

SPEKIFIKASI KEBUTUHAN

PERANGKAT LUNAK

(SKPL)

APLIKASI PENILAIAN MAHASISWA


(ANIMA)

UNTUK

NANDA NURISMAN.,S.Kel.,M.T.

Dipersiapkan Oleh

Rika Laila	14117024
Nuranisda Triawati	14117043
Humairoh	14117081
Wahyu Wiranti	14117082
Muhammad Muttaqin	14117138
Achmad Bany Majesty	14117164

	Program Studi Teknik Informatika ITERA	NOMOR DOKUMEN		HALAMAN
		Revisi	A	

DAFTAR PERUBAHAN

REVISI	DESKRIPSI
A	Menambahkan ERD, UML (<i>Use Case Diagram</i> dan <i>Activity Diagram</i>)
B	
C	
D	
E	
F	
G	

INDEX TGL	-	A	B	C	D	E	F	G
Ditulis oleh								
Diperiksa oleh								
Disetujui oleh								

DAFTAR HALAMAN PERUBAHAN

HALAMAN	REVISI	HALAMAN	REVISI
19-27	A		

DAFTAR ISI

Daftar Tabel.....	6
1 Pendahuluan	7
1.1 Tujuan Penulisan Dokumen.....	7
1.2 Lingkup Masalah	7
1.3 Definisi, Istilah Dan Singkatan.....	8
1.4 Referensi	8
1.5 Deskripsi Umum Dokumen	8
2 Deskripsi Umum Aplikasi Penilaian Mahasiswa.....	10
2.1 Deskripsi Umum Aplikasi Penilaian Mahasiswa	10
2.2 Fungsi Aplikasi Asisten Penilaian Mahasiswa	10
2.3 Lingkungan Operasi.....	11
2.4 Karakteristik Pengguna.....	11
3 Deskripsi Umum Kebutuhan.....	12
3.1 Kebutuhan Antarmuka Eksternal.....	12
3.1.1 Antarmuka Pengguna	12
3.1.2 Antarmuka Perangkat Keras	13
3.1.3 Antarmuka Perangkat Lunak	13
3.2 Deskripsi Kebutuhan Fungsional.....	13
3.3 Deskripsi Kebutuhan Non Fungsional.....	14
3.4 Batasan Perancangan	15
3.5 Keruntutan (<i>Traceability</i>)	Error! Bookmark not defined.
3.6 Ringkasan Kebutuhan.....	15
3.6.1 Ringkasan Kebutuhan Fungsional.....	15
3.6.2 Ringkasan Kebutuhan Non Fungsional	16
4 Fitur Sistem	17
4.1 E-R Diagram.....	17
4.2 <i>Use Case</i> Diagram	18
4.3 <i>Use Case Scenario</i>	19
4.4 <i>Activity Diagram</i>	26
4.5 <i>Mock Up</i>	30

DAFTAR GAMBAR

Gambar 4.1. <i>E-R Diagram</i>	17
Gambar 4.2. <i>Use Case Diagram</i>	18
Gambar 4.3. <i>Activity Diagram Input Matkul</i>	26
Gambar 4.4. <i>Activity Diagram Lihat Jadwal Kuliah</i>	27
Gambar 4.5. <i>Activity Diagram Input Nilai</i>	27
Gambar 4.6. <i>Activity Diagram Input Komponen Nilai</i>	28
Gambar 4.7. <i>Activity Diagram Input Data Mahasiswa</i>	28
Gambar 4.8. <i>Activity Diagram Melihat Grafik Nilai</i>	29
Gambar 4.9. <i>Activity Diagram Mencetak PDF</i>	29
Gambar 4.10. <i>Mock Up Input Matkul</i>	30
Gambar 4.11. <i>Mock Up Input Komponen Nilai</i>	30
Gambar 4.12. <i>Mock Up Input Mahasiswa</i>	31
Gambar 4.13. <i>Mock Up Nilai Mata Kuliah</i>	31
Gambar 4.14. <i>Mock Up Grafik Nilai</i>	32

DAFTAR TABEL

Table 2.1. Karakteristik Pengguna Sistem Aplikasi Penilaian Mahasiswa	11
Table 3.1. Deskripsi Kebutuhan Non Fungsional	14
Table 3.2. Kebutuhan Fungsional.....	15
Table 3.3. Kebutuhan Non Fungsional.....	16
Table 4.1. <i>Use Case Input</i> Matkul.....	19
Table 4.2. <i>Use Case</i> Jadwal Mengajar	20
Table 4.3. <i>Use Case Input</i> Komponen Nilai.....	20
Table 4.4. <i>Use Case Input</i> Daftar Mahasiswa	21
Table 4.5. <i>Use Case</i> Input Nilai.....	22
Table 4.6. <i>Use Case</i> Melihat Hasil dan Grafik Nilai Akhir	24
Table 4.7. <i>Use Case</i> Mencetak PDF.....	25

1 PENDAHULUAN

1.1 TUJUAN PENULISAN DOKUMEN

Dokumen **SKPL** (Spesifikasi Kebutuhan Perangkat Lunak) ini dibuat untuk tujuan sebagai berikut :

1. Mendefinisikan dan menjelaskan hal-hal yang diperlukan dalam pengembangan **Aplikasi Penilaian Mahasiswa** (Anima).
2. Mendefinisikan yang diperlukan dalam pengembangan aplikasi penilaian dan juga sebagai dasar dari pembuatan aplikasi sesuai kebutuhan dengan informasi yang didapat dari klien.
3. Mendeskripsikan bagaimana alur kerja dan pengimplementasian aplikasi tersebut akan berjalan.
4. Mempermudah proses pengembangan Aplikasi Penilaian Mahasiswa pada tahap-tahap berikutnya.

1.2 LINGKUP MASALAH

ITERA (Institut Teknologi Sumatera) adalah sebuah Perguruan Tinggi Negeri yang berlokasi di Provinsi Lampung. ITERA didirikan berdasarkan Peraturan Presiden No. 124 Tahun 2014. Yang bertujuan memajukan, mengembangkan dan menyebar luaskan ilmu pengetahuan, teknologi, seni, ilmu kemanusiaan, untuk meningkatkan kesejahteraan penduduk Sumatera khususnya, dan bangsa Indonesia sejalan dengan dinamika masyarakat Indonesia serta masyarakat dunia, dengan tetap menjunjung tinggi nilai-nilai sosial, kemanusiaan, dan lingkungan melalui wahana Tridharma perguruan tinggi.

ITERA berupaya menjadikan dirinya sebagai perguruan tinggi terbaik di Sumatera. Untuk itu ITERA memiliki dosen dengan kualitas yang sangat baik dan juga diimbangi oleh kecanggihan teknologinya. Aplikasi penilaian ini merupakan sebuah aplikasi yang dapat memberikan kemudahan bagi dosen di ITERA dalam proses pengolahan nilai mahasiswa tanpa perlu koneksi *internet* dikarenakan aplikasi ini menggunakan *platform desktop*. Dosen dapat memasukkan nilai tanpa takut dengan koneksi *internet* yang buruk. Memasukan data berupa mata kuliah, rentang nilai, dan data mahasiswa beserta nilainya. Karena jumlah data mahasiswa yang akan dimasukkan dalam skala yang besar, tentunya ini akan mengefesienkan waktu pengerjaannya.

1.3 DEFINISI, ISTILAH DAN SINGKATAN

Adapun definisi, istilah dan singkatan yang digunakan dalam dokumen SKPL ini adalah sebagai berikut :

- SKPL : Spesifikasi Kebutuhan Perangkat Lunak
- ITERA : Institut Teknologi Sumatera
- ANIMA : Aplikasi Penilaian Mahasiswa
- RDBMS : *Relational Database Management System*
- SQL : *Structure Query Language*
- IEEE : *Institute of Electrical and Electronics Engineers*
- ERD : *Entity Relationship Diagram*
- PDF : *Portable Document Format*
- NIM : Nomor Induk Mahasiswa
- UTS : Ujian Tengah Semester
- UAS : Ujian Akhir Semester

1.4 REFERENSI

Daftar dokumen yang digunakan sebagai acuan atau rujukan dalam penyusunan dokumen SKPL ini adalah sebagai berikut :

1. Dokumen SKPL PT. PRADIPTA INTIMEDIA SELARAS.
2. Dokumen SKPL Sri Lanka Institute of Information Technology, Project ID: ITP-14-MTR-03

1.5 DESKRIPSI UMUM DOKUMEN

Dokumen SKPL ini dibuat untuk memberikan informasi mengenai spesifikasi Aplikasi Penilaian Mahasiswa. Dokumen ini berisikan informasi sebagai berikut :

1. Deskripsi Umum Aplikasi

Aplikasi Penilaian adalah perangkat lunak berbasis *desktop* untuk membantu dalam penginputan data berupa mata kuliah, rentang nilai, mengelola data nilai yang ada menjadi nilai akhir, melakukan operasi pengurutan nilai dan juga membentuk grafik persebaran nilai untuk mempermudah dan mempercepat masa penginputan nilai. Aplikasi ini

dapat dijalankan dari berbagai komputer yang memakai *desktop* juga dapat diakses tanpa menggunakan jaringan *internet*, jadi nantinya aplikasi ini harus sudah terinstal di *desktop* komputer.

2. Deskripsi Umum Kebutuhan Aplikasi Yang Akan Diimplementasikan

Deskripsi umum kebutuhan aplikasi adalah sebuah informasi yang bersifat teknis dalam pengimplementasiannya sebagai acuan dalam pengembangan aplikasi di tahap selanjutnya.

Informasi dalam dokumen **SKPL** ini disajikan dan diorganisasikan sesuai standar *IEEE 830-1998* dengan struktur sebagai berikut :

1. BAB I.

Berisi informasi umum yang merupakan bagian pendahuluan, yang meliputi tujuan penulisan dokumen, lingkup masalah, definisi, istilah dan akronim, referensi, serta deskripsi umum dokumen.

2. BAB II.

Berisi deskripsi umum dari Aplikasi Penilaian ahasiswa yang akan dikembangkan, yang meliputi deskripsi umum sistem aplikasi penilaian mahasiswa, fungsi aplikasi penilaian mahasiswa dan karakteristik pengguna.

3. BAB III.

Berisi informasi mengenai deskripsi umum kebutuhan perangkat lunak yang akan dikembangkan. Bagian ini meliputi informasi mengenai kebutuhan antarmuka eksternal, deskripsi fungsional, *data requirement*, *non functional requirement*, batasan perancangan, keruntutan (*traceability*) dan ringkasan kebutuhan.

2 DESKRIPSI UMUM APLIKASI PENILAIAN MAHASISWA

2.1 DESKRIPSI UMUM APLIKASI PENILAIAN MAHASISWA

Aplikasi Penilaian Mahasiswa merupakan sebuah aplikasi yang dapat membantu dalam pengolahan nilai. Aplikasi ini dapat mengolah nilai tugas, nilai kuis, dan nilai lainnya menjadi nilai akhir. Selain itu, aplikasi ini juga dapat membentuk grafik persebaran nilai yang dapat membantu dalam hal analisis lebih lanjut.

Aplikasi Penilaian Mahasiswa dibangun berbasiskan *desktop* yang dirancang dengan *platform Java Desktop* dan dengan satu pengguna. Aplikasi Penilaian Mahasiswa dapat dijalankan pada *platform* sistem operasi *Windows 7, 8, 10* atau *Linux*, dan *Database RDBMS* sehingga mampu menangani dan mengelola data.

Alur Penggunaan Aplikasi Penilaian Mahasiswa.

1. Pengguna memasukkan data mata kuliah yang diampu beserta jadwal, rentang nilai akhir dan komponen penilaian.
2. Pengguna memasukkan nama mahasiswa yang mengikuti mata kuliah tertentu.
3. Pengguna memasukkan nilai mahasiswa sesuai komponen yang ditentukan, berdasarkan nama mahasiswa yang mengambil mata kuliah.
4. Data-data yang sudah diinput akan otomatis dikelola dan dihitung menjadi nilai akhir.
5. Grafik akan berubah sesuai data nilai akhir yang dimasukkan.
6. Pengguna dapat mengubah rentang nilai suatu mata kuliah jika dibutuhkan.

2.2 FUNGSI APLIKASI ASISTEN PENILAIAN MAHASISWA

Fungsi utama sistem Aplikasi Penilaian Mahasiswa yang dibangun adalah :

1. Memasukkan data berupa mata kuliah, rentang nilai, rumus penilaian, dan data mahasiswa berserta nilainya;
2. Mengolah data nilai yang ada menjadi nilai akhir;
3. Melakukan operasi pengurutan nilai akhir;
4. Membentuk grafik persebaran nilai;
5. Menampilkan jadwal mengajar berdasarkan mata kuliah yang ada;
6. Mencetak nilai akhir dan grafik persebaran nilai dengan format *PDF*.

2.3 LINGKUNGAN OPERASI

SERVER SITE

Perangkat lunak yang akan digunakan adalah sebagai berikut:

1. *Windows* sebagai sistem operasi yang digunakan oleh pengguna.
2. *MySQL* sebagai *RDBMS Server*.

CLIENT SITE

Perangkat lunak yang diusulkan adalah sebagai berikut:

1. *Windows / Linux* sebagai sistem operasi yang digunakan oleh pengguna.

DEVELOPMENT TOOLS

Perangkat lunak yang akan digunakan adalah sebagai berikut :

1. *Netbeans*
2. *XAMPP*

2.4 KARAKTERISTIK PENGGUNA

Karakteristik pengguna terdiri yang akan menggunakan Aplikasi Penilaian Mahasiswa terdiri dari satu pengguna dijelaskan pada **Table 2.1**.

Table 2.1. Karakteristik Pengguna Sistem Aplikasi Penilaian Mahasiswa

Kategori	Tugas	Hak Akses Ke Aplikasi
Pengguna	Menjalankan sistem Aplikasi Penilaian Mahasiswa	Mempunyai hak akses penuh sebagai pengguna aplikasi terhadap data pengguna untuk <i>input</i> dan <i>update</i> data.

3 DESKRIPSI UMUM KEBUTUHAN

3.1 KEBUTUHAN ANTARMUKA EKSTERNAL

Sistem Aplikasi Penilaian Mahasiswa ini dibangun di atas *platform* yang beroperasikan sistem operasi *Windows* dan *Linux*. Dalam operasionalnya, perangkat lunak yang akan dikembangkan memerlukan adanya interaksi dengan komponen-komponen lain diluar perangkat lunak itu seperti: pengguna sebagai seorang pengguna perangkat lunak, perangkat keras dimana perangkat lunak ini akan dijalankan.

3.1.1 ANTARMUKA PENGGUNA

Aplikasi Penilaian Mahasiswa ini menggunakan antarmuka berbasis *desktop* dan pengguna mengoperasikannya menggunakan *keyboard* dan *mouse* dengan sistem operasi *Windows* dan *Linux*.

Perangkat lunak yang akan dikembangkan membutuhkan interaksi dengan pengguna sebagai pemakai aplikasi perangkat lunak. Dalam melakukan interaksi dengan pemakai perangkat lunak ini membutuhkan perangkat untuk melakukan proses transformasi *input* dan *output* dari dan ke pengguna. Perangkat tersebut adalah sebagai berikut:

1. Perangkat *Keyboard*

Keyboard diperlukan sebagai sarana bagi pemakai untuk mengetikkan data masukan yang akan diproses perangkat lunak. Spesifikasi perangkat *keyboard* adalah jenis *keyboard* yang *compatible* untuk sistem operasi *Microsoft Windows*.

2. Perangkat *Mouse*

Perangkat *mouse* digunakan sebagai sarana bagi pemakai untuk memasukkan data input bagi perangkat lunak. Meskipun sebagian besar fungsi *mouse* dapat digantikan dengan perangkat *keyboard* tetapi akan lebih ergonomis apabila pada jenis *input* tertentu digunakan *mouse* sebagai salah satu perangkat yang dibutuhkan sebagai antarmuka dengan pengguna. Perangkat *mouse* yang dapat digunakan adalah semua jenis *mouse* yang *compatible* dengan sistem operasi *Microsoft Windows*.

3. Perangkat *Monitor*

Monitor sebagai sarana untuk menampilkan aplikasi kepada pemakai mempunyai spesifikasi diantaranya: *monitor* mampu menampilkan grafis dengan

kualitas warna yang baik (*true color*) untuk menampilkan data-data dalam bentuk persebaran grafik.

3.1.2 ANTARMUKA PERANGKAT KERAS

Antarmuka perangkat keras yang dibutuhkan dalam perangkat lunak ini meliputi:

1. *Graphic Card / VGA Card*

Kartu grafik yang dibutuhkan dalam mengoperasikan perangkat lunak ini diharapkan mempunyai kapasitas 4 *Megabyte* ke atas. Apabila menggunakan kartu grafik yang punya kemampuan dibawahnya perangkat lunak masih bisa berjalan akan tetapi tampilan yang diberikan kurang baik.

2. *Desktop*

Desktop dibutuhkan untuk menjalankan perangkat lunak tersebut. Apabila perangkat lunak dijalankan selain di *desktop*, perangkat lunak tersebut tidak akan mendukung jalannya program yang telah dibuat oleh *Developer*. Spesifikasi *desktop* yang dibutuhkan adalah yang *compatible* dengan sistem operasi *Windows, Linux*.

3.1.3 ANTARMUKA PERANGKAT LUNAK

Aplikasi ini dijalankan di sistem operasi berbasis *Windows* dan *Linux*, aplikasi ini dibuat dengan menggunakan *Java* dan pengolahan *Database SQL Server*. Untuk koneksi *database* digunakan *MariaDB*.

3.2 DESKRIPSI KEBUTUHAN FUNGSIONAL

Aplikasi Penilaian Mahasiswa berfungsi untuk mengolah nilai mahasiswa memiliki kebutuhan sebagai berikut :

1. Mahasiswa
 - NIM dan Nama mahasiswa
2. Mata Kuliah
 - Jumlah SKS dan nama mata kuliah yang diampu
 - Komponen dan rentang Penilaian
3. Nilai mahasiswa

Dari 3 komponen diatas, akan dikalkulasikan dan menghasilkan informasi sebagai berikut :

1. Nilai akhir mata kuliah
2. Grafik persebaran nilai akhir
3. Mencetak nilai dalam bentuk *PDF*

3.3 DESKRIPSI KEBUTUHAN NON FUNGSIONAL

Kebutuhan Non Fungsional atau batasan layanan yang ditawarkan sistem berupa *Availability*, *Reliability*, *Portability* dan *Response Time* yang akan dijelaskan pada **Table 3.1.**

Table 3.1. Deskripsi Kebutuhan Non Fungsional

SKPL-ID	PARAMETER	<i>REQUIREMENT</i>
SKPL-14	<i>Availability</i>	Ketersediaan aplikasi yang dapat di- <i>update</i> sewaktu-waktu dan harus mampu beroperasi 7 hari perminggu, 24 jam per hari tanpa gagal
	<i>Reliability</i>	Kehandalan sistem terjamin dan memiliki keamanan yang baik
SKPL-15	<i>Ergonomy</i>	N/A
	<i>Portability</i>	Data dapat dilakukan pemindahan dengan mudah dan data ditransfer ke perangkat lain.
	<i>Memory</i>	N/A
	<i>Response time</i>	Aplikasi memproses nilai dalam waktu maksimal 3 detik
	<i>Safety</i>	N/A
SKPL-16	<i>Security</i>	N/A

3.4 BATASAN PERANCANGAN

Batasan khusus dalam perancangan Aplikasi Penilaian Mahasiswa adalah sebagai berikut :

- Bahasa pemrograman yang digunakan adalah *Java*.
- *DBMS* yang digunakan adalah *Mysql*.
- Aplikasi Penilaian Mahasiswa hanya dapat diakses/digunakan oleh pengguna/pemilik aplikasi.
- Informasi yang ditampilkan dapat dicetak dalam format *PDF*.

3.5 RINGKASAN KEBUTUHAN

3.5.1 RINGKASAN KEBUTUHAN FUNGSIONAL

Kebutuhan fungsional merupakan kebutuhan yang berisi proses-proses atau layanan yang nantinya harus disediakan oleh aplikasi. Hal ini mencakup bagaimana aplikasi harus bereaksi pada inputan pengguna dan bagaimana perilaku sistem pada situasi tertentu. Berikut merupakan ringkasan kebutuhan fungsional pada Aplikasi Penilaian Mahasiswa dijelaskan pada **Table 3.2**.

Table 3.2. Kebutuhan Fungsional

SKPL-Id	DESKRIPSI
SKPL-01	Aplikasi mampu melakukan pengelolaan terhadap data nilai mahasiswa, dan melakukan perhitungan nilai akhir
SKPL-02	Aplikasi dapat menambahkan data mata kuliah baru
SKPL-03	Aplikasi dapat memasukkan rentang nilai pada suatu mata kuliah
SKPL-04	Aplikasi dapat mengatur komponen penilaian pada suatu mata kuliah
SKPL-05	Aplikasi dapat memasukkan data mahasiswa yang mengikuti mata kuliah tertentu
SKPL-06	Aplikasi dapat memasukkan nilai mahasiswa berdasarkan komponen nilai yang ditetapkan
SKPL-07	Aplikasi dapat melakukan perhitungan nilai akhir dari suatu mata kuliah berdasarkan inputan nilai dari pengguna
SKPL-08	Aplikasi dapat menampilkan grafik persebaran nilai pada suatu

SKPL-Id	DESKRIPSI
	mata kuliah
SKPL-09	Aplikasi dapat menampilkan jadwal mata kuliah berdasarkan data yang dimasukkan pengguna
SKPL-10	Aplikasi dapat mencetak daftar nilai akhir suatu mata kuliah dengan format <i>PDF</i>
SKPL-11	Aplikasi dapat mengubah rentang nilai suatu mata kuliah saat dibutuhkan (dinamis)
SKPL-12	Aplikasi dapat mengubah komponen penilaian suatu mata kuliah saat dibutuhkan (dinamis)
SKPL-13	Aplikasi dapat melakukan pengurutan (<i>sorting</i>) berdasarkan nilai akhir suatu mata kuliah

3.5.2 RINGKASAN KEBUTUHAN NON FUNGSIONAL

Kebutuhan Non Fungsional kebutuhan yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem. Ringkasan kebutuhan non fungsional secara rinci dijelaskan pada **Table 3.3**.

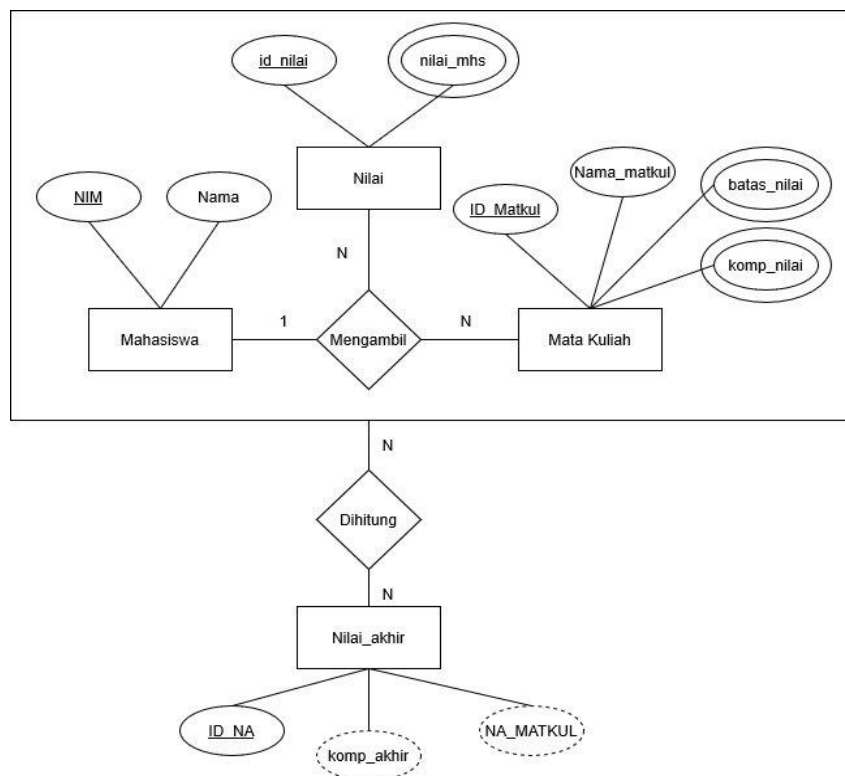
Table 3.3. Kebutuhan Non Fungsional

SKPL-Id	DESKRIPSI
SKPL-17	Ketersediaan aplikasi yang dapat di- <i>update</i> sewaktu-waktu dan harus mampu beroperasi 7 hari perminggu, 24 jam per hari tanpa gagal.
SKPL-18	Mudah digunakan pada lingkungan sistem operasi <i>Windows</i> dan <i>Linux</i> , dengan <i>DBMS SQL Server</i>
SKPL-19	Kecepatan saat penggunaan tergantung spesifikasi komputer
SKPL-20	Mampu memproses nilai dalam waktu maksimal 3 detik
SKPL-21	Antarmuka pengguna menggunakan bahasa Indonesia
SKPL-22	Tidak pernah gagal dalam melakukan proses pengolahan data nilai
SKPL-23	Perangkat lunak hanya dapat diakses oleh pengguna

4 FITUR SISTEM

4.1 E-R Diagram

ERD pada aplikasi Aplikasi Penilaian Mahasiswa memiliki 4 entitas yaitu Mahasiswa, Nilai, Mata Kuliah dan Nilai Akhir. Pada entitas Mahasiswa memiliki 2 atribut yaitu NIM dan Nama. Pada entitas Nilai terdapat atribut id_nilai dan nilai_mhs yang bersifat *multivalue* atribut karena memiliki komponen seperti tugas, tugas besar, kuis, UTS dan UAS. Pada entitas Mata Kuliah memiliki 4 atribut yaitu ID_matkul, Nama matkul, batas_nilai dan komponen nilai. Pada atribut batas nilai dan komponen nilai bersifat *multivalue* atribut karena memiliki banyak komponen berupa tugas, tugas besar, kuis, UTS dan UAS. Pada entitas Nilai akhir memiliki 3 atribut seperti ID_NA, komp_akhir dan NA_Matkul. Atribut komp_akhir dan NA_Matkul bersifat atribut turunan karena nilai pada atribut didapat dari perhitungan beberapa atribut Nilai dan Mata Kuliah. Lebih lengkapnya ERD ditampilkan pada **Gambar 4.1.**



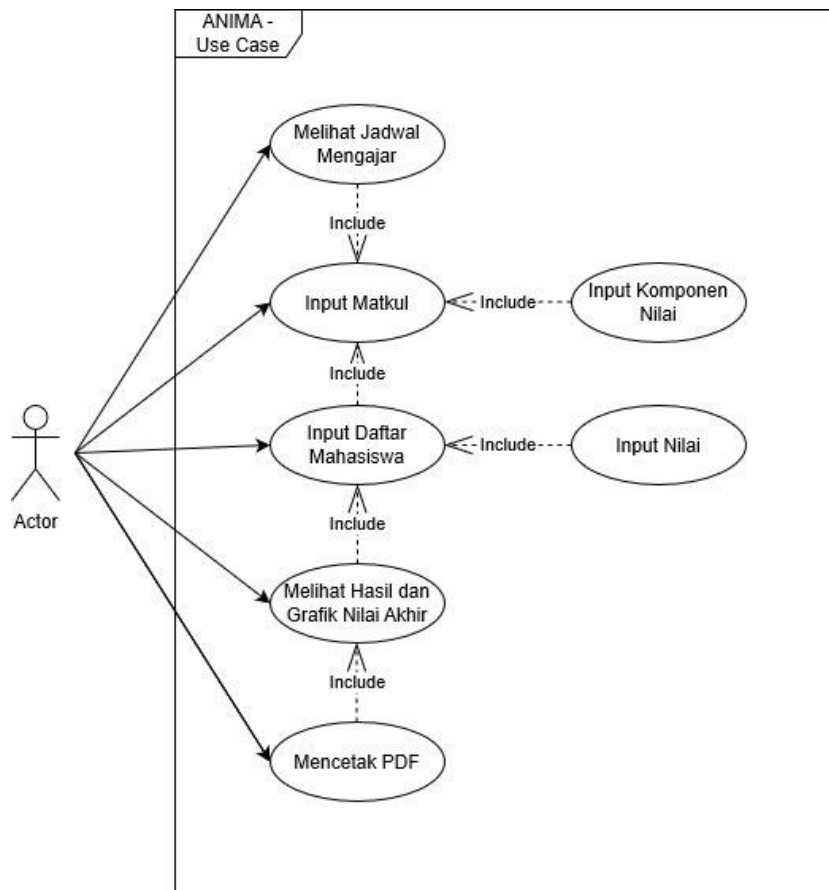
Gambar 4.1. E-R Diagram

4.2 Use Case Diagram

Pada **Gambar 4.2** menampilkan tentang fungsi atau fitur apa saja yang dapat digunakan oleh *user*, pada pengembangan Aplikasi Penilaian Mahasiswa ini terdapat satu aktor yaitu :

1. Pengguna (*user*)

- *Input* Matkul
- *Input* Daftar Mahasiswa
- *Input* Komponen Nilai
- Melihat Jadwal Mengajar
- Melihat Hasil dan Grafik Nilai Akhir
- Mencetak *PDF*



Gambar 4.2. Use Case Diagram

4.3 Use Case Scenario

1) Use Case Input Matkul

Alur jalannya proses *input* matkul, serta syarat dan aksi yang harus dilakukan dari sisi pengguna dan sistem dijelaskan pada **Table 4.1**.

Table 4.1. Use Case Input Matkul

<i>Use case Name</i>	<i>Input Matkul</i>	
<i>Goal</i>	Menambahkan mata kuliah	
<i>Primary Actors</i>	<i>User / Pengguna Aplikasi</i>	
<i>Precondition</i>	1. Sudah masuk Aplikasi 2. <i>User</i> ingin menambahkan daftar mata kuliah	
<i>Post condition</i>	<i>User</i> berhasil menambahkan mata kuliah	
<i>Main flow</i>	<i>Step</i>	<i>Action</i>
	1	<i>User</i> pilih fitur <i>Input Matkul</i>
	2	Sistem menampilkan menu <i>Input Matkul</i>
	3	<i>User</i> memasukkan informasi matkul berupa SKS dan jadwal
	4	Sistem berhasil menambahkan mata kuliah

2) Use Case Jadwal Mengajar

Alur jalannya proses *input* jadwal mengajar, serta syarat dan aksi yang harus dilakukan dari sisi pengguna dan sistem dijelaskan pada **Table 4.2**.

Table 4.2. Use Case Jadwal Mengajar

<i>Use case Name</i>	Melihat Jadwal Mengajar	
<i>Goal</i>	Menampilkan jadwal mengajar <i>user</i>	
<i>Primary Actors</i>	<i>User</i>	
<i>Precondition</i>	1. Sudah masuk Aplikasi 2. <i>User</i> ingin melihat jadwal mengajar semua mata kuliah	
<i>Post condition</i>	<i>User</i> berhasil melihat jadwal mengajar semua mata kuliah	
<i>Main flow</i>	<i>Step</i>	<i>Action</i>
	1	<i>User</i> pilih fitur melihat jadwal mengajar
	2	Sistem menampilkan daftar jadwal mengajar

3) Use Case Input Komponen Nilai

Alur jalannya proses *input* komponen nilai, serta syarat dan aksi yang harus dilakukan dari sisi pengguna dan sistem dijelaskan pada **Table 4.3**.

Table 4.3. Use Case Input Komponen Nilai

<i>Use case Name</i>	<i>Input</i> Komponen Nilai
<i>Goal</i>	Menambah komponen nilai ke mata kuliah
<i>Primary Actors</i>	<i>User</i>
<i>Precondition</i>	1. Sudah masuk Aplikasi 2. <i>User</i> sudah memasukkan minimal 1 mata kuliah 3. <i>User</i> ingin memasukkan komponen nilai ke mata kuliah

<i>Post condition</i>	<i>User</i> berhasil menambahkan komponen nilai ke suatu mata kuliah	
<i>Main flow</i>	<i>Step</i>	<i>Action</i>
	1	<i>User</i> pilih fitur <i>Input</i> Komponen Nilai
	2	Sistem menampilkan menu <i>Input</i> Komponen Nilai
	3	<i>User</i> memasukkan informasi komponen nilai berupa persentase Tugas, Tugas Besar, Kuis, UTS, UAS dan Nilai lainnya
	4	Sistem berhasil menambahkan komponen nilai

4) *Use Case Input* Daftar Mahasiswa

Alur jalannya proses *input* daftar mahasiswa, serta syarat dan aksi yang harus dilakukan dari sisi pengguna dan sistem dijelaskan pada **Table 4.4**.

Table 4.4. *Use Case Input* Daftar Mahasiswa

<i>Use case Name</i>	<i>Input</i> Daftar Mahasiswa
<i>Goal</i>	Menambah dan Memasukkan nama mahasiswa
<i>Primary Actors</i>	<i>User</i>
<i>Precondition</i>	1. Sudah masuk Aplikasi 2. <i>User</i> sudah memasukkan minimal 1 mata kuliah 3. <i>User</i> ingin menambahkan mahasiswa ke mata kuliah

<i>Post condition</i>	<i>User</i> berhasil menambahkan mahasiswa ke suatu mata kuliah	
<i>Main flow</i>	<i>Step</i>	<i>Action</i>
	1	<i>User</i> pilih fitur Daftar Mahasiswa
	2	Sistem menampilkan menu Daftar Mahasiswa
	3	<i>User</i> memilih satu mata kuliah
	4	<i>User</i> memilih menu tambah mahasiswa
	5	Sistem menampilkan halaman untuk data mahasiswa
	6	<i>User</i> memasukkan data mahasiswa berupa NIM dan nama
	7	Sistem berhasil menambahkan nama mahasiswa ke mata kuliah

5) *Use Case Input Nilai*

Alur jalannya proses *input* nilai, serta syarat dan aksi yang harus dilakukan dari sisi pengguna dan sistem dijelaskan pada **Table 4.5**.

Table 4.5. Use Case Input Nilai

<i>Use case Name</i>	<i>Input Nilai</i>
<i>Goal</i>	Memasukkan nilai mahasiswa
<i>Primary Actors</i>	<i>User</i>
<i>Precondition</i>	1. Sudah masuk Aplikasi

	2. <i>User</i> sudah memasukkan minimal 1 mata kuliah dan terdapat minimal 1 mahasiswa 3. <i>User</i> ingin input nilai mahasiswa	
<i>Post condition</i>	<i>User</i> berhasil menambahkan nilai mahasiswa	
<i>Main flow</i>	<i>Step</i>	<i>Action</i>
	1	<i>User</i> pilih fitur <i>Input Nilai</i>
	2	Sistem menampilkan menu <i>Input Nilai</i>
	3	<i>User</i> memilih satu mata kuliah
	4	Sistem menampilkan daftar mata kuliah yang ada
	5	<i>User</i> memilih satu nama mahasiswa yang akan di- <i>input</i> nilainya, kemudian <i>input</i> nilai
	6	Sistem berhasil menambahkan nilai mahasiswa

6) *Use Case* Melihat Hasil dan Grafik Nilai Akhir

Alur jalannya proses untuk melihat hasil dan grafik nilai akhir, serta syarat dan aksi yang harus dilakukan dari sisi pengguna dan sistem dijelaskan pada **Table 4.6**.

Table 4.6. *Use Case* Melihat Hasil dan Grafik Nilai Akhir

<i>Use case Name</i>	Melihat Hasil dan Grafik Nilai Akhir	
<i>Goal</i>	Menampilkan hasil dan grafik nilai akhir kepada <i>User</i>	
<i>Primary Actors</i>	<i>User</i>	
<i>Precondition</i>	1. Sudah masuk Aplikasi 2. <i>User</i> sudah memasukkan minimal 1 mata kuliah dan sudah <i>input</i> semua nilai mahasiswa 3. <i>User</i> ingin melihat nilai akhir suatu mata kuliah	
<i>Post condition</i>	<i>User</i> berhasil melihat hasil dan grafik nilai akhir	
<i>Main flow</i>	<i>Step</i>	<i>Action</i>
	1	<i>User</i> Pilih fitur Daftar Mahasiswa
	2	Sistem menampilkan daftar mahasiswa dan daftar nilai akhir
	3	<i>User</i> memilih menu Tampilkan Grafik Kelas
	4	Sistem menampilkan grafik nilai akhir kelas

7) *Use Case Cetak PDF*

Alur jalannya proses untuk mencetak *PDF*, serta syarat dan aksi yang harus dilakukan dari sisi pengguna dan sistem dijelaskan pada **Table 4.7**.

Table 4.7. Use Case Mencetak PDF

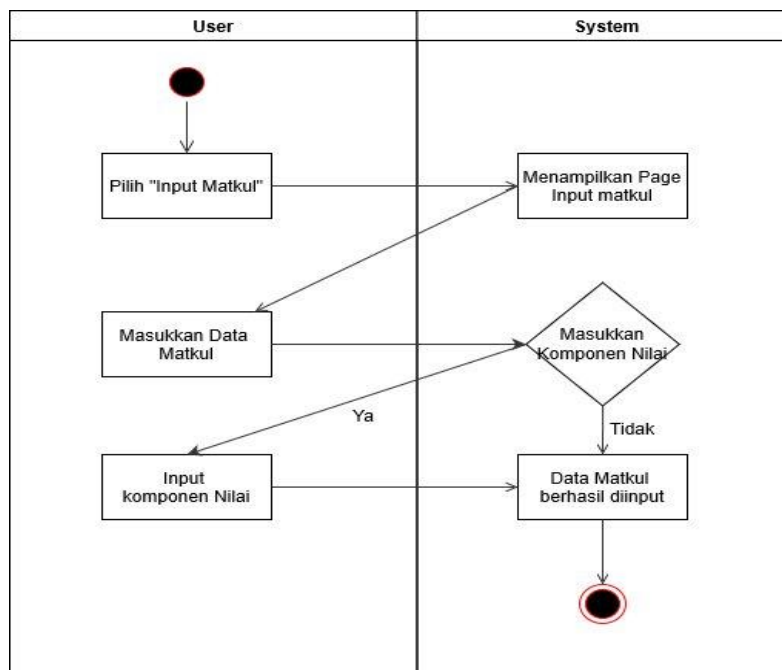
<i>Use case Name</i>	Mencetak <i>PDF</i>	
<i>Goal</i>	Mencetak nilai akhir dalam format <i>.pdf</i>	
<i>Primary Actors</i>	<i>User</i>	
<i>Precondition</i>	1. Sudah masuk Aplikasi 2. <i>User</i> sudah memasukkan minimal 1 mata kuliah dan sudah <i>input</i> semua nilai mahasiswa 3. <i>User</i> mencetak nilai dalam format pdf	
<i>Post condition</i>	<i>User</i> menerima hasil dalam bentuk <i>PDF</i>	
<i>Main flow</i>	<i>Step</i>	<i>Action</i>
	1	<i>User</i> pilih fitur Daftar Mahasiswa
	2	Sistem menampilkan daftar mahasiswa dan daftar nilai akhir
	3	<i>User</i> memilih menu Cetak <i>PDF</i>
	4	Sistem mengubah nilai akhir dalam bentuk <i>PDF</i>

4.4 Activity Diagram

Activity Diagram adalah diagram yang menggambarkan berbagai aliran aktivitas dalam sebuah sistem/aplikasi yang akan dirancang dan bagaimana masing-masing aktivitas berawal, proses yang terjadi dan bagaimana aktivitas tersebut berakhir. Berikut ini adalah *Activity Diagram* setiap proses didalam Aplikasi Penilaian Mahasiswa.

1) *Input Matkul*

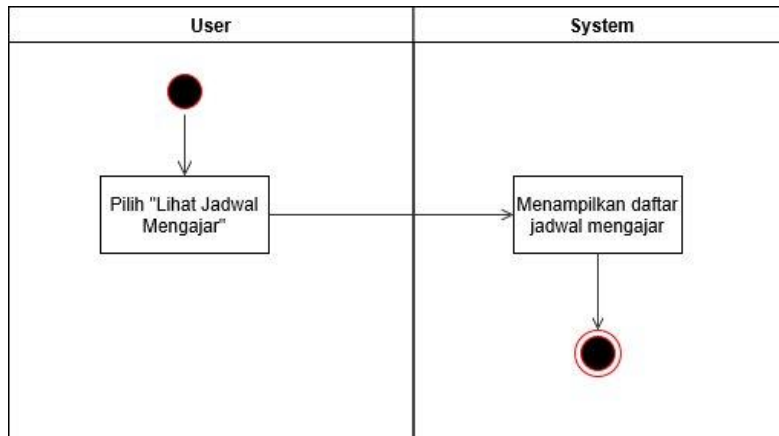
Activity Diagram proses *input* matkul seperti pada **Gambar 4.3** , pertama *user*/pengguna memilih menu *input* matkul dan menunggu aplikasi menampilkan halaman untuk *input* matkul. Setelah halaman terbuka, pengguna memasukkan data Mata kuliah dan memilih untuk memasukkan komponen nilai atau lanjut. Jika ingin memasukkan komponen nilai maka pengguna akan memasukkan apa saja komponen nilai yang digunakan dan jika tidak memasukkan komponen nilai, maka proses akan langsung berakhir.



Gambar 4.3. Activity Diagram Input Matkul

2) Melihat Jadwal Mengajar

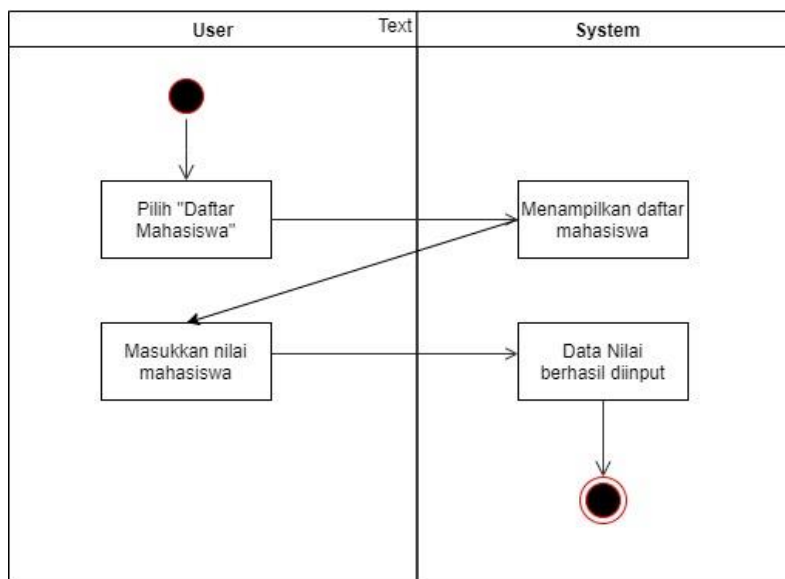
Activity Diagram proses melihat jadwal mengajar seperti pada **Gambar 4.4** , pertama pengguna memilih menu melihat jadwal mengajar dan menunggu aplikasi menampilkan halaman tersebut. Setelah halaman terbuka, aplikasi akan menampilkan jadwal mengajar mata kuliah secara lengkap.



Gambar 4.4. Activity Diagram Lihat Jadwal Kuliah

3) Input Nilai

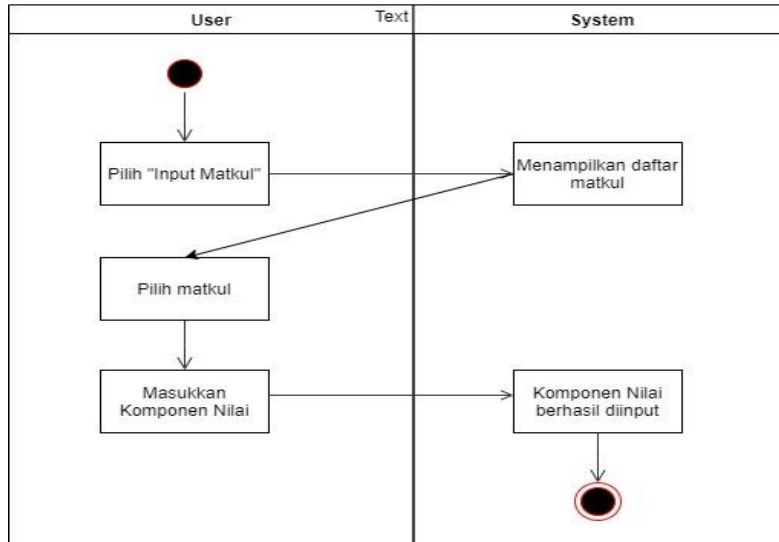
Activity Diagram proses *input* nilai seperti pada **Gambar 4.5** , pertama pengguna memilih menu *input* nilai dan menunggu aplikasi menampilkan halaman untuk *input* nilai. Setelah halaman terbuka akan muncul daftar nama mahasiswa, kemudian pengguna dapat memasukkan nilai mahasiswa yang diinginkan.



Gambar 4.5. Activity Diagram Input Nilai

4) *Input* Komponen Nilai

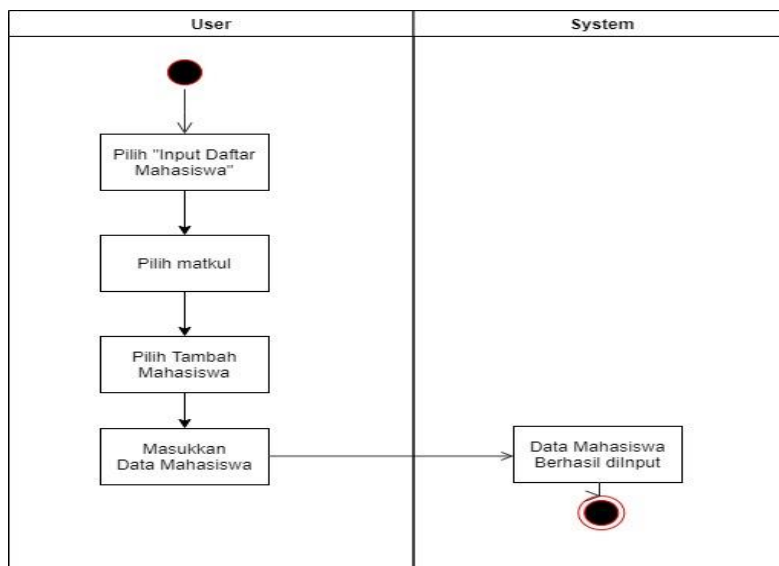
Activity Diagram proses *input* komponen nilai seperti pada **Gambar 4.6** , pertama pengguna memilih menu *input* matkul dan pilih matkul yang akan dimasukkan/diubah komponen nilainya. Setelah halaman terbuka, pengguna memasukkan komponen nilai berupa persentase setiap komponen.



Gambar 4.6. Activity Diagram Input Komponen Nilai

5) *Input* Data Mahasiswa

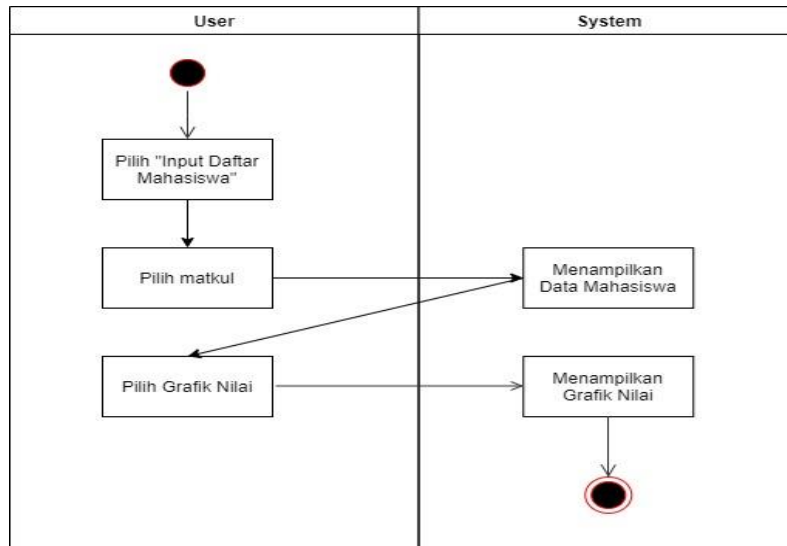
Activity Diagram proses *input* mahasiswa seperti pada **Gambar 4.7** , pertama pengguna memilih menu *input* data mahasiswa dan menunggu aplikasi menampilkan halaman tersebut. Setelah halaman terbuka, pilih matkul yang diinginkan dan isi data mahasiswa berupa nama dan NIM.



Gambar 4.7. Activity Diagram Input Data Mahasiswa

6) Melihat Grafik Nilai

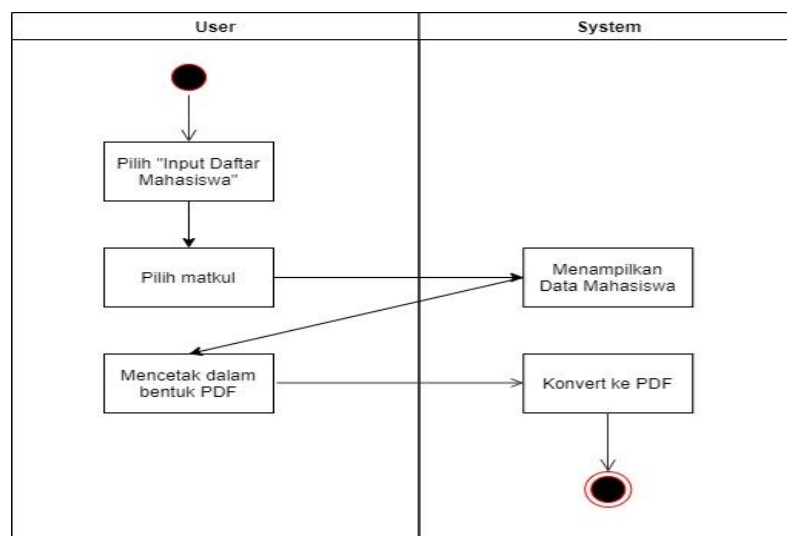
Activity Diagram proses melihat grafik nilai akhir seperti pada **Gambar 4.8** , pertama pengguna memilih menu daftar mahasiswa dan menunggu aplikasi menampilkan halaman tersebut. Setelah halaman terbuka, pilih matkul yang diinginkan dan pilih Tampilkan Grafik Nilai.



Gambar 4.8. Activity Diagram Melihat Grafik Nilai

7) Mencetak PDF

Activity Diagram proses mencetak *PDF* seperti pada **Gambar 4.9** , pertama pengguna memilih menu daftar mahasiswa dan menunggu aplikasi menampilkan halaman tersebut. Setelah halaman terbuka, pilih matkul yang diinginkan maka akan muncul daftar nama mahasiswa dan nilainya, kemudian pilih cetak *PDF*.



Gambar 4.9. Activity Diagram Mencetak PDF

4.5 Mock Up

1) Tampilan *Input* Matkul

Tampilan pada halaman *input* matkul seperti pada **Gambar 4.10** , pada rancangan tampilan *input* matkul ini terdapat bagian untuk *input* data matkul seperti kode, nama mata kuliah, SKS, waktu kuliah dan ruangnya.

Tools Dosen Institut Teknologi Sumatera	
Buat Matakuliah	
Komponen Nilai	Jumlah Komponen
Kode Matakuliah	Isi Inputan
Kode Kelas	Isi Inputan
Nama Matakuliah	Isi Inputan
SKS	Isi Inputan
Waktu	Isi Inputan (Hari, Jam)
Ruangan	Isi Inputan
CANCEL NEXT	

+ Input Mahasiswa

Output Identitas Matakuliah
Identitas Matakuliah

Output komponen nilai
KOMPONEN NILAI

Gambar 4.10. Mock Up *Input* Matkul

2) Tampilan *Input* Komponen Nilai

Tampilan pada halaman *input* komponen nilai seperti pada **Gambar 4.11** , pada rancangan tampilan *input* komponen nilai ini terdapat bagian untuk input persentase nilai tugas, kuis, UTS,UAS dan nilai yang dibutuhkan.

Tools Dosen Institut Teknologi Sumatera	
Buat Matakuliah	
Komponen Nilai	Jumlah Komponen
Nama Komponen	Isi persentase
Nama Komponen	Isi persentase
Nama Komponen	Isi persentase
Nama Komponen	Isi persentase
BACK SAVE	

+ Input Mahasiswa

Output Identitas Matakuliah
Identitas Matakuliah

Output komponen nilai
KOMPONEN NILAI

Gambar 4.11. Mock Up *Input* Komponen Nilai

3) Tampilan *Input* Mahasiswa

Tampilan pada halaman *input* data mahasiswa seperti pada **Gambar 4.12** , pada rancangan tampilan *input* data mahasiswa ini terdapat bagian untuk *input* NIM dan nama mahasiswa yang ingin ditambahkan.

The interface for 'Input Mahasiswa' includes a sidebar menu with options: 'Proyek Perangkat Lunak', 'Manajemen Proyek Perangkat Lunak', 'Dasar Rekayasa Perangkat Lunak', and 'Keamanan Jaringan'. The main area features a 'NAMA MATAKULIAH' header, an 'Output komponen nilai' section, a 'KOMPONEN NILAI' box, and a table with columns 'No', 'NIM', and 'Nama'. Below the table is a form with 'Nama Matkul' and 'NIM' fields, each with an 'Isi' label, and 'CANCEL' and 'ADD' buttons.

Gambar 4.12. Mock Up Input Mahasiswa

4) Tampilan Nilai Mata Kuliah

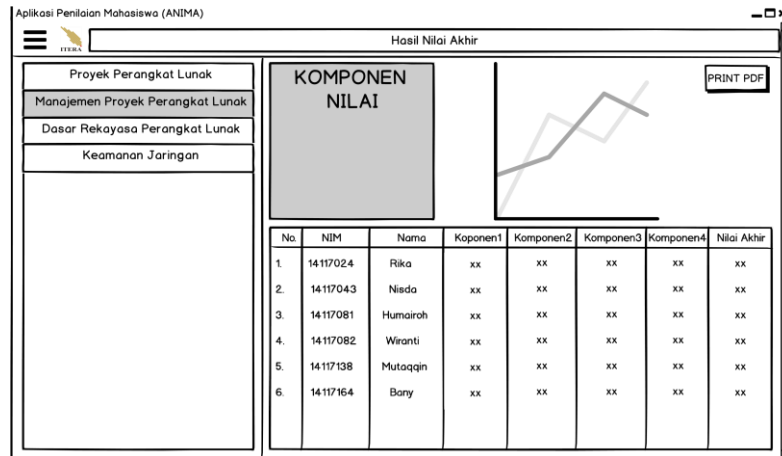
Tampilan pada halaman nilai mata kuliah seperti pada **Gambar 4.13** , pada rancangan tampilan nilai mata kuliah ini terdapat bagian untuk *input* nilai mahasiswa pada bagian tabel dan juga dapat memilih mata kuliah yang diinginkan pada bagian kiri tampilan.

The 'Nilai Mata Kuliah' interface features a sidebar menu with options: 'Proyek Perangkat Lunak', 'Manajemen Proyek Perangkat Lunak', 'Dasar Rekayasa Perangkat Lunak', and 'Keamanan Jaringan'. The main area includes a 'Matakuliah' header, a table with columns 'No', 'NIM', 'Nama', 'Koponen1', 'Komponen2', 'Komponen3', 'Komponen4', and 'Dst.', and a 'SAVE' button. The table contains data for five students: Rika (14117024), Nisda (14117043), Humairoh (14117081), Wiranti (14117082), and Mutaqqin (14117138).

Gambar 4.13. Mock Up Nilai Mata Kuliah

5) Tampilan Grafik Nilai

Tampilan pada halaman grafik nilai mata kuliah seperti pada **Gambar 4.14** , pada rancangan tampilan grafik nilai mata kuliah ini terdapat bagian untuk menampilkan grafik nilai di bagian kanan dan juga dapat memilih mata kuliah yang diinginkan pada bagian kiri tampilan.



Gambar 4.14. Mock Up Grafik Nilai