

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Alat**

Dalam pembuatan aplikasi digunakan alat dan bahan untuk mendukung berjalannya perancangan dan pembuatan aplikasi.

##### **3.1.1 Alat**

Alat yang digunakan dalam penelitian ini berupa Laptop atau notebook dengan spesifikasi sebagai berikut:

1. Processor : Intel(R) Core(TM) i7-4500U CPU @ 1.80GHz  
(4 CPUs), ~2.4 GHz
2. Memory : 4096MB RAM
3. Sistem Manufaktur : ASUSTeK COMPUTER INC.
4. Sistem Model : X450LC
5. Sistem Operasi : Windows 10 Education 64-bit (10.0, Build 16299)

Dengan menggunakan perangkat lunak yang mendukung, yaitu:

1. *Visual Studio 2015*
2. *SQL Server 2014 Management Studio*

##### **3.1.2 Bahan**

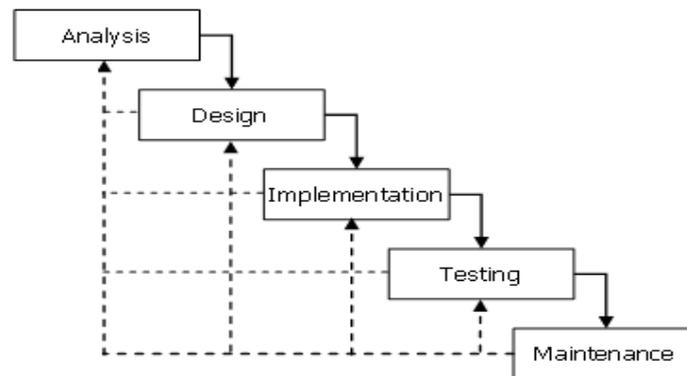
Bahan yang digunakan dalam penelitian ini adalah data-data mahasiswa yang akan melakukan pendaftaran pendadaran atau sidang tugas akhir (TA).

#### **3.2 Langkah Penelitian**

Untuk pengembangan sistem penelitian ini menggunakan model SDLC (Software Development Life Cycle). System Development Life Cycle (SDLC) adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sebuah sistem. SDLC juga merupakan pola yang diambil untuk mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap: rencana (planning), analisis (analysis), desain (design), implementasi (implementation), uji coba (testing) dan pengelolaan (maintenance).

Model SDLC yang dipakai dalam penelitian ini adalah model *Waterfall*. *Waterfall Model* atau *Classic Life Cycle* merupakan model yang paling banya

dipakai dalam *Software Engineering (SE)*. Menurut Bassil (2012) disebut *waterfall* karena tahap demi tahap yang harus dilalui menunggu selesainya tahap sebelumnya dan berjalan berurutan. Model *waterfall* menurut Bassil dapat dilihat pada gambar 3.1.



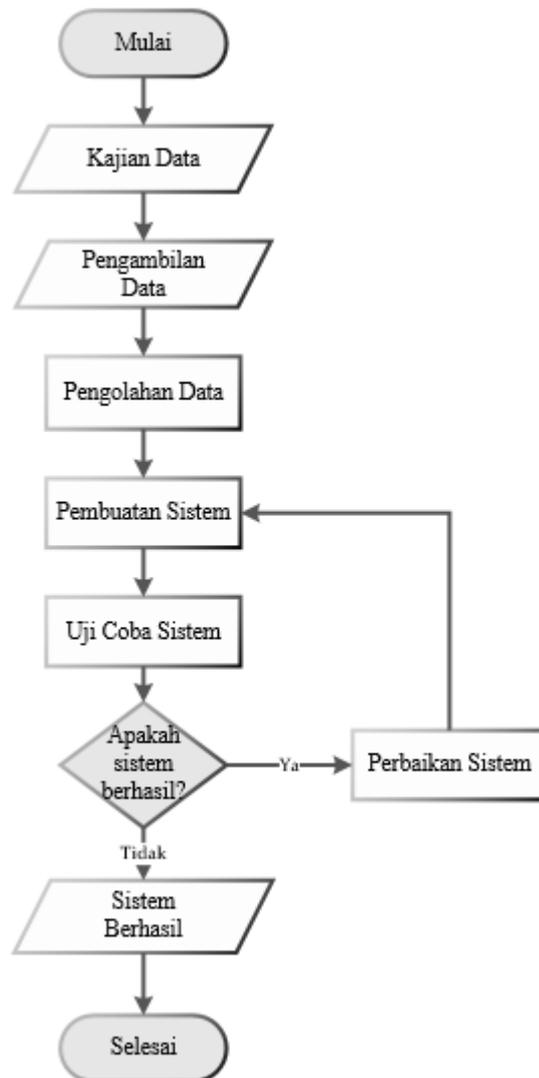
**Gambar 3. 1** Metode Waterfall menurut Bassil (2012)

### 3.3 Analisis Kebutuhan

Analisis kebutuhan yang didapat dari diskusi dengan dosen dan dari data lapangan, yaitu aplikasi yang dibutuhkan dimana memiliki data-data mengenai ruang yang digunakan untuk pendadaran, waktu berlangsungnya pendadaran, tanggal dilaksanakannya pendadaran, data mahasiswa yang akan melaksanakan pendadaran, data dosen yang terlibat, kategori dari judul masing-masing mahasiswa dan data operasional dosen pendadaran. Dibutuhkannya operasional dosen dikarenakan masing-masing dosen memiliki waktu sibuknya sendiri-sendiri, sehingga perlu pengisian data mengenai operasional dosen agar admin dapat menambahkan data kapan saja dosen dapat melaksanakan pendadaran. Untuk dosen penguji diusahakan sesuai dengan kategori yang diambil oleh masing-masing mahasiswa sesuai judul yang diambil. Namun apabila dosen penguji yang sesuai kategori tidak dapat hadir atau telah memiliki jadwal yang penuh, maka dosen penguji dapat bersifat *random*.

### 3.4 *Flowchart Tahapan Penelitian*

Berikut ini merupakan flowchart tahapan penelitian yang dilakukan oleh penulis dalam pembuatan “Sistem Penjadwalan Pendadaran atau Ujian Tugas Akhir Berbasis Deksktop”. *Flowchart* mengenai tahapan penelitian dapat dilihat pada gambar 3.1.



**Gambar 3. 2** *Flowchart Tahapan Penelitian*

### 3.5 *Perbandingan Sebelum dan Setelah Ada Sistem*

Sebelum adanya sistem ini, penjadwalan pendadaran dilakukan secara manual. Mahasiswa mencari sendiri dari dosen penguji, jadwal dosen pembimbing 1, jadwal dosen pembimbing 2, jadwal dosen penguji, hingga ruangan. Semua

dilakukan mahasiswa secara manual. Dan pendaftaran pendadaran sebelumnya dapat dilakukan 4 hari sebelum pelaksanaan.

Dengan adanya mahasiswa yang mencari jadwal masing-masing dosen secara manual sering menimbulkan jadwal yang terjadi secara bersamaan. Sehingga akan membuat mahasiswa serta dosen kurang efektif dalam pelaksanaan pendadaran.

Setelah adanya sistem ini, mahasiswa tidak perlu mencari jadwal dosen secara manual. Dikarenakan admin akan mendaftarkan jadwal dosen-dosen yang dapat melaksanakan pendadaran. Semua sudah disesuaikan dengan adanya ruangan. Apabila tersedia dosen pasti tersedia ruangan. Sehingga mahasiswa dan dosen tidak perlu khawatir apabila terjadi jadwal yang terjadi secara bersamaan.

Tidak hanya jadwal, namun pada sistem ini dosen penguji juga ditentukan oleh sistem. Penentuan dosen penguji ditentukan sesuai kategori judul masing-masing mahasiswa. Sehingga dosen penguji menguasai materi dari mahasiswa yang melakukan pendadaran. Gambaran proses bisnis sebelum dan sesudah dapat dilihat pada gambar 3.3.

**Gambar Proses Bisnis Sebelum dan Sesudah ada Sistem**



**Gambar 3.3** Proses Bisnis Sebelum dan Sesudah

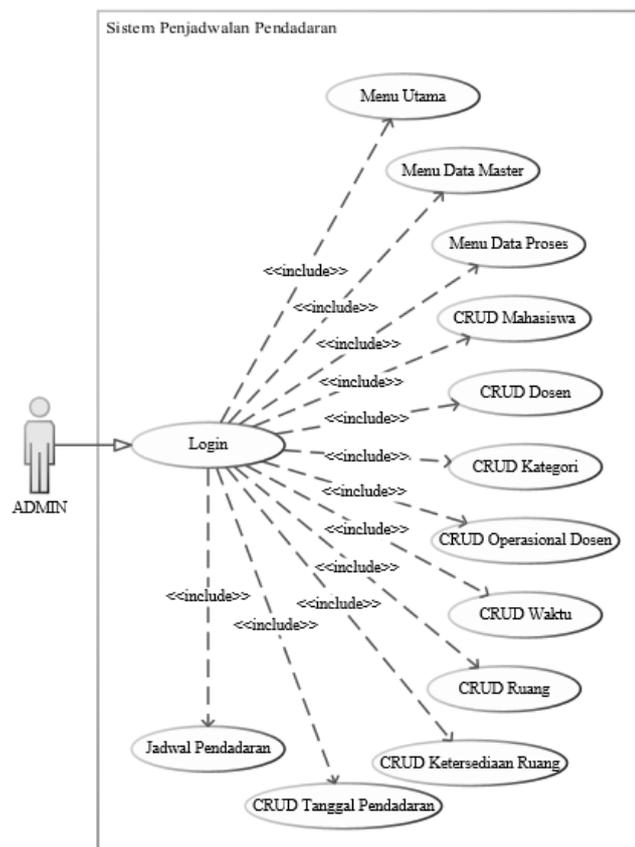
### 3.6 Rancangan Sistem

Rancangan sistem aplikasi *desktop* yang merancang mengenai sistem aktivitas yang ada pada aplikasi *desktop* yang dibuat dengan metode *Unified Modelling Language (UML)* merupakan sebuah bahasa yang berdasarkan grafik

atau gambar untuk memvisualisasikan, membangun dan mendokumentasikan dari sebuah sistem pembangun perangkat lunak berbasis objek, diantaranya menggunakan *use case diagram*, *activity diagram*, *flowchart diagram*, *ER Diagram* dan *class diagram* yang akan dijelaskan sebagai berikut:

### 3.6.1 Use Case Diagram

*Use Case Diagram* merupakan teknik permodelan yang digunakan untuk menjelaskan setiap aktivitas yang dimungkinkan terjadi dilihat dari sudut pandang seorang aktor sebagai pengguna sistem. Gambaran *use case diagram* yang digunakan pada aplikasi *desktop* dapat dilihat pada gambar 3.4.



**Gambar 3.4** Use Case Diagram

Berikut merupakan penjelasan mengenai *use case diagram* pada gambar 3.4:

1. Terdapat 1 aktor pada *use case diagram* pada sistem yaitu admin.
2. Admin dapat mengakses semua menu pada aplikasi *desktop*.

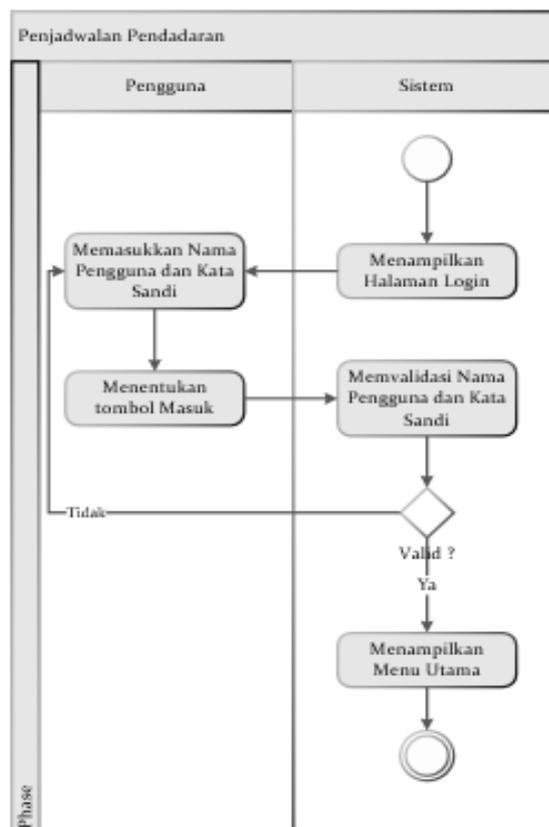
3. CRUD mahasiswa memungkinkan admin untuk menambahkan data mahasiswa yang melakukan pendaftaran pendadaran, mengedit data mahasiswa, menghapus data mahasiswa dan melihat data mahasiswa yang telah terdaftar.
4. CRUD dosen memungkinkan admin untuk menambahkan data dosen yang ada pada program studi Teknik Informatika, mengedit data dosen, menghapus data dosen dan melihat data dosen.
5. CRUD kategori memungkinkan admin untuk menambahkan data kategori yang ada pada program studi Teknik Informatika, mengedit data kategori, menghapus data kategori dan melihat data kategori.
6. CRUD operasional dosen memungkinkan admin untuk menambahkan data operasional dosen sesuai waktu yang telah dikonfirmasi sebelumnya, mengedit data operasional dosen, menghapus data operasional dosen dan melihat data operasional dosen.
7. CRUD waktu memungkinkan admin untuk menambahkan data jam pendadaran yang tersedia, mengedit data jam, menghapus data jam dan melihat data jam pendadaran.
8. CRUD ruang memungkinkan admin untuk menambahkan data ruang yang ada, mengedit data dosen, menghapus data dosen dan melihat data dosen.
9. CRUD ketersediaan ruang memungkinkan admin untuk menambahkan data ketersediaan ruang yang ada sesuai tanggal pendadaran, mengedit data ketersediaan ruang, menghapus data ketersediaan ruang dan melihat data ketersediaan ruang.
10. CRUD tanggal pendadaran memungkinkan admin untuk menambahkan data tanggal pendadaran, mengedit data tanggal pendadaran, menghapus data tanggal pendadaran dan melihat data tanggal pendadaran.
11. Jadwal pendadaran memungkinkan admin untuk melihat data mengenai jadwal bagi para pendaftar.

### 3.6.2 Activity Diagram

Activity Diagram akan menggambarkan aktivitas dalam perangkat lunak yang sedang di rancang, bagaimana aktivitas berawal, kemungkinan terjadinya decision, sampai bagaimana aktivitas berakhir. Berikut ini merupakan activity diagram pada aplikasi “Sistem Penjadwalan Pendaran atau Ujian Tugas Akhir berbasis Desktop” :

#### 1. Activity Diagram Masuk

Gambaran Activity Diagram proses masuk yang digunakan dalam aplikasi desktop dapat dilihat pada gambar 3.5.



**Gambar 3.5** Activity Diagram Masuk

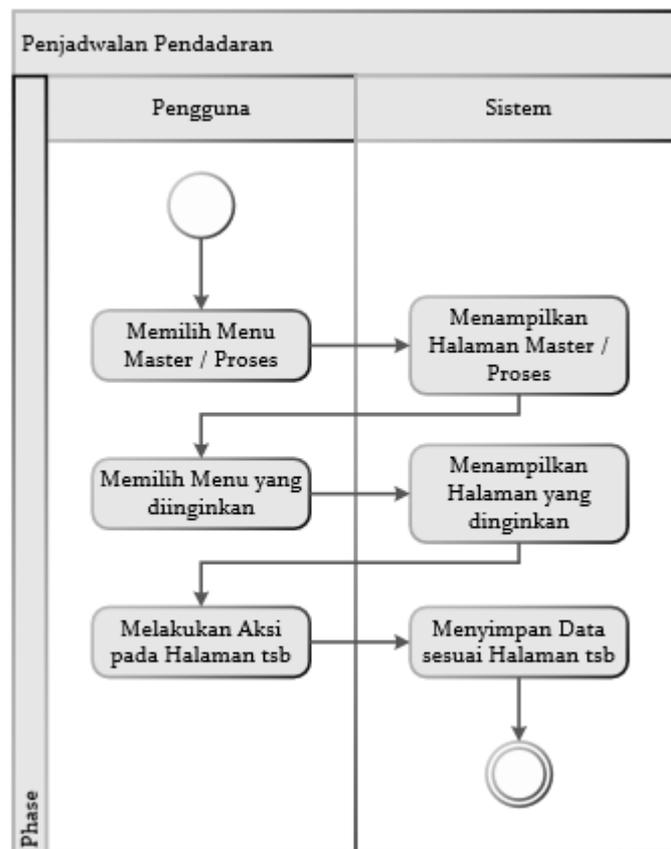
Keterangan:

- Sistem menampilkan halaman Masuk.
- Pengguna akan memasukkan Nama Pengguna dan Kata Sandi.
- Pengguna dapat menekan tombol Masuk, jika ditekan tombol Masuk maka sistem akan melakukan validasi pada Nama Pengguna dan Kata Sandi.

- d. Sistem melakukan validasi pada Nama Pengguna dan Kata Sandi, Jika salah maka sistem akan meminta pengguna untuk memasukkan kembali Nama Pengguna dan Kata Sandi. Jika benar maka sistem akan menampilkan menu utama.

## 2. Activity Diagram Proses

Gambaran Activity Diagram proses yang digunakan dalam aplikasi desktop dapat dilihat pada gambar 3.6.



**Gambar 3.6** Activity Diagram Proses

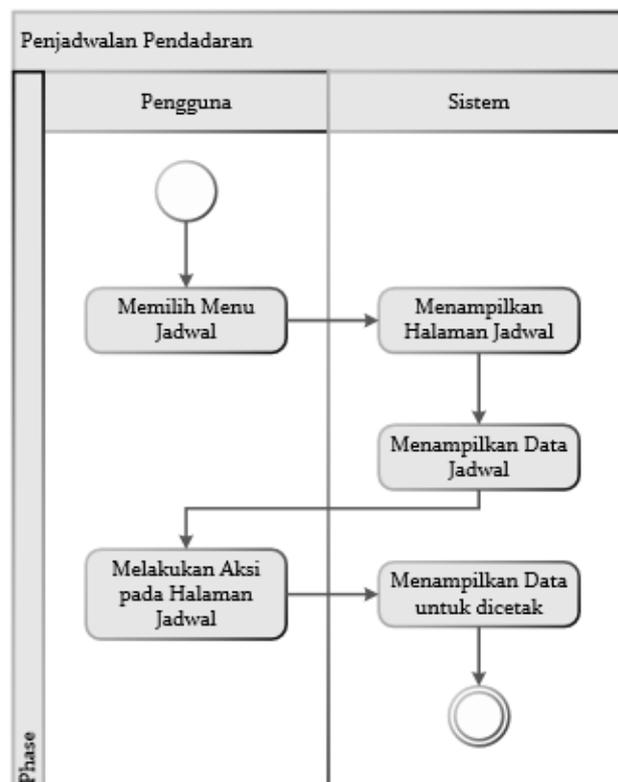
Keterangan:

- Pengguna memilih menu Master atau menu Proses pada menu utama.
- Sistem menampilkan halaman menu Master atau menu Proses.
- Pengguna memilih menu sesuai halaman yang diinginkan.
- Sistem akan menampilkan halaman yang diinginkan pengguna. Halaman yang dipilih dapat meliputi halaman Mahasiswa, Dosen, Kategori, Waktu, Ruang, Tanggal, Ketersediaan Ruang dan Operasional Dosen.

- e. Pengguna dapat melakukan aksi pada halaman , seperti menambah data, mengedit data dan menghapus data.
- f. Sistem akan menyimpan data sesuai dengan aksi yang telah dilakukan.

### 3. Activity Diagram Jadwal

Gambaran Activity Diagram proses jadwal yang digunakan dalam aplikasi desktop dapat dilihat pada gambar 3.7.



**Gambar 3.7** Activity Diagram Jadwal

Keterangan:

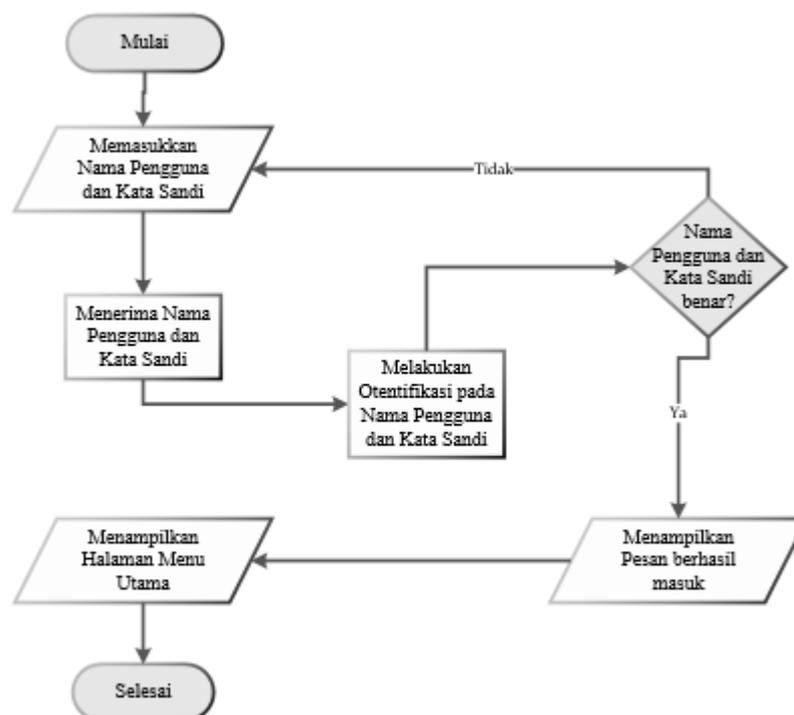
- a. Pengguna memilih menu Jadwal pada menu utama.
- b. Sistem menampilkan halaman Jadwal.
- c. Sistem akan menampilkan data Jadwal pada tabel.
- d. Pengguna memilih menu Cetak pada halaman Jadwal.
- e. Sistem akan melakukan aksi untuk mencetak sesuai data pada tabel.
- f. Sistem akan menampilkan data untuk dicetak.

### 3.6.3 Flowchart Diagram

*Flowchart* merupakan suatu bagan yang menunjukkan alur kerja yang sedang dikerjakan oleh sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur pada sistem. *Flowchart* pada sistem aplikasi ini menjelaskan mengenai alur proses dari mulai menambahkan data mahasiswa yang ingin mendaftarkan diri untuk pendadaran, jam operasional dosen, ruangan, jam hingga mahasiswa mendapatkan jadwal pendadaran itu sendiri. Admin bertugas untuk mengoperasikan sistem dan mengkonfirmasi kepada dosen dan ruangan. *Flowchart* mengenai aplikasi desktop dapat dilihat pada gambar-gambar dibawah ini.

#### 1. *Flowchart* Halaman Masuk

Berikut ini merupakan perancangan prosedur pada halaman Masuk, yang dapat dilihat pada Gambar 3.8.



**Gambar 3. 8** *Flowchart* Halaman Masuk

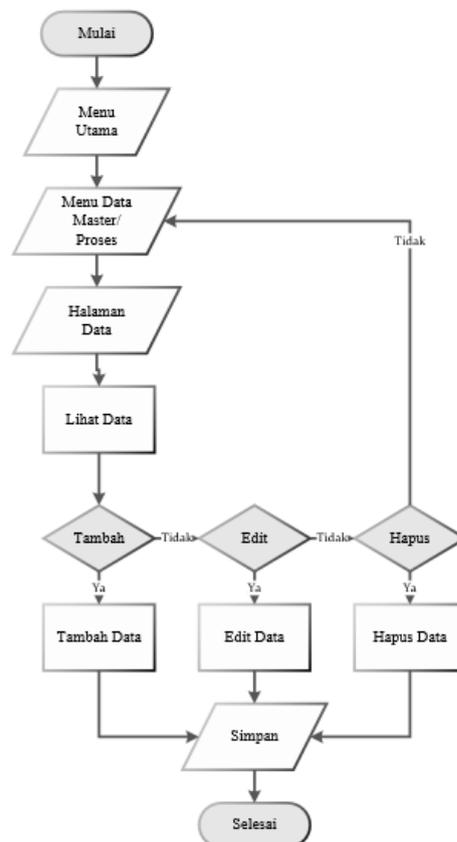
#### Keterangan:

1. Pengguna memasukkan Nama Pengguna dan Kata Sandi pada Halaman Masuk.

2. Sistem akan menerima Nama Pengguna dan Kata Sandi yang telah dimasukkan oleh pengguna.
3. Sistem akan melakukan otentifikasi pada nama Pengguna dan Kata Sandi tersebut.
4. Apabila pengguna benar dalam memasukkan Nama Pengguna dan Kata Sandi, maka sistem akan menampilkan pesan berhasil masuk. Apabila pengguna salah dalam memasukkan Nama Pengguna dan Kata Sandi, maka sistem akan tetap berada pada Halaman Masuk.
5. Setelah pengguna berhasil masuk, maka akan muncul halaman Menu Utama.

## 2. *Flowchart* Halaman Data

Berikut ini merupakan perancangan prosedur pada halaman-halaman Data, yang dapat dilihat pada Gambar 3.9.



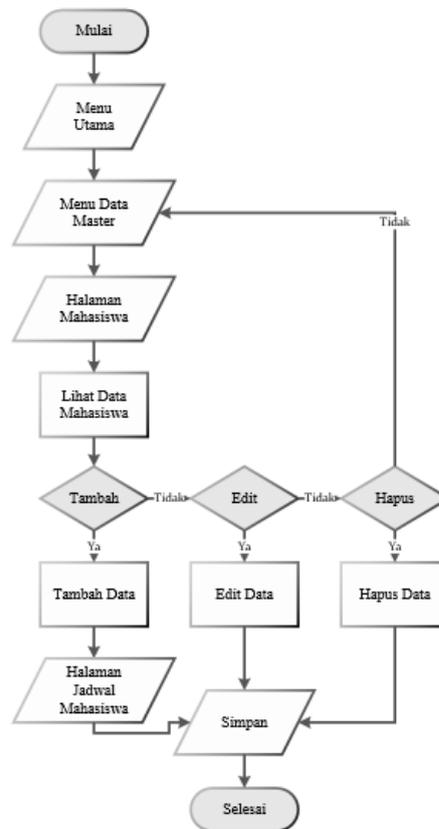
**Gambar 3.9** *Flowchart* Halaman Data

**Keterangan:**

1. Sistem akan menampilkan Halaman Menu Utama.
2. Pengguna dapat memilih Menu Master atau menu Proses.
3. Sistem akan menampilkan menu Master atau menu Proses.
4. Pengguna dapat memilih halaman yang diinginkan pada halaman Menu Master dan Menu Proses. Pengguna dapat menuju ke halaman Dosen, Kategori, Waktu, Ruang, Tanggal, Ketersediaan Ruang dan Operasional Dosen.
5. Sistem akan menampilkan sesuai halaman yang dipilih oleh pengguna.
6. Pengguna dapat melakukan aksi sesuai tombol-tombol yang ada pada halaman, seperti: simpan, edit, hapus dan kembali.
7. Sistem akan melakukan aksi sesuai tombol yang dipilih pengguna.

**3. *Flowchart* Halaman Mahasiswa**

Berikut ini merupakan perancangan prosedur pada halaman Mahasiswa, yang dapat dilihat pada Gambar 3.10.



**Gambar 3. 10** Flowchart Halaman Mahasiswa

**Keterangan:**

1. Sistem akan menampilkan Halaman Menu Utama.
2. Pengguna dapat memilih Menu Master atau menu Proses.
3. Sistem akan menampilkan menu Master atau menu Proses.
4. Pengguna milih halaman mahasiswa.
5. Sistem akan menampilkan halaman Mahasiswa.
6. Pengguna dapat melakukan aksi sesuai tombol-tombol yang ada pada halaman, seperti: simpan, edit, hapus dan kembali.
7. Sistem akan melakukan aksi sesuai tombol yang dipilih pengguna.
8. Apabila pengguna menekan tombol simpan, sistem akan menampilkan halaman jadwal mahasiswa yang bersangkutan. Sehingga data mahasiswa dan jadwalnya akan tersimpan.

#### 4. Flowchart Halaman Jadwal

Berikut ini merupakan perancangan prosedur pada halaman Jadwal, yang dapat dilihat pada Gambar 3.11.



**Gambar 3. 11** Flowchart Halaman Jadwal

#### Keterangan:

1. Sistem akan menampilkan Halaman Menu Utama.
2. Pengguna dapat memilih halaman Jadwal.
3. Sistem akan menampilkan halaman Jadwal.
4. Sistem akan menampilkan data penjadwalan pada tabel.
5. Pengguna dapat memilih tombol cetak dan kembali. Apabila pengguna memilih tombol cetak, maka pengguna dapat mencetak. Apabila pengguna memilih tombol kembali, maka sistem akan kembali di halaman Menu Utama.

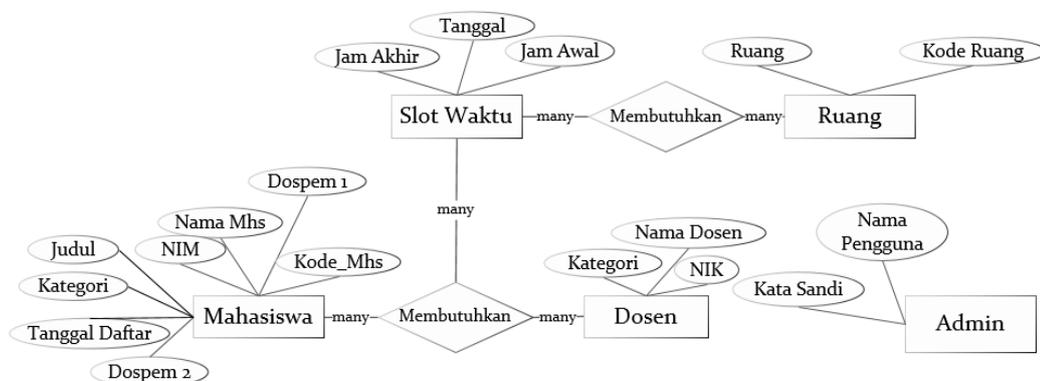
#### 3.6.4 Entity Relationship Diagram

*Entity Relationship* Diagram (ER Diagram) merupakan diagram yang digunakan untuk merancang basis data dan memperlihatkan relasi antar entitas atau

objek dengan atributnya. Tujuan ER Diagram dalam pengembangan aplikasi adalah dapat memberikan gambaran umum tentang sistem yang dikembangkan sehingga dapat memudahkan dalam merancang basis data. Gambaran ER Diagram yang digunakan dalam aplikasi dapat dilihat pada gambar 3.12.

Pada gambar 3.12 dapat dilihat bahwa *database* yang dirancang memiliki 5 buah entitas yaitu:

- a. Admin
- b. Mahasiswa
- c. Dosen
- d. Slot Waktu
- e. Ruang



**Gambar 3. 12** Entity Relationship Diagram

Penjelasan ERD pada gambar 3.12 adalah sebagai berikut:

1. Entitas *Admin* tidak memiliki relasi dengan entitas lain. Jadi entitas *Admin* tidak bergantung pada entitas lain.
2. Entitas mahasiswa mempunyai relasi *many-to-many* dengan entitas dosen, yang artinya mahasiswa dapat memilih lebih dari satu dosen dan dosen dapat dipilih lebih dari satu dosen, maka relasi ini akan menciptakan tabel baru.
3. Entitas dosen mempunyai relasi *one-to-many* dengan entitas slot waktu, yang artinya dosen dapat mengambil lebih dari satu slot waktu dan slot waktu dapat diambil oleh lebih dari satu dosen, maka relasi ini akan menciptakan tabel baru.
4. Entitas slot waktu mempunyai relasi *one-to-many* dengan entitas ruang, yang artinya slot waktu dapat mengambil lebih dari satu ruang dan entitas ruang dapat diambil oleh lebih dari satu slot waktu, maka relasi ini akan menghasilkan tabel baru.

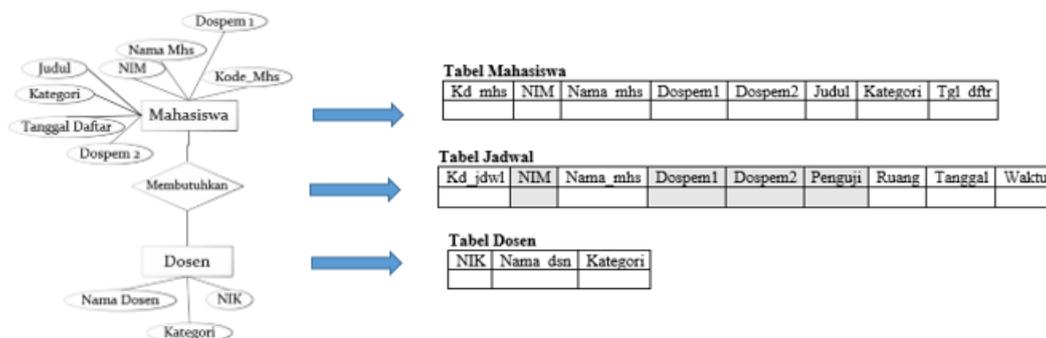
### 3.6.5 Konversi ERD

#### 1. Konversi dengan relasi (*many to many*)

Konversi ERD ke dalam bentuk tabel adalah proses memasukan entitas menjadi nama tabel dan attribute menjadi nama kolom beserta relasinya ke dalam sebuah tabel yang saling berhubungan.

##### a. Entitas Mahasiswa dengan Entitas Dosen

Penjelasan konversi relasi antara entitas mahasiswa dengan entitas dosen pada gambar 3.13.



**Gambar 3. 13** Entitas Mahasiswa dengan Dosen

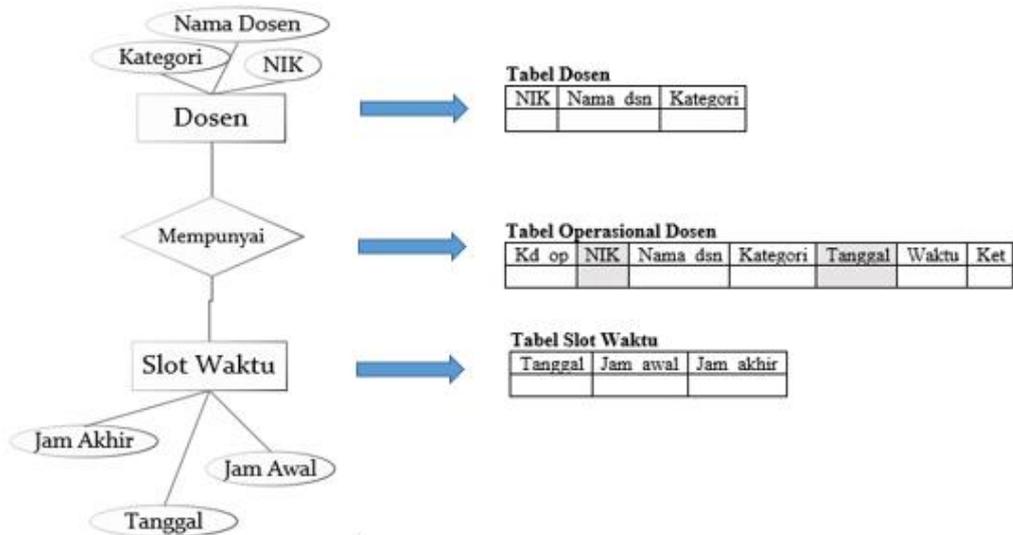
Entitas mahasiswa memiliki relasi *many to many* dengan entitas dosen sehingga tercipta tabel baru yang menghubungkan entitas mahasiswa dengan entitas dosen yang diberi nama tabel jadwal, dimana kolom NIM pada tabel mahasiswa dan kolom Nama dosen pada tabel dosen ditambahkan ke dalam tabel jadwal sebagai *foreign key*, maka perlu ditambahkan kode\_jadwal sebagai *primary key* dan kolom lainnya sebagai penambah informasi seperti, ruang, tanggal dan waktu.

##### b. Entitas Dosen dengan Entitas Slot Waktu

Penjelasan konversi relasi antara entitas dosen dengan entitas slot waktu pada gambar 3.14.

Entitas dosen memiliki relasi *many to many* dengan entitas slot waktu sehingga tercipta tabel baru yang menghubungkan entitas dosen dengan entitas slot waktu yang diberi nama tabel operasional dosen, dimana kolom NIK pada tabel dosen dan kolom tanggal pada tabel slot dosen ditambahkan ke dalam tabel operasional dosen, maka perlu ditambahkan kode\_opdosen sebagai *primary key* dan

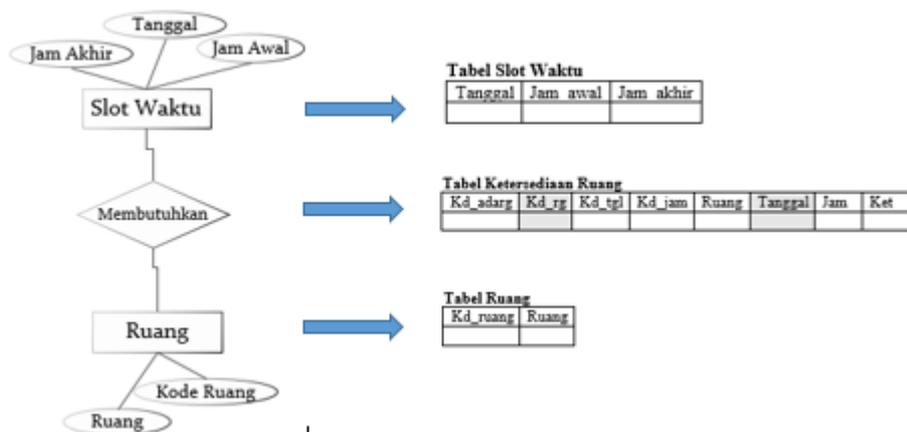
kolom lainnya sebagai penambah informasi seperti, nama dosen, kategori, waktu dan keterangan.



**Gambar 3. 14** Entitas Dosen dengan Slot Waktu

**c. Entitas Slot Waktu dengan Entitas Ruang**

Penjelasan konversi relasi antara entitas slot waktu dengan entitas ruang pada gambar 3.15.



**Gambar 3. 15** Entitas Slot Waktu dengan Ruang

Entitas slot dosen memiliki relasi *many to many* dengan entitas ruang sehingga tercipta tabel baru yang menghubungkan entitas slot waktu dengan entitas ruang yang diberi nama tabel ketersediaan ruang, dimana kolom tanggal pada tabel slot waktu dan kolom kode ruang pada tabel ruang ditambahkan ke dalam tabel ketersediaan ruang sebagai *foreign key*, maka perlu ditambahkan kode\_adaruang sebagai *primary key* dan kolom lainnya sebagai penambah informasi seperti, kode tanggal, kode jam, ruang, jam dan keterangan.

### 3.6.6 Normalisasi Tabel

Normalisasi *database* digunakan untuk menghilangkan dan mengurangi redundansi data dan tujuan lainnya yaitu untuk memastikan dependensi data (data berada pada tabel yang tepat. Berikut merupakan tabel yang perlu dinormalisasi, yaitu:

#### 1. Normal Form

Normal form merupakan keseluruhan data yang dibutuhkan dalam sistem penjadwalan tugas akhir. Sehingga data yang ditambahkan secara mentah dalam suatu tabel pada bentuk ini mungkin terjadi inkonsistensi dan anomali data. Tampilan data norma form dapat dilihat pada gambar 3.16.

Kode	NIM	Mhs	Dospem1	Dospem2	Penguji	Judul	Kategori	Ruang	Tanggal	Waktu	TglDft

**Gambar 3. 16** Normal Form

#### 2. Normalisasi Pertama (1NF)

Normalisasi yang pertama berfungsi untuk membuat tabel terpisah untuk masing-masing kelompok data yang berkaitan dan setiap baris mempunyai kolom yang unik (*primary key*). Dari normal form lalu dipisah menjadi dua baris agar mengurangi kesalahan-kesalahan yang dimungkinkan. Tampilan data normalisasi pertama (1NF) dapat dilihat pada gambar 3.17.

Kd_jdwl	NIM	Mhs	Dospem1	Dospem2	Penguji	Ruang	Tgl	Wkt

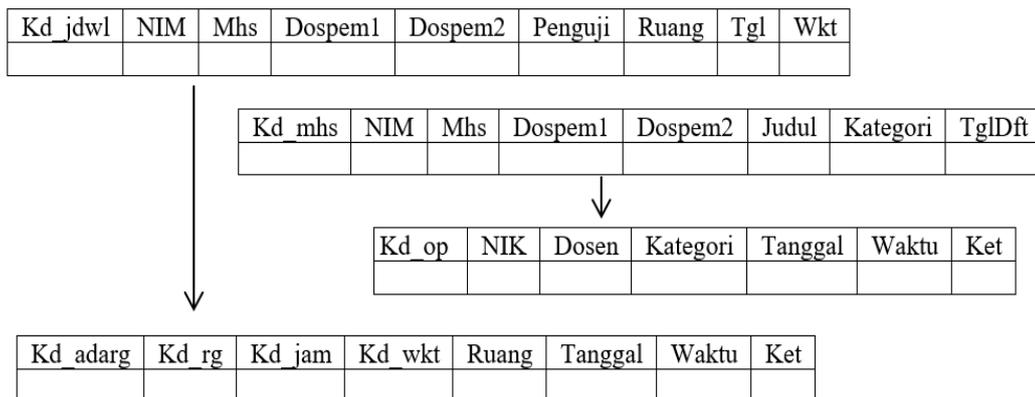
  

Kd_mhs	NIM	Mhs	Dospem1	Dospem2	Judul	Kategori	TglDft

**Gambar 3. 17** Normalisasi Kedua (2NF)

### 3. Normalisasi Kedua (2NF)

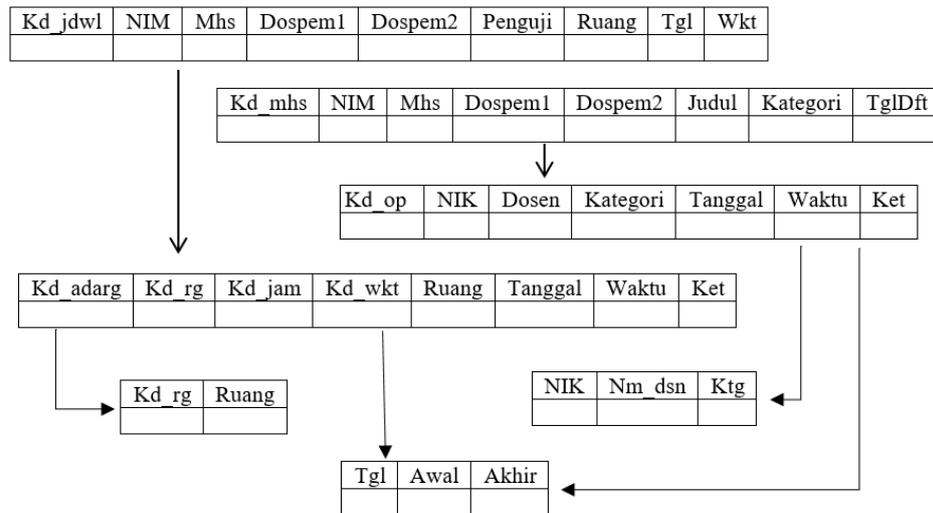
Pada normalisasi kedua ini mempunyai syarat yaitu data telah dibentuk dalam 1NF. Pada normalisasi kedua memiliki fungsi untuk menciptakan hubungan antara tabel baru dan tabel lama dengan menciptakan foreign key dan menghilangkan atau menghapus beberapa subset yang berada pada tabel dan menempatkannya pada tabel yang terpisah. Tampilan normalisasi kedua dapat dilihat pada gambar 3.18.



**Gambar 3. 18** Normalisasi Ketiga (3NF)

### 4. Normalisasi Ketiga (3NF)

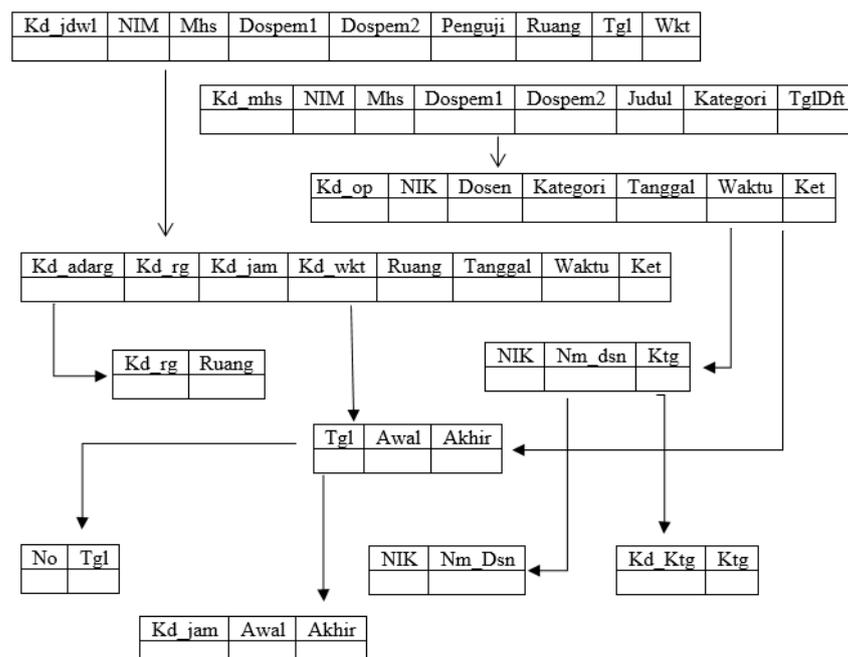
Pada normalisasi ketiga ini mempunyai fungsi untuk menciptakan baris tabel baru yang dimungkinkan untuk mengurangi kesalahan yang ada pada baris sebelumnya. Pada normalisasi ketiga data menjadi lebih rinci dari 2NF. Tampilan data normalisasi ketiga (3NF) dapat dilihat pada gambar 3.19.



**Gambar 3. 19** Normalisasi Ketiga (3NF)

## 5. Normalisasi Keempat (4NF)

Pada normalisasi keempat ini baris yang dibuat pada 3NF dibuat menjadi lebih rinci sehingga tidak dimungkinkan terjadinya redundansi data. Dan ini merupakan normalisasi yang terakhir dikarenakan sudah menjadi yang paling rinci, dan baris tidak dapat dinormalisasi kembali. Tampilan normalisasi keempat (4NF) dapat dilihat pada gambar 3.20.



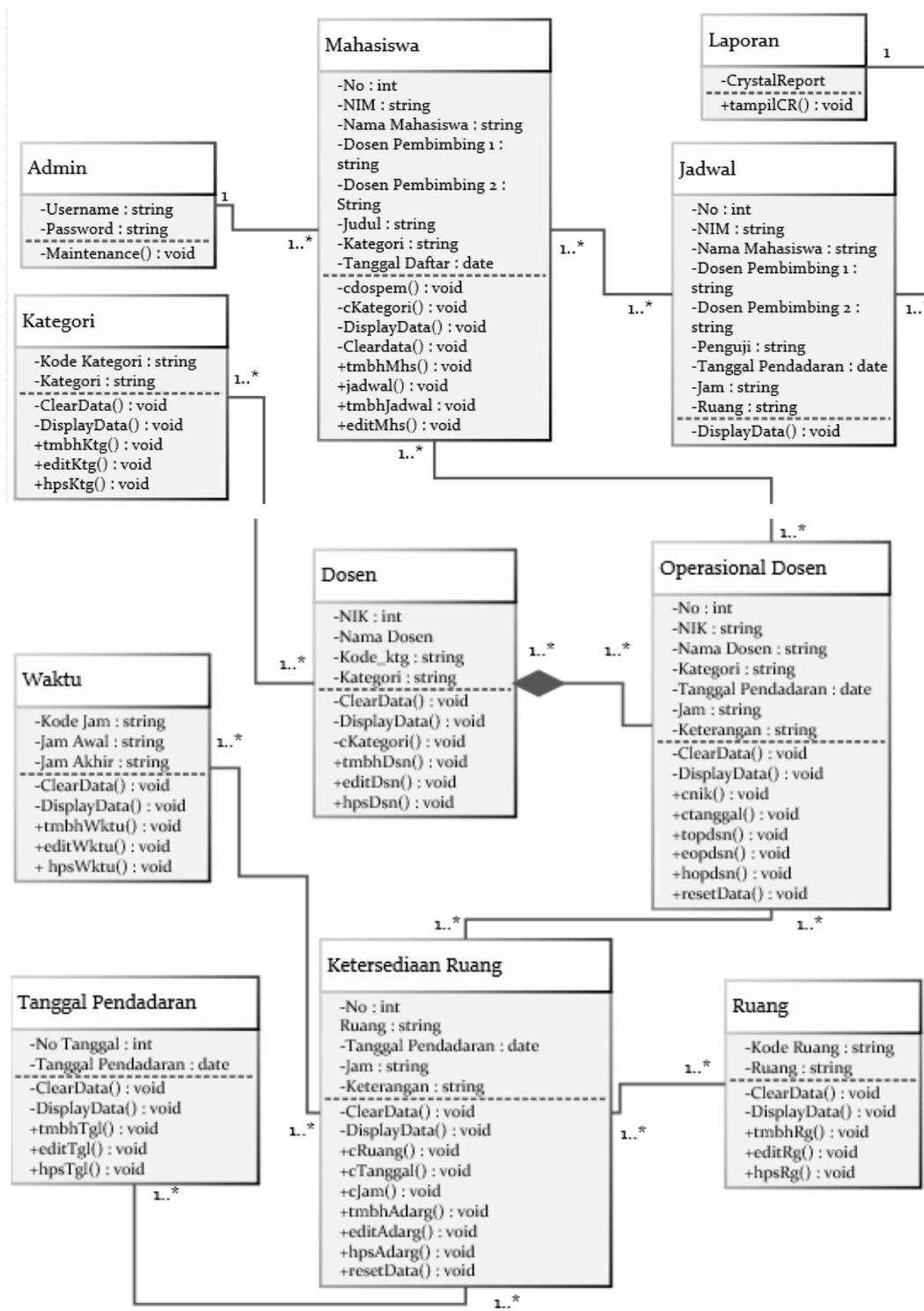
**Gambar 3. 20** Normalisasi Keempat (4NF)

### 3.6.7 Class Diagram

Gambaran Class Diagram yang digunakan dalam aplikasi desktop dapat dilihat pada gambar 3.21.

Pada gambar 3.21 dapat dilihat bahwa *class diagram* memiliki 10 *class*, yaitu:

1. Admin
2. Jadwal
3. Mahasiswa
4. Operasional Dosen
5. Kategori
6. Dosen
7. Ketersediaan Ruang
8. Ruang
9. Jam
10. Tanggal Pendaftaran
11. Laporan



Gambar 3. 21 Class Diagram

### 3.6.8 Kamus Data Database

Berikut ini merupakan tabel-tabel dari kamus database pada aplikasi “Sistem Penjadwalan Pendaran atau Ujian Tugas Akhir Berbasis Dekstop”.

#### 1. Kamus Data Tabel Admin

Kamus data tabel admin yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.1.

**Tabel 3. 1** Kamus Data Tabel Admin

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
pengguna	Varchar	20	Primary Key	Nama Pengguna
Kt_sandi	Varchar	8		Kata Sandi Penguna
Jawab	Varchar	20		

#### 2. Kamus Data Tabel Dosen

Kamus data tabel dosen yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.2.

**Tabel 3. 2** Kamus Data Tabel Dosen

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
nik	Char	20	Primary Key	NIK Dosen
Nm_dsn	Varchar	50		Nama Dosen
Kode_ktg	Varchar	5		Kode Kategori
Kategori	Varchar	30		Nama Kategori

#### 3. Kamus Data Tabel Kategori

Kamus data tabel kategori yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.3.

**Tabel 3. 3** Kamus Data Tabel Kategori

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kdktg	Varchar	5	Primary Key	Kode Kategori
Kategori	Varchar	30		Nama Kategori

#### 4. Kamus Data Tabel Waktu

Kamus data tabel jam yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.4.

**Tabel 3. 4** Kamus Data Tabel Jam

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kode_jam	Nvarchar	8	Primary Key	Kode Jam
Jam_awal	Nvarchar	5		Jam pendadaran dimulai
Jam_akhir	Nvarchar	5		Jam pendadaran berakhir

#### 5. Kamus Data Tabel Ruang

Kamus data tabel ruang yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.5.

**Tabel 3. 5** Kamus Data Tabel Ruang

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kode_rg	Nvarchar	10	Primary Key	Kode Ruangan
Ruang	Varchar	20		Nama Ruangan

#### 6. Kamus Data Tabel Tanggal

Kamus data tabel tanggal yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.6.

**Tabel 3. 6** Kamus Data Tabel Tanggal Pendadaran

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kode_tgl	Int		Primary Key	
Tanggal	Date			Tanggal Pendadaran

### 7. Kamus Data Tabel Ketersediaan Ruang

Kamus data tabel ketersediaan ruang yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.7.

**Tabel 3. 7** Kamus Data Tabel Ketersediaan Ruang

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kode_adarg	Int		Primary Key	No tabel
Kode_rg	Nchar	10		Kode Ruang
Kode_tgl	Int			Kode Tanggal
Kode_jam	Nchar	8		Kode Waktu
Ruang	Varchar	20		Nama Ruangan
Tanggal	Date			Tanggal Pendadaran
Jam	Nvarchar	15		Jam Pendadaran
Ket	Varchar	6		Keterangan Ruangan

### 8. Kamus Data Tabel Operasional Dosen

Kamus data tabel dosen pembimbing 1 yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.8.

**Tabel 3. 8** Kamus Data Tabel Operasional Dosen

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kode_op	Int		Primary Key	No tabel
Nik	Char	20		NIK Dosen
Nm_dsn	Varchar	50		Nama Dosen
Kategori	Varchar	30		Nama Kategori

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Tanggal	Date			Tanggal Pendadaran
Jam	Nvarchar	15		Jam Pendadaran
Ket	Varchar	6		Keterangan Dosen

### 9. Kamus Data Tabel Mahasiswa

Kamus data tabel mahasiswa yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.9.

**Tabel 3. 9** Kamus Data Tabel Mahasiswa

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kode_mhs	Int		Primary Key	
Nim	Char	11		NIM Mahasiswa
Nm_mhs	Varchar	50		Nama Mahasiswa
Dospem1	Varchar	50		Nama Dosen Pembimbing 1
Dospem2	Varchar	50		Nama Dosen Pembimbing 2
Judul	Varchar	300		Judul Skripsi
Kategori	Varchar	30		Kategori Judul Skripsi
Tgl_dft	Date			Tanggal Daftar Pendadaran

### 10. Kamus Data Tabel Penjadwalan

Kamus data tabel penjadwalan yang digunakan dalam aplikasi desktop dapat dilihat pada tabel 3.10.

**Tabel 3. 10** Kamus Data Penjadwalan

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
Kode_jdwl	Int		Primary Key	No Jadwal
Nim	Nchar	11		NIM Mahasiswa
Nm_mhs	Varchar	50		Nama Mahasiswa
Nm_dsn1	Varchar	50		Nama Dosen Pembimbing 1
Nm_dsn2	Varchar	50		Nama Dosen Pembimbing 2
Nm_pg	Varchar	50		Nama Dosen Penguji
Ruang	Varchar	20		Nama Ruangan
Tanggal	Date			Tanggal Pendadaran
Jam	Nvarchar	15		Jam Pendadaran

### 3.7 Rancangan Antar Muka

Rancangan antar muka merupakan bagian yang sangat penting pada sebuah sistem. Karena ini merupakan bagian perangkat lunak sebagai media perantara sistem dan pengguna. Rancangan antar muka ini yang menentukan mudah atau sulitnya suatu sistem yang dibuat. Berikut ini merupakan rancangan antar muka yang dibuat oleh penulis:

#### 3.5.1 Tampilan Halaman Masuk

Ini merupakan tampilan antarmuka pertama kali saat aplikasi desktop diakses. Tampilan halaman masuk ini sebagai langkah awal, sehingga harus memasukkan nama pengguna dan kata sandi terlebih dahulu. Pada halaman ini memiliki dua kontrol *textbox*, yaitu: *textbox* nama pengguna dan *textbox* kata sandi. Halaman ini juga memiliki dua *button*, yaitu: *button* masuk dan *button* keluar, dan tiga *label*, yaitu: *label* masuk, *label* nama pengguna dan *label* kata sandi. Gambaran rancangan antarmuka halaman masuk ini dapat dilihat pada gambar 3.22.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Masuk

Nama Pengguna

Kata Sandi

Masuk Keluar

**Gambar 3. 22** Halaman Masuk

### 3.5.2 Tampilan Halaman Menu

Halaman menu utama merupakan halaman yang akan muncul setelah melewati halaman masuk. Pada halaman menu utama terdapat beberapa pilihan yang akan dipilih oleh pengguna sesuai kebutuhannya. Halaman ini terdapat satu *label* menu dan lima *button*, yaitu: *button* data master, *button* data proses, *button* data jadwal, *button* edit admin dan *button* keluar. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.23.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Menu Utama

Data Master

Data Proses

Data Jadwal

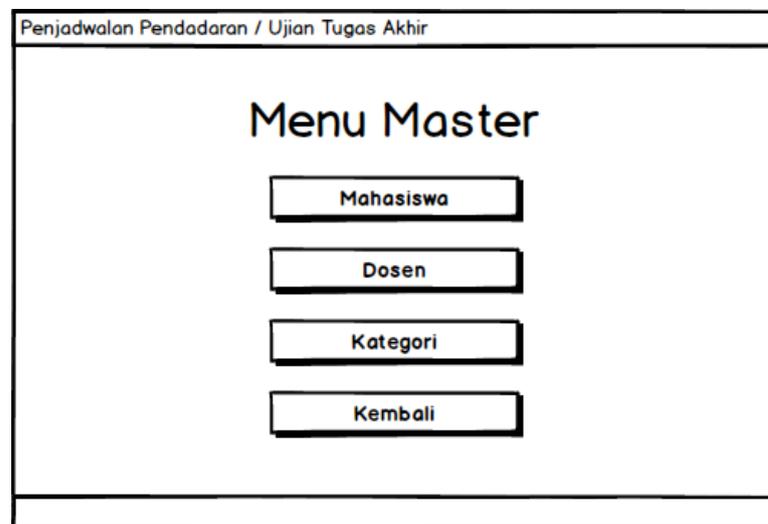
Edit Admin

Keluar

**Gambar 3. 23** Halaman Menu

### 3.5.3 Tampilan Halaman Menu Data Master

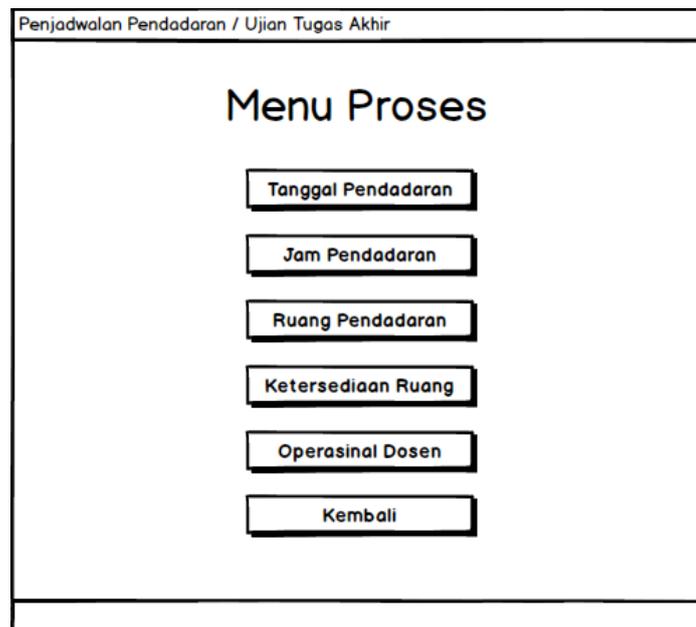
Ini merupakan rancangan halaman antarmuka data master. Halaman ini akan muncul apabila pada halaman menu utama pengguna memilih *button* data master. Pada halaman ini juga memiliki beberapa pilihan bagi pengguna. Halaman menu data master terdapat satu *label* menu data master dan empat *button*, yaitu: *button* mahasiswa, *button* dosen, *button* kategori dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.24.



Gambar 3. 24 Menu Data Master

### 3.5.4 Tampilan Halaman Menu Proses

Ini merupakan rancangan halaman antarmuka proses. Halaman ini akan muncul apabila pada halaman menu utama pengguna memilih *button* proses. Pada halaman ini juga memiliki beberapa pilihan bagi pengguna. Halaman menu proses terdapat satu *label* menu proses dan enam *button*, yaitu: *button* operasional dosen, *button* jam pendadaran, *button* ruangan, *button* tanggal pendadaran, *button* Ketersediaan Ruang dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.25.



**Gambar 3. 25** Menu Proses

### 3.5.5 Tampilan Halaman Mahasiswa

Rancangan antarmuka halaman data mahasiswa ini akan muncul apabila pengguna memilih *button* mahasiswa pada halaman menu data master. Halaman ini merupakan halaman untuk pengisian data mahasiswa. Jadi, pengguna harus menambahkan data mahasiswa yang mendaftar untuk pendadaran pada halaman ini. Pada halaman ini terdapat empat belas *label*, yaitu: *label* mahasiswa, *label* NIM, *label* nama, dua *label* dosen pembimbing 1, dua *label* dosen pembimbing 2, *label* penguji, *label* judul, *label* kategori, *label* ruang, *label* tanggal dan *label* jam, sembilan *textbox*, yaitu: *textbox* nim, *textbox* nama, *textbox* judul, *textbox* dosen pembimbing1, *textbox* dosen pembimbing 2, *textbox* penguji, *textbox* ruang, *textbox* tanggal dan *textbox* jam, tiga *combobox*, yaitu: *combobox* dosen pembimbing 1, *combobox* dosen pembimbing 2 dan *combobox* kategori, satu *datagridview* data mahasiswa dan tiga *button*, yaitu: *button* tambah, *button* edit, dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.26.

Penjadwalan Pendadaran / Ujian Tugas Akhir

### Data Mahasiswa Pendaftar Ujian Tugas Akhir

NIM

Nama

Dosen Pembimbing 1

Dosen Pembimbing 2

Judul

Kategori

Tambah Edit Kembali

Jadwal Pendadaran :

Dosen Pembimbing 1

Dosen Pembimbing 2

Dosen Penguji

Ruang

Tanggal

Waktu

NIM	Nama	Dospem1	Dospem 2	Judul	Bidang	Tanggal Daftar

**Gambar 3. 26** Halaman Mahasiswa

### 3.5.6 Tampilan Halaman Dosen

Rancangan antarmuka halaman data dosen ini akan muncul apabila pengguna memilih *button* dosen pada halaman menu data master. Halaman ini merupakan halaman untuk pengisian data dosen. Jadi, pengguna harus menambahkan data dosen yang ada pada program studi Teknik Informatika sesuai halaman ini. Pada halaman antarmuka dosen terdapat empat *label*, yaitu: *label* dosen, *label* NIK, *label* nama dan *label* kategori, tiga *textbox*, yaitu: *textbox* nik, *textbox* nama dan *textbox* kategori, satu *combobox* kode kategori, satu *datagridview* dan lima *button*, yaitu: *button* tambah, *button* edit, *button* hapus dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.27.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Data Dosen

NIP

Nama

Kode  Kategori

No	NIP	Nama

**Gambar 3. 27** Halaman Dosen

### 3.5.7 Tampilan Halaman Kategori

Rancangan antarmuka halaman data kategori ini akan muncul apabila pengguna memilih *button* kategori pada halaman menu data master. Halaman ini merupakan halaman untuk pengisian data beberapa kategori. Jadi, pengguna harus menambahkan macam-macam kategori yang ada pada program studi Teknik Informatika. Pada halaman antarmuka kategori terdapat tiga *label*, yaitu: *label* kategori untuk judul, *label* kode kategori dan *label* kategori, dua *textbox*, yaitu: *textbox* kode kategori dan *textbox* kategori, satu *datagridview* kategori dan lima *button*, yaitu: *button* tambah, *button* edit, *button* hapus dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.28.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Data Kategori

Kode Kategori

Kateg

No	Kode Kategori	Kategori

**Gambar 3. 28** Halaman Kategori

### 3.5.8 Tampilan Halaman Operasional Dosen

Rancangan antarmuka halaman data operasional dosen ini akan muncul apabila pengguna memilih *button* operasional dosen pada halaman menu proses. Halaman ini merupakan halaman untuk pengisian data mengenai operasional dosen. Jadi, pengguna harus melakukan konfirmasi terlebih dahulu kepada dosen sebelum melakukan pengisian halaman operasional dosen. Pada halaman antarmuka operasional dosen terdapat enam *label*, yaitu: *label* operasional dosen, *label* NIP, *label* nama, *label* kategori, *label* tanggal pendadaran dan *label* jam pendadaran, *textbox* nama, empat *combobox*, yaitu: *combobox* NIK, *combobox* kategori, *combobox* tanggal pendadaran dan *combobox* jam pendadaran, satu *datagridview* data operasional dosen dan lima *button*, yaitu: *button* tambah, *button* edit, *button* hapus dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.29.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Operasional Dosen

NIK

Nama Dosen

Ruang

Tanggal

Jam

No	NIK	Nama Dosen	Kategori	Tanggal	Jam	Ket

**Gambar 3. 29** Operasional Dosen

### 3.5.9 Tampilan Halaman Jam Pendadaran

Rancangan antarmuka halaman data jam pendadaran ini akan muncul apabila pengguna memilih *button* jam pendadaran pada halaman menu proses. Halaman ini merupakan halaman untuk pengisian data mengenai waktu pelaksanaan pendadaran. Jadi, pengguna harus menambahkan waktu awal hingga waktu pendadaran berakhir. Pada halaman antarmuka jam pendadaran terdapat empat *label*, yaitu: *label* jam pendadaran, *label* kode jam, *label* jam awal dan *label* jam akhir, tiga *textbox*, yaitu: *textbox* kode jam, *textbox* jam awal, *textbox* jam akhir, satu *datagridview* data jam pendadaran dan lima *button*, yaitu: *button* tambah, *button* edit, *button* hapus dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.30.

Penjadwalan Pendadaran / Ujian Tugas Akhir

### Data Waktu

Kode Jam

Jam Awal

Jam Akhir

No	Kode Jam	Jam Awal	Jam Akhir

**Gambar 3. 30** Halaman Waktu Pendadaran

### 3.5.10 Tampilan Halaman Ketersediaan Ruang

Rancangan antarmuka halaman data ketersediaan ruang pendadaran ini akan muncul apabila pengguna memilih *button* ketersediaan ruang pada halaman menu proses. Halaman ini merupakan halaman untuk pengisian data mengenai ketersediaan ruang sesuai tanggal pendadaran dan jam pendadaran yang ada. Pada halaman antarmuka jam pendadaran terdapat empat *label*, yaitu: *label* ketersediaan ruang, *label* ruang, *label* tanggal pendadaran dan *label* jam pendadaran, tiga *combobox*, yaitu: *combobox* ruang, *combobox* tanggal pendadaran, *combobox* jam pendadaran, satu *datagridview* data ketersediaan ruang dan lima *button*, yaitu: *button* tambah, *button* edit, *button* hapus dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.31.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Ketersediaan Ruang

Kode Ruang

Ruang

Kode Tanggal

Tanggal

Kode Jam

Tanggal

No	Ruangan	Tanggal	Jam Pendadaran

**Gambar 3. 31** Halaman Ketersediaan Ruang

### 3.5.11 Tampilan Halaman Ruang

Rancangan antarmuka halaman data ruang pendadaran ini akan muncul apabila pengguna memilih *button* ruang pendadaran pada halaman menu proses. Halaman ini merupakan halaman untuk pengisian data mengenai ruang pelaksanaan pendadaran. Pada halaman antarmuka jam pendadaran terdapat tiga *label*, yaitu: *label* ruang pendadaran, *label* kode ruang dan *label* nama ruang, dua *textbox*, yaitu: *textbox* kode ruang dan *textbox* nama ruang, satu *datagridview* data jam pendadaran dan lima *button*, yaitu: *button* tambah, *button* edit, *button* hapus dan *button* kembali. Gambaran rancangan antarmuka halaman menu utama dapat dilihat pada gambar 3.32.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Data Ruang

Kode Ruang

Ruang

No	Kode Ruang	Ruang

**Gambar 3. 32** Halaman Ruang Pendadaran

### 3.5.12 Tampilan Halaman Tanggal Pendadaran

Rancangan antarmuka halaman tanggal pendadaran ini akan muncul apabila pengguna memilih *button* tanggal pendadaran pada halaman menu proses. Halaman ini merupakan halaman untuk pengisian data mengenai tanggal pelaksanaan pendadaran. Pada halaman antarmuka jam pendadaran terdapat tiga *label*, yaitu: *label* tanggal pendadaran, *label* kode tanggal dan *label* tanggal, satu *datetimepicker* tanggal, satu *datagridview* data jam pendadaran dan lima *button*, yaitu: *button* tambah, *button* edit, *button* hapus dan *button* kembali. Gambaran rancangan antarmuka halaman tanggal pendadaran dapat dilihat pada gambar 3.33.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Data Tanggal

Tanggal

No	Tanggal

**Gambar 3. 33** Halaman Tanggal Pendadaran

### 3.5.13 Tampilan Halaman Jadwal Pendadaran

Rancangan antarmuka halaman jadwal pendadaran ini akan muncul apabila pengguna memilih *button* jadwal pendadaran pada halaman menu. Halaman ini merupakan halaman untuk melihat jadwal pendadaran yang telah ditentukan oleh sistem sesuai data yang ada. Pada halaman antarmuka jadwal pendadaran terdapat satu *label* jadwal pendadaran, satu *datagridview* data jam pendadaran dan dua *button*, yaitu: *button* cetak dan *button* kembali. Gambaran rancangan antarmuka halaman jadwal pendadaran dapat dilihat pada gambar 3.34.

Penjadwalan Pendadaran / Ujian Tugas Akhir

## Jadwal Pendadaran

No	NIM	NmMhs	Dospem 1	Dospem 2	Penguji	Tgl Ujian	Jam Ujian	Ruangan

**Gambar 3. 34** Halaman Jadwal Pendadaran