem
Tujuan	Untuk menambahkan data golongan, dalam hal ini adalah admin dapat menambah, menghapus dan mengedit data golongan yang sudah ada
Kondisi awal	<p>7. <i>Login</i> sebagai admin sistem</p> <p>8. Tersambung dengan Internet</p>
Skenario utama	<p>7. Admin memilih menu Data golongan</p> <p>8. Sistem menampilkan data golongan</p>

Kondisi alternatif	<p>10. Jika admin ingin menambah data golongan, admin menekan tombol <i>icon</i> tambah, jika berhasil maka akan langsung tersimpan kedalam database</p> <p>11. Jika admin ingin menghapus data golongan, admin menekan tombol <i>icon</i> sampah.</p> <p>12. Jika admin ingin mengedit data golongan, admin menekan tombol <i>icon</i> edit</p>
Kondisi akhir	Sistem menampilkan data golongan sesuai yang berada di <i>database</i>

f. Skenario mengelola data Ruang

Skenario ketika admin melakukan pengelolaan data golongan akan dijelaskan pada Tabel 4.10.

Tabel 4. 10 Mengelola data ruang

Kelola Data Ruang	
Pengguna	Admin sistem
Tujuan	Untuk menambahkan data Ruang , dalam hal ini adalah admin dapat menambah, menghapus dan mengedit data Ruang yang sudah ada
Kondisi awal	<p>9. <i>Login</i> sebagai admin sistem</p> <p>10. Tersambung dengan Internet</p>
Skenario utama	<p>9. Admin memilih menu Data Ruang</p> <p>10. Sistem menampilkan data Ruang</p>

Kondisi alternatif	<p>13. Jika admin ingin menambah data ruang, admin menekan tombol <i>icon</i> tambah, jika berhasil maka akan langsung tersimpan kedalam database</p> <p>14. Jika admin ingin menghapus data ruang, admin menekan tombol <i>icon</i> sampah.</p> <p>15. Jika admin ingin mengedit data ruang, admin menekan tombol <i>icon</i> edit</p>
Kondisi akhir	Sistem menampilkan data ruang sesuai yang berada di <i>database</i>

g. Skenario mengelola data Jadwal

Skenario ketika admin dan panitia melakukan pengelolaan data jadwal akan dijelaskan pada Tabel 4.11.

Tabel 4. 11 Mengelola data Jadwal

Kelola Data Ruang	
Pengguna	Admin sistem
Tujuan	Untuk menambahkan data Ruang , dalam hal ini adalah admin dapat menambah, menghapus dan mengedit data Ruang yang sudah ada
Kondisi awal	<p>11. <i>Login</i> sebagai admin sistem</p> <p>12. Tersambung dengan Internet</p>
Skenario utama	<p>11. Admin memilih menu Data Ruang</p> <p>12. Sistem menampilkan data Ruang</p>

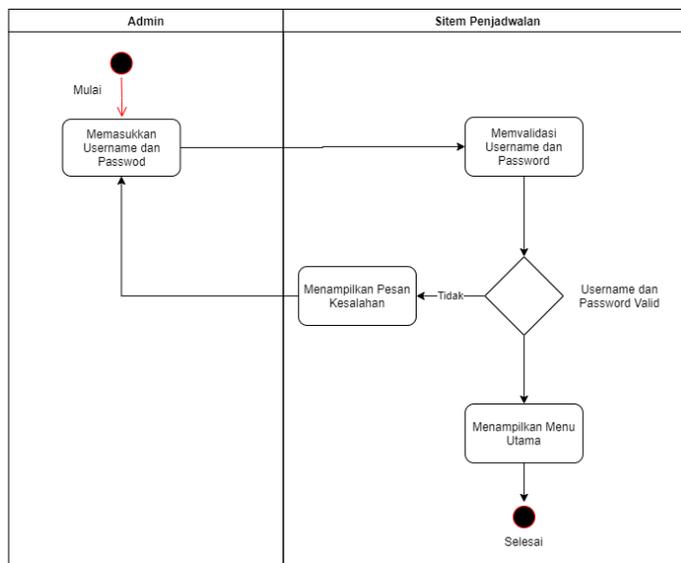
Kondisi alternatif	<p>16. Jika admin ingin menambah data ruang, admin menekan tombol <i>icon</i> tambah, jika berhasil maka akan langsung tersimpan kedalam database</p> <p>17. Jika admin ingin menghapus data ruang, admin menekan tombol <i>icon</i> sampah.</p> <p>18. Jika admin ingin mengedit data ruang, admin menekan tombol <i>icon</i> edit</p>
Kondisi akhir	Sistem menampilkan data ruang sesuai yang berada di <i>database</i>

4.1.7. Activity Diagram

Activity Diagram merupakan rancangan aliran aktivitas atau aliran kerja dalam sebuah sistem yang akan dijalankan. Activity Diagram juga digunakan untuk mendefinisikan atau mengelompokan alur tampilan dari sistem tersebut. *Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah. Berikut adalah *activity diagram* yang terdapat pada sistem

a. *Activity Diagram* Melakukan Login

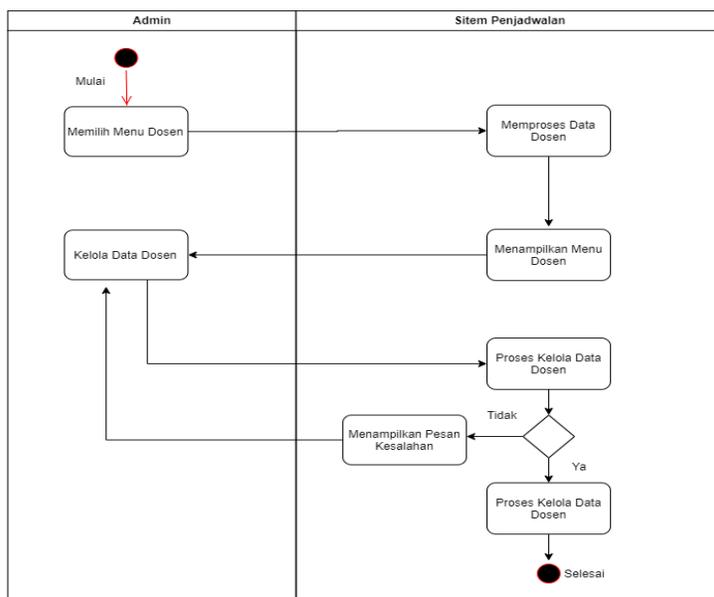
Pada proses *login* secara umum dimana proses diawali dengan mengakses *login* pada sistem kemudian memasukkan *username* dan *password* yang selanjutnya akan diproses oleh sistem jika *username* dan *password valid* maka akan di arahkan ke menu utama tapi jika tidak sesuai maka akan ada notifikasi ketidak sesuaian data baik *usernama* maupun *password*, seperti pada Gambar 4.3.



Gambar 4. 3 Activity Diagram Melakukan Login

b. Activity Diagram Kelola Data Dosen

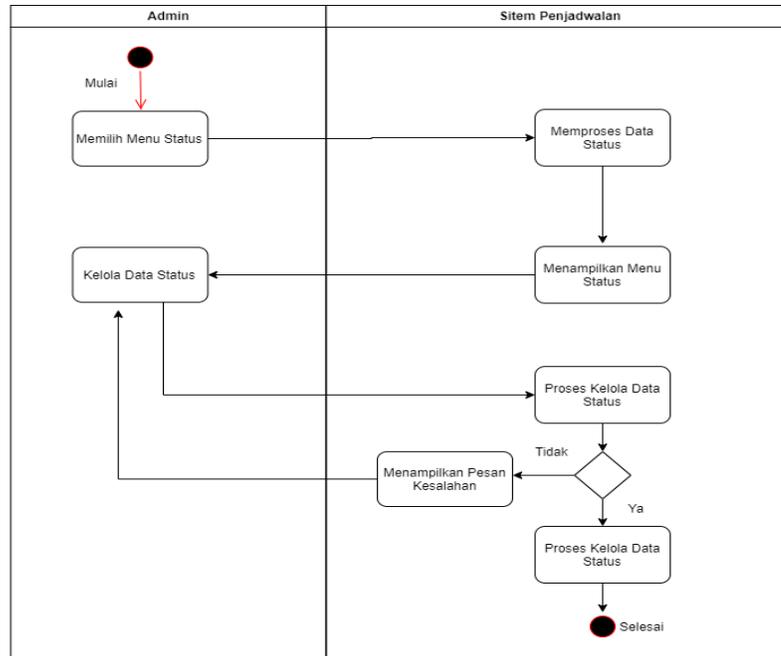
Pada *activity* data dosen, admin dapat mengelola data dosen mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.4.



Gambar 4. 4 Activity Diagram Kelola Data Dosen

c. *Activity Diagram Kelola Data Status Dosen*

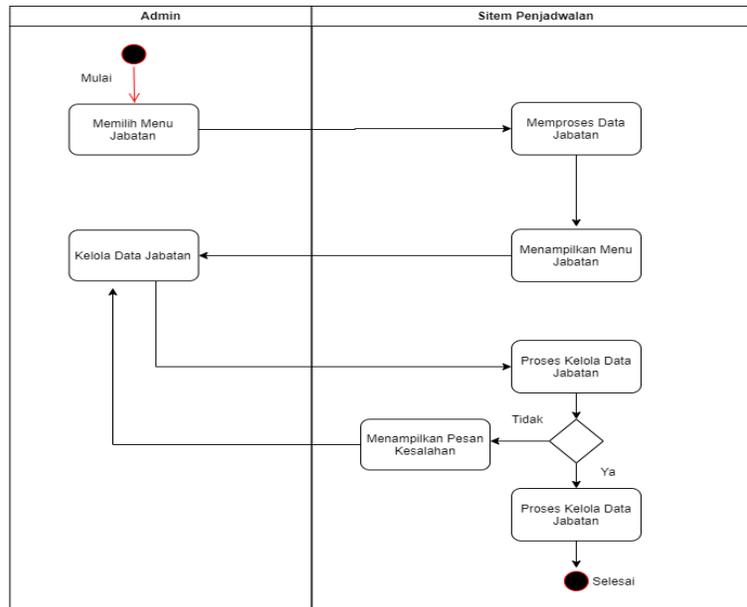
Pada *activity* data status, admin dapat mengelola data status mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.5.



Gambar 4. 5 *Activity Diagram Kelola Data Status*

d. *Activity Diagram Kelola Data Jabatan*

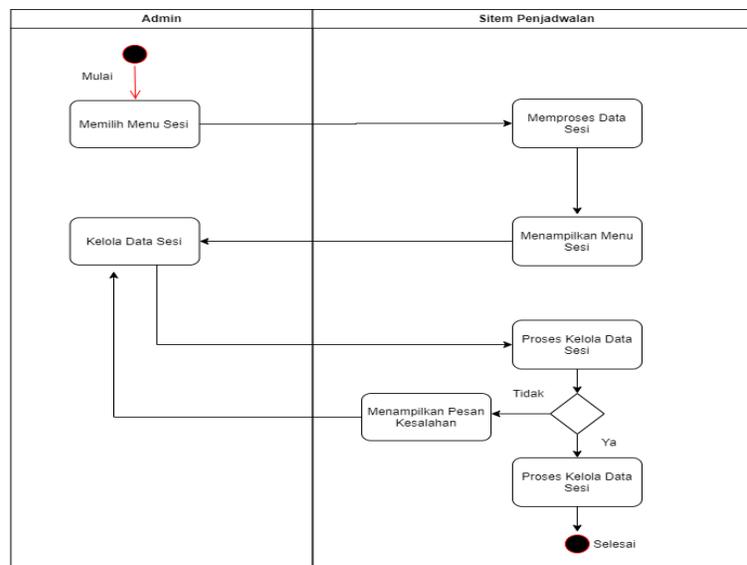
Pada *activity* data jabatan, admin dapat mengelola data jabatan mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.6.



Gambar 4. 6 *Activity Diagram* Kelola Data Jabatan

e. *Activity Diagram* Kelola Data Sesi

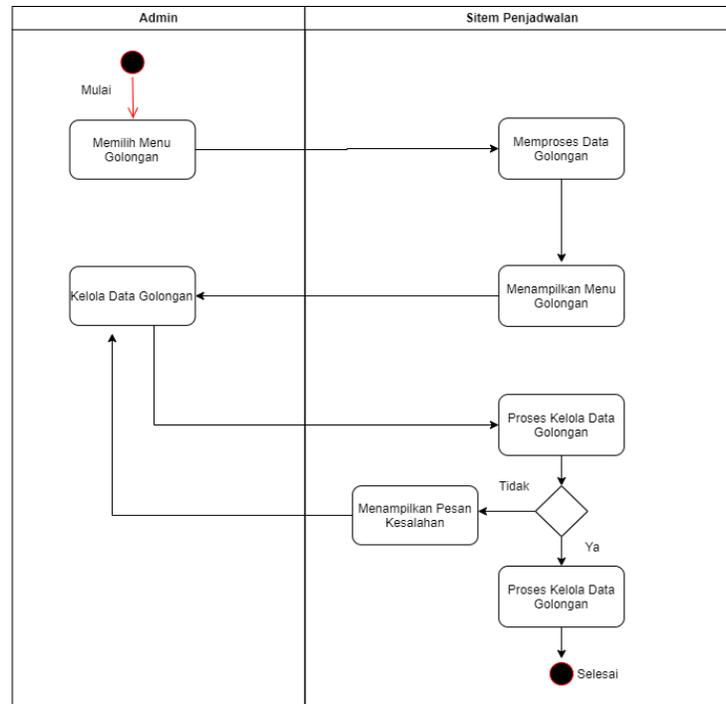
Pada *activity* data sesi, admin dapat mengelola data sesi mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.7.



Gambar 4. 7 *Activity Diagram* Kelola Data Sesi

f. *Activity Diagram* Kelola Data Golongan

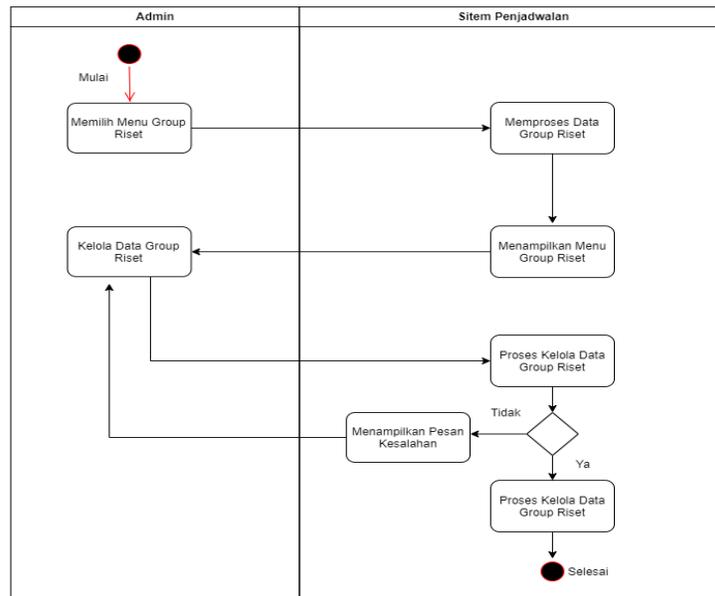
Pada *activity* data golongan, admin dapat mengelola data golongan mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.8.



Gambar 4. 8 *Activity Diagram* Kelola Data Golongan

g. *Activity Diagram* Kelola Data Group Riset

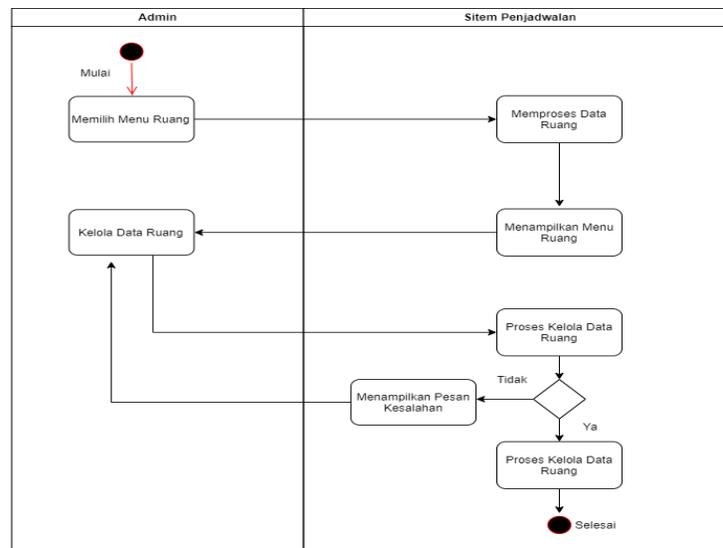
Pada *activity* data golongan, admin dapat mengelola data golongan mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.9.



Gambar 4. 9 Activity Diagram Kelola Data Group Riset

h. Activity Diagram Kelola Data Ruang

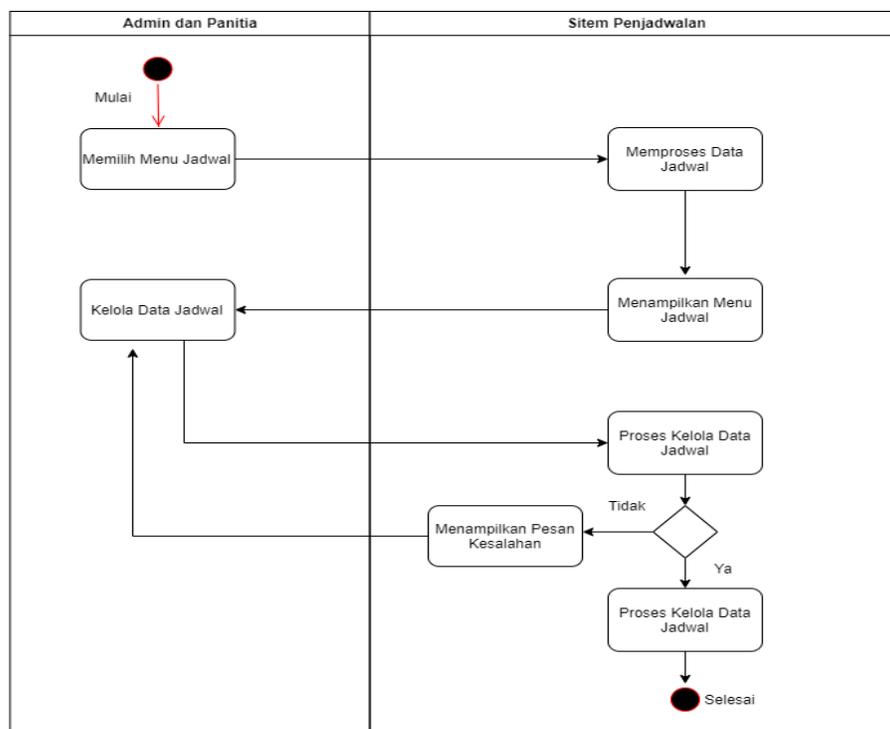
Pada *activity* data ruang, admin dapat mengelola data ruang mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.10.



Gambar 4. 10 Activity Diagram Kelola Data Ruang

i. *Activity Diagram* Kelola Data Jadwal

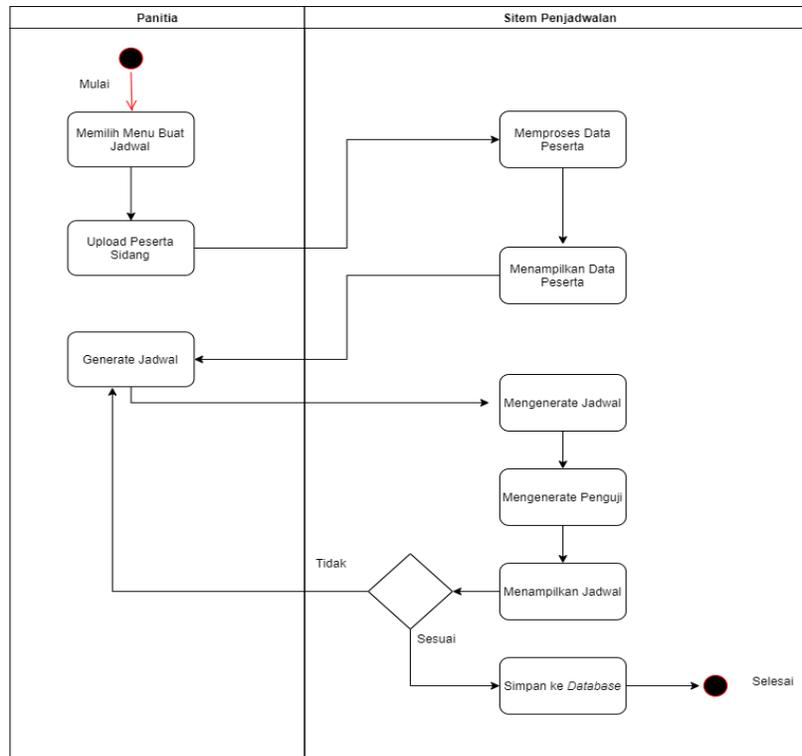
Pada *activity* data jadwal, admin dapat mengelola data jadwal mulai dari menambahkan data baru, mengubah data yang sudah ada dan menghapus data. Dan jika terjadi kesalahan sistem akan mengirim notifikasi kesalahan dan jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.11.



Gambar 4. 11 *Activity Diagram* Kelola Data Jadwal

j. *Activity Diagram* Membuat Jadwal

Pada *activity* membuat, panitia dapat membuat jadwal mulai dari mengimpor data peserta sidang, kemudian *generate* jadwal, dan menyimpan jadwal jika kondisi berhasil maka akan tersimpan pada *database*, seperti pada Gambar 4.12.

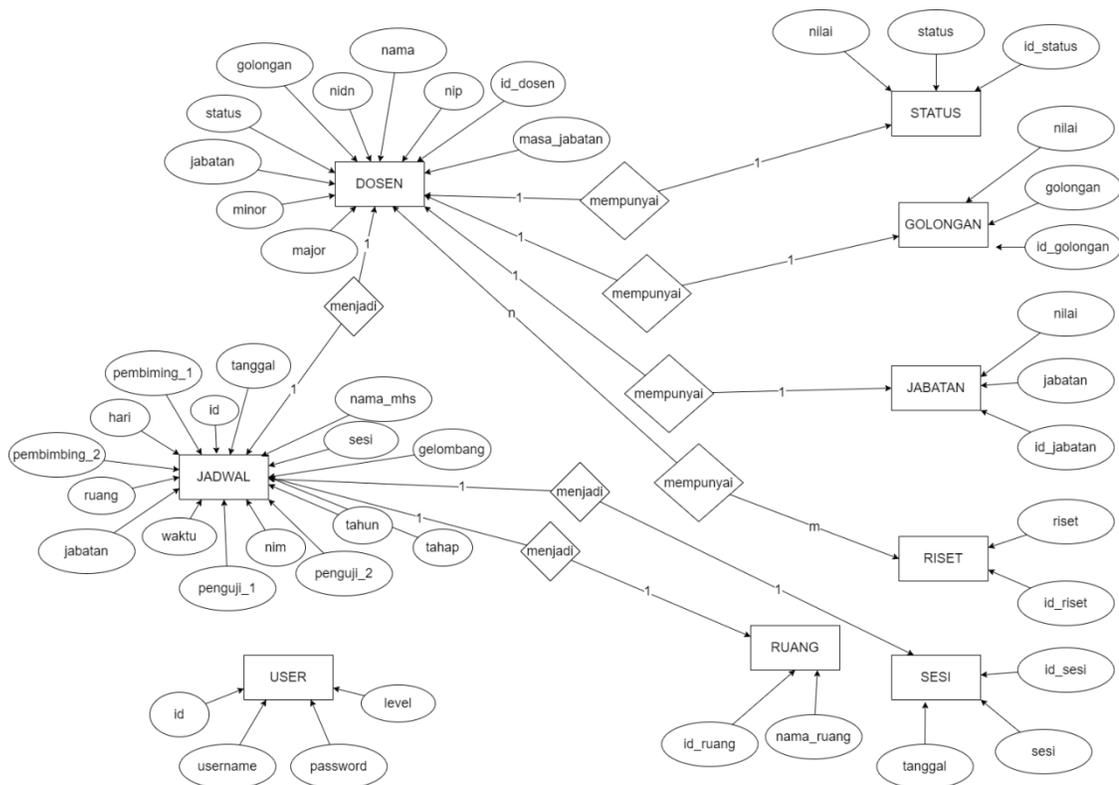


Gambar 4. 12 Activity Diagram Membuat Jadwal

4.2. Perancangan

4.2.1. Perancangan Basis Data

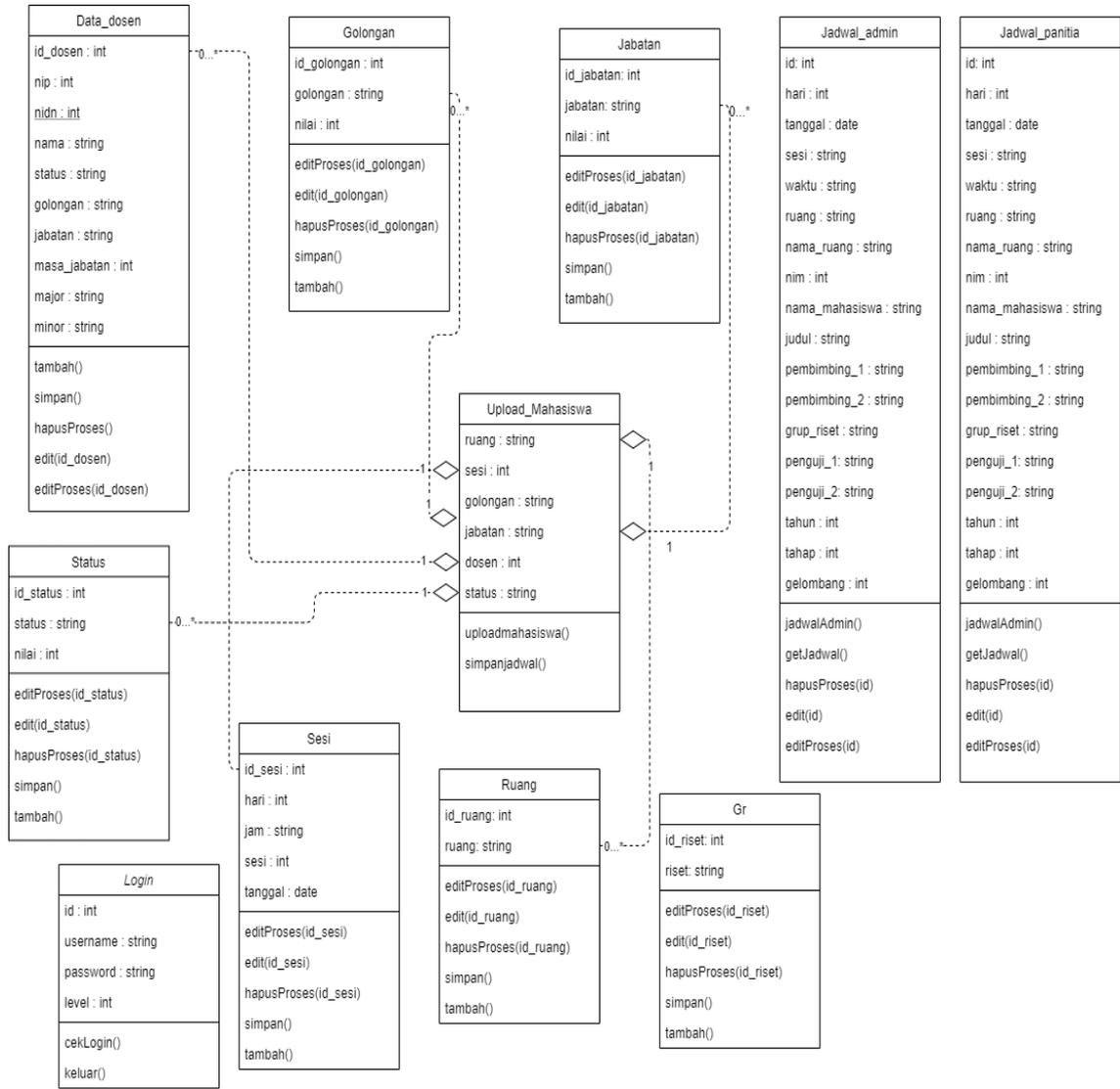
Perancangan basis data pada sistem yang dibangun terdapat 8 (delapan) entitas utama beserta atribut untuk penyimpanan data. Entitas tersebut yaitu data dosen, jabatan, golongan, status dosen, grup riset, ruang, sesi, dan jadwal yang kemudian akan diimplementasikan pada database Berikut *entity relationship diagram* yang terdapat pada Gambar 4.13.



Gambar 4. 13 *Entity Relationship Diagram*

4.2.2. Class Diagram

Class diagram digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem. Diagram kelas memberi gambaran atau diagram statis tentang struktur sistem dan hubungan yang ada didalamnya. Diagram tersebut terdapat pada Gambar 4.14.



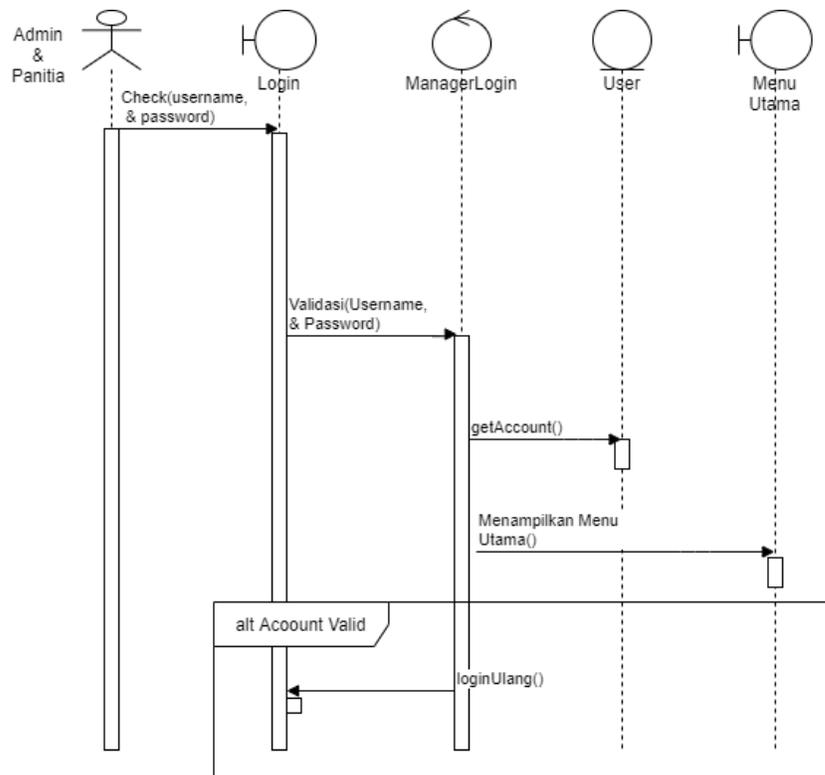
Gambar 4. 14 Class Diagram

4.2.3. Sequence Diagram

Sequence Diagram berguna untuk menggambarkan bentuk interaksi antara kelompok obyek – obyek dengan property di dalam sistem.

a. *Sequence Diagram Melakukan Login*

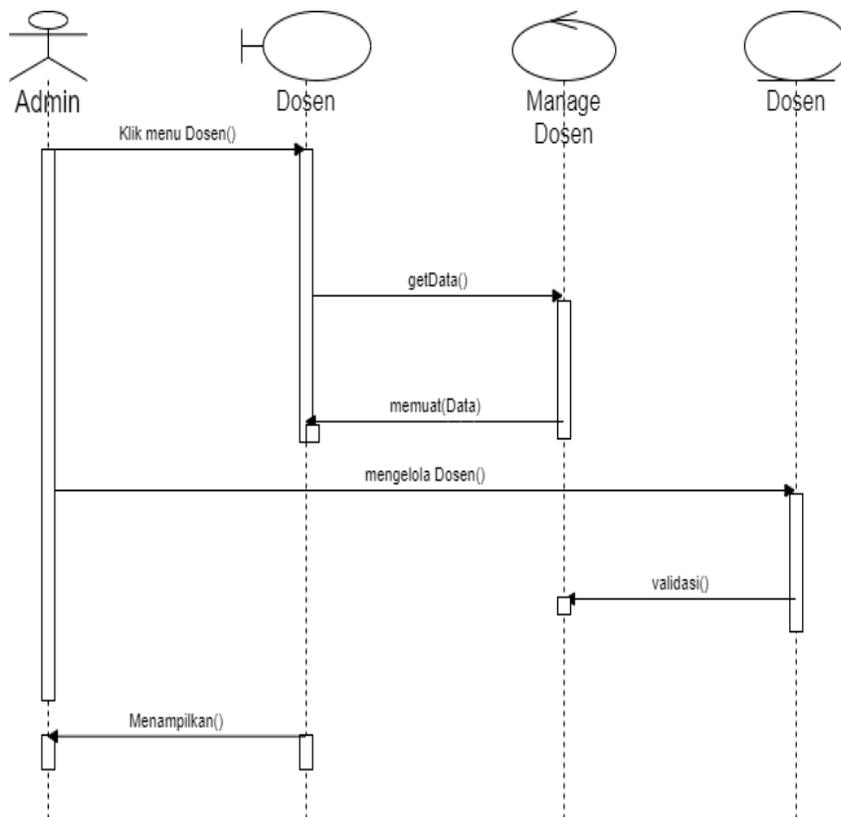
Pada diagram ini, dimulai dari aktor admin memasukan *username* dan *password* pada halaman *login*, lalu diproses oleh admin kontrol untuk mengecek *username* dan *password* pada tabel admin. Jika *username* dan *password* sesuai, maka akan dialihkan pada halaman beranda, seperti yang terdapat pada Gambar 4.15.



Gambar 4. 15 *Sequence Diagram Melakukan Login*

b. *Sequence Diagram Mengelola data dosen*

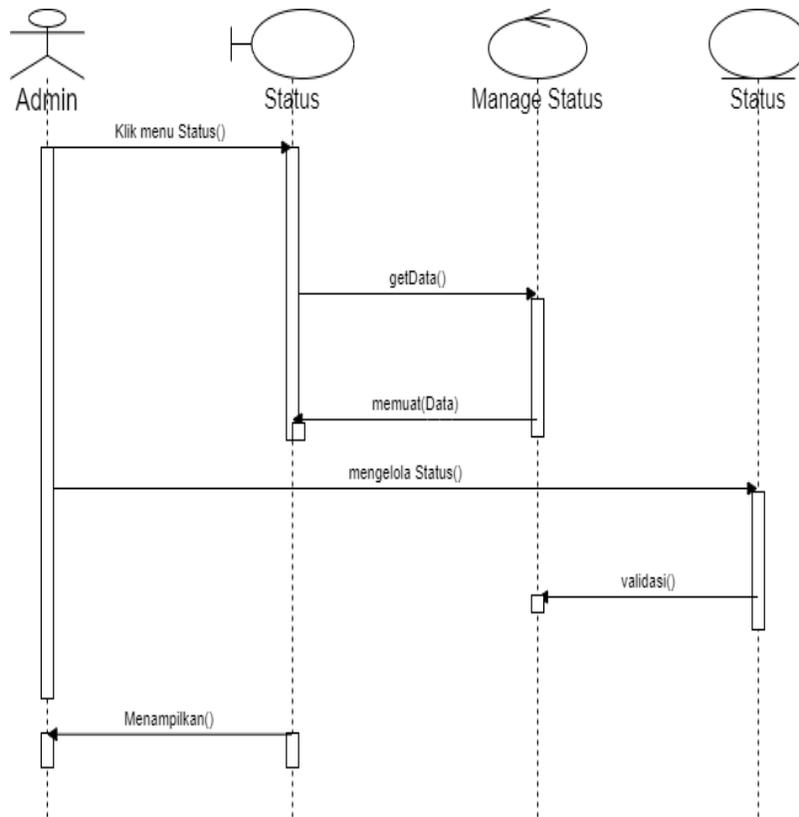
Pada diagram ini dimulai dari aktor admin memilih menu dosen, lalu sistem akan memuat data pada tabel dosen dan menampilkannya pada halaman dosen. Pada halaman tersebut admin dapat menghapus, edit dan menambah dosen, seperti yang terdapat pada Gambar 4.16.



Gambar 4. 16 *Sequence Diagram* Mengelola data Dosen

c. *Sequence Diagram* Mengelola data status

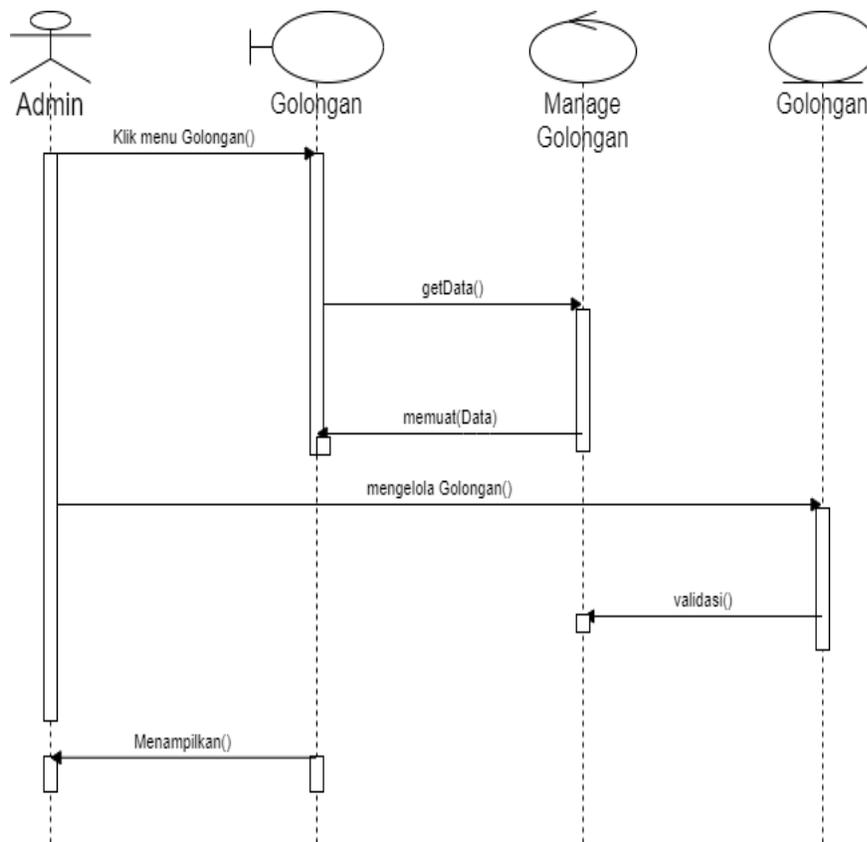
Pada diagram ini dimulai dari aktor admin memilih menu status, lalu sistem akan memuat data pada tabel status dan menampilkannya pada halaman status. Pada halaman tersebut admin dapat menghapus, edit dan menambah status, seperti yang terdapat pada Gambar 4.17.



Gambar 4. 17 *Sequence Diagram* Mengelola data Status

d. *Sequence Diagram* Mengelola data golongan

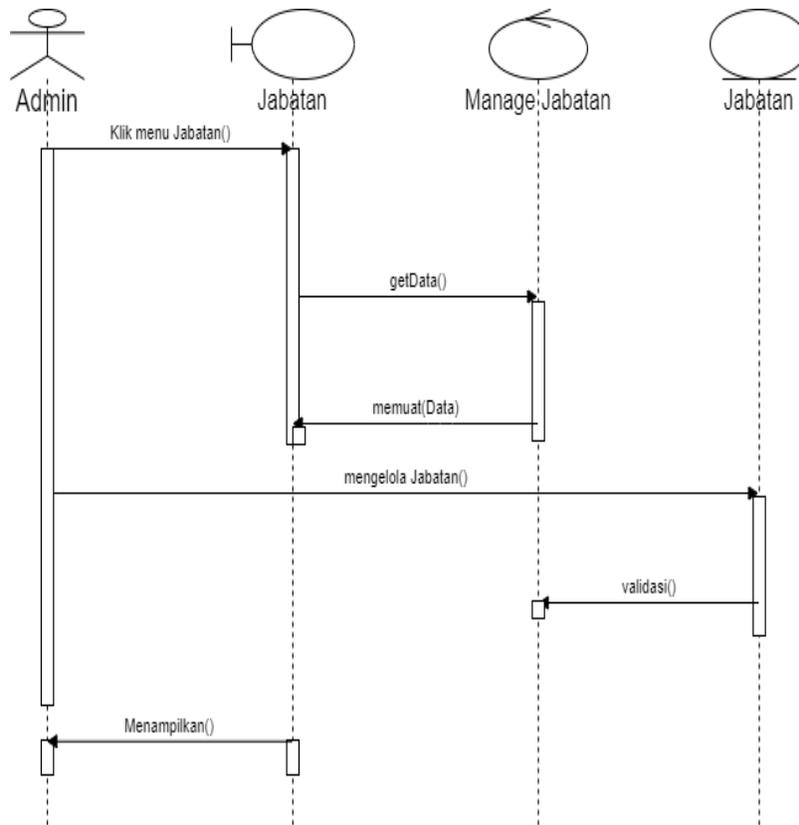
Pada diagram ini dimulai dari aktor admin memilih menu golongan, lalu sistem akan memuat data pada tabel golongan dan menampilkannya pada halaman golongan. Pada halaman tersebut admin dapat menghapus, edit dan menambah golongan, seperti yang terdapat pada Gambar 4.18.



Gambar 4. 18 *Sequence Diagram* Mengelola data Golongan

e. *Sequence Diagram* Mengelola data jabatan

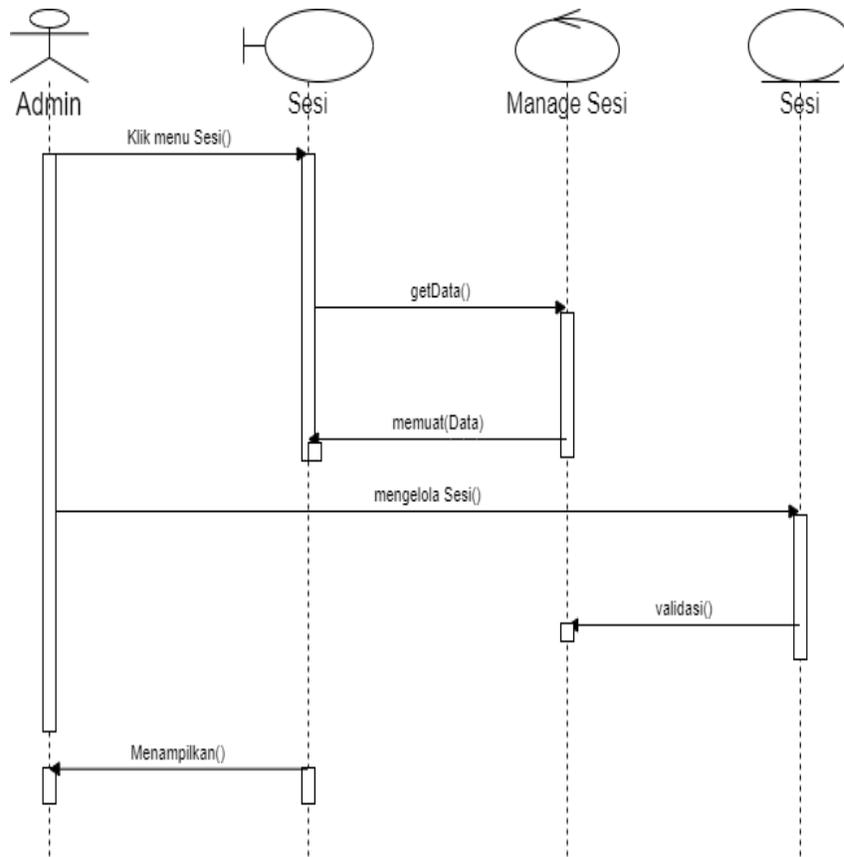
Pada diagram ini dimulai dari aktor admin memilih menu jabatan, lalu sistem akan memuat data pada tabel jabatan dan menampilkannya pada halaman jabatan. Pada halaman tersebut admin dapat menghapus, edit dan menambah jabatan, seperti yang terdapat pada Gambar 4.19.



Gambar 4. 19 *Sequence Diagram* Mengelola data Jabatan

f. *Sequence Diagram* Mengelola data sesi

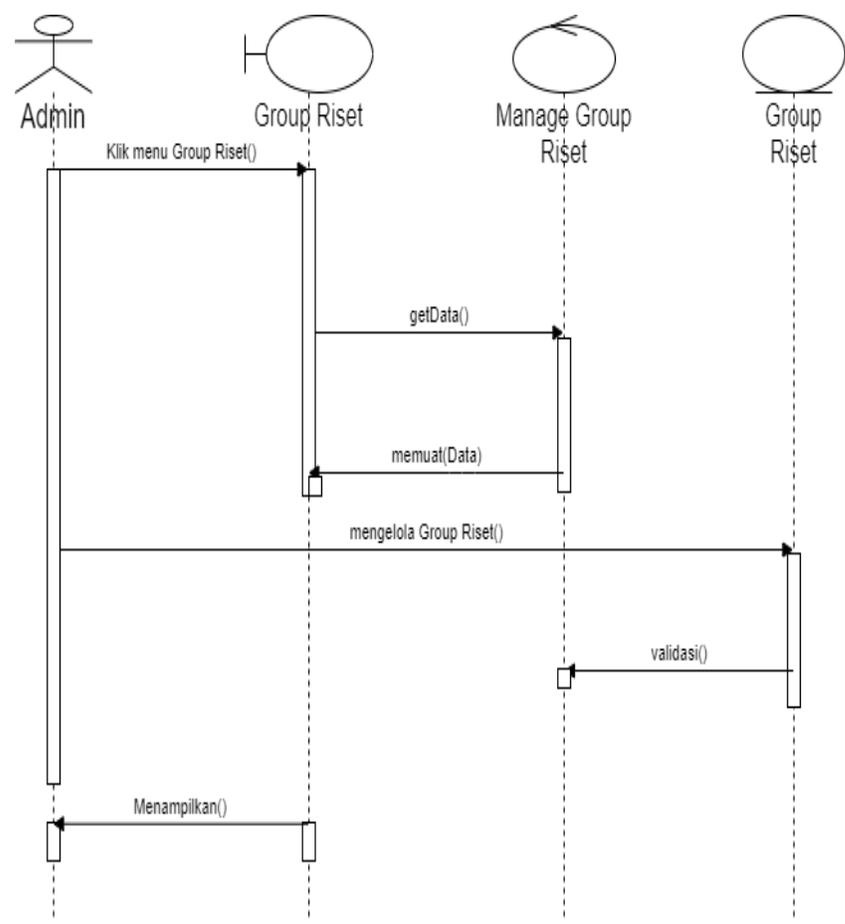
Pada diagram ini dimulai dari aktor admin memilih menu sesi, lalu sistem akan memuat data pada tabel sesi dan menampilkannya pada halaman sesi. Pada halaman tersebut admin dapat menghapus, edit dan menambah sesi, seperti yang terdapat pada Gambar 4.20.



Gambar 4. 20 *Sequence Diagram* Mengelola data Sesi

g. *Sequence Diagram* Mengelola data *group riset*

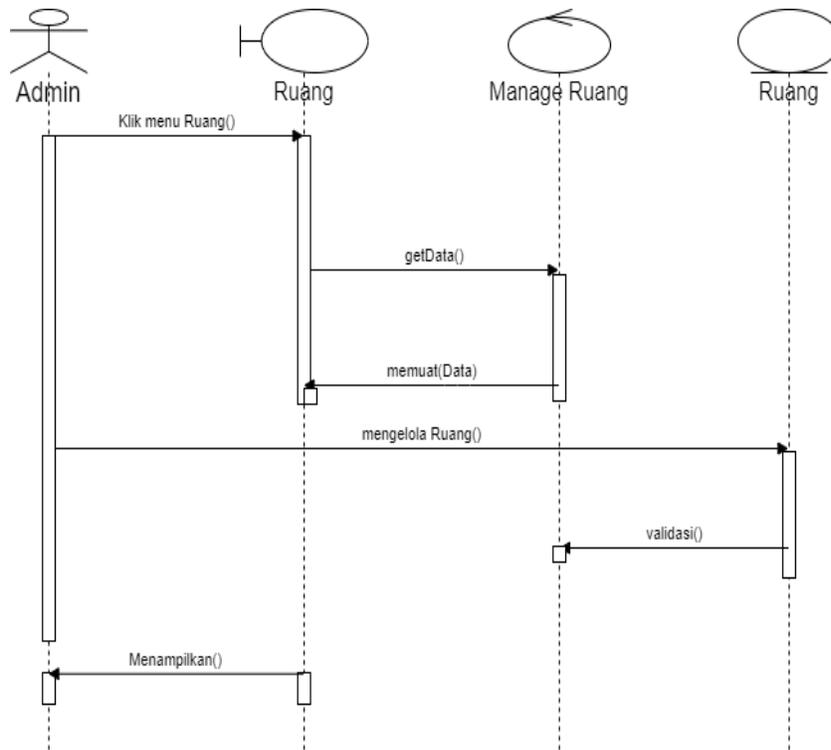
Pada diagram ini dimulai dari aktor admin memilih menu *group riset*, lalu sistem akan memuat data pada tabel *group riset* dan menampilkannya pada halaman *group riset*. Pada halaman tersebut admin dapat menghapus, edit dan menambah *group riset*, seperti yang terdapat pada Gambar 4.21.



Gambar 4. 21 *Sequence Diagram* Mengelola data *Group Riset*

h. *Sequence Diagram* Mengelola data ruang

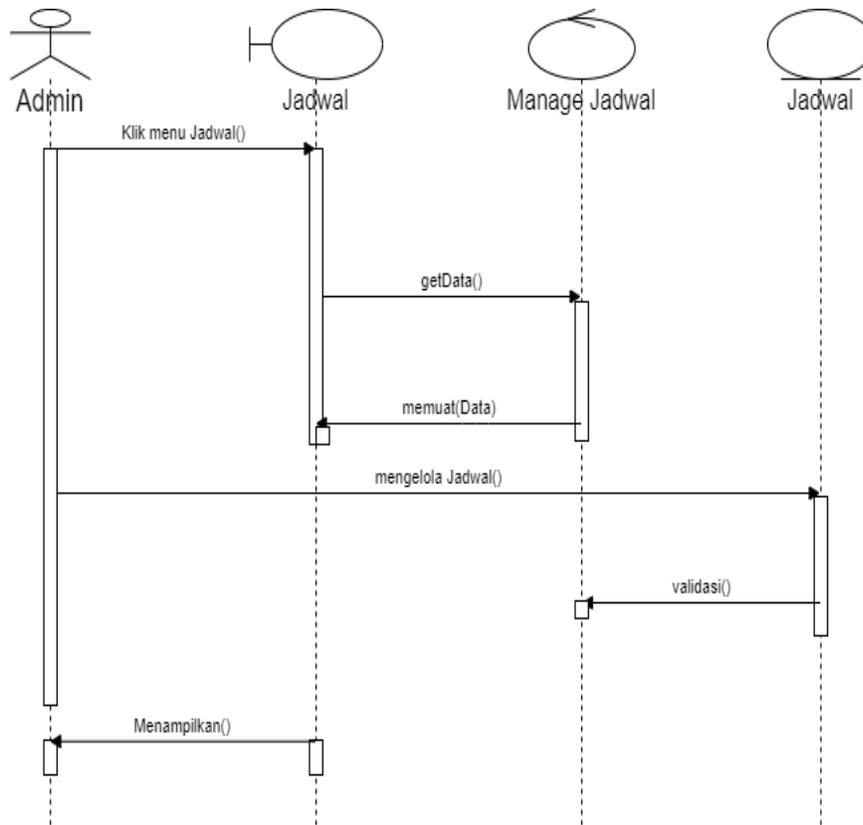
Pada diagram ini dimulai dari aktor admin memilih menu ruang, lalu sistem akan memuat data pada tabel ruang dan menampilkannya pada halaman ruang. Pada halaman tersebut admin dapat menghapus, edit dan menambah ruang, seperti yang terdapat pada Gambar 4.22.



Gambar 4. 22 *Sequence Diagram* Mengelola data Ruang

i. *Sequence Diagram* Mengelola data jadwal

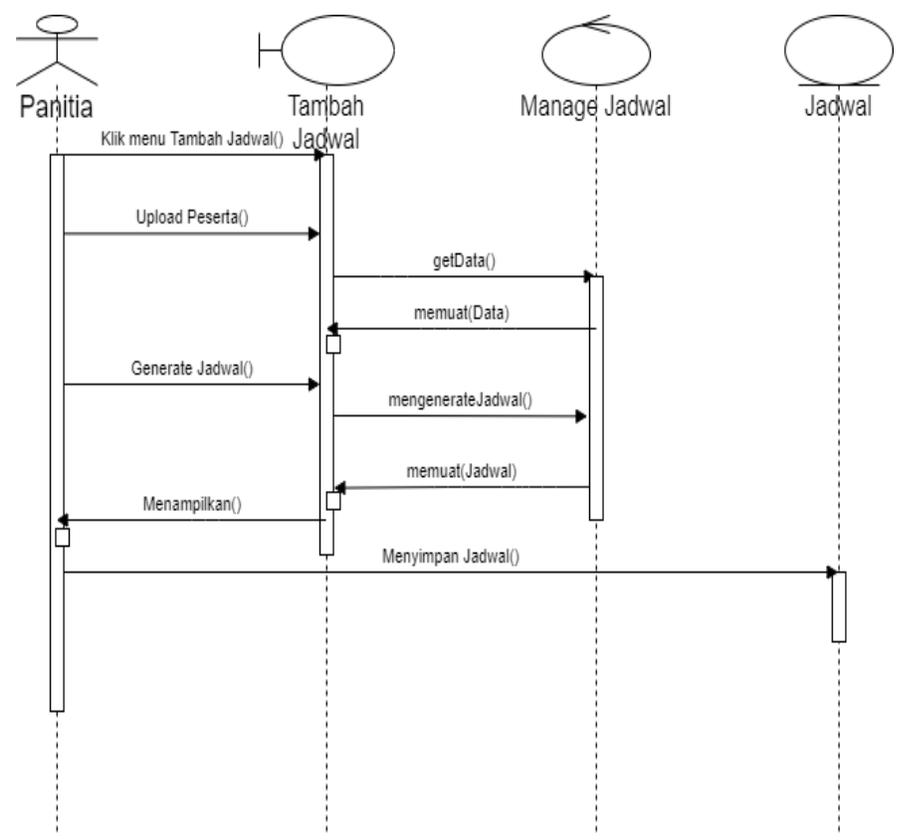
Pada diagram ini dimulai dari aktor admin atau panitia memilih menu jadwal, lalu sistem akan memuat data pada tabel jadwal dan menampilkannya pada halaman jadwal. Pada halaman tersebut admin dapat menghapus, edit dan menambah jadwal, seperti yang terdapat pada Gambar 4.23.



Gambar 4. 23 *Sequence Diagram* Mengelola data Jadwal

j. *Sequence Diagram* Tambah Jadwal

Pada diagram ini dimulai dari aktor panitia memilih menu tambah jadwal, lalu sistem akan memuat form upload peserta, kemudian setelah file peserta berhasil ter-upload selanjutnya panitia melakukan *generate* jadwal, ketika berhasil *generate*, pada halaman tersebut admin dapat menambah jadwal, seperti yang terdapat pada Gambar 4.24.



Gambar 4. 24 *Sequence Diagram* Tambah Jadwal

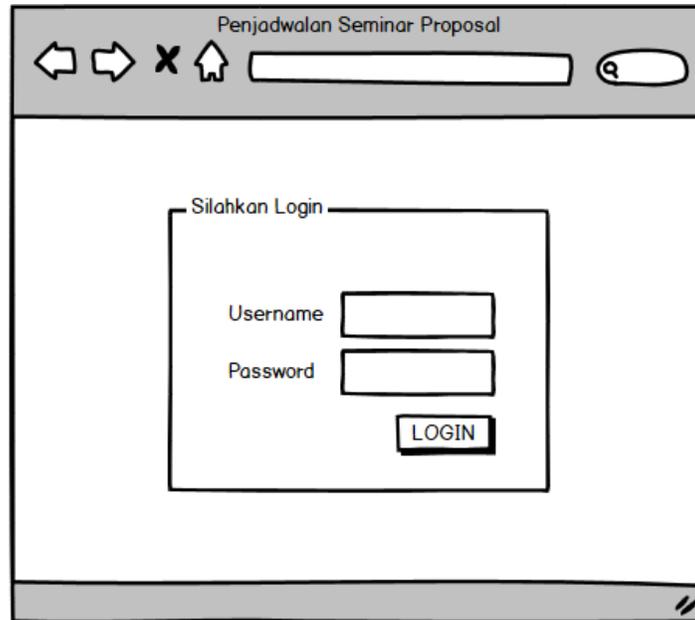
4.2.3. Perancangan Aplikasi

Pada perancangan aplikasi sistem penjadwalan ini akan digambarkan rancangan user interface atau *mockups* sebagai gambaran aplikasi yang akan dibuat sehingga mampu memaksimalkan fitur-fitur yang dibutuhkan sesuai kebutuhan awal dan memudahkan alur dalam pembuatan tanpa harus mendesain ulang ketika proses pembuatan atau implementasi aplikasi dilakukan.

a. Rancangan Tampilan *Login*

Pada rancangan tampilan *login* terdapat fitur *Inputan* username dan password diharapkan admin dan panitia dapat malekaukan akses pada sistem melalui *login* yang tersedia. Dan pada tampilan *login* juga dirancang disertai

session untuk memverifikasi akun, berikut tampilan *mockup* halaman *login* Gambar 4.25.

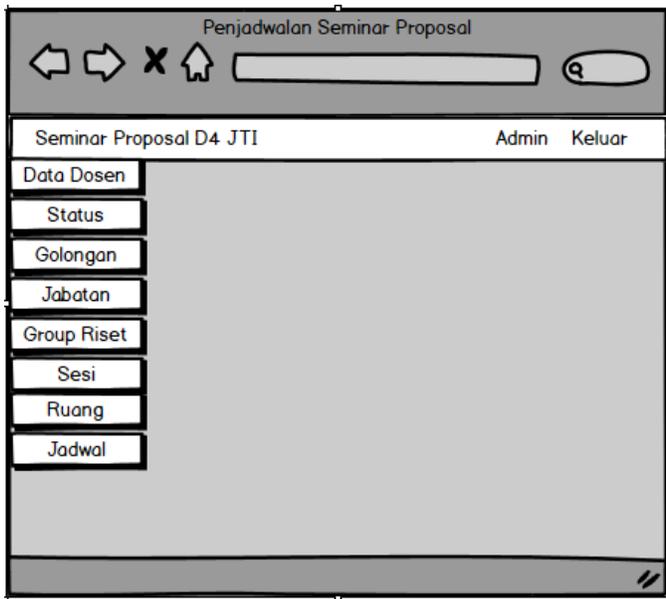


The image shows a browser window mockup for a system titled "Penjadwalan Seminar Proposal". The browser's address bar is empty. The main content area contains a login form with the heading "Silahkan Login". The form includes two input fields: "Username" and "Password", followed by a "LOGIN" button. The browser's navigation icons (back, forward, stop, home) and a search bar are visible in the top left corner.

Gambar 4. 25 *Mockups* Tampilan *Login*.

b. Rancangan Tampilan Admin

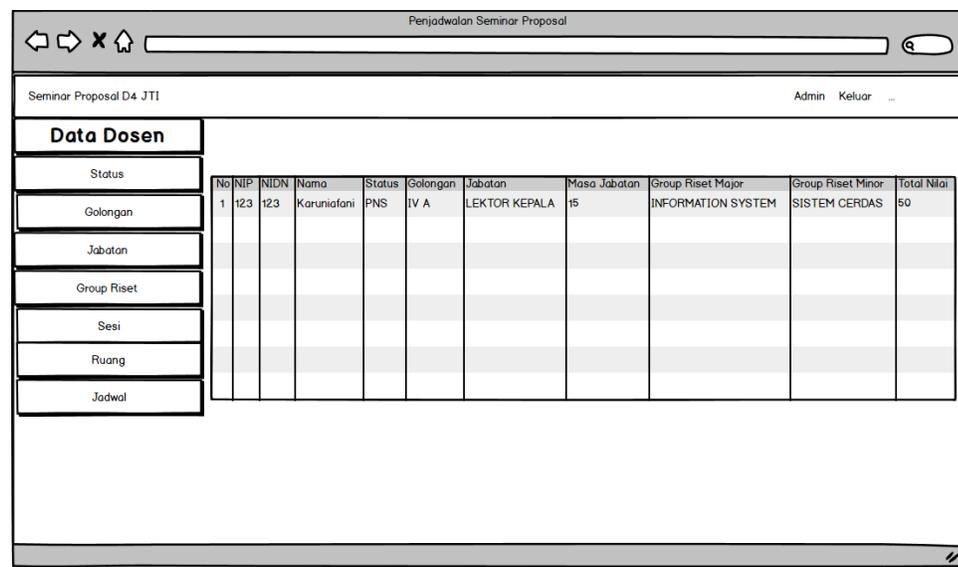
Pada rancangan tampilan admin terdapat fitur Dosen, Status, Jabatan, Golongan, *Group* Riset, Sesi, Ruang, dan Jadwal. Admin dapat malekaukan akses pada sistem melalui *login* yang tersedia. Berikut rancangan *mockups* tampilan halaman admin Gambar 4.26.



Gambar 4. 26 Mockups Tampilan Admin

c. Rancangan Tampilan Data Dosen

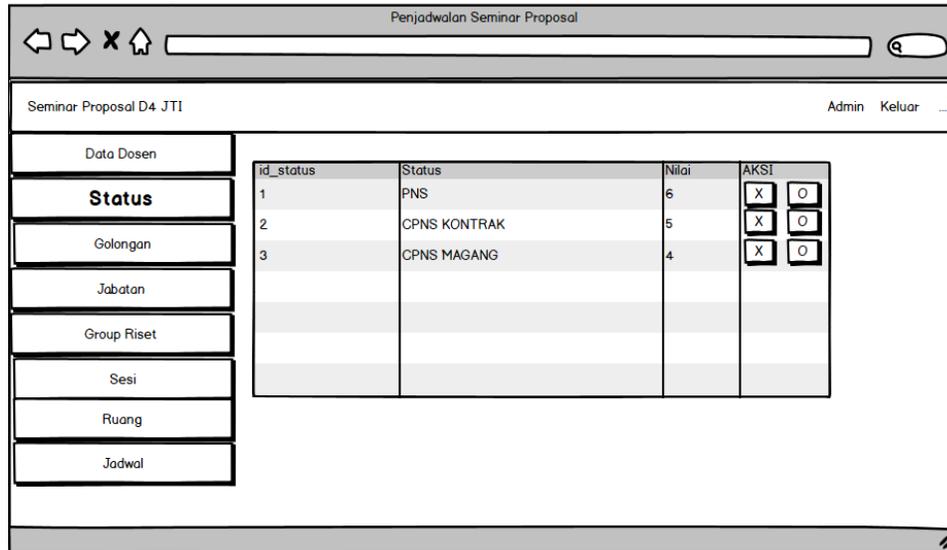
Pada rancangan tampilan data dosen merupakan kumpulan profil dosen yang didapatkan dari hasil wawancara, berikut rancangan tampilan data dosen pada Gambar 4.27.



Gambar 4. 27 Mockups Tampilan data Dosen

d. Rancangan Tampilan Status

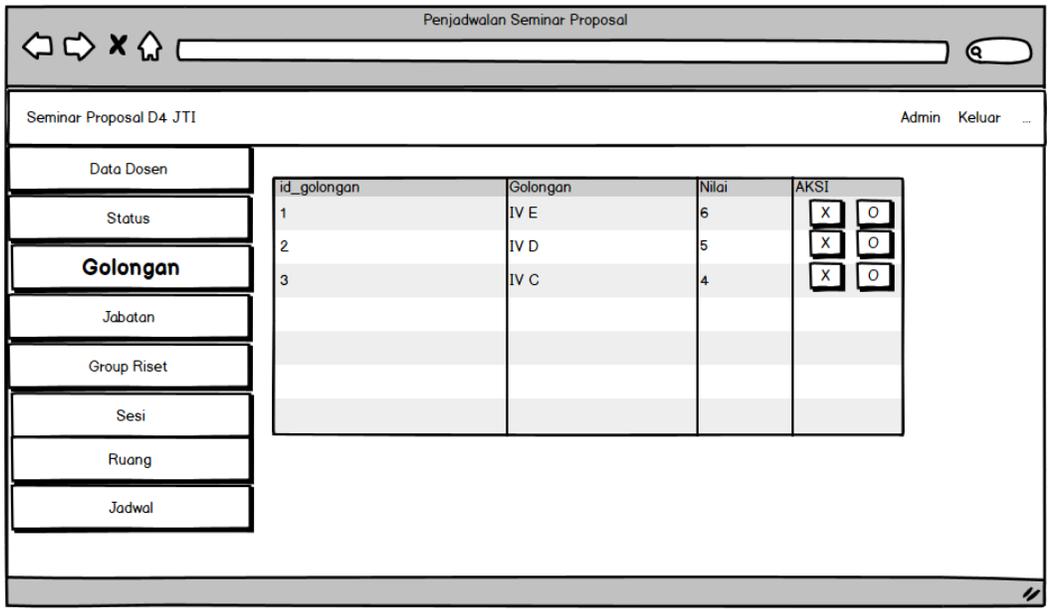
Pada rancangan tampilan data status merupakan kumpulan data status, berikut rancangan tampilan data status pada Gambar 4.28.



Gambar 4. 28 *Mockups* Tampilan data Status

e. Rancangan Tampilan Golongan

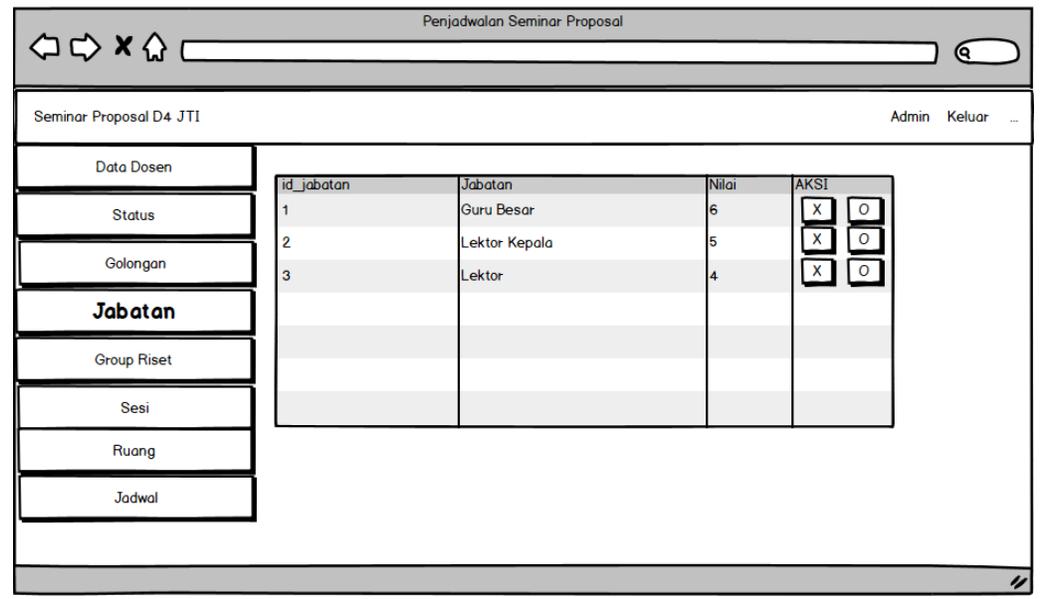
Pada rancangan tampilan data dosen merupakan kumpulan data golongan, berikut rancangan tampilan data golongan pada Gambar 4.29.



Gambar 4. 29 Mockups Tampilan data Golongan

f. Rancangan Tampilan Jabatan

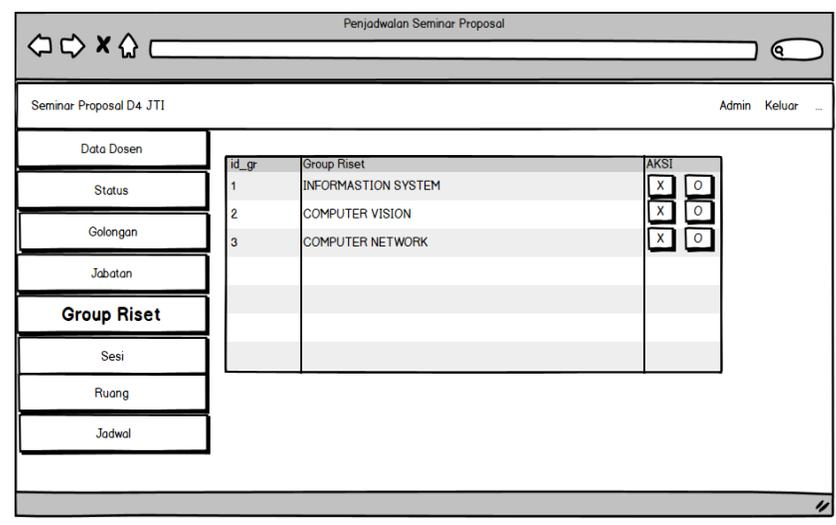
Pada rancangan tampilan data dosen merupakan kumpulan data jabatan, berikut rancangan tampilan data jabatan pada Gambar 4.30.



Gambar 4. 30 Mockups Tampilan data Jabatan

g. Rancangan Tampilan Group Riset

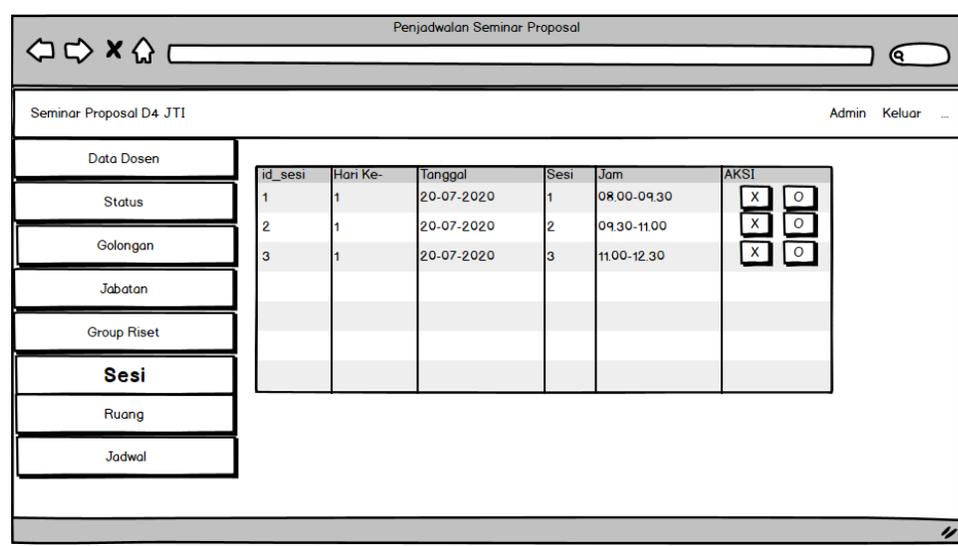
Pada rancangan tampilan data dosen merupakan kumpulan data *group* riset, berikut rancangan tampilan data *group* riset pada Gambar 4.31.



Gambar 4. 31 *Mockups* Tampilan *Group* Riset

h. Rancangan Tampilan Sesi

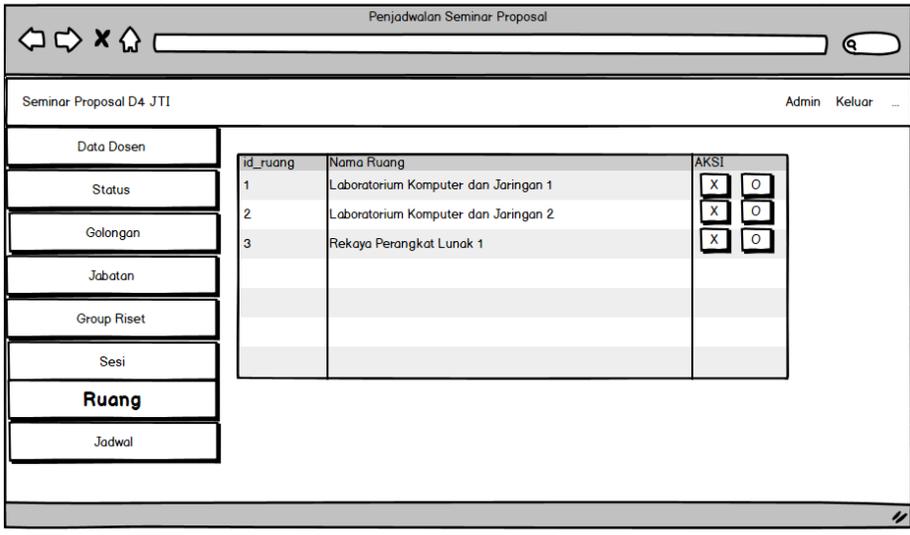
Pada rancangan tampilan data dosen merupakan kumpulan data sesi yang tersedia untuk dilakukan ujian, berikut rancangan tampilan data sesi pada Gambar 4.32.



Gambar 4. 32 *Mockups* Tampilan data Sesi

i. Rancangan Tampilan Ruang

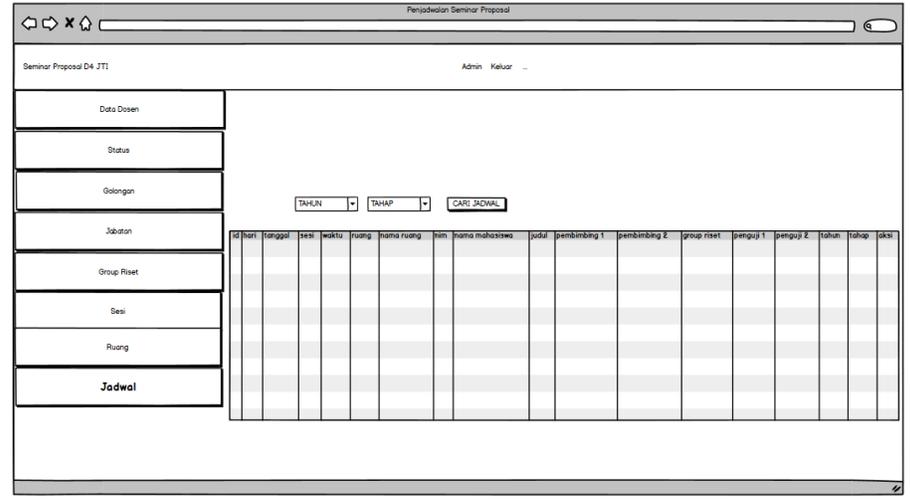
Pada rancangan tampilan data ruang merupakan kumpulan data ruang yang tersedia untuk dilakukan ujian, berikut rancangan tampilan data ruang pada Gambar 4.33.



Gambar 4. 33 Mockups Tampilan data Ruang

j. Rancangan Tampilan Jadwal

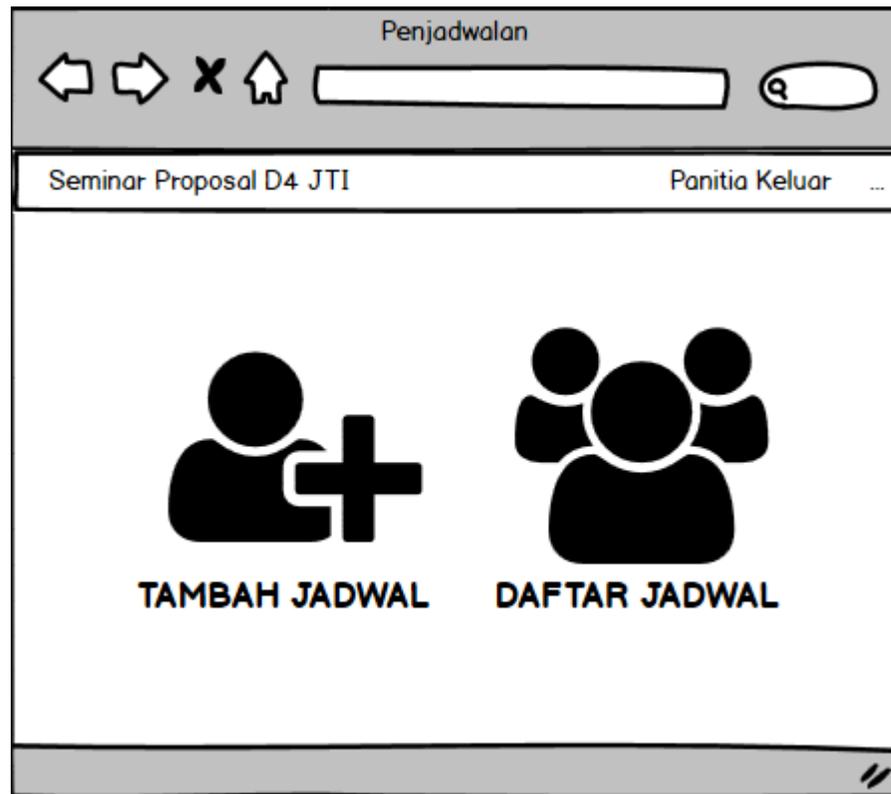
Pada rancangan tampilan data jadwal merupakan kumpulan jadwal yang telah berhasil di *generate* dan di simpan pada database, berikut rancangan tampilan data jadwal pada Gambar 4.34.



Gambar 4. 34 Mockups Tampilan data Jadwal

k. Rancangan Halaman Panitia

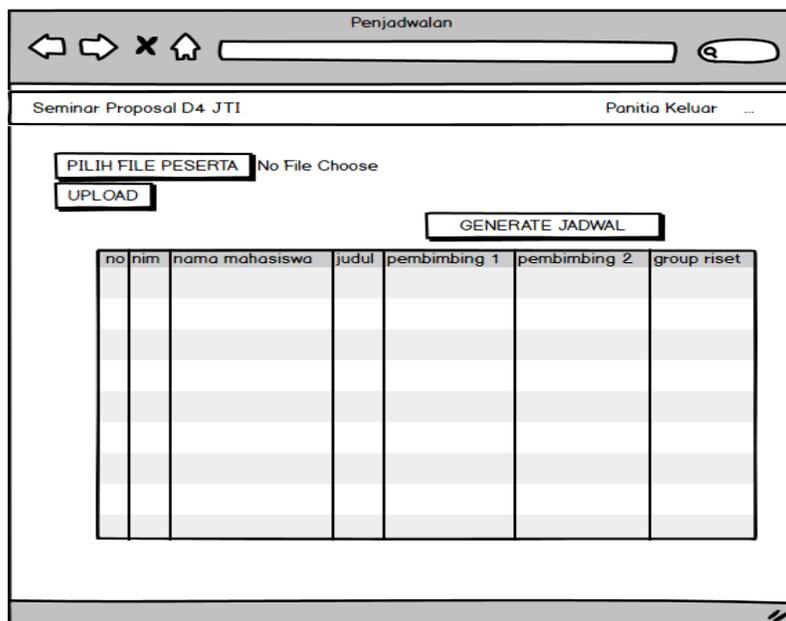
Pada rancangan tampilan panitia terdapat fitur Tambah Jadwal, dan Daftar Jadwal. Panitia dapat melakukan akses pada sistem melalui *login* yang tersedia. Berikut rancangan *mockups* tampilan halaman admin Gambar 4.35.



Gambar 4. 35 *Mockups* Tampilan Panitia

l. Rancangan Halaman Tambah Jadwal

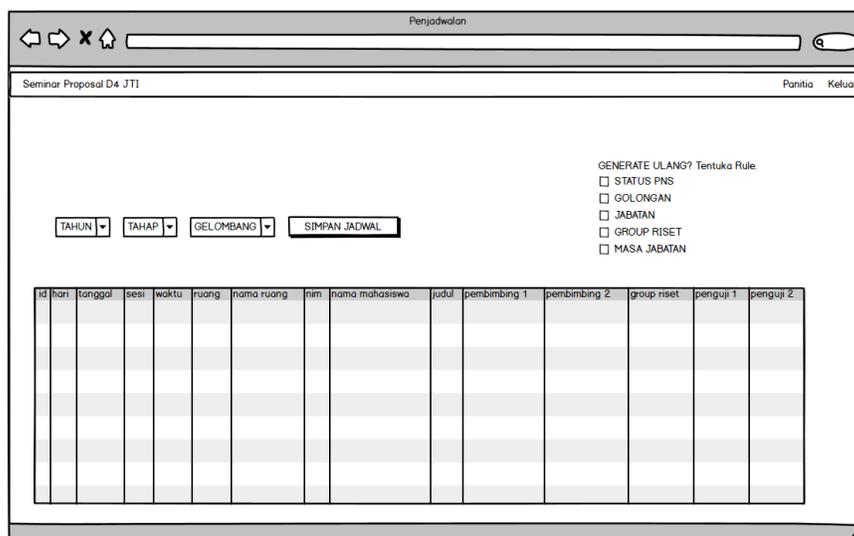
Pada rancangan tampilan tambah jadwal, terdapat fitur upload peserta ujian dan fitur *generate* jadwal. Berikut rancangan *mockups* tampilan halaman tambah jadwal Gambar 4.36.



Gambar 4. 36 *Mockups* Tampilan Tambah Jadwal

m. Rancangan Halaman Hasil *Generate* Jadwal

Pada rancangan tampilan hasil *generate*, terdapat tampilan hasil *generate* jadwal, dan fitur *generate* ulang berdasarkan rule yang ingin digunakan, serta fitur simpan jadwal, berikut rancangan *mockups* tampilan hasil *generate* jadwal Gambar 4.37.



Gambar 4. 37 *Mockups* Tampilan Hasil *Generate* Jadwa