

BAB III

METODE PENELITIAN

3.1 Tempat dan Waktu Penelitian

Tempat yang digunakan penulis dalam melakukan penelitian ini adalah Prodi Teknik Informatika Universitas Muhammadiyah Yogyakarta. Adapun penelitian ini dilaksanakan mulai bulan April sampai Juli 2017.

3.2 Alur Penelitian

Penelitian sistem pengambilan ide skripsi berbasis *website* menggunakan metode *waterfall*. Model ini dipilih karena menyediakan alur hidup perangkat lunak dengan fase *one by one*, sehingga meminimalisir kesalahan yang mungkin terjadi. Alur penelitian dilakukan dalam beberapa tahap, sebagai berikut:

1. Analisis kebutuhan

Analisis kebutuhan mencakup kebutuhan perangkat keras, perangkat lunak, dan *requirement* isi dan interaksi menu pada aplikasi. Sebelum membuat program aplikasi berbasis *website*, terlebih dahulu dilakukan pengumpulan kebutuhan dengan membagikan kuesioner terhadap beberapa calon pengguna. Kuesioner yang terkumpul akan menghasilkan data terkait bagaimana fitur-fitur *website* yang diharapkan oleh pengguna.

2. Desain sistem

Sebelum aplikasi dibuat, penulis merancang desain aplikasi *website*. Pembuatan rancangan ini ditujukan agar *website* yang dibuat sesuai dengan kebutuhan yang telah ditentukan.

3. Implementasi

Perancangan yang telah dibuat sebelumnya diterjemahkan dalam *coding* yang membentuk sebuah aplikasi berbasis *website*. Pembuatan aplikasi *website* pun dilakukan dengan menggunakan peralatan yang telah ditentukan dari analisis kebutuhan.

4. Pengujian

Sistem akan diuji sesuai dengan *requirement* yang telah ditentukan sebelumnya. Pengujian aplikasi berbasis *website* dilakukan dengan menggunakan metode *black box testing*.

5. Pemeliharaan

Pemeliharaan *software* termasuk didalamnya adalah pengembangan. Ketika *software* dijalankan mungkin masih ada *error* atau fitur yang perlu ditambahkan. Pengembangan pun perlu dilakukan untuk memperbaiki *software* menjadi lebih baik.

3.3 Analisis Kebutuhan

Dalam penelitian ini, dilakukan analisis kebutuhan yang mencakup kebutuhan perangkat keras, perangkat lunak, dan *requirement* sistem.

3.3.1 Perangkat Keras

Adapun perangkat keras yang digunakan dalam perancangan dan pembuatan aplikasi berbasis *website* ini, yaitu:

1. *Processor* intel core i5.
2. RAM 4 GB.
3. *Harddisk* 500 GB.
4. Monitor, *keyboard*, dan *mouse*.

3.3.2 Perangkat Lunak

Adapun perangkat lunak yang digunakan dalam perancangan dan pembuatan aplikasi berbasis *website* ini, yaitu:

1. Sistem operasi windows 8.1.
2. *Web server*: XAMPP.
3. *Web browser*: Mozilla firefox.
4. *Text editor*: Sublime Text.
5. Bahasa pemrograman: PHP, Javascript, HTML, CSS.
6. *Database server*: MySQL.

3.3.3 Requirement Sistem

Kebutuhan fungsional sistem adalah fungsi-fungsi yang harus ada dalam sistem. Fungsi-fungsi tersebut, meliputi:

1. Fasilitas *login* bagi mahasiswa, dosen, dan administrator.
2. Mahasiswa dapat mengubah profil, mengambil ide skripsi, dan melihat pengumuman.
3. Fitur pencarian dan sorting untuk mempermudah pencarian data.
4. Dosen dapat mengubah profil, menambah, mengubah, menghapus ide skripsi, dan menyeleksi mahasiswa yang telah mengambil ide skripsi.
5. Administrator dapat mengelola akun mahasiswa dan dosen, meliputi *reset password*, menghapus akun, dan aktivasi akun pengguna.

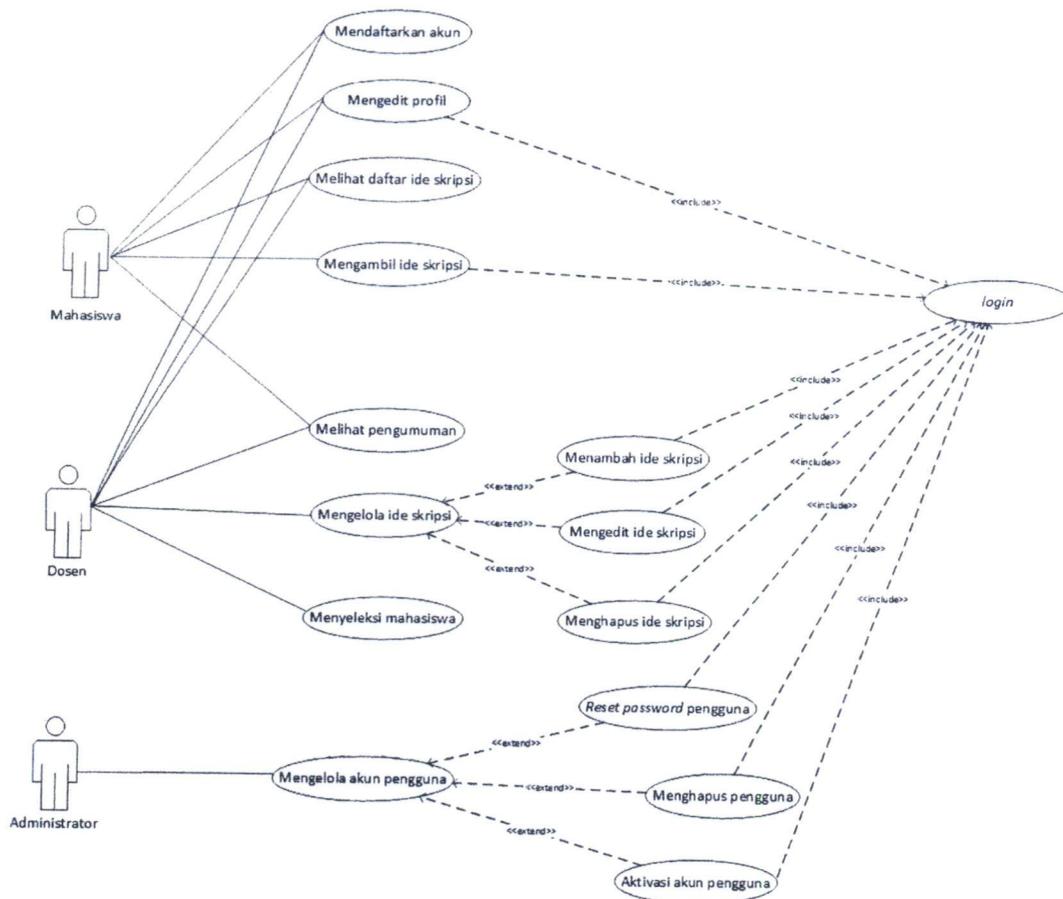
3.5 Desain Sistem

Metode perancangan yang digunakan dalam *website* ini adalah *Unified Model Language* (UML). Model UML yang digunakan antara lain, *use case diagram*, *activity diagram*, dan *class diagram*.

3.5.1 Use Case Diagram

Penjelasan tentang Gambar 3.1 adalah sebagai berikut:

1. Terdapat 3 aktor dalam *use case diagram website*, yaitu mahasiswa, dosen, dan administrator.
2. Mahasiswa dapat melakukan pendaftaran akun, *login*, mengedit profil, melihat daftar ide skripsi, mengambil ide skripsi, dan melihat pengumuman.
3. Dosen dapat melakukan registrasi akun, *login*, mengedit profil, melihat data ide skripsi, melihat pengumuman, mengelola ide skripsi, dan menyeleksi mahasiswa.
4. Administrator dapat melakukan *login* dan mengelola akun pengguna meliputi *reset password*, menghapus akun, dan aktivasi akun pengguna.



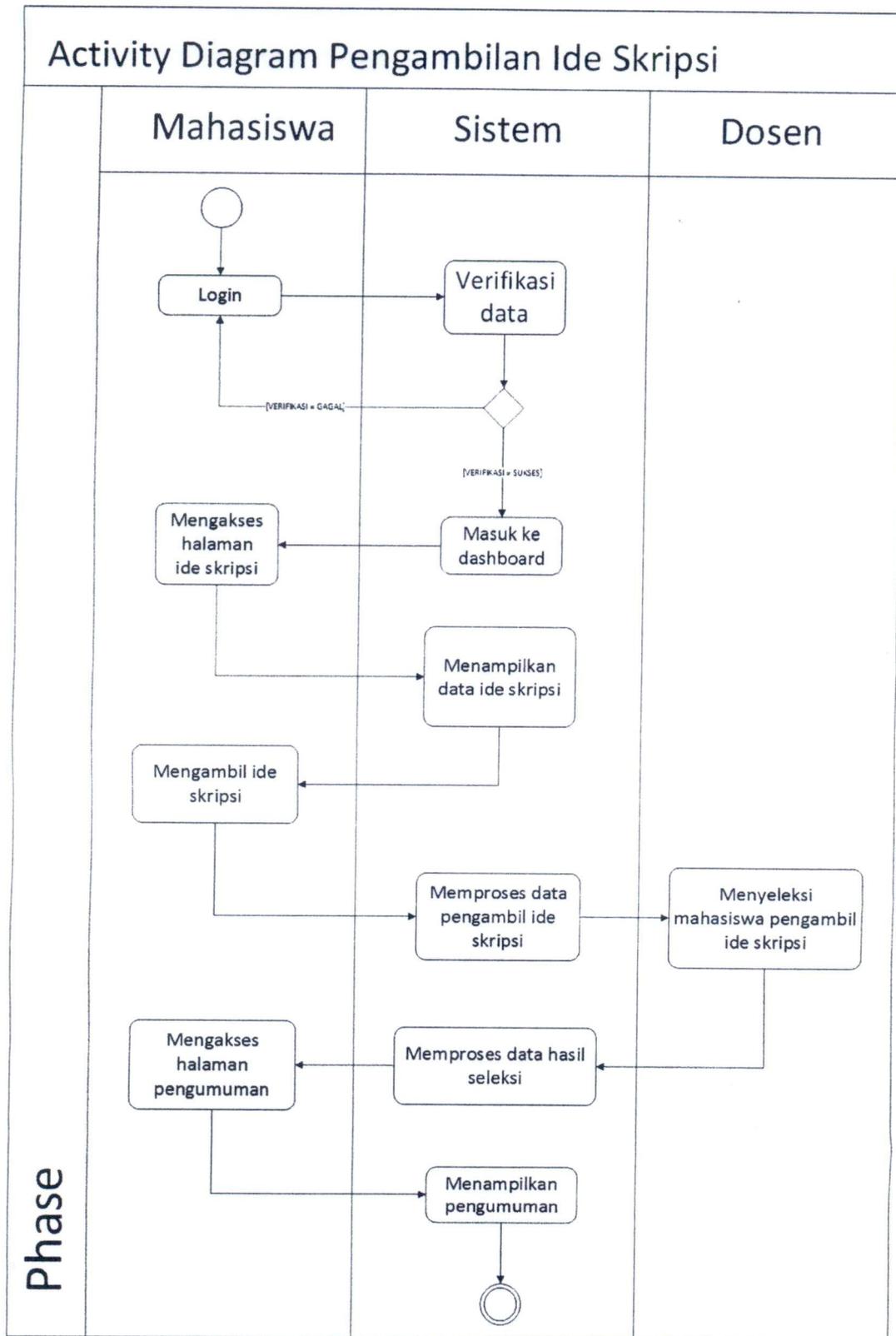
Gambar 3.1 Use case diagram website

3.5.2 Activity Diagram

Berdasarkan *use case diagram* yang dibuat sebelumnya, maka dapat diperoleh *activity diagram* berdasarkan aktor yang terlibat dalam *use case diagram*. *Activity diagram* untuk *website* ini dibagi menjadi tiga yaitu *activity diagram* pengambilan ide skripsi, *activity diagram* pengelolaan ide skripsi, dan *activity diagram* pengelolaan akun.

A. Activity Diagram Pengambilan Ide Skripsi

Gambaran *activity diagram* pengambilan ide skripsi yang digunakan dalam *website* Sistem Pengambilan Ide Skripsi dapat dilihat pada Gambar 3.2.



Gambar 3. 2 Activity diagram pengambilan ide skripsi

Penjelasan tentang gambar 3.2 adalah sebagai berikut.

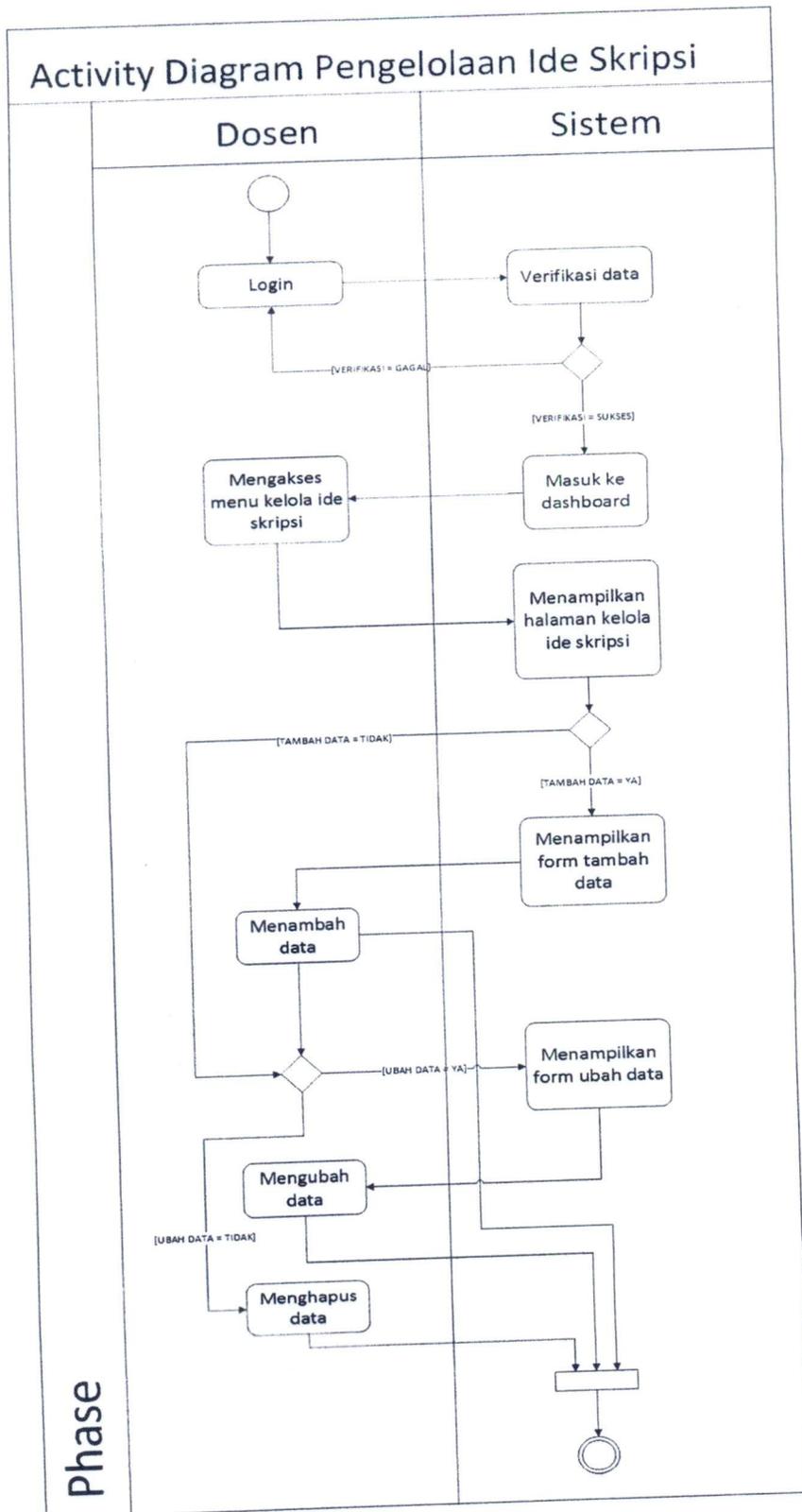
1. Alur dari kegiatan pengambilan ide skripsi yang dilakukan oleh mahasiswa, yaitu masuk ke halaman *login* terlebih dahulu. Apabila *login* sudah terverifikasi benar, sistem akan mengarahkan ke halaman *dashboard*. Namun bila verifikasi *login* gagal, pengguna akan diarahkan kembali ke halamana *login*.
2. Mahasiswa mengakses halaman ide skripsi untuk melihat daftar ide skripsi yang ada. Mahasiswa dapat mengambil salah satu ide skripsi, kemudian data mahasiswa yang mengambil ide skripsi akan diproses oleh sistem.
3. Setelah data pengambil ide skripsi diproses oleh sistem, dosen akan menyeleksi mahasiswa pengambil ide skripsi. Hasil seleksi mahasiswa akan diproses oleh sistem.
4. Mahasiswa mengakses halaman pengumuman untuk mengetahui hasil seleksi.

B. *Activity Diagram* Pengelolaan Ide Skripsi

Gambaran *activity diagram* pengambilan ide skripsi yang digunakan dalam *website* dapat dilihat pada Gambar 3.3.

Penjelasan tentang Gambar 3.3 adalah sebagai berikut.

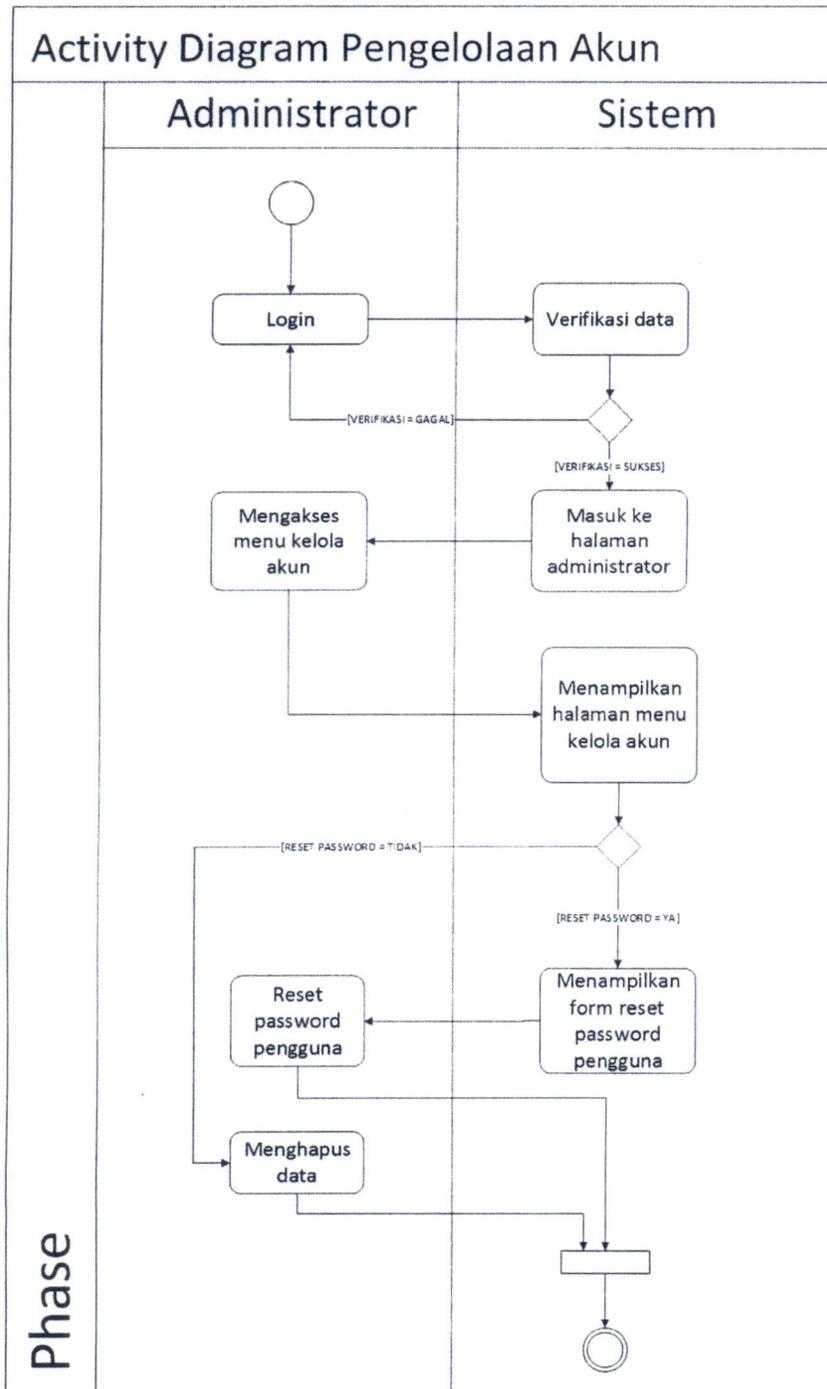
1. Alur dari kegiatan pengelolaan ide skripsi yang dilakukan oleh dosen, yaitu masuk ke halaman *login* terlebih dahulu. Apabila *login* sudah terverifikasi benar, sistem akan mengarahkan ke halaman *dashboard*. Namun bila verifikasi *login* gagal, pengguna akan diarahkan kembali ke halamana *login*.
2. Dosen mengakses halaman kelola ide skripsi untuk mengelola ide skripsi.
3. Setelah sistem menampilkan halaman ide skripsi, Dosen dapat memilih menambah data ide skripsi dengan mengisi *form* tambah data yang ditampilkan oleh sistem.
4. Dosen dapat memilih mengubah data ide skripsi dengan mengisi *form* ubah data yang ditampilkan oleh sistem.
5. Dosen dapat memilih untuk menghapus data ide skripsi.



Gambar 3.3 Activity diagram pengelolaan ide skripsi

C. Activity Diagram Pengelolaan Akun

Gambaran *activity diagram* pengelolaan akun yang digunakan dalam *website* dapat dilihat pada Gambar 3.4.



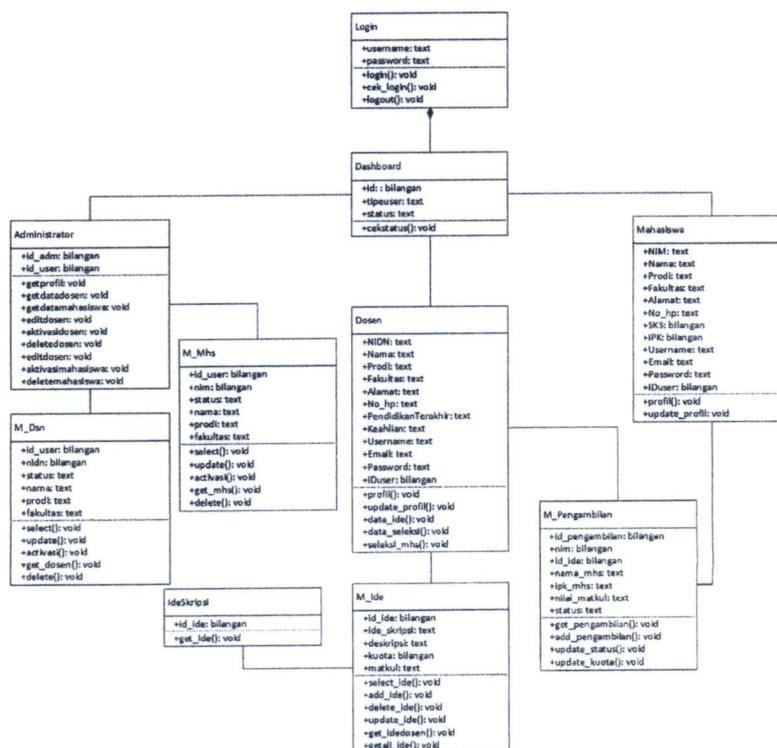
Gambar 3. 4 Activity diagram pengelolaan akun

Penjelasan tentang Gambar 3.4 adalah sebagai berikut.

1. Alur dari kegiatan pengelolaan akun pengguna yang dilakukan oleh administrator, yaitu masuk ke halaman *login* terlebih dahulu. Apabila *login* sudah terverifikasi benar, sistem akan mengarahkan ke halaman administrator. Namun bila verifikasi *login* gagal, administrator akan diarahkan kembali ke halaman *login*.
2. Administrator mengkases menu kelola akun untuk mengelola akun pengguna.
3. Setelah sistem menampilkan halaman menu kelola akun, Administrator dapat memilih *reset password* pengguna dengan mengisi *form reset password* yang ditampilkan oleh sistem.
4. Administrator dapat memilih untuk menghapus akun pengguna.

3.5.3 Class Diagram

Gambaran *Class diagram* yang digunakan dalam *website* dapat dilihat pada Gambar 3.5.



Gambar 3.5 *Class diagram website*

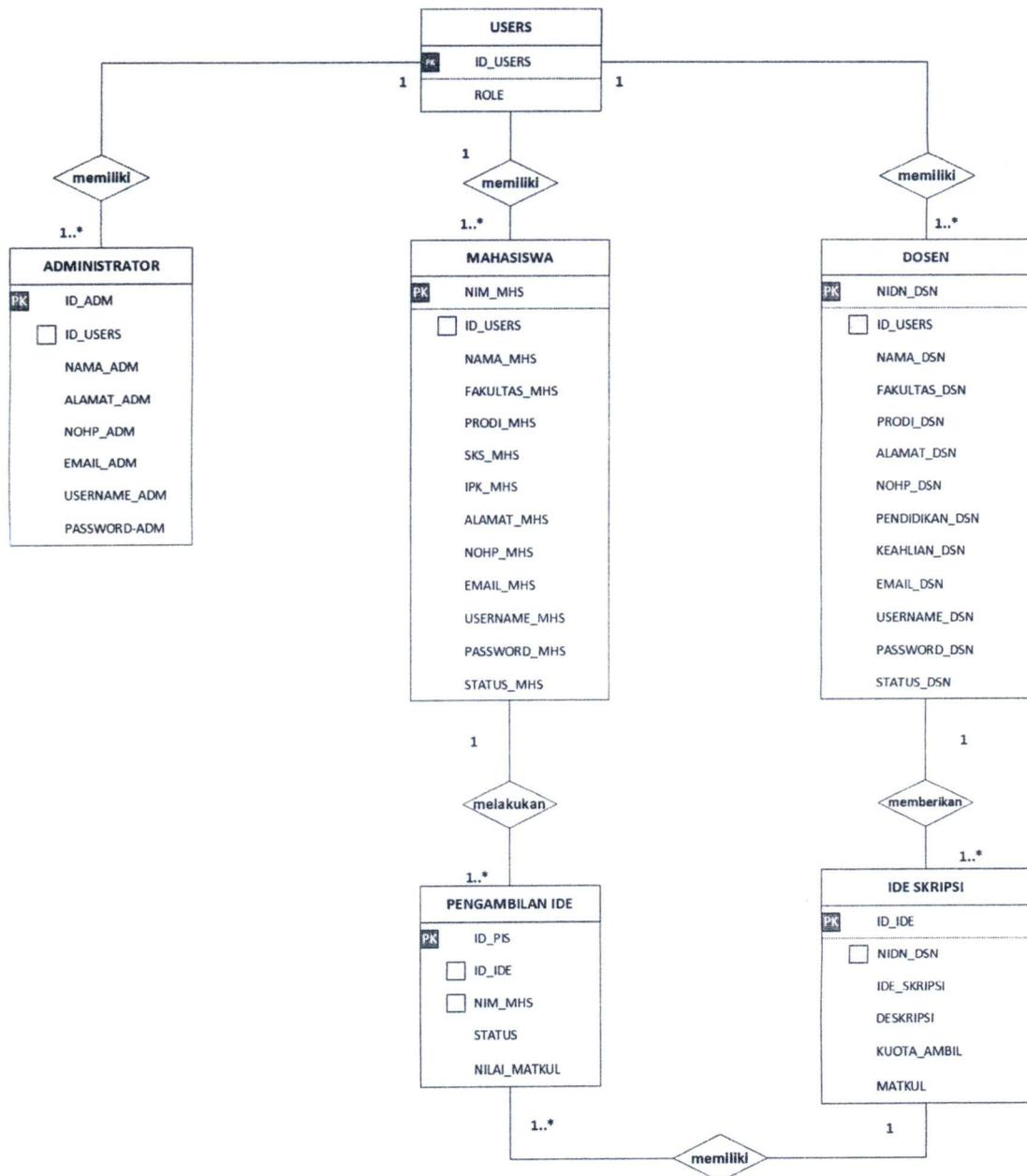
Penjelasan tentang Gambar 3.5 adalah sebagai berikut:

1. Kelas Dashboard memiliki hubungan *composition* dengan kelas Login, artinya kelas Dashboard merupakan bagian dari kelas Login. Kelas Dashboard tidak dapat berdiri sendiri apabila kelas Login tidak ada.
2. Kelas Dashboard memiliki fungsi untuk memeriksa tipe pengguna yang *login* dan status akun apakah aktif atau non aktif.
3. Kelas Administrator memiliki fungsi aktivasi akun, *reset password* dan *delete* akun mahasiswa ataupun dosen.
4. Kelas Administrator memiliki relasi *association* dengan kelas M_Dsn, dimana Administrator dapat mengakses data dosen melalui *method* `get_dosen()`.
5. Kelas Administrator memiliki relasi *association* dengan kelas M_Mhs, dimana Administrator dapat mengakses data mahasiswa melalui *method* `get_mhs()`.
6. Kelas Dosen memiliki fungsi untuk mengedit profil dosen, mengelola ide skripsi, dan menyeleksi mahasiswa.
7. Kelas Dosen memiliki relasi *association* dengan kelas M_Ide, dimana Dosen dapat menambah ide skripsi melalui *method* `add_ide()`, mengedit ide skripsi melalui *method* `update_ide()`, dan menghapus ide skripsi melalui *method* `delete_ide()`.
8. Kelas IdeSkripsi memiliki fungsi untuk menampilkan data ide skripsi.
9. Kelas IdeSkripsi memiliki relasi *association* dengan kelas M_Ide, dimana IdeSkripsi mengambil data ide skripsi berdasarkan `id_ide` melalui *method* `select_ide()`.
10. Kelas Mahasiswa memiliki fungsi untuk mengedit profil mahasiswa.
11. Kelas Mahasiswa memiliki relasi *association* dengan kelas M_Pengambilan, dimana Mahasiswa dapat menambah pengambilan ide skripsi melalui *method* `add_pengambilan()`.
12. Kelas Dosen memiliki memiliki relasi *association* dengan kelas M_Pengambilan, dimana Dosen dapat mengambil data pengambil ide

skripsi melalui method `get_pengambilan` dan mengubah status melalui `method update_status()`.

3.5.4 ER Diagram

Dalam sistem Ide Skripsi yang dibuat terdapat beberapa entitas yang membentuk sebuah *ER diagram*. *ER diagram* dapat dilihat pada Gambar 3.6.



Gambar 3. 6 ER Diagram sistem pengambilan ide skripsi

Penjelasan ERD pada Gambar 3.6 adalah sebagai berikut:

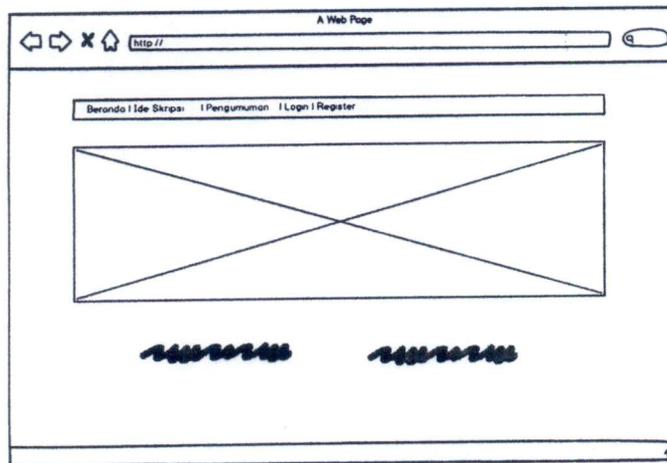
1. Entitas *Users* mempunyai relasi *one to many* dengan entitas Administrator, artinya satu peran (*role*) dapat memiliki banyak *users* sesuai dengan peran dalam entitas Administrator yaitu sebagai admin.
2. Entitas *Users* mempunyai relasi *one to many* dengan entitas Mahasiswa, artinya satu peran (*role*) dapat memiliki banyak *users* sesuai dengan peran dalam entitas Mahasiswa yaitu sebagai mahasiswa.
3. Entitas *Users* mempunyai relasi *one to many* dengan entitas Dosen, artinya satu peran (*role*) dapat memiliki banyak *users* sesuai dengan peran dalam entitas Dosen yaitu sebagai dosen.
4. Entitas Dosen mempunyai relasi *one to many* dengan entitas Ide Skripsi, artinya satu Dosen dapat memberikan ide lebih dari satu ide skripsi dalam entitas Ide Skripsi.
5. Entitas Ide Skripsi mempunyai relasi *one to many* dengan entitas Pengambilan Ide, artinya satu Ide Skripsi dapat diambil oleh lebih dari satu mahasiswa dalam entitas Pengambilan Ide.
6. Entitas Mahasiswa mempunyai relasi *one to many* dengan entitas Pengambilan Ide, artinya satu mahasiswa dapat mengambil lebih dari satu ide skripsi dengan persyaratan mahasiswa tersebut telah dinyatakan belum diterima oleh dosen untuk mengerjakan ide skripsi, setelah dinyatakan belum diterima mahasiswa baru dapat mengambil ide skripsi kembali.

3.5.5 Rancangan Antarmuka (*User Interface*)

User interface merupakan tampilan grafis yang berfungsi menjadi sarana komunikasi antar pengguna dengan sistem.

A. Rancangan Antarmuka Halaman Beranda

Rancangan antarmuka halaman beranda merupakan tampilan utama dari *website*. Gambaran rancangan antarmuka halaman beranda dapat dilihat pada Gambar 3.7.



Gambar 3. 7 Rancangan antarmuka halaman utama

B. Rancangan Antarmuka Halaman Register Mahasiswa

Rancangan antarmuka register mahasiswa digunakan oleh mahasiswa untuk mendaftarkan akun. Gambaran rancangan antarmuka halaman register mahasiswa dapat dilihat pada Gambar 3.8.

Gambar 3. 8 Rancangan antarmuka halaman register mahasiswa

C. Rancangan Antarmuka Halaman Register Dosen

Rancangan antarmuka dosen digunakan oleh dosen untuk mendaftarkan akun. Gambaran rancangan antarmuka halaman register dosen dapat dilihat pada Gambar 3.9.

A Web Page

> Pendaftaran dosen

Nama

Keahlian

NIK/ NIDN

Email

Prodi

Password

Fakultas

Ketik ulang password

Alamat

No hp

Pendidikan terakhir

Submit

Gambar 3. 9 Rancangan antarmuka halaman register dosen

D. Rancangan Antarmuka Halaman *Login*

Rancangan antarmuka halaman *login* digunakan oleh pengguna untuk masuk ke dalam website. Pengguna harus memasukkan *username* dan *password* untuk *login*. Gambaran rancangan antarmuka halaman *login* dapat dilihat pada Gambar 3.10.

A Web Page

> Login

Username

Password

Submit

[Daftar akun mahasiswa](#)

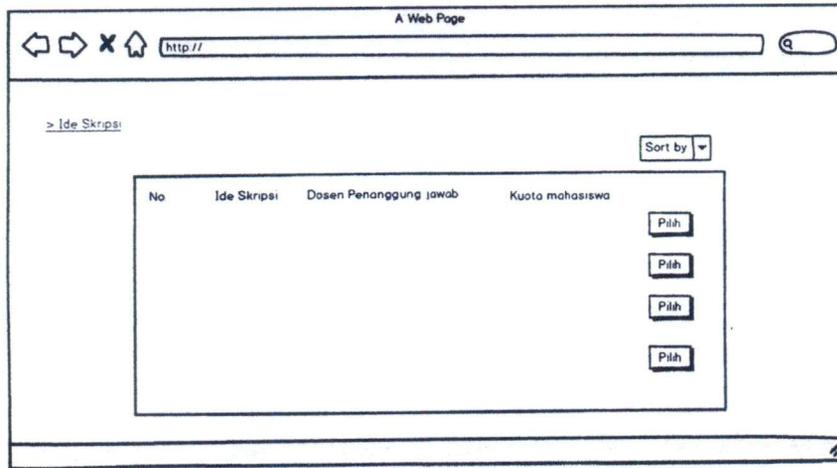
[Daftar akun dosen](#)

Gambar 3. 10 Rancangan antarmuka halaman *login*

E. Rancangan Antarmuka Halaman Ide Skripsi

Rancangan antarmuka halaman ide skripsi berfungsi untuk memberikan informasi data ide skripsi yang ditawarkan oleh dosen. Pada halaman ini,

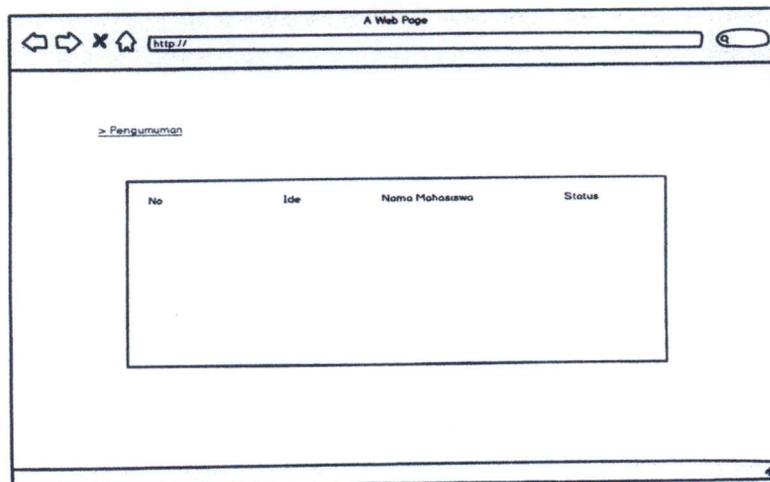
mahasiswa dapat mengambil salah satu ide skripsi yang diinginkan. Gambaran rancangan antarmuka halaman ide skripsi dapat dilihat pada Gambar 3.11.



Gambar 3.11 Rancangan antarmuka halaman ide skripsi

F. Rancangan Antarmuka Halaman Pengumuman

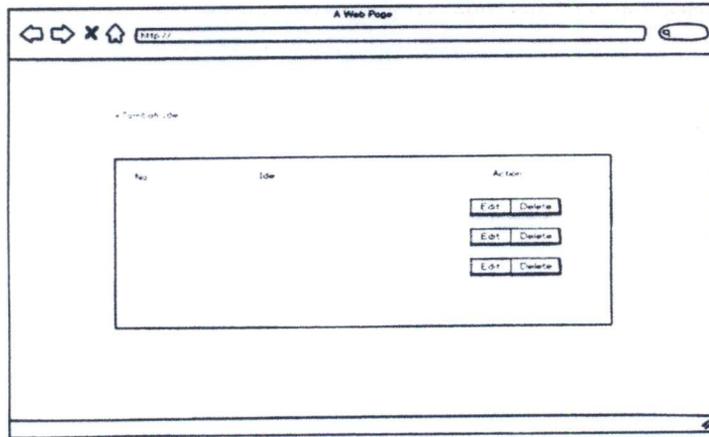
Rancangan antarmuka halaman pengumuman berisi data mahasiswa yang diterima oleh dosen untuk mengambil ide skripsi. Gambaran rancangan antarmuka pengumuman dapat dilihat pada Gambar 3.12.



Gambar 3.12 Rancangan antarmuka halaman pengumuman

G. Rancangan Antarmuka Halaman Kelola Ide Skripsi

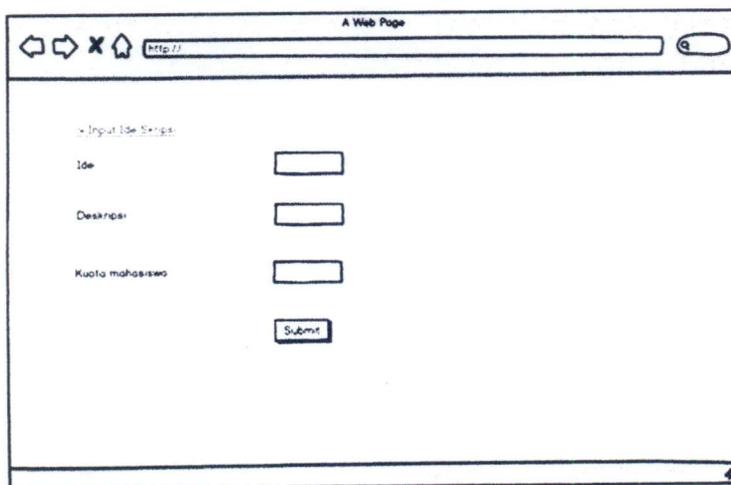
Rancangan antarmuka halaman kelola ide skripsi digunakan oleh dosen untuk mengelola ide skripsi, antara lain menambah, mengubah dan menghapus ide skripsi. Gambaran rancangan antarmuka kelola ide skripsi dapat dilihat pada Gambar 3.13.



Gambar 3. 13 Rancangan antarmuka halaman kelola ide skripsi

H. Rancangan Antarmuka Halaman Input Ide Skripsi

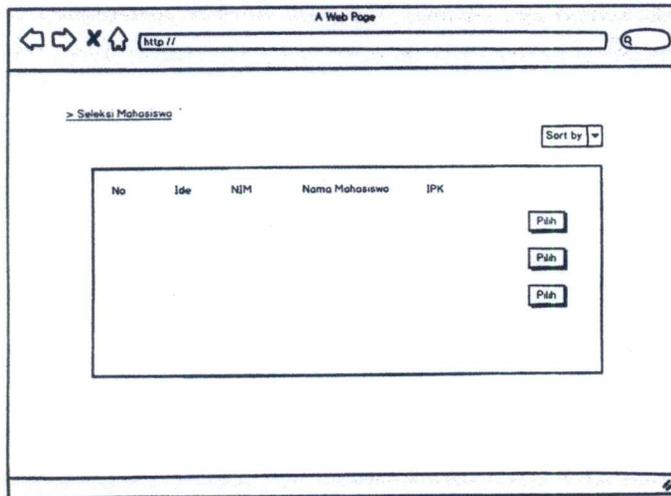
Rancangan antarmuka halaman input ide skripsi digunakan oleh dosen untuk menambah ide skripsi dengan mengisi form tambah ide skripsi. Gambaran rancangan antarmuka halaman input ide skripsi dapat dilihat pada Gambar 3.14.



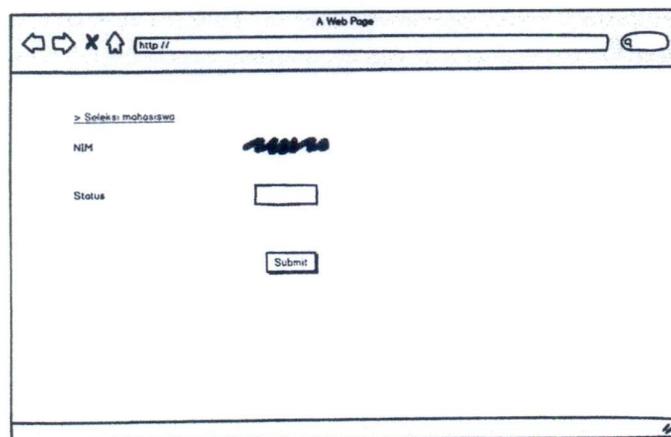
Gambar 3. 14 Rancangan antarmuka halaman input ide skripsi

I. Rancangan Antarmuka Halaman Seleksi Mahasiswa

Rancangan antarmuka halaman seleksi mahasiswa digunakan oleh dosen untuk memilih mahasiswa yang berhak mengambil ide skripsi. Gambaran rancangan antarmuka halaman seleksi mahasiswa dapat dilihat pada Gambar 3.15 dan Gambar 3.16.



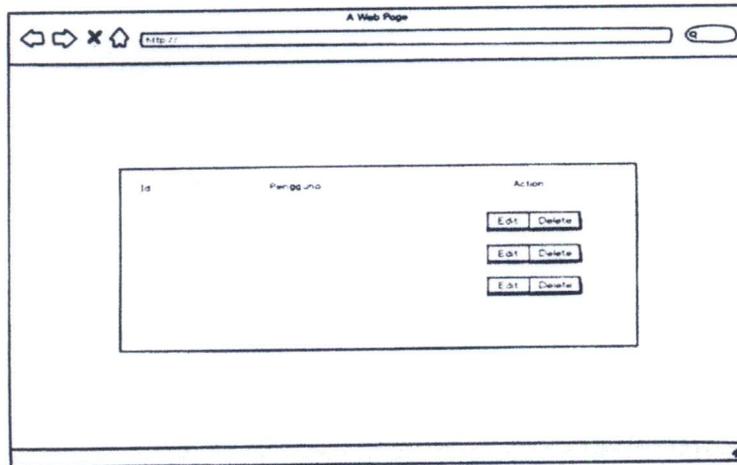
Gambar 3. 15 Rancangan antarmuka seleksi mahasiswa



Gambar 3. 16 Rancangan antarmuka seleksi mahasiswa

J. Rancangan Antarmuka Halaman Administrator

Rancangan antarmuka halaman administrator digunakan oleh administrator untuk mengelola akun pengguna. Gambaran rancangan antarmuka halaman administrator dapat dilihat pada Gambar 3.17.



Gambar 3. 17 Rancangan antarmuka halaman administrator

3.6 Pengujian

Pegujian perangkat lunak adalah investigasi yang dilakukan untuk mendapatkan informasi mengenai kualitas produk yang diuji. Pengujian dalam pengembangan aplikasi bertujuan untuk menyesuaikan kebutuhan yang diperlukan dengan rancangan aplikasi.

Metode pengujian yang digunakan dalam aplikasi berbasis website ini adalah *black box testing*. *Blackbox testing* adalah pengujian perangkat lunak dengan menguji fungsionalitas aplikasi tanpa melihat struktur internal.

Hal-hal yang menjadi perhatian dalam pengujian adalah sebagai berikut:

1. Aplikasi dapat menyimpan dan menampilkan ide skripsi yang diinput oleh dosen.
2. Aplikasi dapat menyimpan dan menampilkan mahasiswa yang mengambil ide skripsi.
3. Aplikasi dapat menyimpan dan menampilkan pengumuman mahasiswa yang diterima oleh dosen.