BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

Implementasi adalah proses penerapan rancangan program yang telah dibuat pada bab sebelumnya atau aplikasi dalam melaksanakan sistem informasi pemrograman yang telah dibuat, hasil dari tahapan implementasi ini adalah suatu sistem pengolahan data yang sudah dapat berjalan dengan baik. Dengan demikian dapat diketahui apakah perangkat lunak Pengembangan Sistem Informasi Skripsi dan Tugas Akhir Universitas Komputer Indonesia sesuai dengan tujuan yang diharapkan.

5.1.1 Batasan Implementasi

Dalam rencana penerapan sistem yang terkomputerisi, yakni agar sistem siap dioperasikan, maka perlu diadakan kegiatan-kegiatan dari penerapannya. Adapun langkah-langkah yang perlu dilakukan untuk menerapkan sistem tersebut adalah pembuatan program, testing program, pelatihan serta terima dokumentasi, akan tetapi dalam penulisan tugas akhir ini implementasi sistem yang dilakukan hanya sampai pada tahap pembuatan program

5.1.2 Implementasi Perangkat Lunak

Untuk mendukung kelancaran sistem informasi yang dirancang, maka sistem ini memerlukan perangkat lunak. Perangkat lunak digunakan untuk mendukung kinerja sistem operasi dan aplikasi database. Adapun perangkat lunak yang digunakan adalah sebagai berikut :

1. Operating System : Microsoft Windows XP,Vista,7,8
2. Browser : Internet Explore, Mozilla Firefox, Opera, Google Chrome
3. Web Server : XAMPP versi > 1.4
4. Adobe Reader untuk mencetak

5.1.3 Implementasi Perangkat Keras

Untuk mendukung terlaksananya penerapan aplikasi PHP yang dirancang, maka sistem pengoperasian ini membutuhkan komputer sebagai server yang memudahkan dalam menjalankan program aplikasi dengan spesifikasi sebagai berikut :

Server :

a. Menggunakan minimal processor Intel Pentium dual core 2,8 GHz atau sekelasnya.
b. Menggunakan RAM minimal 1 Gb.
c. Tersedianya Hard Drive sebagai media penyimpanan, minimal 100 Gb
d. Monitor, Mouse, Keyboard sebagai peralatan antar muka.

Client:

a. Menggunakan minimal processor Intel Pentium Dual Core IV 2.0 GHz atau sekelasnya.
b. Menggunakan RAM minimal 1 Gb.
c. Tersedianya Hard Drive sebagai media penyimpanan, minimal 100 Gb
d. Monitor, Mouse, Keyboard sebagai peralatan antar muka.
e. Printer

Konfigurasi jaringan (Networking):

a. Switch HUB 8 port 10/100 Mbps i buah
b. Lan Card 3 buah
c. Kabel UTP
d. Conector Rj 45

5.1.4 Implementasi Basis Data (Sintaks SQL)

Implementasi basis data yang digunakan memakai MySQL dan perancangan Database menggunakan PHPMyAdmin. Adapun implementasi perancangan basis data adalah sebagai berikut:

```
-- ----------------------------
-- Table structure for `ci_sessions`
-- ----------------------------
DROP TABLE IF EXISTS `ci_sessions`;
CREATE TABLE `ci_sessions` (
  `session_id` varchar(40) NOT NULL DEFAULT '0',
  `ip_address` varchar(45) NOT NULL DEFAULT '0',
  `user_agent` varchar(120) NOT NULL,
  `last_activity` int(10) unsigned NOT NULL DEFAULT '0',
  `user_data` text NOT NULL,
```

```
  PRIMARY KEY (`session_id`),
  KEY `last_activity_idx` (`last_activity`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;


-- ----------------------------
-- Table structure for `daftar_proposal`
-- ----------------------------
DROP TABLE IF EXISTS `daftar_proposal`;
CREATE TABLE `daftar_proposal` (
  `id` int(20) NOT NULL AUTO_INCREMENT,
  `periode_id` smallint(3) NOT NULL,
  `judul_skripsi_id` int(11) NOT NULL,
  `pembimbing_1` bigint(11) NOT NULL DEFAULT '0',
  `pembimbing_2` bigint(11) DEFAULT '0',
  `pembimbing_3` bigint(11) DEFAULT '0',
  `tgl_daftar` date NOT NULL,
  `tgl_acc` date DEFAULT NULL,
  `status` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  UNIQUE KEY `mhs` (`judul_skripsi_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;


-- ----------------------------
-- Table structure for `jenis_skripsi`
-- ----------------------------
DROP TABLE IF EXISTS `jenis_skripsi`;
CREATE TABLE `jenis_skripsi` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `jenis_skripsi` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;


-- ----------------------------
-- Table structure for `judul_skripsi`
-- ----------------------------
DROP TABLE IF EXISTS `judul_skripsi`;
CREATE TABLE `judul_skripsi` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `Mahasiswa_id` int(11) NOT NULL DEFAULT '0',
  `judul_indo` varchar(200) DEFAULT NULL,
  `judul_eng` varchar(200) DEFAULT NULL,
  `des_indo` text,
  `des_eng` text,
  `jenis_skripsi_id` int(11) DEFAULT NULL,
  `dsn_pembimbing_id` int(11) DEFAULT NULL,
  `tahun_ajaran_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`,`Mahasiswa_id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-- ----------------------------
-- Table structure for `jurusan`
-- ----------------------------
DROP TABLE IF EXISTS `jurusan`;
CREATE TABLE `jurusan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `jurusan` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;

 ----------------------------
-- Table structure for `Mahasiswa`
-- ----------------------------
DROP TABLE IF EXISTS `Mahasiswa`;
CREATE TABLE `Mahasiswa` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nim` int(8) NOT NULL,
  `nama` varchar(30) NOT NULL,
  `tempat_lahir` varchar(20) NOT NULL,
  `tanggal_lahir` date NOT NULL,
  `jenis_kelamin` char(1) NOT NULL,
  `alamat1` varchar(200) NOT NULL,
  `alamat2` varchar(200) NOT NULL,
  `email` varchar(50) NOT NULL,
  `telepon` varchar(15) NOT NULL,
  `jj` smallint(3) NOT NULL,
  `angkatan` year(4) NOT NULL,
  `sks` smallint(3) NOT NULL,
  `bapak` varchar(30) NOT NULL,
  `Ibu` varchar(50) NOT NULL,
  `alamat_ortu` varchar(200) NOT NULL,
  `telepon_ortu` varchar(15) NOT NULL,
  `pekerjaan_ortu` varchar(20) DEFAULT NULL,
  `foto` varchar(40) DEFAULT NULL,
  `wali_id` bigint(11) NOT NULL DEFAULT '0',
  `status` tinyint(1) DEFAULT '0',
  `password` varchar(200) DEFAULT NULL,
  `jurusan_id` int(11) NOT NULL,
  `access_token` varchar(300) DEFAULT NULL,
  `step` tinyint(1) DEFAULT NULL,
  `lulus` tinyint(1) NOT NULL,
  `semester` int(2) DEFAULT NULL,
  PRIMARY KEY (`id`,`nim`,`email`)
) ENGINE=MyISAM AUTO_INCREMENT=31 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `Mahasiswa_persyaratan`
-- ----------------------------
DROP TABLE IF EXISTS `Mahasiswa_persyaratan`;
CREATE TABLE `Mahasiswa_persyaratan` (
```

```
  `id` int(50) NOT NULL AUTO_INCREMENT,
  `daftar_id` int(10) DEFAULT NULL,
  `persyaratan_id` int(10) DEFAULT NULL,
  `class_name` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `Mahasiswa_tahun_ajaran`
-- ----------------------------
DROP TABLE IF EXISTS `Mahasiswa_tahun_ajaran`;
CREATE TABLE `Mahasiswa_tahun_ajaran` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `Mahasiswa_id` int(11) DEFAULT NULL,
  `tahun_ajaran_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=61 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `periode`
-- ----------------------------
DROP TABLE IF EXISTS `periode`;
CREATE TABLE `periode` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `pendaftaran` varchar(50) NOT NULL,
  `tgl_awal` date NOT NULL,
  `tgl_akhir` date NOT NULL,
  `status` tinyint(1) DEFAULT NULL,
  `tahun_ajaran_id` int(10) NOT NULL,
  `nilai` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `persyaratan`
-- ----------------------------
DROP TABLE IF EXISTS `persyaratan`;
CREATE TABLE `persyaratan` (
  `id` int(50) NOT NULL AUTO_INCREMENT,
  `persyaratan` varchar(100) DEFAULT NULL,
  `step` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `publik_info`
-- ----------------------------
DROP TABLE IF EXISTS `publik_info`;
CREATE TABLE `publik_info` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `page` varchar(200) DEFAULT NULL,
```

```
  `desc` text,
  `title` varchar(255) NOT NULL,
  `staff_id` int(11) NOT NULL,
  `tipe_info` tinyint(4) NOT NULL,
  `date_created` datetime NOT NULL,
  `date_updated` datetime NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `room`
-- ----------------------------
DROP TABLE IF EXISTS `room`;
CREATE TABLE `room` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `room_name` varchar(11) NOT NULL DEFAULT '0',
  `qouta` int(11) DEFAULT NULL,
  `status` tinyint(1) NOT NULL,
  `step_id` int(11) DEFAULT NULL,
  `staff_id` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `rule`
-- ----------------------------
DROP TABLE IF EXISTS `rule`;
CREATE TABLE `rule` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `rule` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `staff`
-- ----------------------------
DROP TABLE IF EXISTS `staff`;
CREATE TABLE `staff` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nip` bigint(11) NOT NULL,
  `nama` varchar(40) NOT NULL,
  `jenis_kelamin` char(1) NOT NULL,
  `tempat_lahir` varchar(20) NOT NULL,
  `tanggal_lahir` date NOT NULL,
  `alamat` varchar(200) NOT NULL,
  `email` varchar(50) NOT NULL,
  `telepon` varchar(15) NOT NULL,
  `foto` varchar(40) DEFAULT NULL,
  `jj` smallint(3) NOT NULL DEFAULT '0',
  `status_staff_id` int(11) NOT NULL DEFAULT '0',
  `jurusan_id` int(11) NOT NULL,
```

```
  `username` varchar(20) NOT NULL,
  `password` varchar(200) DEFAULT NULL,
  `access_token` varchar(300) DEFAULT NULL,
  `acc_admin` tinyint(1) NOT NULL,
  PRIMARY KEY (`id`,`nip`,`email`)
) ENGINE=InnoDB AUTO_INCREMENT=7 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `staff_rules`
-- ----------------------------
DROP TABLE IF EXISTS `staff_rules`;
CREATE TABLE `staff_rules` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `staff_id` int(11) DEFAULT NULL,
  `rule_id` int(11) DEFAULT NULL,
  `option_pembimbing` int(1) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=11 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `status_staff`
-- ----------------------------
DROP TABLE IF EXISTS `status_staff`;
CREATE TABLE `status_staff` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `status` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `step`
-- ----------------------------
DROP TABLE IF EXISTS `step`;
CREATE TABLE `step` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `step` varchar(200) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=latin1;

-- ----------------------------
-- Table structure for `tahun_ajaran`
-- ----------------------------
DROP TABLE IF EXISTS `tahun_ajaran`;
CREATE TABLE `tahun_ajaran` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `tahun_awal` year(4) NOT NULL,
  `tahun_akhir` year(4) NOT NULL,
  `smt` tinyint(1) NOT NULL DEFAULT '0',
  `status` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;
```

```
-- ----------------------------
-- Table structure for `uji_program`
-- ----------------------------
DROP TABLE IF EXISTS `uji_program`;
CREATE TABLE `uji_program` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `Mahasiswa_id` int(11) NOT NULL DEFAULT '0',
  `periode_id` int(11) NOT NULL,
  `nilai` int(11) DEFAULT NULL,
  `status` tinyint(1) DEFAULT NULL,
  `input_admin` tinyint(1) DEFAULT NULL,
  `room_id` int(11) DEFAULT NULL,
  `no_kursi` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`,`Mahasiswa_id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=latin1;
```

5.1.5 Implementasi Antar Muka

Tampilan antar muka *(interface)* merupakan tampilan program yang akan langsung berhadapan dengan user. User akan berinteraksi memberikan input atau menerima output melalui tampilan *interface* ini. Antar muka sangatlah penting untuk dipikirkan secara matang. pada implementasi antar muka ini akan dijelaskan secara lebih rinci dalam *point* penggunaan program.

5.1.6 Implementasi Instalasi Program

Berikut ini adalah cara menginstall XAMPP:

i. Jalankan file **xampp-win32-1.4.12-installer.exe** (atau versi lainnya)

ii. Kemudian akan tampil pilihan untuk memilih bahasa ketika proses instalasi berjalan. Silakan pilih bahasa **Indonesian** atau **English**, kecuali anda menguasai bahasa lainnya. Pada contoh ini saya memilih bahasa Indonesian karena saya cinta bahasa Indonesia.



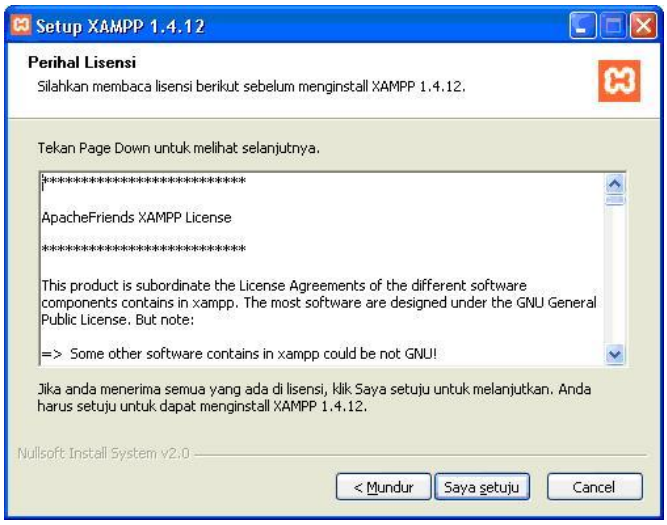
Gambar V.1 XAMPP Pemilihan Bahasa

iii. Proses instalasi akan dimulai. Klik **Maju** untuk memulainya.



Gambar V.2 XAMPP Setup

iv. Akan muncul lisensi software. Silahkan membacanya jika anda mau, tetapi saya lebih suka untuk tidak membacanya karena terlalu banyak. Klik **Saya Setuju** untuk melanjutkan.



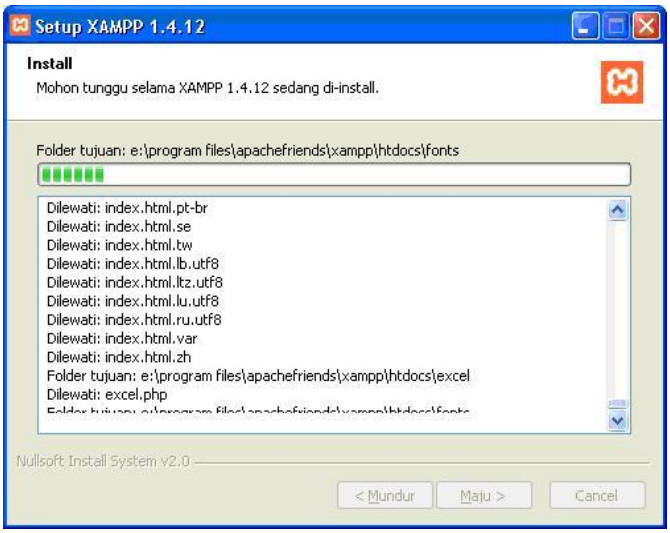
Gambar V.3 XAMPP Lisensi Software

v. Selanjutnya silakan anda **pilih lokasi install** untuk XAMPP. Kemudian klik **install**



Gambar V.4 XAMPP Lisensi Software

vi. Tunggu beberapa saat sampai proses instalasi selesai.



Gambar V.5 XAMPP Loading Install

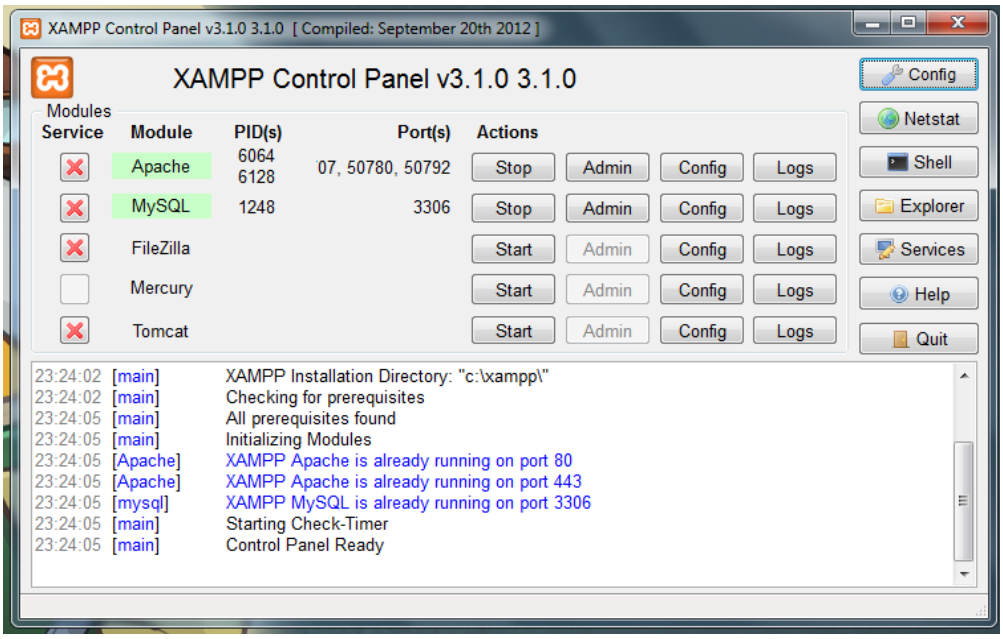
vii. Instalasi selesai



Gambar V.6 XAMPP Install Selesai

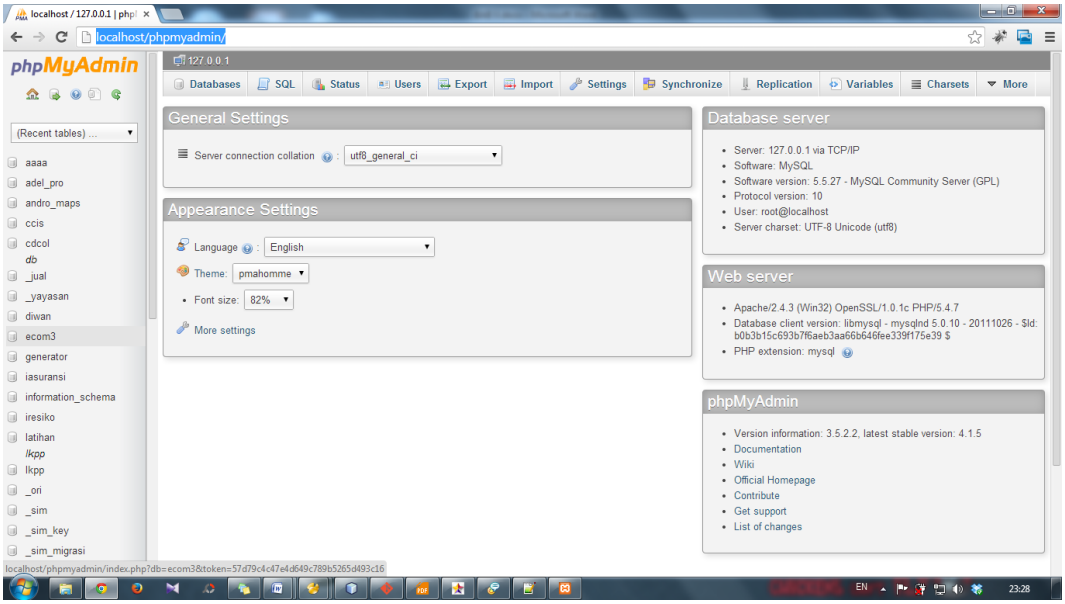
viii. Copy Aplikasi ke folder htdocs didalam folder XAMPP, bila location install di root C: maka copy ke C:\xampp\htdocs

ix. Jalankan Service Apache dan MySql di Window Control XAMPP



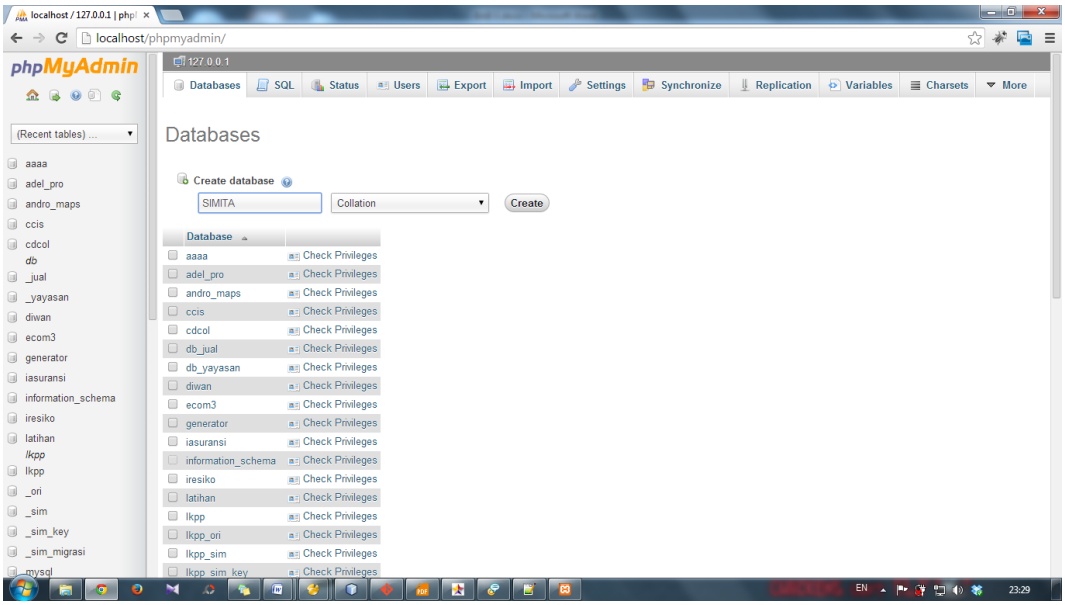
Gambar V.7 aktivasi service

x. Buka URL http://localhost/phpmyadmin/ pada browser



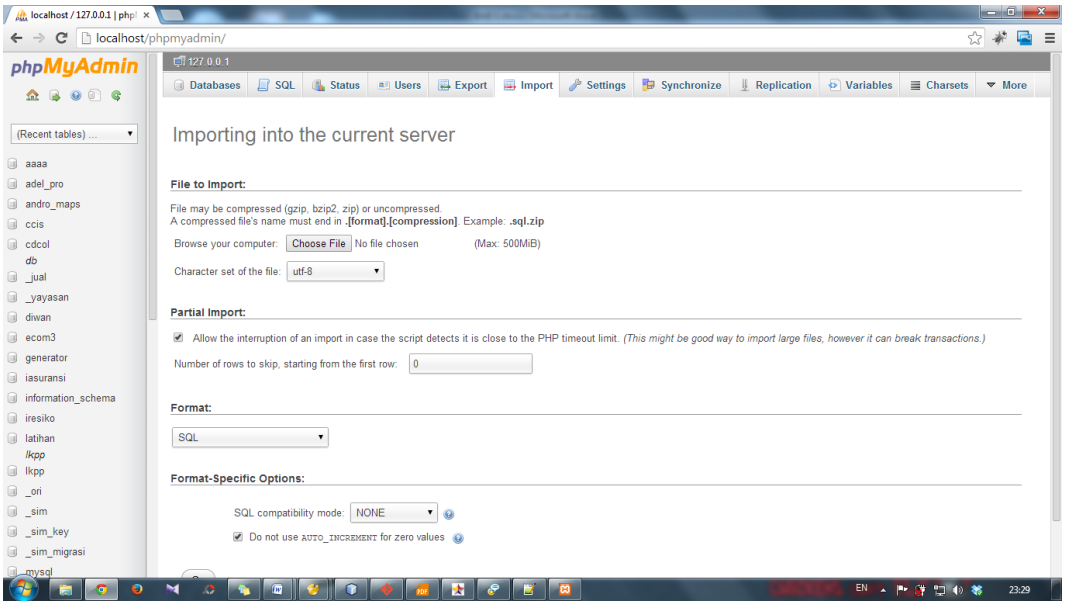
Gambar V.8 Phpmyadmin

xi. Buat database baru dengan nama yang di inginkan, misalkan simita



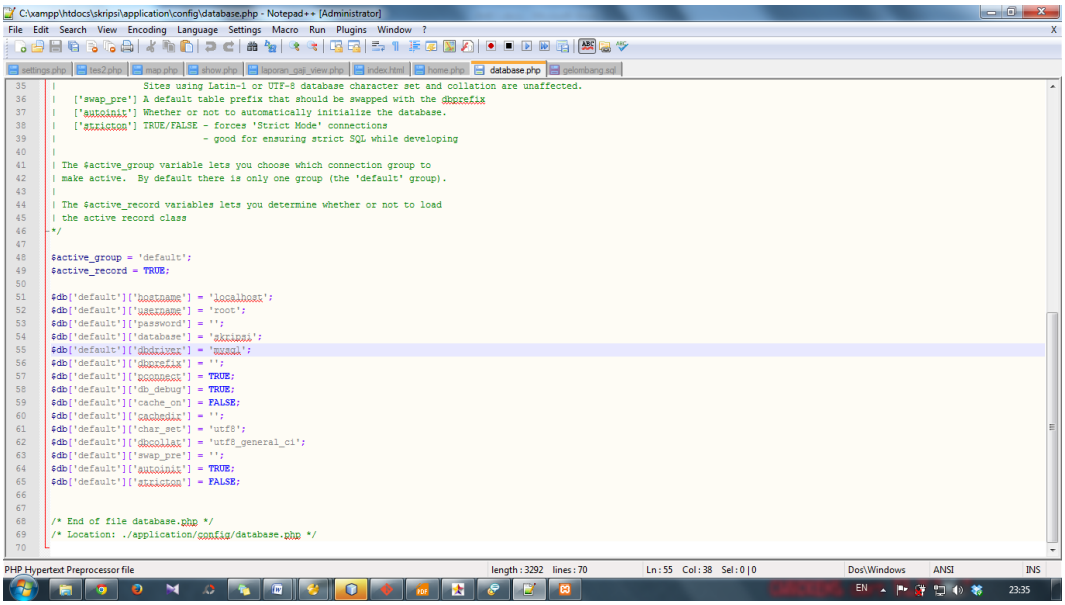
Gambar V.9 Membuat Database

xii. Masuk ke menu *Import*, masukan file Dump SQL (Ekstensi .sql / sql.zip) setelah tekan tombol *GO*



Gambar V.10 import database

xiii. Setelah semua data di database sudah masuk. Maka konfigurasi file database.php di C:\xampp\htdocs\nama_aplikasi\application\config



Gambar V.11 konfigurasi database

xiv. Setelah semua konfigurasi dan data selesai maka langkah terakhir adalah membuka aplikasi website di browser dengan mengetikan URL http://localhost/nama_aplikasi/



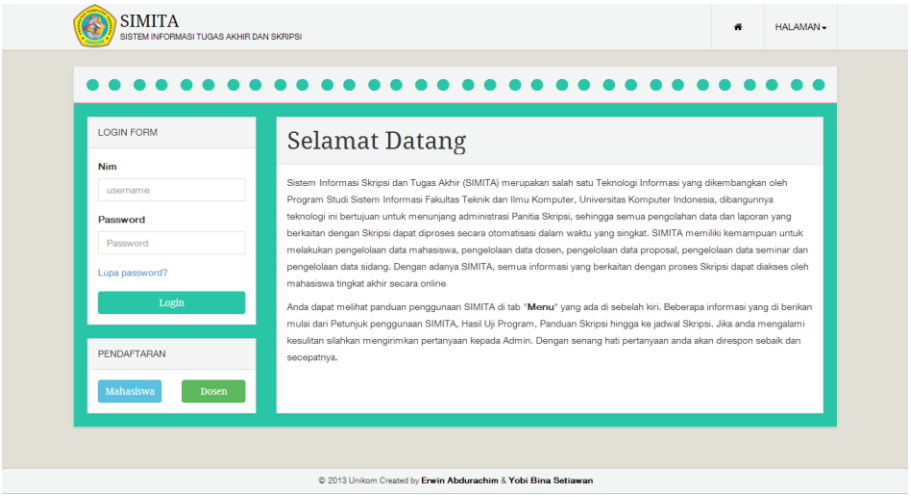
Gambar V.12 Instalasi Aplikasi Berhasil

5.1.7 Penggunaan Program

Penggunaan program merupakan langkah-langkah dalam menggunakan dan menjalankan suatu sistem informasi yang dilakukan oleh pengguna (user). Langkah-langkah penggunaan program sebagai berikut :

1. Halaman Utama dan *Login*

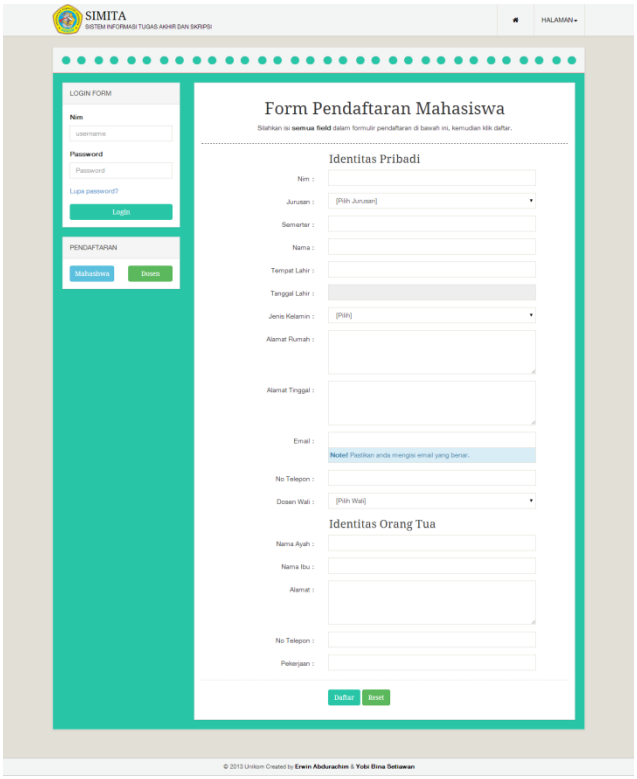
   Halaman ini menampilkan informasi seputar gambaran umum mengenai Sistem Informasi Tugas Akhir dan Skripsi, dalam halam ini terdapat form login untuk Mahasiswa.



Gambar V.13 Antar Muka Halaman Utama dan *Login*

2. Halaman Pendaftaran Mahasiswa

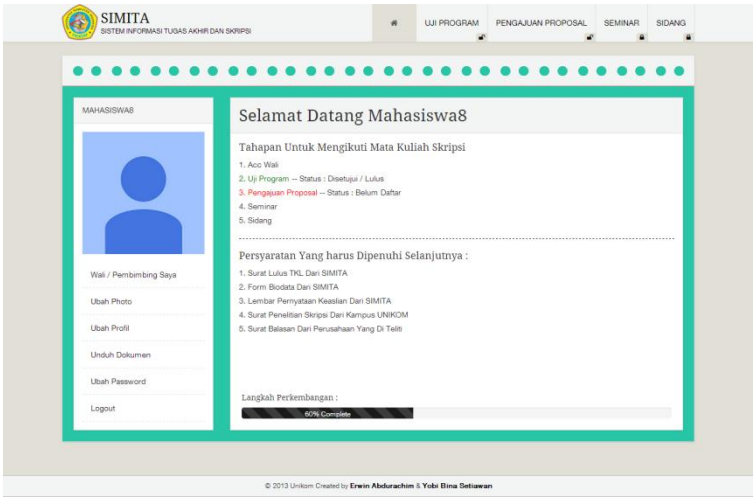
   Pada Halaman ini Mahasiswa dapat mendaftarkan dirinya untuk mengikuti proses Tugas Akhir atau Skripsi.



Gambar V.14 Antar Muka Halaman Pendaftaran Mahasiswa

3. Halaman Utama Mahasiswa

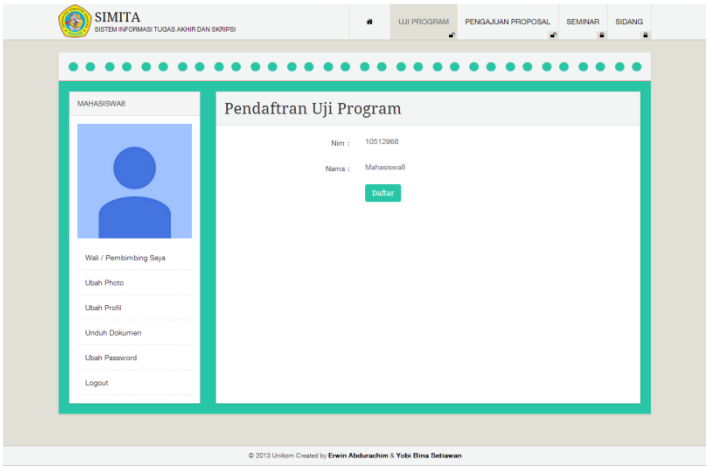
Setelah Mahasiswa berhasil login maka Mahasiswa dapat melihat informasi mengenai status tahapan mengenai mata kuliah Tugas Akhir atau Skripsi.



Gambar V.15 Antar Muka Halaman Utama Mahasiswa

4. Halaman Pendaftaran Uji Program

   Setakah Mahasiswa telah disetujui oleh dosen wali, maka pada halaman ini Mahasiswa dapat mendaftrakan dirinya untuk mengikuti uji program.

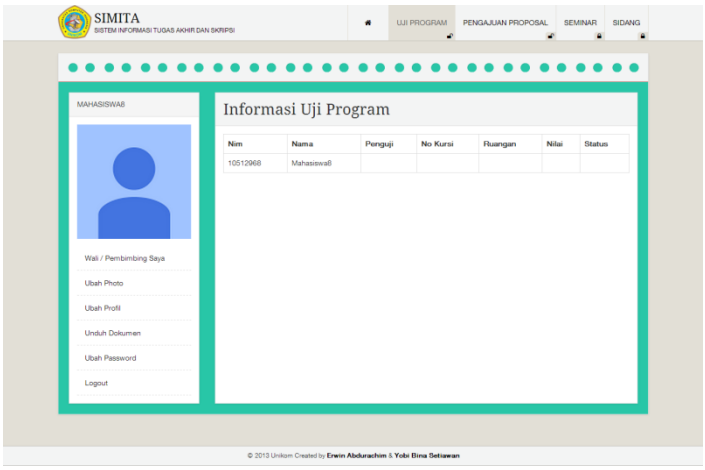


Gambar V.16 Antar Muka Halaman Pendaftaran Uji Program

Ketika mahasiswa menekan tombol daftar maka mahasiswa akan terdaftar ke dalam sistem yang akan akan mengikuti uji program.

5. Halaman Informasi Uji Program

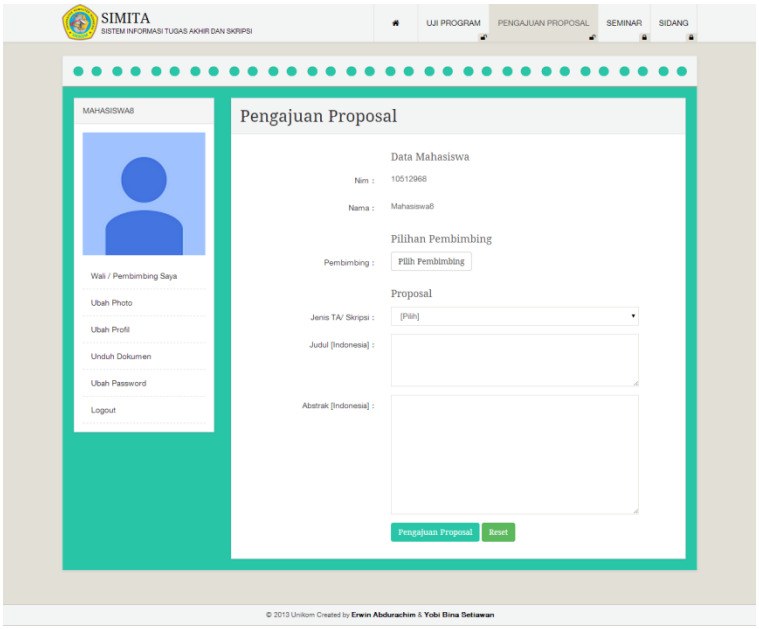
   Halaman ini menampilkan informasi mengenai uji program setelah Mahasiswa melakukan pendaftaran uji program.



Gambar V.17 Antar Muka Halaman Informasi Uji Program

6. Halaman Pengajuan Proposal

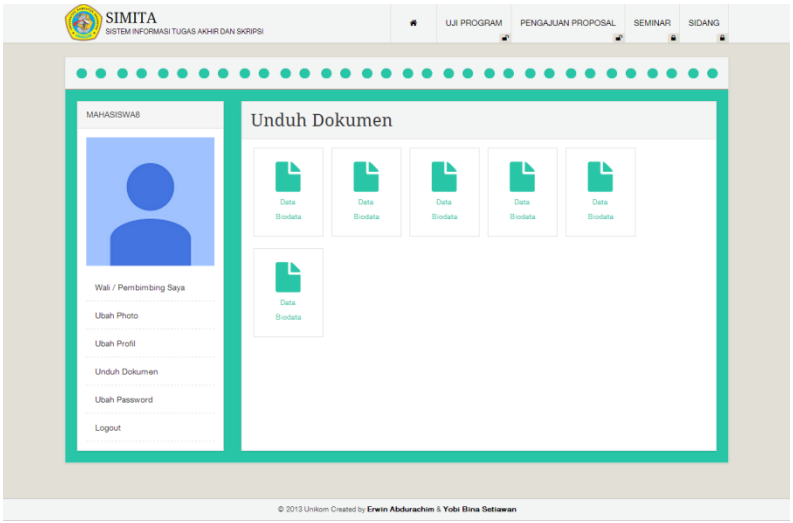
   Setelah Mahasiswa lulus dalam pelaksanaan uji program maka pada halaman ini Mahasiswa dapat melakukan pengajuan proposal.



Gambar V.18 Antar Muka Halaman Pengajuan Proposal

7. Halaman Unduh Dokumen

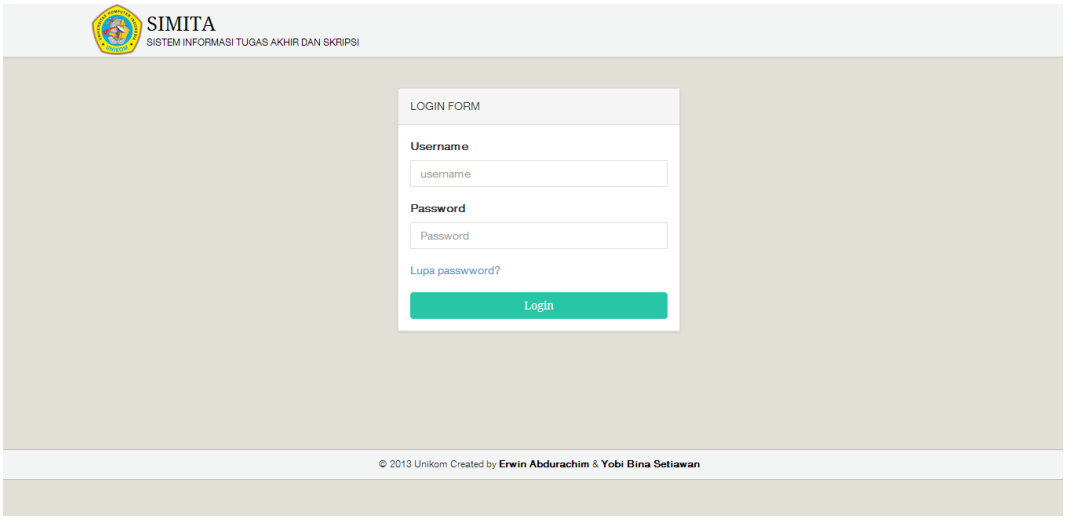
Pada halaman ini Mahasiswa dapat mengunduh dokumen yang tersedia di dalam sistem.



Gambar V.19 Antar Muka Halaman Pengajuan Proposal

8. Halaman Pendaftaran Dosen(*Staff*)

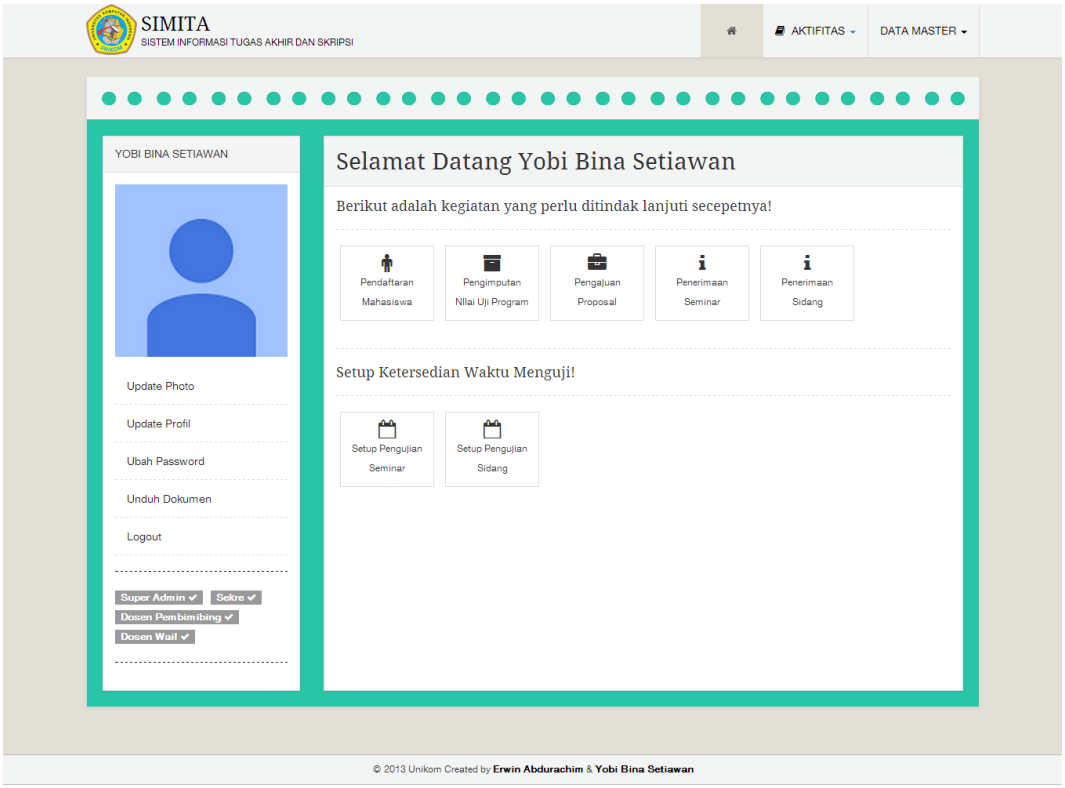
Halaman ini *Staff* dapat login dan masuk ke dalam sistem.



Gambar V.20 Antar Muka Halaman Login *Staff*

9. Halaman Utama Dosen(*Staff*)

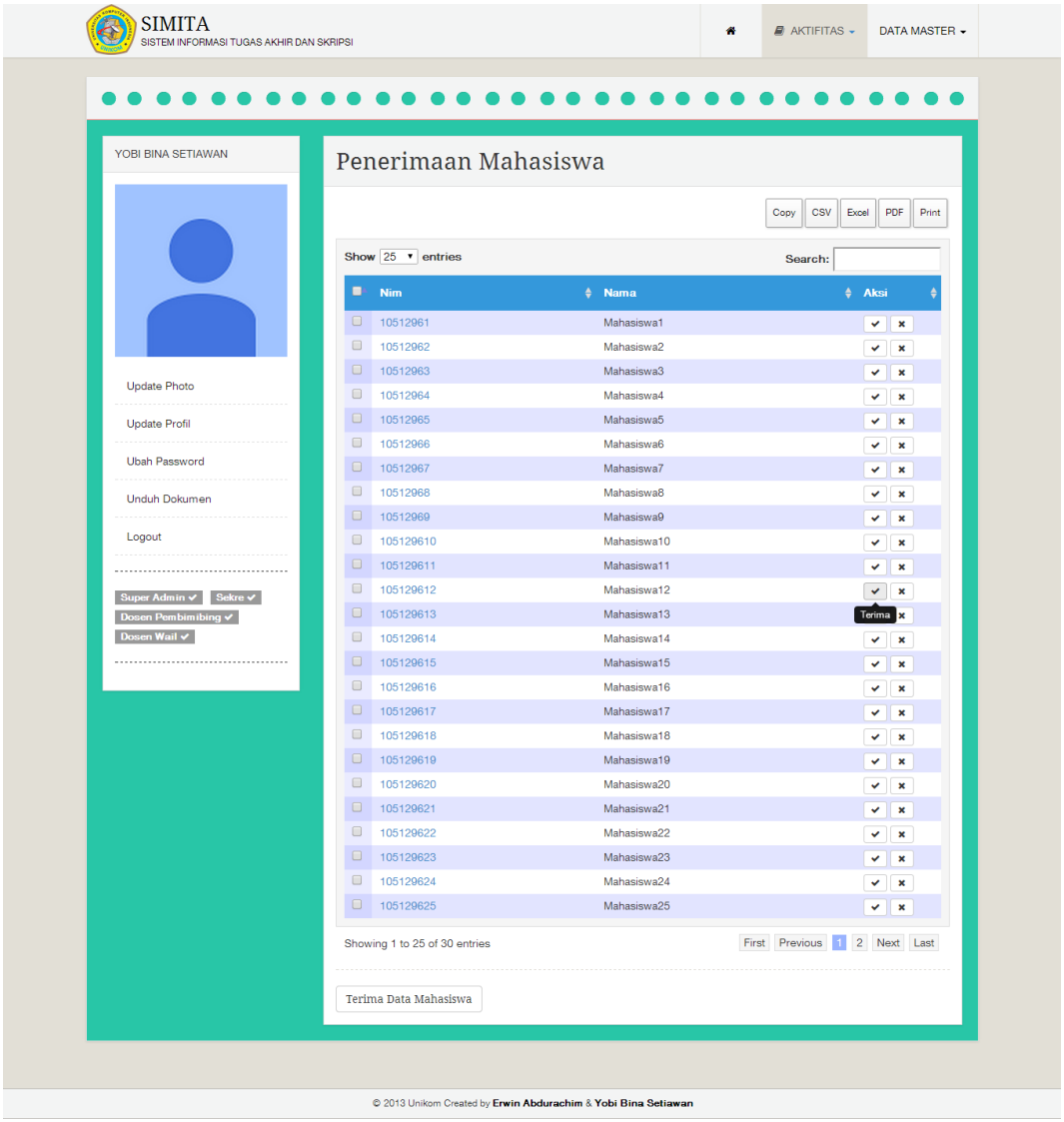
Setelah *Staff* berhasil login maka akan menampilkan halaman seperti di bawah ini.



Gambar V.21 Antar Muka Halaman Utama *Staff*

10. Halaman Penerimaan Mahasiswa

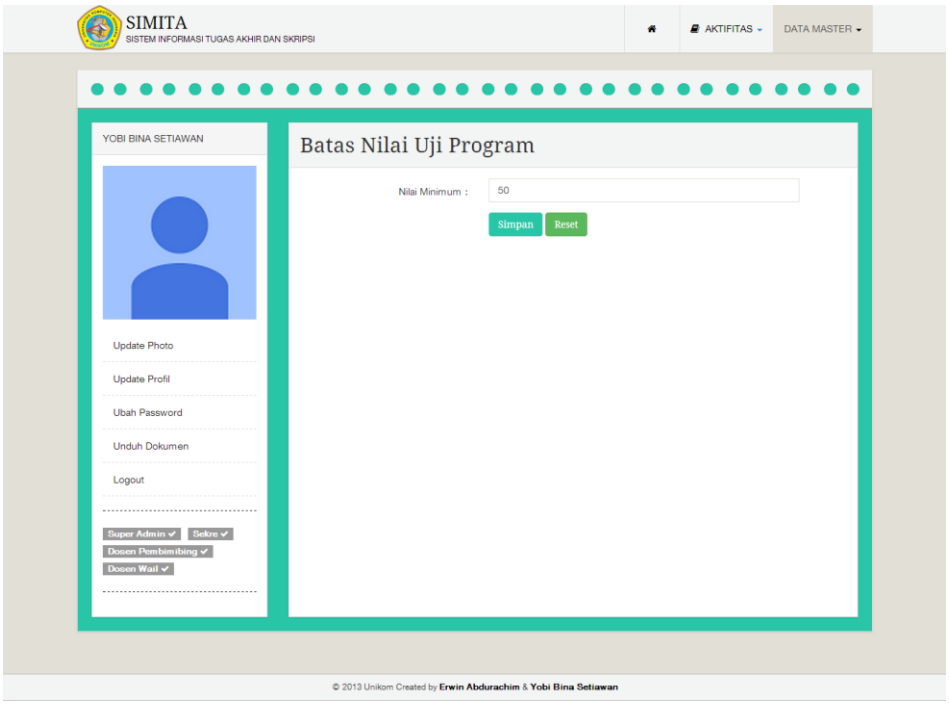
Pada halaman ini Dosen Wali akan menvalidasi data Mahasiswa yang mengikuti Tugas Akhir atau Skripsi



Gambar V.22 Antar Muka Halaman Penerimaan Mahasiswa

11. Halaman Input Batas Nilai Uji Program

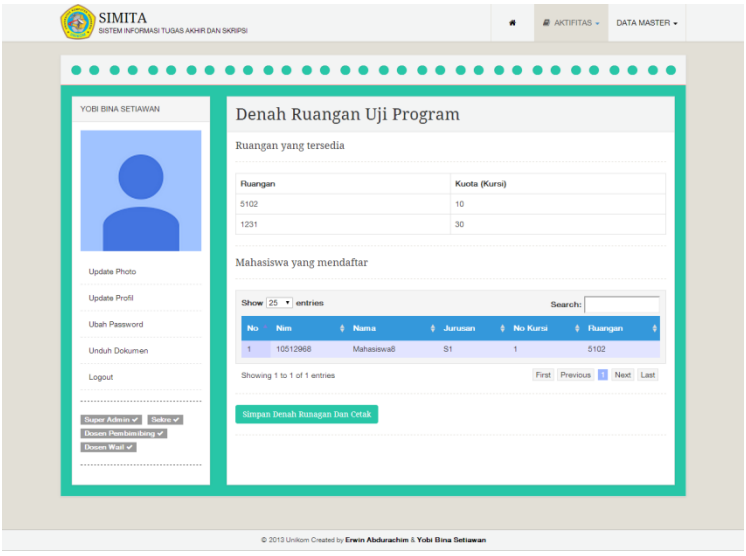
Pada halaman ini *Staff* admin mengatur batas nilai untuk uji program.



Gambar V.23 Antar Muka Halaman Batas Nilai Uji Program

12. Halaman Ruangan Uji Program

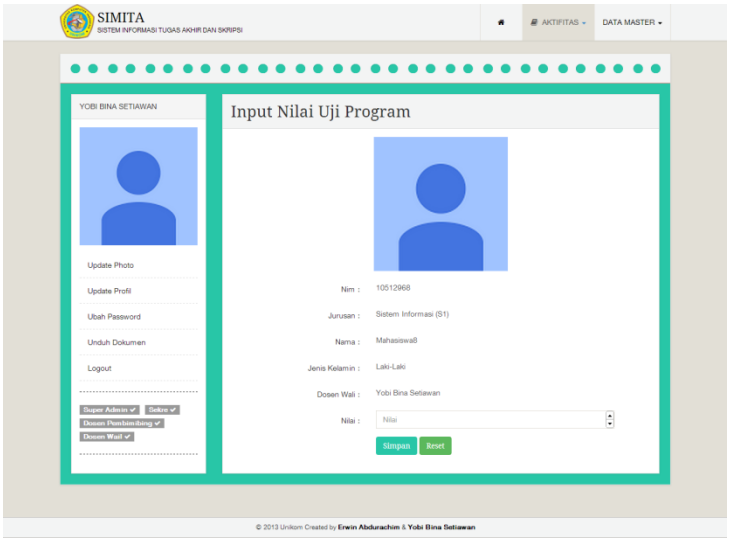
Pada halaman ini *Staff Admin* dapat generate data ruangan untuk uji program.



Gambar V.24 Antar Muka Halaman Ruangan Uji Program

13. Halaman Input Nilai Uji Program

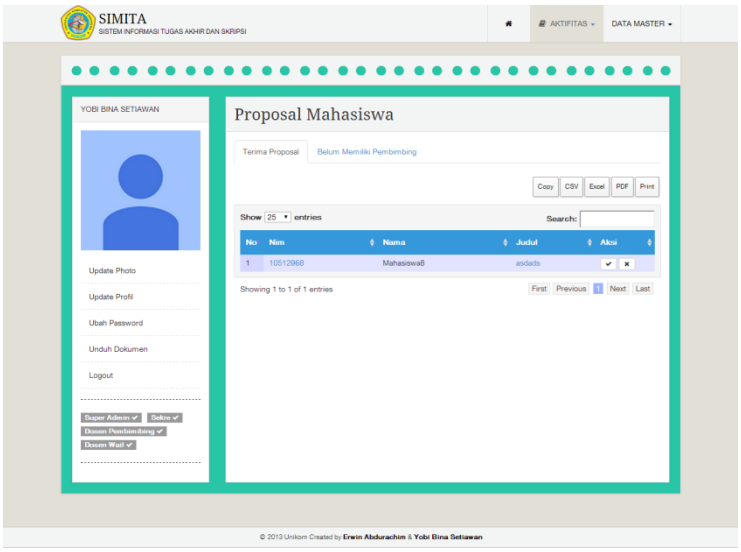
Halaman ini S*taff admin* dapat menginputkan nilai uji program mahasiswa.



Gambar V.25 Antar Muka Halaman Input Nilai Uji Program

14. Halaman Penerimaan Proposal

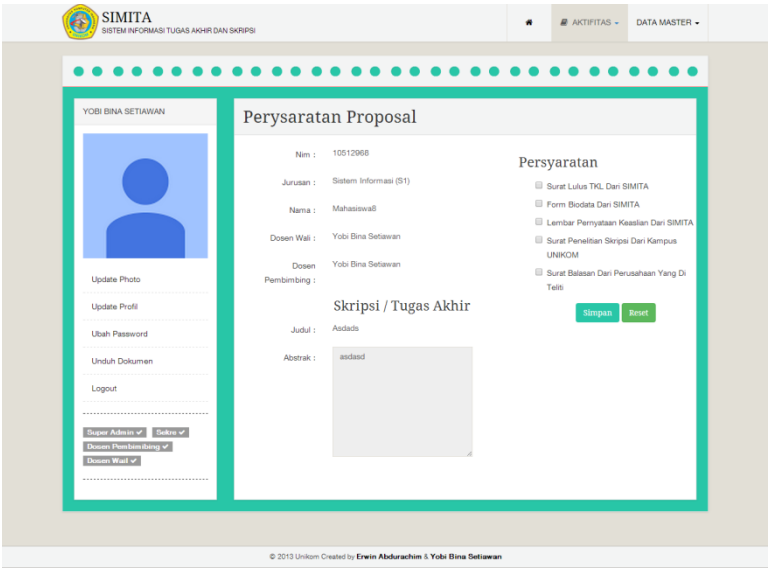
Pada halaman ini *Staff Admin* dapat menentukan apakan proposal mahasiswa dapat diterima.



Gambar V.26 Antar Muka Halaman Penerimaan Proposal

15. Halaman Persyaratan Proposal

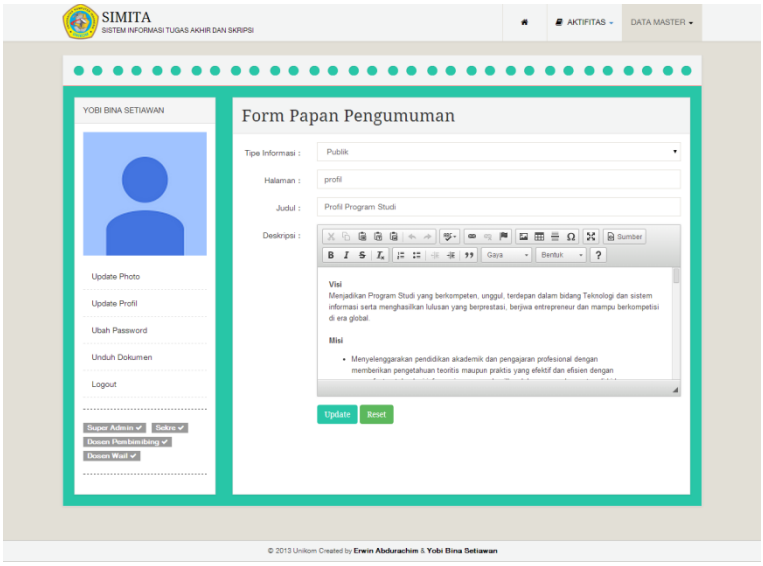
Halaman ini Sekretariat akan memeriksa suatu kelengkapan persyaratan proposal mahasiswa.



Gambar V.27 Antar Muka Halaman Persyaratan Proposal

16. Halaman Input Papan Pengumuman

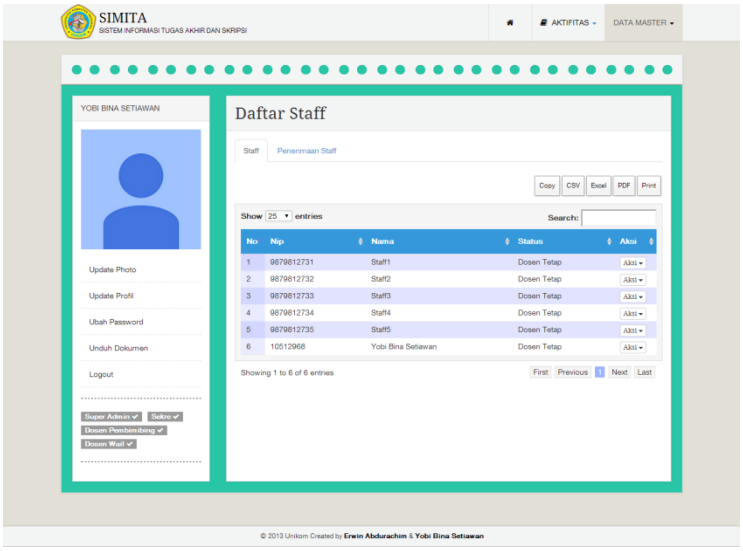
Pada halaman ini *Staff Admin* dapat membuat dan memberikan informasi papan pengumuan.



Gambar V.28 Antar Muka Halaman Penerimaan Mahasiswa

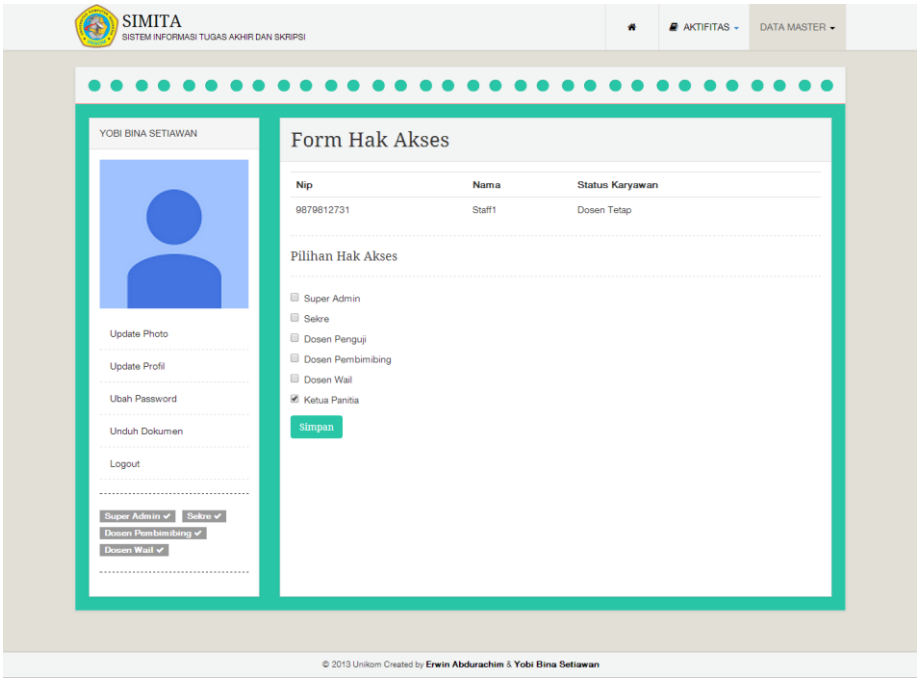
17. Pengelola Data Master *Staff*

Pada halaman ini *Staff admin* dapat mengelola data staff



Gambar V.29 Antar Muka Halaman Penerimaan Mahasiswa

Ketika *Staff* mendaftarakan dirinya ke dalam sistem maka pada halaman ini admin dapat memberikan hak akses sesuai kepada *Staff* tersebut

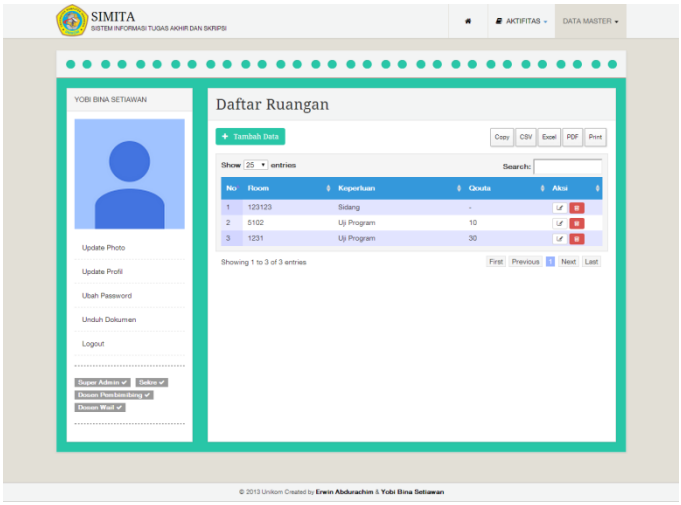


Gambar V.30 Antar Muka Halaman Memberi Hak Akses Kepada *Staff*

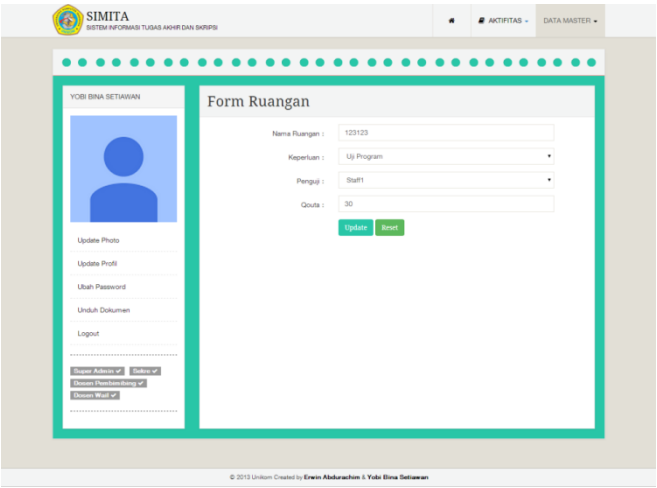
Pada gambar di atas maka Admin dapat menentukan hak akses suatu staff yang terpilih dan menyipannya ke dalam sistem.

18. Mengelola Data Master Ruangan

Pada halaman ini *Staff Admin* dapat mengelola data ruangan.



Gambar V.31 Antar Muka Halaman Daftar Ruangan

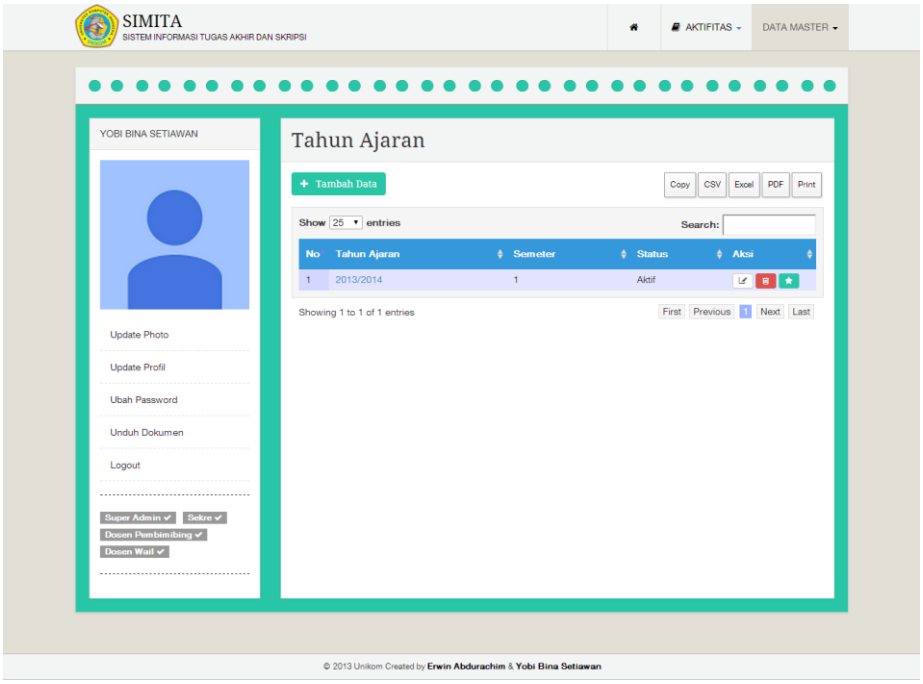


Gambar V.32 Antar Muka Halaman Form Data Ruangan

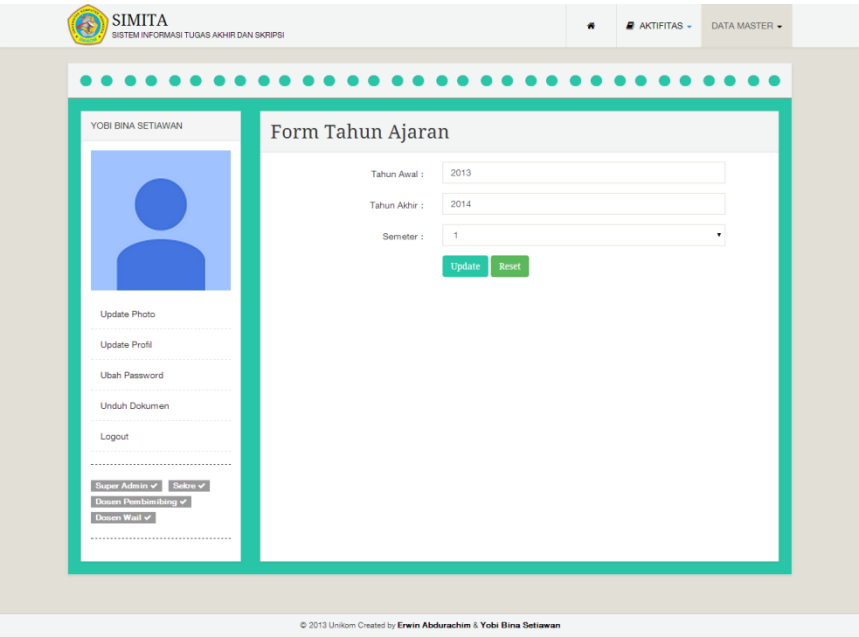
Pada form di atas maka admin dapat menambaha dan mengedit ruangan sesuai kebutuhan.

19. Mengelola Data Tahun Ajaran

Pada halaman ini *Staff Admin* dapat mengelola data Tahun Ajaran.



Gambar V.33 Antar Muka Halaman Daftar



Gambar V.34 Antar Muka Halaman Form Tahun Ajaran

Pada form di atas maka admin dapat menambaha dan mengedit tahun ajaran sesuai dengan kebutuhan.

5.2 Pengujian

Pengujian merupakan bagian yang tidak kalah pentingnya dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangakat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu mampu merepresentasikan kajian pokok dari spesifikasi, analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri.

5.2.1 Rencana Pengujian

Proses rencana pengujian meliputi input/output. Proses pengujian input/output adalah mencoba program dengan memasukan data ke dalam form form masukan yang

telah disediakan. Pada tahap ini merupakan kelanjutan dari tahap implementasi yaitu melakukan pengujian- pengujian terhadap aplikasi yang telah dibangun. Adapun hal-hal yang akan diuji melalui teknik pengujian black box adalah sebagai berikut :

5.2.2 Kasus dan Hasil Pengujian

Berikut adalah hasil dari pengujian yang telah dilakukan :

1. Pengecekan Login

Tabel V.1 Pengujian Login

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Username: Password:	Tercantum pada textbox username dan password	Dapat mengisi login sesuai yang diharapkan	[x] Diterima [ ] Ditolak
Klik tombol submit	Login berhasil sesuai dengan hak aksesnya/ berdasarkan hak aksesnya	Tombol login dapat sesuai dengan yang diharapkan	[x] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
UserName : Password :	Tidak dapat login	User tidak dapat login	[x] Diterima

			[ ] Ditolak
Klik tombol submit	Tidak dapat masuk ke form menu utama sesuai dengan hak aksesnya	User tidak dapat login	[x] Diterima [ ] Ditolak

2. Pengecekan Pendaftaran Mahasiswa

Tabel V.2 Pengujian Pendaftaran Mahasiswa

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data Mahasiswa	Terdaftar kedalam sistem	Terdaftar kedalam sistem sesuai yang diharapkan	[x] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data Mahasiswa yang salah	Menampilkan pesan *error*	User tidak terdaftar ke sistem	[x] Diterima [ ] Ditolak

3. Pengecekan Uji Program

Tabel V.3 Pengujian Uji program

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Menekan Tombol Daftar	Terdaftar kedalam uji program	Terdaftar kedalam uji program sesuai yang diharapkan	[x] Diterima [ ] Ditolak

4. Pengecekan Pangajuan Proposal

Tabel V.4 Pengujian Pengajuan Proposal

Kasus dan Hasil Uji (Data Normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data Judul dan pemilihan pembimbing	Terdaftar dalam mengjukan proposal	Terdaftar dalam mengjukan proposal sesuai yang diharapkan	[x] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	**Kesimpulan**
Data Judul dan	Menampilkan pesan	User tidak	[x] Diterima

pemilihan pembimbing yang salah	*error*	terdaftar ke dalam pengajuan proposal	[ ] Ditolak

5.3 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji sample diatas dapat ditarik kesimpulan bahwa perangkat lunak berjalan dengan baik dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.