

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Waktu dan Tempat**

Penulis melakukan penelitian bertempat di Fakultas Teknik Universitas Muhammadiyah Yogyakarta. Adapun waktu penelitian dilaksanakan pada semester genap tahun ajaran 2017/2018.

#### **3.2 Peralatan Pendukung**

Penelitian ini membutuhkan peralatan pendukung dalam pembuatan sistem penjadwalan berbasis *website* ini terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*).

##### **3.2.1 Perangkat Keras**

Perangkat keras yang dibutuhkan dalam membangun sistem penjadwalan sidang tugas akhir berbasis *web* antara lain :

1. Laptop HP Pavilion 14-G008AU
2. Processor *AMD Quad-Core A8-6410* 2.4GHz
3. RAM 4 GB

##### **3.2.2 Perangkat Lunak**

Perangkat lunak yang dibutuhkan dalam membangun sistem penjadwalan sidang tugas akhir berbasis *web* adalah :

1. Netbeans IDE 8.2
2. XAMPP
3. Database Server : *phpMyAdmin*
4. Sistem operasi *windows* 10

### **3.3 Metode Pengumpulan Data**

#### **3.3.1 Studi Literatur**

Studi literatur yang dilakukan oleh peneliti dengan mempelajari bahan-bahan tentang pembuatan sistem penjadwalan sidang tugas akhir atau skripsi yang didapat dari berbagai sumber, baik berupa jurnal, buku-buku referensi, *blog* dan *internet*. Sehingga mendapatkan bahan-bahan yang telah dipelajari oleh peneliti untuk di implementasikan ke pembuatan sistem informasi penjadwalan sidang tugas akhir dengan baik. Selain itu juga peneliti mempelajari berbagai teknologi dengan kebutuhan sistem dan pengguna yang menjadi referensi pembangunan sistem yang lebih baik.

#### **3.3.2 Observasi**

Metode pengumpulan data yang dilakukan dalam penelitian ini dengan cara mengamati lingkungan sumber data secara langsung. Observasi bertujuan untuk mengumpulkan data dengan cara mengamati prosesnya dan permasalahan yang terjadi secara langsung. Observasi dilakukan di lingkungan Fakultas Teknik Universitas Muhammadiyah Yogyakarta.

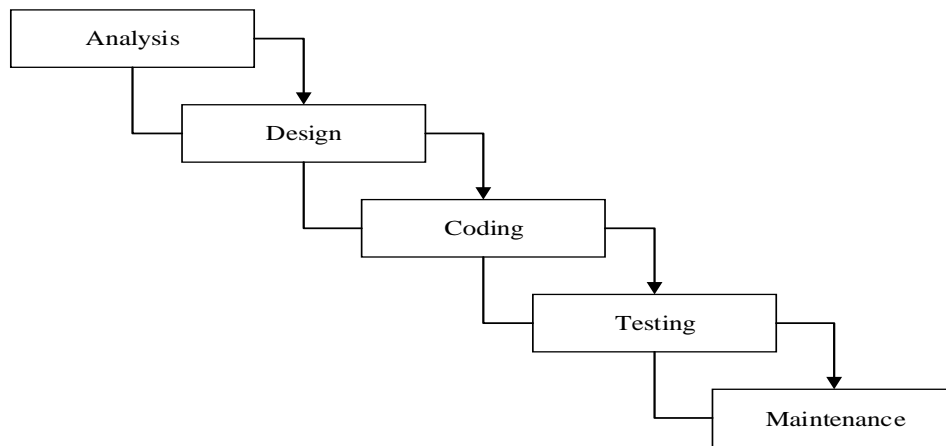
#### **3.3.3 Wawancara**

Pengumpulan data dengan wawancara merupakan teknik pengumpulan data dengan mengajukan sejumlah pertanyaan secara lisan. Wawancara dilakukan dengan cara mewawancarai langsung terhadap pihak terkait yang berguna untuk mendapatkan informasi maupun data – data mengenai penjadwalan sidang pendadaran yang dibutuhkan dan yang akan ditampilkan di *website* yang akan dibuat.

### **3.4 Metode Pengembangan Sistem**

Dalam metode pengembangan sistem penjadwalan sidang tugas akhir atau skripsi menggunakan metode SDLC (*Software Development Life Cycle*) model *waterfall*. Peneliti menggunakan model *waterfall* karna model tersebut mudah

untuk dipahami dan digunakan dalam pengembangan suatu sistem. Ada beberapa tahapan model *waterfall* seperti pada gambar 3.1



**Gambar 3.1** Metode *Waterfall*

### 3.4.1 *Analysis*

Analisis merupakan tahapan awal yang dilakukan untuk pengembangan *website*. Melakukan pengumpulan informasi terkait rancangan dan data yang dibutuhkan dalam penelitian pembuatan *website* dengan cara observasi dan wawancara terhadap pengguna dan user.

### 3.4.2 *Design*

Proses lanjutan setelah selesainya tahapan analisis maka dilakukan tahap *design*. Tahap *design* ini dilakukan untuk membuat rancangan sistem dan alur jalannya sistem sebelum mengerjakan pembuatan *website*.

### 3.4.3 *Coding*

Setelah tahapan *design* selesai maka di lanjutkan dengan ke tahapan *coding* yang bertujuan untuk membuat fitur-fitur pada *website* berfungsi dengan baik seluruhnya, dalam tahap ini peneliti menggunakan Netbeans IDE 8.2 dengan bahasa pemrograman PHP, untuk *web server* dan *database* menggunakan menggunakan XAMPP *Control Panel* dan *phpMyAdmin*. Sistem ini dikerjakan secara nyata dengan penggunaan komputer dan perangkat lunak.

#### **3.4.4 Testing**

Tahap berikutnya adalah *testing*. Tahap pengujian juga merupakan tahap yang menentukan apakah *software* tersebut berfungsi dengan baik dan berjalan tanpa adanya *error* dan hasilnya harus sesuai dengan kebutuhan pengguna atau belum.

#### **3.4.5 Maintenance**

Proses terakhir adalah tahap pemeliharaan sistem. Proses ini harus memiliki tahap pemeliharaan sistem jika terjadi *error* atau jika ada fitur-fitur yang harus ditambahkan.

### **3.5 Analisis Kebutuhan**

Analisis kebutuhan digunakan untuk menentukan kebutuhan sistem. Hal pertama yang harus dilakukan, yaitu *website* yang dibutuhkan saat dimana memiliki data mengenai ruang untuk digunakan sidang skripsi, waktu berlangsungnya sidang skripsi, tanggal dilaksanakannya sidang skripsi mahasiswa yang akan melaksanakan sidang skripsi, dosen yang terlibat, judul masing - masing mahasiswa dan data operasional dosen sidang skripsi. Operasional dosen dibutuhkan karena setiap dosen memiliki kesibukan masing – masing, sehingga perlu pengisian data yang mengenai operasional dosen supaya admin bisa memasukkan data kapan saja dosen bisa melaksanakan sidang skripsi.

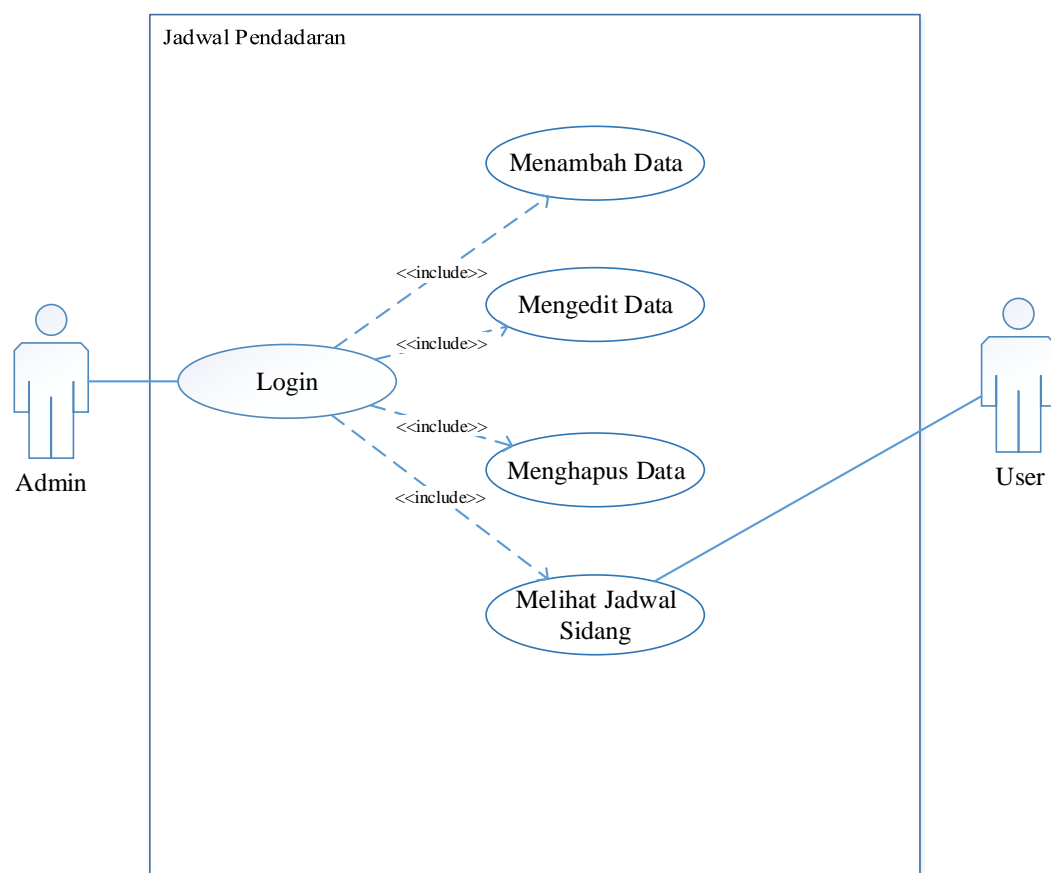
### **3.6 Rancangan Sistem**

Rancangan sistem penjadwalan sidang tugas akhir atau skripsi berbasis web ini peneliti menggunakan diagram *Unified Modelling Language* (UML). Tahapan ini membutuhkan design dari sistem yang akan dibangun dan merancang pemodelan database diperlukan model UML yang terstruktur dengan menggunakan *Use Case Diagram*, *Activity Diagram* dan *Entity Relationship Diagram*.

### 3.6.1 Use Case Diagram

*Use case diagram* merupakan salah satu dari model UML yang sering digunakan untuk menggambarkan sebuah sistem yang dapat berkerja. *Use case* diagram terdiri dari beberapa actor atau satu. *Use case* diagram juga dapat digambarkan sebagai orang yang akan menjalankan atau berinteraksi dengan sistem.

Berikut adalah *Use case* diagram sistem di *web* penjadwalan sidang skripsi.



**Gambar 3.2** Use Case Diagram

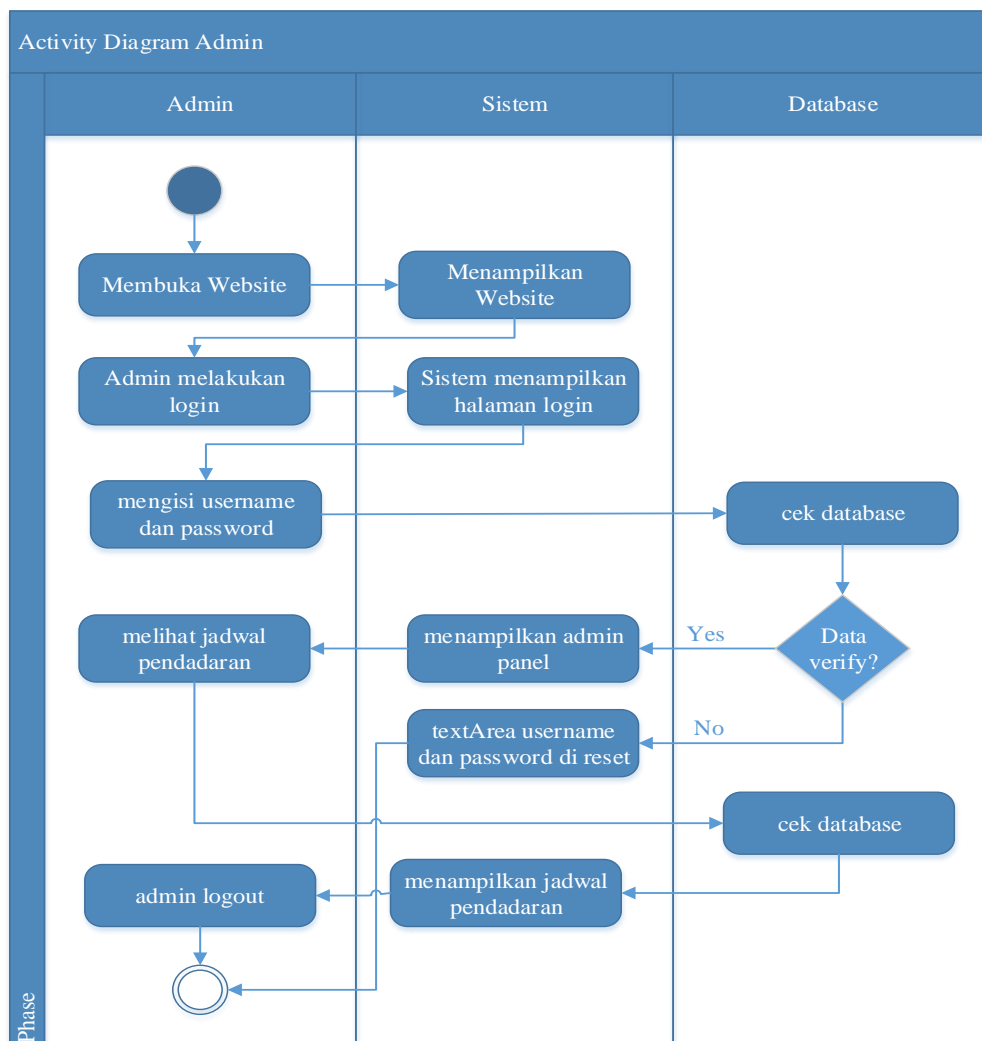
Gambar 3.2 menjelaskan bagaimana antara admin dan user. Ketika membuka web hanya admin yang bisa login dan dapat menambah data, mengedit data, menghapus data, melihat jadwal sidang. User hanya bisa melihat jadwal sidang.

### 3.6.2 Activity Diagram

*Activity diagram* bukan sebuah *tool* sempurna yang bertugas sebagai menganalisis masalah dari sistem melainkan digunakan untuk menganalisa proses. *Activity diagram* menggambarkan jenis alur aktifitas yang sedang dirancang pada perangkat lunak, mulai dari alur berawal, *decision* untuk aktifitas yang memungkinkan terjadi, sampai alur tersebut berakhir. Berikut adalah *activity diagram* yang ada disistem penjadwalan sidang skripsi:

#### 1. Activity Diagram Admin Menampilkan Jadwal Pendaran

Activity diagram admin pada web sistem penjadwalan sidang skripsi dapat dilihat pada gambar 3.3



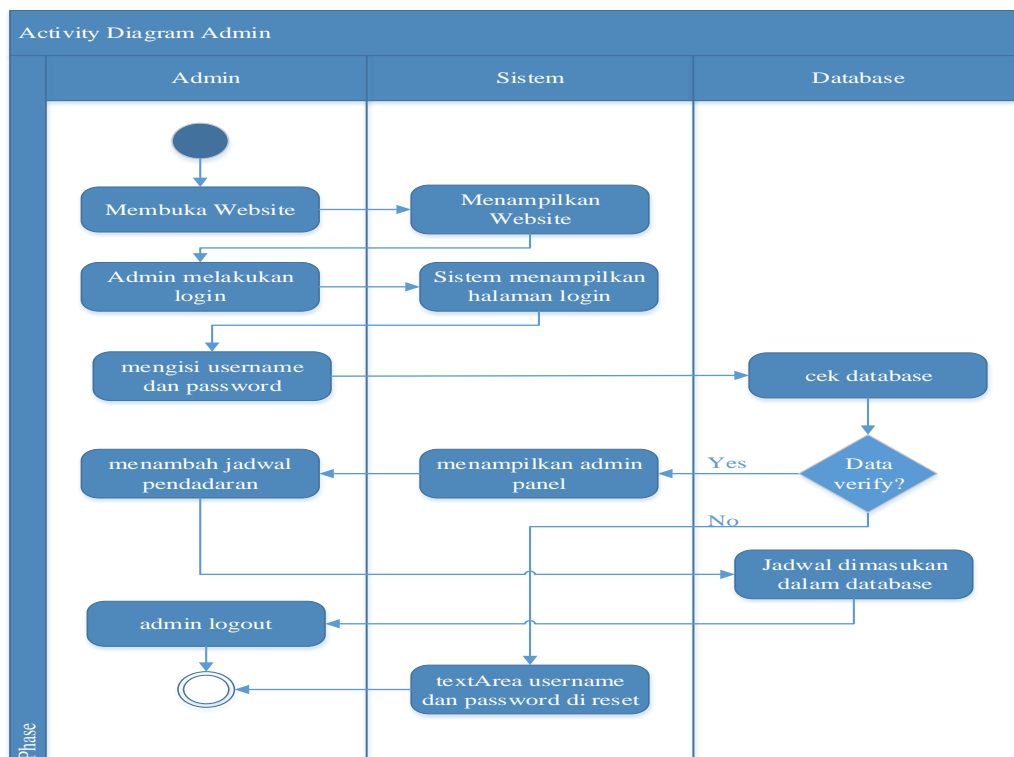
**Gambar 3.3** Activity Diagram Admin Read Data

Keterangan Gambar 3.3 Sebagai berikut:

- a. Admin membuka website
- b. Sistem menampilkan halaman login
- c. Admin mengisi *username* dan *password*
- d. Jika *username* dan *password* ada di database maka sistem menampilkan admin panel.
- e. Jika *username* dan *password* ada di database maka sistem menampilkan akan mereset *username* dan *password*.
- f. Admin melihat jadwal pendaran, database mengumpulkan jadwal pendaran.
- g. Sistem menampilkan jadwal pendaran.
- h. Admin melakukan log out.

## 2. Activity Diagram Admin Menambah Data Jadwal Pendaran

Activity diagram admin pada web sistem penjadwalan sidang skripsi dapat dilihat pada gambar 3.4



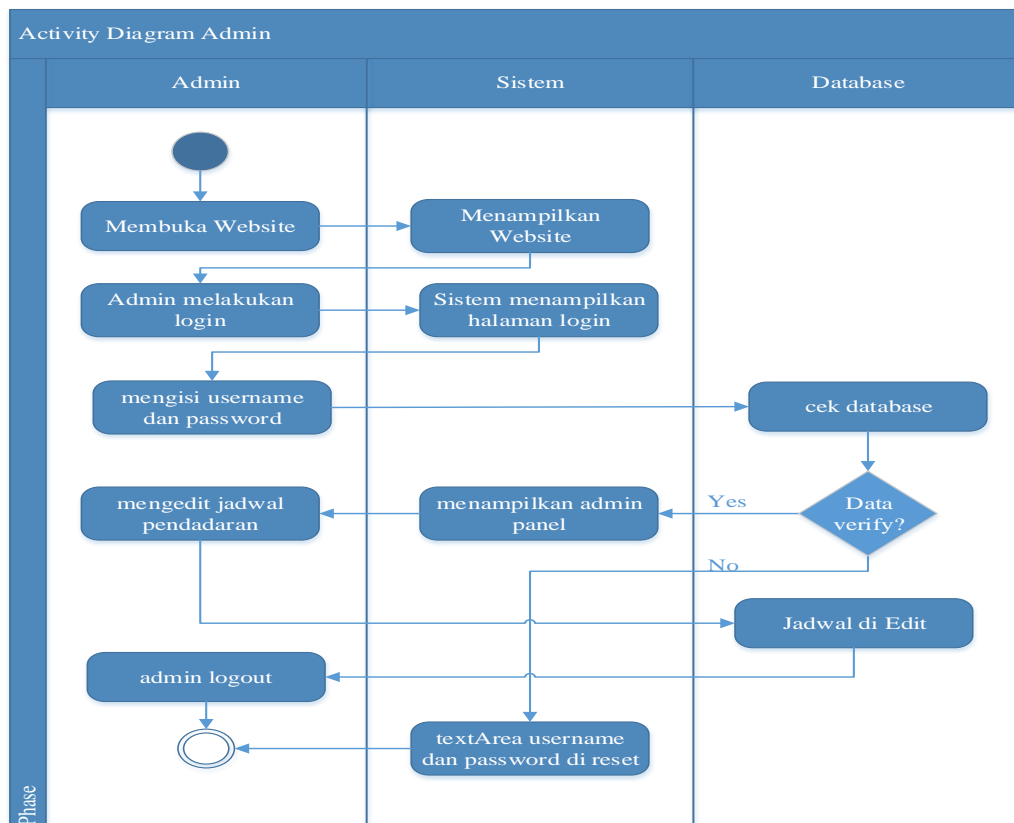
**Gambar 3.4** Activity Diagram Admin Create Data

Keterangan Gambar 3.4 Sebagai berikut:

- a. Admin membuka website
- b. Sistem menampilkan halaman login
- c. Admin mengisi *username* dan *password*
- d. Jika *username* dan *password* ada di database maka sistem menampilkan admin panel.
- e. Jika *username* dan *password* ada di database maka sistem menampilkan akan mereset *username* dan *password*.
- f. Admin menambah data jadwal pendaran, jadwal pendaran baru dimasukkan dalam database.
- g. Admin melakukan log out.

### 3. Activity Diagram Admin Mengedit Data Jadwal Pendaran

Activity diagram admin pada web sistem penjadwalan sidang skripsi dapat dilihat pada gambar 3.5



**Gambar 3.5** Activity Diagram Admin Update Data

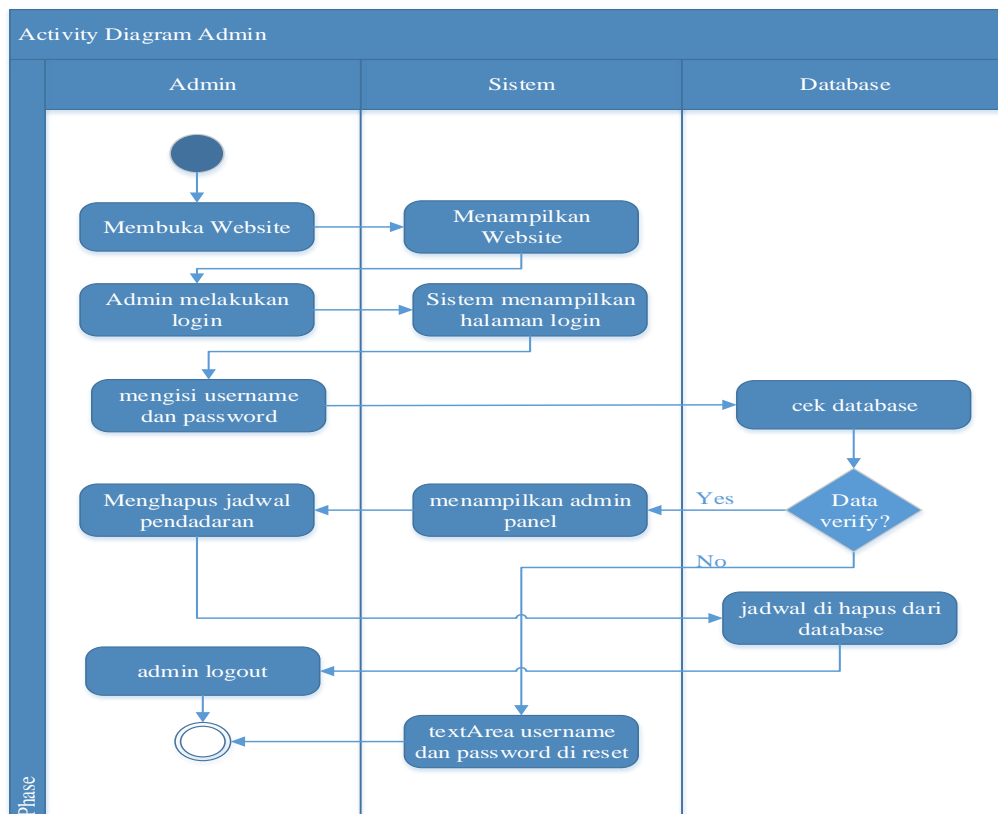


Keterangan Gambar 3.5 Sebagai berikut:

- a. Admin membuka website
- b. Sistem menampilkan halaman login
- c. Admin mengisi *username* dan *password*
- d. Jika *username* dan *password* ada di database maka sistem menampilkan admin panel.
- e. Jika *username* dan *password* ada di database maka sistem menampilkan akan mereset *username* dan *password*.
- f. Admin mengedit data jadwal pendaran, jadwal pendaran di edit dalam database.
- g. Admin melakukan log out.

#### 4. Activity Diagram Admin Menghapus Data Jadwal Pendaran

Activity diagram admin pada web sistem penjadwalan sidang skripsi dapat dilihat pada gambar 3.6



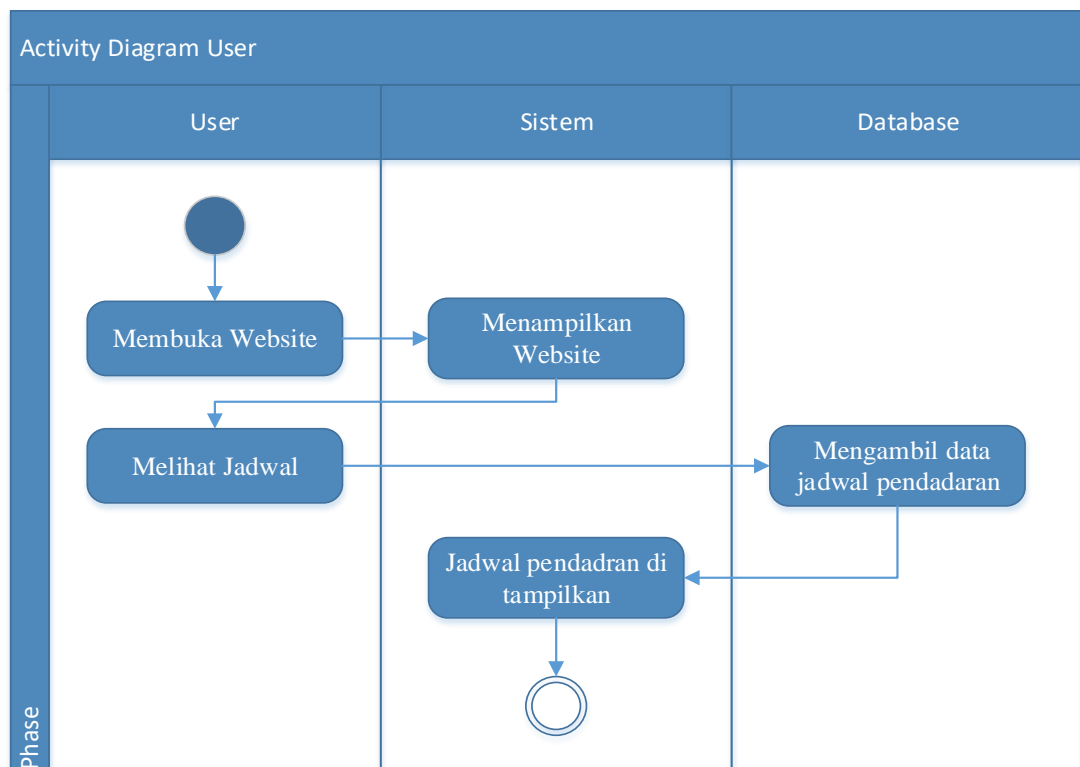
**Gambar 3.6** Activity Diagram Admin Delete Data

Keterangan Gambar 3.6 Sebagai berikut:

- a. Admin membuka website
- b. Sistem menampilkan halaman login
- c. Admin mengisi *username* dan *password*
- d. Jika *username* dan *password* ada di database maka sistem menampilkan admin panel.
- e. Jika *username* dan *password* ada di database maka sistem menampilkan akan mereset username dan password.
- f. Admin menghapus data jadwal pendadaran, jadwal pendadaran dihapus dari database.
- g. Admin melakukan log out.

#### 5. Activity Diagram User

Activity diagram User pada web sistem penjadwalan sidang skripsi dapat dilihat pada gambar 3.7



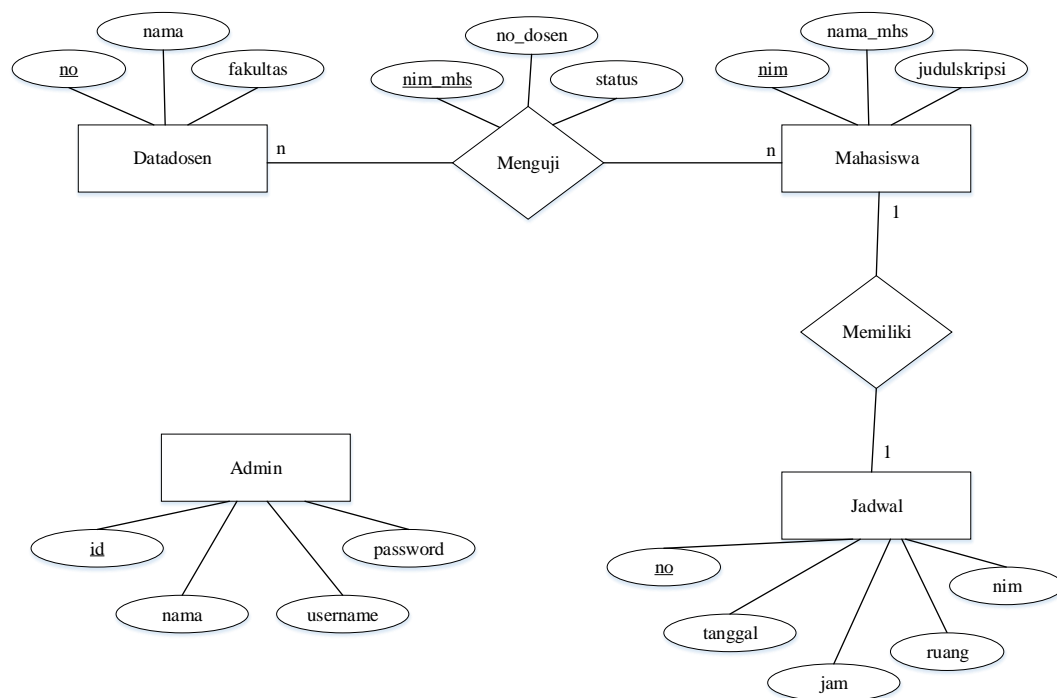
**Gambar 3.7** Activity Diagram User

Keterangan Gambar 3.7 sebagai berikut:

- a. User membuka website
- b. User melihat jadwal, database mengambil data jadwal pendaran.
- c. Sistem menampilkan jadwal pendaran.

### 3.6.3 Entity Relationship Diagram

Entity Relationship Diagram (ER Diagram) merupakan diagram yang digunakan untuk merancang basis data dan memperlihatkan relasi antara entitas atau objek dengan atributnya. Tujuan ER Diagram adalah dapat memberikan gambaran umum tentang sistem yang akan dikembangkan sehingga memudahkan untuk merancang basis data. Gambaran ERD yang digunakan dalam aplikasi bisa dilihat pada gambar 3.8



**Gambar 3.8** Entity Relationship Diagram

Penjelasan entity relationship diagram pada gambar 3.8:

1. Entitas datadosen memiliki relasi *many to many* dengan entitas mahasiswa, sehingga dihasilkan atribut baru yaitu *nim\_mhs*, *no\_dosen*, dan *status* yang merupakan hasil dari kedua entitas tersebut.

2. Entitas mahasiswa memiliki relasi *one to one* dengan entitas jadwal, maksudnya satu mahasiswa hanya dapat memiliki satu jadwal.
3. Entitas admin tidak memiliki relasi dengan entitas apapun.

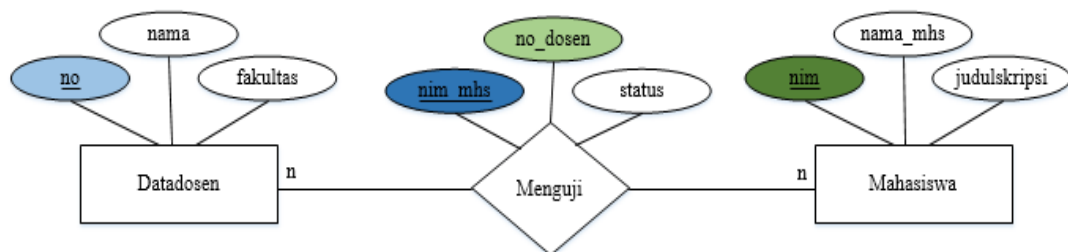
### 3.6.4 Konversi ERD

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

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

#### A. Entitas Datadosen dengan Entitas Mahasiswa

Penjelasan konversi relasi antara entitas datadosen dengan entitas mahasiswa pada gambar 3.9



**Gambar 3.9** Konversi Relasi Entitas Datadosen dengan Entitas Mahasiswa

#### Tabel Datadosen

No	Nama	Fakultas

#### Tabel Dosen\_Mahasiswa

Nim_mhs	No_dosen	Status

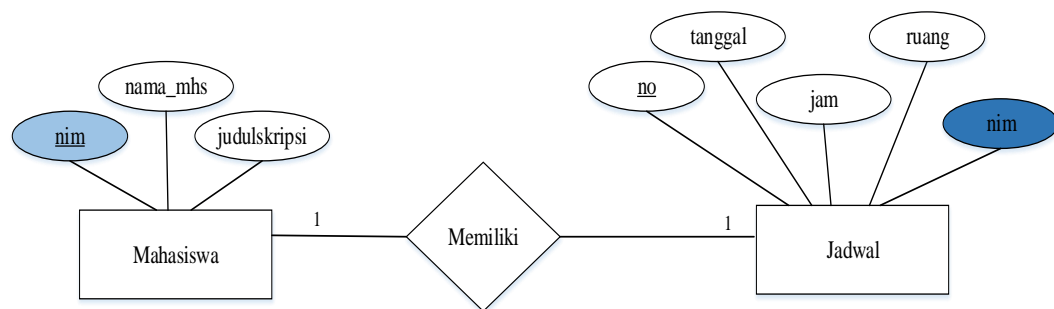
#### Tabel Mahasiswa

Nim	Nama_mhs	Judulskripsi

Entitas datadosen memiliki relasi *many to many* dengan entitas mahasiswa sehingga tercipta tabel baru yang menghubungkan entitas datadosen dengan entitas mahasiswa yang diberi nama tabel dosen\_mahasiswa, maka kolom no pada tabel datadosen sebagai *primary key* perlu disertakan kedalam tabel dosen\_mahasiswa sebagai *foreign key*, dan pada kolom nim pada tabel mahasiswa sebagai *primary key* perlu disertakan ke dalam tabel dosen\_mahasiswa. Relasi antar kedua entitas tersebut *many to many* maka menghasilkan kelas baru yaitu tabel dosen\_mahasiswa.

### B. Entitas Mahasiswa dengan Entitas Jadwal

Penjelasan konversi relasi entitas mahasiswa dengan entitas jadwal pada gambar 3.10



**Gambar 3.10** Konversi Relasi Entitas Mahasiswa dengan Entitas Jadwal

#### Tabel Mahasiswa

Nim	Nama_mhs	Judulskripsi

#### Tabel Jadwal

No	Tanggal	Jam	Ruang	Nim

Entitas mahasiswa memiliki relasi *one to one* dengan entitas jadwal yang artinya satu mahasiswa memiliki satu jadwal, maka kolom nim yang ada pada tabel mahasiswa perlu di tambahkan ke tabel jadwal yang berfungsi sebagai *foreign key* pada tabel jadwal.

### 3.7 Rancangan Basis Data

Dalam membuat aplikasi sistem penjadwalan pendaran atau ujian tugas akhir berbasis web ini menggunakan *phpmyadmin* sebagai database untuk mengaksesnya. Berikut adalah tabel – tabel yang digunakan:

#### 1. Tabel Database Admin

Tabel Admin yang digunakan dalam aplikasi *website* dapat dilihat pada tabel 3.1

**Tabel 3.1** Tabel Admin

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
id	Int	11	PK	Nomor ID
nama	Varchar	30		Nama ID
username	Varchar	30		Nama Admin
password	Varchar	30		Password Admin

#### 2. Tabel Database datadosen

Tabel datadosen yang digunakan dalam aplikasi *website* dapat dilihat pada tabel 3.2

**Tabel 3.2** Tabel Datadosen

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
no	Int	11	PK	Nomor ID
nama	Varchar	30		Nama Dosen
fakultas	Varchar	30		Fakultas Dosen

### 3. Tabel Database dosen\_mhs

Tabel dosen\_mhs yang digunakan dalam aplikasi *website* dapat dilihat pada tabel 3.3.

**Tabel 3.3** Tabel Dosen\_mhs

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
no	Int	11	PK	Nomor ID
no_dosen	Int	11		No Urut Dosen
nim_mhs	Varchar	11		NIM Mahasiswa
status	Varchar	12		Status Dosen

### 4. Tabel Database Jadwal

Tabel Jadwal yang digunakan dalam aplikasi *website* dapat dilihat pada tabel 3.4

**Tabel 3.4** Tabel Jadwal

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
no	Int	11	PK	Nomor ID
tanggal	Varchar	10		Tanggal Pendaftaran
ruang	Varchar	5		Ruangan Pendaftaran
jam	Varchar	5		Waktu Pendaftaran
nim	Varchar	11	FK	NIM Mahasiswa

### 5. Tabel Database mahasiswa

Tabel mahasiswa yang digunakan dalam aplikasi *website* dapat dilihat pada tabel 3.5

**Tabel 3.5** Tabel Mahasiswa

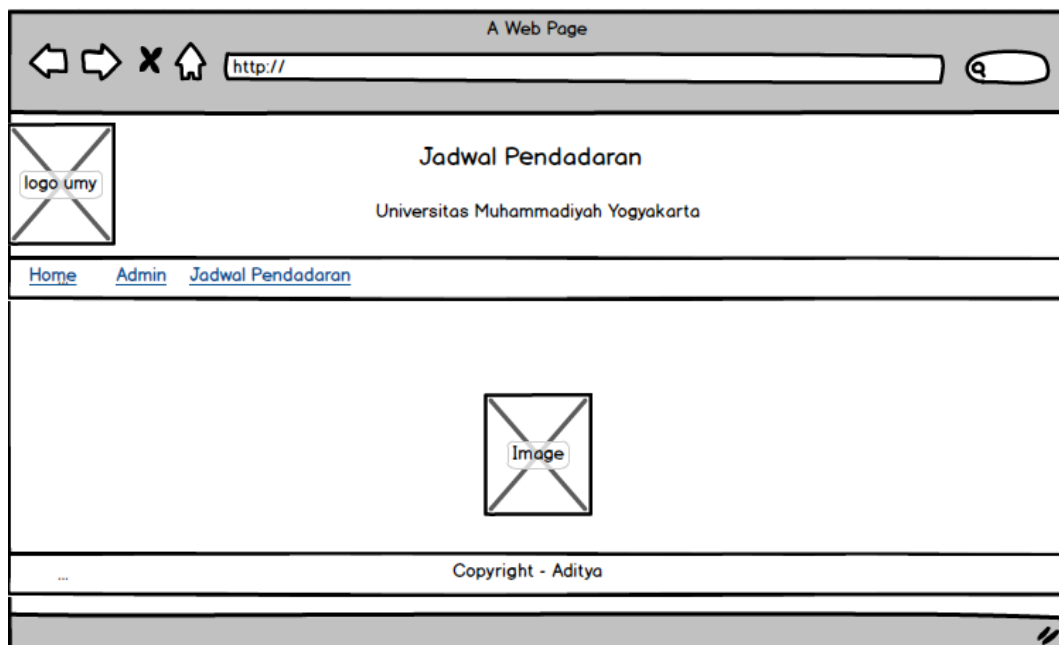
<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang Data</b>	<b>Key</b>	<b>Keterangan</b>
nim	Varchar	11	PK	Nomor ID
nama_mhs	Varchar	50		Nama Mahasiswa
judulskripsi	Varchar	100		Judul Skripsi

### 3.8 Rancangan Antar Muka

Perancangan antar muka merupakan tampilan dari suatu perangkat lunak yang berperan sebagai media perantara sistem dan pengguna. Rancangan antar muka yang menentukan mudah atau sulitnya suatu sistem yang akan dibuat. Berikut adalah rancangan antar muka yang ada pada aplikasi ini.

#### 3.8.1 Tampilan Halaman Utama

Ini merupakan tampilan antarmuka saat website diakses. Tampilan halaman utama ini sebagai langkah awal. Pada rancangan antarmuka terdapat beberapa pilihan menu yang akan dipilih oleh pengguna sesuai dengan kebutuhannya. Menu home hanya menampilkan halaman utama. Menu admin berfungsi untuk mengisi data mahasiswa yang ingin mengajukan sidang pendadaran dan hanya bisa diakses oleh petugas. Sedangkan menu jadwal pendadaran berfungsi untuk menampilkan data jadwal yang akan melakukan sidang pendadaran. Berikut ini merupakan rancangan antarmuka halaman utama dapat dilihat pada gambar 3.11

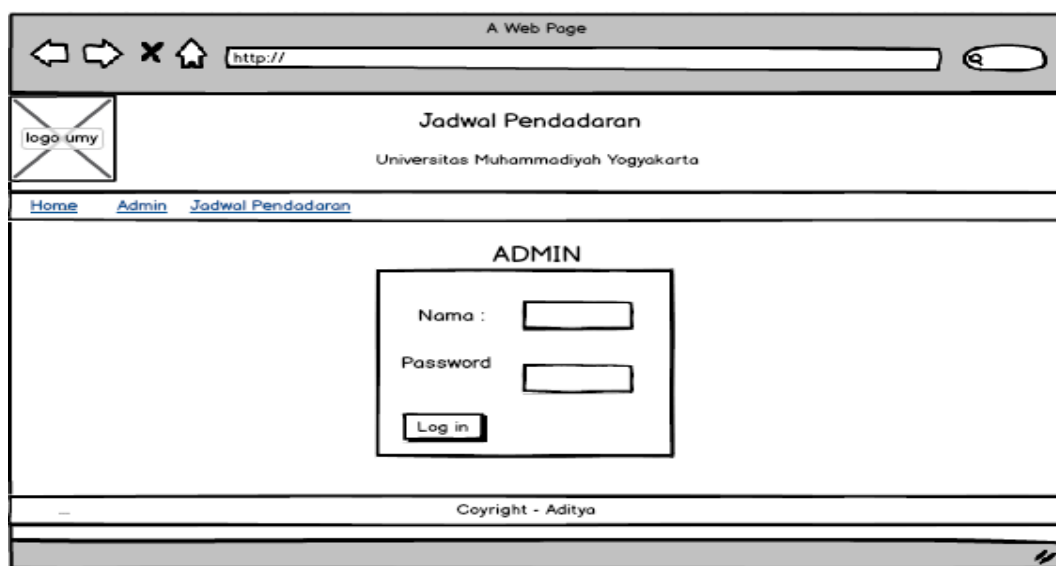


**Gambar 3.11** Halaman Utama



### 3.8.2 Tampilan Halaman Admin

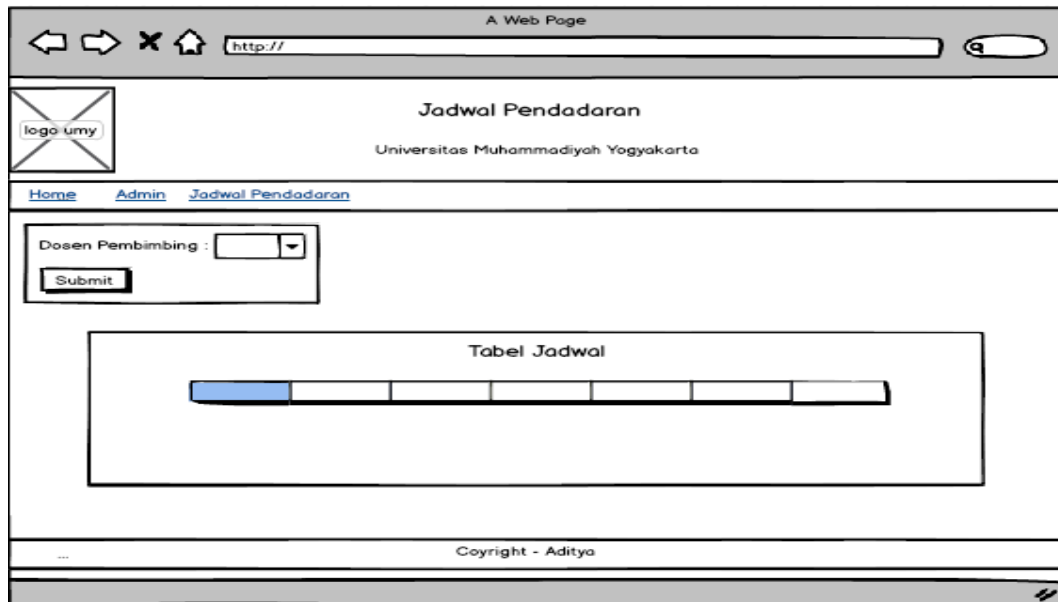
Tampilan halaman masuk ini sebagai admin sehingga harus memasukkan nama petugas dan kata sandi terlebih dahulu. Pada halaman ini memiliki dua kontrol *textbox*, yaitu: *textbox* nama dan *textbox* password. Halaman ini juga memiliki satu *button*, yaitu: *button* Log in, dan tiga *label*, yaitu: *label* admin, *label* nama, dan *label* password. Berikut ini merupakan rancangan antarmuka halaman utama dapat dilihat pada gambar 3.12



**Gambar 3.12** Halaman Admin

### 3.8.3 Tampilan Halaman Jadwal Pendaftaran

Ini merupakan rancangan halaman antarmuka jadwal pendaftaran untuk si user. Pada halaman ini memiliki satu kontrol *combo box*, yaitu: *combo box* dosen pembimbing. Halaman ini terdapat satu *button*, yaitu: *button* submit dan tabel jadwal berfungsi untuk menampilkan jadwal sidang pendaftaran mahasiswa. Berikut ini merupakan rancangan antarmuka halaman utama dapat dilihat pada gambar 3.13



**Gambar 3.13** Halaman Jadwal Pendaftaran

### 3.8.4 Tampilan Pendaftaran Sidang Skripsi

Rancangan antarmuka halaman pendaftaran sidang skripsi ini akan muncul apabila petugas masuk di halaman admin sebelumnya. Halaman ini merupakan halaman untuk pengisian data mahasiswa yang akan melaksanakan sidang skripsi. Admin harus menambahkan data mahasiswa yang mendaftar untuk pendadaran pada halaman ini. Pada halaman ini terdapat sepuluh *label*, yaitu: *label* nama, *label* nim, *label* tanggal, *label* hari, *label* ruangan, *label* jam, *label* dosen pembimbing 1, *label* dosen pembimbing 2, *label* dosen penguji, dan *label* judul skripsi, tiga *textbox*, yaitu: *textbox* nama, *textbox* nim, dan *textbox* judul skripsi, enam *combo box*, yaitu: *combo box* hari, *combo box* ruangan, *combo box* jam, *combo box* dosen pembimbing 1, *combo box* dosen pembimbing 2, dan *combo box* dosen penguji, satu *date chooser*, yaitu: *date chooser* tanggal, dua *button*, yaitu: *button* submit dan *button* reset. Berikut ini merupakan rancangan antarmuka halaman utama dapat dilihat pada gambar 3.14

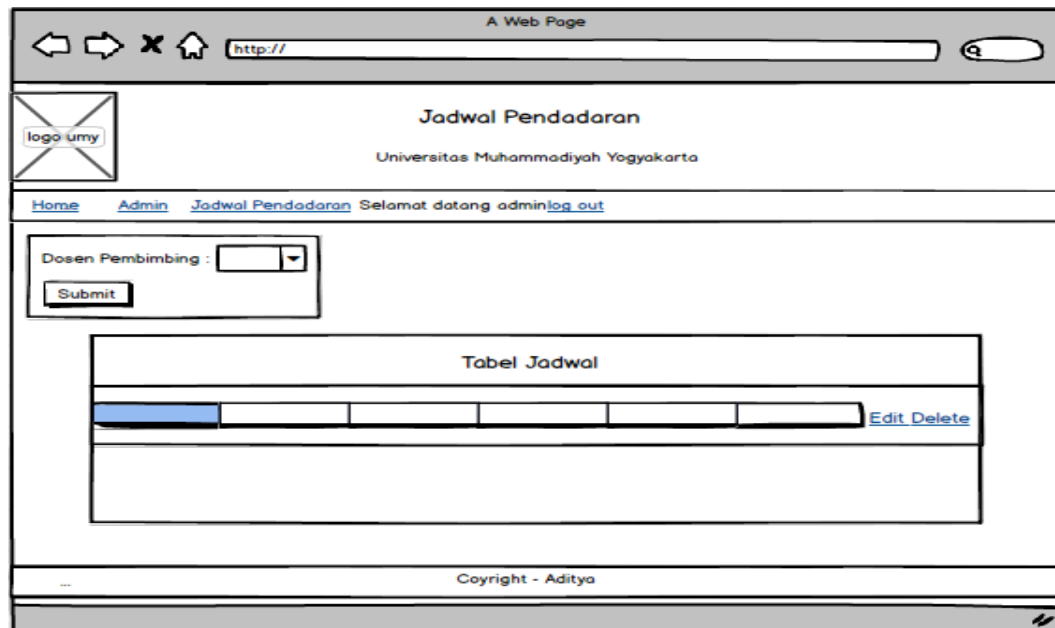
The image shows a web browser window with the following content:

- Browser title: A Web Page
- Address bar: http://
- Page title: Jadwal Pendaran
- Page subtitle: Universitas Muhammadiyah Yogyakarta
- Navigation menu: [Home](#), [Admin](#), [Jadwal Pendaran](#), [Selamat datang admin! admin@umy](#)
- Form title: Pendaftaran Sidang Skripsi
- Form fields:
  - Nama :
  - Nim :
  - Tanggal :
  - Hari :
  - Ruangan :
  - Jam :
  - Dosen Pembimbing 1 :
  - Dosen Pembimbing 2 :
  - Dosen Penguji :
  - Judul Skripsi :
- Buttons: ,
- Page footer: Copyright - Aditya

**Gambar 3.14** Halaman Pendaftaran Sidang Skripsi

### 3.8.5 Tampilan Jadwal Pendaran Admin

Ini merupakan rancangan halaman antarmuka jadwal pendaran admin. Pada halaman ini memiliki satu kontrol *combo box*, yaitu: *combo box* dosen pembimbing. Halaman ini terdapat satu *button*, yaitu: *button* submit dan dua menu *link*, yaitu: *link* edit dan *link* delete. Berikut ini merupakan rancangan antarmuka halaman utama dapat dilihat pada gambar 3.15



**Gambar 3.15** Halaman Jadwal Pendaftaran Admin

### 3.8.6 Tampilan Form Edit Data Mahasiswa

Rancangan antarmuka halaman form edit data mahasiswa ini akan muncul apabila petugas masuk di halaman jadwal pendaftaran didalam admin. Halaman ini merupakan halaman untuk mengedit data mahasiswa jika ada perubahan disetiap datanya. Pada halaman ini terdapat sepuluh *label*, yaitu: *label* nama, *label* nim, *label* tanggal, *label* hari, *label* ruangan, *label* jam, *label* dosen pembimbing 1, *label* dosen pembimbing 2, *label* dosen penguji, dan *label* judul skripsi, tiga *textbox*, yaitu: *textbox* nama, *textbox* nim, dan *textbox* judul skripsi, enam *combo box*, yaitu: *combo box* hari, *combo box* ruangan, *combo box* jam, *combo box* dosen pembimbing 1, *combo box* dosen pembimbing 2, dan *combo box* dosen penguji, satu *date chooser*, yaitu: *date chooser* tanggal, satu *button*, yaitu: *button* submit. Berikut ini merupakan rancangan antarmuka halaman utama dapat dilihat pada gambar 3.16

A Web Page

http://

logo-umy

Jadwal Pendadaran  
Universitas Muhammadiyah Yogyakarta

[Home](#) [Admin](#) [Jadwal Pendadaran](#) Selamat datang admin[log out](#)

Form Edit Data Mahasiswa

Nama :	<input type="text" value="XXXXXXXX"/>
Nim :	<input type="text" value="XXXXXXXX"/>
Tanggal :	<input type="text" value="/ /"/>
Hari :	<input type="text" value=""/>
Ruangan :	<input type="text" value=""/>
Jam :	<input type="text" value=""/>
Dosen Pembimbing 1 :	<input type="text" value=""/>
Dosen Pembimbing 2 :	<input type="text" value=""/>
Dosen Penguji :	<input type="text" value=""/>
Judul Skripsi :	<input type="text" value="XXXXX"/>

Copyright - Aditya

Gambar 3.16 Halaman Form Edit Data Mahasiswa