

# **Implementasi Sistem Pelayanan Akademik Mahasiswa Berbasis Web Dengan Memanfaatkan Framework Laravel Pada Jenjang Pascasarjana Universitas Negeri Jakarta Menggunakan Metode Waterfall**

**Muhammad Rafli Akbar<sup>1</sup>, Anna Maulidita Widy Anggraena<sup>2</sup>, Adip Idi Surya<sup>3</sup>, Sulthon  
Abdillah<sup>4</sup>**

Pendidikan Teknik Informatika dan Komputer, Fakultas Teknik, Universitas Negeri Jakarta / Jl. Rawamangun  
Muka Raya No.11, Rawamangun, DKI Jakarta 13220,  
(021) 4898486

[MuhammadRafliAkbar\\_1512620022@mhs.unj.ac.id<sup>1</sup>](mailto:MuhammadRafliAkbar_1512620022@mhs.unj.ac.id),  
[AnnaMauliditaWidyAnggraena\\_1512620074@mhs.unj.ac.id<sup>2</sup>](mailto:AnnaMauliditaWidyAnggraena_1512620074@mhs.unj.ac.id),  
[AdipIdiSurya\\_1512620030@mhs.unj.ac.id<sup>3</sup>](mailto:AdipIdiSurya_1512620030@mhs.unj.ac.id), [SulthonAbdillah\\_1512620096@mhs.unj.ac.id<sup>4</sup>](mailto:SulthonAbdillah_1512620096@mhs.unj.ac.id)

## **ABSTRACT**

*Computer technology began to develop rapidly and affect human life in everyday life. One of the current implementations of computer technology is the Web-Based Student Academic Service System by Utilizing the Laravel Framework at the Postgraduate Level, Jakarta State University. The purpose of this implementation is to make it easier for postgraduate students who need an academic service system through an application. Several methods were carried out in this implementation including data collection, data types, and system development methods. The results of planning and analysis in system design show that this system was developed with the PHP programming language using the Laravel framework which has features that can make it easier for users and developers who will continue developing the system with new features.*

*Keywords:*

## **ABSTRAK**

Teknologi komputer mulai berkembang pesat dan berpengaruh pada kehidupan manusia dalam sehari-hari. Salah satu implementasi teknologi komputer saat ini adalah Sistem Pelayanan Akademik Mahasiswa Berbasis Web dengan Memanfaatkan Framework Laravel pada Jenjang Pascasarjana Universitas Negeri Jakarta. Tujuan implementasi ini adalah mempermudah mahasiswa pascasarjana yang membutuhkan sistem pelayanan akademik melalui sebuah aplikasi. Beberapa metode dilakukan dalam pengimplementasian ini antara lain pengumpulan data, jenis data, dan metode pengembangan sistem. Hasil perencanaan dan analisa dalam perancangan sistem menunjukkan bahwa sistem ini dikembangkan dengan bahasa pemrograman PHP dengan menggunakan framework laravel yang memiliki fitur-fitur yang dapat memudahkan pengguna serta pengembang yang akan melanjutkan pengembang sistem dengan fitur-fitur baru.

Kata Kunci: Laravel, Waterfall, Universitas Negeri Jakarta

# I. PENDAHULUAN

## A. Latar Belakang

Pada era digital seperti pada saat ini, dimana semua kegiatan manusia dibantu dengan teknologi komputer seperti sistem kasir, *learning management system*, belanja online, pelayanan masyarakat, dan lain-lain. Pada Pascasarjana Universitas Negeri Jakarta, proses pelayanan akademik mahasiswa masih dilakukan secara manual yaitu mahasiswa harus mengajukan semua hal secara langsung secara *offline*, misalnya mengajukan topik penelitian, ujian komprehensif, seminar proposal tesis, bimbingan tesis, ujian tesis dan juga yudisium. Proses pencatatan *progress* saat mahasiswa melaksanakan penyelesaian studi sangat berpengaruh pada data. Karena jika data yang dicatat salah, maka mahasiswa akan kesulitan untuk melakukan tahapan selanjutnya dalam menyelesaikan studi.

Program Pascasarjana Universitas Negeri Jakarta berdiri sejak tahun 1978, Program Pascasarjana merupakan program terstruktur pertama di lingkungan Lembaga Pendidikan Tenaga Kependidikan ( LPTK ) di Indonesia. Program Magister yang dibuka adalah Pendidikan Teknologi, Pendidikan Bahasa, Pendidikan Olahraga, Pendidikan Lingkungan dan Pendidikan Kimia. Program Pascasarjana Universitas Negeri Jakarta harus memberikan pelayanan yang optimal kepada mahasiswanya, terutama pelayanan untuk melaksanakan bimbingan dan pengajuan berbagai ujian yang ada pada kurikulum pascasarjana. Proses pelayanan mahasiswa dimulai dari mengajukan topik penelitian, melaksanakan ujian komprehensif, melaksanakan seminar proposal, melakukan bimbingan, melaksanakan ujian dan yudisium.

Dengan dokumentasi pelayanan mahasiswa yang hanya berupa tulisan di buku laporan, sehingga segala sesuatu yang bersangkutan dengan data seperti pencarian riwayat topik penelitian mahasiswa, pencarian riwayat bimbingan mahasiswa, pencarian riwayat ujian mahasiswa dan pengelolaan kelulusan mahasiswa akan mengalami kesulitan. Sistem Pelayanan yang masih manual banyak sekali terdapat kekurangan, seperti memerlukan waktu yang cukup lama dalam mengelola data, redudansi data, ketidakakuratan dalam proses penilaian mahasiswa, serta keterlambatan dalam memberikan informasi maupun laporan.

Program Pascasarjana Universitas Negeri Jakarta dirasa perlu untuk merubah metode administrasi pelayanan akademik mahasiswa yang saat ini sedang berjalan, yaitu metode manual menjadi metode pelayanan akademik yang terkomputerisasi. Dengan kemajuan teknologi saat ini sangat memungkinkan untuk membuat suatu sistem administrasi pelayanan akademik mahasiswa. Hal ini sangat mutlak diperlukan agar perkembangan dan

keberlangsungan mahasiswa Pascasarjana Universitas Negeri Jakarta dapat terpantau dan terdata secara terperinci. Sistem administrasi pelayanan mahasiswa ini akan dibangun dengan menggunakan *Framework* Laravel sebagai pembuatan sistemnya dan MySQL sebagai pengelolaan basis datanya.

Dengan permasalahan yang didapat dari latar belakang di atas, maka dibutuhkan sebuah sistem yang dapat mengontrol proses administrasi pelayanan akademik mahasiswa, mengelola data kelulusan mahasiswa, dan pengelolaan laporan kelulusan mahasiswa. Berdasarkan uraian tersebut penulis mengambil pembahasan mengenai “Implementasi Sistem Pelayanan Akademik Mahasiswa Berbasis Web dengan Memanfaatkan *Framework* Laravel Pada Jenjang Pascasarjana Universitas Negeri Jakarta” sebagai solusi dari permasalahan yang ada pada Program Pascasarjana Universitas Negeri Jakarta.

## **B. Rumusan Masalah**

Berdasarkan latar belakang masalah di atas maka dapat dirumuskan sebuah permasalahan yang ada, yaitu bagaimana merancang dan membangun sebuah Sistem yang dapat mengontrol proses administrasi pelayanan akademik mahasiswa, mengelola data kelulusan mahasiswa, dan mengelola laporan kelulusan mahasiswa pada Program Pascasarjana Universitas Negeri Jakarta.

## **C. Batasan Masalah**

Adapun batasan masalah dari Implementasi *Framework* Laravel untuk membangun Sistem Pelayanan Akademik pada Program Pascasarjana Universitas Negeri Jakarta meliputi :

1. Sistem ini dibangun dan dirancang meliputi proses input data oleh mahasiswa, program studi, dosen pembimbing, penelaah, penguji dan direktur pascasarjana.
2. Alat bantu yang digunakan dalam pembuatan sistem adalah UML ( *Unified Modeling Language* )
3. Sistem Pelayanan Akademik dibangun berbasis *web apps online* dengan *Framework Laravel*, sehingga memerlukan koneksi internet ketika mengoperasikannya.
4. Sistem ini menghasilkan laporan data yang menampilkan informasi kelulusan mahasiswa, jumlah mahasiswa yang mengajukan topik penelitian, serta laporan kelulusan mahasiswa.
5. Sistem ini memiliki hak akses yang hanya bisa digunakan oleh super admin, admin pascasarjana, fakultas, program studi, dosen penguji, dosen pembimbing, penelaah,

dan mahasiswa.

#### **D. Tujuan Penelitian**

Tujuan diadakannya penelitian ini adalah untuk menghasilkan sebuah Aplikasi Sistem Pelayanan Akademik untuk mempermudah mahasiswa pascasarjana dalam menyelesaikan studinya di Program Pascasarjana Universitas Negeri Jakarta.

#### **E. Manfaat Penelitian**

##### **1. Bagi Penulis**

Sebagai sarana untuk menerapkan ilmu yang didapat selama perkuliahan, menambah wawasan, pengetahuan dan pengalaman untuk mengimplementasikan sistem pelayanan akademik mahasiswa dan juga untuk menyelesaikan tugas mata kuliah.

##### **2. Bagi Pembaca**

Dapat digunakan sebagai referensi dan literatur apabila pembaca hendak melakukan penelitian lebih lanjut tentang sistem pelayanan akademik mahasiswa dan mengembangkannya dengan menambahkan inovasi baru yang belum ada pada sistem ini.

## **II. KAJIAN TEORI**

### **A. Pascasarjana Universitas Negeri Jakarta**

#### **1. Sejarah**

Program Pascasarjana Universitas Negeri Jakarta yang berdiri sejak tahun 1978 merupakan Program Pascasarjana terstruktur pertama di lingkungan Lembaga Pendidikan Tenaga Kependidikan (LPTK) di Lembaga Pendidikan Tenaga Kependidikan (Indonesia). Pada tahun 1980, Sekolah Pascasarjana UNJ dikukuhkan menjadi Fakultas Pascasarjana IKIP Jakarta. Selanjutnya berdasarkan Keputusan Presiden Nomor 45 Tahun 1982 tentang Susunan Organisasi IKIP Jakarta, salah satu fakultas yang berwenang adalah Fakultas Pascasarjana. Program pendidikan meliputi Program Magister dan Program Doktor. Program Magister yang dibuka adalah Pendidikan Teknologi, Pendidikan Bahasa, Pendidikan Olahraga, Pendidikan Lingkungan, dan Pendidikan Kimia.

Kebijakan selanjutnya adalah pada tahun 1999 pemerintah memberikan perluasan mandat kepada IKIP Jakarta dengan mengubah namanya menjadi Universitas Negeri Jakarta (UNJ). Sejak saat itu, Pascasarjana memiliki kewenangan untuk membuka Program Studi non kependidikan. Sesuai dengan kebutuhan masyarakat, pada tahun ajaran 2006-2007 dibuka Program Magister Pendidikan Dasar pada tahun 2006, Linguistik Terapan, dan S2 Pengelolaan Lingkungan pada tahun 2009. Program Magister Bimbingan Konseling 2014.

Selanjutnya Program Studi Doktor Pendidikan Anak Usia Dini mendapatkan izin operasional pada tahun 2019. Pada tahun 2020 terjadi perubahan nama nomenklatur dari program studi magister Pendidikan Kependudukan dan Lingkungan Hidup menjadi Pendidikan Lingkungan, dari Pendidikan Olahraga menjadi Pendidikan Jasmani, Pendidikan Bahasa menjadi Bahasa Indonesia Pendidikan. Selanjutnya Program Studi Doktor Pendidikan Olahraga menjadi Pendidikan Jasmani, dan Pendidikan Bahasa menjadi Linguistik Terapan.

Buku Pencatatan Identitas dan Bimbingan Mahasiswa terdapat pada Gambar 2.1 dan 2.2 berikut



Gambar 2.1 Buku Pencatatan Identitas Mahasiswa Pascasarjana.

## 2. Visi dan Misi

- a. Visi: Menjadi Pusat Pendidikan – berdasarkan Penelitian Pascasarjana yang bereputasi di Asia.
- b. Misi : Berkontribusi bagi kemajuan ilmu pengetahuan, teknologi, dan seni melalui pelaksanaan Tridharma Perguruan Tinggi untuk berkontribusi bagi Indonesia dan masyarakat global.

## 3. Biodata

- a. Alamat : Kampus A Universitas Negeri Jakarta, Gedung Bung Hatta, Jl. Rawamangun Muka Jakarta Timur, 13220
- b. Telp : 021 4721340
- c. Email : tu.pps@unj.ac.id

## 4. Lokasi

Program Pascasarjana Universitas Negeri Jakarta beralamat di Jalan Rawamangun Muka Jakarta Timur. Lokasi pada Google Map dapat dilihat pada Gambar 2.3 berikut:



Gambar 2.3 Lokasi Gedung Pascasarjana Universitas Negeri Jakarta

## B. Pengertian Sistem

Dalam bukunya Jogiyanto (2010) terdapat pengertian sistem yang diungkapkan oleh beberapa pakar manajemen yang dijelaskan dari berbagai sudut pandang yang berbeda, namun mempunyai tujuan yang sama. Definisi sistem berkembang sesuai dengan konteks dimana pengertian sistem itu digunakan. Berikut adalah definisi sistem secara umum:

1. Kumpulan dari bagian-bagian yang bekerja sama untuk mencapai tujuan yang sama.  
Contoh : Sistem Informasi, Sistem Komputer, Sistem Administrasi
2. Sekumpulan objek-objek yang saling berelasi dan berinteraksi serta hubungan antar objek bias dilihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan.

Dengan demikian, secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi saling berinteraksi dan saling bergantung satu sama lain.

### 1. Elemen Sistem

- a) Perangkat Keras (*Hardware*): *Hardware*/Perangkat keras adalah peralatan di sistem komputer yang secara fisik terlihat dan dapat dipegang.
- b) Perangkat Lunak (*Software*): *Software*/Perangkat Lunak adalah program yang berisi perintah perintah untuk melakukan pengolahan data. *Software* tidak terlihat secara fisik.
- c) Teknisi (*Brainware*): Manusia yang terlibat didalam mengoperasikan serta mengatur sistem computer
- d) Basis Data (*DataBase*): Basis Data terdiri dari kata Basis yang diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul dan Data sebagai representasi fakta dunia nyata mewakili suatu objek seperti manusia (pegawai,



siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya.

## 2. Karakteristik Sistem

Untuk memahami atau mengembangkan suatu sistem, maka perlu membedakan unsur-unsur dari sistem yang membentuknya. Berikut adalah karakteristik sistem yang dapat membedakan suatu sistem yang dapat membedakan suatu sistem dengan sistem lainnya:

- a. Batasan (*Boundary*): Penggambaran dari suatu sistem atau unsur mana yang termasuk didalam sistem dan mana yang di luar sistem.
- b. Lingkungan (*Environment*): Segala sesuatu diluar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
- c. Masukkan (*Input*): Sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
- d. Keluaran (*Output*): Sumber daya atau produksi (informasi, laporan, dokumen, tampilan layar komputer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
- e. Komponen (*Component*): Kegiatan-kegiatan atau diproses dalam suatu sistem yang mentransformasikan input menjadi bentuk setengah jadi (output). Komponen ini bisa merupakan subsistem dari sebuah sistem.
- f. Penghubung (*Interface*): Tempat dimana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
- g. Penyimpanan (*Storage*): Area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku, dan sebagainya. Penyimpanan merupakan suatu media penyangga diantara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama.
- h. Sasaran (*Objective*). Suatu sistem pasti mempunyai tujuan (goal) atau sasaran (objective). Jika Suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.



### 3. Klasifikasi Sistem

Menurut Jogyanto (2010), sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya adalah sebagai berikut:

- a. Sistem diklasifikasikan sebagai sistem abstrak dan sistem fisik. Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, Sistem fisik merupakan sistem yang ada secara fisik.
- b. Sistem diklasifikasikan sebagai sistem alamiah(*natural system*) dan sistem buatan manusia(*human made system*).Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia.
- c. Sistem diklasifikasikan sebagai suatu sistem tertentu(*deterministic system*) dan sistem tertentu(*probalistic system*).Sistem tertentu beroperasi dengan tingah laku yang sudah ada diprediksi. Interaksi diantara bagian-bagiannya dapat dideteksi dengan pasti,sehingga keluarannya dari sistem dapat diramalkan.
- d. Sistem diklasifikasikan sebagai sistem tertutup dan sistem terbuka. Sistem tertutup merupakan sistem yang tidak berhubungan dengan lingkungan luar.Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan pihak luar.Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem lainnya.

### C. Administrasi

Istilah Administrasi sering kita dengar terlebih dalam bidang yang berurusan dengan catat-mencatat, pembukuan, surat-menyurat, pembuatan agenda, dan sebagainya. Ilmu mengenai administrasi dalam instansi pemerintahan maupun perusahaan diperlukan untuk pelaksanaan kegiatan pemerintah atau perusahaan. Apabila dalam suatu instansi pengelolaan administrasinya baik maka instansi tersebut juga akan dapat berjalan dengan baik.

Menurut Sondang P. Siagian dalam bukunya Administrasi Pembangunan (2010) “Administrasi adalah keseluruhan proses pelaksanaan dari keputusan keputusan yang telah diambil dan pelaksanaan itu pada umumnya dilakukan oleh dua orang manusia atau lebih untuk mencapai tujuan yang telah ditentukan sebelumnya”.

Sedangkan menurut Irra Chisyanti Dewi (2011) Administrasi dalam arti sempit yaitu: Administrasi berasal dari kata *Administratie* (bahasa Belanda), yang diartikan sebagai

pekerjaan tulis menulis atau ketatausahaan atau kesekretariatan, meliputi kegiatan: menerima, mencatat, menghimpun, mengolah, mengadakan, mengirim, menyimpan. Administrasi dalam arti luas yaitu: Administrasi merupakan proses kerjasama beberapa individu dengan cara yang efisien dalam mencapai tujuan sebelumnya.

Berdasarkan pengertian mengenai administrasi menurut para ahli, administrasi dapat dikatakan sebagai proses kerjasama yang melibatkan sedikitnya dua orang untuk mencapai sasaran atau tujuan yang sudah ditentukan sebelum mereka memulai kegiatan, ataupun persediaan bahan baku yang menunggu penggunaannya dalam suatu proses produksi.


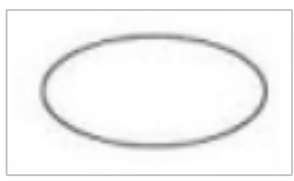
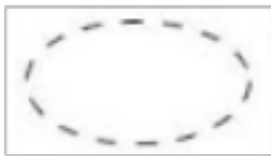



#### **D. *Unified Modeling Language ( UML )***



Menurut Adi Nugroho (2010), UML (*Unified Modeling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi objek. Pemodelan (*modeling*) sesungguhnya digunakan untuk menyederhanakan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipahami dan dipelajari. UML juga menyediakan standar notasi ataupun diagram yang digunakan untuk pemodelan system.

##### **1. *Use Case Diagram***

Menurut Adi Nugroho (2010), *Use Case Diagram* digunakan untuk memodelkan fungsionalitas - fungsionalitas sistem/perangkat lunak dilihat dari pengguna yang ada di luar sistem (yang sering dinamakan sebagai aktor). *Use Case* pada dasarnya merupakan unit fungsionalitas koheren yang diekspresikan sebagai transaksi-transaksi yang terjadi antara aktor dan sistem. Kegunaan dari *use case diagram* adalah untuk mendaftarkan aktor-aktor dan *use case – use case* dan memperlihatkan aktor-aktor mana yang berpartisipasi dalam masing-masing *use case*.

Tabel 3.1 Notasi-Notasi *Use Case Diagram*

NOTASI	KEGUNAAN	SIMBOL
<i>Actor</i>	Menggambarkan semua objek di luar sistem (bukan hanya pengguna sistem / perangkat lunak) yang berinteraksi dengan sistem yang dikembangkan.	
<i>Use Case</i>	Menggambarkan fungsionalitas	
<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerjasama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi)	
<i>Association</i>	Lintasan komunikasi antara actor dengan use case.	
<i>Include</i>	Penambahan perilaku sesuatu use case dasar yang secara eksplisit mendeskripsikan penambahan tersebut.	 <<include>>
<i>Exclude</i>	Penambahan perilaku sesuatu use case dasar.	 <<extend>>

<i>Generalization</i>	Relasi antara pengklasifikasi yang memiliki deskripsi yang bersifat lebih umum dengan berbagai pengklasifikasi yang lebih spesifik, digunakan dalam struktur pewarisan.	
<i>Dependency</i>	Relasi antar dua elemen mode	

Tabel 3.1 menunjukan notasi-notasi *Use Case* yang terdiri dari *Actor*, *usecase*, *collaboration*, *association*, *include*, *extend*, *generalization*, dan *dependency*


## 2. *Class Diagram*



Menurut Adi Nugroho (2010), dalam notasi UML, himpunan kelas kelas beserta hubungan / relasi / asosiasi antar kelas biasanya digambarkan menggunakan sebuah diagram UML yang dinamakan diagram kelas (class diagram). Jika kita perhatikan lebih jauh, sesungguhnya diagram kelas memiliki dua kegunaan / fungsi yang sangat penting, yaitu:

- Mempresentasikan keadaan statis kelas-kelas yang terlibat dalam sistem. Kelas-kelas ini bisa saja merupakan kelas-kelas dalam bahasa pemrograman dan kelas-kelas persisten yang hadir dalam bentuk tabel tabel yang ada di sistem basis data relasional.
- Hubungan antar kelas dalam sistem atau perangkat lunak yang sedang kita kembangkan dapat terlihat dengan mudah.

Notasi-notasi yang digunakan dalam class diagram UML terdapat pada Tabel 3.2 berikut

Tabel 3.2 Notasi-Notasi *Class Diagram*

NOTASI	KETERANGAN	SIMBOL
<i>Class</i>	Class adalah balok-balok pembangun pada pemrograman berorientasi objek. Sebuah class digambarkan sebagai sebuah kotak yang terbagi menjadi 3 bagian. Bagian atas adalah bagian nama dari class. Bagian tengah mendefinisikan atribut class. Bagian bawah mendefinisikan method dari sebuah class.	Class1
<i>Composition</i>	Jika sebuah class tidak bisa berdiri sendiri dan harus menjadi bagian dari class yang lain, maka class tersebut memiliki relasi composition terhadap class tempatnya	
<i>Association</i>	Sebuah asosiasi merupakan sebuah relationship paling umum antara 2 class, dan dilambangkan oleh sebuah garis yang menghubungkan antar 2 class. Garis ini dapat melambangkan tipe – tipe relationship dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah relationship.	<p>_____ 1..n _____</p> <p>owned</p> <p>by1</p>

<i>Dependency</i>	Kadang kala class menggunakan class yang lain. Hal ini disebut dependency. Umumnya dependency digunakan untuk menunjukkan operasi pada suatu class yang menggunakan class yang lain.	
<i>Generalization</i>	Sebuah generalization dilambangkan dengan sebuah panah dengan kepala panah yang tidak solid yang mengarah ke arah "parent"-nya / induknya	

Tabel 3.2 menunjukan notasi-notasi Class Diagram yang terdiri dari *Class composition, association, dependency, aggregation, and generalization*.



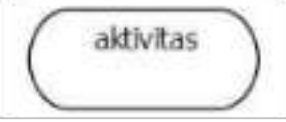
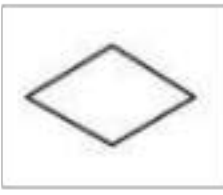
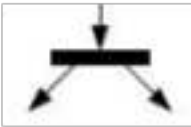

### 3. Activity Diagram

Menurut Adi Nugroho (2010), diagram aktivitas (activity diagram) sesungguhnya merupakan bentuk khusus dari state machine yang bertujuan untuk memodelkan komputasi-komputasi dan aliran - aliran kerja yang terjadi dalam sistem / perangkat lunak yang sedang dikembangkan. State pada diagram aktivitas merepresentasikan state dari komputasi yang dieksekusi, bukan state dari suatu objek biasa.

Suatu diagram aktivitas memuat di dalamnya activity state dimana suatu activity state merepresentasikan eksekusi pernyataan dalam suatu prosedur atau kinerja suatu aktivitas dalam suatu aliran kerja. Alih-alih menunggu selesainya atau event seperti yang terjadi pada state tunggu, activity state menunggu selesainya komputasi. Saat suatu aktivitas selesai maka akan berlanjut ke activity state berikutnya yang terlihat pada diagram aktivitas. Penyelesaian transisi dalam suatu diagram aktivitas biasanya akan terpicu saat aktivitas sebelumnya selesai.

Berikut adalah notasi - notasi / simbol-simbol yang digunakan pada activity diagram yang terdapat pada Tabel 3.3 dibawah ini

Tabel 3.3 Notasi-notasi Activity Diagram

NOTASI	KETERANGAN	SIMBOL
<i>Initial</i>	Titik awal untuk memulai suatu aktivitas.	
<i>Final</i>	Titik akhir untuk mengakhiri aktivitas.	
<i>Activity</i>	Menandakan sebuah aktivitas	
<i>Decision</i>	Pilihan untuk mengambil keputusan	
<i>Fork</i>	Menujukan kegiatan yang dilakukan secara paralel	
<i>Join</i>	Untuk menggabungkan beberapa secara paralel menjadi satu	

Tabel 3.3 menunjukkan notasi-notasi *Activity Diagram* yang terdiri dari *Initial*, *final*, *activity*, *decision*, *fork*, dan *join*



#### **4. Sequence Diagram**

Menurut Adi Nugroho (2010), SequenceDiagram memperlihatkan interaksi sebagai diagram dua matra (dimensi). Matra vertikal adalah sumbu waktu, waktu bertambah dari atas ke bawah. Matra horizontal memperlihatkan peran pengklasifikasian yang merepresentasikan objek - objek mandiri yang terlibat dalam kolaborasi. Masing - masing pengklasifikasian direpresentasikan sebagai kolom-kolom vertikal dalam sequence diagram yang sering disebut sebagai garis waktu (life line). Selama objek ada, peran digambarkan menggunakan garis tegas. Selama aktivitas prosedur pada objek aktif, garis waktu digambarkan sebagai garis ganda. Pesan-pesan digambarkan sebagai suatu tanda panah dari garis waktu suatu objek ke garis waktu objek lainnya.

Panah-panah menggambarkan aliran pesan antar peran pengklasifikasian digambarkan dalam urutan waktu kejadiannya dari atas ke bawah.

#### **E. MySQL**

Menurut (Nugroho, 2013), "MySQL adalah software atau program Database Server". Sedangkan SQL adalah bahasa pemrogramannya, bahasa permintaan (query) dalam database server termasuk dalam MySQL itu sendiri. SQL juga dipakai dalam software database server lain, seperti SQL Server, Oracle, PostgreSQL, dan lainnya.

Saat ini pengembangan MySQL berada dibawah naungan perusahaan MySQLAB. Sebagai software DBMS, MySQL memiliki sejumlah fitur seperti:

1. Multiplatform MySQL tersedia di platform Windows, Linux, Unix, dan lain-lain.
2. Andal, cepat, dan mudah digunakan MySQL tergolong sebagai database server yang andal,dapat menangani database yang besar dengan kecepatan tinggi,mendukung banyak sekali fungsi untuk mengakses database, dan sekaligus mudah untuk digunakan.
3. Jaminan keamanan akses MySQL mendukung keamanan database dengan berbagai kriteria akses.
4. Dukungan SQL, SQL merupakan standar dalam pengaksesan database relasional. Pengetahuan SQL akan memudahkan siapa pun menggunakan MySQL.

## **F. Framework Laravel**

Framework adalah struktur konseptual dasar yang berisi kumpulan fungsi untuk tujuan tertentu yang sudah siap untuk digunakan, sehingga pembuatan aplikasi dapat dilakukan dengan lebih cepat karena kode programnya tidak di buat dari awal.

Laravel merupakan Framework PHP yang menekankan pada kesederhanaan dan fleksibilitas pada desainnya. Laravel dirilis di bawah lisensi MIT dengan sumber kode yang disediakan di Github. Laravel dibangun dengan basis MVC (Model-View-Controller) sama seperti framework PHP lainnya. Laravel dilengkapi command line tool yang bernama "Artisan" yang bisa digunakan untuk packaging bundle dan instalasi bundle. Framework Laravel dibuat oleh Taylor Otwell. Proyek Laravel dimulai pada April 2011, Awal mula proyek ini dibuat karena Otwell sendiri tidak menemukan framework yang uptodate dengan versi PHP. Mengembangkan framework yang sudah ada juga bukan merupakan ide yang bagus karena keterbatasan sumber daya. Maka dari itu, Otwell membuat sendiri framework bernama Laravel. Oleh karena itu Laravel mensyaratkan PHP versi 5.3 keatas. (Rohman, 2014). Framework Laravel juga memiliki beberapa keunggulan sebagai berikut:

- a. Menggunakan Command Line Interface (CLI) Artisan.
- b. Menggunakan package manager PHP Composer.
- c. Penulisan kode program lebih singkat, mudah dimengerti, dan ekspresif.

Fitur framework Laravel yang ditekankan pada penelitian ini adalah Blade, Migration, Eloquent ORM, Resource Controller, dan Middleware. Berikut adalah penjelasan mengenai lima fitur tersebut:

### **1. Blade**

Blade adalah template engine. Pada dasarnya Blade adalah view namun dengan menggunakan Blade akan mempermudah untuk mengatur tampilan website dan menampilkan data. Untuk membuat file view menjadi Blade yaitu dengan menambahkan ekstensi .blade.php pada file view. Dan cara untuk memanggil file Blade sama dengan cara untuk memanggil file view biasa.

dashboard.blade.php
<pre> &lt;body&gt; &lt;div id="app"&gt; &lt;div class="main-wrapper"&gt; @include('partials.navbarDashboard')  &lt;div class="main-sidebar"&gt; @include('partials.sidebarDashboard') &lt;/div&gt;  @yield('container')                                  @include('partials.footerDashboard') &lt;/div&gt;  &lt;/div&gt;  @include('partials.jsDashboard')  @yield('script') &lt;/body&gt; </pre>

## 2. Migration

Migration adalah fitur yang menyediakan cara baru untuk membuat database. Menggunakan migrasi menggantikan pembuatan database menggunakan database Command Line Interface (CLI) atau aplikasi manajemen database menggunakan kelas. Langkah-langkah untuk menggunakan migrasi adalah membuat kelas dan menjalankan perintah migrasi dari Artisan Command Line Interface (CLI).

Keuntungan menggunakan migration adalah class yang dibuat bisa dipakai untuk membuat database pada berbagai macam Relation Database Management System (RDBMS) yang didukung oleh Laravel. Sebagai contoh misalnya aplikasi yang digunakan selama ini menggunakan database MySQL, kemudian karena alasan pengembangan aplikasi maka akan dilakukan penggantian database ke PostgreSQL. Pada proses tersebut langsung melakukan perintah migrate melalui Command Line Interface (CLI) artisan, tidak perlu membuat class lagi.

Keuntungan lain dari menggunakan migration adalah semua perubahan yang

dilakukan pada database akan disimpan pada suatu tabel. Sehingga bisa dilakukan pembatalan (rollback) pada database jika melakukan perubahan yang tidak benar.

### 3. Eloquent ORM

Eloquent ORM adalah implementasi dari ActiveRecord yang digunakan untuk mengatur relasi antar tabel di database. Pada Eloquent ORM tabel direpresentasikan dalam bentuk kelas dan data yang tersimpan didalam tabel direpresentasikan dalam bentuk objek. Relasi yang dapat diatur menggunakan Eloquent ORM adalah sebagai berikut:

- a. One-to-One: relasi satu ke satu, digunakan method `hasOne` dan `belongsTo`.
- b. One-to-Many: relasi satu ke banyak, digunakan method `hasMany` dan `belongsTo`.
- c. Many-to-One: relasi banyak ke satu, digunakan method `belongsTo` dan `hasMany`.
- d. Many-to-Many: relasi banyak ke banyak, digunakan method `belongsToMany`.

### 4. Resource Controller

Resource Controller adalah fitur yang digunakan untuk mempercepat pembuatan controller. Sebagai contoh misalnya ada controller yang menangani semua HTTP request terhadap data dosen, untuk membuat controller tersebut hanya perlu mengetikkan perintah berikut.

```
php artisan make:controller TopikPenelitianController --resource
```

Perintah diatas akan menghasilkan controller `TopikPenelitianController.php` yang disimpan pada folder `app/Http/Controllers`. Tabel 3.5 dibawah ini adalah daftar action yang dapat dilakukan oleh controller `TopikPenelitianController.php`.

Tabel 3.5 Daftar Action TopikPenelitianController.php

No.	Verb	URI Action Route Name
1.	GET	/topikPenelitian index topikPenelitian
2.	POST	/pengajuanTopikPenelitian
3.	POST	/penelitian/{id}/setuju setuju setujuTopikPenelitian
4.	POST	/penelitian/{id}/tolak tolak tolakTopikPenelitian

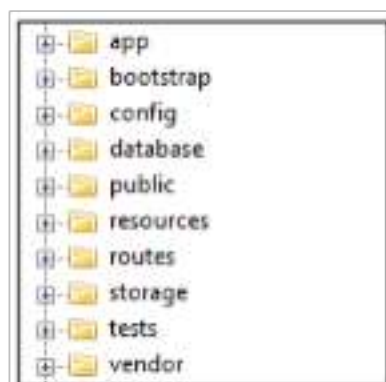
## 5. Middleware

Middleware adalah fitur yang menyediakan mekanisme untuk memfilter HTTP request yang masuk ke aplikasi. Beberapa Middleware dari Laravel yaitu Authenticate, EncryptCookies, RedirectIfAuthenticated, dan VerifyCsrfToken.

Sebagai pembahasan akan dibahas Middleware Authenticate, Middleware tersebut akan memeriksa apakah user sudah login atau belum. Jika user sudah login maka request akan dilanjutkan ke halaman yang dikehendaki oleh user. Tetapi jika user belum login maka Middleware Authenticate akan mengarahkan user ke halaman login.

Jika Middleware yang sudah ada pada Laravel kurang sesuai dengan kebutuhan ataupun tidak sesuai dengan kebutuhan maka dapat dibuat sendiri Middleware yang sesuai dengan kebutuhan.

Struktur folder dari framework Laravel 5.3 yang masih default dapat dilihat pada Gambar 3.1 dibawah ini:



Gambar 3.1 Struktur Folder Laravel

Berikut adalah keterangan pada Gambar 3.1 diatas

1. Folder app adalah folder yang berisi kode program inti dari aplikasi yang akan dibuat. Model dan controller tersimpan pada folder ini.
2. Folder bootstrap adalah folder yang berisi konfigurasi autoloading dan terdapat juga folder cache yang menyimpan file-file yang dihasilkan secara otomatis oleh Laravel untuk mengoptimasi kinerja dari sistem yang dihasilkan. .
3. Folder config adalah folder yang berisi semua file konfigurasi aplikasi.
4. Folder database adalah folder yang berisi file database migration dan seeds.
5. Folder public adalah folder yang berisi file index.php. File tersebut digunakan sebagai entry point untuk menangani semua request yang masuk ke aplikasi. Pada folder ini juga dapat disimpan beberapa aset dari aplikasi seperti gambar, JavaScript, dan CSS.
6. Folder resources adalah folder yang berisi file view dari aplikasi yang dibuat. Selain itu terdapat juga file language yang digunakan aplikasi.
7. Folder routes adalah folder yang berisi file yang digunakan untuk mendefinisikan semua route ke aplikasi. Secara default ada tiga file route yang disediakan Laravel yaitu api.php, console.php, dan web.php.
8. Folder storage adalah folder yang berisi template Blade yang dikompilasi, file session, file cache, dan file lainnya yang dihasilkan secara otomatis oleh Laravel.
9. Folder tests adalah folder yang berisi semua file test yang dibuat untuk aplikasi.
10. Folder vendor adalah folder yang menyimpan semua library yang digunakan.

### **III. METODOLOGI PENELITIAN**

#### **A. Pengumpulan Data**

Data merupakan sumber atau bahan mentah yang sangat berharga bagi suatu proses yang akan menghasilkan informasi, dalam pengumpulan sumber data perlu penanganan secara cermat, sehingga menghasilkan data yang akurat dan berkualitas. Beberapa metode dalam pengumpulan sumber data yang digunakan :

##### **1. Wawancara**

Metode pengumpulan sumber data ini diperoleh secara langsung dengan melakukan wawancara langsung dengan bapak Irfan selaku Kepala Divisi Aplikasi PUSTIKOM Universitas Negeri Jakarta saat ini. Dengan mengajukan pertanyaan mengenai bagaimana proses kelulusan mahasiswa pascasarjana saat ini? sehingga mendapatkan data berupa informasi tentang proses kelulusan mahasiswa saat ini yang dilakukan manual yaitu mahasiswa mengajukan secara langsung ke prodi dan dosen pembimbing ketika mengajukan topik penelitian, lalu kemudian baru disetujui jika mahasiswa sudah memenuhi syarat.

##### **2. Observasi**

Metode pengumpulan sumber data ini diperoleh dengan pengamatan secara langsung tentang proses akademik mahasiswa agar didapat data yang sesuai kebutuhan. Data yang diperoleh berupa informasi tentang penyelesaian akademik mahasiswa yang ditulis kedalam buku belum terkomputerisasi.

##### **3. Studi Pustaka**

Studi Pustaka dilakukan dengan cara mempelajari teori-teori literature dan buku-buku yang berhubungan dengan sistem pelayanan akademik dan framework laravel sebagai dasar dalam pelaksanaan penelitian. Data ini berupa jurnal tentang proses administrasi, modul pelayanan akademik dengan menggunakan sistem yang terkomputerisasi.

#### **B. Jenis Data**

##### **1. Data Primer**

Merupakan data yang didapat secara langsung dari narasumber yang bersangkutan dengan penelitian melalui wawancara dengan Bapak Irfan selaku Kepala Divisi Aplikasi PUSTIKOM Universitas Negeri Jakarta dan pengamatan dari penulis,



seperti profil Program Pascasarjana, data mahasiswa, data dosen, dan laporan program pascasarjana.

## **2. Data Sekunder**

Merupakan data yang diperoleh secara tidak langsung dari berbagai literatur yang berhubungan dengan sistem akademik. Dalam penelitian ini data sekunder berupa jurnal tentang sistem akademik, jurnal tentang pelayanan akademik dan jurnal tentang konsep Laravel untuk sistem pelayanan akademik mahasiswa.

## **C. Metode Pengembangan Sistem**

Metode pengembangan sistem yang digunakan adalah model Waterfall yaitu sebuah model pengembangan sistem dimana antara satu fase ke fase yang lain dilakukan secara berurutan. Biasanya sebuah langkah akan diselesaikan terlebih dahulu sebelum melanjutkan ke fase berikutnya. Keuntungan menggunakan metode ini, proses pengembangan model fase one by one, sehingga meminimalisir kesalahan yang akan terjadi. Fase-fase dari Gambar 1.1 Metode Waterfall di atas sebagai berikut:

### *a. Requirements Definition*

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Pada tahap ini dilakukan pengumpulan hal-hal yang diperlukan untuk membuat sistem pelayanan akademik pascasarjana, seperti pengumpulan data, dan analisa kebutuhan sistem.

### *b. System and Software Design*

Setelah data dikumpulkan maka selanjutnya melakukan desain program yang dibangun dengan alat bantu perancangan UML, menggunakan Use Case Diagram, Class Diagram, Activity Diagram, dan Sequence Diagram, sedangkan untuk antarmuka sistem dengan menggunakan Balsamiq Mockup.

### *c. Implementation and Unit Testing*

Tahap ini sistem pertama kali dikembangkan di program kecil yang disebut Unit, yang akan diintegrasikan pada tahap selanjutnya. Setiap Unit dikembangkan dan diuji fungsionalitasnya, proses ini disebut Unit Testing. Untuk sistem Pelayanan Akademik Pascasarjana ini dibuat menggunakan bahasa pemrograman PHP dengan Framework Laravel.

*d. Integration and System Testing*

Pada tahap ini semua unit yang sudah dibuat pada tahap Implementation and Unit Testing kemudian diintegrasikan kedalam sistem. Setelah semua Unit terintegrasi selanjutnya dilakukan pengujian sistem dengan menggunakan Blackbox dan Whitebox testing untuk mengecek apakah program tersebut sudah sesuai dengan kebutuhan, dan tombol berjalan sesuai dengan fungsinya.

*e. Operation and Maintenance*

Tahap akhir dalam Metode *Waterfall* setelah pengujian selesai maka, sistem ini dioperasikan pada Sistem Pelayanan Akademik Pascasarjana sesuai dengan fungsinya serta dilakukan pemeliharaan termasuk dalam memperbaiki kerusakan yang ditemukan pada tahap sebelumnya.

**D. Pengujian Kotak Hitam ( *Blackbox* )**

Pengujian bertujuan untuk mencari kesalahan. Pengujian yang baik adalah pengujian yang memiliki kemungkinan besar dalam menemukan kesalahan. Karena itu, harus merancang dan mengimplementasikan sistem berbasis komputer atau produk dengan “kemampuan untuk diuji”. Pada saat yang sama, tes itu sendiri harus menunjukkan serangkaian karakteristik yang bertujuan untuk menemukan sebanyak mungkin kesalahan dengan usaha sekecil mungkin (Pressman, 2012).

Pengujian kotak hitam (*blackbox*) disebut juga pengujian perilaku yang berfokus pada persyaratan fungsional perangkat lunak. Artinya, teknik pengujian kotak hitam memungkinkan untuk membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan fungsional untuk program. Pengujian kotak hitam bukan teknik alternatif untuk kotak putih. Sebaliknya, ini merupakan pendekatan pelengkap yang mungkin dilakukan untuk mengungkap kelas kesalahan yang berbeda dari yang diungkap oleh metode kotak putih. Pengujian kotak hitam (*blackbox*) digunakan untuk menemukan kesalahan seperti:

1. Fungsi yang salah atau hilang
2. Kesalahan antar muka
3. Kesalahan dalam struktur data atau akses basis data eksternal
4. Kesalahan perilaku atau kinerja
5. Kesalahan inisialisasi dan penghentian

## **IV. PEMBAHASAN**

### **A. Perencanaan Sistem**

Perencanaan sistem adalah proses membuat sebuah laporan perencanaan sistem yang menggunakan sumber sistem informasi yang berhubungan dan mendukung tujuan bisnis dan operasi organisasi. Hal-hal yang diperlukan yaitu membuat studi kelayakan untuk Sistem Pelayanan Akademik Mahasiswa Pascasarjana Universitas Negeri Jakarta yang akan dibuat, seperti membuat kajian bagaimana proses sistem yang berjalan dengan sistem yang baru dan bagaimana pengaruhnya, sehingga dapat berfungsi secara maksimal.

Batasan ruang lingkup yang akan dirumuskan sebelum membuat perencanaan adalah mengumpulkan data, menentukan masalah dan mendiskusikan kebutuhan sistem dalam proses pelayanan akademik mahasiswa pascasarjana Universitas Negeri Jakarta.

Dengan permasalahan berupa proses pelayanan dan pencatatan akademik mahasiswa yang masih dilakukan dengan cara manual yaitu ditulis pada maka dibutuhkan perancangan sistem yang detail untuk dapat melakukan pencatatan topik penelitian, penentuan dosen pembimbing / promotor, pengajuan komprehensif, pengajuan seminar disertasi, seminar proposal, telaah disertasi, ujian tesis, ujian terbuka, dan ujian tertutup.

### **B. Analisa Sistem**

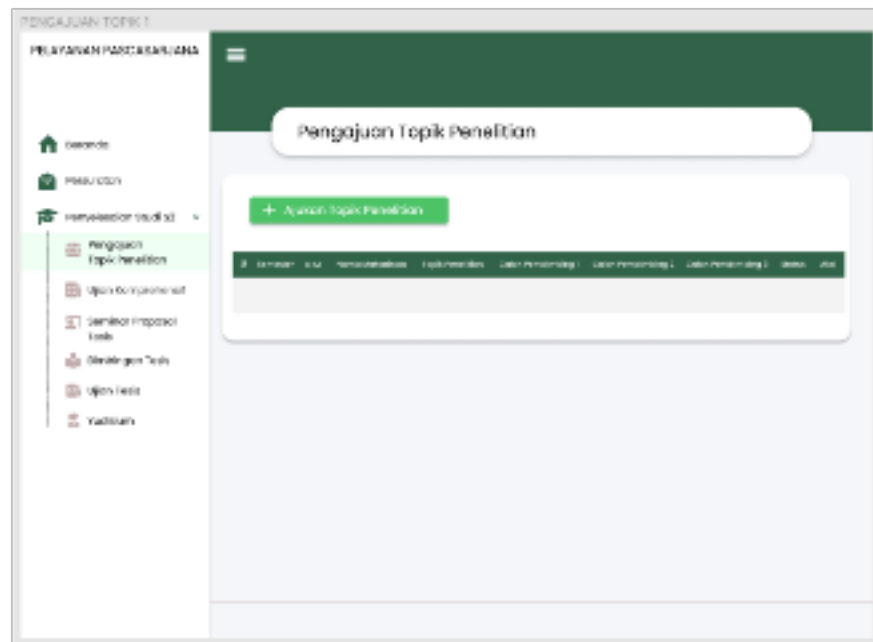
Pada saat ini proses administrasi yang berjalan pada Pascasarjana Universitas Negeri Jakarta masih menggunakan metode manual yaitu dengan ditulis pada buku mahasiswa masing masing, data topik penelitian, riwayat bimbingan, dan pengujian juga masih dicatat secara manual, sehingga memerlukan waktu yang cukup lama dalam mengelola data, memungkinkan terjadinya redundansi data, ketidakakuratan dalam proses pencatatan topik penelitian, serta keterlambatan dalam memberikan informasi maupun laporan kepada direktur pascasarjana. Dengan permasalahan tersebut maka Pascasarjana Universitas Negeri Jakarta dirasa perlu untuk merubah metode pencatatan dan pengadministrasian yang saat ini sedang berjalan, yaitu metode manual menjadi metode administrasi pembayaran yang terkomputerisasi.

### **C. Analisa Kebutuhan**

Analisa kebutuhan merupakan hal yang penting untuk mengetahui kebutuhan-kebutuhan yang nantinya akan digunakan untuk mendukung proses pembuatan sistem baru.

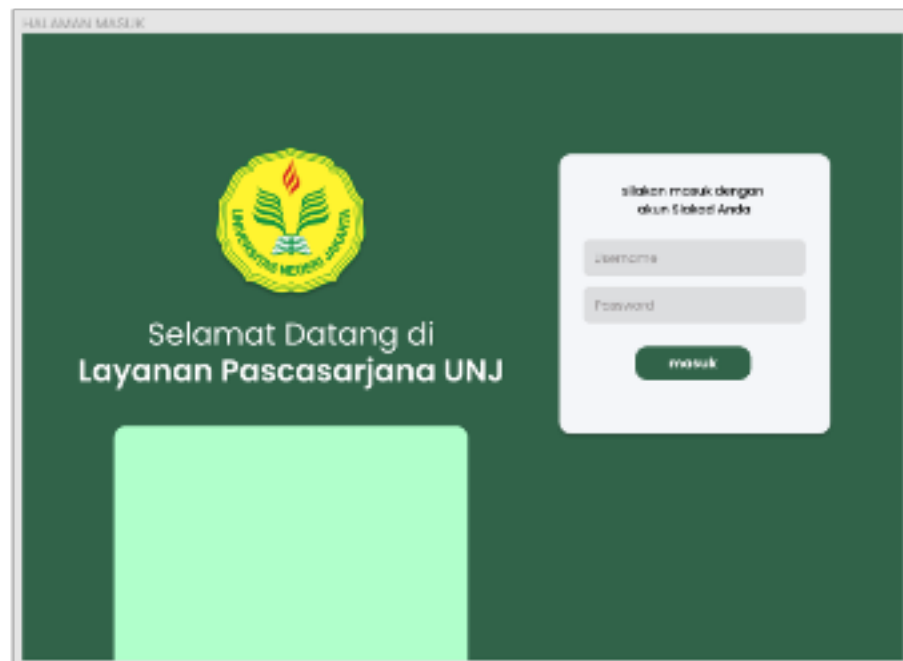
#### D. Perancangan Antar Muka

1. Desain Halaman Awal Sistem, terdapat pada Gambar 4.14 berikut



Gambar 4.14 Desain Halaman Awal Sistem

2. Desain Halaman Login, terdapat pada Gambar 4.15 berikut



Gambar 4.15 Desain Halaman Login

3. Desain Halaman Pengajuan, terdapat pada Gambar 4.16 berikut

The screenshot displays a web application interface for 'PENGALUAN TOPIC 2'. A sidebar on the left contains navigation links: 'Beranda', 'Pembuatan', 'Penyusunan', 'Pengeajuan Topik Tesis', 'Ujian Tesis', 'Bimbingan', 'Ujian Tesis', and 'Penyusunan'. The main content area features a modal window titled 'Pengeajuan Pembimbing Tesis'. This modal contains a sub-header 'Pengeajuan Judul / Topik Penelitian dan Calon Pembimbing Tesis' and a close button (X). The form fields are organized into two columns: 'Judul / Topik Penelitian' and 'File Topik Penelitian (Proposal Mini)' at the top; followed by three rows for 'NIDN Calon Pembimbing 1', 'NIDN Calon Pembimbing 2', and 'NIDN Calon Pembimbing 3', each with a corresponding 'Nama Lengkap Calon Pembimbing' field. A green 'Simpan' button is located at the bottom center of the modal.

## E. Implementasi Sistem

### 1. Implementasi Database

Penulis menggunakan MySQL untuk menyimpan data, Database diberi nama “pasca” yang berisi beberapa tabel, yaitu :



a. akun\_pasca

Name	Type	Length	Decimals	Not null	Virtual
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>
name	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>
email	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>
password	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>
role	int	11		<input type="checkbox"/>	<input type="checkbox"/>
created_at	timestamp			<input checked="" type="checkbox"/>	<input type="checkbox"/>

b. bimbingan

Name	Type	Length	Decimals	Not null	Virtual	Key
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama_bimbingan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama_peserta_bimbingan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
id_bimbingan	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
id_peserta	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
tanggal_bimbingan	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	
durasi	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
tempat	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
created_at	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	
updated_at	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	

c. komprehensif

Name	Type	Length	Decimals	Not null	Virtual	Key
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama_bimbingan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama_peserta_bimbingan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
id_bimbingan	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
id_peserta	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
tanggal_bimbingan	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	
durasi	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
tempat	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
created_at	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	
updated_at	timestamp			<input type="checkbox"/>	<input type="checkbox"/>	

h. seminar disertasi pembimbing status



i. seminar\_disertasi\_penguji

Save	Add Field	Insert Field	Delete Field	Primary Key	Move Up	Move Down	
Fields	Indexes	Foreign Keys	Checks	Triggers	Options	Comments	SQL Preview
Name			Type	Length	Decimal	Not null	Virtual
seminarid			int	11		<input type="checkbox"/>	<input type="checkbox"/>
nidn			varchar	255		<input type="checkbox"/>	<input type="checkbox"/>
nama			varchar	255		<input type="checkbox"/>	<input type="checkbox"/>
status			int	11		<input type="checkbox"/>	<input type="checkbox"/>
nilai			float			<input type="checkbox"/>	<input type="checkbox"/>
skorlinea1			int	11		<input type="checkbox"/>	<input type="checkbox"/>
skorlinea2			int	11		<input type="checkbox"/>	<input type="checkbox"/>
skorlinea3			int	11		<input type="checkbox"/>	<input type="checkbox"/>
skorlinea4			int	11		<input type="checkbox"/>	<input type="checkbox"/>
skorlinea5			int	11		<input type="checkbox"/>	<input type="checkbox"/>
skorlinea6			int	11		<input type="checkbox"/>	<input type="checkbox"/>
skorlinea7			int	11		<input type="checkbox"/>	<input type="checkbox"/>
skorlinea8			int	11		<input type="checkbox"/>	<input type="checkbox"/>
total			int	11		<input type="checkbox"/>	<input type="checkbox"/>

j. seminar\_proposal

Save	Add Field	Insert Field	Delete Field	Primary Key	Move Up	Move Down	
Fields	Indexes	Foreign Keys	Checks	Triggers	Options	Comments	SQL Preview
Name	Type	Length	Decimals	Not null	Virtual	Key	
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>		
nidn	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
sewaoran	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
kearahank	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
judul	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
file	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
panjang	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
pinjam	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
waktu/loan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
tempat/loan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
nilai	float			<input type="checkbox"/>	<input type="checkbox"/>		
created/c	timestamp			<input type="checkbox"/>	<input type="checkbox"/>		
updated/c	timestamp			<input type="checkbox"/>	<input type="checkbox"/>		

k. seminar\_proposal\_pembimbing\_status

Save	Add Field	Insert Field	Delete Field	Primary Key	Move Up	Move Down	
Fields	Indexes	Foreign Keys	Checks	Triggers	Options	Comments	SQL Preview
Name	Type	Length	Decimals	Not null	Virtual	Key	
sempid	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>		
nidn	varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>		
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>		

l. seminar\_proposal\_penguji

Save	Add Field	Insert Field	Delete Field	Primary Key	Move Up	Move Down	
Fields	Indexes	Foreign Keys	Checks	Triggers	Options	Comments	SQL Preview
Name	Type	Length	Decimals	Not null	Virtual	Key	
sempid	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>		
nidn	varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>		
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
nilai	float			<input type="checkbox"/>	<input type="checkbox"/>		
skorlinea1	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
skorlinea2	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
skorlinea3	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
skorlinea4	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
skorlinea5	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
skorlinea6	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
skorlinea7	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
total	int	11		<input type="checkbox"/>	<input type="checkbox"/>		

m. telaah\_disertasi

Save Add Field Insert Field Delete Field Primary Key Move Up Move Down									
Field	Indexes	Foreign Keys	Checks	Triggers	Options	Comment	SQL Preview		
Name	Type	Length	Decimals	Not null	Virtual	Key			
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>				
nm	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
username	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
localProdi	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
fileDraftDisertasi	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
idPencapaian	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
namaPembina	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
status	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
usiaPengantar	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
komputasi	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
ukuran	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
id	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
status	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
createdAt	timestamp			<input type="checkbox"/>	<input type="checkbox"/>				
updatedAt	timestamp			<input type="checkbox"/>	<input type="checkbox"/>				

n. token\_pasca

Save Add Field Insert Field Delete Field Primary Key Move Up Move Down									
Field	Indexes	Foreign Keys	Checks	Triggers	Options	Comment	SQL Preview		
Name	Type	Length	Decimals	Not null	Virtual	Key			
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>				
username	int	11		<input type="checkbox"/>	<input type="checkbox"/>				
token	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				

o. ujian\_terbuka

Save Add Field Insert Field Delete Field Primary Key Move Up Move Down									
Field	Indexes	Foreign Keys	Checks	Triggers	Options	Comment	SQL Preview		
Name	Type	Length	Decimals	Not null	Virtual	Key			
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>				
nm	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
username	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
localProdi	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
judul	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
variabel	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
file	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
fileDisertasi	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
fileGugatan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
fileKlausur	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
nilai	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
id	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
waktu	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
tanggal	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
nilai	float			<input type="checkbox"/>	<input type="checkbox"/>				
pass	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
createdAt	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
updatedAt	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				

p. ujian terbuka pembimbing status

Save Add Field Insert Field Delete Field Primary Key Move Up Move Down									
Field	Indexes	Foreign Keys	Checks	Triggers	Options	Comment	SQL Preview		
Name	Type	Length	Decimals	Not null	Virtual	Key			
userId	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>				
id	varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>				
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>				
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>				


q. ujian\_terbuka\_penguji

Name	Type	Length	Decimal	Not null	Virtual	Key
ujidid	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>	1
nama	varchar	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>	2
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
nilai	float			<input type="checkbox"/>	<input type="checkbox"/>	
koordinat1	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
koordinat2	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
koordinat3	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
koordinat4	int	11		<input type="checkbox"/>	<input type="checkbox"/>	
koordinat5	int	11		<input type="checkbox"/>	<input type="checkbox"/>	

r. ujian\_tertutup

Name	Type	Length	Decimal	Not null	Virtual	Key
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>	1
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama2	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama3	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama4	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama5	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama6	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama7	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama8	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama9	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama10	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama11	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama12	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama13	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama14	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama15	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama16	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama17	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama18	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama19	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama20	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama21	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama22	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama23	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama24	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama25	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama26	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama27	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama28	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama29	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama30	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama31	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama32	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama33	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama34	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama35	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama36	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama37	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama38	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama39	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama40	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama41	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama42	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama43	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama44	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama45	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama46	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama47	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama48	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama49	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama50	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama51	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama52	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama53	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama54	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama55	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama56	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama57	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama58	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama59	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama60	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama61	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama62	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama63	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama64	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama65	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama66	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama67	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama68	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama69	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama70	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama71	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama72	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama73	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama74	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama75	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama76	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama77	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama78	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama79	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama80	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama81	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama82	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama83	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama84	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama85	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama86	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama87	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama88	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama89	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama90	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama91	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama92	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama93	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama94	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama95	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama96	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama97	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama98	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama99	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama100	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama101	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama102	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama103	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama104	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama105	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama106	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama107	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama108	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama109	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama110	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama111	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama112	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama113	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama114	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama115	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama116	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama117	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama118	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama119	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama120	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama121	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama122	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama123	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama124	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama125	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama126	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama127	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama128	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama129	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama130	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama131	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama132	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama133	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama134	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama135	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama136	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama137	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama138	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama139	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama140	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama141	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama142	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama143	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama144	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama145	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama146	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama147	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama148	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama149	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama150	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama151	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama152	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama153	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama154	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama155	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama156	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama157	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama158	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama159	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama160	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama161	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama162	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama163	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama164	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama165	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama166	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama167	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama168	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama169	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama170	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama171	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama172	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama173	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama174	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama175	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama176	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama177	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama178	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama179	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama180	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama181	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama182	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama183	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama184	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama185	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama186	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama187	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama188	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama189	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama190	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama191	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama192	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama193	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama194	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama195	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama196	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama197	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama198	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama199	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama200	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama201	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama202	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama203	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama204	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama205	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama206	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama207	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama208	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama209	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama210	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama211	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
nama212	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	

u. ujian\_tesis

Name	Type	Length	Nullable	Valid	Key
id	int	11	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
name	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
username	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
email	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
last_name	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
first_name	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
sex	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
password	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
article_id	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
title	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
body	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
summary	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
tags	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
post	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
author	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
status	int	11	<input type="checkbox"/>	<input type="checkbox"/>	
timestamp	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
meta_title	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
meta_desc	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	
url_slug	varchar	255	<input type="checkbox"/>	<input type="checkbox"/>	

v. ujian\_tesis\_pembimbing\_status

Name	Type	Length	Nullable	Default	Key
id	int	11	<input checked="" type="checkbox"/>	---	
nick	varchar	255	<input checked="" type="checkbox"/>	---	
name	varchar	255	<input type="checkbox"/>	---	
status	varchar	255	<input type="checkbox"/>	---	

w. ujian tesis penguji

Field	Indexes	Foreign Keys	Checks	Triggers	Options	Comment	SQL Preview
Name	Type	Length	Decimal	Not null	Virtual	Key	
tsord	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
id	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
name	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
value	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
real	float			<input type="checkbox"/>	<input type="checkbox"/>		
storehouse1	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
storehouse2	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
storehouse3	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
storehouse4	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
storehouse5	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
storehouse6	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
storehouse7	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
date	int	11		<input type="checkbox"/>	<input type="checkbox"/>		

x. use logs

Name	Type	Length	Decimal	Not null	Virtual	Key
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>	
identifier	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
log	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	
createdat	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>	

y. yudisium

Show   Add Field   Insert Field   Delete Field   Primary Key   Move Up   Move Down							
Field	Index	Foreign Key	Checks	Triggers	Options	Comment	SQL Preview
Name	Type	Length	Decimals	Not null	Virtual	Key	
id	int	11		<input checked="" type="checkbox"/>	<input type="checkbox"/>	Key	
nam	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
nama_lu	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
nama	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
kodePadi	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
formPembelian	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
jumlah	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
sertifikatHati	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
diBj	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
arsol	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
foto	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
namaFis	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
di	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
status	int	11		<input type="checkbox"/>	<input type="checkbox"/>		
pesan	varchar	255		<input type="checkbox"/>	<input type="checkbox"/>		
createdAt	timestamp			<input type="checkbox"/>	<input type="checkbox"/>		
updatedAt	timestamp			<input type="checkbox"/>	<input type="checkbox"/>		

## 2. Implementasi Tampilan Sistem

Tampilan aplikasi terdiri dari login, dashboard mahasiswa, dashboard dosen, dashboard direktur pascasarjana. Untuk lebih jelasnya dapat dilihat pada tampilan berikut :

### a. Tampilan halaman login



### b. Tampilan dashboard mahasiswa



c. Tampilan dashboard dosen



d. Tampilan dashboard direktur pascasarjana



## **V. PENUTUP**

### **A. Kesimpulan**

Berdasarkan pembahasan sebelumnya, penulis dapat mengambil beberapa kesimpulan sebagai berikut:

1. Sistem Pelayanan Akademik Mahasiswa Pascasarjana ini dapat meminimalisir terjadinya redudansi data pada Pascasarjana Universitas Negeri Jakarta.
2. Dengan adanya sistem ini, semua kegiatan dapat terkomputerisasi dan terdata dengan rapih, sehingga memudahkan pengambilan data sewaktu-waktu diperlukan.
3. Sistem Pelayanan Akademik ini dapat mempermudah dosen dan mahasiswa dalam proses bimbingan tesis dan disertasi pada Pascasarjana Universitas Negeri Jakarta.
4. Sistem ini dikembangkan dengan bahasa pemrograman PHP dengan menggunakan framework laravel yang memiliki fitur-fitur yang dapat memudahkan penggunaanya serta memudahkan bagi pengembang yang akan melanjutkan pengembang sistem dengan fitur-fitur baru.
5. Sistem ini dapat mengurangi resiko hilangnya topik penelitian mahasiswa karena data tersimpan di database serta mengurangi tumpukan dokumen karena tidak memakan tempat dan mempermudah dalam pencarian data ketika dibutuhkan

### **B. Saran**

Berdasarkan kesimpulan diatas untuk pengembangan lebih lanjut, penulis memberikan saran untuk pengembangan dari Sistem Pelayanan Akademik Pascasarjana Universitas Negeri Jakarta ini yaitu sebagai berikut :

1. Perlu adanya penambahan fitur notifikasi pada sistem terkait pengajuan topik penelitian, pengajuan bimbingan, pengajuan pengujian dan pengajuan lain nya.
2. Perlu adanya penambahan komunikasi langsung ( live chat ) agar mempermudah mahasiswa menghubungi dosen pembimbingnya.
3. Perlu adanya penambahan fitur rekapitulasi data pada halaman direktur pascasarjana agar direktur dapat mudah dalam merekap data.



## VI. REFRENSI

- Dewi, Irra Chrisyanti. 2011, *Pengantar Ilmu Administrasi*. PT Prestasi Pustakaraya, Jakarta.
- Jogiyanto, 2010, *Analisis dan Desain Sistem Informasi*, Edisi IV, Andi Offset, Yogyakarta
- Khan,M.Sajid.,Naumann,Earl.,Williams,Paul. 2012. *Identifying The Key Drivers of Customer Satisfaction and Repurchase Intentions An Empirical Investigation of Japanese B2B Services*.American University of Sharjah, United Arab Emirates.
- Nugroho, Bunafit. 2013, *Dasar Pemrograman Web PHP – MySQL dengan Dreamweaver*, Gava Media, Yogyakarta.
- Simarmata, Janner 2010, *Rekayasa Perangkat Lunak*, Andi Offset, Yogyakarta.
- Sommerville, Ian. 2011. *Software Engineering (Rekayasa Perangkat Lunak)*. Erlangga, Jakarta.
- Sondang P. Siagian. 2010. *Administrasi Pembangunan*, Bumi Aksara, Jakarta.
- Rohman, A. 2014. *Mengenal Framework “Laravel” (Best PHP Frameworks For 2014)*.  
ilmuit.org
- Qteishat, M.K., Alshhibli, H.H., Al-ma’aitah, M.A. 2014. *The impact of e-ticketing technique customer satisfaction: an empirical analysis*. SISTEM - Journal of Information Systems and Technology Management ISSN online: 1807-1775