

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

Untuk


Aplikasi Peminjaman Ruangan dan Asset UMRAH



Disusun oleh:

2201020103	Muhammad Chandra Ramadhan
2201020099	Syawal Nopyandi
2201020048	Janumelah
2201020018	Aditya Firmansyah
2201020011	Muhammad Zaidan Nugroho

**Program Studi Teknik Informatika
Fakultas Teknik dan Teknologi Kemaritiman
Universitas Maritim Raja Ali Haji (UMRAH)
2024**

	Program Studi Teknik Informatika FT - UMRAH	Spesifikasi Kebutuhan Perangkat Lunak Aplikasi Peminjaman Ruangan dan Asset UMRAH	Tgl: 07 November 2024
---	--	--	----------------------------------

DAFTAR ISI

DAFTAR ISI.....	2
DAFTAR GAMBAR.....	5
DAFTAR TABEL.....	7
I. PENDAHULUAN.....	8
1.1. Tujuan Penulisan Dokumen.....	8
1.2. Ruang Lingkup.....	8
1.3. Definisi, Istilah, dan Singkatan.....	8
1.4. Referensi.....	9
1.5. Ikhtisar Dokumen (Deskripsi Umum Dokumen).....	10
II. DESKRIPSI UMUM PERANGKAT LUNAK.....	10
2.1. Deskripsi Umum Sistem.....	10
2.2. Fungsi Produk.....	10
2.3. Karakteristik Pengguna.....	10
2.4. Lingkungan Operasi.....	10
III. KEBUTUHAN SPESIFIK.....	11
IV. DESKRIPSI RINCI PERANCANGAN.....	15
4.1. Deskripsi Rinci Tabel.....	15
4.1.1. Tabel <users>.....	15
4.1.2. Tabel <admin>.....	16
4.1.3. Tabel <sessions>.....	16
4.1.4. Tabel <rooms>.....	17

4.1.5. Tabel <assets>.....	17
4.1.6. Tabel <students>.....	18
4.1.7. Tabel <faculty>.....	18
4.1.8. Tabel <major>.....	19
4.1.9. Tabel <requests>.....	19
4.1.10. Tabel <request_type>.....	20
4.1.11. Tabel <request_status>.....	20
4.2. Entity Relationship Diagram.....	21
4.3. Denormalisasi.....	21
4.3.1 Denormalisasi - Data 1.....	22
4.3.2 Denormalisasi - Data 2.....	24
4.4. Relasi Antar Tabel.....	25
4.5. Class Diagram.....	26
4.5.1. Class Diagram.....	26
4.6. Activity Diagram.....	27
4.6.1. Activity Diagram Register.....	27
4.6.3. Activity Diagram Peminjaman.....	29
4.6.4. Activity Diagram Pengembalian.....	30
4.6.5. Activity Diagram Tambah Ruangan.....	31
4.6.6. Activity Diagram Tambah Asset.....	31
4.6.7. Activity Diagram Edit Ruangan.....	32
4.6.8. Activity Diagram Edit Asset.....	32
4.7. Collaboration Diagram.....	33
4.7.1. Pengelolaan Ruangan dan Aset.....	33
4.7.2. Peminjaman Ruangan dan Aset.....	34
4.8. Deskripsi Fungsional.....	35
4.8.1. Context Diagram.....	35

4.8.2. DFD Level 1.....	36
4.9. Antarmuka Pengguna.....	38
4.10. Spesifikasi Field Data Layer.....	47
4.11. Pengujian Sistem.....	50
LAMPIRAN.....	58

DAFTAR GAMBAR

Gambar 1. Entity Relationship Diagram	21
Gambar 2. Denormalisasi - Data 1	22
Gambar 3. Denormalisasi - Data 2	24
Gambar 4. Class Diagram	26
Gambar 5. Activity Diagram Register	27
Gambar 6. Activity Diagram Login	28
Gambar 7. Activity Diagram Peminjaman	29
Gambar 8. Activity Diagram Pengembalian	30
Gambar 9. Activity Diagram Tambah Ruangan	31
Gambar 10. Activity Diagram Tambah Asset	31
Gambar 11. Activity Diagram Edit Ruangan	32
Gambar 12. Activity Diagram Edit Asset	32
Gambar 13. Collaboration Diagram Pengelolaan Ruangan dan Aset	33
Gambar 14. Collaboration Diagram Peminjaman Ruangan dan Aset	34
Gambar 15. Context Diagram	35
Gambar 16. DVD lv 1.....	36
Gambar 17. Flowchart	37
Gambar 18. Tampilan Beranda	38
Gambar 19. Tampilan Beranda 2	39
Gambar 20. Tampilan Register	40
Gambar 21. Tampilan Login	40
Gambar 22. Tampilan Peminjaman Ruangan dan Asset	41
Gambar 23. Tampilan Ruangan 1	42
Gambar 24. Tampilan Halaman Profile User	42
Gambar 25. Tampilan Menu Detail Peminjaman	43

Gambar 26. Tampilan Menu Update Profile User	43
Gambar 27. Tampilan Menu Dashboard Admin.....	44
Gambar 28. Tampilan Menu Peminjaman Admin.....	44
Gambar 29. Tampilan Menu Pengembalian Admin.....	45
Gambar 30. Tampilan Menu Ruangan Admin.....	45
Gambar 31. Tampilan Menu Asset Admin.....	46
Gambar 32. Tampilan Kontak	46

DAFTAR TABEL

Tabel 1. Definisi.....	8
Tabel 2. Deskripsi Rinci.....	15
Tabel 3. Spesifikasi Field Data Layer.....	47
Tabel 4. Hasil Pengujian Black Box.....	50

I. PENDAHULUAN

1.1. Tujuan Penulisan Dokumen

Dokumen ini dibuat dengan tujuan sebagai dasar perancangan terhadap kebutuhan aplikasi yang akan dikembangkan, dan diharapkan dapat digunakan oleh pengembang sebagai pedoman dan juga panduan dalam merancang aplikasi atau sistem yang akan dikembangkan.

1.2. Ruang Lingkup

Aplikasi Peminjaman Ruangan dan Aset UMRAH yang akan dirancang merupakan perangkat lunak berbasis website yang bertujuan untuk memenuhi kebutuhan pelayanan dan manajemen peminjaman ruangan dan asset UMRAH dalam lingkup fakultas sehingga dapat membantu kualitas pelayanan di lingkungan fakultas baik terhadap mahasiswa maupun staff yang bertugas.

1.3. Definisi, Istilah, dan Singkatan

Berikut adalah tabel yang menyertakan semua definisi, istilah, dan singkatan yang digunakan dalam dokumen ini.

Tabel 1. Definisi

Istilah	Definisi
User	Seseorang yang berinteraksi dengan aplikasi mobile/web
Admin/Administrator	Administrator sistem yang diberikan hak spesifik untuk mengontrol sistem
Asset	Sumber daya apa pun yang dimiliki atau dikendalikan oleh bisnis atau entitas ekonomi.
Framework	Kerangka kerja yang mencakup library/perpustakaan kode, dan model perangkat lunak.
Frontend	Bagian dari pengembangan aplikasi yang berfokus pada tampilan dan interaksi antarmuka.

Backend	Bagian dari pengembangan aplikasi yang berfokus dalam menyediakan kebutuhan yang prosesnya tak terlihat oleh pengguna/interaksi secara tidak langsung.
Client	Bagian dari aplikasi yang berinteraksi langsung dengan pengguna, biasanya berupa antarmuka pengguna di browser atau aplikasi.
HTTP	<i>Hypertext Transfer Protocol</i>
JSON	<i>JavaScript Object Notation</i>
CSS	<i>Cascading Style Sheet</i>
DESC	<i>Description</i>
RAT	<i>Rational</i>
DEP	<i>Dependency / ketergantungan</i>
SKPL	<i>Spesifikasi Kebutuhan Perangkat Lunak</i>
API	<i>Application Programming Interface</i>
DBMS	<i>Database Management System</i>
Web	<i>Website</i>

1.4. Referensi

- [1] IEEE Software Engineering Standards Committee, "IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications", October 20, 1998.
- [2] Feldt R, "re_lecture5b_100914", unpublished.

1.5. Ikhtisar Dokumen (Deskripsi Umum Dokumen)

Dokumen ini bertujuan untuk membahas spesifikasi yang terperinci terhadap aplikasi Peminjaman Ruangan dan Asset UMRAH yang akan dikembangkan untuk memenuhi tugas mata kuliah Perancangan dan Implementasi Perangkat Lunak (PIPL) dan Pemrograman Web.

II. DESKRIPSI UMUM PERANGKAT LUNAK

2.1. Deskripsi Umum Sistem

Sistem yang dirancang merupakan perangkat lunak berbasis website yang terdiri dari Backend-server sebagai penyedia sumber daya dan kontrol terhadap sistem, serta Frontend yang berfungsi sebagai antarmuka yang digunakan oleh user untuk berinteraksi dengan sistem, kedua sistem ini akan berjalan pada satu server utama sehingga aplikasi dapat berjalan lebih optimal dan efisien.

2.2. Fungsi Produk

Aplikasi yang akan dikembangkan bertujuan untuk memenuhi kebutuhan akan peminjaman ruangan dan aset UMRAH secara sistematis dan mudah untuk digunakan.

2.3. Karakteristik Pengguna

Pengguna terdiri dari 2 kelas yaitu mahasiswa sebagai pengguna yang dapat mengakses fitur layanan pada aplikasi, serta staff atau admin sebagai pengelola sistem dan juga yang bertugas melayani permintaan pengguna yang masuk.

2.4. Lingkungan Operasi

Perangkat lunak yang dikembangkan dirancang dapat berjalan dalam lingkungan operasi Windows dengan minimum versi 10 atau Linux pada sisi backend serta Node.js sebagai framework yang bertugas menjalankan program pada sisi Backend. Pada sisi client atau frontend aplikasi dirancang untuk berjalan pada lingkungan browser seperti Google Chrome, Firefox dan browser lainnya yang mendukung bahasa pemrograman seperti HTML 5, CSS dan juga javascript sebagai client-side scripting. MongoDB juga dibutuhkan pada sisi backend sebagai DBMS yang bertugas sebagai database manager.

III. KEBUTUHAN SPESIFIK

Bagian ini berisi seluruh kebutuhan fungsionalitas dan kualitas sistem yang dijabarkan secara rinci.

3.1. Kebutuhan Antarmuka Eksternal

Data keluaran dan masukkan yang diterima serta dikirimkan oleh client dan server akan dikemas dalam bentuk JavaScript Object Notation (JSON) melalui protokol Hypertext Transfer Protocol (HTTP) dengan menerapkan konsep REST API yang mengutamakan keterbacaan, skalabilitas, dan kesederhanaan, sehingga data akan lebih mudah untuk diterima dan diproses baik untuk penggunaan internal maupun eksternal.

3.1.1. Antarmuka Perangkat Keras

Karena aplikasi seluler maupun portal web tidak memiliki perangkat keras khusus, maka tidak ada antarmuka perangkat keras.

3.1.2. Antarmuka Perangkat Lunak

Aplikasi peminjaman ruangan dan aset UMRAH digunakan untuk mendapatkan informasi mengenai ketersediaan, dan penggunaan terhadap ruangan maupun aset melalui database. Komunikasi antara database dan web memiliki operasi yang berkaitan dengan membaca, dan memodifikasi data. Terdapat hak akses yang dikhususkan untuk staff/admin yang dapat memungkinkan melakukan operasi terhadap database seperti membaca, menambah, menghapus, serta memodifikasi data.

3.1.3. Antarmuka Komunikasi

Komunikasi antar sistem akan dilakukan menggunakan protokol HTTP dengan antarmuka REST API sebagai media untuk melakukan pertukaran data atau resource.

3.2. Kebutuhan Fungsional

Bagian berikut mencakup semua kebutuhan yang menentukan semua tindakan mendasar dari sistem perangkat lunak yang dikembangkan.

3.2.1. User Class 1 – User

3.2.1.1. *Kebutuhan Fungsional 1.1*

ID : KF1

TITLE : Registrasi User

DESC : Pengguna memerlukan data profil/akun yang akan diperlukan untuk masuk pada web, sehingga pengguna diperlukan mendaftar terlebih dahulu agar dapat mengakses aplikasi web. Pengguna harus memberikan nama user, kata sandi, dan alamat email.

RAT : Agar user dapat mendaftar di aplikasi web.

DEP : Tidak ada

3.2.1.2. *Kebutuhan Fungsional 1.2*

ID : KF2

TITLE : User login

DESC : Mengingat bahwa user telah terdaftar, maka user perlu untuk memasukkan data profil/akun ke aplikasi web. Data yang dimasukkan akan diproses oleh aplikasi web dan akan dicek apakah data tersebut tersedia di database atau tidak.

RAT : Agar user dapat masuk ke aplikasi web.

DEP : KF1

3.2.1.3. *Kebutuhan Fungsional 1.3*

ID : KF3

TITLE : Melihat Data Aset

DESC : User dapat melihat data aset sebelum melakukan peminjaman terhadap ruangan yang ingin digunakan pada halaman peminjaman.

RAT : Agar user dapat memilih dan melihat aset yang ingin dipinjam..

DEP : KF2

3.2.1.4. *Kebutuhan Fungsional 1.4*

ID : KF4

TITLE : Melihat Data Ruangan

DESC : User dapat melihat data ruangan sebelum melakukan peminjaman terhadap ruangan yang ingin digunakan pada halaman peminjaman.

RAT : Agar user dapat memilih dan melihat ruangan yang ingin dipinjam..

DEP : KF2

3.2.1.5. *Kebutuhan Fungsional 1.5*

ID : KF5

TITLE : Pengajuan Peminjaman Aset

DESC : User dapat melakukan pinjaman terhadap aset yang tersedia.

RAT : Agar user dapat mengajukan pinjaman aset dalam aplikasi.

DEP : KF2

3.2.1.6. *Kebutuhan Fungsional 1.6*

ID : KF6

TITLE : Pengajuan Peminjaman Ruangan

DESC : User dapat melakukan pinjaman terhadap ruangan yang tersedia.

RAT : Agar user dapat mengajukan pinjaman ruangan dalam aplikasi.

DEP : KF2

3.2.1.7. *Kebutuhan Fungsional 1.7*

ID : KF7

TITLE : Pengajuan Pengembalian Aset

DESC : Setelah user mengajukan peminjaman aset dan aset tersebut telah selesai digunakan, maka selanjutnya user harus melaporkan atau mengajukan pengembalian terhadap aset tersebut.

RAT : Agar user dapat melakukan pengembalian terhadap aset yang dipinjam.

DEP : KF2

3.2.1.8. *Kebutuhan Fungsional 1.8*

ID : KF8

TITLE : Pengajuan Pengembalian Ruangan

DESC : Setelah user mengajukan peminjaman ruangan dan ruangan tersebut telah selesai digunakan, maka selanjutnya user harus melaporkan atau mengajukan pengembalian terhadap aset tersebut.

RAT : Agar user dapat melakukan pengembalian terhadap aset yang dipinjam.

DEP : KF2

3.2.1.9. *Kebutuhan Fungsional 1.9*

ID : KF9

TITLE : Logout

DESC : Setelah user selesai menggunakan aplikasi, user dapat melakukan logout atau menghapus sesi yang sedang berjalan.

RAT : Agar user dapat keluar dari sesi aplikasi.

DEP : KF2

3.2.2. User Class 2 – Admin

3.2.2.1. *Kebutuhan Fungsional 2.1*

ID : KF10

TITLE : Admin login

DESC : Mengingat bahwa admin telah terdaftar, maka admin perlu untuk memasukkan data profil/akun ke aplikasi web. Data yang dimasukkan akan diproses oleh aplikasi web dan akan dicek apakah data tersebut tersedia di database atau tidak.

RAT : Agar user dapat masuk ke aplikasi web.

DEP : Tidak ada

3.2.2.2. *Kebutuhan Fungsional 2.2*

ID : KF11

TITLE : Mengelola Data Aset dan Ruangan

DESC : Admin dapat mengelola dan melihat data aset serta ruangan yang ada dalam database.

RAT : Agar data dapat dimanipulasi dan dikelola oleh operator atau admin.

DEP : KF10

3.2.2.3. *Kebutuhan Fungsional 2.3*

ID : KF12

TITLE : Konfirmasi Peminjaman dan Pengembalian

DESC : Setelah user mengajukan peminjaman ataupun pengembalian, langkah selanjutnya adalah admin akan mengecek data pengajuan kemudian memutuskan apakah pengajuan atau pengembalian tersebut disetujui atau tidak.

RAT : Untuk memfilter pengajuan peminjaman yang dibolehkan.

DEP : KF10

3.2.2.4. *Kebutuhan Fungsional 2.4*

ID : KF13

TITLE : Logout

DESC : Admin dapat keluar dari sesi aplikasi setelah selesai menggunakan aplikasi.

RAT : Agar admin dapat keluar dari sesi dan tidak terjadi penyalahgunaan hak akses.

DEP : KF10

IV. DESKRIPSI RINCI PERANCANGAN

4.1. Deskripsi Rinci Tabel

Berikut adalah sekumpulan tabel-tabel rinci basis data yang merupakan bagian dari **Tabel 2. Deskripsi Rinci.**

4.1.1. Tabel <users>

Identifikasi>Nama : *users*

Deskripsi isi : Cluster untuk menyimpan data pengguna untuk login.

Jenis : Tabel data induk/transaksi

Primary key : *user_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
user_id	String(10)	No	ID unik pengguna
username	String(50)	No	Username pengguna
password	String(150)	No	Password pengguna
name	String(50)	No	Nama pengguna
role	String	No	Role atau Level pengguna
email	String	No	Email kontak pengguna

4.1.2. Tabel <admin>

Identifikasi>Nama : *admin*

Deskripsi isi : Cluster untuk menyimpan data pengguna dengan hak akses admin untuk login.

Jenis : Tabel data induk/transaksi

Primary key : *nip*

Id Field	Tipe & length	Boleh NULL	Deskripsi
user_id	String(10)	No	ID unik pengguna
nip	String(20)	No	Username pengguna
phone	String(13)	No	No hp pengguna
gender	String	No	Gender pengguna

4.1.3. Tabel <sessions>

Identifikasi>Nama : *sessions*

Deskripsi isi : Cluster untuk menyimpan sesi login pengguna

Jenis : Table data induk/transaksi

Primary key : *session_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
session_id	String(10)	No	ID unik sesi pengguna
user_id	String(10)	No	ID unik pengguna
token	String(50)	No	Kode unik utk sesi
last_accessed	Timestamp	No	Waktu akses terakhir pengguna
expired	Timestamp	No	Waktu berakhir sesi

4.1.4. Tabel <rooms>

Identifikasi>Nama : *rooms*

Deskripsi isi : Cluster untuk menyimpan data ruangan

Jenis : Tabel data induk/transaksi

Primary key : *room_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
room_id	String(10)	No	ID unik ruangan
room_name	String(20)	No	Nama ruangan
capacity	Integer	No	Kapasitas orang dalam ruangan

4.1.5. Tabel <assets>

Identifikasi>Nama : *assets*

Deskripsi isi : Cluster untuk menyimpan data aset

Jenis : Tabel data induk/transaksi

Primary key : *asset_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
asset_id	String(10)	No	ID unik aset
room_id	String(10)	No	ID unik ruangan
asset_name	String(20)	No	Nama aset
quantity	Integer	No	Jumlah aset yang tersedia

4.1.6. Tabel <students>

Identifikasi>Nama : *students*

Deskripsi isi : Cluster untuk menyimpan data mahasiswa

Jenis : Tabel data induk/transaksi

Primary key : *nim*

Id Field	Tipe & length	Boleh NULL	Deskripsi
nim	String(20)	No	Nomor Induk Mahasiswa
user_id	String(10)	No	ID unik pengguna
phone	String(13)	No	No hp pengguna
major_id	Integer	No	Jurusan pengguna
gender	String	No	Gender pengguna

4.1.7. Tabel <faculty>

Identifikasi>Nama : *faculty*

Deskripsi isi : cluster untuk menyimpan data jenis fakultas

Jenis : table data induk/transaksi

Primary key : *faculty_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
faculty_id	String	No	ID unik fakultas
faculty_name	String(10)	No	Nama fakultas

4.1.8. Tabel <major>

Identifikasi>Nama : *major*

Deskripsi isi : cluster untuk menyimpan data jenis jurusan

Jenis : table data induk/transaksi

Primary key : *major_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
major_id	String	No	ID unik jurusan
faculty_id	String(10)	No	ID unik ruangan
major_name	String(50)	No	Nama jurusan

4.1.9. Tabel <requests>

Identifikasi>Nama : *request*

Deskripsi isi : Cluster untuk menyimpan data permohonan/pengajuan

Jenis : Tabel data induk/transaksi

Primary key : *request_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
request_id	String(10)	No	ID unik permohonan atau pengajuan
user_id	String(10)	No	ID unik pengguna
req_type_id	Integer	No	ID unik jenis permohonan atau pengajuan
status_id	Integer	No	ID unik status permohonan atau pengajuan

created_at	Timestamp	No	Waktu dibuatnya permohonan atau pengajuan
------------	-----------	----	---

4.1.10. Tabel <request_type>

Identifikasi>Nama : *request_type*

Deskripsi isi : Cluster untuk menyimpan data jenis permohonan atau pengajuan

Jenis : Tabel data induk/transaksi

Primary key : *request_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
request_type_id	Integer	No	ID unik jenis permohonan atau pengajuan
request_name	String(20)	No	Nama jenis permohonan atau pengajuan

4.1.11. Tabel <request_status>

Identifikasi>Nama : *request_status*

Deskripsi isi : Cluster untuk menyimpan data status permohonan atau pengajuan

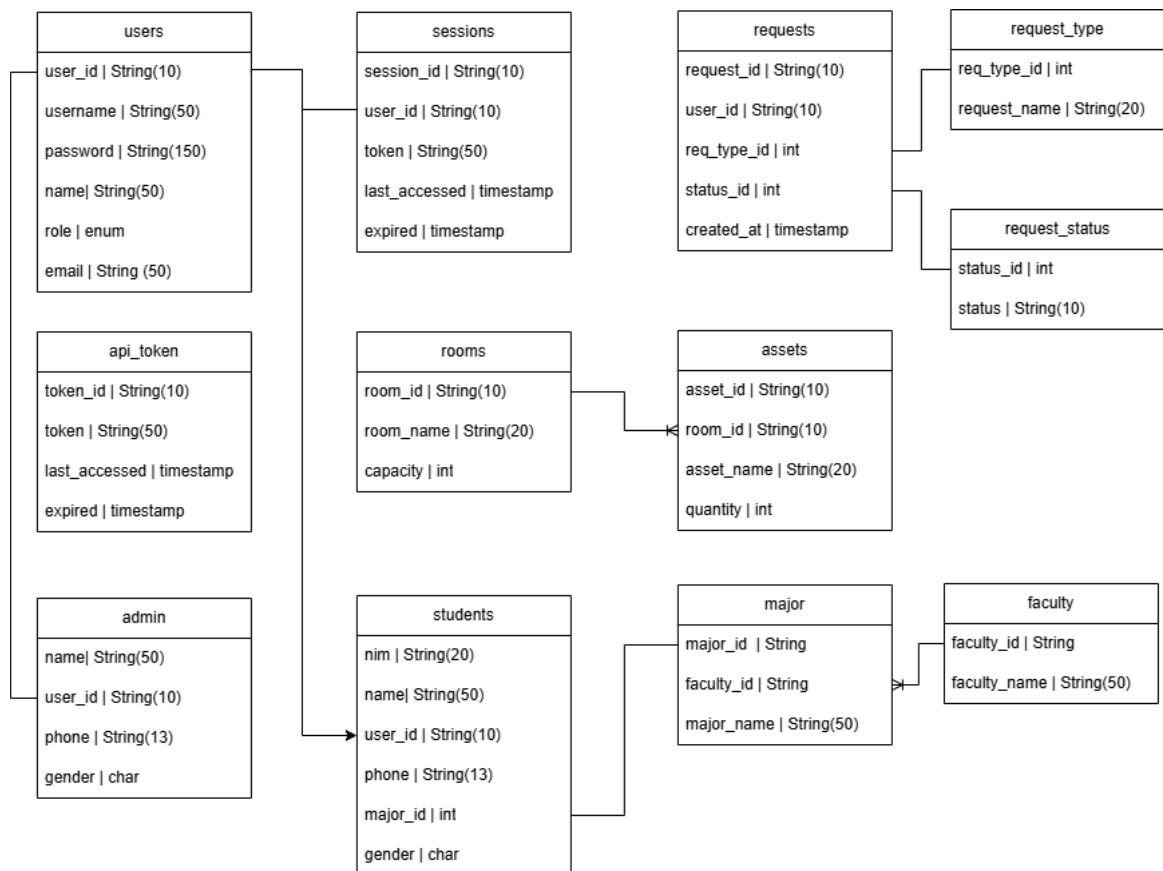
Jenis : Tabel data induk/transaksi

Primary key : *status_id*

Id Field	Tipe & length	Boleh NULL	Deskripsi
status_id	Integer	No	ID unik status permohonan atau pengajuan
status	String(10)	No	Status/keterangan

4.2. Entity Relationship Diagram

Perancangan relasi antar entitas di database ditunjukkan pada Gambar 1. Database yang akan digunakan menggunakan konsep NoSQL dimana tidak ada relasi antar tabel melainkan struktur data yang disimpan adalah dalam bentuk dokumen yang tidak memiliki aturan terhadap struktur isi atau kolomnya.

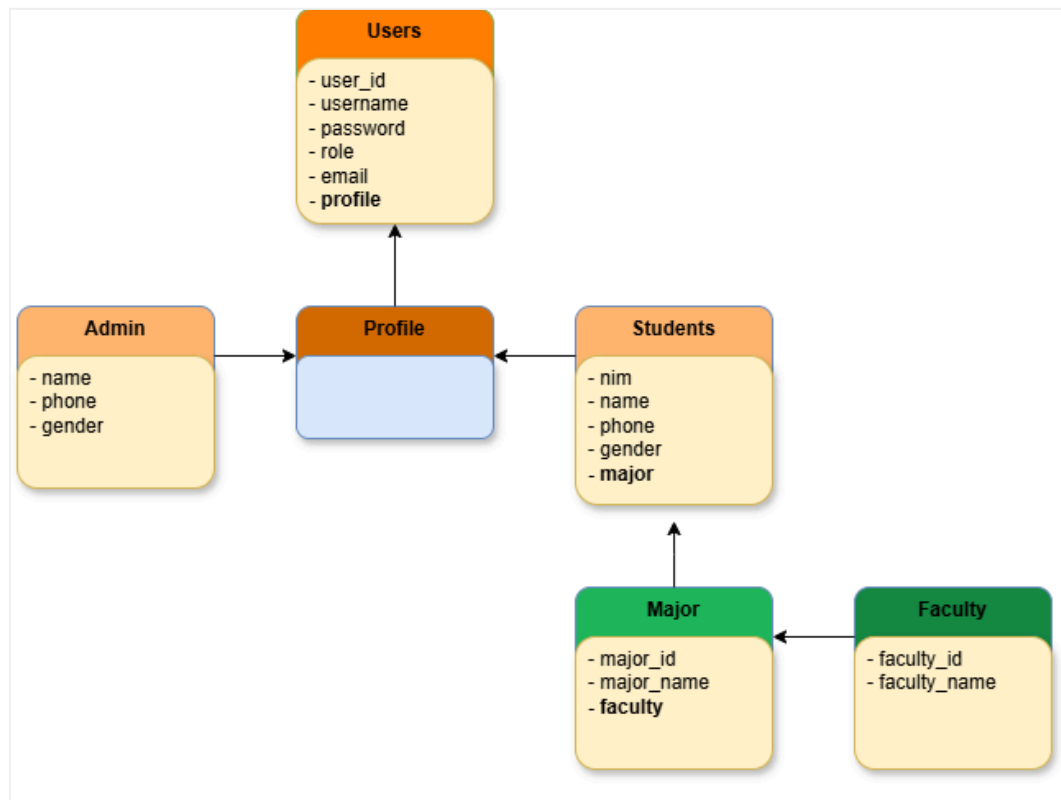


Gambar 1. Entity Relationship Diagram

4.3. Denormalisasi

Pada desain sistem yang dikembangkan struktur database yang dipakai adalah NoSQL dan DBMS (Database Management System) adalah MongoDB dengan format datanya adalah koleksi berbentuk JSON. Denormalisasi adalah konsep penting pada penerapan database berbasis NoSQL yang berguna untuk meningkatkan efisiensi *read* dan mengurangi kompleksitas operasi pengambilan data. Berikut beberapa document JSON yang di denormalisasi.

4.3.1 Denormalisasi - Data 1



Gambar 2. Denormalisasi - Data 1

Gambar berikut merupakan denormalisasi untuk koleksi *users* dimana koleksi lainnya seperti *sessions*, *api_token* akan digabungkan menjadi objek bagian koleksi *users* dan untuk *admin* dan *students* dikelompokkan ke dalam objek atau entitas baru yaitu *profile* secara terpisah tergantung role pengguna, dan untuk *major* dan *faculty* itu digabungkan menjadi objek pada bagian *profile* khusus untuk role mahasiswa.

Contoh data untuk koleksi *Users* dalam bentuk JSON:

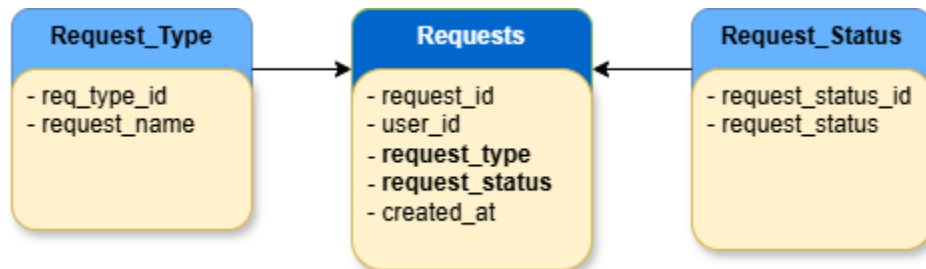
Bentuk data Document Users dengan profile untuk role "admin"

```
{
  "user_id": "UA001",
  "username": "admin_01",
  "password": "8c6976e5",
  "role": "admin",
  "email": "admin@example.com",
  "profile": {
    "name": "User Admin",
    "phone": "081200000000",
    "gender": "Laki-laki"
  }
}
```

Bentuk data Document Users dengan profile untuk role "student"

```
{
  "user_id": "US001",
  "username": "student_01",
  "password": "04f8996d",
  "role": "student",
  "email": "student@example.com",
  "profile": {
    "nim": "2201020000",
    "name": "User Student",
    "phone": "081200000000",
    "gender": "Laki-laki",
    "major": {
      "major_id": "M001",
      "major_name": "Teknik Informatika",
      "faculty": {
        "faculty_id": "F001",
        "faculty_name": "Engineering"
      }
    }
  }
}
```

4.3.2 Denormalisasi - Data 2



Gambar 3. Denormalisasi - Data 2

Gambar berikut merupakan bentuk denormalisasi yang kedua untuk koleksi *requests* dimana setiap *requests* memiliki objek atau entitas *request_type*, dan *request_status* yang diambil dan di denormalisasi dari awalnya koleksi tersendiri menjadi objek dalam koleksi *requests*.

Contoh data untuk koleksi *Requests* dalam bentuk JSON:

```
Bentuk data Document Requests
{
  "request_id": "R001",
  "user_id": "US001",
  "request_type": {
    "req_type_id": 0,
    "request_name": "Ruangan"
  },
  "request_status": {
    "status_id": 2,
    "status": "Accept"
  },
  "created_at": "2024-06-12T10:30:00Z"
}
```


4.4. Relasi Antar Tabel

Database yang digunakan menerapkan konsep cluster atau dokumen sehingga tidak ada relasi atau hubungan khusus antar tabel, namun berikut adalah gambaran terhadap hubungan antar tabel dalam konsep Relational Database Management System (RDBMS).

4.4.1. User to Student (One to One)

Setiap mahasiswa hanya dapat memiliki satu buah akun.

4.4.2. User to Admin (One to One)

Admin hanya dapat memiliki satu buah akun.

4.4.3. Session to User (One to One)

Setiap user hanya mempunyai satu sesi login.

4.4.4. Room to Asset (One to Many)

Setiap ruangan memiliki banyak aset didalamnya.

4.4.5. Req.Type to Request (One to One)

Sebuah pengajuan hanya dapat memiliki satu jenis pengajuan.

4.4.6. Req.Status to Request (One to One)

Sebuah pengajuan hanya dapat memiliki satu buah status yang akan diperbarui setiap prosesnya.

4.4.7. Student to Major (One to One)

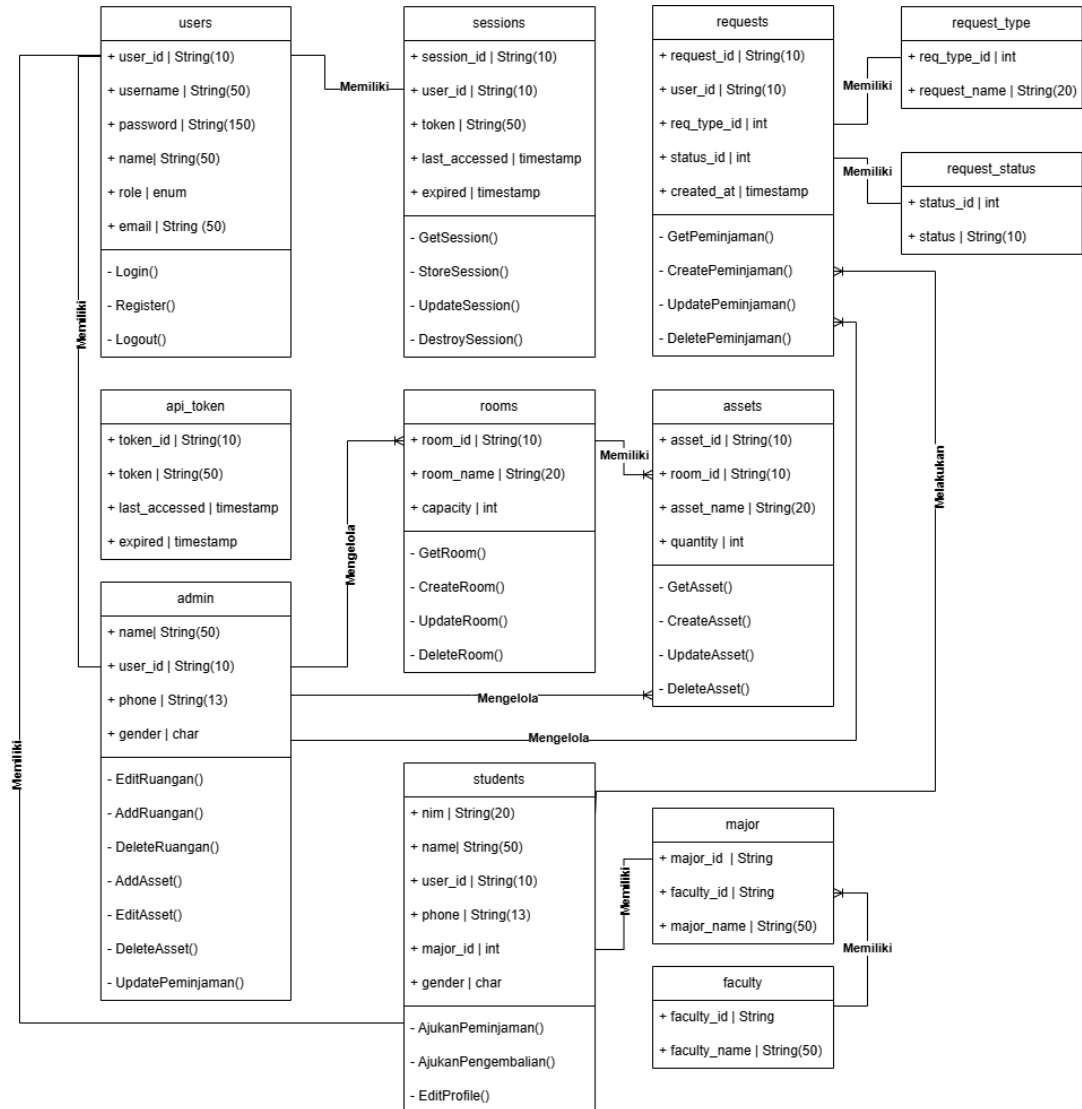
Setiap mahasiswa hanya memiliki satu fakultas.

4.4.8. Faculty to Major (One to Many)

Setiap fakultas memiliki banyak jurusan didalamnya.

4.5. Class Diagram

4.5.1. Class Diagram

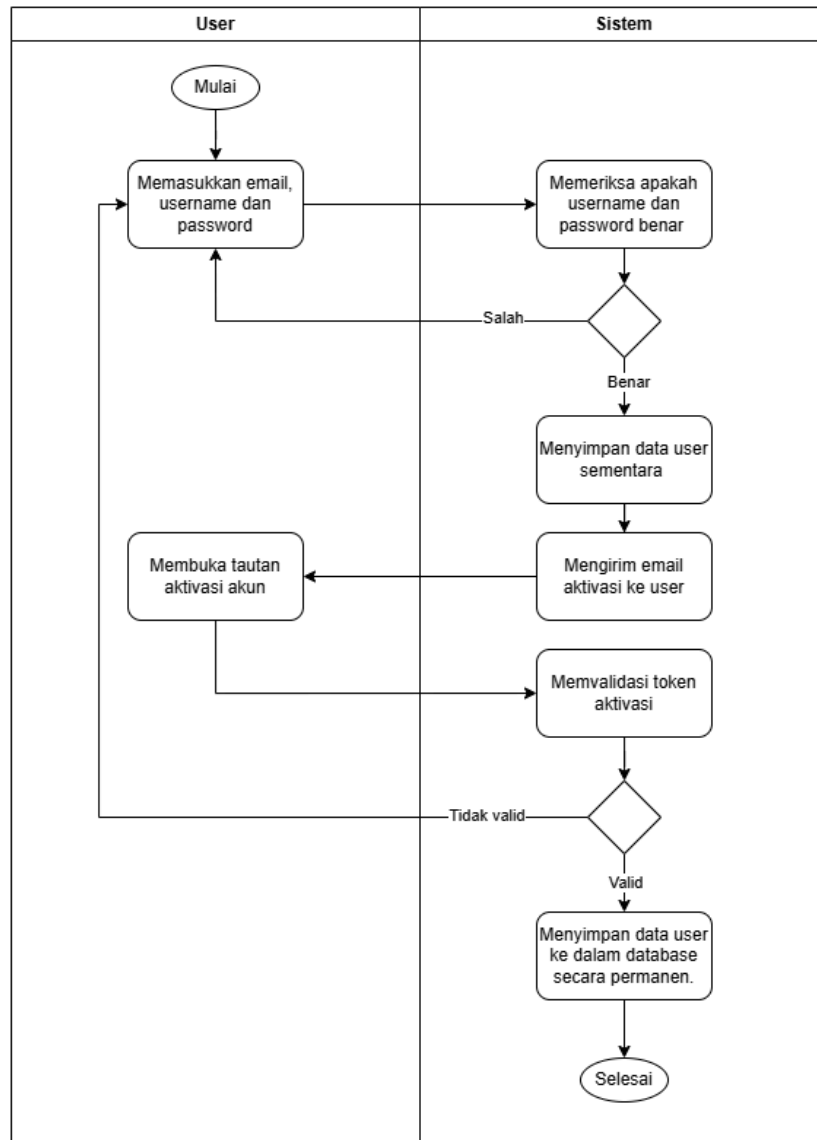


Gambar 4. Class Diagram

4.6. Activity Diagram

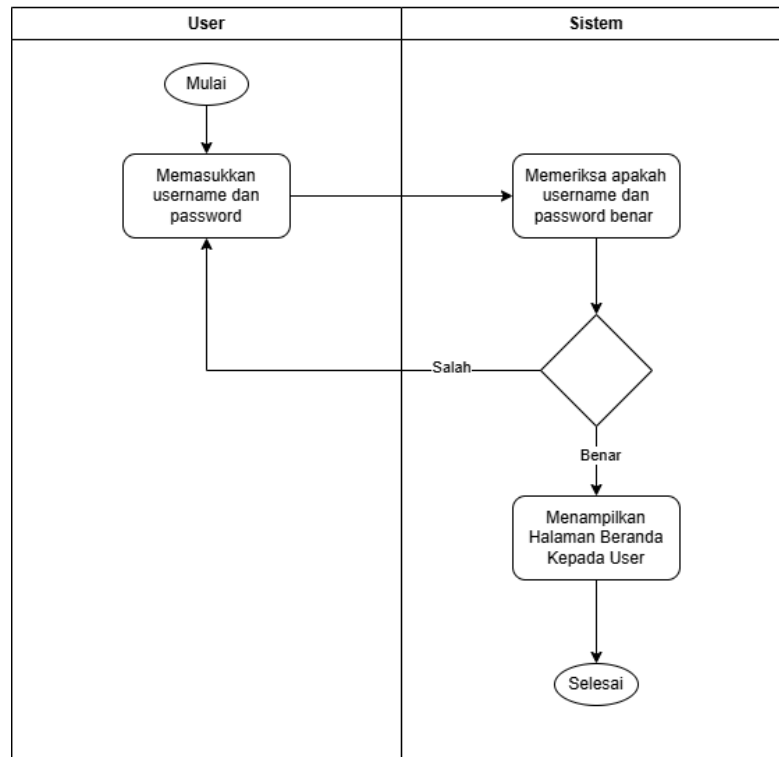
Berikut adalah sekumpulan activity diagram yang menentukan alur kerja suatu fungsi dalam sistem aplikasi yang dikembangkan.

4.6.1. Activity Diagram Register



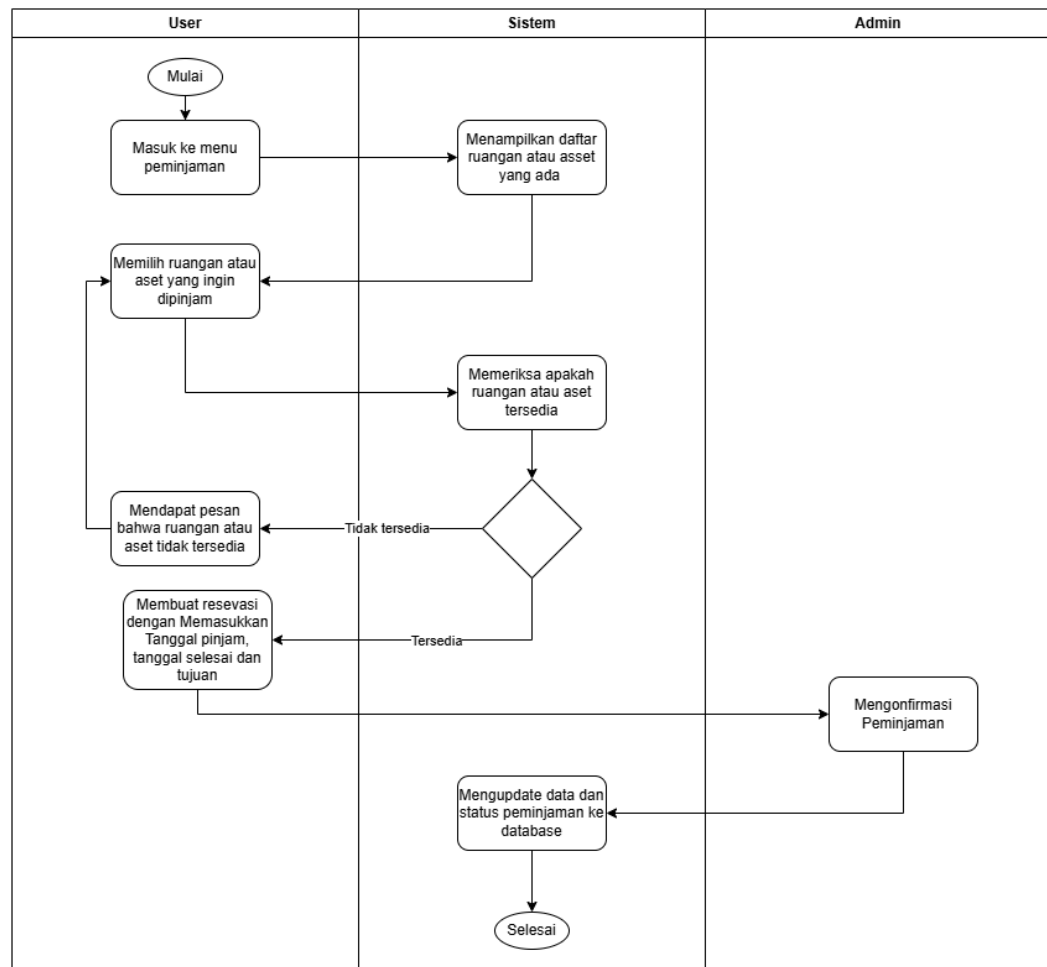
Gambar 5. Activity Diagram Register

4.6.2. Activity Diagram Login



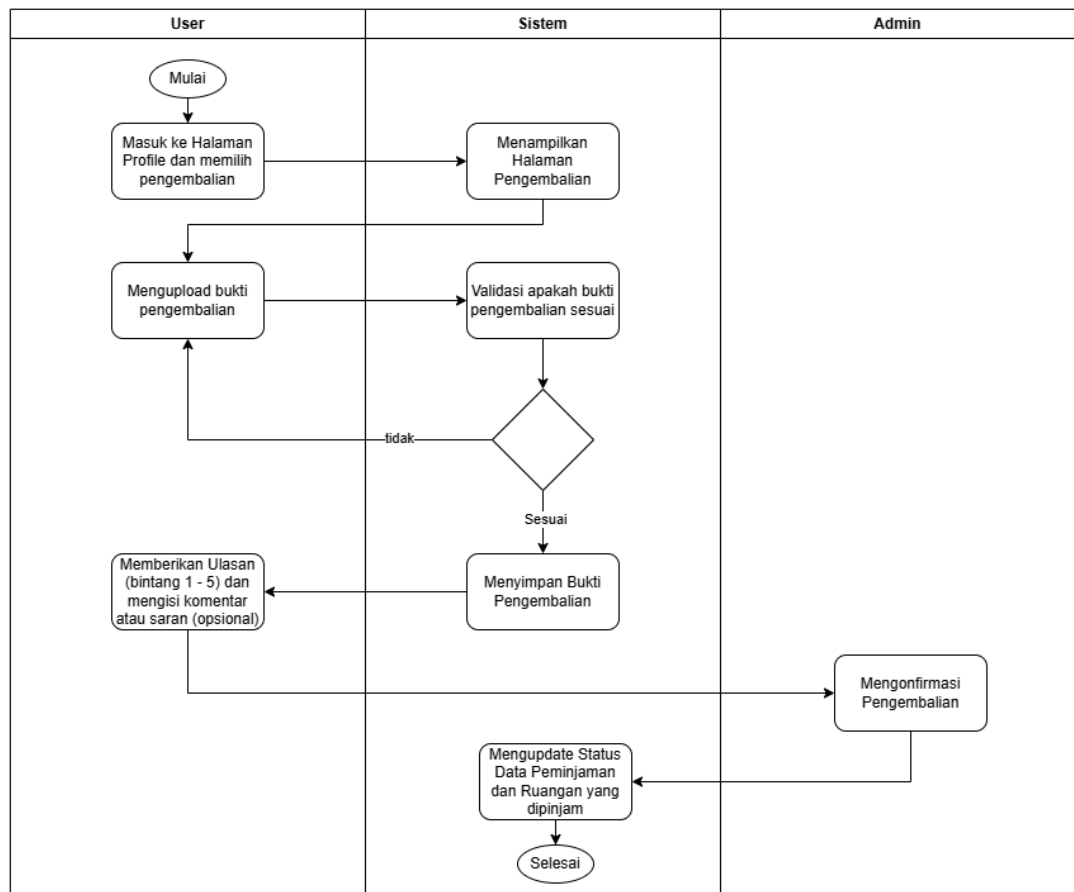
Gambar 6. Activity Diagram Login

4.6.3. Activity Diagram Peminjaman



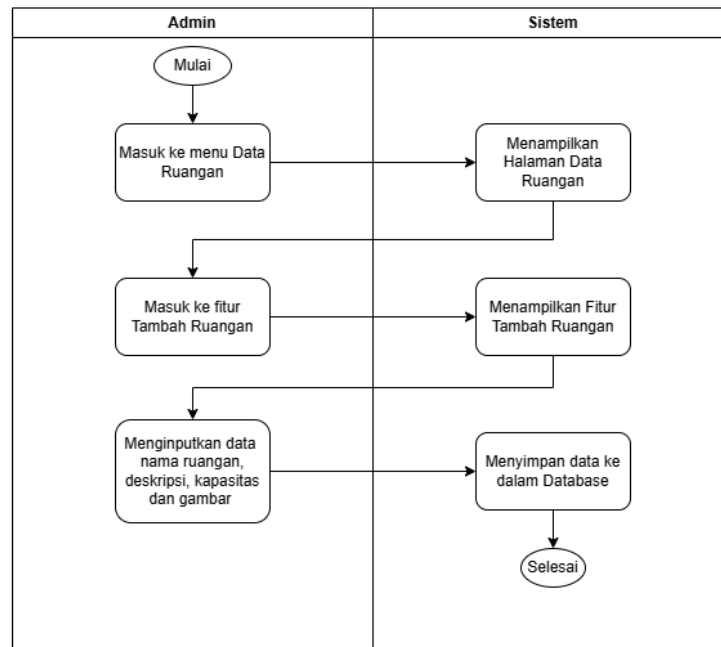
Gambar 7. Activity Diagram Peminjaman

4.6.4. Activity Diagram Pengembalian



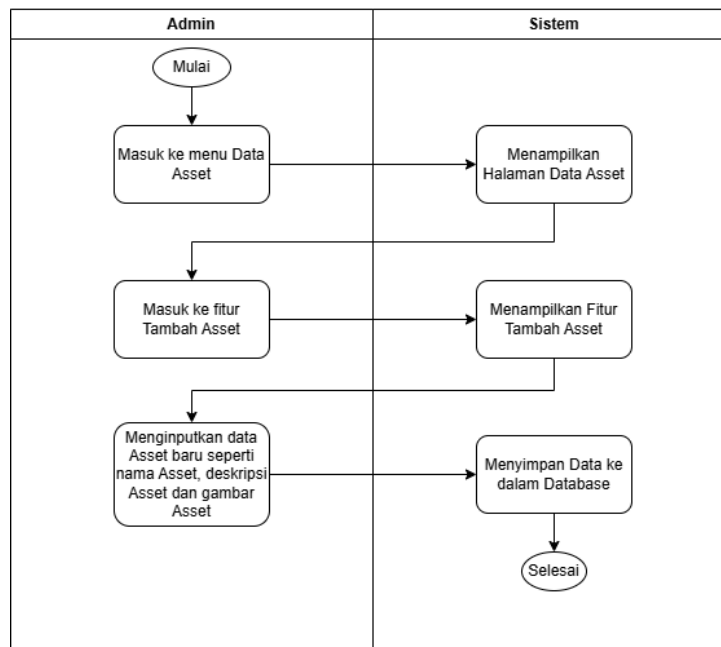
Gambar 8. Activity Diagram Pengembalian

4.6.5. Activity Diagram Tambah Ruangan



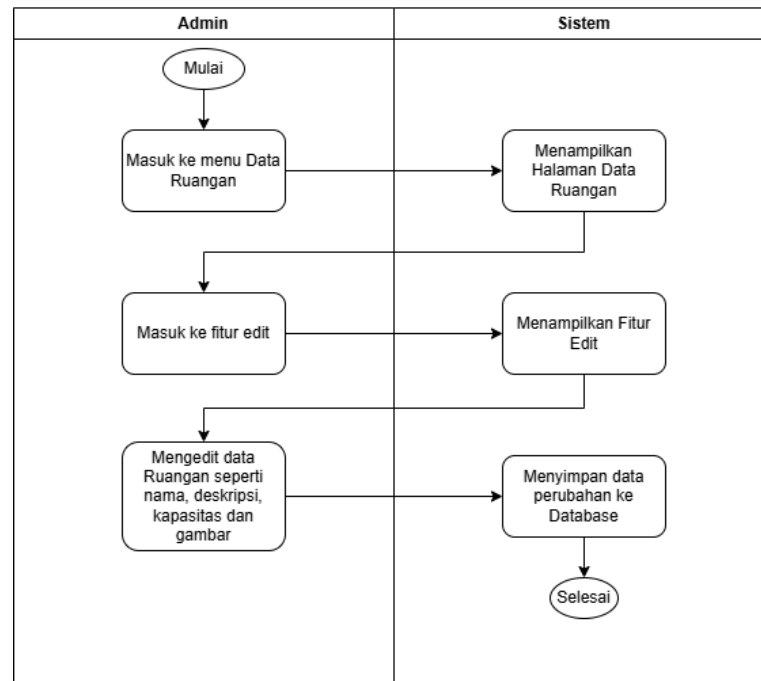
Gambar 9. Activity Diagram Tambah Ruangan

4.6.6. Activity Diagram Tambah Asset



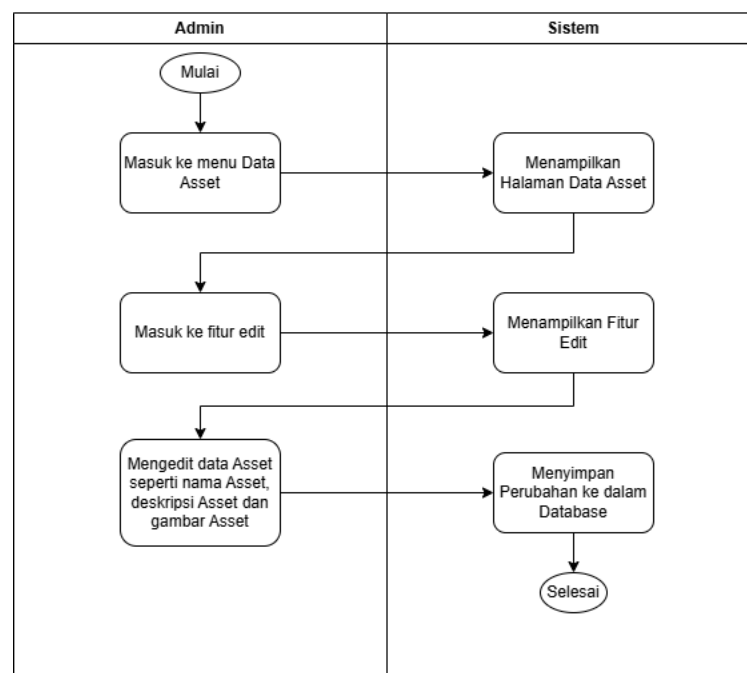
Gambar 10. Activity Diagram Tambah Asset

4.6.7. Activity Diagram Edit Ruangan



Gambar 11. Activity Diagram Edit Ruangan

4.6.8. Activity Diagram Edit Asset

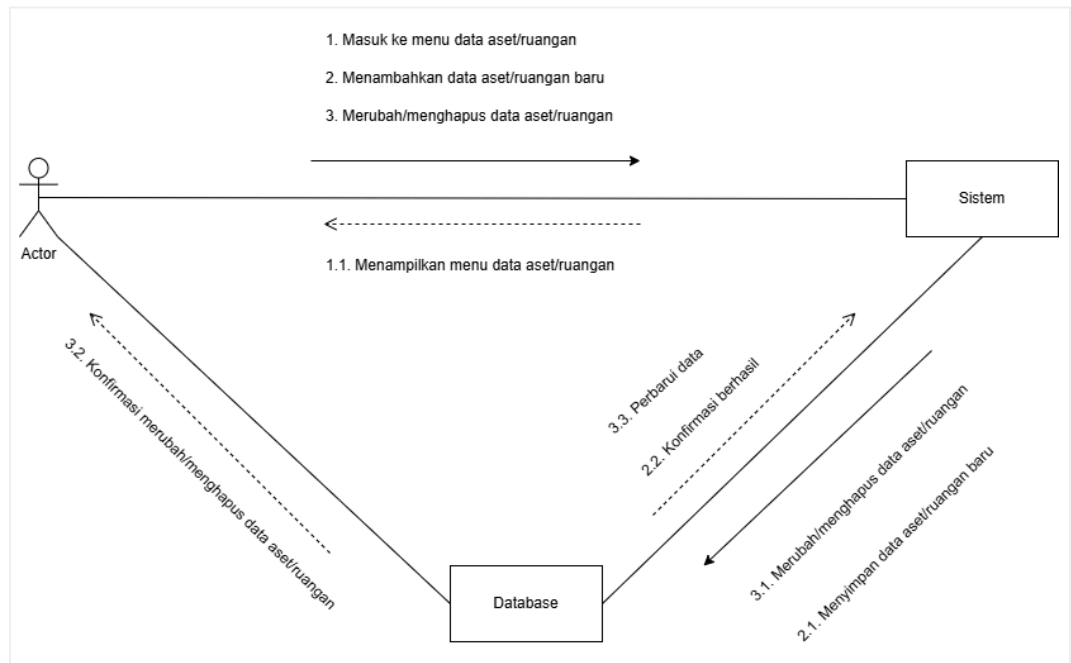


Gambar 12. Activity Diagram Edit Asset

4.7. Collaboration Diagram

4.7.1. Pengelolaan Ruangan dan Aset

Gambar berikut merupakan collaboration diagram untuk proses Pengelolaan terhadap Ruangan dan Aset UMRAH.

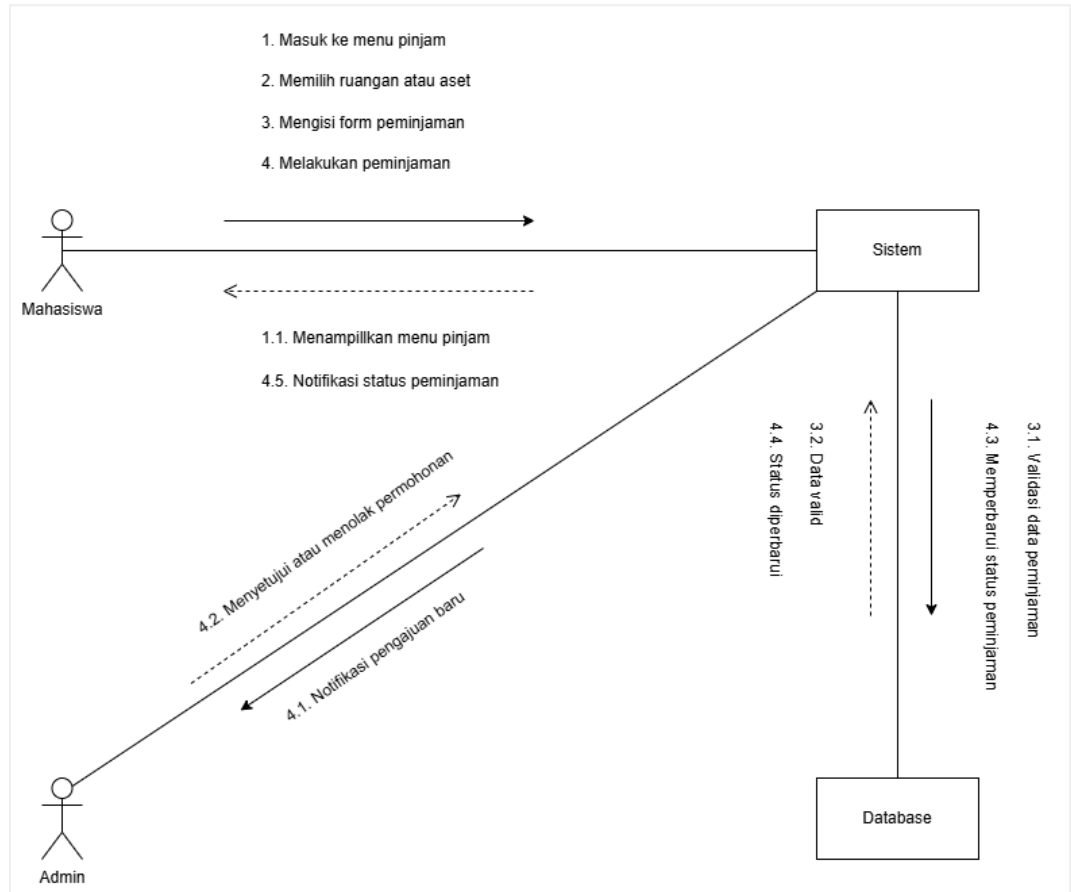


Gambar 13. Collaboration Diagram Pengelolaan Ruangan dan Aset

Diagram ini menggambarkan proses interaksi antara aktor (Admin), sistem, dan database dalam fitur pengelolaan data ruangan atau aset. Proses ini meliputi penambahan, pengubahan, dan penghapusan data ruangan atau aset.

4.7.2. Peminjaman Ruangan dan Aset

Gambar berikut merupakan collaboration diagram untuk proses Peminjaman terhadap Ruangan dan Aset UMRAH.



Gambar 14. Collaboration Diagram Peminjaman Ruangan dan Aset

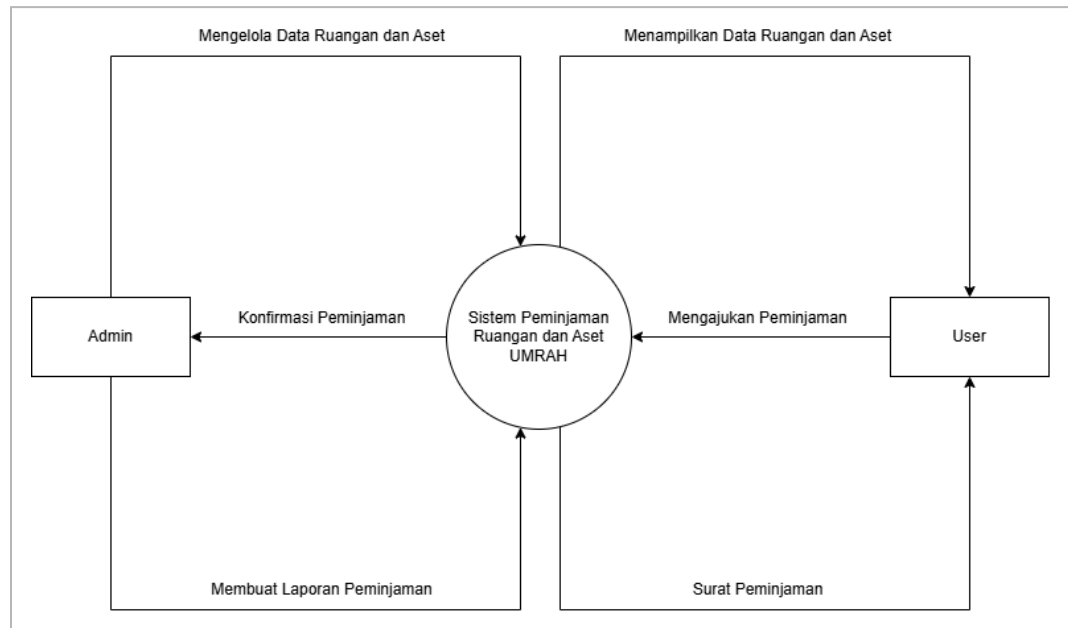
Pada diagram ini ditunjukkan bagian peranan dari masing-masing aktor dalam keterkaitannya satu sama lain terhadap sistem aplikasi dalam melakukan transaksi peminjaman terhadap ruangan dan aset pada UMRAH.

Proses ini dimulai dari pengguna atau mahasiswa masuk ke menu program untuk melakukan peminjaman dimana admin akan menerima notifikasi terhadap transaksi peminjaman yang kemudian admin tersebut dapat menyetujui atau menolak permohonan peminjaman tersebut yang dimana hasil permohonannya akan diberitahukan kepada user melalui notifikasi.

4.8. Deskripsi Fungsional

4.8.1. Context Diagram

Gambar berikut merupakan context diagram pada Sistem Peminjaman Ruangan dan Aset UMRAH.



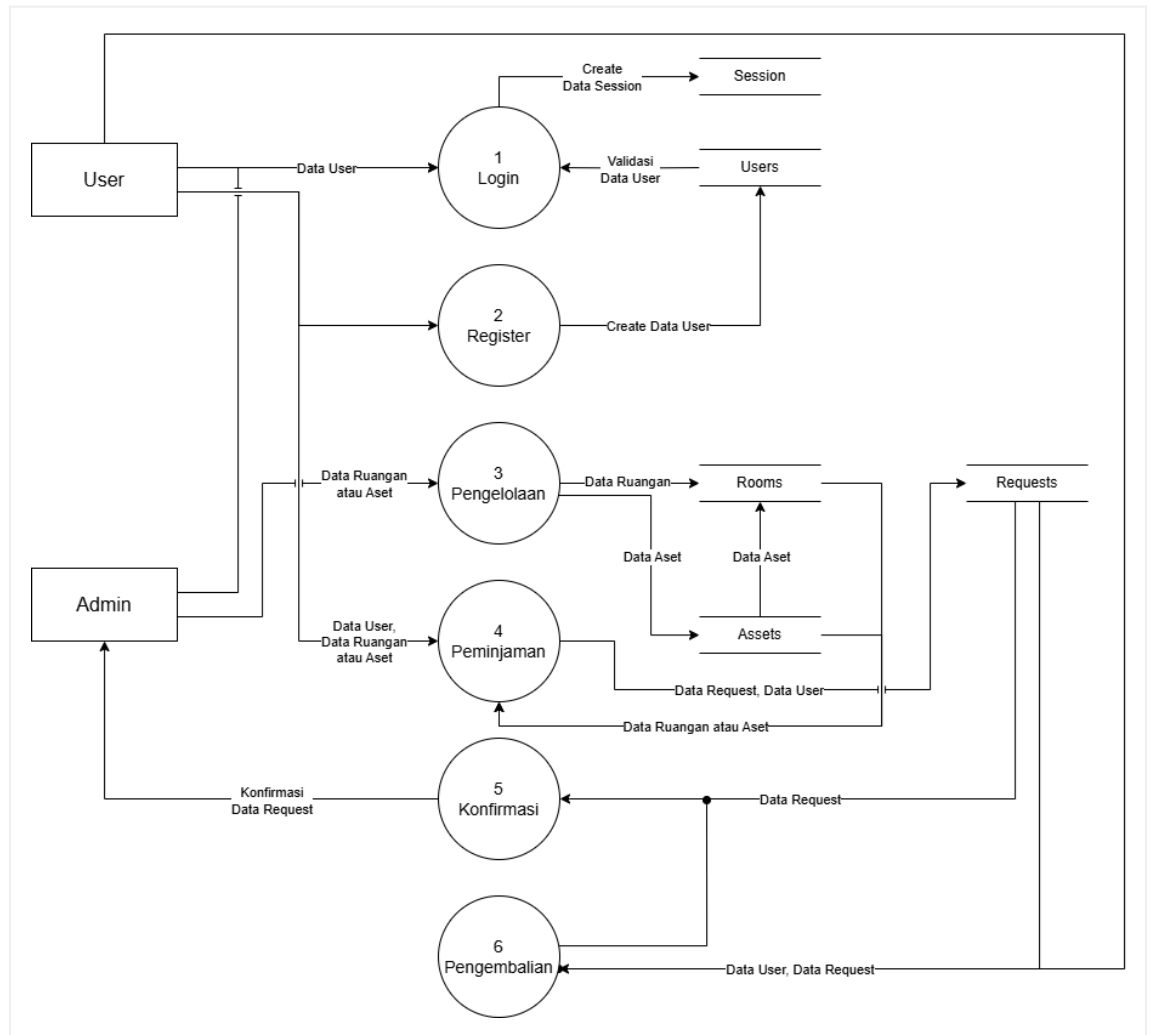
Gambar 15. Context Diagram

Pada dasarnya sistem ini bertujuan untuk sarana melakukan transaksi peminjaman terhadap ruangan dan aset pada UMRAH, dalam sistem tersebut User dapat melihat detail data ruangan maupun aset yang tersedia atau sedang dipinjam, data-data tersebut dikelola oleh Admin dan ketika terdapat ruangan atau aset baru Admin akan bertanggung jawab untuk mengupdate data tersebut.

Ketika User ingin meminjam suatu ruangan ataupun aset, User tersebut dapat memilih data ruangan atau aset yang perlu dipinjam dan peminjaman tersebut akan diajukan kepada Admin untuk diproses dan dikonfirmasi, ketika data telah dikonfirmasi Admin akan membuat laporan dengan data User dan kemudian akan memberikan surat peminjaman kepada User agar dapat meminjam ruangan atau aset yang diperlukan tersebut.

4.8.2. DFD Level 1

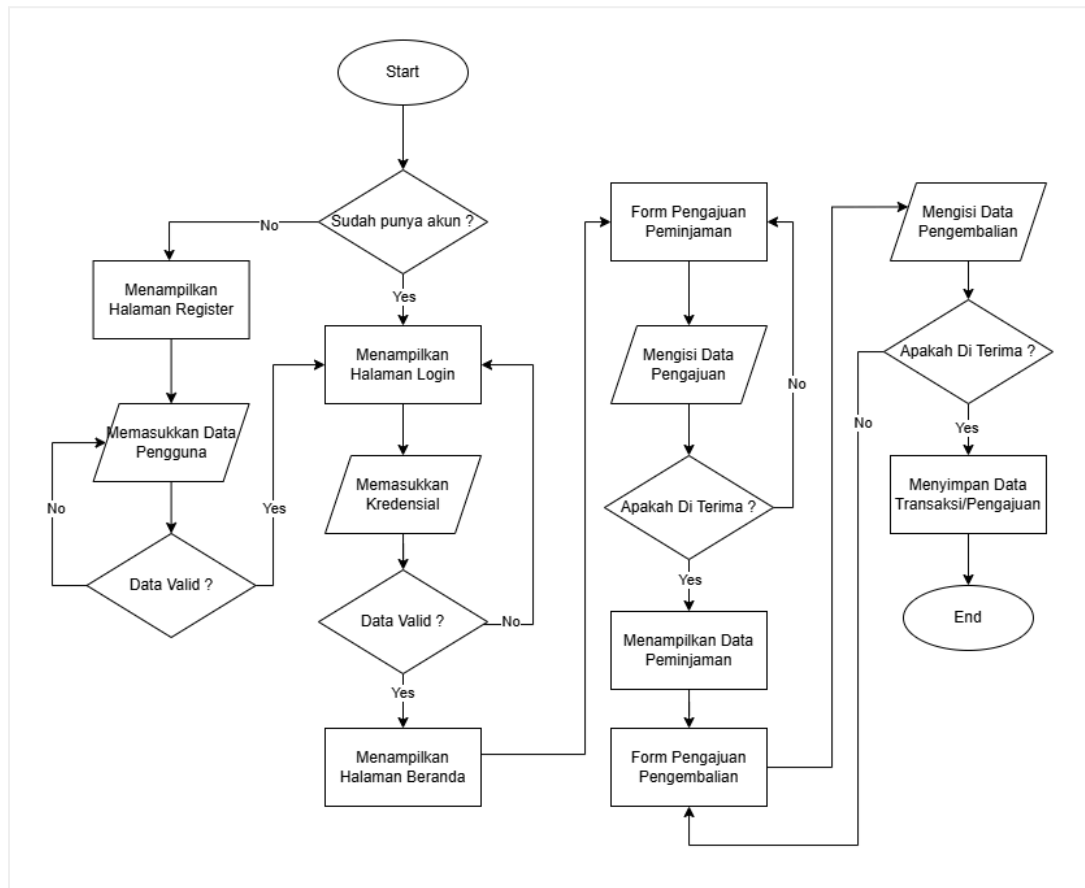
Gambar berikut adalah DFD Level 1 yang merupakan hasil pemecahan terhadap Context Diagram pada Sistem Peminjaman Ruangan dan Aset UMRAH.



Gambar 16. DFD lv 1

Pada gambar DFD Level 1 ini dilakukan pemecahan terhadap Context Diagram dengan tujuan untuk memberikan detail terhadap data yang digunakan pada setiap hubungan atau proses yang ada pada Sistem Peminjaman Ruangan dan Aset UMRAH.

4.8.3. Flowchart



Gambar 17. Flowchart

Proses dimulai dari pengguna mengakses sistem. Pengguna dapat membuat dan mengaktivasi akun pada halaman *registrasi* jika belum memiliki akun. Jika sudah memiliki akun, pengguna memasukkan kredensial di halaman *login*, dan jika valid, diarahkan ke halaman *beranda*.

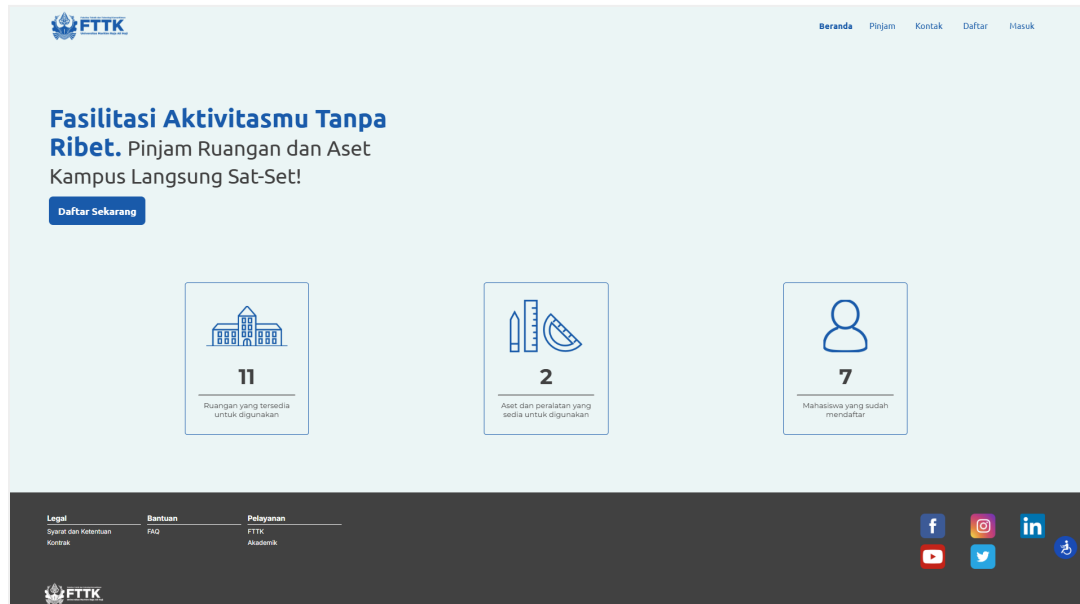
Pengguna dapat melakukan peminjaman dapat mengakses formulir pengajuan peminjaman pada halaman *beranda*, mengisi data pengajuan, dan menunggu keputusan apakah pengajuan diterima. Jika diterima, data peminjaman ditampilkan pada halaman *profile* pengguna.

Dan ketika pengguna telah selesai menggunakan ruangan dan aset peminjaman, pengguna dapat mengakses formulir pengajuan pengembalian untuk mengisi data pengembalian yang tersedia pada halaman *profile* pengguna. Sistem memvalidasi data pengembalian, dan jika diterima, data transaksi disimpan.

4.9. Antarmuka Pengguna

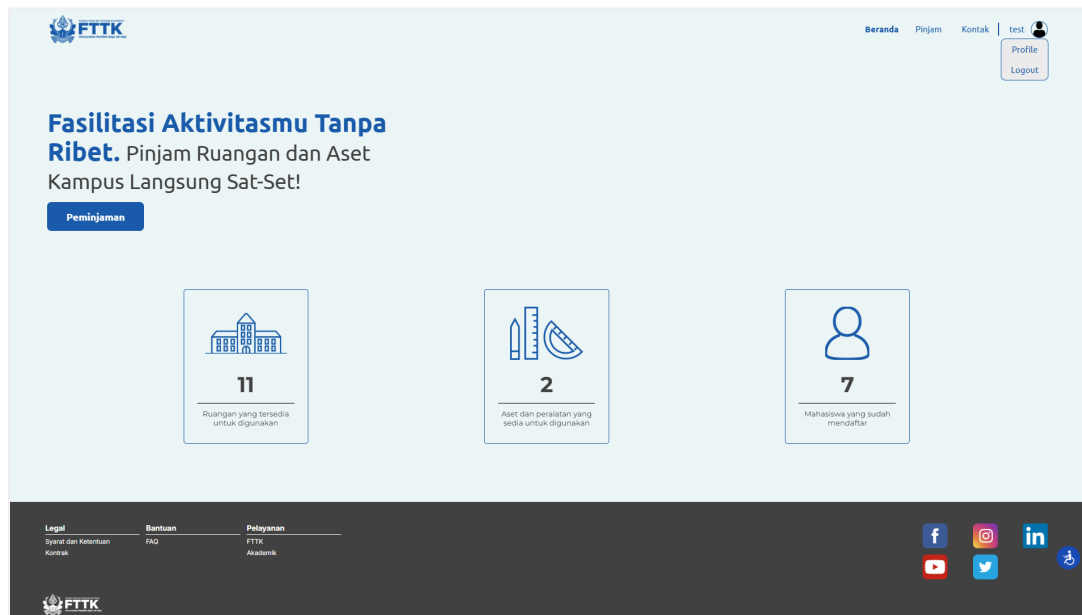
Aplikasi "Peminjaman Ruangan dan Asset UMRAH" memiliki antarmuka berbasis web, yang memungkinkan akses melalui perangkat apa pun pada jaringan internet. Berikut adalah penjelasan mengenai antarmuka yang menggunakan tampilan desktop:

4.9.1. Tampilan beranda 1.1



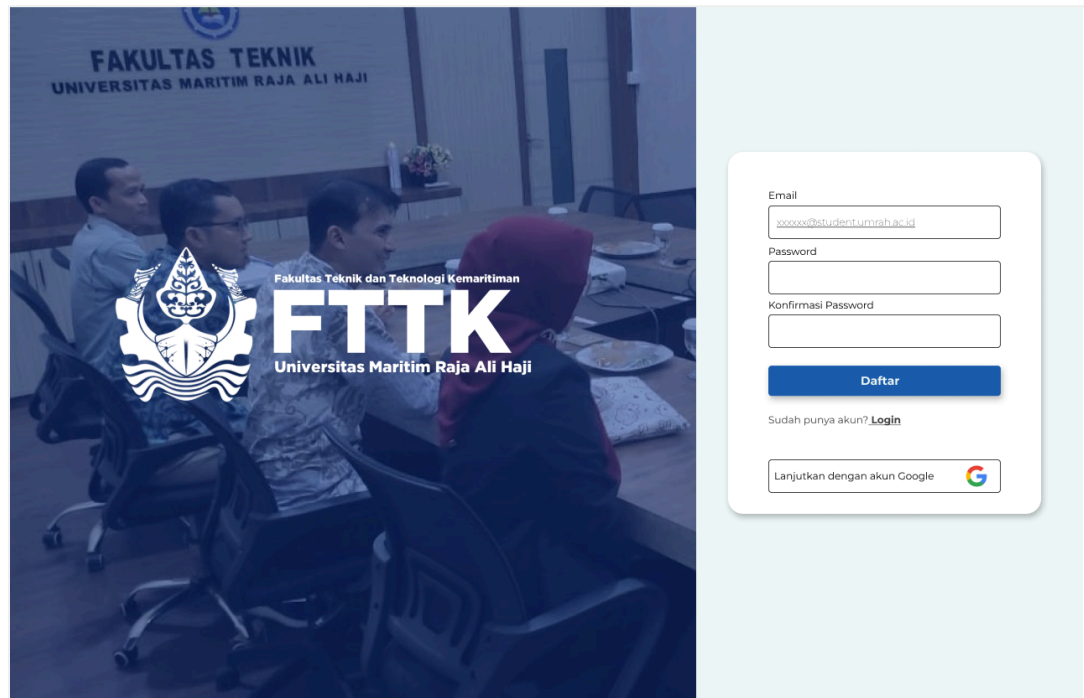
Gambar 18. Tampilan Beranda

4.9.2. Tampilan beranda 1.2



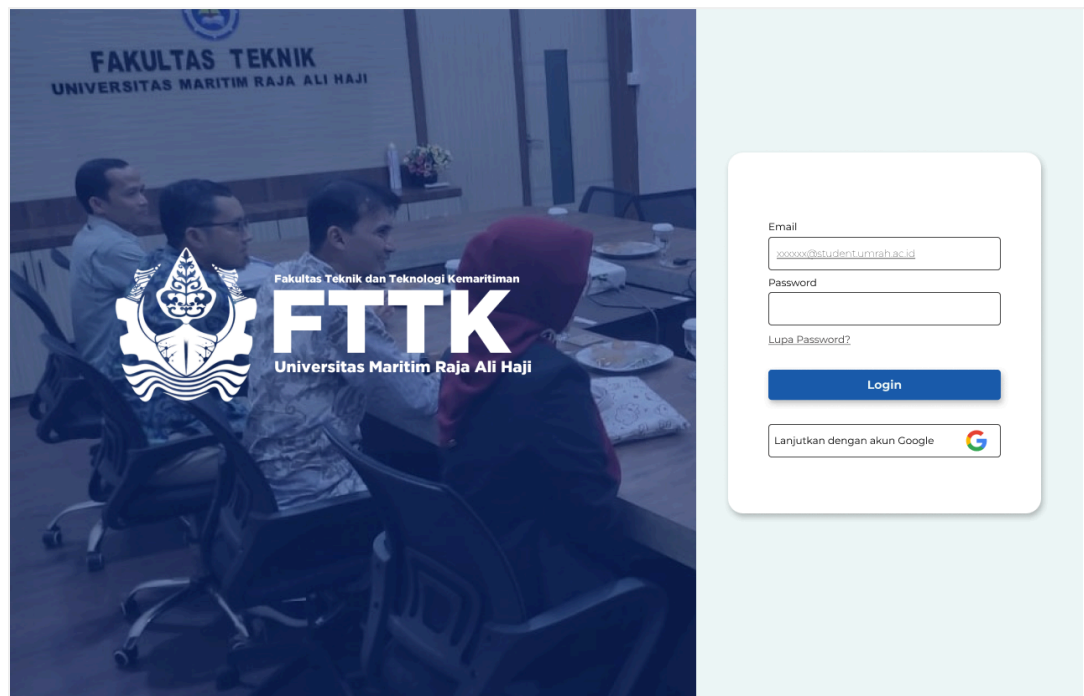
Gambar 19. Tampilan Beranda 2

4.9.3. Tampilan register



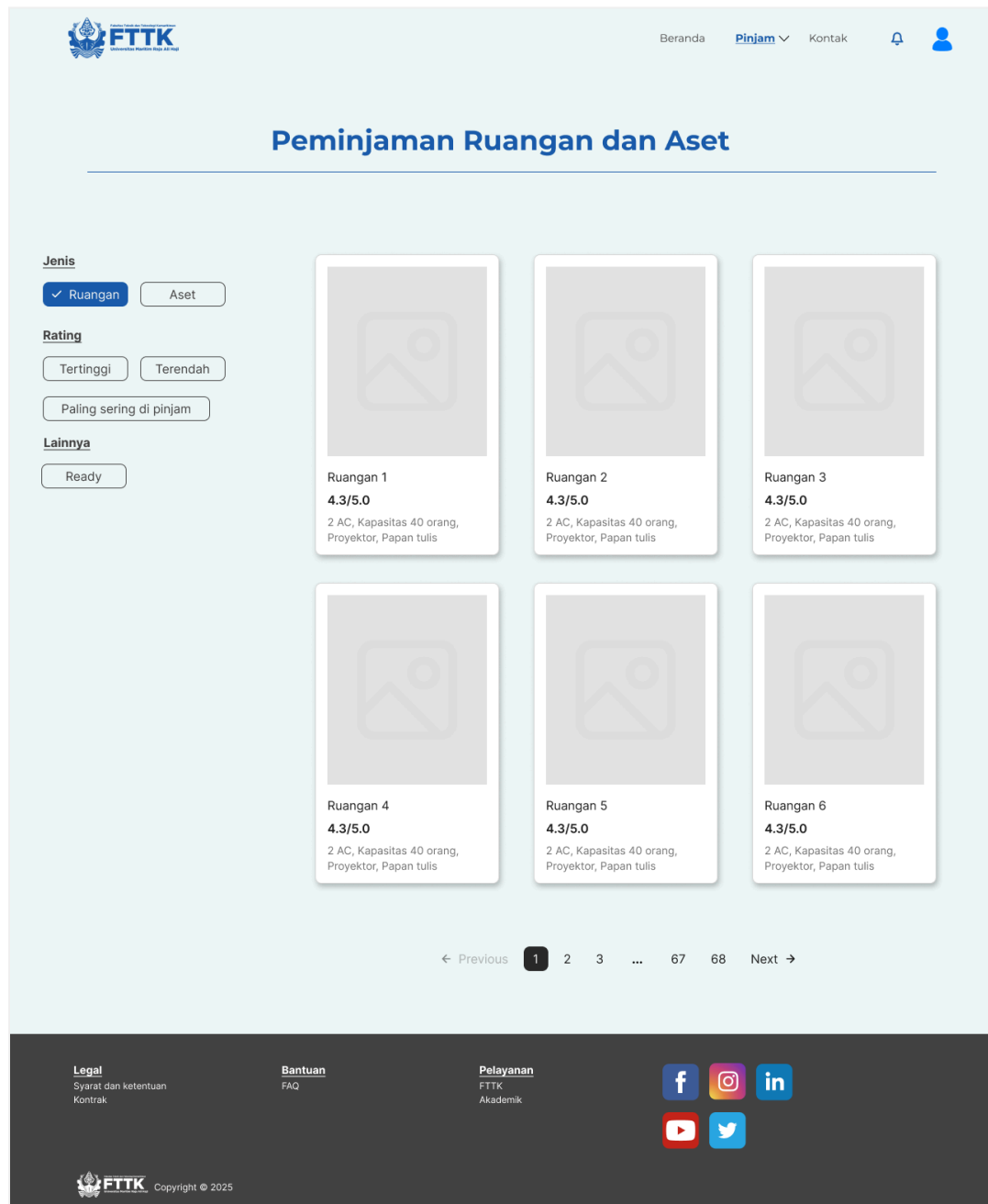
Gambar 20. Tampilan Register

4.9.4. Tampilan login



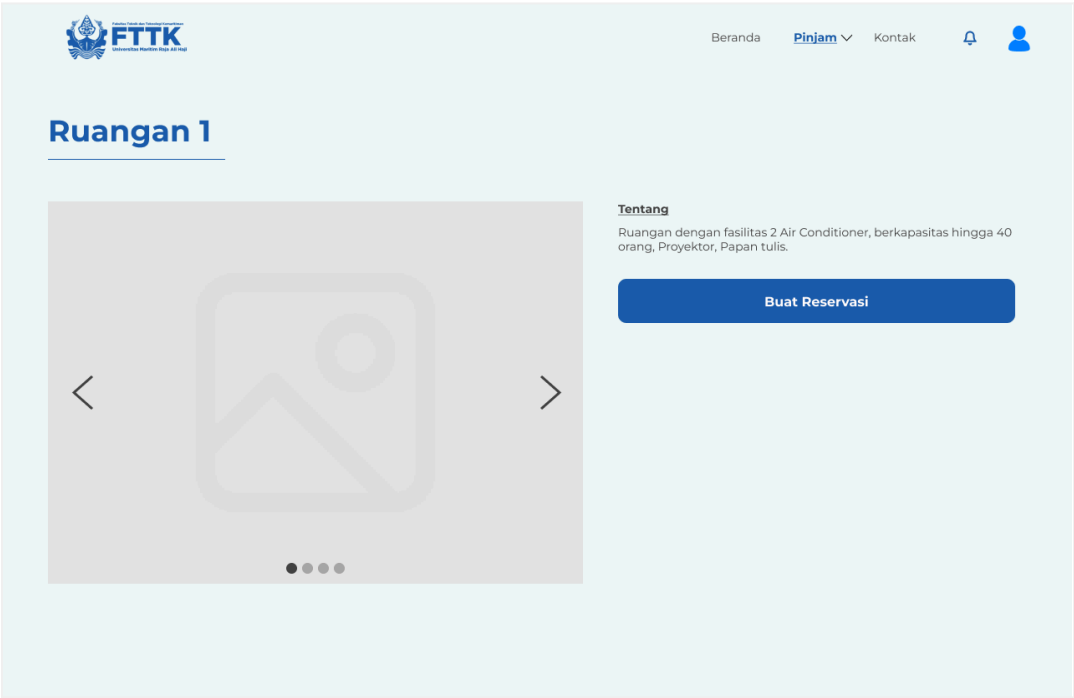
Gambar 21. Tampilan Login

4.9.5. Tampilan peminjaman ruangan dan aset untuk user



Gambar 22. Tampilan Peminjaman Ruangan dan Asset

4.9.6. Tampilan Ruangan 1 untuk User



Gambar 23. Tampilan Ruangan 1

4.9.7. Tampilan Menu Profile User



Gambar 24. Tampilan Halaman Profile User

4.9.8. Tampilan Menu Detail Peminjaman

Detail Peminjaman

Request ID : P000000001

ID Ruangan : R00001

Status : Pengembalian

Note : -

Tanggal Peminjaman : 2024-12-2 7:0:0 - 2024-12-3 7:0:0

Tujuan : Nonton onepiece

Close

Gambar 25. Tampilan Menu Detail Peminjaman

4.9.9. Tampilan Menu Update Profile

Foto Profil

Nama : Muhammad Chandra Ramadhan

NIM : 22010201030

No. Whatsapp : 087719756442

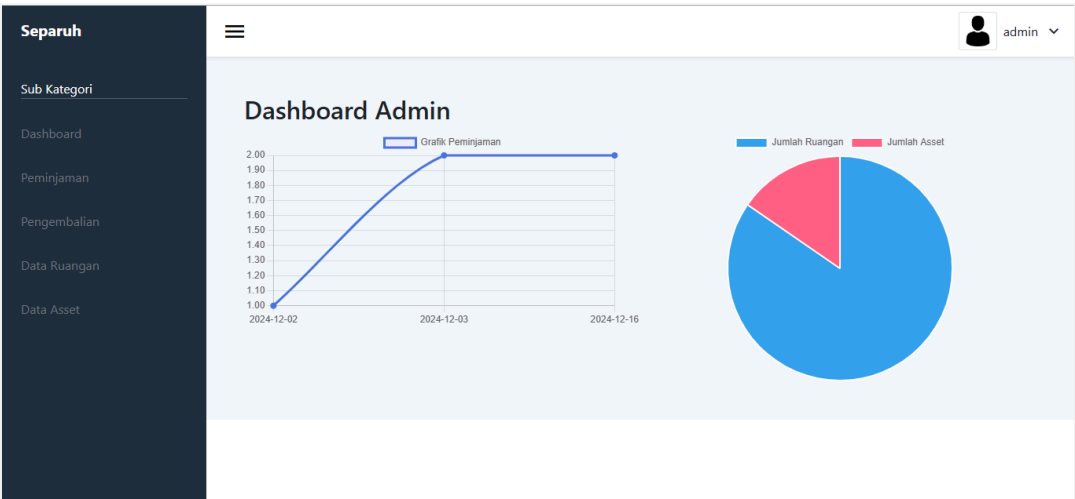
Jurusan : Teknik Informatika

Jenis Kelamin : Laki-Laki

Simpan

Gambar 26. Tampilan Menu Update Profile User

4.9.10. Tampilan Menu Dashboard Admin



Gambar 27. Tampilan Menu Dashboard Admin

4.9.11. Tampilan Menu Peminjaman

The screenshot shows the "Peminjaman" menu. It features a table with the following data:

ID	Peminjaman	Item ID	Tujuan	Waktu Mulai	Waktu Selesai	Aksi
P000000001		R00001	Musyawarah besar HMTI	2024-12-2 7:0:0	2024-12-3 7:0:0	-
P000000002		R00002	Mubes ROBOTIKA UMRAH	2024-12-4 7:0:0	2024-12-5 7:0:0	-
P000000003		R00006	Kegiatan LDK	2024-12-4 0:0:0	2024-12-4 0:0:0	-
P000000004		R00011	Rapat ROBOTIK	2024-12-16 7:0:0	2024-12-17 7:0:0	-
P000000005		R00003	Simulasi lomba GEMASTIK Programming tingkat Nasional	2024-12-16 0:0:0	2024-12-18 0:0:0	Terima Tolak

Gambar 28. Tampilan Menu Peminjaman Admin

4.9.12. Tampilan Menu Pengembalian

ID Peminjaman	Item ID	Tujuan	Waktu Mulai	Waktu Selesai	Aksi
P000000001	R00001	Musyawaharah besar HMTI	2024-12-2 7:0:0	2024-12-3 7:0:0	Proses
P000000002	R00002	Mubes ROBOTIKA UMRAH	2024-12-4 7:0:0	2024-12-5 7:0:0	Proses
P000000003	R00006	Kegiatan LDK	2024-12-4 0:0:0	2024-12-4 0:0:0	Dikembalikan
P000000004	R00011	Rapat ROBOTIK	2024-12-16 7:0:0	2024-12-17 7:0:0	Dikembalikan

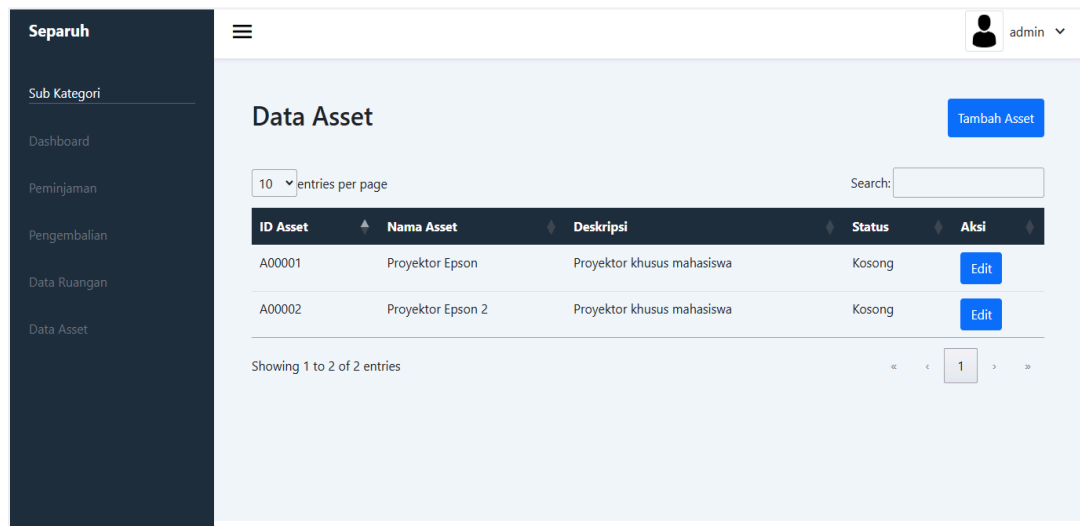
Gambar 29. Tampilan Menu Pengembalian Admin

4.9.13. Tampilan Menu Ruangan Admin

ID Ruangan	Nama Ruangan	Deskripsi	Kapasitas	Status	Aksi
R00001	Auditorium FTKK	Ruangan audit dengan 4 AC dan juga Proyektor	100	Dipakai	Edit
R00002	Lab TI	Ruangan laboratorium untuk praktikum	40	Dipakai	Edit
R00003	Lab TI 2	Ruangan laboratorium untuk praktikum	40	Dipakai	Edit
R00004	Lab TI 3	Ruangan laboratorium untuk praktikum	40	Kosong	Edit
R00005	Lab TI 4	Ruangan laboratorium untuk praktikum	40	Kosong	Edit

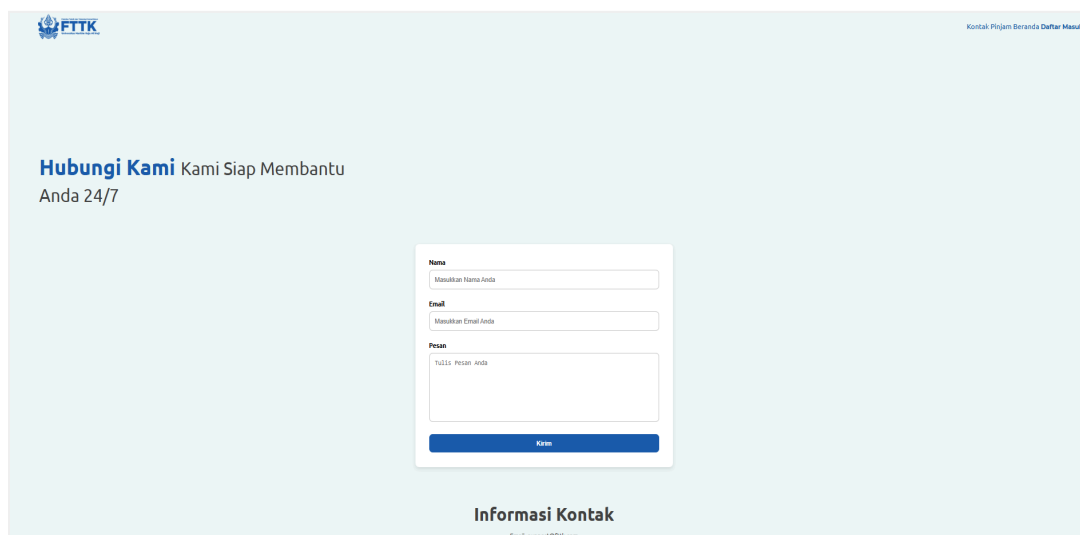
Gambar 30. Tampilan Menu Ruangan Admin

4.9.14. Tampilan Menu Asset Admin



Gambar 31. Tampilan Menu Asset Admin

4.9.15. Tampilan Kontak



Gambar 32. Tampilan Kontak

4.10. Spesifikasi Field Data Layer

Tabel 3. Spesifikasi Field Data Layer

Id_Objek	Jenis	Keterangan
ButtonRegis	Button	Jika di klik, akan diarahkan ke halaman login
InputUsername	Input	Untuk memasukkan nama pengguna/username
InputPassword	Input	Untuk memasukkan Password pengguna
ButtonForgotPassword	Button	Untuk mengganti password
ButtonLogin	Button	Untuk melanjutkan ke tahap peminjaman aset/barang
ButtonGoogle	Button	Untuk melakukan login dengan google
ContactButton	Button	Untuk menampilkan kontak yang bisa dihubungi
InfoCard	Card	Menampilkan Ruangan/aset yang tersedia dan jumlah mahasiswa yang mendaftar
ProfileLogoButton	Button	Untuk menampilkan pilihan untuk mengunjungi profile atau melakukan logout
LogoutButton	Button	Untuk melakukan logout
ProfileButton	Button	Untuk membuka halaman profile
UserTable	Table	Menampilkan Data diri User

Id_Objek	Jenis	Keterangan
UserRentTable	Table	Menampilkan data ruangan/aset yang dipinjam
UpdateBioButton	Button	Untuk melakukan pembaruan pada biodata
InputProfile	Input	Untuk memasukkan data pengguna
SaveButton	Button	Untuk menyimpan pembaruan biodata
DetailButton	Button	Untuk menampilkan secara detail tentang ruangan atau aset yang dipinjam oleh user
Return1Button	Button	Untuk masuk ke menu pengembalian
ChooseFileinputButton	Input Button	Untuk memilih file sebagai bukti pengembalian
StarInput	Input	Untuk memberikan rating pada ruangan/aset yang baru saja dipinjam
SuggestionInput	Input	Untuk memberikan saran pada ruangan/aset yang baru saja dipinjam
CloseButton	Button	Untuk membatalkan proses pengembalian
Return2Button	Button	Untuk mengkonfirmasi pengembalian
ButtonPeminjaman	Button	Untuk masuk ke menu peminjaman
ButtonRoom	Button	Untuk melakukan filter agar yang tertampil hanya ruangan

Id_Objek	Jenis	Keterangan
ButtonAssets	Button	Untuk melakukan filter agar yang tertampil hanya aset
ButtonHighest	Button	Untuk melakukan filter agar menampilkan ruangan/aset dari rating tertinggi
ButtonLowest	Button	Untuk melakukan filter agar menampilkan ruangan/aset dari rating terendah
ButtonMostRent	Button	Untuk melakukan filter agar menampilkan ruangan/aset dari rating paling sering dipinjam
ButtonReady	Button	Untuk melakukan filter agar menampilkan ruangan/aset yang tersedia
PaginationButton	Button	Untuk berpindah ke halaman yang diinginkan
ButtonNext	Button	Untuk berpindah ke halaman selanjutnya
ButtonPrevious	Button	Untuk berpindah kehalaman sebelumnya
ItemCard	Card	Untuk meninjau ruangan/aset yang ingin dipinjam
HomeButton	Button	Untuk kembali ke halaman beranda
TableRooms	Data Grid	Menampilkan daftar ruangan dari database pada halaman peminjaman ruangan.
TableAssets	Data Grid	Menampilkan daftar aset pada halaman peminjaman aset.
ButtonSubmitReq	Button	Jika di klik, mengirimkan permohonan peminjaman ruangan atau aset.

Id_Objek	Jenis	Keterangan
InputSearchRoom	Search Box	Inputan teks untuk mencari data ruangan yang tersedia.
LabelRoomDetails	Label	Menampilkan detail ruangan yang dipilih user.
ButtonLogout	Button	Jika diklik, menghapus sesi aktif dan mengarahkan ke halaman login.

4.11. Pengujian Sistem

Tabel 4. Hasil Pengujian Black Box

No	Aktivitas Pengujian	Realisasi yang diharapkan	Hasil Pengujian	Kesimpulan
1	Klik Masuk/Login pada menu home	Memuat halaman login	Sistem dapat menampilkan halaman login	[✓] Berhasil [] Tidak
2	Klik Daftar/Register pada menu home	Memuat halaman pendaftaran	Sistem dapat menampilkan halaman registrasi	[✓] Berhasil [] Tidak
3	Melakukan logout session	Sistem akan menghapus data sesi pengguna dari server.	Sistem dapat menghapus data sesi pengguna dari server dan redirect ke halaman	[✓] Berhasil [] Tidak

			beranda.	
4	Klik Kontak pada halaman home	Memuat halaman kontak	Sistem dapat menampilkan halaman kontak	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
5	Klik menu pinjam pada halaman home	Memuat halaman peminjaman	Sistem dapat menampilkan halaman peminjaman	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
6	Klik menu Profile pada halaman beranda	Memuat halaman profile pengguna serta data peminjaman yang diajukan.	Sistem dapat menampilkan halaman profile pengguna serta menampilkan data peminjaman pengguna.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
7	Klik tombol Sign-in untuk masuk ke dalam sistem	Memverifikasi kredensial pengguna	Sistem dapat memverifikasi dan menyimpan sesi pengguna	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

8	Klik tombol Register untuk membuat pengguna baru	Memverifikasi dan membuat data pengguna baru	Sistem dapat memverifikasi dan membuat data pengguna baru	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
9	Masuk sebagai admin	Menampilkan halaman admin	Sistem dapat memverifikasi sesi dan menampilkan data admin	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
10	Mengklik tombol Item Ruangan	Menampilkan detail informasi mengenai ruangan	Sistem dapat menampilkan data Ruangan yang sesuai.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
11	Mengklik tombol Item Asset	Menampilkan detail informasi mengenai asset	Sistem dapat menampilkan data Asset yang sesuai.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
12	Mengklik tombol Reservasi	Membuat data reservasi atau peminjaman ke dalam database.	Sistem dapat menerima permintaan pengguna dan menyimpan data permintaan ke dalam database	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

13	Mengklik Tombol Pengembalian	Mengajukan pengembalian kepada admin	Sistem dapat menangani permintaan pengguna dan mengupdate status peminjaman menjadi pengembalian.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
14	Input data diri	Sistem akan menampilkan menu pengeditan data diri pengguna.	Sistem dapat menampilkan menu pengeditan data diri pengguna.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
15	Mengklik tombol update data diri pengguna	Sistem akan memperbarui data pengguna ke dalam database.	Sistem dapat menangani permintaan pengguna dan memperbarui data profil pengguna.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
16	Menekan menu peminjaman pada halaman admin	Sistem akan menampilkan halaman dan data peminjaman pengguna.	Sistem dapat menampilkan halaman peminjaman serta menampilkan data peminjaman pengguna.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

17	Menekan menu pengembalian pada halaman admin	Sistem akan menampilkan halaman pengembalian admin serta data pengajuan pengembalian pengguna.	Sistem dapat menampilkan halaman pengembalian serta menampilkan data pengajuan pengembalian pengguna.	[✓] Berhasil [] Tidak
18	Menekan menu Dashboard pada halaman admin	Sistem akan menampilkan halaman admins serta data statistik peminjaman dalam bentuk grafik.	Sistem dapat menampilkan halaman dashboard admin dan menampilkan data peminjaman dalam bentuk grafik.	[✓] Berhasil [] Tidak
19	Menekan menu Ruangan pada halaman admin	Sistem akan menampilkan halaman management ruangan serta data ruangan pada database.	Sistem dapat menampilkan halaman ruangan dan juga menampilkan data ruangan dalam format tabel.	[✓] Berhasil [] Tidak
20	Menekan menu Asset pada halaman admin	Sistem akan menampilkan halaman management asset serta data asset pada database.	Sistem dapat menampilkan halaman asset dan juga menampilkan data asset dalam format tabel.	[✓] Berhasil [] Tidak

21	Menekan tombol terima pada halaman Peminjaman Admin	Sistem akan mengupdate status peminjaman menjadi 'Accepted' atau diterima.	Sistem dapat mengupdate status peminjaman menjadi 'Accepted' pada database.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
22	Menekan tombol tolak pada halaman Peminjaman Admin	Sistem akan mengupdate status peminjaman menjadi 'Rejected' atau diterima.	Sistem dapat mengupdate status peminjaman menjadi 'Rejected' pada database.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
23	Menekan tombol Tambah Ruangan pada halaman Ruangan Admin	Sistem akan menampilkan Form Input Data Ruangan Baru.	Sistem dapat menampilkan Form Input Data Ruangan.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
24	Menekan tombol Simpan pada menu Tambah Ruangan Admin	Sistem akan menyimpan data ruangan baru ke dalam database.	Sistem dapat menyimpan dan memvalidasi data ruangan baru ke dalam database.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
25	Menekan tombol Edit data Ruangan Admin	Sistem akan menampilkan menu edit data ruangan.	Sistem dapat dengan responsif menampilkan menu edit data ruangan.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

26	Menekan tombol Update data Ruangan Admin	Sistem akan menyimpan perubahan pada data ruangan yang ingin di update.	Sistem dapat memvalidasi dan menyimpan perubahan pada data ruangan yang ingin di update.	[✓] Berhasil [] Tidak
27	Melakukan Pencarian Data Ruangan	Sistem akan menampilkan data ruangan yang sesuai dengan kunci pencarian	Sistem dapat menampilkan data ruangan yang sesuai dengan kunci pencarian	[✓] Berhasil [] Tidak
28	Menekan tombol Tambah Asset pada halaman Asset Admin	Sistem akan menampilkan Form Input Data Asset Baru.	Sistem dapat menampilkan Form Input Data Asset.	[✓] Berhasil [] Tidak
29	Menekan tombol Simpan pada menu Tambah Asset Admin	Sistem akan menyimpan data asset baru ke dalam database.	Sistem dapat menyimpan dan memvalidasi data asset baru ke dalam database.	[✓] Berhasil [] Tidak
30	Menekan tombol Edit data Asset Admin	Sistem akan menampilkan menu edit data aset.	Sistem dapat dengan responsif menampilkan menu edit data aset.	[✓] Berhasil [] Tidak

31	Menekan tombol Update data Asset Admin	Sistem akan menyimpan perubahan pada data aset yang ingin di update.	Sistem dapat memvalidasi dan menyimpan perubahan pada data aset yang ingin di update.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
32	Melakukan Pencarian Data Asset	Sistem akan menampilkan data asset yang sesuai dengan kunci pencarian	Sistem dapat menampilkan data asset yang sesuai dengan kunci pencarian	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak
33	Menekan tombol aksesibilitas beranda	Sistem akan menampilkan menu aksesibilitas untuk pengguna disabilitas.	Sistem dapat menampilkan menu aksesibilitas untuk pengguna disabilitas secara responsif.	<input checked="" type="checkbox"/> Berhasil <input type="checkbox"/> Tidak

LAMPIRAN

Link Source Code :

<https://github.com/MChandraR/SEPARUH-Website>

Link Website :

<https://www.separuh.site/>