BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Di dalam era globalisasi seperti sekarang ini dimana informasi memegang peranan penting yang vital maka perlu dikembangkan sistem informasi secara tepat, cepat dan akurat. Menurut Thomas L. Friedman (2000) "Globalisasi memiliki dimensi ideologi dan teknologi. Dimensi teknologi adalah teknologi informasi yang menyatukan dunia".

Teknologi adalah suatu bentuk proses yang meningkatkan nilai tambah (Miarso, 2007). Kemajuan teknologi informasi komputer memungkinkan efisien, efektivitas, produktivitas, ketepatan, kecepatan, dan kualitas informasi yang sangat dibutuhkan dalam menyusun suatu perencanaan, pelaksanaan, dan pemantauan kegiatan kerja. Penggunaan sistem informasi yang berbasis web sangat diperlukan dalam suatu instansi/lembaga untuk menunjang kegiatan kerja supaya berjalan dengan baik dan lancar sesuai dengan sasaran yang di harapkan.

Dalam pelaksanaan penelitian ini, kami memilih sebuah institusi pendidikan yaitu Politeknik Kediri. Politeknik Kediri merupakan perguruan tinggi vokasi program kerjasama antar Direktorat Jenderal Pendidikan Tinggi dengan Pemerintah Kota kediri dalam rangka meningkatkan kapasitas pendidikan tinggi vokasi melalui pendirian politeknik melalui Naskah Kesepahaman (MoU) Dirjen Dikti Depdiknas dengan Pemerintah Kota kediri (Walikota Kediri). Untuk itu kami bermaksud untuk melakukan penelitian dalam rangka Proyek 1 pada Perguruan Tinggi Politeknik Kediri.

Politeknik Kediri merupakan Perguruan Tinggi Vokasi dengan jumlah mahasiswa keseluruhan lebih dari 600 mahasiswa. sehingga data data perlu terorganisir dengan baik, terutama dalam bidang rekapitulasi nilai mahasiswa. Politeknik Kediri masih menggunakan nilai keseluruhan tanpa mencantumkan nilai

detail dari mahasiswa. Sedangkan mahasiswa perlu mengetahui nilai secara keseluruhan untuk acuan semester selanjutnya.

Rekapitulasi nilai merupakan ringkasan atau rincian data nilai dari seorang pelajar selama satu semester. Disamping itu didalam sebuah lembaga pendidikan rekapitulasi nilai ini sangatlah penting. Sehingga, lembaga pendidikan perlu mengikuti perkembangan teknologi dan sistem informasi dengan melakukan peningkatan dan pengembangan pada sistem yang mereka miliki agar dapat terus mendukung kegiatan lembaga pendidikan di masa depan.

Melalu kinerja sistem informasi maka akan mempermudah penyelesaian suatu perkara. Sedangkan saat ini sistem rekapitulasi nilai di politeknik kediri masih kurang transparan. Hal tersebut mengakibatkan mahasiswa tidak mengetahui nilai untuk setiap pertemuan dalam satu semester.

Upaya mengatasi permasalahan yang telah didapatkan adalah dengan membangun aplikasi Sistem Informasi Rekapitulasi Nilai Mahasiswa Berbasis Web. aplikasi ini ditujukan untuk petugas atau staff, dosen dan mahasiswa agar dalam rekapitulasi nilai menjadi lebih praktis dan transparan. Selain itu, dengan aplikasi ini pengguna atau user dapat memasukan, mengedit, dan mencetak data.

1.2 RUMUSAN MASALAH

Rumusan masalah dalam penelitian ini adalah Bagaimana merancang dan membuat Sistem Informasi Rekapitulasi Nilai Mahasiswa Berbasis Web?

1.3 BATASAN MASALAH

Batasan masalah dalam penelitian ini adalah :

a. Berfokus pada rekapitulasi nilai yang bertempat di Politeknik Kediri
b. Admin tidak dapat mengedit data profil pada dosen atau mahasiswa.

1.4 MANFAAT PENELITIAN

Manfaat dalam penelitian ini yaitu mempermudah staf dan dosen dalam melakukan proses rekapitulasi dan mahasiswa dapat melihat nilai secara rapid an transparan.

1.5 TUJUAN PENELITIAN

Tujuan pembuatan aplikasi ini adalah merancang dan membuat sebuah Sistem Informasi Rekapitulasi Nilai Mahasiswa Berbasis Web agar admin atau petugas dapat lebih mudah dalam melalukan proses rekapitulasi nilai seperti memasukkan, mengedit, serta mencetak data mahasiswa dan dosen melalui form yang telah disediakan dan sudah terkomputerisasi.

1.6 METODOLOGI PENELITIAN

Dalam Pembuatan tugas Proyek 1 untuk memperoleh data-data yang valid dalam penelitian, dibutuhkan beberapa metode penelitian, adapun metode penelitian yang dimaksud antara lain:

1.6.1 Perencanaan Sistem

Dalam tahapan ini, penulis mulai mengidentifikasi dan mengevaluasi permasalahan–permasalahan yang terjadi serta mendefinisikan kebutuhan-kebutuhan yang dibutuhkan untuk membangun aplikasi sistem informasi tersebut.

1.6.2 Analisis Kebutuhan

Dalam tahapan ini penulis mulai mengidentifikasikan kebutuhan–kebutuhan apa saja yang dibutuhkan untuk membuat aplikasi sistem informasi tersebut.

1.6.3 Perancangan Sistem

Dalam tahapan ini mulai dibuat desain sistem untuk menentukan proses dan data apa yang dibutuhkan dalam membuat aplikasi tersebut. Hasil perancangan sistem dapat berupa *flowchart*, DFD (*Data Flow Diagram*) atau ERD (*Entity Relationship Diagram*) serta perancangan dalam bentuk lain untuk memudahkan dalam membuat sebuah aplikasi.

1.6.4 Implementasi dan Pembahasan

Dalam tahapan ini merupakan tahapan dimana hal–hal yang dibuat dalam perancangan sistem di implementasikan menjadi sebuah kode (program) yang siap dioperasikan.

1.6.5 Uji Coba

Tahapan selanjutnya adalah tahapan uji coba. Tahapan uji coba dilakukan untuk mengetahui apakah proses–proses yang sudah diimplementasikan pada tahap sebelumnya sudah sesuai atau belum, jika belum sesuai perlu dilakukan perbaikan agar semua sistem yang sudah dibuat dapat berjalan dengan baik dan digunakan sebagaimana mestinya.

1.7 SISTEMATIKA PENULISAN

Di dalam laporan ini, kami menyusun pembahasan materi menjadi enam bab. Materi tersebut disusun dengan sistematika penulisan sebagai berikut :

1.7.1 BAB I Pendahuluan

Pada bab ini berisikan mengenai latar belakang, rumusan masalah, tujuan penelitan, batasan masalah, manfaat penelitian, metode penelitian serta sistenatika penelitian.

1.7.2 BAB II Landasan Teori

Bab ini akan membahas mengenai landasan teori yang berhubungan dengan permasalahan dan penyelesaiannya yang dibahas dalam laporan ini.

1.7.3 BAB III Analisa dan Perancangan

Bab ini berisi analisis yang terdiri dari gambaran sistem, spesifikasi alat berupa *software* dan *hardware* yang dibutuhkan. Dan perancangan aplikasi yang terdiri dari rancangan DFD (*Data Flow Diagram*) mulai dari level 0, level 1, *database,* ERD, dan desain *interface* program.

1.7.4 BAB IV Implementasi dan Pembahasan

Bab ini berisi implementasi yang terdiri dari langkah kerja dan *user interface* program. Hasil penelitian terdiri dari tabel tingkat pengenalan dan grafik tingkat pengenalan. Serta uji coba yang menampilkan uji coba program yang telah dibuat beserta keterangan hasil uji coba yang telah dilakukan.

1.7.5 BAB V Kesimpulan dan Saran

Bab ini berisi kesimpulan dan saran terhadap pengembangan dari penelitian tugas Laporan ini agar menjadi program yanag lebih sempurna.

BAB II

LANDASAN TEORI

Bab ini akan membahas mengenai teori-teori yang berhubungan dengan permasalahan dan penyelesaian pada penelitian serta penelitian sebelumnya yang relevan dengan penelitian ini.

2.1 SISTEM INFORMASI

Menurut John F. Nash (1995) Sistem informasi menjadi alat peyajian data informasi dengan berbagai bentuk agar bermanfaat untuk penerima. Dan memiliki tujuan yaitu membrikan informasi dalam perncanaan, memulai, pengorganisasian, operasional perusahaan dalam pelayanan sinergi organisai dalam proses pengendalian pengambilan keputusan.

2.2 PENELITIAN TERDAHULU

NO	NAMA JURNAL	PENULIS	INTISARI
1.	SISTEM INFORMASI PENGOLAHAN DATA NILAI SISWA BERBASIS WEB PADA SEKOLAH MENENGAH ATAS (SMA) MUHAMMADIYAH PACITAN	Susy Kusuma Wardani	Permasalahan yang ada adalah SMA MUHAMMADIYAH PACITAN mengalami Kendala yaitu Kesulitan dalam menganalisis nilai siswa, dan sering terjadi kesalahan dalam membuat rata-rata nilai sehingga proses pengolahan nilai masih dilakukan dengan cara konvensional yaitu masing-masing guru bidang study mengumpulkan nilai siswa yaitu sumbernya dari nilai tugas, nilai ulangan harian dan nilai ulangan semester kepada wali

			kelas. Hingga dibuatnya lah Sistem Informasi nilai berbasis web yang dapat mengolah nilai siswa, pencarian, meng-update, menyimpan, rekap nilai siswa dan laporan nilai yang dapat di akses dengan mudah dan efektif.
2.	SISTEM INFORMASI PENILAIAN AKADEMIK SISWA KURIKULUM 2013 BERBASIS WEB DI SMAN 1 TRENGGALEK	Esti Merindasari, Triyanna Widyaningtyas, M. Zainal Arifin	Produk yang telah dikembangkan adalah sistem informasi akademik siswa berbasis web di SMAN 1 TRENGGALEK dengan mengadopsi pedoman penilaian Kurikulum 2013. Sistem ini bertujuan untuk memudahkan proses penilaian akademik siswa. Dengan aplikasi yang dikembangkan, proses penilaian akademik siswa yang dilakukan oleh guru dapat diselesaikan dengan mudah. Akurasi data yang diolah menjadi informasi dapat menjadi data dukung dalam mengambil keputusan terkait dengan akademik siswa.
3.	PEMBANGUNAN SISTEM INFORMASI PENILAIAN HASIL BELAJAR SISWA	Nursahid, Berliana Kusuma Riasti, Bambang Eka Purnama	Sistem informasi penilaian di SMA NEGERI 2 REMBANG saat ini masih dalam cara

	SEKOLAH MENENGAH ATAS (SMA) NEGERI 2 REMBANG BERBASIS WEB		konvensional dengan mencatat pada buku daftar nilai kemudian direkap hasilnya secara manual. Mulai dari nilai kognitif yang meliputi nilai ulangan harian, ulangan tengah semester, ulangan semester, nilai psikomotorik, nilai afektif yang diolah kemudian hasilnya di tulis leger secara manual. Sehingga perlu terbangunnya suatu sistem informasi penilaian di SMA NEGERI 2 REMBANG yang dapat membantu proses penilaian yang dilakukan oleh bapak, ibu guru mata pelajaran sesuai dengan kurikulum yang diterapkan disekolah tersebut.

2.3 INTERNET

1. Definisi Internet

Internet merupakan sekumpulan jaringan komputer yang saling terhubung satu sama lain secara fisik dan juga memiliki kemampuan untuk membaca dan menguraikan berbagai protokol komunikasi tertentu yang sering kita kenal dengan istilah *Internet Protocol* (IP) serta *Transmission Control Protocol* (TCP) (Allan,2005).

2. Internet Service

1.) *World Wide Web* (WWW)

World Wide Web (WWW) atau dikenal dengan sebutan web saja, merupakan sistem yang menyebabkan pertukaran data di internet menjadi mudah dan efesien (Ellsworth, 1997). Web merupakan pada

awalnya ruang informasi dalam internet dengan menggunakan teknologi *hypertext,* pengguna dituntun untuk menemukan informasi dengan mengikuti *link* yang disediakan dalam dokumen web yang ditampilkan oleh web browser.

Kini internet identik dengan web, karena kepopuleran web sebagai standar *interface* pada layanan-layanan yang ada pada internet, dari awal sebagai penyedia informasi, hingga kini digunakan untuk komunikasi lewat e-mail sampai dengan *chatting,* bahkan melakukan transaksi bisnis. Web memudahkan pengguna komputer untuk berinteraksi dengan pelaku internet lainnya dan menelusuri informasi dalam internet.

2.) *Uniform Resource Locators* (URL)

URL yang merupakan singkatan dari *Uniform Resource Locators* adalah cara standar yang digunakan untuk menentukan situs atau halaman pada internet. URL merupakan cara standar untuk menampilkan informasi tentang jenis, isi , dan lokasi file: nama file, lokasi komputer di internet, letak file di dalam komputer, dan protocol internet yang digunakan untuk mengakses file itu. Internet itu sangat besar, merupakan interkoneksi terdistribusi, tempat yang sangat tidak seragam, dan URL menstandarkan dari keanekaragaman ini (Ellsworth, 1997). Format umum dari sebuah URL adalah *Protocol Transfer*: //nama_host/Path/nama_file.

3.) *Hyper Text Markup Language*(HTML)

HTML merupakan singkatan dari *Hyper Text Markup Languange*.HTML digunakan untuk membangun suatu halaman web. HTML bukan suatu bahasa pemrograman melainkan suatu bahasa markup. HTML digunakan untuk melakukan markup atau penandaan terhadap suatu dokumen teks. Tanda tersebut gigunakan untuk menentukan format atau style dari teks yang ditandai. Dokumen HTML disebut sebagai *markup languange* karena mengandung tanda-tanda

(*tag*) yang didahului tanda "<" dan diakhiri dengan tanda ">" yang digunakan untuk menentukan tampilan suatu teks dalam suatu dokumen.

2.4 PEMROGRAMAN WEB

1. PHP

PHP atau singkatan dari *Personal Home Page* merupakan sebuah bahasa skrip yang tertanam dalam HTML untuk dieksekusi bersifat *server side.*

2. *Framework* Laravel

Framework Laravel merupakan suatu struktur konseptual dasar yang digunakan untuk memecahkan atau menangani suatu masalah yang kompleks. Singkatnya *framework* Laravel adalah wadah atau kerangka kerja dari sebuah *website* yang akan dibangun. Dengan menggunakan kerangka tersebut waktu yang digunakan dalam membuat *website* lebih singkat dan memudahkan dalam melakukan perbaikan. Laravel adalah *framework* berbasis PHP yang sifatnya *open source*, dan menggunakan konsep *model – view – controller*. Laravel berada dibawah lisensi MIT *License* dengan menggunakan Github sebagai tempat berbagi code menjalankannya (Naista, 2017).

3. MySQL

MySQL merupakan sebuah basis data yang mengandung satu atau beberapa kolom. Tabel terdiri atas sejumlah basis dan setiap baris mengandung satu atau beberapa kolom. Didalam PHP telah menyediakan fungsi untuk koneksi ke basis data dengan sejumlah fungsi untuk pengaturan baik menghubungkan maupun memutuskan koneksi *server database* MySQL sebagai sarana untuk mengumpulkan informasi.(Yeni Kustiyahningsih, Devie Rosa Anamisa, 2010: 145-146).Keunggulan dari MySQL adalah :

a.) Salah satu jenis *software* yang *protable, Software portable* ini berarti MySQL bisa dijalankan untuk mengolah *database multi platform*. Sistem operasi Windows, Linux, Mac, dan sebagainya bisa menggunakan DBMS MySQL ini, sehingga hal ini membuat MySQL menjadi lebih baik dari segi efisiesnsi dan juga fungsionalitas yang lebih baik.

b.) Bisa dijalankan oleh banyak *user* dalam satu waktu tanpa perlu mengalami kendala seperti crash, dan semacamnya.

2.5 XAMPP

XAMPP adalah sebuah *software* yang memiliki tugas untuk menjalankan *website* berbasis PHP dengan menggunakan pengolah data berupa MySQL pada komputer lokal. XAMPP ini juga dikenal sebagai Cpanel server yang mampu melakukan *preview website* tanpa harus tersambung dengan jaringan internet ataupun online.

2.6 DFD (*Data Flow Diagram)*

Menurut (Kendall & Kendall, 2003), *data flow diagram* adalah grafik yang menggambarkan pandangan sejauh mungkin mengenai *input,* proses dan *output* sistem, yang berhubungan dengan *input*, proses dan *output* dari model sistem secara umum. DFD merupakan alat bantu dalam menggambarkan atau menjelaskan sistem yang sedang berjalan logis.

Fungsi *Data Flow Diagram* (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD adalah alat pembuatan model yang memberikan penekanan hanya pada fungsi sistem dan juga merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yang mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program.

Simbol yang digunakan dalam membuat DFD antara lain kesatuan luar, arus, proses dan simpan data. Kesatuan luar merupakan kesatuan lingkungan di luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem. Data Arus mengalir diantara proses, simpanan data dan kesatuan luar. Arus data menunjukkan arus dari data yang dapat berupa masukan untuk sistem

atau hasil dari proses sistem yang ditunjukkan dengan simbol panah. Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk menghasilkan arus data yang akan keluar dari proses. Simpan data merupakan simpanan dari data yang dapat berupa suatu file atau database di sistem komputer, suatu arsip atau catatan manual, suatu kotak tempat data di meja seseorang, suatu tabel acuan manual, suatu agenda atau buku .

Ketentuan-ketentuan dalam penggambaran DFD yaitu :

1. Diagram *Context* : menggambarkan satu lingkaran besar yang dapat mewakili seluruh proses yang terdapat di dalam suatu sistem. Merupakan tingkatan tertinggi dalam DFD dan biasanya diberi nomor 0 (nol). Semua entitas eksternal yang ditunjukkan pada diagram *context* berikut aliran-aliran data utama menuju dan dari sistem. Diagram ini sama sekali tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan.
2. Diagram Nol (diagram level-1) : merupakan satu lingkaran besar yang mewakili lingkaran-lingkaran kecil yang ada di dalamnya. Merupakan pemecahan dari diagram Konteks ke diagram Nol. di dalam diagram ini memuat penyimpanan data.
3. Diagram Rinci : merupakan diagram yang menguraikan proses apa yang ada dalam diagram Nol. Berikut simbol-simbol bagan DFD pada tabel 2.1 :

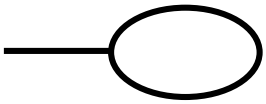
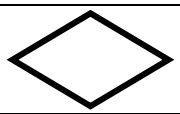
Tabel 2.1 Simbol-Simbol DFD

Notasi Yourdon / DeMarco	Notasi Game & Sarson	Keterangan
		Simbol Entitas eksternal / terminator menggambarkan asal atau tujuan data di luar sistem.
		Simbol lingkaran menggambarkan entitas atau proses dimana aliran data masuk ditransformasikan ke aliran data keluar.
		Simbol aliran data menggambarkan aliran data.
		Simbol file menggambarkan tempat data di simpan.

2.7 ERD (*Entity Relational Diagram*)

(ERD) merupakan suatu model data yang dikembangkan berdasarkan objek." *Entity Relationship Diagram* (ERD) digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. *Entity Relationship Diagram* (ERD) didasarkan pada suatu persepsi bahwa *real world* terdiri atas obyek-obyek dasar tersebut. Penggunaan *Entity Relationship Diagram* (ERD) relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis sistem, *Entity Relationship Diagram* (ERD) berguna untuk memodelkan sistem yang nantinya, basis data akan di kembangkan (Sutanta,2011). ERD memiliki beberapa komponen dan dideklarasikan ke dalam simbol-simbol di bagan ERD pada tabel 2.2 :

Tabel 2.2Komponen ERD

Simbol / Komponen ERD	Keterangan
	Entitas, adalah segala sesuatu yang dapat digambarkan oleh data. Dapat berupa oreng, tempat, kejadian atau konsep informasi yang direkam.
	**Atribut**, merupakan pendeskripsian karakteristk dari entitas. Disebut juga sebagai data *field* yang menggambarkan suatu entitas. Atribut terbagi menjadi 2, yaitu : - *Simple Attribute*, misal kd_kategori - *Composite Attribute*, misal nama_kategori
	**Relasi**, merupakan penghubung antara suatu entitas dengan entitas lain di dalam sebuah sistem.

BAB III

ANALISIS PERANCANGAN SISTEM

3.1 Analisis Kebutuhan

Analisis kebutuhan sistem bertujuan untuk mengidentifikasi kebutuhan dari sistem baik berupa kebutuhan yang bersifat fungsional maupun non fungsional yang dibutuhkan untuk membangun sebuah Rekapitulasi Nilai Mahasiswa. Dengan adanya analisis kebutuhan sistem penulis menentukan langkah terbaik yang dapat diambil untuk membangun sebuah sistem berdasarkan kebutuhan dari sistem yang dibuat.

3.1.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan akan proses-proses yang akan dilakukan oleh sistem. Kebutuhan fungsional pada sistem ini adalah sebagai berikut :

1. Admin

Admin dapat melakukan berbagai aktifitas diantaranya sebagai berikut:

1. Admin dapat mengelola data semua user.
2. Admin dapat mengelola data semua nilai mahasiswa.
3. Admin dapat mengelola data semua mata kuliah.

2. Dosen

Dosen dapat melakukan berbagai aktifitas diantaranya sebagai berikut:

1. Dosen dapat mengelola nilai mahasiswa.
2. Dosen dapat melihat profil akun miliknya.
3. Dosen dapat mencari data nilai mahasiswa.

3. Mahasiswa

Mahasiswa dapat melakukan aktifitas diantaranya sebagai berikut:

1. Mahasiswa dapat melihat nilai mata kuliah per-semester.
2. Mahasiswa dapat melihat profil akun miliknya.

3.1.2 Analisis Perangakat

Dalam membuat aplikasi survey berbasi website, dibutuhkan beberapa komponen utama yang mencangkup perangkat keras (*hardware*) dan perangkat lunak (*software*).

3.1.3 Kebutuhan Perangkat Keras (*Hardware*)

Spesifikasi perangkat keras komputer yang digunakan untuk pembuatan program ini adalah:

1. Komputer dengan *processor* AMD A8-7410 APU
2. Komputer dengan *memory* RAM 4 GB.
3. Komputer dengan *harddisk* 500GB sebagai media penyimpanan.

3.1.4 Perangakat Lunak (*Software*)

Selain perangkat lunak, dalam membangun sebuah Sistem Informasi Reakapitulasi Nilai Mahasiswa juga dibutuhkan perangkat lunak untuk membantu berjalannya sistem. Perangkat lunak yang digunakan untuk membangun sistem informasi adalah sebagai berikut :

1. Sistem Operasi : Windows Xp, Windows 2007,Windows 2008, Windows 2010
2. Aplikasi SQL Server : XAMPP ,SQLyog
3. Web Browser : Mozilla Firefox, Chrome
4. Aplikasi pendukung : Sublime, notepad ++

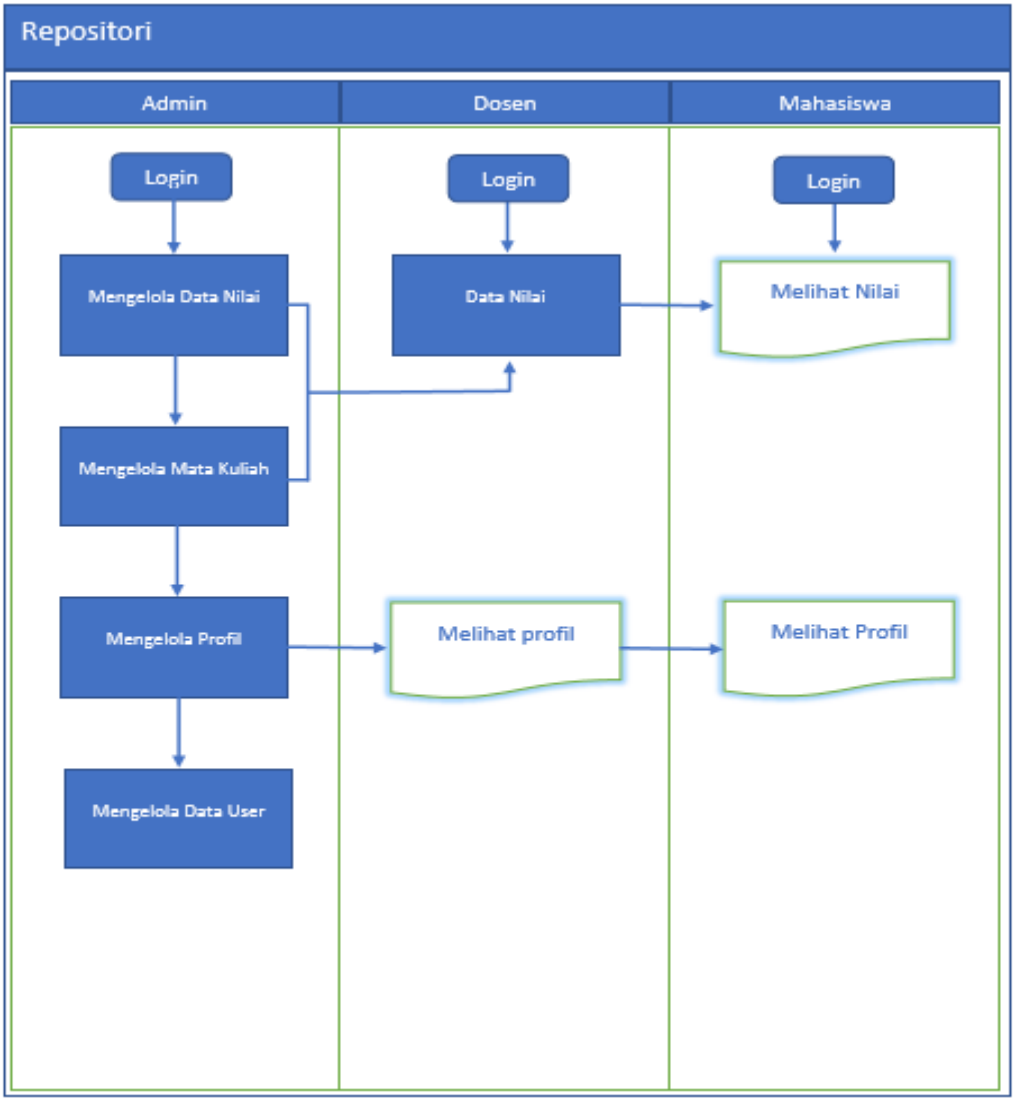
3.2 Perancangan

3.2.1 Perancangan Proses Bisnis

Proses bisnis perlu dipahami dengan baik karena proses bisnis memberikan gambaran apa saja harus dilakukan untuk bisnis yang akan dibuat sehingga penulis dapat merumuskan dan menata alur bisnis tersebut dalam suatu sistem dengan rapi serta agar penulis tidak salah dalam memahami langkah-langkah dari pembuatan sistem. Maka dari itu penulis harus paham dahulu apa yang

sebenarnya maksud dan tujuan proses bisnis tersebut. Proses bisnis merupakan kumpulan aktifitas atau pekerjaan yang saling terkait untuk menyelesaikan suatu masalah tertentu atau yang menghasilkan produk.

Pada Gambar 3.1 dijelaskan bahwa admin melakukan login, mengelola data dosen dan mengelola data mahasiswa. Dosen dapat mengelola nilai mahasiswa dan melihat profil. Mahasiswa dapat melihat nilai per- mata kuliah dan melihat IPK. Perancangan proses bisnis dari rekapitulasi nilai ini dapat ditunjukkan di gambar 3.1.



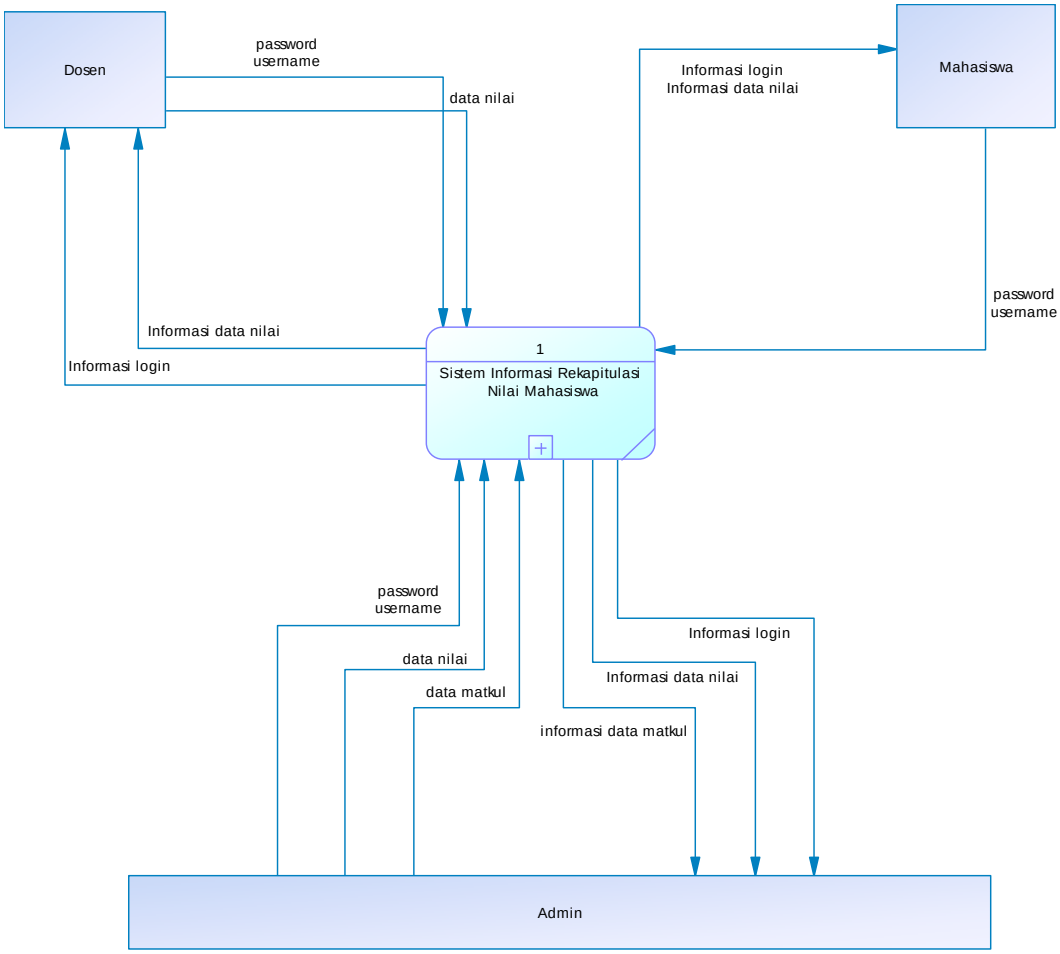
Gambar 3.1 Perancangan Proses Bisnis

3.2.2 Perancangan Diagram Context dan DFD level 0

DFD (*Data Flow Diagram*) merupakan diagram alir yang dipresentasikan dengan lambang-lambang tertentu. Dengan adanya DFD maka penulisan suatu program akan menjadi lebih mudah dalam pelaksanaannya, karena menggunakan lambang-lambang yang bersifat standar yang ditetapkan secara umum dalam penulisan desain. Penulisan DFD ini bertujuan untuk menggambarkan sistem yang sedang berjalan. Berikut ini adalah DFD pada Sistem Informasi Rekapitulasi Nilai Mahasiswa Politeknik Kediri diantaranya :

1. Diagram Context

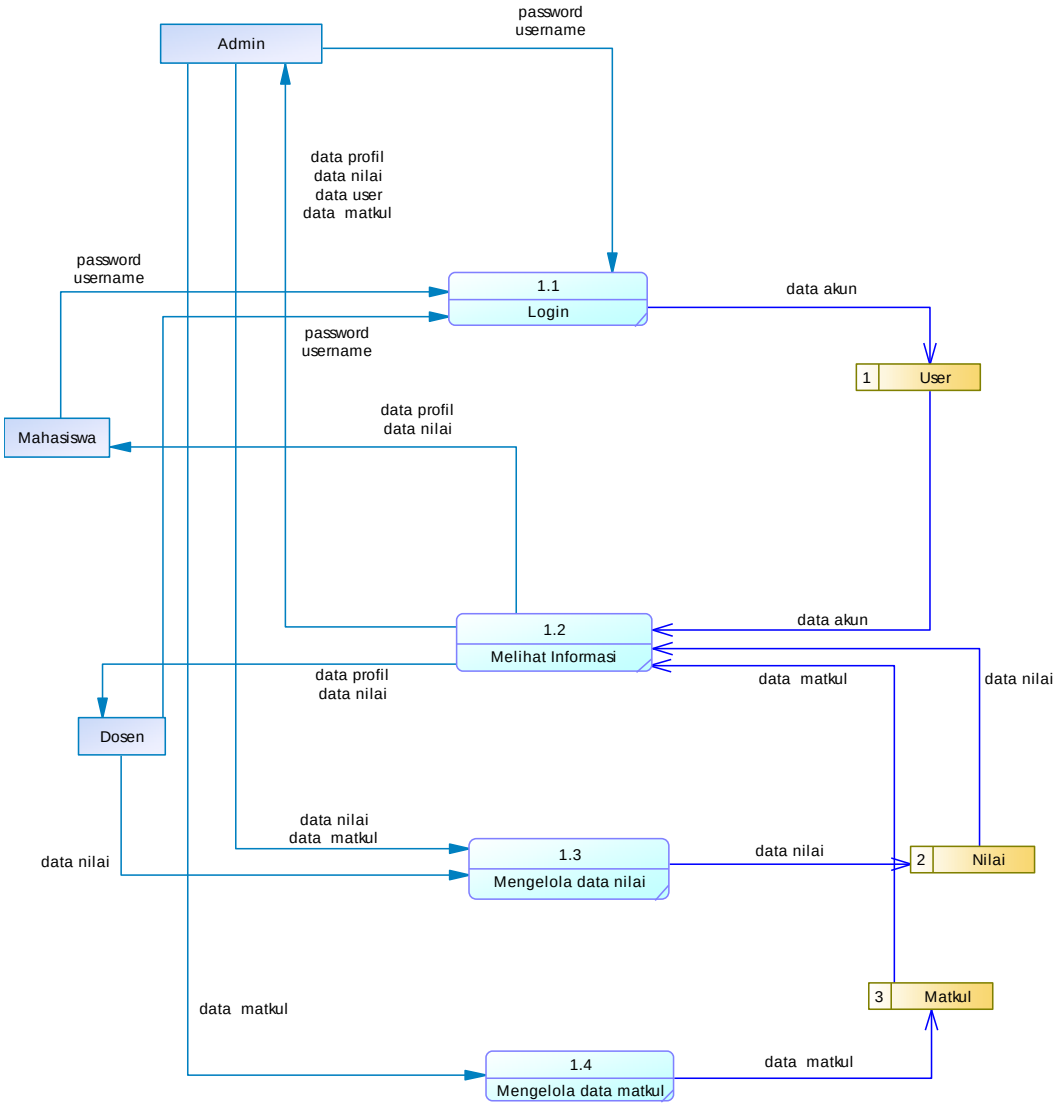
   Diagram context adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram context merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke dalam sistem atau output dari sistem yang memberi gambaran tentang keseluruhan sistem. Pada Gambar 3.2 Data flow diagram level 0 pada Rekapitulasi Nilai Mahasiswa Berbasis Web yang terhubung oleh sistem, yaitu admin, dosen dan mahasiswa akan di tunjukkan pada Gambar 3.2 :



Gambar 3.2 Diagram Context

2. DFD Level 0

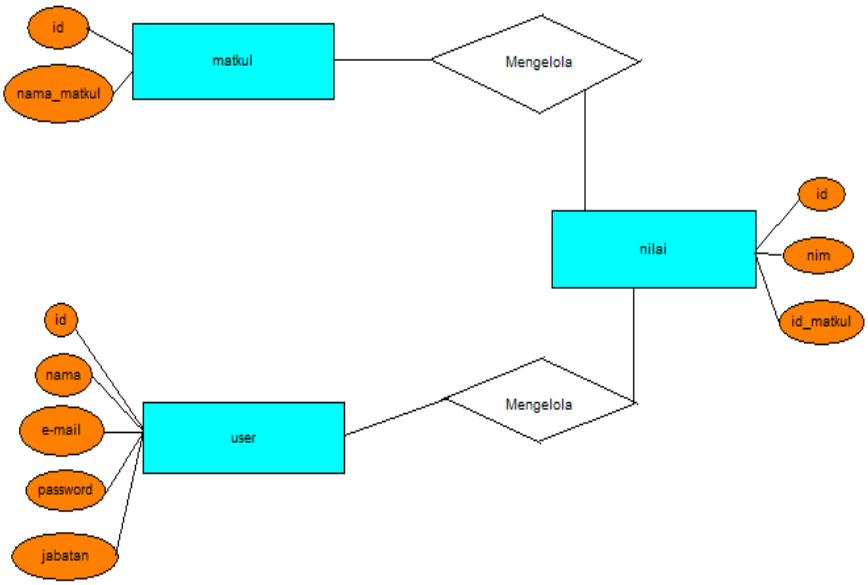
DFD level 0 menggambarkan tentang seluruh input – output yang telah di pecah-pecah menjadi beberapa bagian dalam sistem aplikasi. Dalam level ini akan dijelaskan proses kerja yang dilakukan oleh setiap entitas terhadap sistem di Rekapitulasi Nilai Mahasiswa Berbasis Web yang akan di tunjukkan pada Gambar 3.3



Gambar 3.3 DFD Level 0

3.2.3 Perancangan ERD (Entity Relationship Diagram)

Perancangan ERD terdiri dari data nilai, matkul dan user. ERD pada penelitian ditunjukkan pada gambar 3.4.



Gambar 3.4 ERD Rekapitulasi Nilai Mahasiswa Berbasis Web

Pada Gambar 3.4 dapat dilihat bahwa terdapat 5 entitas, adapun entitas tersebut adalah:

1. Entitas nilai memiliki atribut id, nim, dan id_matkul.
2. Entitas matkul memiliki atribut id dan nama_matkul.
3. Entitas user memiliki atribut id, nama, e_mail, password, dan jabatan.

3.2.4 Perancangan Data Base

Aplikasi ini membutuhkan suatu *database* yang berguna untuk penyimpanan seluruh data yang diperlukan dalam aplikasi yang dirancang. Seluruh data yang diperlukan dalam aplikasi ini akan disimpan dalam *database*. *Database* yang digunakan yaitu *MySQL*. Pada Rekapitulasi Nilai Mahasiswa menggunakan *database* yang berisi 3 tabel. Yaitu tabel data_admin, data_dosen, dan data_mahasiswa.

1. Tabel Admin

   Tabel Admin digunakan untuk menyimpan data dari admin. Pada tabel admin terdapat id, nim, nama, mata kuliah, nilai dan alamat. Struktur tabel user seperti tabel 3.5 berikut.

Tabel 3.5 Data_admin

No	Nama Field	Tipe Data	Keterangan
1	id	int(10)	Primary Key, Auto Increment
2	nim	int(10)	
3	nama	varchar(100)	
4	mata kuliah	varchar(100)	
5	nilai	int(10)	
6	alamat		

Penjelasan dari table 3.5 :

1. id adalah *primary key* dari tabel admin.
2. nim digunakan untuk menyimpan data nim dari admin.
3. nama digunakan untuk menyimpan data jurusan dari admin
4. mata kuliah digunakan untuk menyimpan data mata kuliah dari admin.
5. nilai digunakan untuk menyimpan data nilai yang dari admin.

2. Tabel Dosen

Tabel Dosen digunakan untuk menyimpan data dari dosen . Pada tabel terdapat id, nim, matkul, sks dan nilai. Struktur tabel mata_kuliah seperti tabel 3.6 berikut.

Tabel 3.6 Data_dosen

No	Nama Field	Tipe Data	Keterangan
1	nim	int(10)	Primary Key, Auto Increment
2	matkul	varchar(100)	

3	sks	int(10)	
4	nilai	Int(10)	

Penjelasan dari table 3.6 :

1. nim adalah *primary key* dari tabel dosen.
2. matkul digunakan untuk menyimpan data matkul dari tabel dosen.
3. sks digunakan untuk menyimpan data sks dari tabel dosen.
4. nilai digunakan untuk menyimpan data nilai dari tabel dosen

3. Tabel Mahasiswa

Tabel mahasiswa digunakan untuk menyimpan data dari nilai. Pada tabel mahasiswa terdapat id, nim, matkul, dan nilai. Struktur tabel nilai seperti tabel 3.7 berikut.

Tabel 3.7 Nilai

No	Nama Field	Tipe Data	Keterangan
1	id_	int(10)	Primary Key, Auto Increment
2	nim	int(10)	
3	matkul	int(10)	
4	nilai	int(10)	

Penjelasan dari table 3.7 :

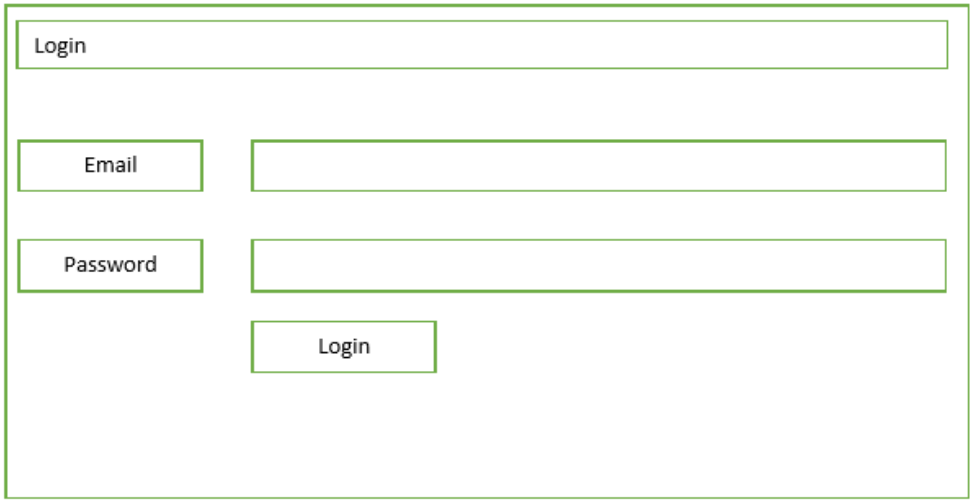
1. id adalah *primary key* dari tabel mahasiswa.
2. nim digunakan untuk menyimpan data nim dari tabel mahasiswa.
3. matkul digunakan untuk menyimpan data id matkul dari tabel mahasiswa.
4. nilai digunakan untuk menyimpan data nilai dari tabel mahasiswa.

3.2.5 Perancangan Antar Muka

Rancangan antar muka diperlukan dalam proses pembuatan aplikasi halaman situs web karena bagian ini langsung berhubungan dengan user. Berikut ini rancangan antar muka dari Sistem Informasi Rekapitulasi Nilai Mahasiswa diantaranya sebagai berikut :

a. Rancangan antar muka halaman *Login*

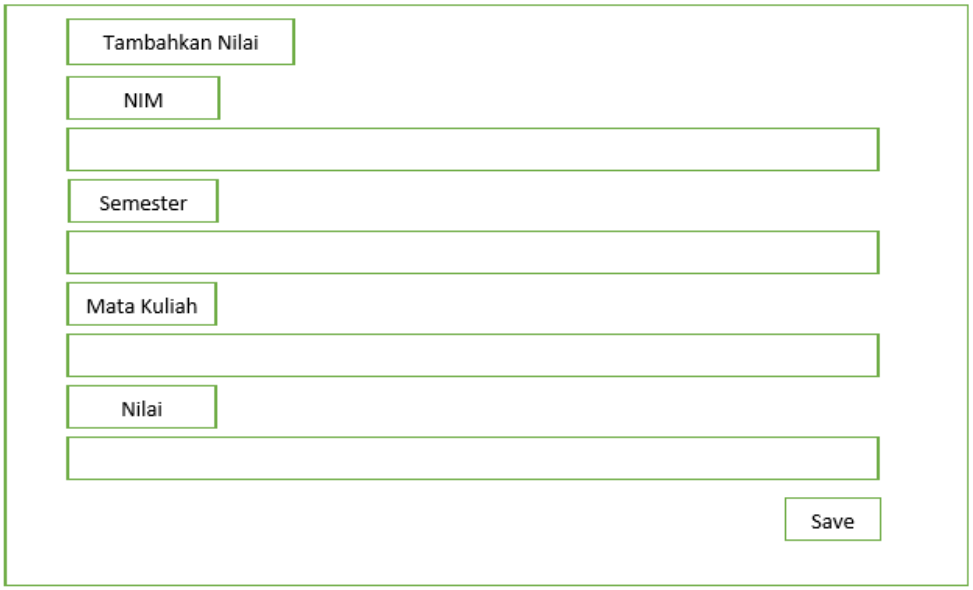
Halaman *login* merupakan tampilan sebelum masuk beranda seperti yang di tunjukkan pada Gambar 3.8



Gambar 3.8 Rancangan antar muka halaman *Login*

b. Rancangan antar muka halaman Admin Tambah Nilai

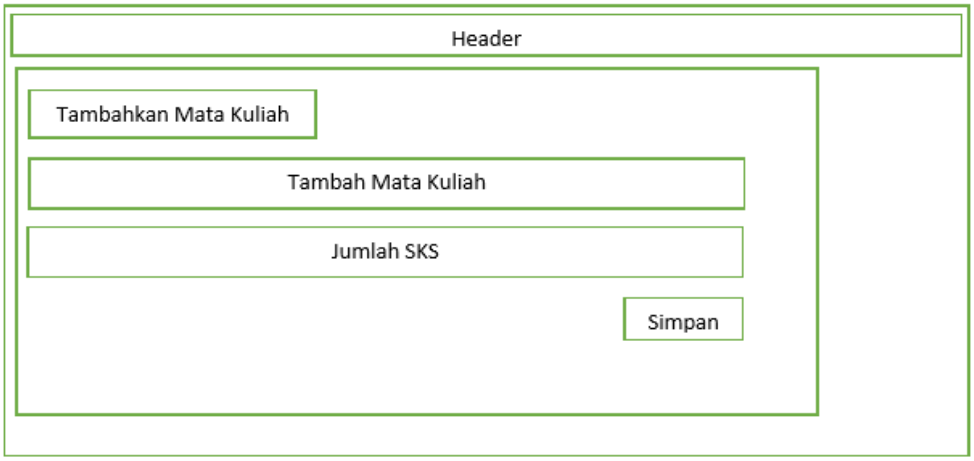
Halaman admin berisi input data untuk nilai mahasiswa seperti yang di tunjukkan pada Gambar 3.9.



Gambar 3.9 Rancangan antar muka halaman Admin Tambah Nilai

c. Rancangan antar muka halaman Admin Tambah Mata Kuliah

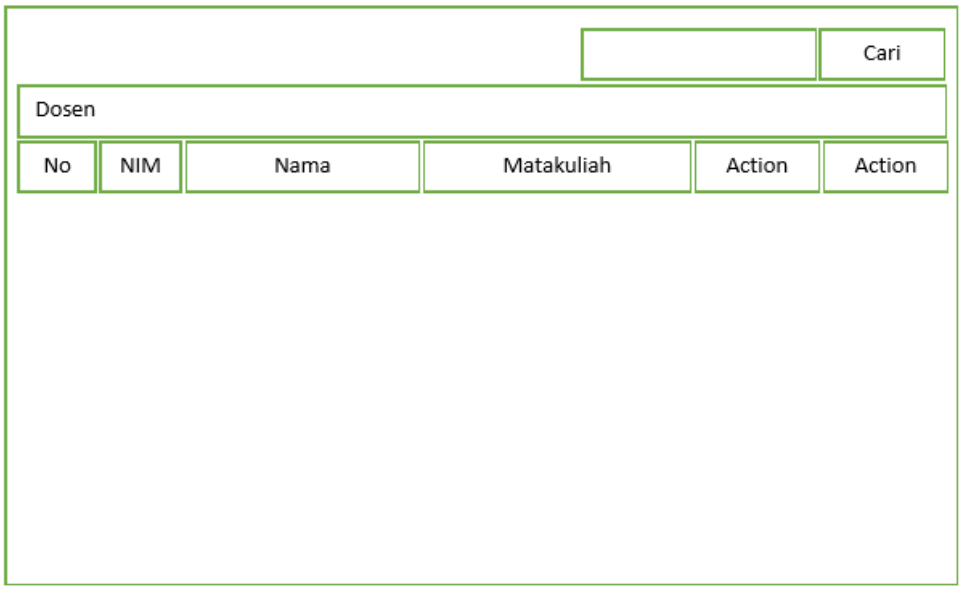
Halaman admin berisi input data untuk mata kuliah dan sks seperti yang di tunjukkan pada Gambar 3.10.



Gambar 3.10 Rancangan antar muka halaman Admin Tambah Mata Kuliah

d. Rancangan antar muka halaman Dosen

Halaman dosen berisi input data untuk nilai mahasiswa seperti yang di tunjukkan pada Gambar 3.11.



Gambar 3.11 Rancangan antar muka halaman Dosen

e. Rancangan antar muka halaman Mahasiswa

Halaman mahasiswa untuk melihat nilai mata kuliah seperti yang di tunjukkan pada Gambar 3.12.

Gambar 3.12 Rancangan antar muka halaman Mahasiswa

3.2.6 Rancangan Uji Coba Sistem

Rancangan Uji Coba Sistem diperlukan dalam prose pembuatan aplikasi halaman situs web karena bagian ini merupakan bagian yang penting sebgai rancangan alur sistem yang nantinya akan dibuat.

Tabel 3.13 Rancangan Uji Coba Sistem

No.	Objek Pengujian	Skenario Pengujian	Hasil Yang Diharapkan
1	Daftar (Dosen atau Mahasiswa)	Membuka halaman daftar dengan mengisikan data diri sesuai yang diperlukan aplikasi untuk sebagai *user.*	Aplikasi menyimpan data login member ke dalam tabel data_user.
2	*Login*(Admin, Dosen, dan Mahasiswa)	Membuka halaman login dengan melakukan *login* dengan hak akses sebagai admin, dosen, atau mahasiswa.	Aplikasi memproses l*oading form login* dan menuju ke halaman beranda yaitu halaman admin, dosen atau mahasiswa.
3	Data *User* (Admin)	Admin membuka menu "*user*" yang selanjutnya melakukan penambahan, perubahan dan penghapusan data semua *user.*	Aplikasi menyimpan data *user* ke dalam tabel *user.*
4	*User* Awal (Dosen)	Terbuka ketika dosen melakukan login sebagai *user.*	Aplikasi memproses *keyword* yang dimasukkan oleh dosen dan menampilkan halaman.

5	*User* Awal (Mahasiswa)	Terbuka ketika mahasiswa melakukan login sebagai *user.*	Aplikasi memproses *keyword* yang dimasukkan oleh mahasiswa dan menampilkan halaman.
6	*User* Tambah (Dosen)	Membuka halaman dosen dan melakukan melakukan upload nilai.	Aplikasi memproses *keyword* yang dimasukkan oleh user dan menampilkan data.
7	*User* Ubah(Dosen)	Membuka halaman halaman dosen dan melakukan ubah nilai.	Aplikasi menyimpan data nilai ke dalam tabel nilai.
8	*User* (Mahasiswa)	Membuka halaman mahasiswa dan melihat nilai.	Aplikasi memproses menampilkan data.

BAB IV

IMPLEMENTASI DAN PEMBAHASAN

4.1 IMPLEMENTASI SISTEM INFORMASI REKAPITULASI NILAI MAHASISWA

Proses implementasi dari perancangan aplikasi yang dilakukan pada bab sebelumnya akan dijelaskan pada bab ini. Implementasi bertujuan untuk menerjemahkan keperluan perangkat lunak ke dalam bentuk sebenarnya yang dimengerti oleh komputer atau dengan kata lain tahap implementasi ini merupakan tahapan lanjutan dari tahap perancangan yang sudah dilakukan. Dalam tahap implementasi ini akan dijelaskan mengenai file-file yang digunakan dalam membangun sistem, tampilan aplikasi beserta potongan-potongan *script* program untuk menampilkan halaman.

4.1.1. Basis Data

Basis data merupakan pusat data yang digunakan oleh aplikasi di mana data tersebut akan ditampilkan sesuai dengan kebutuhan tiap pengguna. Berikut struktur tabel pada basis data.

a. Tabel *Users*

Tabel *user* difungsikan sebagai tempat untuk menyimpan data users berupa identitas *user*. Pembuatan tabel dapat dilakukan dengan contoh *script* membuat tabel *users* di bawah ini.

```
CREATE TABLE `users` (
 `id` INT(10) UNSIGNED NOT NULL,
 `ni` INT(11) NOT NULL,
 `name` VARCHAR(191) COLLATE utf8mb4_unicode_ci NOT NULL,
 `jurusan` VARCHAR(20) COLLATE utf8mb4_unicode_ci NOT NULL,
 `email` VARCHAR(191) COLLATE utf8mb4_unicode_ci NOT NULL,
 `email_verified_at` TIMESTAMP NULL DEFAULT NULL,
 `password` VARCHAR(191) COLLATE utf8mb4_unicode_ci NOT NULL,
 `alamat` TEXT COLLATE utf8mb4_unicode_ci,
```

```
 `remember_token` VARCHAR(100) COLLATE utf8mb4_unicode_ci DEFAULT
NULL,
 `created_at` TIMESTAMP NULL DEFAULT NULL,
 `updated_at` TIMESTAMP NULL DEFAULT NULL,
 `priv` VARCHAR(10) COLLATE utf8mb4_unicode_ci NOT NULL
)          ENGINE=INNODB          DEFAULT          CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
```

Struktur tabel yang berhasil dibuat dapat dilihat pada gambar 4.1 struktur tabel *users,* di mana tabel *user* memiliki 12 struktur yaitu id sebagai *primary key*, ni, name, jurusan, email, email_verified_at, password, alamat, remember_token, created_at, updated_at dan priv.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(10)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	ni	int(11)			No	None			Change Drop More
3	name	varchar(191)	utf8mb4_unicode_ci		No	None			Change Drop More
4	jurusan	varchar(20)	utf8mb4_unicode_ci		No	None			Change Drop More
5	email	varchar(191)	utf8mb4_unicode_ci		No	None			Change Drop More
6	email_verified_at	timestamp			Yes	NULL			Change Drop More
7	password	varchar(191)	utf8mb4_unicode_ci		No	None			Change Drop More
8	alamat	text	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
9	remember_token	varchar(100)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
10	created_at	timestamp			Yes	NULL			Change Drop More
11	updated_at	timestamp			Yes	NULL			Change Drop More
12	priv	varchar(10)	utf8mb4_unicode_ci		No	None			Change Drop More

Gambar 4.1 Struktur tabel users

b. Tabel Nilai

Tabel nilai difungsikan sebagai tempat untuk menyimpan data nilai. Pembuatan tabel dapat dilakukan dengan contoh *script* membuat tabel nilai di bawah ini.

```
CREATE TABLE `nilais` (
  `id` INT(10) UNSIGNED NOT NULL,
  `nim` INT(10) UNSIGNED NOT NULL,
  `id_matkul` INT(10) UNSIGNED NOT NULL,
  `nilai` INT(11) NOT NULL,
  `semester` INT(1) DEFAULT NULL,
  `created_at` TIMESTAMP NULL DEFAULT NULL,
  `updated_at` TIMESTAMP NULL DEFAULT NULL
) ENGINE=INNODB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
```

Struktur tabel yang berhasil dibuat dapat dilihat pada gambar 4.2 struktur tabel nilai, di mana tabel nilai memiliki 7 struktur yaitu id sebagai *primary key*, nim, id_matkul, nilai, semester, created_at, dan updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(10)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	nim	int(10)		UNSIGNED	No	None			Change Drop More
3	id_matkul	int(10)		UNSIGNED	No	None			Change Drop More
4	nilai	int(11)			No	None			Change Drop More
5	semester	int(1)			Yes	NULL			Change Drop More
6	created_at	timestamp			Yes	NULL			Change Drop More
7	updated_at	timestamp			Yes	NULL			Change Drop More

Gambar 4.2 Struktur tabel nilai

c. Tabel Matkul

Tabel matkul difungsikan sebagai tempat untuk menyimpan data matkul. Pembuatan tabel dapat dilakukan dengan contoh *script* membuat tabel berita di bawah ini.

```
CREATE TABLE `matkuls` (
`id` INT(10) UNSIGNED NOT NULL,
`matkul` VARCHAR(191) COLLATE utf8mb4_unicode_ci NOT NULL,
`sks` INT(10) NOT NULL,
`created_at` TIMESTAMP NULL DEFAULT NULL,
`updated_at` TIMESTAMP NULL DEFAULT NULL
) ENGINE=INNODB DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_unicode_ci;
```

Struktur tabel yang berhasil dibuat dapat dilihat pada gambar 4.3 struktur tabel berita, di mana tabel berita memiliki 5 struktur yaitu id sebagai *primary key*, matkul. Sks, created_at, dan updated_at.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(10)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
2	matkul	varchar(191)	utf8mb4_unicode_ci		No	*None*			Change Drop More
3	sks	int(10)			No	*None*			Change Drop More
4	created_at	timestamp			Yes	*NULL*			Change Drop More
5	updated_at	timestamp			Yes	*NULL*			Change Drop More

Gambar 4.3 Struktur tabel berita

4.1.2. Aplikasi Web

a. Halaman Pembukaan

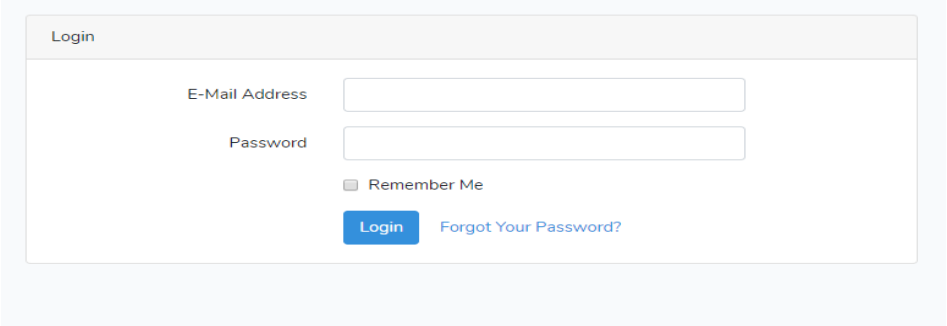
Halaman pembukaan ini merupakan halaman awal dari web bagi admin, dosen dan mahasiswa serta digunakan untuk masuk ke dalam halaman utama web.

Gambar 4.4 Halaman Pembukaan

Pada gambar 4.4 halaman ini merupakan tampilan pertama pada web yang berfungsi untuk masuk kedalam kehalaman utama.

b. Halaman *Login User*

Halaman *login user* merupakan halaman awal ketika masuk pada sistem untuk menuju ke halaman selanjutnya.



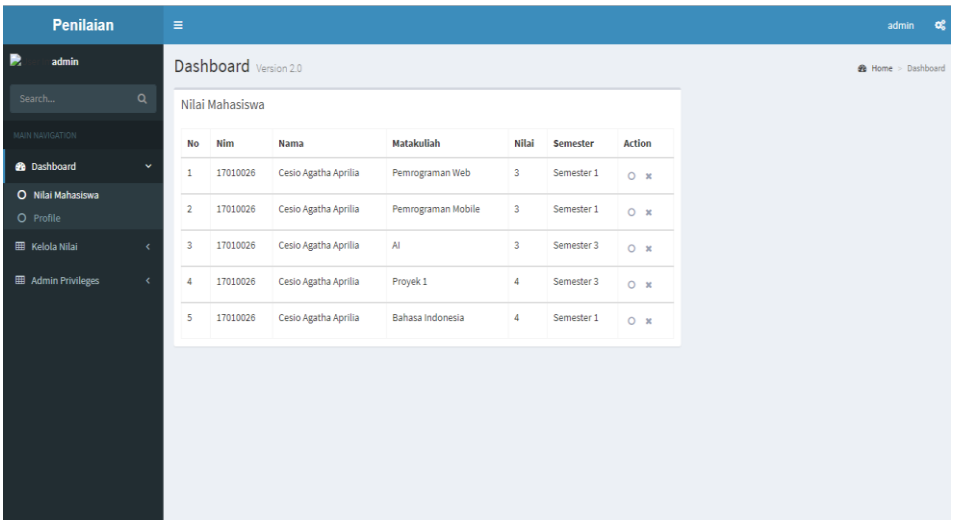
Gambar 4.5 Halaman *Login User*

Pada gamabr 4.5 merupakan halaman *login user* yang digunakan untuk masuk ke halaman selanjutnya dengan mengisi *email* dan *password*. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman *login user*.

```
<form method="POST" ACTION="{{ route('login') }}">
            @csrf

                <input id="email" TYPE="email" class="form-control{{
$errors->has('email') ? ' is-invalid' : '' }}" NAME="email" VALUE="{{
old('email') }}" required autofocus>
                <input id="password" TYPE="password" class="form-
control{{ $errors->has('password') ? ' is-invalid' : '' }}"
NAME="password" required>
                  <input class="form-check-input" TYPE="checkbox"
NAME="remember" id="remember" {{ OLD('remember') ? 'checked' : ''
}}>
                <button TYPE="submit" class="btn btn-primary">
                  {{ __('Login') }}
                </button>
          </form>
```

c. Halaman Nilai Mahasiswa pada akun Admin

Halaman nilai mahasiswa pada akun admin merupakan halaman awal ketika masuk pada sistem dan berfungsi menampilkan nilai dan menu-menu yang digunakan untuk masuk ke halaman lainnya.



Gambar 4.6 Halaman Nilai Mahasiswa pada akun Admin

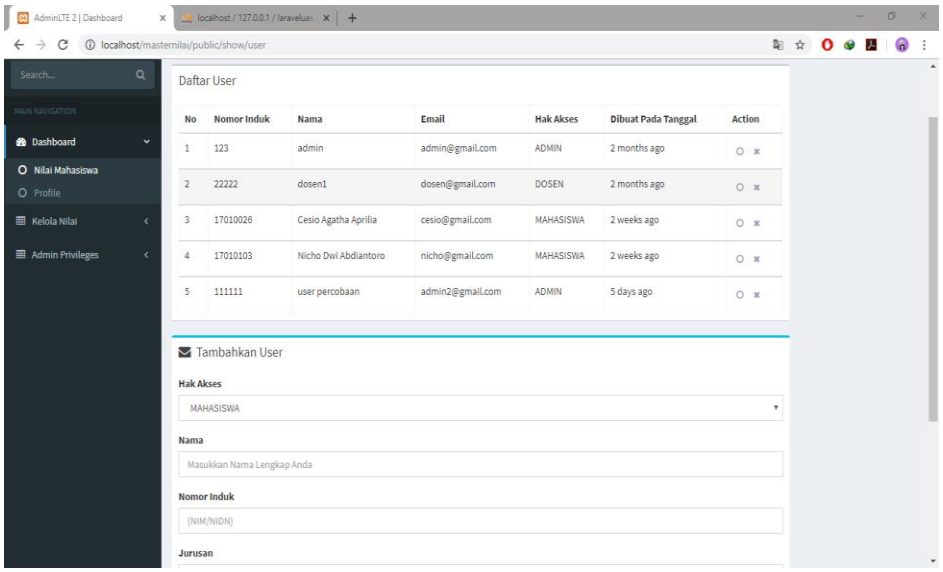
Pada gambar 4.6 Halaman nilai mahasiswa pada akun admin yang digunakan untuk melihat, mengedit dan menghapus nilai mahasiswa. Dan disediakan menu-menu yang berfungsi untuk membuka halaman lainnya. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman nilai mahasiswa pada akun admin.

```
  <?php if ((Auth::user()->priv) === 'ADMIN' OR (Auth::user()->priv) ===
'DOSEN'): ?>
     <?php $no = 0; ?>
     <?php foreach ($nilai as $nilai): ?>
     <?php $no++; ?>
      <tbody>
        <td>{{$no}}</td>
        <td>{{$nilai->user['ni']}}</td>
        <td>{{$nilai->user['name']}}</td>
        <td>{{$nilai->matkul['matkul']}}</td>
        <td>{{$nilai->nilai}}</td>
        <td>Semester {{$nilai->semester}}</td>
        <td>
          <form class="form-horizontal" action="{{ route('nilai.delete',$nilai)
}}" method="post">
            <a href="{{route('nilai.edit',$nilai)}}" class="btn btn-box-tool" data-
toggle="tooltip" title="Edit">
              <i class="fa fa-circle-o"></i></a>
```

```
            {{csrf_field()}}
            {{method_field('DELETE')}}
            <button type="submit" class="btn btn-box-tool" data-
toggle="tooltip" title="Remove"><i class="fa fa-times"></i></button>
          </form>
        </td>

      </tbody>
    <?php endforeach; ?>
  <?php endif; ?>
```

d. Halaman *User Manage* pada akun Admin

Halaman *user manage* pada akun admin merupakan halaman untuk menambahkan user yang dapat masuk pada web sesuai dengan hak akses masing masing.



Gambar 4.7 Halaman *User Manage* pada akun Admin

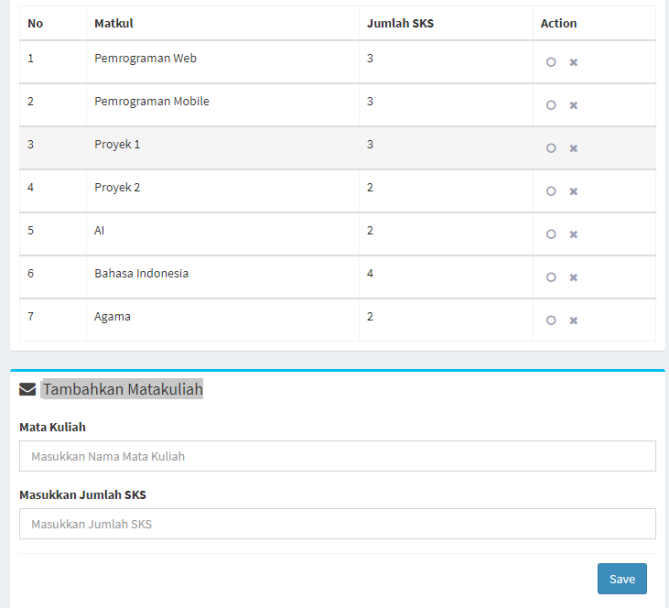
Pada gambar 4.7 merupakan halaman yang digunakan untuk melihat *user* yang sudah terdaftar dan untuk menambahkan *user* baru sesuai hak akses masing masing. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman *User Manage*.

```
<form class="" action="{{ route('user.update',$user) }}" method="post">
          @csrf
          {{ method_field('PATCH')}}
  <input type="text" class="form-control" name="name"
      placeholder="Masukkan Nama Lengkap Anda" value="{{$user-
      >name}}">
    <input type="text" class="form-control" name="ni"
      placeholder="(NIM/NIDN)" value="{{$user->ni}}">
            <select class="form-control" name="priv">
```

```
            <option value="MAHASISWA">MAHASISWA</option>
            <option value="DOSEN">DOSEN</option>
            <option value="ADMIN">ADMIN</option>
          </select>
        <input type="hidden" name="priv" value="{{Auth::user()->priv}}">
            <input type="email" class="form-control" name="email"
placeholder="Email" value="{{$user->email}}" required>
          <input type="alamat" class="form-control" name="alamat"
placeholder="Alamat" value="{{$user->alamat}}" required>
        <input type="submit" name="" value="Save" class="pull-right btn
btn-primary"> </form>
```

e. Halaman Tambah Mata Kuliah pada akun Admin

Halaman tambah mata kuliah pada akun Admin merupakan halaman untuk menambahkan matakuliah yang akan ditampilkan di halaman nilai mahasiswa.



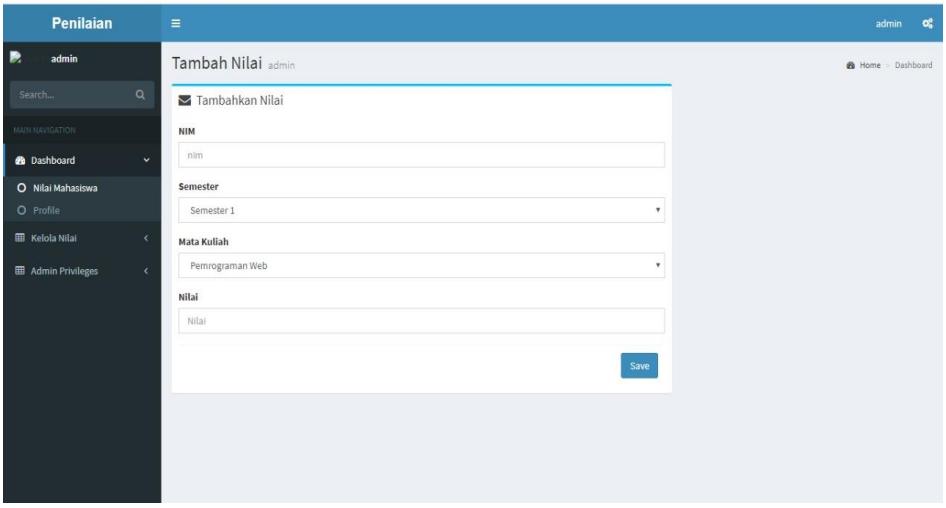
Gambar 4.8 Halaman Tambah mata kuliah pada akun Admin

Pada gambar 4.8 merupakan halaman yang digunakan untuk menambahkan matakuliah, pada halaman tersebut admin dapat menambahkan dan menghapus matakuliah. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman tambah matakuliah pada akun admin.

```
<form class="" action="{{ route('matkul.store') }}" method="post">
        {{ csrf_field() }}
<input type="text" class="form-control" name="matkul"
     placeholder="Masukkan Nama Mata Kuliah" >
        <input type="text" class="form-control" name="sks"
     placeholder="Masukkan Jumlah SKS" >
                <input type="submit" name="" value="Save"
     class="pull-right btn btn-primary">
              </form>
```

f. Halaman Tambah Nilai Mahasiswa pada akun Admin

Halaman tambah nilai mahasiswa pada akun admin merupakan halaman untuk menambahkan data nilai mahasiswa.



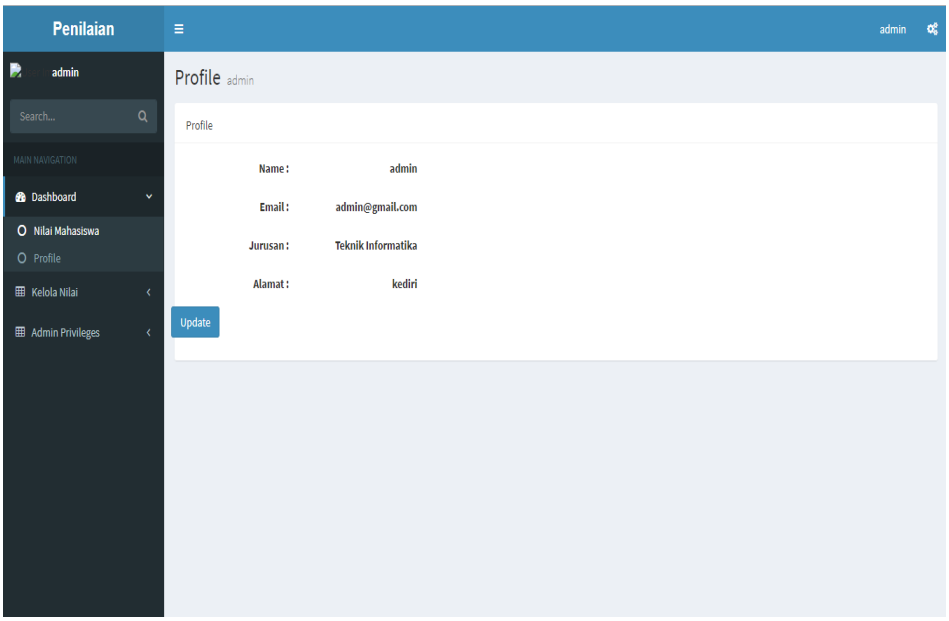
Gambar 4.9 Halaman Tambah Nilai pada akun Admin

Pada gambar 4.9 merupakan halaman yang digunakan untuk menambahkan data nilai mahasiswa, pada halaman tersebut admin dapat menambahkan dan menghapus data nilai mahasiswa. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman tambah nilai mahasiswa pada akun admin.

```
  <?php $no = 0; ?>
  <?php foreach ($nilai as $nilai): ?>
  <?php $no++; ?>
      <form class="form-horizontal" action="{{ route('nilai.delete',$nilai) }}"
  method="post">
 </form>
   </tbody>
  <?php endforeach; ?> <?php endif; ?>
```

g. Halaman Profil pada akun Admin

Halaman profil pada akun admin merupakan halaman untuk menambahkan data profil admin.



Gambar 4.10 Halaman Profil pada akun Admin

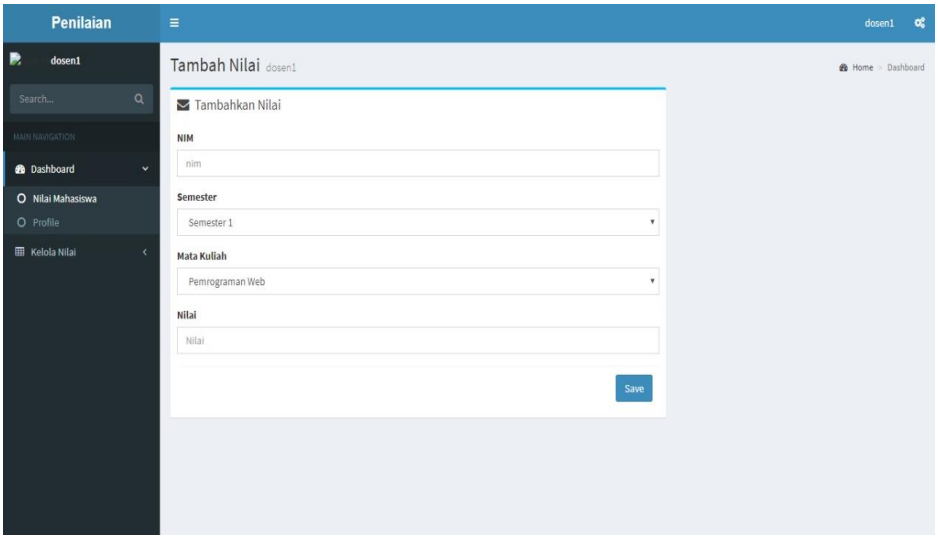
Pada gambar 4.10 merupakan halaman yang digunakan untuk menambahkan data profil, pada halaman tersebut admin dapat menambahkan dan menghapus data profil. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman profil pada akun admin.

```
<?php if ((Auth::user()->priv) === 'ADMIN' OR (Auth::user()->priv)
=== 'DOSEN'): ?>
<div class="active tab-pane" id="profile">
<?php endif; ?>
<label for="inputName" class="col-sm-2 control-label">Name :</label>
<label  class="col-sm-2 control-label">{{Auth::user()->name}}</label>
<label for="inputEmail" class="col-sm-2 control-label">Email :</label>
<label  class="col-sm-2 control-label">{{Auth::user()->email}}</label>
<label for="inputEmail" class="col-sm-2 control-label">Jurusan
:</label>
<label  class="col-sm-2 control-label">{{Auth::user()->jurusan}}</label>
<label for="inputEmail" class="col-sm-2 control-label">Alamat :</label>
<label  class="col-sm-2 control-label">{{Auth::user()->alamat}}</label>
</div>
<div class="form-group">
<a href="{{route('user.edit',Auth::user()->id)}}" class="btn btn-
primary" data-toggle="tooltip" title="Edit">Update</a>
</div>
```

```
</form>
```

h. Halaman Tambah Nilai Mahasiswa pada akun Dosen

Halaman tambah nilai mahasiswa pada akun dosen merupakan halaman awal ketika masuk pada sistem dan berfungsi menampilkan nilai dan menu-menu yang digunakan untuk masuk ke halaman lainnya.



Gambar 4.11 Halaman Tambah Nilai Mahasiswa pada akun dosen

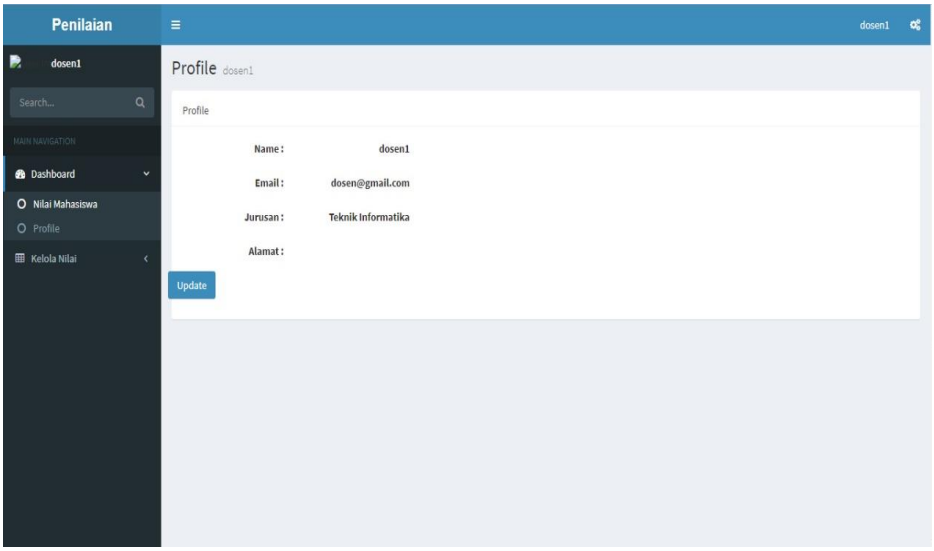
Pada gambar 4.11 merupakan halaman tambah nilai mahasiswa pada akun dosen yang digunakan untuk melihat, mengedit dan menghapus nilai mahasiswa. Dan disediakan menu-menu yang berfungsi untuk membuka halaman lainnya. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman nilai mahasiswa pada akun dosen.

```
<?php if ((Auth::user()->priv) === 'ADMIN' OR (Auth::user()->priv)
=== 'DOSEN'): ?>
  <tr>

    <th>No</th>
    <th>Nim</th>
    <th>Nama</th>
    <th>Matakuliah</th>
    <th>Nilai</th>
    <th>Semester</th>
    <th>Action</th>
  </tr>
  </thead>
  <?php $no = 0; ?>
  <?php foreach ($nilai as $nilai): ?>
```

```
  <?php $no++; ?>
   <tbody>
     <td>{{$no}}</td>
     <td>{{$nilai->user['ni']}}</td>
     <td>{{$nilai->user['name']}}</td>
     <td>{{$nilai->matkul['matkul']}}</td>
     <td>{{$nilai->nilai}}</td>
     <td>Semester {{$nilai->semester}}</td>
     <td>
       <form class="form-horizontal" action="{{ route('nilai.delete',$nilai) }}"
method="post">
         <a href="{{route('nilai.edit',$nilai)}}" class="btn btn-box-tool" data-
toggle="tooltip" title="Edit">
           <i class="fa fa-circle-o"></i></a>
           {{csrf_field()}}
           {{method_field('DELETE')}}
           <button type="submit" class="btn btn-box-tool" data-
toggle="tooltip" title="Remove"><i class="fa fa-times"></i></button>
       </form>
     </td>

   </tbody>
  <?php endforeach; ?>
     <?php endif; ?>
```

i. Halaman Profil pada akun dosen

Halaman profil pada akun dosen merupakan halaman untuk menambahkan data profil dosen.



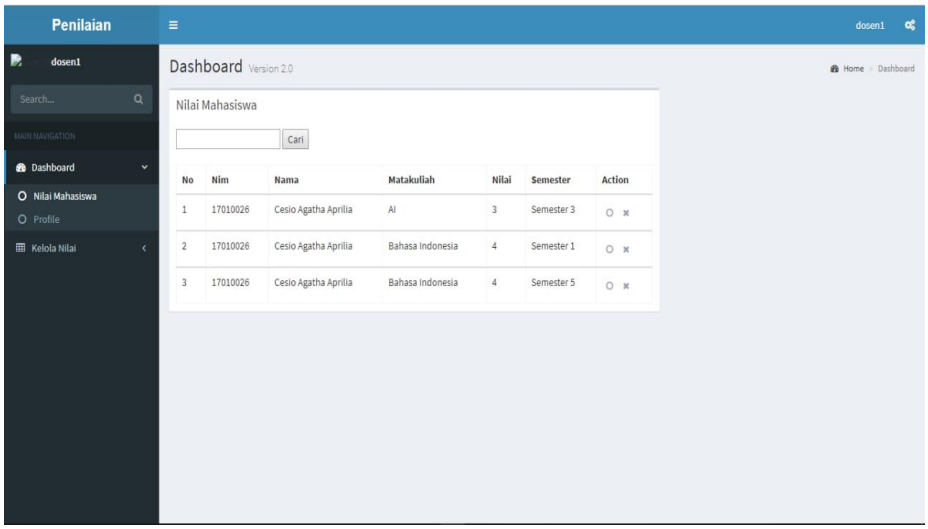
Gambar 4.12 Halaman Profil pada akun Dosen

Pada gambar 4.12 merupakan halaman yang digunakan untuk menambahkan data profil, pada halaman tersebut dosen dapat menambahkan dan menghapus data profil. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman profil pada akun dosen.

```
            <?php if ((Auth::user()->priv) === 'ADMIN' OR (Auth::user()->priv)
=== 'DOSEN'): ?>
         <div class="active tab-pane" id="profile">
        <?php endif; ?>
class="form-horizontal">
          <div class="form-group">
            <label for="inputName" class="col-sm-2 control-label">Name
:</label>
            <label  class="col-sm-2 control-label">{{Auth::user()-
>name}}</label>
          </div>
          <div class="form-group">
            <label for="inputEmail" class="col-sm-2 control-label">Email
:</label>
            <label  class="col-sm-2 control-label">{{Auth::user()-
>email}}</label>
          </div>
          <div class="form-group">
            <label for="inputEmail" class="col-sm-2 control-label">Jurusan
:</label>
            <label  class="col-sm-2 control-label">{{Auth::user()-
>jurusan}}</label>
          </div>
          <div class="form-group">
            <label for="inputEmail" class="col-sm-2 control-label">Alamat
:</label>
            <label  class="col-sm-2 control-label">{{Auth::user()-
>alamat}}</label>
          </div>
          <div class="form-group">
              <a href="{{route('user.edit',Auth::user()->id)}}" class="btn btn-
primary" data-toggle="tooltip" title="Edit">Update</a>
          </div>
         </form>
```

j. Halaman Nilai Mahasiswa pada akunDosen

Halaman Nilai mahasiswa padaakun dosen di gunakan untuk melihat nilai mahasiswa.



Gambar 4.13 Halaman Nilai Mahasiswa pada akun Dosen

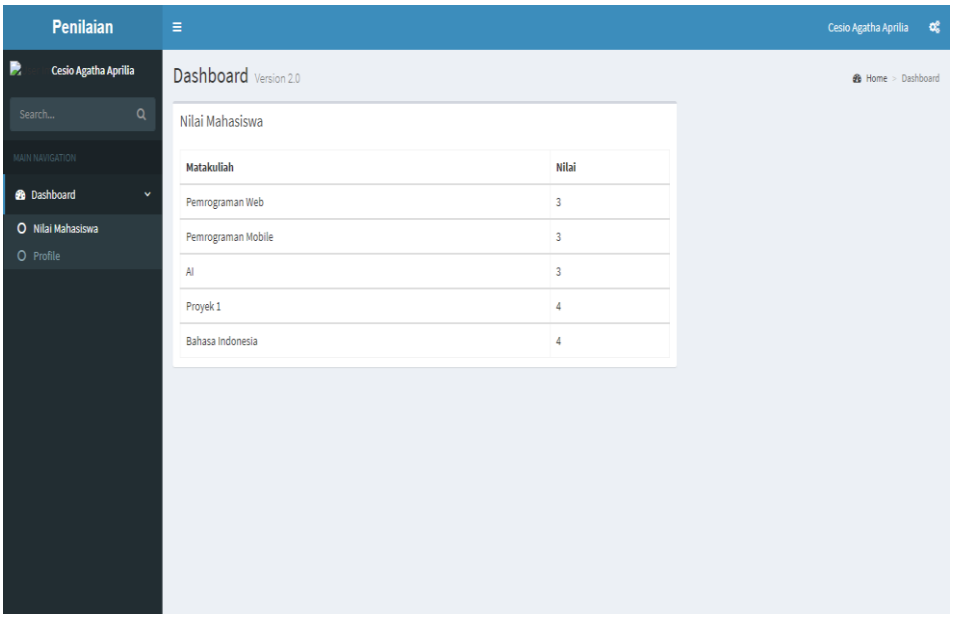
Pada gambar 4.13 merupakan halaman nilai mahasiswa pada akun dosen yang dapat digunakan untuk melihat nilai mahasiswa dan dosen dapat mencari data nilai mahasiswa berdasarkan NIM (Nomor Induk Mahasiswa). Berikut adalah potongan kode yang digunakan untuk menampilkan halaman nilai mahasiswa pada akun dosen.

```
<form class="form group" action="{{route('cari.nilai')}}" method="post">
        {{csrf_field()}}
        <input type="text" name="nim" value="">
        <input type="submit" name="" value="Cari">

    </form>
     <?php $no = 0; ?>
  <?php foreach ($nilai as $nilai): ?>
  <?php $no++; ?>
   <tbody>
    <td>{{$no}}</td>
    <td>{{$nilai->user['ni']}}</td>
    <td>{{$nilai->user['name']}}</td>
    <td>{{$nilai->matkul['matkul']}}</td>
    <td>{{$nilai->nilai}}</td>
    <td>Semester {{$nilai->semester}}</td>
    <td>
     <form class="form-horizontal" action="{{ route('nilai.delete',$nilai) }}" method="post">
       <a href="{{route('nilai.edit',$nilai)}}" class="btn btn-box-tool" data-toggle="tooltip" title="Edit">
        <i class="fa fa-circle-o"></i></a>
```

```
                {{csrf_field()}}
                {{method_field('DELETE')}}
                <button type="submit" class="btn btn-box-tool" data-
        toggle="tooltip"        title="Remove"><i        class="fa        fa-
        times"></i></button>
            </form>
           </td>

          </tbody>
  <?php endforeach; ?>
```

k. Halaman Nilai Mahasiswa pada akun Mahasiswa

Halaman nilai mahasiswa pada akun mahasiswa di gunakan untuk melihat nilai mahasiswa.



Gambar 4.14 Halaman Nilai Mahasiswa pada akun Mahasiswa

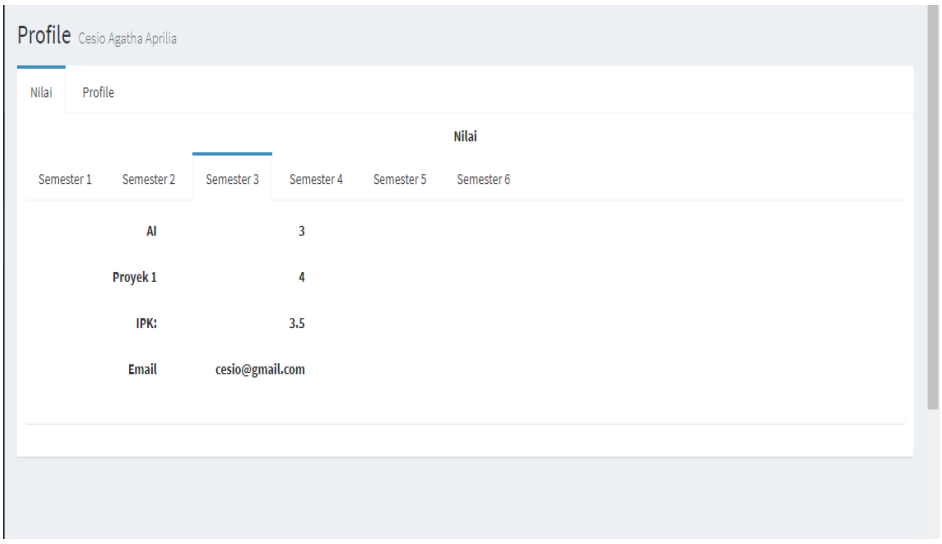
Pada gamabr 4.14 merupakan halaman nilai mahasiswa pada mahasiswa yang dapat digunakan untuk melihat nilai mahasiswa. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman nilai mahasiswa pada akun mahasiswa.

```
<?php if ((Auth::user()->priv) === 'MAHASISWA'): ?>
  <tr>
   <th>Matakuliah</th>
   <th>Nilai</th>
  </tr>
```

```
  </thead>
  <?php foreach ($nilai as $nilai): ?>
   <?php if ((Auth::user()->ni)===($nilai->nim)): ?>

   <tbody>
      <td>{{$nilai->matkul['matkul']}}</td>
      <td>{{$nilai->nilai}}</td>
   </tbody>
   <?php endif; ?>
  <?php endforeach; ?>

<?php endif; ?>
```

l. Halaman Profil Nilai pada akun Mahasiswa

Halaman profil nilai pada akun mahasiswa digunakan untuk melihat Index Prestasi Kumulatif (IPK) pada setiap semester.



Gambar 4.15 Halaman Profil Nilai pada akun Mahasiswa

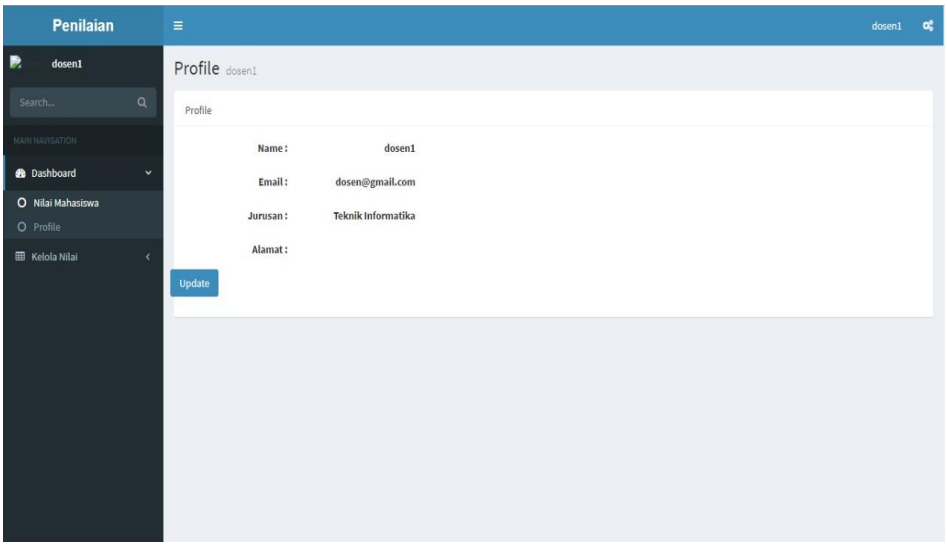
Pada gambar 4.15 merupakan halaman yang digunakan untuk melihat Index Prestasi Kumulatif (IPK) pada setiap semester.Berikut adalah potongan kode yang digunakan untuk menampilkan halaman Profil Nilai pada akun Mahasiswa:

```
<?php
   $nilaiakhir = 0;
   $jmldata = 0;
   $sks = 0
?>
```

```
<?php foreach ($nilai1 as $nilai): ?>
<?php if ($nilai->semester === 1 AND $nilai->nim === (Auth::user()->ni)): ?>
  <?php
      $jmldata++;
      $nilaia = $nilai->nilai*$nilai->matkul['sks'];
      $sks = $nilai->matkul['sks'] + $sks;
      $nilaiakhir = $nilaia + $nilaiakhir;
  ?>
   <label for="" class="col-sm-2 control-label">{{$nilai-
  >matkul['matkul']}}:</label>
   <label  class="col-sm-2 control-label">{{$nilai->nilai}}</label>
<?php endif; ?>
<?php endforeach; ?>
  <?php
  if ($nilaiakhir>0) {
    $nilaiakhir =  $nilaiakhir/$jmldata;
  }
   ?>
  {{$nilaiakhir}}
<label for="inputEmail" class="col-sm-2 control-label">Email</label>
<label  class="col-sm-2 control-label">{{Auth::user()->email}}</label>
```

m. Halaman Profil pada akun Mahasiswa

Halaman profil pada akun mahasiswa merupakan halaman untuk menambahkan data profil mahasiswa.



Gambar 4.16 Halaman Profil pada akun Mahasiswa

Pada gambar 4.16 merupakan halaman yang digunakan untuk menambahkan data profil, pada halaman tersebut mahasiswa dapat

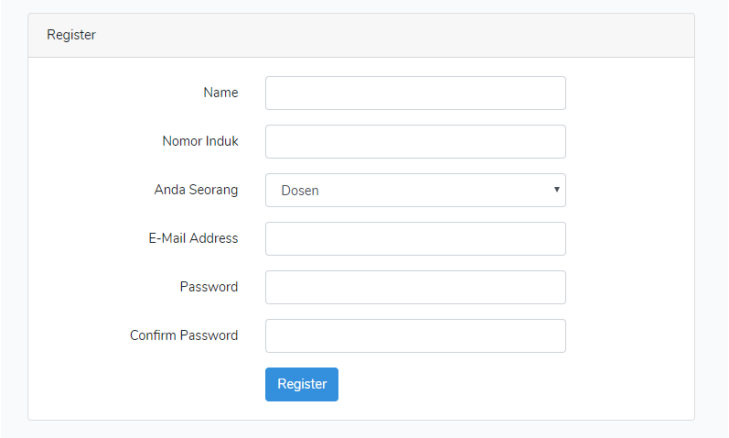
menambahkan dan menghapus data profil. Berikut adalah potongan kode yang digunakan untuk menampilkan halaman profil pada akun mahasiswa.

```
<?php
    $nilaiakhir = 0;
    $jmldata = 0;
    $sks = 0
?>
<?php foreach ($nilai1 as $nilai): ?>
 <?php if ($nilai->semester === 1 AND $nilai->nim === (Auth::user()->ni)): ?>
  <?php
      $jmldata++;
      $nilaia = $nilai->nilai*$nilai->matkul['sks'];
      $sks = $nilai->matkul['sks'] + $sks;
      $nilaiakhir = $nilaia + $nilaiakhir;
  ?>
   <label for="" class="col-sm-2 control-label">{{$nilai-
         >matkul['matkul']}}:</label>
   <label  class="col-sm-2 control-label">{{$nilai->nilai}}</label>
 <?php endif; ?>
<?php endforeach; ?>
 <label for="" class="col-sm-2 control-label">IPK:</label>
 <label  class="col-sm-2 control-label">
  <?php
  if ($nilaiakhir>0) {
   $nilaiakhir =  $nilaiakhir/$jmldata;
  }
   ?>
  {{$nilaiakhir}}
 </label>
 <label for="inputEmail" class="col-sm-2 control-label">Email</label>
 <label  class="col-sm-2 control-label">{{Auth::user()->email}}</label>
```

```
</form>
```

n. Halaman Registrasi

Halaman Registrasi digunakan untuk mendaftar sebagai *user* (Dosen atau mahasiswa).



Gambar 4.17 Halaman Registrasi

Pada gambar 4.17 merupakan Halaman Registrasi digunakan untuk mendaftar sebagai *user* (Dosen atau mahasiswa). Berikut adalah potongan kode yang digunakan untuk menampilkan halaman Registrasi.

```
<form method="POST" ACTION="{{ route('register') }}">
                @csrf
      <input id="name" TYPE="text" class="form-control{{ $errors-
>has('name') ? ' is-invalid' : '' }}" NAME="name" VALUE="{{ old('name') }}"
required autofocus>
      <input id="ni" TYPE="text" class="form-control{{ $errors->has('ni') ? '
is-invalid' : '' }}" NAME="ni" VALUE="{{ old('ni') }}" required autofocus>
                  <SELECT class="form-control" NAME="priv">
                    <OPTION VALUE="DOSEN">Dosen</OPTION>
                    <OPTION VALUE="MAHASISWA">Mahasiswa</OPTION>
                  </SELECT>
                  <label FOR="email" class="col-md-4 col-form-label text-
md-right">{{ __('E-Mail Address') }}</label>
                    <input id="email" TYPE="email" class="form-control{{
$errors->has('email') ? ' is-invalid' : '' }}" NAME="email" VALUE="{{
old('email') }}" required>
                <label FOR="password" class="col-md-4 col-form-label text-
md-right">{{ __('Password') }}</label>
```

```
                        <input id="password" TYPE="password" class="form-
control{{ $errors->has('password') ? ' is-invalid' : '' }}" NAME="password"
required>
                    <label FOR="password-confirm" class="col-md-4 col-form-
label text-md-right">{{ __('Confirm Password') }}</label>
                        <input id="password-confirm" TYPE="password"
class="form-control" NAME="password_confirmation" required>
                        <button TYPE="submit" class="btn btn-primary">
                           {{ __('Register') }}
                        </button>
                </form>
```

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari pembahasan pada bab sebelumnya maka dapat diambil kesimpulan sebagai berikut:

1. Perancangan Sistem Informasi Rekapitulasi Nilai Mahasiswa telah berhasil diimplementasikan dengan menggunakan bahasa pemrograman PHP dan *Framework* Laravel.
2. Pengguna dari sistem informasi ini terdiri dari tiga pengguna yaitu, admin sebagai pengelola sistem informasi secara keseluruhan, dosen sebagai pengguna dan mengelola nilai mahasiswa dan mahasiswa dapat melihat hasil rekapitulasi nilai dalam sistem

5.2 Saran

Saran yang dapat diberikan untuk pengembangan lebih lanjut dari Sistem Informasi Rekapitulasi Nilai Mahasiswa adalah sebagai berikut :

1. Pengembangan sistem yang sesuai dengan perkembangan teknologi informasi, seperti penambahan fitur – fitur yang mampu mengakomodasi seluruh kegiatan Rekapitulasi Nilai Mahasiswa.
2. Aplikasi dapat dikembangkan pada pemrograman *mobile*.

DAFTAR PUSTAKA

Ellsworth. (1997). *Definisi Internet.*

Esti Merindasari1), T. W. (2015). SISTEM INFORMASI PENILAIAN AKADEMIK SISWA KURIKULUM 2013 BERBASIS WEB DI SMAN 1 TRENGGALEK. *Seminar Nasional Sistem Informasi Indonesia, 2-3 November 2015*, 484-488.

Hardi, R., & Hardianto. (2015). Pengembangan Sistem Informasi Perpustakaan. *Jurnal Ilmiah Teknologi Informasi Terapan*, 15-21.

Miarso. (2007). Teknologi.

Naista. (2017). Framework Laravel.

Nash, J. F. (1995). Pengertian Sistem Informasi : Informatika. 35-40.

Nursahid, B. K. (2015). Pembangunan Sistem Informasi Penilaian Hasil Belajar SiswaSekolah Menengah Atas (SMA) Negeri 2 Rembang Berbasis Web. *Indonesian Journal on Networking and Security - Volume 4 No 2 – 2015 –*, 54-63.

Wardani, S. K. (2013). SISTEM INFORMASI PENGOLAHAN DATA NILAI SISWA BERBASIS WEB PADA SEKOLAH MENENGAH ATAS (SMA) MUHAMMADIYAH PACITAN. *Indonesian Jurnal on Networking and Security (IJNS)*, 30-37.