BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem Yang Berjalan

Analisis sistem merupakan penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan dan hambatan yang terjadi dan kebutuhan yang diharapkan dapat di usulkan perbaikan perbaikan.

Pelaksanaan uji program sistem yang berjalan pada Program Studi Sistem informasi Universitas Komputer Indonesia ini bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah yang dihadapai sistem untuk dijadikan landasan usulan perancangan analisis sistem yang berjalan yang dilakukan berdasarkan urutan kejadian yang ada pada Program Studi Sistem Informasi Universitas Komputer Indonesia.

4.1.1 Analisis Dokumen

Dalam analisis dokumen ini, akan menghasilkan beberapa dokumen yang digunakan dalam proses pengolahan data. Tujuan dari analisis dokumen adalah untuk mengetahui dokumen apa saja yang ada dalam sistem serta hal yang berkaitan dengan dokumen tersebut. Adapun dokumen-dokumen yang digunakan dalam pelaksanaan uji program pada Program Studi Sistem Informasi Universitas Komputer Indonesia, adalah :

1. Kartu Lulus Uji Program

Deskripsi : Dokument dibuat oleh Sistem Informasi Skrisi dan Tugas Akhir(SIMITA), merupakan hasil proses uji program.

Fungsi : Untuk mahasiswa sebagai tanda bukti bahwa mahasiswa telah mengikuti uji program dan dinyatakan kompeten.

Sumber : SIMITA

Rangkap : 1

4.1.2 Analisis Prosedur Yang Sedang Berjalan

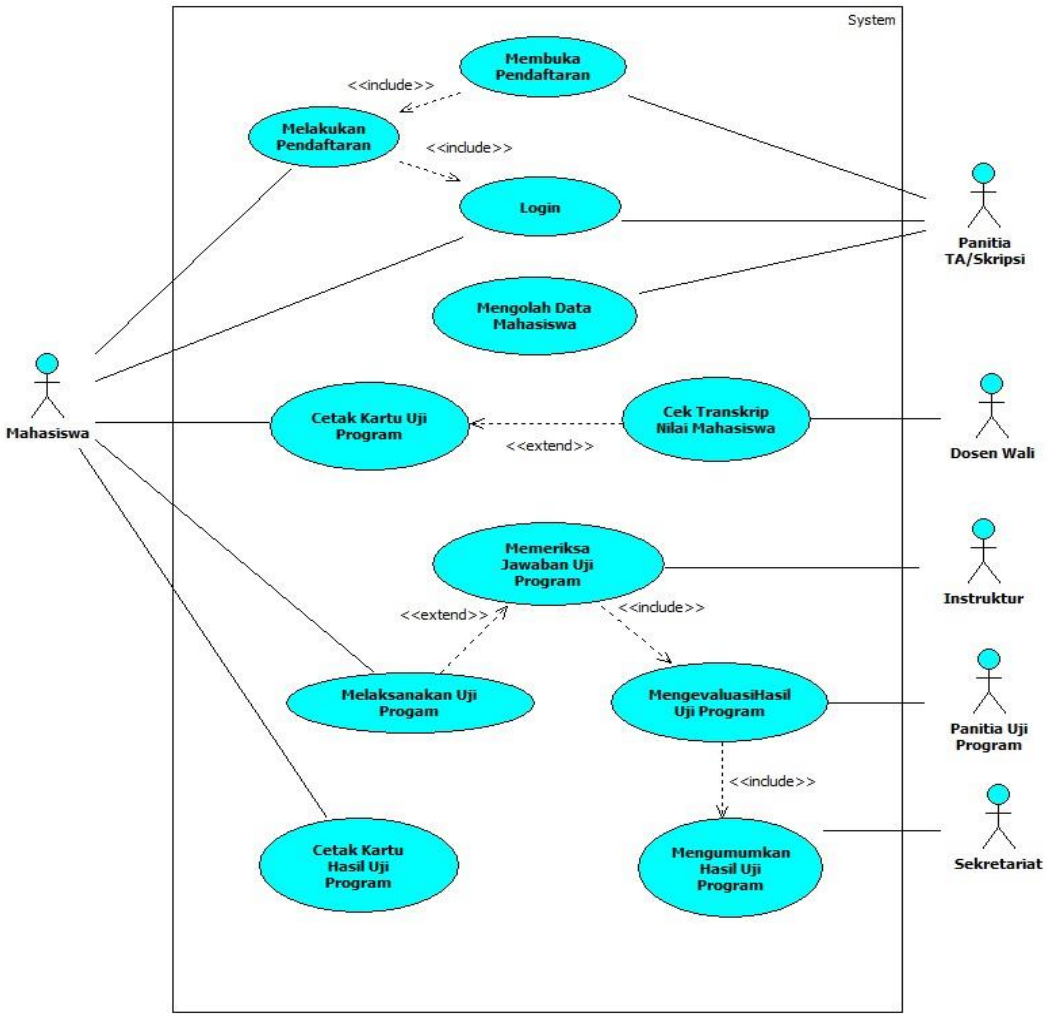
Analisis prosedur yang sedang berjalan menguraikan secara sistematis akitfitas-aktifitas yang terjadi dalam proses uji program pada Progam Studi Sistem Informasi Universitas Komputer Indonesia. Untuk menentukan kebutuhan dari pengguna yaitu degnan cara menganalisis sistem yang berjalan, kemudian mengevaluasinya. Adapun hasil dari proses analisis uji program yang sedang berjalan adalah sebagai berikut :

1. Panitia TA/Skirpsi membuka pendaftaran pada Sistem Informasi Skripsi TA (SIMITA).
2. Mahasiswa melakukan pendaftaran pada SIMITA yang diakses pada situs *website* *http://is.unikom.ac.id/simita/*.
3. Panitia Tugas Akhir dan Skripsi mengolah data mahasiswa yang sudah mendaftar pada SIMITA.
4. Dosen wali mengecek transkrip nilai mahasiswa untuk mengikuti uji program sebelum mengajukan Tugas Akhir dan Skripsi.

5. Mahasiswa mencetak kartu uji program pada SIMITA untuk mengikuti uji program.
6. Mahasiswa melakukan uji program sesuai jadwal, jenis ujian yang sudah dipilih dan ditentukan oleh panitia uji program.
7. Instruktur memeriksa jawaban uji program yang sudah dijawab oleh mahasiswa.
8. Panitia uji program mengevaluasi hasil uji program mahasiswa.
9. Sekretariat mengumumkan hasil uji program mahasiswa.
10. Mahasiswa menerima hasil uji program yang sudah valid.
11. Mahasiswa mencetak kartu lulus uji program.

4.1.2.1 *Use Case Diagram* Sistem Yang Berjalan

Use case diagram mendeskripsikan fungsi dari sebuah sistem dari perspektif pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan system disebut *scenario.*



Gambar 4.1 *Use Case Diagram* sistem yang sedang berjalan

4.1.2.2 Skenario *Use case* yang sedang berjalan

Skenario *Use Case* digunakan untuk memudahkan dalam menganalisa scenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario.

1. Nama *Use Case* : Membuka pendaftaran

   Aktor : Panitia TA/Skripsi

   Tujuan : *Use case* ini bertujuan untuk membuka pendaftaran pada SIMITA yang ditujukan untuk mahasiswa yang akan

megikuti uji program sebelum mengambil skripsi atau tugas akhir.

Tabel 4.1 Skenario *Use Case* Membuka pendaftaran yang sedang berjalan

Aksi Aktor	Reaksi Sistem
1. Membuka pendaftaran pada SIMITA.	
	2. Pendafttaran dibuka untuk mahasiswa.

2. Nama *Use Case* : Melakukan pendaftaran

   Aktor : Mahasiswa

   Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa mahasiswa melakukan pendaftara pada situs website *http://is.unikom.ac.id/simita/*

Tabel 4.2 Skenario *Use Case* Melakukan pendaftaran

Aksi Aktor	Reaksi Sistem
1. Aktor membuka situs *website pada* *http://is.unikom.ac.id/simita/.*	
2. Aktor memilih menu pendaftara pada *website* *http://is.unikom.ac.id/simita/.*	

	3. Sistem menyimpan data mahasiswa sebagai pada *database.*

3. Nama *Use Case* : Login

   Aktor : Mahasiswa, panitia TA/Skripsi

   Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa aktor melakukan proses login pada situ *website*

Tabel 4.3 Skenario *Use Case* Login

Aksi Aktor	Reaksi Sistem
1. Aktor membuka situs *website* *http://is.unikom.ac.id/simita/.*	
	2. Sistem memverifikasi *username* dan *password* yang sudah terdaftar.
3. Aktor masuk pada halaman *website.*	

4. Nama *Use Case* : Mengolah data mahasiswa

   Aktor : Panitia TA/Skripsi

   Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa aktor mempunyai tugas untuk mengolah data siapa saja mahasiswa yang sudah mendaftar pada SIMITA

Tabel 4.4 Skenario *Use Case* Mengolah data peserta

Aksi Aktor	Reaksi Sistem
1. Mendapatkan data peserta.	
2. Membuat penjadwalan peserta.	
	3. Menampilkan jadwal uji program.

5. Nama *Use Case* : Cek transkrip nilai mahasiswa

   Aktor : Dosen wali

   Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa dosen wali mengecek transkrip nilai mahasiswa sebagai salah satu syarat mengikuti uji program sebelum mengambil TA/Skripsi

Tabel 4.5 Skenario *Use Case* Cek transkrip nilai mahasiswa

Aksi Aktor	Reaksi Sistem
1. Aktor memeriksa transkrip nilai mahasiswa.	
	2. Menerima transkrip nilai mahasiswa valid.

6. Nama *Use Case* : Cetak kartu uji program

   Aktor : Mahasiswa

Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa mahasiswa melakukan proses pencetakan kartu uji program dengan cara mendownloadnya terlebih dahulu

Tabel 4.6 Skenario *Use Case* Cetak kartu uji program

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu download kartu uji program pada SIMITA.	
	2. Sistem menerima perintah aktor untuk mendownload kartu uji program.
3. Menerima kartu uji program kemudian di cetak,	

7. Nama *Use Case* : Melaksanakan uji program

   Aktor : Mahasiswa

   Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa mahasiswa melaksanakan proses uji program

Tabel 4.7 Skenario *Use Case* melaksanakan uji program

Aksi Aktor	Reaksi Sistem
1. Aktor menandatangani kehadiran mengikuti uji program	

2. Aktor Mengisi identitas atau biodata mahasiswa pada lembar jawaban.	
3. Aktor Menjawab soal uji program yang berbetnuk essay pada lembar jawaban.	
	4. Menerima jawaban uji program yang sudah diisi atau dilakukan.

8. Nama *Use Case* : Memeriksa jawaban hasil uji program

   Aktor : Instruktur

   Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa instruktur mengawas dan memeriksa jawaba uji program mahasiswa

Tabel 4.8 Skenario *Use Case* Memeriksa jawaban hasil uji program

Aksi Aktor	Reaksi Sistem
1. Aktor Memerikasi jawaban uji program mahasiswa	
2. Instruktur menilai jawaban uji program mahasiswa	
	3. Mendapatkan hasil nilai uji program sementara

9. Nama *Use Case* : Mengevaluasi hasil uji program

   Aktor : Panitia uji program

   Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa nilai hasil uji program yang dilakukan oleh mahasiswa dievaluasi oleh panitia TA/Skripsi agar dapat diketahui kompeten atau tidak.

Tabel 4.9 Skenario *Use Case* Mengevaluasi hasil uji program

Aksi Aktor	Reaksi Sistem
1. Aktor melakukan evaluasi terhadap hasil nilai uji program mahawasiswa	
	2. Sistem mendapatkan hasil keputusan evaluasi nilai uji program mahasiswa

10. Nama *Use Case* : Mengumumkan hasil nilai uji program

    Aktor : Sekretariat jurusan

    Tujuan : *Use case* ini bertujuan untuk menggambarkan bahwa hasil nilai ujian program yang sudah valid diinformasikan kepada mahasiswa

Tabel 4.10 Skenario *Use Case* Mengumumkan hasil nilai uji program

Aksi Aktor	Reaksi Sistem

1. Aktor menerima hasil nilai uji program peserta	
2. Akor mengumumkan hasil nilai uji program peserta	
	3. Menampilkan hasil nilai uji program peserta

11. Nama *Use Case* : Cetak kartu hasil uji program

    Aktor : Mahasiswa

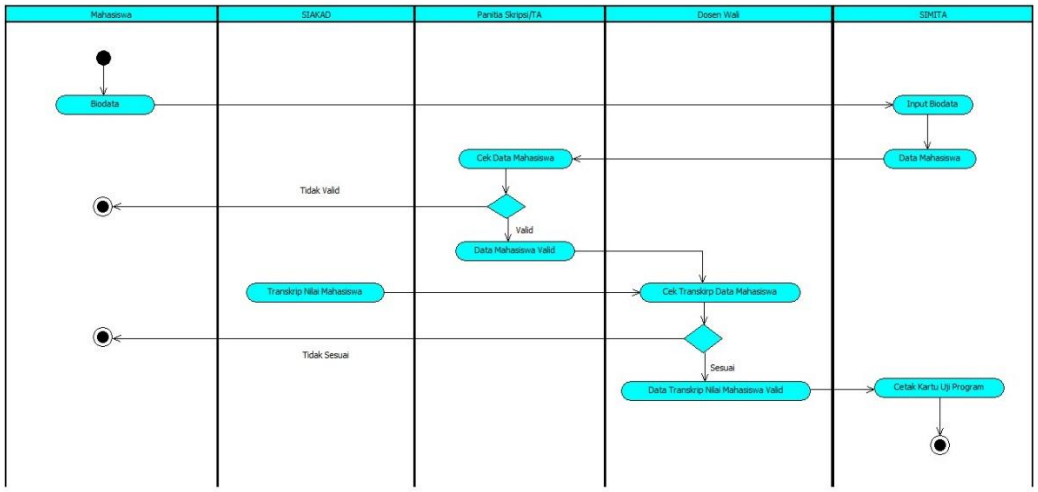
    Tujuan : *Use case* ini bertujuan untuk menggambarkan mahasiswa melakukan pencetakan kartu hasil uji program bahwa mahasiswa telah mengikuti uji program dan dinyatakan lulus atau kompeten.

Tabel 4.11 Skenario *Use Case* Cetak kartu hasil uji program

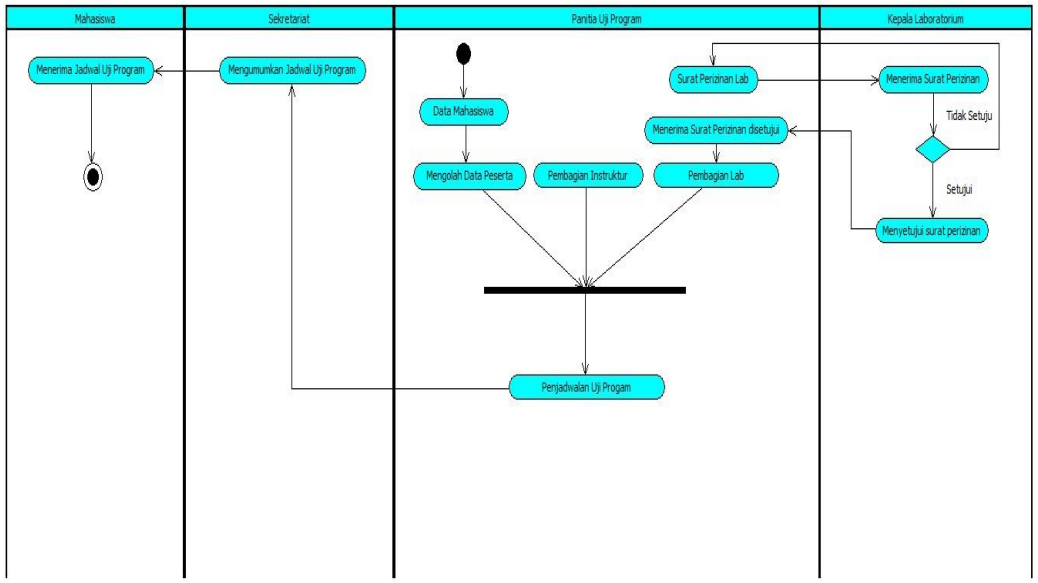
Aksi Aktor	Reaksi Sistem
1. Aktor login melalui SIMITA untuk pencetakan kartu	
	2. Memverifikasi *login* aktor
3. Aktor mencetak atau mendownload kartu hasil uji program	

4.1.2.3 *Activity Diagram* yang Berjalan

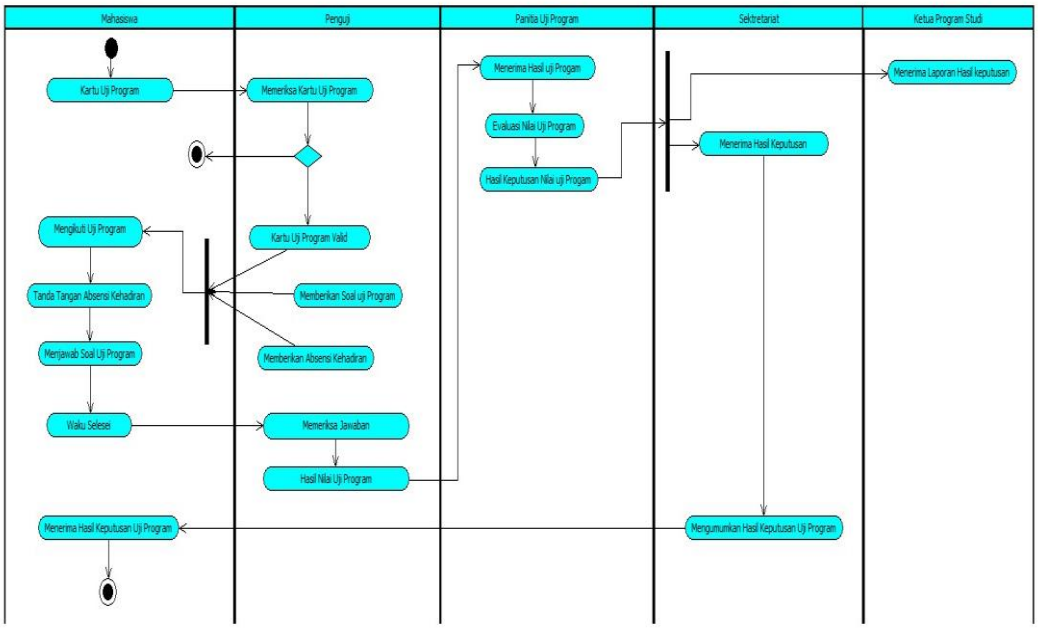
Berikut ini adalah *Activity Diagram* yang menggambarkan aktivitas yang terjadi pada sistem yang berjalan, *Activity Diagram* ini mendeskripsikan logikal prosedural, porses bisnis dan aliran kerja dalam banyak kasus.



Gambar 4.2 *Activity Diagram* pendaftaran uji program yang sedang berjalan



Gambar 4.3 *Activity Diagram* pengolahan data dan penjadwalan uji program yang sedang berjalan



Gambar 4.4 *Activity Diagram* pelaksanaan uji program yang sedang berjalan

4.1.3 Evaluasi Sistem yang Berjalan

Dari hasil analisis sistem yang sedang berjalan, bahwa kegiatan dari proses pelaksanaan dan penilaian uji program masih memiliki kekurangan dan kelemahan antara lain sebagai berikut :

Tabel 4. 12 Evaluasi Sistem yang Sedang Berjalan

No.	Permasalahan	No.	Perancangan yang diusulkan / Solusi
1.	Pekaksanaan uji program yang dilaksanakan masih manual walaupun sudah menggunakan komputer.	1.	Membuat suatu *website* ujian kompetensi berbasis intranet dimana *website* tersebut merupakan sarana untuk mengukur kompetensi mahasiswa secara konsep dan teori.

2.	Dalam proses melakukan penliain uji program, penilaian yang dilakukan masih belum objektif		Membuat soal uji program dengan sistem *multiple choice* (pilihan ganda) pada sebuah *website* intranet dan jawaban yang telah dipilih oleh peserta akan dinilai oleh sistem dengan mencocokan jawaban peserta dengan kunci jawaban yang sudah dibuat dan disimpan pada database.

Evaluasi sistem ini dibuat untuk mengetahui permasalahan yang timbul, meminimalisasi kekurangan dan kelemahan pada sistem yang sedang berjalan sehingga diharapkan dengan adanya perancangan sistem ini dapat membantu masalah yang ada pada sistem.

4.2 Perancangan Sistem

Perancangan sistem dibuat sebagai tahapan untuk mempersiapkan proses implementasi sistem yang diinginkan, dan untuk menggambarkan secara jelas proses-proses sesuai kebutuhan. Dengan metode pendekatan yang digunakan adalah metode pendekatan berorientasi objek aka metode yang digunakan untuk menggambarkan seluruh proses dan objek adalah dengan menggunakan *Unified Model Language.*

Adapun perancangan sistem ini meliputi :

1. Perancangan *Use Case Diagram* dan skenario
2. Perancagnan *Activity Diagram*
3. Perancangan *Sequence Diagram*
4. Perancangan *Class Diatram*

5. Perancangan *Component Diagram*

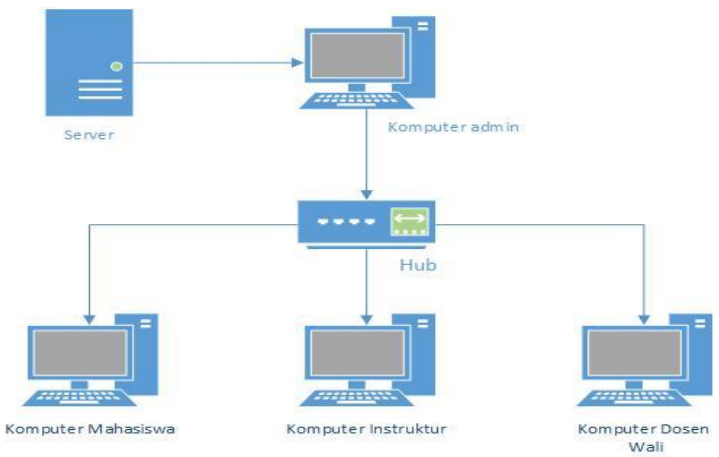
6. Perancangan *Deployment Diagram*

4.2.1 Tujuan Perancangan Sistem

Tujuan perancangan sistem adalah merancang Sistem Informasi Penilaian Uji Kompetensi Berbasis *Intranet* dengan mengguanakan bahasa pemrograman PHP dan MySql sebagai *database* (tempat penyimpanan data). Sistem ini dibuat untuk membantu proses pelaksanaan uji kompetensi, penilaian serta laporan hasil nilai para peserta atau mahasiswa yang mengikuti uji kompetensi.

4.2.2 Gambaran Umum Sistem yang Diusulkan

Website yang akan dibangun memiliki 2 tipe *user* dimana *user* tersebut adalah perserta serta instruktur serbagai administrator. Administrator atau instruktur disini memegang akses penuh terhadap penginputan soal-soal uji kompetesi, jawaban, pengelolaan data peserta, waktu pengerjaan serta pencetakan laporan hasil uji kompetensi yang sudah dilaksanakan oleh perserta atau mahasiswa. Sedangkan perserta atau mahasiswa mempunyai peranan terhadap pemilihan uji kompetensi apa yang akan dipilih, dan pemilihan jawaban terhadap soal-soal yang diujikan. Adapaun skema atau gambaran umum dari aplikasi yang akan dibangun :



Gambar 4.5 Gambaran umum sistem yang di usulkan

4.2.3 Perancangan Prosedur Sistem Yang Diusulkan

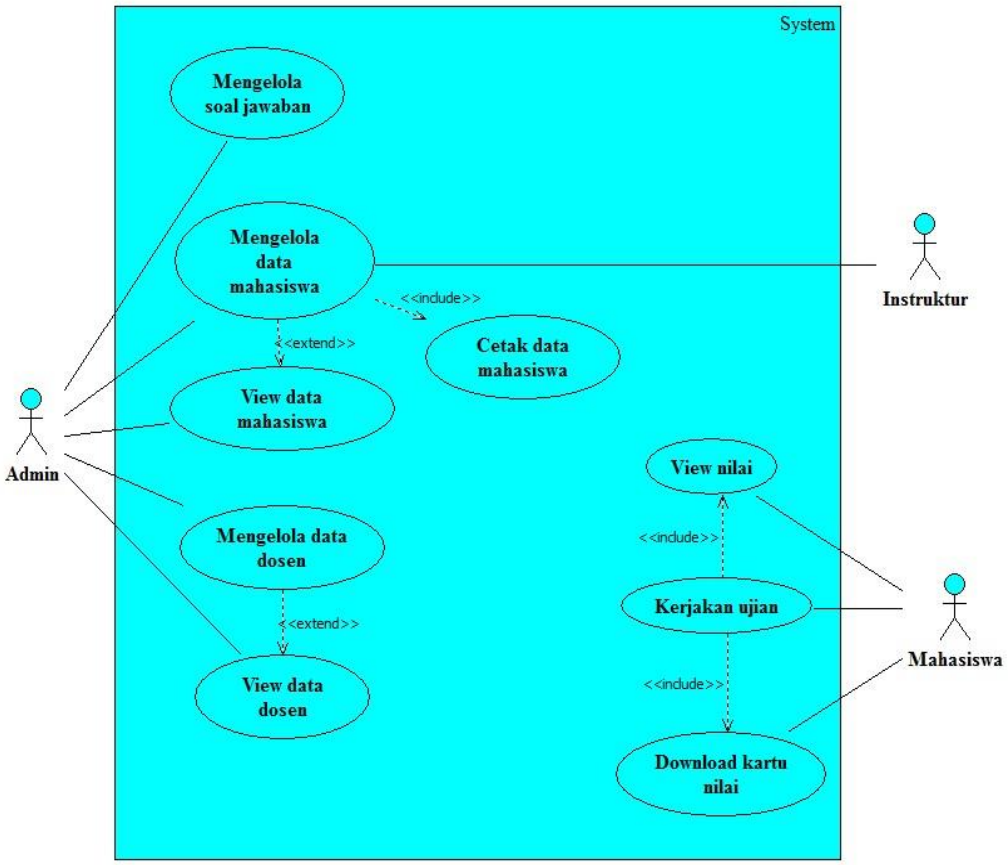
Berdasarkan hasil analisis yang dilakukan proses pendaftaran yang sedang berjalan sudah berjalan dengan baik, adapun perancangan prosedur sistem yang diusulkan yaitu pada proses pelaksanaan dan penilaiaan uji program yang dianggap masih manual dan bisa dimaksimalkan proses kerjanya dengan komputerisasi, berikut perancangan prosedur sistem yang diusulkan adalah :

1. Admin menyiapkan atau menginput dan menyimpan soal dan jawaban yang akan diujiankan ke dalam sistem yang diusulkan.
2. Panitia TA/Skirpsi membuka pendaftaran pada Sistem Informasi Skripsi TA (SIMITA).
3. Mahasiswa melakukan pendaftaran pada SIMITA yang diakses pada situs *website* *http://is.unikom.ac.id/simita/*.
4. Admin mengimport database data mahasiswa yang sudah melakukan pendaftaran dari SIMITA ke dalam sistem yang diusulkan.
5. Dosen wali menyeleksi data mahasiswa yang berhak mengikuti uji kompetensi berdasarkan jumlah sks yang sudah diikuti.
6. Admin tidak mengelola penjadwalan uji kompetensi.
7. Mahasiswa melakukan uji program sesuai jadwal, jenis ujian yang sudah dipilih dan ditentukan oleh panitia uji program.
8. Instruktur memberikan pengarahan sebelum mahasiswa menjawab soal uji kompetensi.

9. Mahasiswa menjawab soal-soal yang sudah disediakan pada sistem.
10. Mahasiswa mengetahui nilai hasil uji kompetensi sementara.
11. Mahasiswa mencetak nilai hasil uji kompetensi sementara.
12. Panitia uji program mengevaluasi hasil uji kompetensi yang sudah dilaksanakan oleh mahasiswa.
13. Mahasiswa menerima hasil uji program yang sudah valid.
14. Mahasiswa mencetak kartu lulus uji program pada SIMITA.

4.2.3.1 *Use Case Diagram* yang Diusulkan

Diagram *use case* ini memperlihatkan pada kita hubungan-hubungan yang terjadi antara actor-aktor dengan *use case* dalam sistem. adapun *use case diagram* yang diusulkan adalah sebagai berikut :



Gambar 4.6 *Use case* pelaksanaan dan penilaian uji kompetensi yang diusulkan

Definisi Aktor

Definisi actor dibuat dalam bentuk table sebagai berikut :

Tabel 4.13 Definisi Aktor

No.	Aktor	Deskripsi
1.	Admin	Admin mempunyai hak akses penuh atas pengelolaan sistem, dari pengelolaan soal jawaban, pengelolaan data mahasiswa, pengelolaan dosen, pencetakan data mahasiswa, dosen, dan hasil ujian.
2.	Mahasiswa	Mahasiswa mempunyai hak akses untuk mengerjakan ujian dan juga melakukan pencetakan kartu nilai hasil uji kompetensi sementara.
3.	Instruktur	Instruktur selain mengawas juga diberikan hak akses untuk mengelola data mahasiswa.

4.2.3.2 Skenario *Use case* yang diusulkan

1. Nama *Use Case* : Mengelola soal jawaban

   Aktor : Admin

   Tujuan : *Use case* ini menggambarkan bahwa aktor mengelola soal dan jawaban.

Tabel 4.14 Skenario *Use Case* mengelola soal jawaban

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu soal.	
2. Menginput soal dan jawaban yang sudah disepakati oleh panitia Skripsi dan TA pada form yang disediakan.	
3. Aktor memilih tombol simpan pada sistem.	
	4. Menyimpan soal dan jawaban pada database yang telah diinput oleh admin.

2. Nama *Use Case* : Mengelola data mahasiswa

   Aktor : Admin, instruktur

   Tujuan : *Use case* ini menggambarkan bahwa aktor mengelola data mahasiswa yang akan mengikuti uji kompen dan sudah mendaftar pada SIMITA.

Tabel 4.15 Skenario *Use Case* Mengelola data mahasiswa

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu daftar mahasiswa.	

2. Aktor mengimport database data pendaftaran mahasiswa dari SIMITA.	
	3. Menyimpan data mahasiswa yang sudah di import ke dalam database.

3. Nama *Use Case* : View data mahasiswa

   Aktor : Admin

   Tujuan : *Use case* ini berfungsi sebagai validasi atas inputan yang sudah dilakukan oleh admin. Oleh karenanya pada *use case* diagram diatas. *View* data mahasiswa ini adalah perluasan dari *use case* mengelola data mahasiswa.

Tabel 4.16 Skenario *Use Case* View data mahasiswa

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu tampilkan data mahasiswa.	
	2. Sistem menampilkan data mahasiswa.

4. Nama *Use Case* : Cetak data mahasiswa

   Aktor : Admin

Tujuan : *Use case* ini menggambarkan bahwa aktor melakukan pencetakan data mahasiswa yang sudah terdaftar pada sistem.

Tabel 4.17 Skenario *Use Case* Cetak data mahasiswa

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu cetak data mahasiswa pada sistem.	
	2. Sistem menampilkan preview data mahasiswa untuk di cetak.
3. Aktor memilih tombol cetak atau print pada sistem	

5. Nama *Use Case* : Mengelola data instruktur

   Aktor : Admin

   Tujuan : *Use case* ini menggambarkan bahwa aktor mengelola data instruktur seperti menambahm mengedit, ataupun menghapus.

Tabel 4.18 Skenario *Use Case* Mengelola data instruktur

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu daftar dosen. Menginput data dosen wali.	

	2. Sistem menampilkan form input data instruktur.
3. Aktor menginput data instruktur wali pada form yang sudah disediakan.	
	4. Aktor memilih tombol simpan pada sistem.
5. Sistem menyimpan data dosen wali kedalam database.	

6. Nama *Use Case* : View data instruktur

   Aktor : admin

   Tujuan : *Use case* ini berfungsi sebagai validasi atas inputan yang sudah dilakukan oleh admin. Oleh karenanya pada *use case* diagram diatas. *View* data dosen wali ini adalah perluasan dari *use case* mengelola data dosen.

Tabel 4.19 Skenario *Use Case* View data instruktur

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu tampilkan data dosen	
	2. Sistem menampilkan data dosen

7. Nama *Use Case* : Kerjakan ujian

   Aktor : Mahasiswa

   Tujuan : *Use case* ini menggambarkan bahwa aktor mengerjakan soal-soal yang sudah disediakan oleh sistem.

Tabel 4.20 Skenario *Use Case* kerjakan ujian

Aksi Aktor	Reaksi Sistem
1. Aktor memilih jawaban yang benar pada sistem.	
2. Aktor akan memilih tombol koreksi dan simpan pada sistem jika sudah selesei mengerjakan soal yang ada pada sistem.	
	3. Sistem akan menampilkan nilai, jumlah yang benar dan jumlah soal pada sistem.

8. Nama *Use Case* : View nilai

   Aktor : Mahasiswa

   Tujuan : *Use case* ini berfungsi sebagai validasi atas pemilihan jawaban yang sudah dilakukan oleh mahasiswa. Oleh karenanya pada *use case* diagram diatas. *View* nilai ini adalah perluasan dari *use case* kerjakan ujian.

Tabel 4.21 Skenario *Use Case* view nilai

Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu tampilkan data dosen	
	2. Sistem menampilkan data dosen

9. Nama *Use Case* : Download kartu nilai

   Aktor : Mahasiswa

   Tujuan : *Use case* ini menggambarkan bahwa aktor mendownload kartu nilai hasil uji kompetensi sementara.

Tabel 4.22 Skenario *Use Case* Pilih jawaban

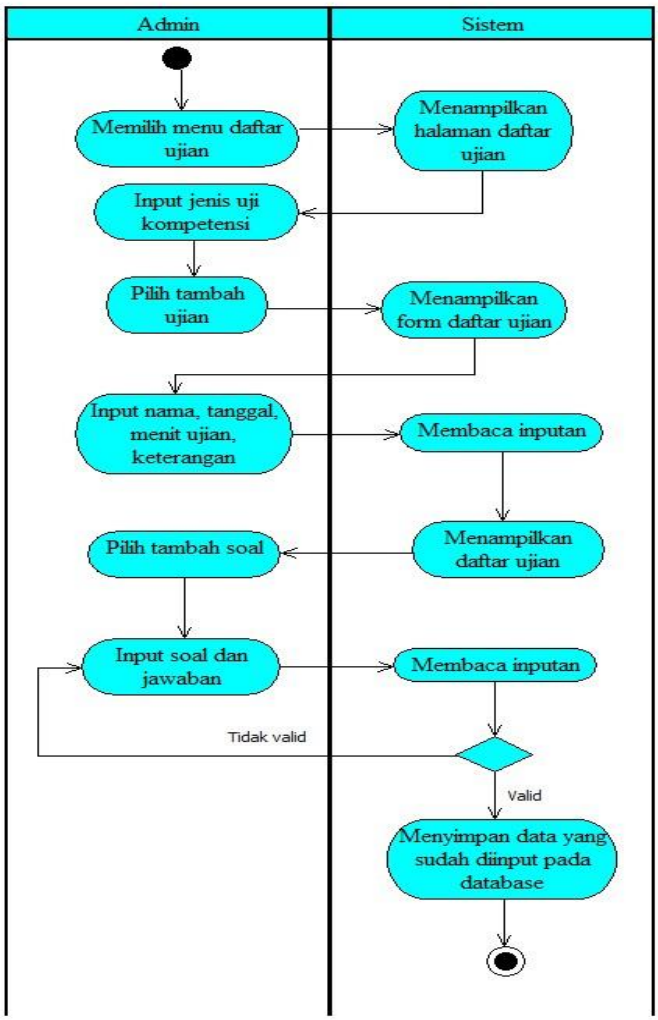
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu cetak kartu nilai hasil uji kompetensi sementara.	
	2. Sistem menampilkan preview data mahasiswa untuk di cetak.
3. Mahasiswa memilih download pada sistem.	

4.2.3.3 *Activity diagram* yang diusulkan

Activity diagram adalah teknik untuk mendeskripsikan logika prosedural, proses bisinis dan aliran kerja dalam banyak kasus.

4.2.3.3.1 *Activity diagram* mengelola soal

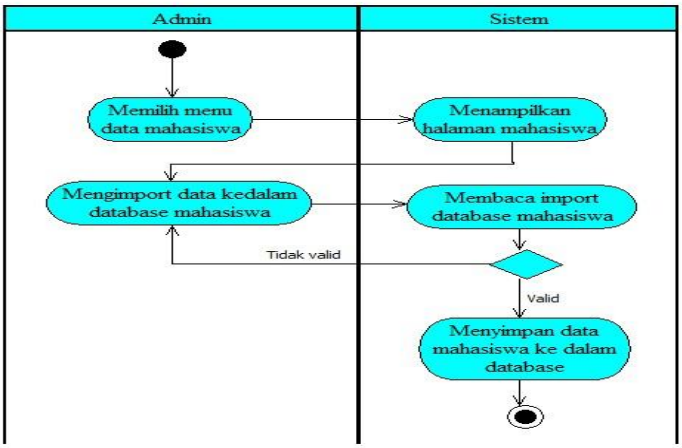
Admin melakukan input soal dan jawaban pada menu soal yang terdapat pada sistem, kemudian soal yang sudah di input di simpan pada sebuah database yang sudah dibuat.



Gambar 4.7 *Activity diagram* mengelola soal

4.2.3.3.2 *Activity diagram* mengelola data mahasiswa

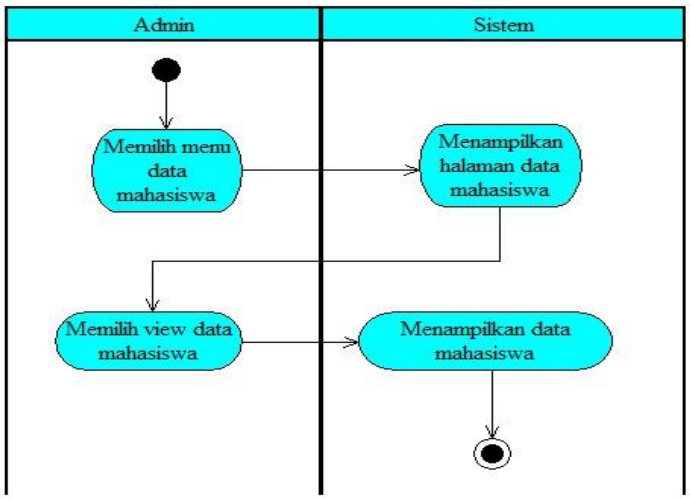
Admin melakukan import database data mahasiswa yang sudah mendaftar pada SIMITA.



Gambar 4.8 *Activity diagram* import data mahasiswa

4.2.3.3.3 *Activity diagram* view data mahasiswa

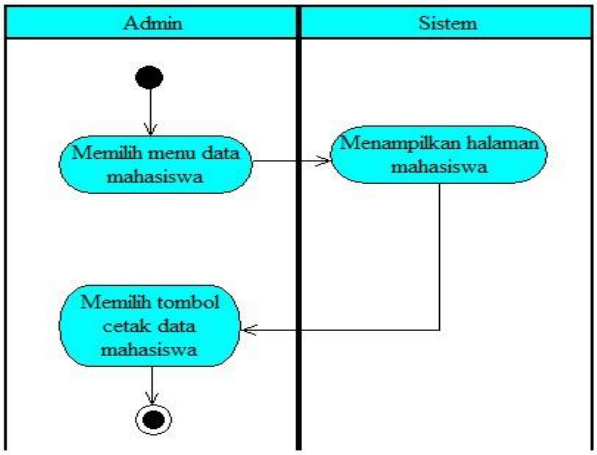
Admin melakukan view data mahasiswa yang sudah diimport agar diketahui bahwa mahasiswa sudah terdaftar pada sistem.



Gambar 4.9 *Activity diagram* view data mahasiswa

4.2.3.3.4 *Activity diagram* cetak data mahasiswa

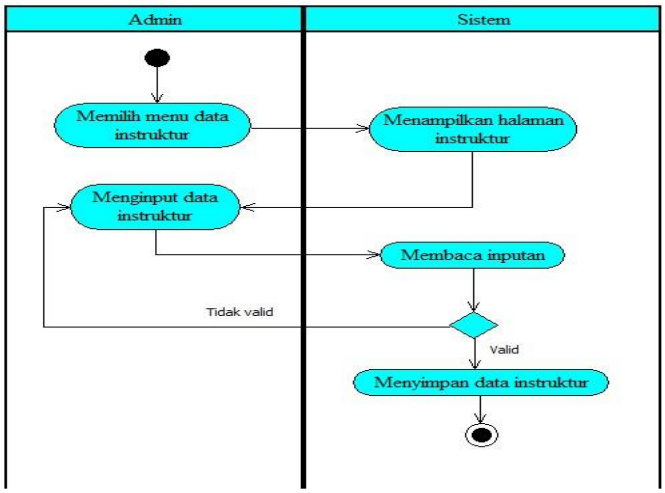
Admin melakukan cetak data mahasiswa yang sudah diimport agar diketahui bahwa mahasiswa sudah terdaftar pada sistem.



Gambar 4.10 *Activity diagram* cetak data mahasiswa

4.2.3.3.5 *Activity diagram* mengelola data instruktur

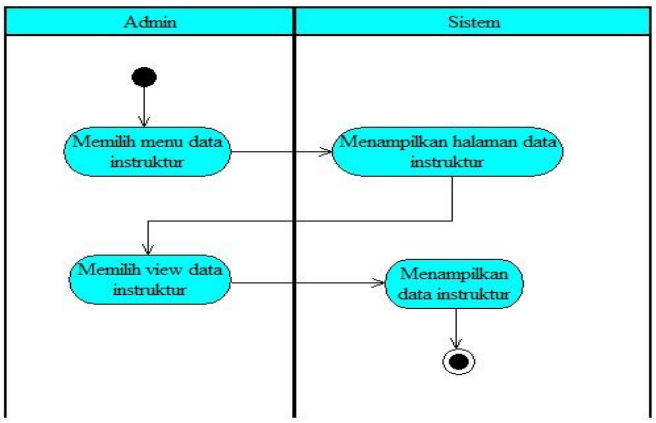
Admin melakukan input data instruktur, lalu admin menyimpan data instruktur pada *database*.



Gambar 4.11 *Activity diagram* input data instruktur

4.2.3.3.6 *Activity diagram* view data instruktur

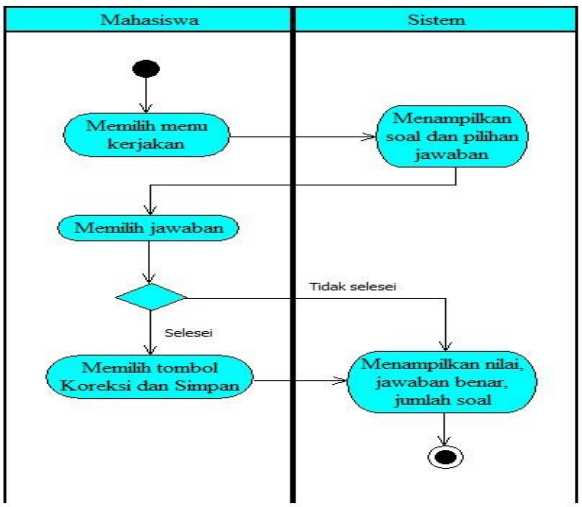
Admin melakukan view data instruktur yang sudah diimport agar diketahui bahwa dosen wali sudah terdaftar pada sistem.



Gambar 4.12 *Activity diagram view* data instruktur

4.2.3.3.7 *Activity diagram* kerjakan ujian

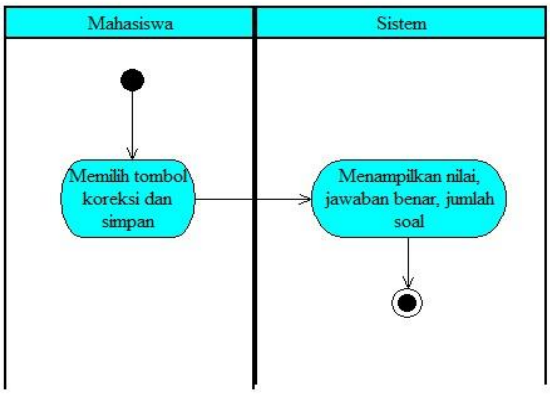
Aktor melakukan menjawab soal uji kompetensi yang akan diikuti dengan cara memilih pilihan jawaban yang sudah tersedia pada sistem.



Gambar 4.13 *Activity diagram* kerjakan ujian

4.2.3.3.8 Activity diagram view nilai

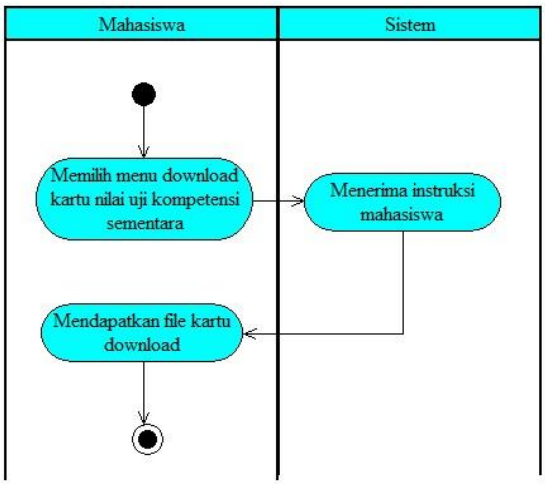
Aktor mengetahui *view* nilai mahasiswa dari hasil jawaban ujian kompetensi yang sudah diikuti.



Gambar 4.14 Activity diagram view nilai

4.2.3.3.9 Activity diagram download kartu nilai

Aktor yang sudah mengikuti ujian dan mendapatkan hasilnya, aktor mendownload kartu hasil uji kompetensi sementara yang sudah disediakan oleh sistem



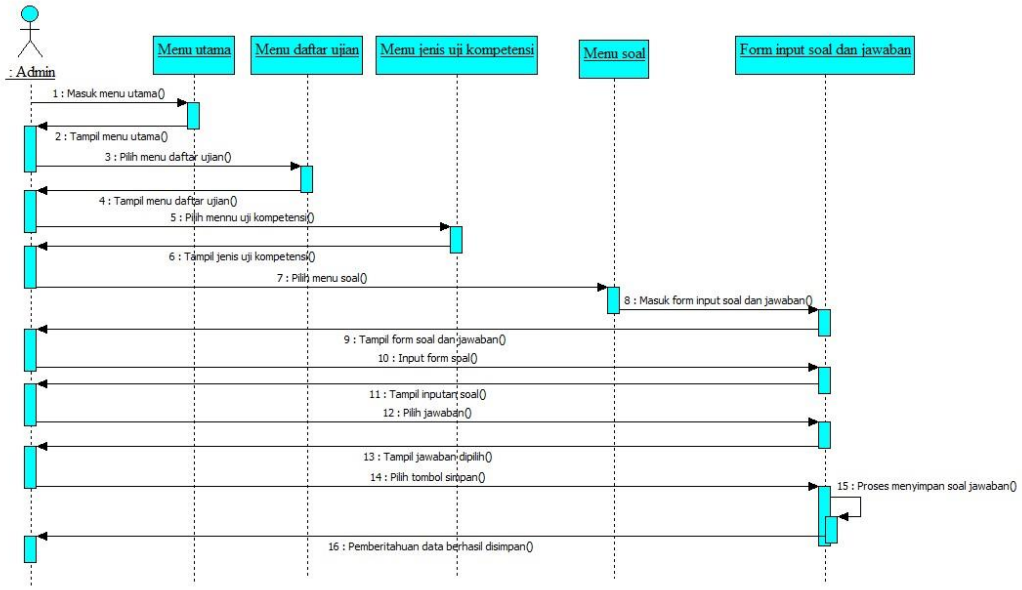
Gambar 4.15 Activity diagram download kartu

4.2.3.4 *Sequence diagram* yang diusulkan

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukan sejumlah contoh objek dam *message* (pesan) yang diletakkan dianrara objek-objek ini didalam *use case*.

4.2.3.4.1 *Sequence diagram* mengelola soal

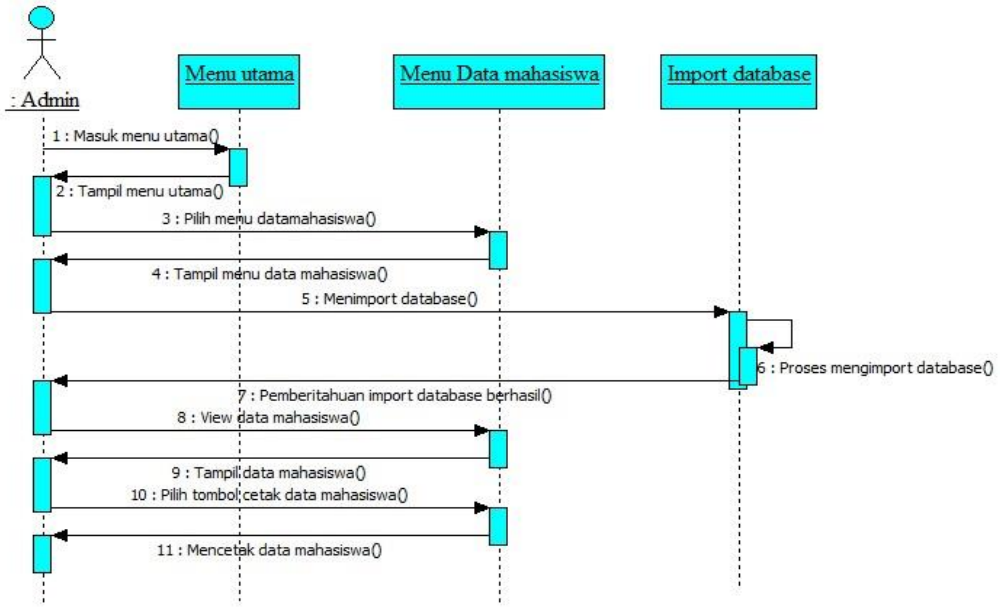
Pada *sequence diagram* mengelola soal ini admin dapat menginput, merubah, menghapus soal dan jawaban,kemudian admin menyimpan soal dan jawaban yang telah diinput. Berikut gambar *sequence diagram* input soal dan jawaban :



Gambar 4.16 *Sequence diagram* mengelola soal

4.2.3.4.2 *Sequence diagram* mengelola data mahasiswa

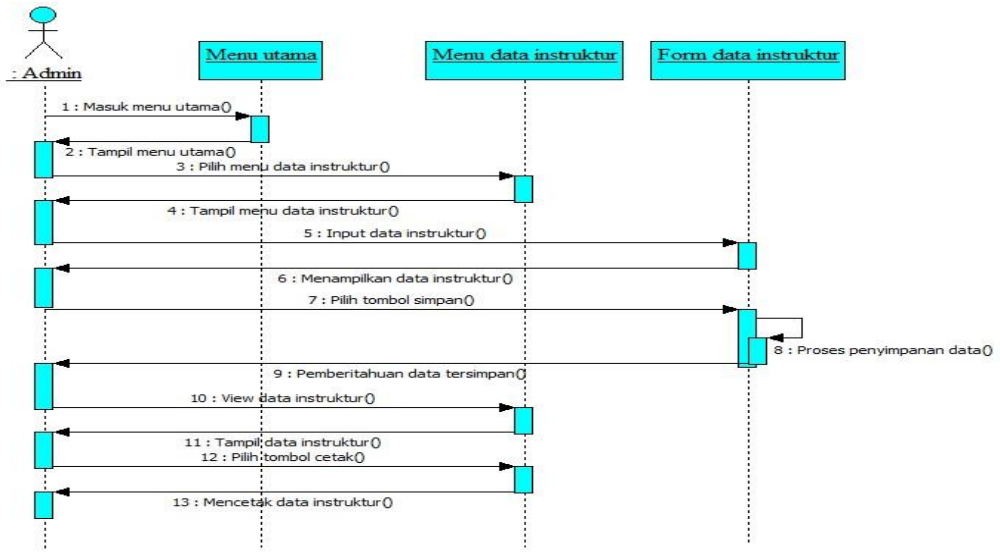
Pada *sequence diagram* mengelola data mahasiswa, admin mengimport *database* data mahasiswa yang didapat dari SIMITA. Berikut gambar *sequence diagram* import data mahasiswa :



Gambar 4.17 *Sequence diagram* mengelola data mahasiswa

4.2.3.4.3 *Sequence diagram* mengelola data instruktur

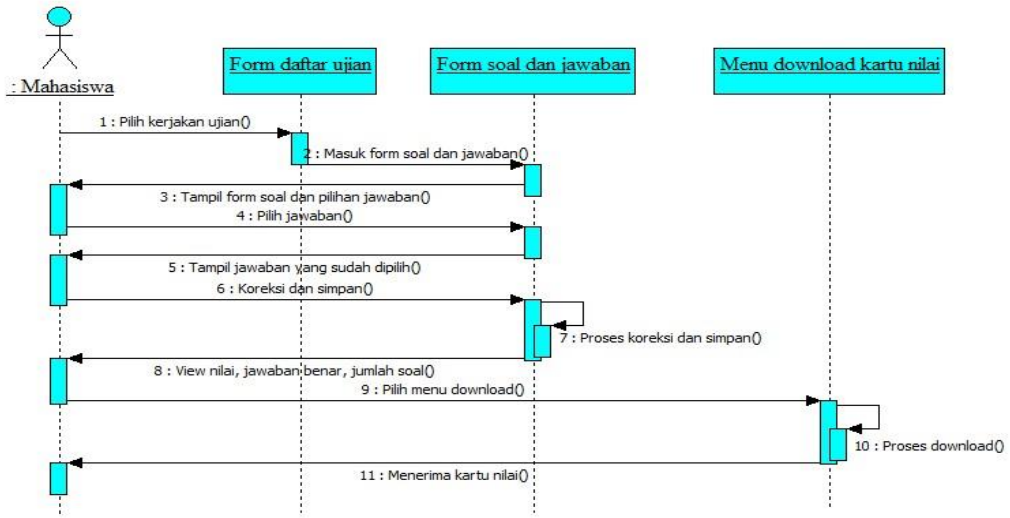
Pada *sequence diagram* input data instruktur, admin menginput data instruktur. Berikut gambar *sequence diagram* input data instruktur :



Gambar 4.18 *Sequence diagram* mengelola data instruktur

4.2.3.4.4 *Sequence diagram* kerjakan ujian

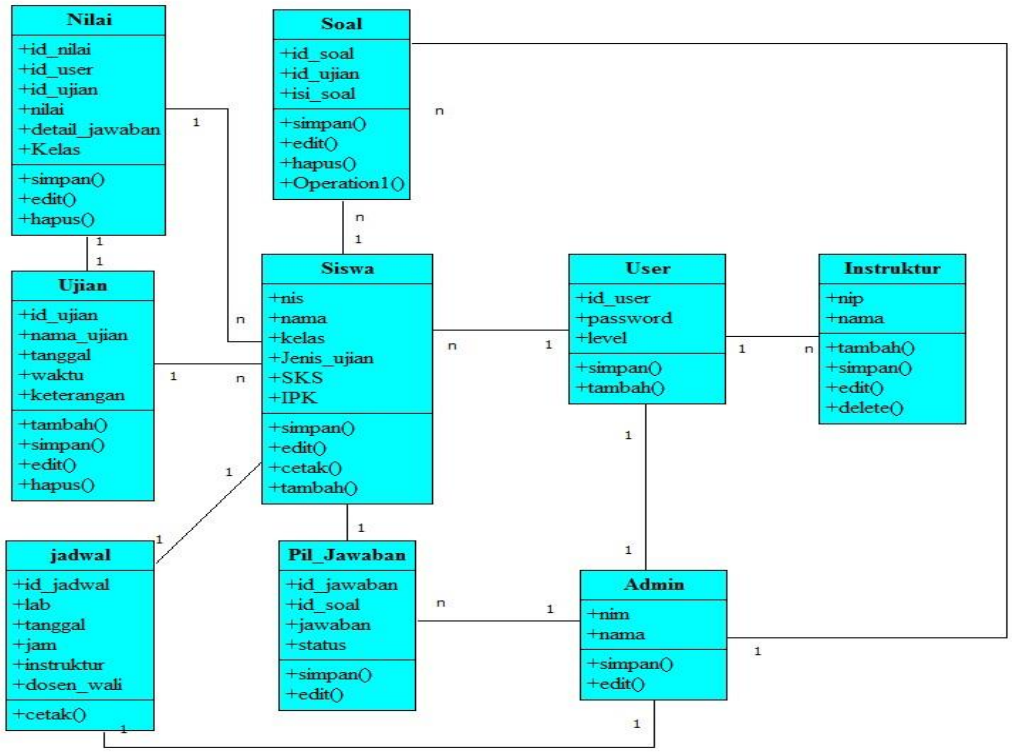
Pada *sequence diagram* kerjakan ujian, menggambarkan bahwa mahasiswa mengerjakan ujian dengan cara memilih jawaban dengan sistem *multiple choice* (pilihan ganda). Berikut gambar *sequence diagram* kerjakan ujian :



Gambar 4.19 *Sequence diagram* kerjakan ujian

4.2.3.5 *Class diagram* yang diusulkan

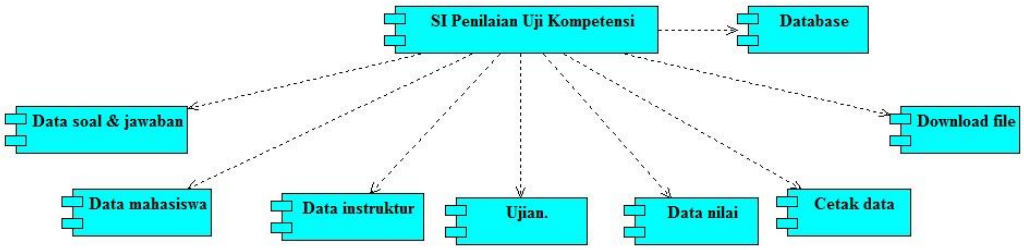
Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam sistem atau perangkat lunak yang sedang dikembangkan. Diagram kelas memberikan gambaran tentang sistem atau perangkat lunak dan relasi-relasi yang ada. Adapun *class diagram* dalam tahap perancangan sistem yang diusulkan ini adalah sebagai berikut :



Gambar 4.20 *Class diagram* uji kompetensi

4.2.3.6 Component diagram

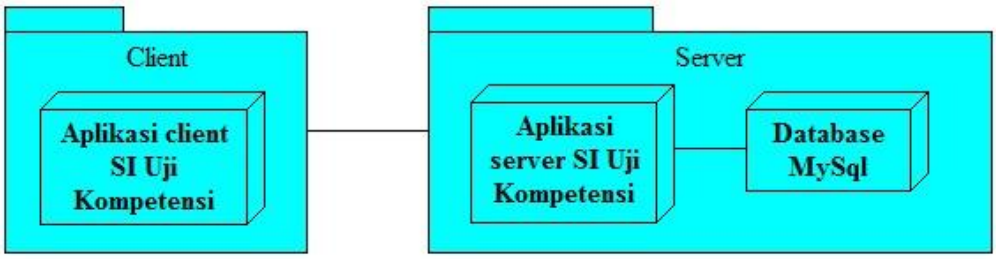
Component diagram adalah bagian fisik dari sebuah sistem, *component diagram* ini diimplementasikan meskipun pada sistem yang kecil. Perancagnan menu pada sistem informasi yang diusulkan digambarkan oleh *component diagram* sebagai berikut :



Gambar 4.21 *Component diagram* uji kompetensi

4.2.3.7 Deployment diagram

Deployment diagram menunjukan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware.*



Gambar 4.22 *Deployment diagram* uji kompetensi

4.2.4 Kodefikasi (pengkodean)

Pengkodean dibuat untuk mendefinisikan suatu objek secara singkat, dengan adanya sistem pengkodean diharapkan dapat mengklasifikasikan data, memasukan data kedalam komputer dan untuk mengambil informasi yang dibutuhkan.

1. Kode NIM

   Bentuk umum : X XX XX XXX contoh : 1-05-09-041

   Keterangan : 1 menunjukan kode fakultas

   05 menunjukan kode program studi

   09 menunjukan angkatan (tahun masuk)

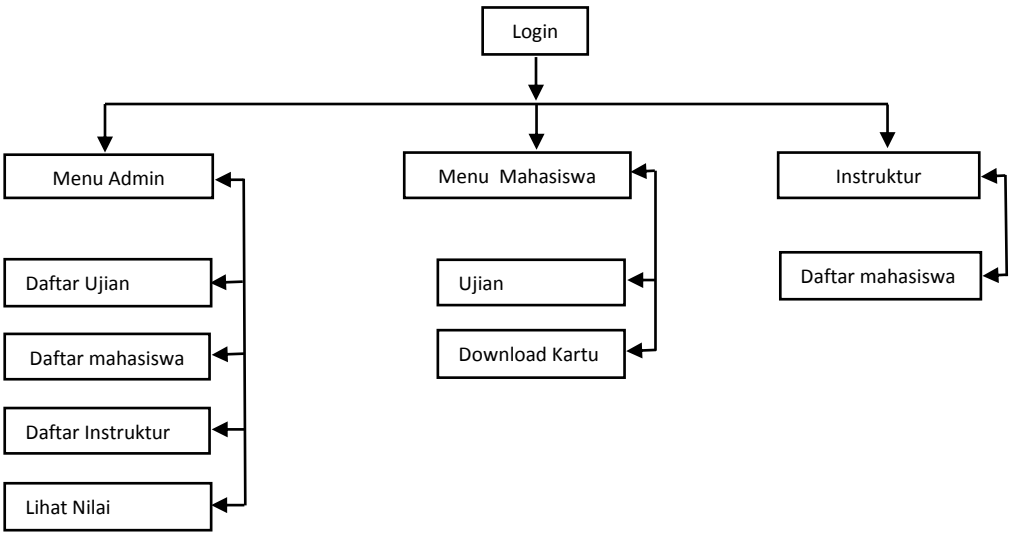
   041 menunjukan nomor urut mahasiswa

4.2.5 Perancangan antar muka

Tujuan utama dari perancangan antar muka adalah agar *interface* yang terbentuk bersifat komunikatif dan mudah digunakan oleh pemakai sehingga tujuan dari pembuatan aplikasi tercapai.

4.2.5.1 Struktur menu

Pada perancagnan struktur menu ini akan dijelaskan tentang menu-menu yang ada tersusun pada sistem yang telah dibuat ini. Berikut gambar struktur menunya :



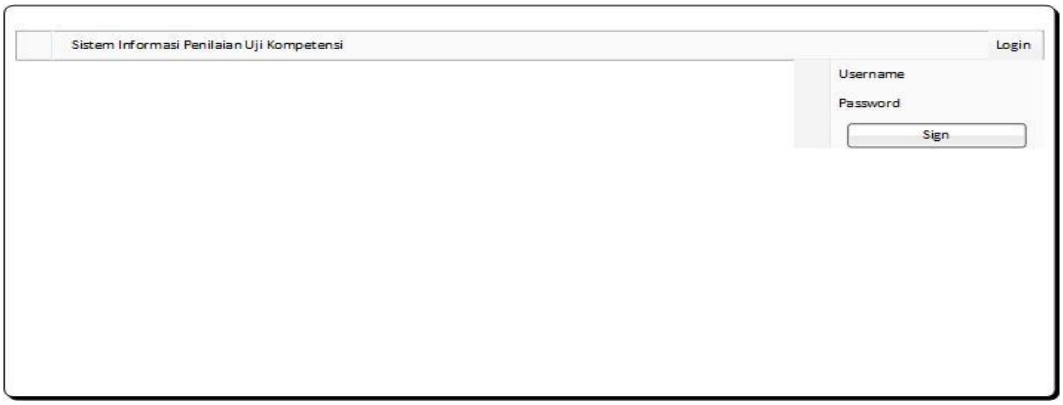
Gambar 4.23 Struktur Menu Sistem Informasi Penilaian Uji Kompetensi

4.2.5.2 Perancangan input

perancangan input yaitu sebuah tampilan yang dirancang sebagai tempat untuk memasukan data-data diberikan oleh pengguna sistem. berikut perancangan input pada sistem informasi penilaian uji kompetensi :

1. desain form login

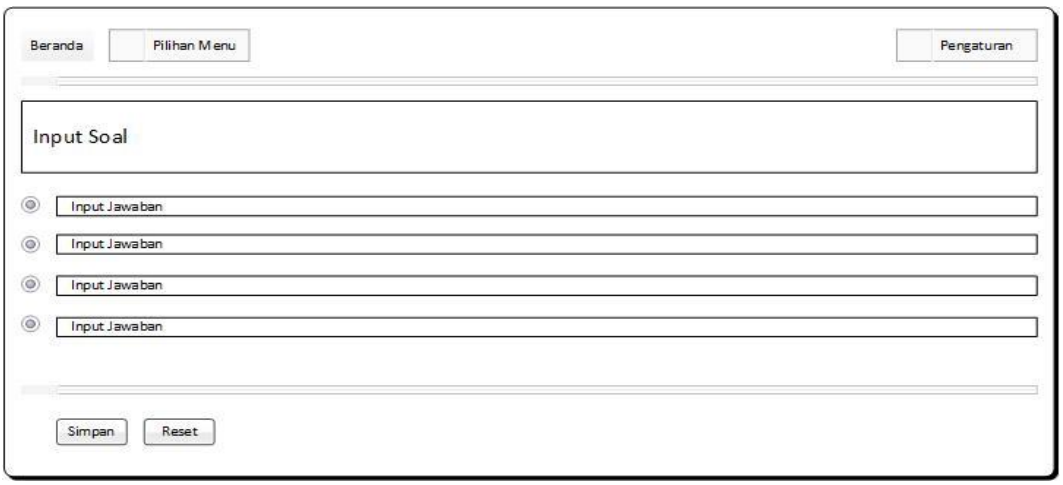
Form login pada sistem ini penggunanya yaitu admin, mahasiswa, dan dosen wali. Melalui form ini pengguna yang boleh masuk sistem adalah pengguna yang mengetahui *usename* dan *password*. Berikut perancangan antarmuka form loginn :



Gambar 4.24 Desain form Login

2. Desain form *input* soal dan jawaban

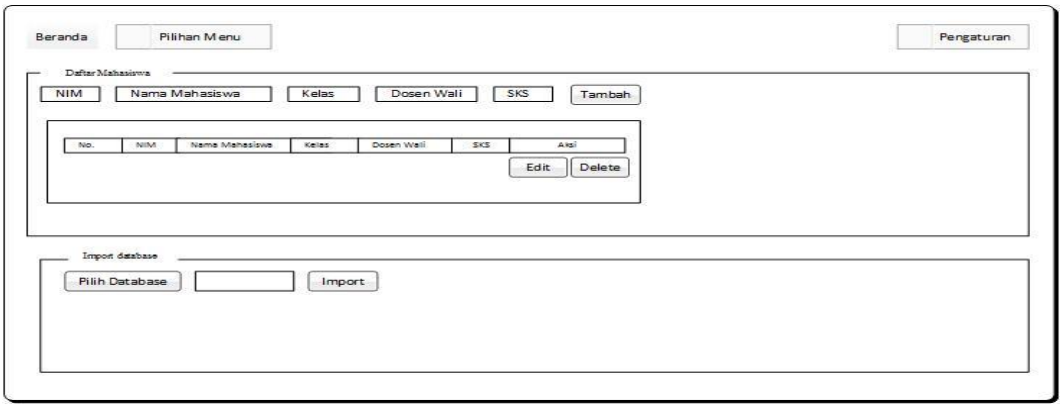
Halaman *input* soal dan jawaban ini hanya admin saja yang menginput. Didalam halaman input soal dan jawaban terdapat satu inputan untuk soal dan 4 inputan untuk jawaban serta tombol simpan dan reset. Berikut perancangannya :



Gambar 4.25 Desain form *Input* soal dan jawaban

3. Desain menu daftar mahasiswa

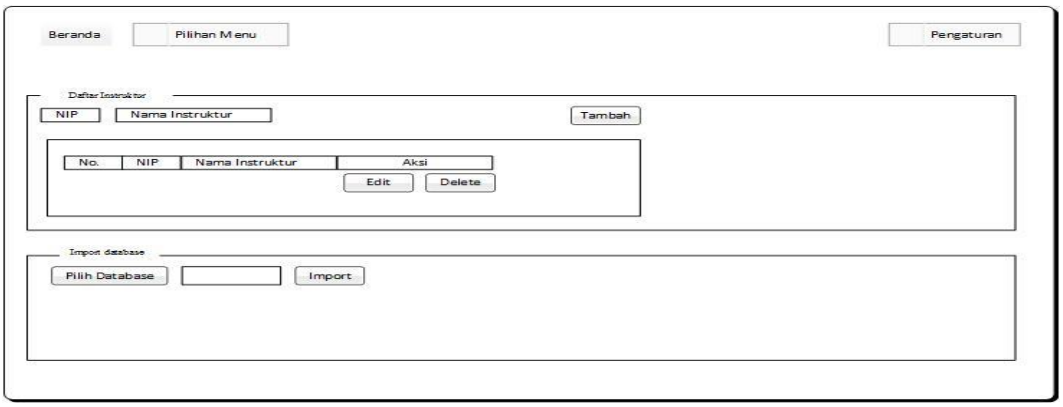
Halaman data mahasiswa ini dikelola oleh admin. Didalam halaman data mahasiswa ini selain ada halaman tambah mahasiswa terdapat import database sebagai salah satu pilihan untuk menambahkan data mahasiswa dari *database* lain dengan format *.sql*. Berikut perancanganan antarmukanya menu daftar mahasiswa:



Gambar 4.26 Desain form data mahasiswa

4. Desaim menu daftar instruktur

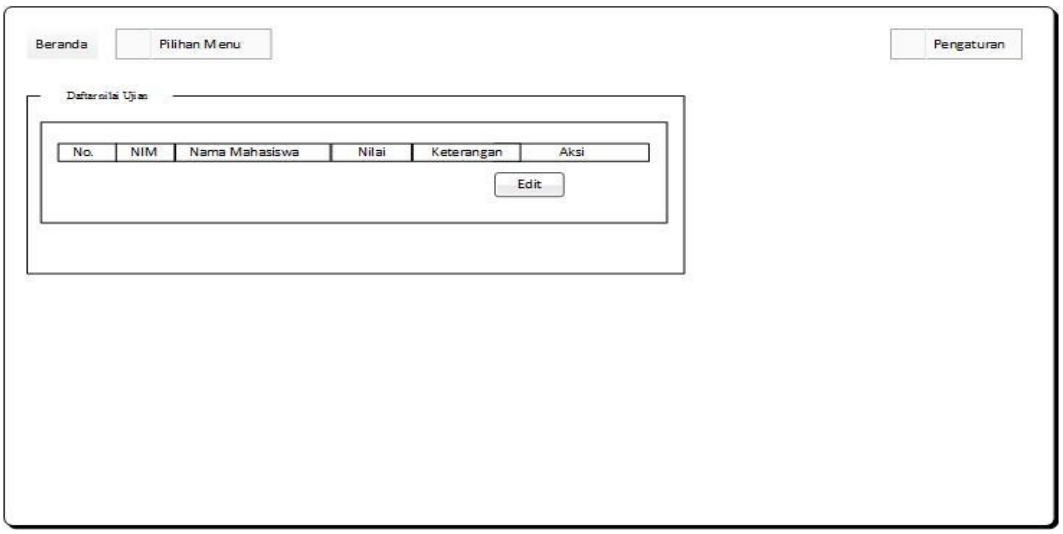
Halaman data dosen ini dikelola oleh admin. Didalam form data instruktur ini admin menginputkan data pada form yang sudah tersedia. Berukut perancanganan antarmuka menu daftar instruktur:



Gambar 4.27 Desain menu daftar instruktur

5. Desain menu lihat nilai

Pada menu lihat nilai ini admin mengelola nilai dari hasil ujian yang diikuti mahasiswa. Berikut perancangan antar muka menu lihat nilai :



Gambar 4.28 Desain menu lihat nilai

6. Desain menu form ujian

Menu form ujian ini digunakan oleh mahasiswa dalam melakukan ujian kompetensi. Berikut perancangan antarmuka menu form ujian :



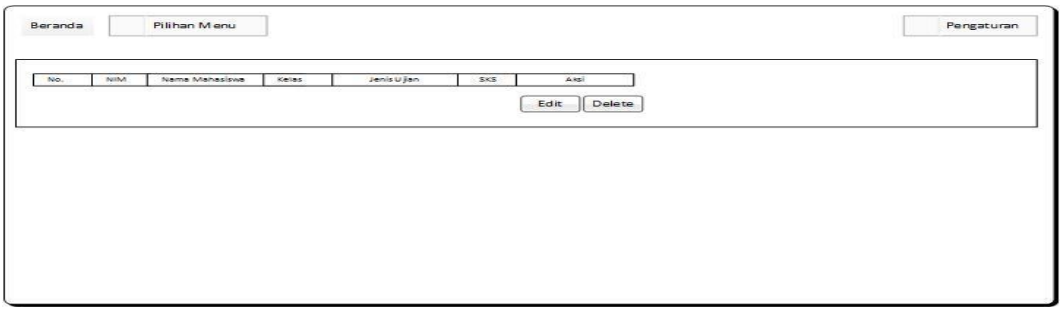
Gambar 4.29 Desain menu form ujian

4.2.5.3 Perancangan output

Dalam perancangan output ini akan menggambarkan mengenai format output yang digunakan didalam sistem ini. Rancagan output merupakan rencana pembuatan antarmuka untuk keluaran sebuah program.

1. Desain tampilan output data mahasiswa

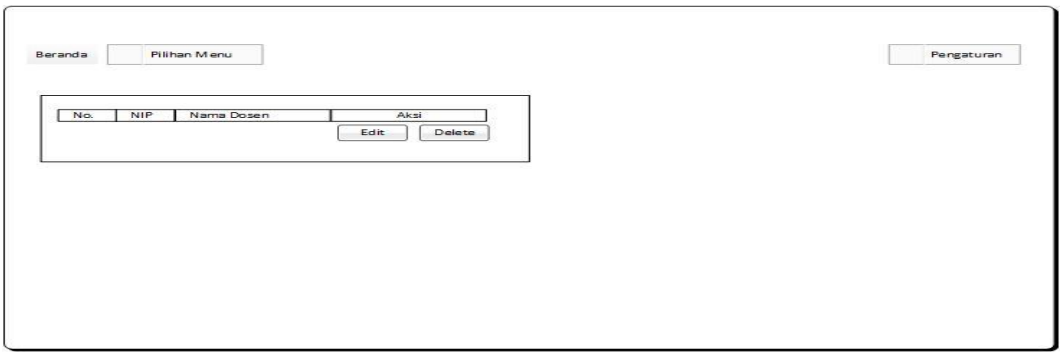
Desain tampilan keluaran ini adalah tampilan berupa table yang berisi data-data mahasiswa yang sudah diimport atau diinput oleh admin dan tersimpan dalam *database*.



Gambar 4.30 Desain tampilan output data mahasiswa

2. Desain tampilan output data instruktur

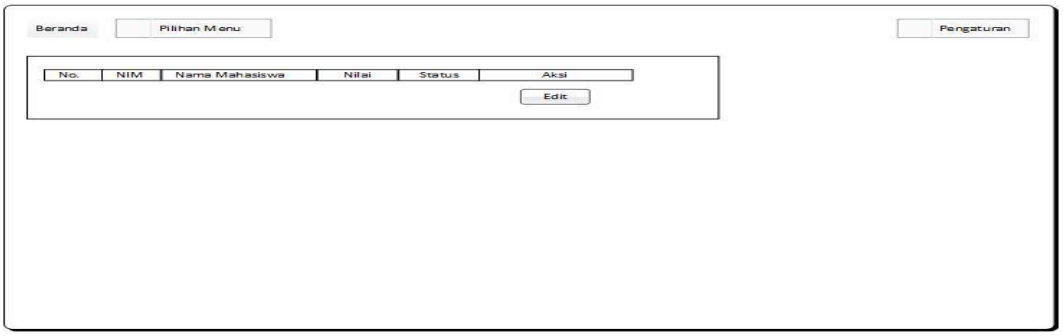
Desain tampilan keluaran ini adalah tampilan berupa table yang berisi data-data instruktur yang sudah diinput oleh admin dan tersimpan dalam *database*.



Gambar 4.31 Desain tampilan output data instruktur

3. Desain tampilan output nilai hasil ujian mahasiswa

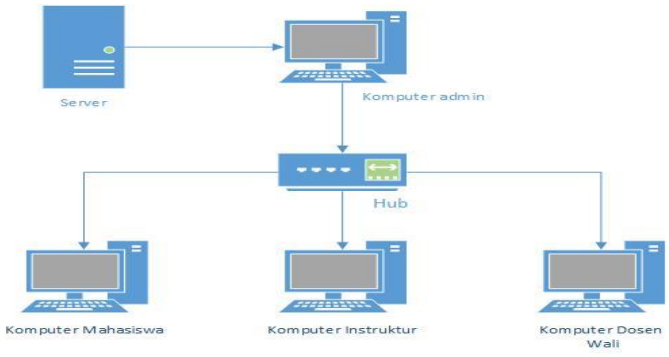
Desain tampilan keluaran ini adalah tampilan berupa table yang berisi data-data mahasiswa yang sudah melaksanakan ujian. Berikut desain tampilan nilai hasil ujiannya :



Gambar 4.32 Desain tampilan output nilai hasil ujian mahasiswa

4.2.6 Perancangan aristektur jaringan

Perancangan arsitektur jaringan adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program komputer. Pada perancangan arsitektur ini disesuaikan dengan fungsinya bagi pengguna agar tidak mengalami kesulitan dalam memilih menu-menu yang diinginkan yaitu perancangan menu *admin* dan perancangan menu user. Berikut adalah gambaran perancangan arsitektru yang penulis rancang.



Gambar 4.33 Arsitektur jaringan yang diusulkan