

**MEMBANGUN APLIKASI SELEKSI KELAYAKAN
PENGAJUAN PROPOSAL SKRIPSI BERBASIS WEB
DI PROGRAM STUDI TEKNIK INFORMATIKA
UNIVERSITAS BALE BANDUNG**

SKRIPSI

Karya Tulis sebagai Syarat untuk Memperoleh
Gelar Sarjana Komputer dari Fakultas Teknologi Informasi
Universitas Bale Bandung

Disusun Oleh:

**ROYNA NASRULLOH
NPM. C1A160043**



**PROGRAM STRATA 1
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG
BANDUNG**

2020

LEMBAR PERSETUJUAN PEMBIMBING

Membangun Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis
Web Di Program Studi Teknik Informatika Universitas Bale Bandung

Disusun Oleh:

ROYNA NASRULLOH
NPM. C1A160043

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

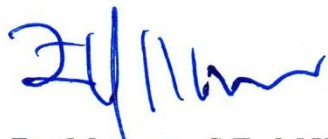
Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2020

Disetujui Oleh:

Pembimbing 1



Zen Munawar, S.T., M.Kom.
NIDN. 0422037002

Pembimbing 2



Sutiyono, S.T., M.Kom.
NIDN. 0407047706

LEMBAR PENGESAHAN PENGUJI

Membangun Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis
Web Di Program Studi Teknik Informatika Universitas Bale Bandung

Disusun Oleh:

ROYNA NASRULLOH
NPM. C1A160043

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

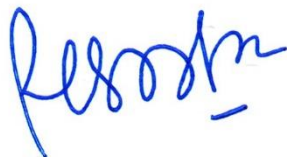
Pada

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2020

Disetujui Oleh:

Penguji 1



Rosmalina, ST., M.Kom.
NIDN. 04104808122

Penguji 2



Iim Abdurohim, M.T.,
NIDN. 0413107002

LEMBAR PENGESAHAN PROGRAM STUDI

Membangun Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis
Web Di Program Studi Teknik Informatika Universitas Bale Bandung

Disusun Oleh:

ROYNA NASRULLOH
NPM. C1A160043

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar
SARJANA KOMPUTER

Pada

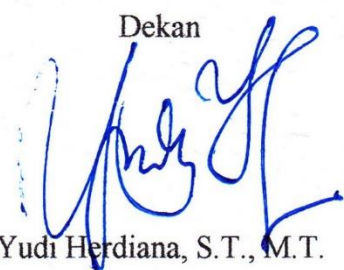
**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2020

Disetujui Oleh:

Mengetahui,


Dekan


Yudi Herdiana, S.T., M.T.

NIK. 04104808008

Mengesahkan,

Ketua Program Studi


Yaya Suharya, S.Kom., M.T.,

NIK. 01043170007

LEMBAR PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Royna

NIM : C1A160043

Judul Skripsi : **MEMBANGUN APLIKASI SELEKSI KELAYAKAN
PENGAJUAN PROPOSAL SKRIPSI BERBASIS WEB DI PROGRAM
STUDI TEKNIK INFORMATIKA UNIVERSITAS BALE BANDUNG**

Menyatakan dengan sebenarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan pemogramana yang tercantum sebagai bagian dari skripsi ini. Jika terdapat karya orang lain, saya mencantumkan sumber yang jelas.

Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS BALE BANDUNG.

Demikian surat pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun

Baleendah, Juli 2020

Yang membuat
Pernyataan

ROYNA NASRULLOH
NIM. C1A160043

ABSTRAK

Skripsi adalah istilah yang digunakan di Indonesia untuk mengilustrasikan suatu karya tulis ilmiah berupa paparan tulisan hasil penelitian sarjana S1 yang membahas suatu permasalahan/fenomena dalam bidang ilmu tertentu dengan menggunakan kaidah-kaidah yang berlaku, Skripsi bertujuan agar mahasiswa mampu menyusun dan menulis suatu karya ilmiah. Dalam melakukan pengajuan proposal skripsi khususnya di Program Studi Teknik Informatika Universitas Bale Bandung masih menggunakan cara konvensional yaitu dengan pengumpulan data secara *offline*.

Dengan cara tersebut tentu tidak efisien karena mahasiswa harus datang ke bagian tata usaha untuk melakukan pengumpulan persyaratan skripsi dan jika masih ada yang salah mahasiswa harus kembali ke bagian tata usaha tentu itu tidak efektif serta merepotkan bagian tata usaha karena mahasiswa yang cukup banyak di Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung Aplikasi pengumpulan persyaratan proposal skripsi merupakan sebuah sistem untuk memudahkan mahasiswa dalam pengumpulan persyaratan skripsi khususnya di Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.

Dengan adanya aplikasi ini tentu membantu proses pengelolaan persyaratan skripsi, serta dalam perancangan aplikasi ini dengan menggunakan metode Waterfall maka akan sangat memudahkan pengguna serta mudah dipahami serta petugas akan sangat memudahkan mahasiswa dan tata usaha dalam melihat persyaratan yang sudah terpenuhi untuk setiap mahasiswanya.

Kesimpulan dari penelitian kali ini adalah Membangun Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis Web di Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung untuk memudahkan proses pengumpulan persyaratan skripsi serta memudahkan penentuan mahasiswa layak skripsi maupun dengan yang tidak dilihat dari pengumpulan persyaratan skripsi.

Kata Kunci: *Seleksi Kelayakan, Pengajuan Proposal, Skripsi, Web.*

ABSTRACT

Skripsi is a term used in Indonesia to illustrate a scientific paper in the form of exposure to the results of undergraduate research that discusses a problem / phenomenon in a particular field of science by using applicable rules, the thesis aims to make students able to compile and write a scientific work. In submitting a thesis proposal, especially in the Informatics Engineering Study Program at the University of Bale Bandung, it still uses the conventional method of offline data collection.

In this way it is certainly inefficient because students must come to the administration department to collect Skripsi requirements and if there is still something wrong the student must return to the administration section of course it is ineffective and troublesome the administration section because there are quite a lot of students in the Engineering Study Program Informatics Faculty of Information Technology, University of Bale Bandung and will spend a lot of time, besides the conventional collection of requirements will be constrained if the weather is raining, it is not possible to collect the requirements for submission of skripsi proposals. Skripsi proposal application collection application is a system to facilitate students in collecting thesis requirements, especially in the Information Engineering Study Program, Faculty of Information Technology, Bale Bandung University.

With this application, it certainly helps to help the process of managing skripsi requirements, and in designing this application using the Waterfall method it will greatly facilitate the user and be easily understood and officers will greatly facilitate students and administration in seeing the requirements that have been met for each student.

The conclusion of this research is to build the application for Web-Based Skripsi Proposal Selection in Information Engineering Study Program, Faculty of Information Technology, Bale Bandung University to facilitate the process of collecting thesis requirements and facilitate the determination of skripsi-worthy students and those not seen from skripsi-gathering requirements.

Keywords: Eligibility Selection, Submission of Proposals, Thesis, Web.

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadirat Allah SWT yang mana berkat Rahmat dan Karunia-Nya penulis dapat menyelesaikan Laporan Penelitian ini dengan judul “Membangun Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis Web Di Program Studi Teknik Informatika Universitas Bale Bandung” dengan baik dan tepat pada waktunya. Penelitian ini disusun untuk menyelesaikan tugas akhir guna memperoleh gelar Sarjana Komputer.

Laporan ini dibuat dengan berbagai observasi, wawancara dan beberapa bantuan dari berbagai pihak yang membantu menyelesaikan proses pengerjaan Penelitian ini. Oleh karena itu, penulis ucapkan terima kasih sebesar-besarnya kepada:

1. Bapak Yudi Herdiana, S.T., M.T., selaku Dekan Fakultas Teknologi Informasi Universitas Bale Bandung.
2. Bapak Yaya Suharya, S.Kom., M.T., selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
3. Bapak Zen Munawar, S.T., M.Kom., selaku pembimbing satu di Prodi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
4. Sutiyono, S.T., M.Kom., selaku pembimbing dua di Prodi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
5. Rosmalina, ST., M.Kom selaku penguji satu di Prodi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
6. Iim Abdurrohman, M.T., selaku Penguji dua di Prodi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
7. Keluarga yang memberikan dukungan secara moral dan materil serta do'a yang tak pernah putus dalam proses pengerjaan laporan.
8. Rekan-rekan seperjuangan yang saling membantu dalam proses pengerjaan laporan penelitian.

Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan, oleh karena itu kritik dan saran yang membangun dari semua pihak yang bersifat membangun selalu penulis harapkan demi kesempurnaan laporan ini.

Bandung, Juli 2020

Royana Nasrulloh

DAFTAR ISI

LEMBAR PERSETUJUAN PEMBIMBING	i
LEMBAR PENGESAHAN PENGUJI	ii
LEMBAR PENGESAHAN PROGRAM STUDI.....	iii
LEMBAR PERNYATAAN	iv
ABSTRAK	v
ABSTRACT.....	vi
KATA PENGANTAR	vii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian	2
1.5 Metodologi Penelitian	3
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 Landasan Teori.....	6
2.2 Dasar Teori.....	8
2.2.1 Aplikasi	8
2.2.2 Seleksi	8
2.2.3 Skripsi	9
2.2.4 World Wide Web	10
2.2.5 Metode SDLC	10
2.2.6 Sublime Text.....	12
2.2.7 Web Browser.....	13
2.2.8 Web Server.....	14
2.2.9 XAMPP	15
2.2.10 HyperText Markup Language (HTML)	16
2.2.11 Cascading Style Sheet (CSS)	17
2.2.12 PHP	17

2.2.13	MYSQL.....	19
2.2.14	Unified Modelling Language (UML)	23
2.2.15	Bootstrap	28
BAB III METODOLOGI PENELITIAN.....		30
3.1	Kerangka Pikir	30
3.2	Deskripsi	31
2.2.1.	Pengumpulan Data.....	31
2.2.2.	Rumusan Masalah.....	32
2.2.3.	Metode Waterfall	32
BAB IV ANALISIS DAN PERANCANGAN		34
4.1	Analisis.....	34
4.1.1	Instrumen Penelitian.....	34
4.1.2	Analisis Sistem.....	35
4.1.3	Analisis Kebutuhan	36
4.1.4	Hasil Analisis	38
4.2	Perancangan	38
4.2.1	Diagram Arus Data UML.....	39
BAB V IMPLEMENTASI DAN PENGUJIAN		66
5.1	Implementasi	66
5.1.1	User Interface	66
5.2	Pengujian.....	78
5.2.1	Kegagalan Pengujian dengan Menggunakan Metode Black Box	79
5.2.1	Hasil Pengujian dengan Menggunakan Metode Black Box.....	80
BAB VI KESIMPULAN DAN SARAN		84
6.1	Kesimpulan	84
6.2	Saran	84
DAFTAR PUSTAKA		1
DAFTAR LAMPIRAN.....		3

DAFTAR TABEL

Tabel 2.1. Tipe Data Numerik.....	18
Tabel 2.2. Tipe Data Tanggal dan Waktu.....	20
Tabel 2.3. Use Case Diagram.....	24
Tabel 2.4. Activity Diagram.....	26
Tabel 4.1. Perangkat Keras	37
Tabel 4.2 Server Hosting	38
Tabel 4.3 Deskripsi Tata Usaha dan Mahasiswa	40
Tabel 4.4 Use Case Tata Usaha	41
Tabel 4.5 Use Case Mahasiswa	41
Tabel 4.6 Tabel Admin	51
Tabel 4.7 Tabel File Upload	52
Tabel 4.8 Tabel File	52
Tabel 4.9 Tabel Mahasiswa	53
Tabel 4.10 Tabel Pemberitahuan	53
Tabel 4.11 Tabel Pesan	54
Tabel 5.1 Pengujian Kesalahan	79
Tabel 5.2 Pengujian Perangkat Admin	80
Tabel 5.3 Pengujian Perangkat Mahasiswa	82

DAFTAR GAMBAR

Gambar 1.1 Ilustrasi Model Waterfall.....	4
Gambar 1.2 Flowcart Alur Aplikasi.....	6
Gambar 2.1 Diagram UML.....	23
Gambar 3.1 Kerangka Pikir.....	29
Gambar 4.1 Use Case	42
Gambar 4.2 Activity Login Mahasiswa	43
Gambar 4.3 Activity Pengumpulan Persyaratan Pertama.....	43
Gambar 4.4 Activity Pengumpulan Peryaratan Kedua.....	44
Gambar 4.5 Activity Pesan.....	45
Gambar 4.6 Activity Edit Profil Mahasiswa.....	45
Gambar 4.7 Activity Logout Mahasiswa	46
Gambar 4.8 Activity Login Admin.....	46
Gambar 4.9 Activity Update Status Mahasiswa.....	47
Gambar 4.10 Activity Tambah Mahasiswa.....	48
Gambar 4.11 Activity Edit Informasi.....	49
Gambar 4.12 Activity Cetak Rekap Mahasiswa.....	50
Gambar 4.13 Activity Logout.....	51
Gambar 4.14 Class Diagram.....	52
Gambar 4.15 Mocup Login Mahasiswa.....	53
Gambar 4.16 MockUp Persyaratan Pertama	53
Gambar 4.17 MockUp Persyaratn Kedua	54
Gambar 4.18 MockUp Tampilan Menu Pesan.....	55
Gambar 4.19 Mockup Tampilan Login Admin.....	55
Gambar 4.20 MockUp Tampilan Menu Utama.....	56
Gambar 4.21 MockUp Tampilan Menu Mahasiswa.....	56
Gambar 4.22 MockUp Tambah Mahasiswa	57
Gambar 4.23 Mockup Tampilan Rekap Mahasiswa.....	58

Gambar 4.24 MockUp Tampilan Update Status.....	59
Gambar 4.25 MockUp Tampilan Menu Pesan.....	60
Gambar 4.26 MockUp Tampilan Menu Informasi.....	60
Gambar 4.27 MockUp Tampilan Ganti Passwrd Admin.....	61
Gambar 4.28 MockUp Tampilan Logout.....	62
Gambar 5.1. User Interface Login Mahasiswa.....	63
Gambar 5.2 user interface User Interface Tampilan Menu Utama.....	64
Gambar 5.3 User Interface Tampilan Upload Peryaratan Pertama.....	65
Gambar 5.4 User Interface Tampilan Menu Pesan.....	65
Gambar 5.5 User Interface Tampilan Logi Admin.....	67
Gambar 5.6 User Interface Tampilan Menu Utama.....	67
Gambar 5.7 User Interface Tampilan Menu Mahasiswa.....	68
Gambar 5.8 User Interface Tampilan Menu Pilih rekap mahasiswa.....	68
Gambar 5.9 User Interface Tampilan Update Status.....	69
Gambar 5.10 User Interface Tambah Mahasiswa.....	69
Gambar 5.11 User Interface Tampilan Menu Pesan.....	70
Gambar 5.12 User Interface Tampilan Menu Informasi.....	71
Gambar 5.13 User Interface Tampilan Menu Ganti Password.....	71
Gambar 5.14 User Interface Tampilan Logout Admin.....	72
Gambar 5.15 Tampilan Kesalahan Login.....	77
Gambar 5.16 Tampilan Kesalahan Upload File.....	78

.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan zaman, teknologi telah mengalami banyak perubahan dari masa ke masa. Karena teknologi lebih memudahkan manusia dalam beraktivitas salah satunya aplikasi berbasis WEB. *Web* adalah sebuah kumpulan halaman pada suatu domain di internet yang dibuat dengan tujuan tertentu dan saling berhubungan serta dapat diakses secara luas melalui halaman depan (*home page*) menggunakan sebuah browser menggunakan *URL website*.

Skripsi adalah istilah yang digunakan di Indonesia untuk mengilustrasikan suatu karya tulis ilmiah berupa paparan tulisan hasil penelitian sarjana S1 yang membahas suatu permasalahan/fenomena dalam bidang ilmu tertentu dengan menggunakan kaidah-kaidah yang berlaku, skripsi bertujuan agar mahasiswa mampu menyusun dan menulis suatu karya ilmiah.

Untuk persyaratan pengajuan skripsi di Program Studi Teknik Informatika Universitas Bale Bandung, masih menggunakan cara konvensional yaitu dengan pengumpulan data secara *Offline* karena mahasiswa harus pergi ke bagian Tata Usaha dan jika data yang dikumpulkan masih ada yang salah, mahasiswa tersebut harus kembali ke bagian tata usaha untuk melakukan konfirmasi. Tentu itu sangat tidak efektif karena Mahasiswa harus datang ke Tata Usaha dan itu sangat merepotkan Tata Usaha dikarenakan mahasiswa cukup banyak.

Dengan ini penulis berencana membangun aplikasi guna membantu Mahasiswa dan Tata Usaha dengan aplikasi yang berjudul “*Membangun Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis Web di Program Studi Teknik Informatika Universitas Bale Bandung*”.

1.2 Rumusan Masalah

Adapun rumusan masalah yakni sebagai berikut:

1. Bagaimana Mahasiswa agar lebih mudah dalam mengumpulkan persyaratan skripsi
2. Bagaimana prosedur yang tadinya panjang menjadi sederhana.
3. Bagaimana mengimplementasikan aplikasi agar dapat di akses oleh Mahasiswa dan Tata Usaha.

1.3 Batasan Masalah

Pembatasan suatu masalah digunakan untuk menghindari adanya penyimpangan maupun pelebaran pokok masalah supaya penelitian tersebut lebih terarah dan memudahkan dalam pembahasan sehingga tujuan penelitian akan tercapai. Beberapa batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Aplikasi tersebut hanya dapat digunakan oleh Mahasiswa dan Tata Usaha di Program Studi Teknik Informatika Universitas Bale Bandung.
2. Aplikasi Ini hanya menggunakan Teknologi Web seperti CSS, PHP, dan MySQL.
3. Aplikasi tersebut hanya melakukan pengumpulan persyaratan skripsi

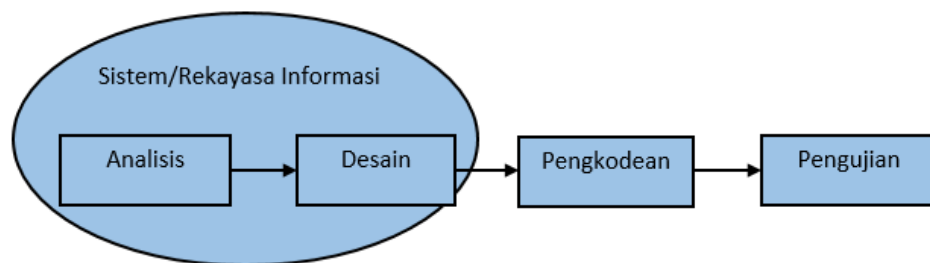
1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

1. Memudahkan Mahasiswa dalam pengumpulan persyaratan skripsi.
2. Memeudahkan serta menyederhanakan prosedur yang tadinya panjang menjadi pendek.

1.5 Metodologi Penelitian

Metode yang digunakan dalam pembangunan aplikasi seleksi kelayakan pengajuan proposal skripsi berbasis web ini adalah waterfall. Alasan menggunakan metode ini adalah karena metode waterfall melakukan pendekatan secara sistematis dan berurutan dalam membangun suatu sistem. Proses metode waterfall yaitu pada pengerjaan dari suatu sistem dilakukan secara berurutan. Sistem yang dihasilkan akan berkualitas baik, dikarenakan perancangannya secara bertahap sehingga tidak terfokus pada tahap tertentu.



Gambar 1.1 Ilustrasi Model Waterfall

- Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi Kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

- Desain

Desain perangkat lunak adalah proses multistep yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dapat dihasilkan pada tahap ini juga perlu didokumentasikan.

- **Pembuatan Kode Program**

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

- **Pengujian**

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

- **Pendukung (support) atau Pemeliharaan (Maintenance)**

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.(Rosa A.S, M.Shalahuddin,2013:29-30)

1.6 Sistematika Penulisan

Gambaran mengenai keseluruhan skripsi dan pembahasannya dapat dijelaskan dalam sistematika penulisan sebagai berikut:

- a. **Bab 1 Pendahuluan**

Bagian pendahuluan menjelaskan mengenai latar belakang masalah, Rumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian, sistematika penulisan.

- b. **Bab II Tinjauan Pustaka**

Bagian ini membahas mengenai kajian teoritis yang meliputi Landasan Teori, Dasar Teori

c. Bab III Metodologi Penelitian

Bagian ini membahas mengenai komponen dari metode penelitian yaitu Kerangka Pikir, Dasar Teori

d. Bab IV Analisis dan Perancangan

Bagian ini membahas mengenai deskripsi hasil dan temuan penelitian sesuai dengan rumusan masalah atau pertanyaan penelitian berupa. Analisis, Instrumen Penelitian, Analisis Sistem, Analisis Kebutuhan, Hasil Analisis, Perancangan

e. Bab V Implementasi dan Pengujian

Pada bab ini membahas mengenai pengimplementasian suatu pemograman yang telah dibuat setelah itu melakukan tahap pengujian pada aplikasi yang dibangun. Implementasi, Pengujian

f. Bab VI Kesimpulan dan Saran

Pada bab akhir laporan skripsi ini membahas mengenai kesimpulan dan saran terhadap aplikasi yang telah penulis bangun . Kesimpulan, Saran

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

Pada penelitian kali ini penyusun mempelajari penelitian terdahulu sebagai acuan dalam Implementasi Metode *Waterfall* Pada Pembuatan Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis Web sebagai berikut:

1. Diyan Agus Pratama, Rizki Yudhi Dewantara (2018), yang berjudul “Analisis Dan Perancangan Sistem Informasi Perekrutan Karyawan Berbasis Web”

“Pada penelitian ini mengemukakan bagaimana rancangan sistem informasi rekrutmen karyawan berbasis web yang melakukan beberapa tahap dalam analisis dan perancangan sistem diantaranya 1. Investigasi sistem, 2. Analisis sistem, 3. Perancangan sistem, 4. Pengujian sistem, 5. Pengembangan sistem. Serta dalam penelitian ini menggunakan metode deskriptif kualitatif sumber data yang digunakan adalah data primer dan sekunder yang diperoleh dari metode pengumpulan data berupa wawancara, dan dokumentasi. Berdasarkan penelitian yang telah dilakukan, sistem informasi rekrutmen karyawan yang diterapkan di PT Sumber Abadi Bersama masih menggunakan cara manual sehingga menyebabkan beberapa permasalahan. Permasalahan tersebut diantaranya menyeleksi berkas lamaran dengan kualifikasi yang dibutuhkan perusahaan. Dengan menggunakan beberapa gambaran dalam memebangun sistem informasi rekrutmen karyawan diantaranya Flowchart, flow diagram. Hingga kan menghasilkan sebuah rancangan serta gambaran sistem informasi rekrutmen karyawan tersebut”.

2. Ika Yuniva, Ahmad Syafi’I (2018), “Pendekatan Model Waterfall Dalam Perancangan Web Sistem Informasi Penerimaan Siswa Baru Menggunakan Framework Bootstrap”.

“pada penelitian ini penulis mendapatkan kesimpulan bahwa metode waterfall yang digunakan sebagai metode penelitian memiliki beberapa tahapan diantaranya: 1. Analisis kebutuhan perangkat lunak yaitu aplikasi atau Software yang diperlukan penulis untuk perancangan aplikasi yang akan penulis buat. 2. Desain yaitu penulis akan merancang proses multistep yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. 3. Pembuatan Kode Program yaitu desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain. 4. Pengujian yaitu pengujian fokus pada perangkat lunak secara logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. 5. Pendukung atau pemeliharaan (*maintenance*) yaitu ketika sebuah web yang telah dibuat terdapat kesalahan yang muncul dan tidak terdeteksi saat pengujian maka atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru”.

3. **Fera Zulkarnain, (2015) yang berjudul “Analisis Pengembangan Sistem Pendataan Siswa Berbasis PHP Dan Mysql Guna Mempermudah Pengelolaan Data Siswa Di SMK N 2 Wonosari ”.**

“pada penelitian ini penulis mendapatkan kesimpulan bahwa pendataan siswa mencakup beberapa kebutuhan diantaranya: 1. Identitas 2. Data orang tua 3. Data kondisi rumah 4. Kondisi sosial 5. Akademik siswa. Pada penelitian ini penulis membangun aplikasi agar mempermudah Tata Usaha dalam mengelola data Siswa dengan melakukan beberapa tahap dalam pengumpulan data seperti yang telah di sebutkan di atas.”

2.2 Dasar Teori

Pada pembuatan aplikasi ini penyusun mempunyai bahan dan alat untuk pembuatan aplikasi maka penyusun mempelajari teori-teori yang ada sebagai acuan dalam Implementasi aplikasi yang akan penulis buat yaitu sebagai berikut:

2.2.1 Aplikasi

Aplikasi adalah suatu perangkat lunak (*software*) atau program komputer yang beroperasi pada sistem tertentu yang diciptakan dan dikembangkan untuk melakukan perintah tertentu.

Menurut Hengky W, Pramana. Pengertian aplikasi adalah suatu perangkat lunak yang dibuat khusus untuk memenuhi kebutuhan berbagai aktivitas dan pekerjaan, misalnya; pelayanan masyarakat, aktivitas niaga, periklanan, game, dan berbagai aktivitas lainnya.

Menurut Rachmad Hakim S, pengertian aplikasi adalah sebuah *software* yang dibuat untuk tujuan tertentu, misalnya; untuk mengelola dokumen, permainan (*game*), dan lain sebagainya.

Menurut Harip Santoso, pengertian aplikasi adalah suatu kelompok file (*report, class, form*) yang dibuat untuk mengeksekusi kegiatan tertentu yang saling berhubungan, misalnya; aplikasi *fixed asset* dan aplikasi *payroll*.

2.2.2 Seleksi

Seleksi adalah usaha pertama yang dilakukan perusahaan untuk memperoleh karyawan yang kualifikasi dan kompeten yang akan menjabat serta mengerjakan suatu pekerjaan pada perusahaan atau instansi.

Menurut Malayu Hasibuan, seleksi adalah suatu kegiatan pemilihan dan penentuan pelamar yang akan diterima atau ditolak untuk menjadi karyawan perusahaan. Seleksi ini didasarkan kepada spesifikasi tertentu dari setiap perusahaan bersangkutan.

Menurut Ambar T Sulistiyani dan Rosidah seleksi merupakan serangkaian langkah kegiatan yang dilaksanakan untuk memutuskan apakah seseorang pelamar

diterima atau ditolak, dalam suatu instansi tertentu setelah menjalani serangkaian tes yang dilaksanakan.

2.2.3 Skripsi

Menurut Drs. Djarwanto (1992) skripsi merupakan sebuah karya ilmiah yang disusun oleh seorang mahasiswa program sarjana dari hasil-hasil penelitiannya atas dasar analisis data primer dan atau analisis sekunder.

Jika melihat definisi skripsi dalam kamus besar Bahasa Indonesia (KBBI), skripsi merupakan kerangka ilmiah yang bawib ditulis oleh mahasiswa sebagai bagian dari persyaratan akhir pendidikan akademisnya. Jadi, skripsi adalah karya tulis ilmiah hasil penelitian mahasiswa sarjana strata satu (S1) yang membahas hasil penelitiannya sesuai kaidah penulisan dan aturan skripsi.

Skripsi adalah suatu karya tulis ilmiah yang merupakan pembuktian dari hasil kegiatan akhir berupa penelitian mahasiswa tingkat Sarjana dan merupakan:

- Uraian yang membahas suatu proses kegiatan kerja, membahas suatu masalah dalam bidang ilmu tertentu dengan menggunakan kaidah kaidah yang berlaku dalam bidang ilmu tersebut.
- Hasil kegiatan yang terencana dan terarah secara sistematis untuk memperoleh data dan informasi tentang suatu proses kegiatan kerja.
- Hasil kegiatan yang terencana, terarah secara sistematis, dan terkendali, untuk memperoleh data dan informasi tentang suatu masalah dalam bidang ilmu tertentu, dengan menggunakan metode ilmiah.

Kegiatan yang dilakukan dalam rangka penulisan SKRIPSI, dapat dilaksanakan melalui studi kepustakaan yang disertai dengan pengumpulan data primer dan data skunder. Data primer adalah data yang diperoleh mahasiswa di lapangan, baik melalui wawancara maupun hasil pengukuran langsung lainnya, data skunder adalah data yang diperoleh mahasiswa dengan memanfaatkan hasil pihak lain, seperti publikasi ilmiah, jurnal, majalah ilmiah.

Dengan demikian, kegiatan yang dilakuka bisa berupa:

- Studi kepustakaan, serta pengumpulan data *primer* dan *skunder*;
- Studi kepustakaan dan pengumpulan data *primer*.

- Studi kepustakaan dan pengumpulan data *skunder*.

2.2.4 World Wide Web

World Wide Web (WWW) adalah suatu program yang ditemukan oleh Tim Berners-Lee pada tahun 1991. Awalnya Berners-Lee hanya ingin menemukan cara untuk menyusun arsip-arsip risetnya. Untuk itu, beliau mengembangkan suatu sistem untuk keperluan pribadi. Sistem itu adalah program piranti lunak yang diberi nama *Enquire*. Dengan program itu, Berners-Lee berhasil menciptakan jaringan yang menyatukan berbagai arsip sehingga memudahkan pencarian informasi yang dibutuhkan. Ini lah yang kelak menjadi dasar dari sebuah perkembangan pesat yang dikenal sebagai WWW.

Pada tahun 1989 Berners-Lee membuat pengajuan untuk proyek pembuatan hypertext global, kemudian pada bulan Oktober 1990, “*Waring Wera Wanua*” sudah dapat dijalankan dalam lingkungan CERN (Pusat Penelitian Fisika Partikel Eropa) pada musim panas tahun 1991, WWW secara resmi digunakan secara luas pada jaringan Internet (Priyanto dan jauhari. 2020)

2.2.5 Metode SDLC

Software Development Life Cycle (SDLC) adalah proses pengembangan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan best practice atau cara-cara yang sudah teruji baik). Seperti halnya proses metamorfosis pada kupu-kupu, untuk menjadi kupu-kupu yang indah maka dibutuhkan beberapa tahap untuk dilalui, sama halnya dengan membuat perangkat lunak, memiliki daur tahapan yang dilalui agar menghasilkan perangkat lunak yang berkualitas.

Tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut:

- Inisiasi (*initiation*)

Tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

- Pengembangan konsep sistem (*system concept development*)

Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

- Perancangan (*planning*)

Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.

- Analisis kebutuhan (*requirements analysis*)

Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. membuat dokumen kebutuhan fungsional.

- Desain (*design*)

Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem fokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

- Pengembangan (*development*)

Mengkonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan; membuat basis data dan mempersiapkan prosedur kasus pengujian; mempersiapkan berkas atau *file* pengujian, pengkodean, pengompilasian, memperbaiki dan membersihkan program; peninjauan pengujian.

- Integrasi dan pengujian (*integration and test*)

Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasi pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user*. menghasilkan laporan analisis pengujian.

- Implementasi (*implementation*)

Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan pada *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

- Operasi dan pemeliharaan (*operations and maintenance*)

Mendeskrripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

- Disposisi (disposition)

Mendeskripsikan aktivitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

Analisis dan desain sering dikelompokkan sebagai proses sistem/rekayasa informasi karena pada tahapan inilah informasi mengenai kebutuhan perangkat lunak banyak dikumpulkan dan diintegrasikan. Ada beberapa model SDLC yang dapat digunakan diantaranya.

1. Model SDLC air terjun (*Waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*).
2. Model *Prototype* digunakan untuk menggali spesifikasi kebutuhan pelanggan secara lebih detail tetapi beresiko tinggi terhadap membengkaknya biaya dan waktu proyek.
3. Model *Rapid Application Development* (RAD) adalah model proses pengembangan perangkat lunak yang bersifat inkremental terutama untuk waktu pengerjaan yang pendek.
4. Model *iteratif* merupakan gabungan dari model *waterfall* dan model prototipe. Model ini cocok digunakan pengembang dengan *turnover* staf yang tinggi.
5. Model *spiral* menyediakan pengembangan dengan cara cepat dengan perangkat lunak yang memiliki versi yang terus bertambah fungsinya (*increment*).

Dari beberapa model SDLC diatas tentu semuanya memiliki kelemahan dan kelebihan pada setiap model SDLC. (Rosa A.S, M. Shalahuddin. (2019):26-28)

2.2.6 Sublime Text

Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi Phyton API. Terciptanya aplikasi ini terinspirasi dari aplikasi Vim, Aplikasi ini sangatlah fleksibel dan powerfull. Fungsionalitas dari aplikasi ini dapat di kembangkan dengan menggunakan *sublime-packages*. *Sublime Text* bukanlah aplikasi *open source* dan juga aplikasi yang dapat digunakan dan didapatkan secara gratis, akan

tetapi beberapa fitur pengembangan fungsionalitas (*packages*) dari aplikasi ini merupakan hasil dari temuan dan mendapat dukungan penuh dari komunitas serta memiliki lisensi aplikasi gratis.

Sublime Text mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur *syntax highlight* hampir di semua bahasa pemrograman yang didukung ataupun dikembangkan oleh komunitas seperti; C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML. Biasanya bagi bahasa pemrograman yang didukung ataupun belum terdukung secara default dapat lebih dimaksimalkan atau didukung dengan menggunakan *add-ons* yang bisa di *download* sesuai kebutuhan user. (Hasan, 2017).

2.2.7 Web Browser

Web browser secara umum adalah suatu perangkat lunak atau software yang digunakan untuk mencari informasi atau mengakses situs-situs yang ada di internet. Perangkat ini akan lebih memudahkan pengguna dalam mengakses data atau mencari referensi yang dibutuhkan. Ada berbagai macam perangkat web browser yang kini digunakan seperti Mozilla Firefox, Google Chrome, Opera, Safari, Internet Explorer dan lain sebagainya.

Setiap web browser yang digunakan tentunya memiliki kelebihan dan kekurangan, meskipun demikian beberapa browser seperti Mozilla dan Chrome banyak digunakan karena dinilai lebih cepat saat mengakses data. Terkadang *web browser* malah lebih sering diartikan sebagai mesin pencari seperti *Google* padahal sebenarnya web browser berbeda dengan mesin pencari atau search engine dimana *web browser* berdiri sendiri sementara *search engine* biasanya merupakan bagian dari suatu *web browser*.

Pengertian web browser menurut Wikipedia adalah sebuah aplikasi atau perangkat lunak yang digunakan mengambil, menyajikan, dan melintasi berbagai sumber informasi yang ada di dalam jaringan internet (*world wide web*). Sumber-sumber informasi yang dimaksud itu terdiri dari berbagai macam bentuk seperti

misalnya berupa halaman situs, gambar, video, infografis, konten, dan yang lainnya.

2.2.8 Web Server

Web Server adalah suatu program komputer yang mempunyai tanggung jawab atau tugas menerima permintaan HTTP dari komputer Klien, yang dikenal dengan nama *web browser* dan melayani mereka dengan menyediakan respon HTTP berupa konten data, biasanya berupa halaman web yang terdiri dari dokumen HTML dan objek terkait seperti gambar dan lain-lain.

Untuk dapat menjalankan PHP yang disertai dengan MySQL dapat digunakan dua jenis *web server* yaitu *Online mode* dan *Offline mode*. Pada *Online mode*, selain komputer harus mempersiapkan domain dan hosting serta koneksi internet yang memadai untuk mengelolanya sehingga harus keluar biaya ekstra terlebih dahulu. Sedangkan pada *Offline mode* yang anda persiapkan cukup komputer dan beberapa *software* untuk membuat *web server* lokal. Pada kedua cara kedua ini komputer PC akan dibuat menjadi *web server* lokal atau localhost sebelum benar-benar mengupload-nya ke web server Internet.

Ada beberapa jenis *software* untuk membangun *web server* lokal atau *localhost* yang *support* sistem Windows diantaranya adalah WampServer, Appserv, XAMPP, PHPTriad atau Vertrigo. (Madcoms, Andi. 2016: 5)

Web server dapat diartikan sebagai perangkat lunak yang dijalankan di sistem operasi pada komputer *server* maupun desktop, yang berfungsi untuk menerima permintaan (*request*) dalam bentuk protokol pada jaringan. Umumnya protokol yang digunakan adalah HTTP (*Hypertext Transfer Protocol*) dan HTTPS (*Hypertext Transfer Protocol Secure*). *Request* dari *client* tersebut kemudian dibalas (*reply*) dengan cara mengirimkan hasil permintaan tersebut melalui *web browser*. (Putu Agus Eka Pratama. 2015: 448)

2.2.9 XAMPP

XAMPP adalah sebuah paket perangkat lunak (*software*) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / MariaDB (sekarang), PHP, dan Perl. Sementara imbuhan huruf “X” yang terdapat pada awal kata berasal dari istilah *cross platform* sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti OS Linux, OS Windows, Mac OS, dan juga Solaris.

Sejarah mencatat, *software* XAMPP pertama kali dikembangkan oleh tim proyek bernama *Apache Friends* dan sampai saat ini sudah masuk dalam rilis versi 7.3.9 yang bisa didapatkan secara gratis dengan label GNU (*General Public License*).

XAMPP adalah aplikasi *Web Server* berbasis instan (siap saji) yang dapat digunakan baik di sistem operasi *Linux* maupun di sistem operasi *Windows*. XAMPP merupakan aplikasi *Open Source*, sehingga sumber kodenya terbuka dan memudahkan di dalam pengembangan dan kostumisasi sesuai kebutuhan. XAMPP tersedia untuk sistem operasi *windows* maupun *Linux*. (Putu Agus Eka Pratama. 2015: 449)

Sebetulnya, jika anda ingin supaya *web* dinamis anda (misal dalam PHP) bisa diakses secara lokal menggunakan *web server* lokal, maka anda harus menginstal PHP (untuk *script*), Apache (*web server*) dan MySQL (*database*) satu demi satu. Namun sekarang, ada beberapa komunitas programmer yang menyediakan solusi praktis dalam menginstal aplikasi-aplikasi tadi. Contohnya XAMPP.

XAMPP merupakan sebuah aplikasi open source yang didalamnya berisi kebutuhan kumpulan kebutuhan ketika akan membuat suatu aplikasi web yang diantaranya seperti yang disebutkan diatas. XAMPP sendiri mempunyai arti di setiap hurufnya diantaranya : (Hidayatullah dan Kawistara, 2017)

- X yang berarti *cross platform* karena XAMPP bisa dijalankan di Windows, Linux, Mac dcb
- A yang berarti Apache sebagai web server-nya
- M yang berarti MySQL sebagai Database Management System (DBMS)-nya

- PP yang berarti PHP dan *Perl* sebagai bahasa yang didukung. (Priyanto Hidayatullah, Jauhari Khairul Kawistara. 2017: 125)

2.2.10 *HyperText Markup Language (HTML)*

Hypertext Markup Language (HTML) adalah bahasa standar yang digunakan untuk menampilkan halaman *web*. Yang bisa dilakukan dengan HTML yaitu :

- Mengatur tampilan dari halaman web dan isinya.
- Membuat tabel dalam halaman *web*.
- Mempublikasikan halaman *web* secara *online*.
- Membuat form yang bisa digunakan untuk menangani registrasi dan transaksi via *web*.
- Menambahkan objek-objek seperti citra, audio, video, animasi, java applet, dalam halaman *web*.
- Menampilkan area gambar (kanvas) di *browser*. (Priyanto Hidayatullah, Jauhari Khairul Kawistara. 2017:15)

HTML adalah singkatan dari *Hypertext Markup Language*. HTML merupakan file teks yang ditulis menggunakan aturan-aturan kode tertentu untuk kemudian disajikan ke user melalui suatu aplikasi web browser. Setiap informasi yang tampil di web selalu dibuat menggunakan kode HTML. Oleh karena itu, dokumen HTML sering disebut juga sebagai web page (halaman web). Untuk membuat dokumen. Untuk membuat dokumen HTML, kita tidak tergantung pada aplikasi tertentu; karena dokumen HTML dapat dibuat menggunakan aplikasi Text Editor apapun, bisa Notepad (untuk lingkungan MS Windows). Emacs atau Vi Editor (untuk lingkungan Linux), dan sebagainya.

Kode HTML dari suatu halaman web dapat kita lihat dari aplikasi web browser yang kita gunakan, yaitu dengan cara memilih menu View/ Source (untuk Internet Explorer) dan View / Page Source (untuk Mozilla Firefox). (Budi Raharjo. 2018: 494)

2.2.11 Cascading Style Sheet (CSS)

CSS (*Cascading Style Sheet*) adalah salah satu bahasa desain *web* (*style sheet language*) yang mengontrol format tampilan sebuah halaman *web* yang ditulis dengan menggunakan penanda markup language. Biasanya CSS digunakan untuk mendesain sebuah halaman HTML dan XHTML, tetapi sekarang CSS bisa di aplikasikan untuk segala dokumen XML, termasuk SVG dan XUL bahkan ANDROID.

CSS dibuat untuk memisahkan konten utama dengan tampilan dokumen yang meliputi *layout*, warna dan *font*. Pemisahan ini dapat meningkatkan daya akses konten pada *web*, menyediakan lebih banyak *fleksibilitas* dan kontrol dalam spesifikasi dari sebuah karakteristik dari sebuah tampilan, memungkinkan untuk membagi halaman untuk sebuah *formatting* dan mengurangi kerumitan dalam penulisan kode dan struktur dari konten, contohnya teknik *tableless* pada desain *web*.

Jika kita menggunakan CSS kita bisa menyimpan format dan menggunakannya kapanpun dan dimanapun kita inginkan. Seperti terbentuknya kita dengan *Formatting and style* dalam membuat dokumen *office*, maka *style sheets* juga sangat penting dalam membuat halaman HTML yang dinamis.

Dengan *style sheet*, anda bisa membuat efek-efek tertentu untuk konten *web* anda. Cascading Style sheet (CSS) sudah didukung oleh hampir semua *web browser* karena CSS distandarkan oleh *World Wide Web Consortium* (W3C). (Priyanto Hidayatullah, Jauhari Khairul Kawistara. 2017:55)

2.2.12 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa script yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs web dinamis.

PHP sering digunakan untuk membangun sebuah CMS. PHP adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan *web*. Disebut bahasa pemrograman *server side* karena PHP diProses pada komputer *server*. Hal

ini berbeda dibandingkan dengan bahasa pemrograman *Client-side* seperti *JavaScript* yang di proses pada web browser (*client*). (Madcoms, Andi. 2016:2)

PhpMyAdmin adalah sebuah aplikasi open source yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan PhpMyAdmin, anda dapat membuat database, membuat tabel, meng-insert, menghapus dan meng-update data dengan GUI dan erasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual.

Karena berbasis web, maka PhpMyAdmin dapat dijalankan di banyak OS, selama dapat menjalankan web server dan MySQL. phpMyAdmin dapat di download secara gratis di <http://www.phpmyadmin.net>. Jika anda menggunakan paket software web server xampp, maka anda tidak perlu meng install phpMyAdmin secara terpisah. (Madcoms, Andi. 2016:12)

PHP (*Hypertext Preprocessor*) atau disingkat dengan PHP ini adalah suatu bahasa *scripting* khususnya digunakan untuk *web development*. Karena sifatnya yang *server side scripting*, maka untuk menjalankan PHP harus menggunakan *web server*. Mempunyai banyak kelebihan menjadi alasan utama kenapa PHP lebih dipilih sebagai basis umum dalam membuat sebuah *web*. (Priyanto Hidayatullah, Jauhari Khairul Kawistara. 2017:223)

a. Kelebihan PHP

PHP berbasis *server side scripting*. PHP sendiri dapat melakukan tugas-tugas yang dilakukan dengan mekanisme CGI (*Common Gateway Interface*) yaitu suatu mekanisme yang berjalan di *Web Server*, bertugas untuk melayani komunikasi dua arah antara *Web server* dan *Web browser*. seperti mengambil, mengumpulkan data dari *database*, meng-*generate* halaman dinamis, atau bahkan menerima dan mengirim *cookie*. (42) dan yang menjadi keutamaan PHP itu sendiri adalah PHP bisa digunakan di berbagai *operating system*, diantaranya Linux, Unix, Windows, Mac OsX, RISC OS, dan operating system lainnya. Di bawah ini adalah kelebihan PHP diantaranya

- PHP Berbasis *Server Side Scripting*
- *Command Line Scripting* pada PHP

- PHP dapat Membuat Aplikasi Desktop
- Digunakan untuk Berbagai Macam *Platform* OS
- Mendukung Berbagai Macam *Web Server*
- *Object Oriented Programming* atau *Procedural*
- *Output file* PHP pada XHTML, HTML, dan XML
- Mendukung Banyak RDBMS (*Database*)
- Mendukung Banyak Komunikasi
- Pengolahan Teks yang Sangat Baik

2.2.13 MySQL

Database Management System (DBMS) adalah aplikasi yang dipakai untuk mengelola basis data. DBMS biasanya menawarkan beberapa kemampuan yang terintegrasi seperti:

1. Membuat, menghapus menambahkan, dan memodifikasi basis data
2. Pada beberapa DBMS pengelolaannya berbasis windows (bentuk jendela-jendela) sehingga lebih mudah digunakan
3. Tidak semua orang bisa mengakses basis data yang ada sehingga memberikan keamanan bagi data
4. Kemampuan berkomunikasi dengan program aplikasi lain. Misalnya dimungkinkan untuk mengakses basis data MySQL menggunakan aplikasi yang dibuat menggunakan PHP.
5. Kemampuan pengaksesan melalui komunikasi antar komputer (*client server*)

MySQL adalah salah satu aplikasi DBMS yang sudah sangat banyak digunakan oleh para pemrograman aplikasi web. Contoh DBMS lainnya adalah: PostgreSQL (*freeware*), SQL Server, MS Access dari Microsoft, DB2 dari IBM, Oracle dan Oracle Corp, Dbase, FoxPro, dsb.

Kelebihan dari MySQL adalah gratis, handal, selalu di-update dan banyak forum yang memfasilitasi para pengguna jika memiliki kendala. MySQL juga

menjadi DBMS yang sering di bundling dengan *web server* sehingga proses instalasinya jadi lebih mudah.

MySQL menggunakan tipe data numerik standar ANSI. Berikut ini adalah tipe data numerik yang biasanya digunakan beserta penjelasannya. (Priyanto dan Jauhari.2020)

a. Tipe Data Dalam MySQL

1. Tipe Data Numerik

Tabel 2.1 Tipe Data Numerik

Tipe Data	Deskripsi
INT	Nilai integer yang bisa bertanda atau tidak. Jika bertanda, maka rentan yang diperoleh adalah -2147483647, sampai 2147483647, sedangkan jika tidak bertanda maka rentannya dari 0 sampai 4294967295
TINYINT	Nilai integer yang sangat kecil. Rentangnya -128-127 untuk yang bertanda dan 0-255 untuk yang tidak, bertanda.
SMALLINT	Nilai integer yang sangat kecil dengan rentang -31768 sampai 31767 untuk yang bertanda sedangkan untuk yang tidak bertanda dari 0-65535.
MEDIUMINT	Integer dengan ukuran sedang dengan rentang -8388608 sampai 8388607 atau 0 sampai 16777215
BIGINT	Integer dengan besar dengan rentang -9223372036854775808 sampai 9223372036854775808 atau 0 sampai 18446744073709551615.
FLOAT (MCD)	Bilangan pecahan dengan panjang (termasuk jumlah desimal) M dan

	jumlah desimal D. Presisi desimal nya bisa sampai 24 digit. Defaultnya Float (10,2). Bilangan float selalu bisa bertanda.
DOUBLE (M,D)	Adalah bilangan pecahan dengan presisi dua kali lipat. Panjang (termasuk jumlah desimal) M dan jumlah desimal D. Presisi desimalnya bisa sampai 53 digit. Defaultnya Double (16,4). Bilangan float selalu bisa bertanda. Sinonim dari DOUBLE adalah REAL
DECIMAL (M,D)	Adalah bilang pecahan dan harus definisikan M dan D-nya. Setiap desimal membutuhkan tempat 1 byte. Sinonim dari DECIMAL adalah NUMERIC.

2. Tipe Data Tanggal dan Waktu

Berikut ini adalah tipe data tanggal dan waktu di dalam MySQL

Tabel 2.2 Tipe Data Tanggal dan Waktu

Tipe Data	Deskripsi
DATE	Adalah tipe data tanggal dengan format YYYY-MM-DD, antara 1000-01-01 and 9999-12-31. Contoh: 17 Agustus 1945 akan disimpan sebagai 1945-08-17.
DATE TIME	Adalah komunikasi tanggal dan waktu dengan format YYYY-MM-

	DD HH: MM:SS dan rentang data antara 1000-01-01 00:00:00 sampai dengan 9999-12-31 23:59:59. Contoh: pukul 10:00 pagi pada tanggal 17 Agustus 1945 akan disimpan sebagai 1945-08-17 10:00:00
TIMESTAMP	Sebuah penanda waktu antara 1 Januari 1970 tengah malam dengan tahun 2037. Formatnya mirip dengan DATE TIME tetapi tanpa pembatas di antara angkanya. Contoh: pukul 10:00 pagi pada tanggal 17 Agustus 1945 akan disimpan sebagai 19450817100000
TIME	Menyimpan waktu dalam format HH:MM:SS. Contoh pukul 10:00 akan disimpan menjadi 10:00:00
YEAR(M)	Menyimpan data tahun dalam format 2 atau 4 digit, jika M diisi dengan nilai 2, maka rentang tahunnya dari 1970-2069 sedangkan jika M diisi dengan nilai 4 maka YEAR bisa bernilai 1901 sampai dengan 2155. Default nilai M adalah 4

2.2.14 Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

Kompleksitas pengembangan perangkat lunak secara logika sama dengan mengelola banyak kepala yang memiliki tingkat pemahaman dan pemikiran yang berbeda untuk membuat sebuah benda. Semakin banyak benda yang harus disatukan maka semakin sulit mengelolanya.

Pemodelan adalah gambaran dari realita yang simpel dan dituangkan dalam bentuk pemetaan dengan aturan tertentu. Pemodelan perangkat lunak memiliki beberapa abstrak, misalnya sebagai berikut:

- Petunjuk yang terfokus pada proses yang dimiliki oleh sistem
- Spesifikasi struktur secara abstrak dari sebuah sistem (belum detail)
- Spesifikasi lengkap dari sebuah sistem yang sudah final
- Spesifikasi umum atau khusus sistem.

Perangkat pemodelan adalah suatu model yang digunakan untuk menguraikan sistem menjadi bagian-bagian yang dapat diatur dan mengomunikasikan ciri konseptual dan fungsional kepada pengamat.

Peran perangkat pemodelan :

- Komunikasi

Perangkat pemodelan digunakan sebagai alat komunikasi antara pemakai dengan analis sistem maupun *developer* dalam pengembangan sistem.

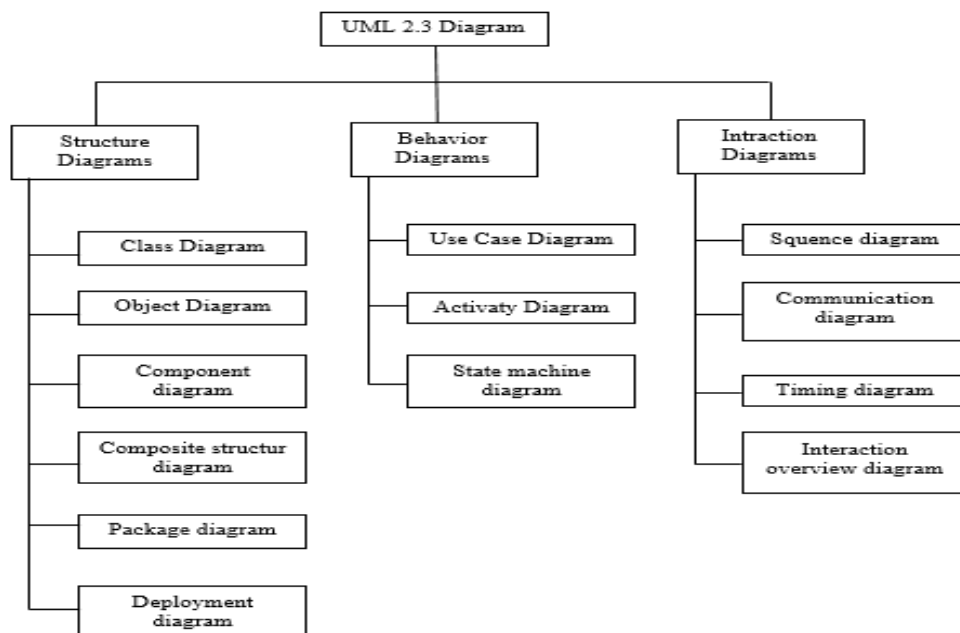
- Eksperimentasi

Pengembangan sistem yang bersifat “*trial and error*”

- Prediksi

Model meramalkan bagaimana suatu sistem akan bekerja.

Pada UML terdiri dari 13 macam diagram yang dikelompokkan pada 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar di bawah ini. (Rosa A. S, M. Shalahiddin. 2019:133)



Gambar 2.1 Diagram UML

a. Use Case Diagram

Use case atau diagram *use case* merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.





Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut *aktor* dan *use case*.



- Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

- Usec case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor. (Rosa A.S, M.Shalahuddin, 2019:155)

Berikut adalah simbol Simbol yang digunakan pada use case diagram bisa dilihat pada tabel 1.

Tabel 2.3 : Simbol Use Case Diagram

NO	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
3		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4		<i>Extend</i>	Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada

			suatu titik yang diberikan
5		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
6		<i>Include</i>	Menspesifikasikan bahwa use case sumber secara eksplisit.

b. Activity Diagram



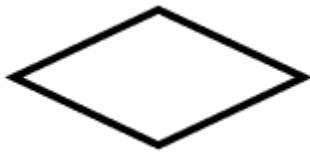

Diagram activity atau *activity* diagram merupakan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.


Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut ini:

- Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki rancangan antarmuka tampilan.
- Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
- Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas: (Rosa A.S, M.Shalahuddin, 2019:161)

Table 2.4 : Simbol Activity Diagram

NO	Simbol	Nama	Keterangan
1		Status Awal	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
2		Aktivitas	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
3		Percabangan	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4		Penggabungan	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

5		Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
---	---	--------------	---

c. Class Diagram

Diagram kelas atau *class diagram* merupakan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- Operasi atau metode adalah fungsi-fungsi dimiliki oleh suatu kelas.

Diagram kelas dibuat agar pembuat atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron. Banyak berbagai kasus, perancangan kelas yang dibuat tidak sesuai dengan kelas-kelas yang dibuat pada perangkat lunak, sehingga tidak ada gunanya lagi sebuah perancangan karena apa yang dirancang dan hasil jadinya tidak sesuai. (Rosa A.S, M. Shalahuddin. 2019:141)

2.2.15 Bootstrap

Bootstrap merupakan sebuah Framework CSS yang paling banyak diminati oleh para *developer website*. *Class-class* CSS dalam *bootstrap* sudah dibakukan sehingga pengerjaan sebuah *project* berbasis *web* menjadi semakin mudah dilakukan secara bersama-sama dengan sebuah tim. Kita dengan mudah dapat mendesain tampilan *website* yang *responsif* dengan menggunakan Bootstrap. Responsif maksudnya lebar halaman website akan disesuaikan secara otomatis berdasarkan perangkat yang digunakan untuk mengaksesnya, baik itu ketika diakses menggunakan PC, laptop, tablet, ataupun smartphone sehingga *website* akan menyesuaikan dengan lebar perangkat yang digunakan pengunjung.

Dalam mendesain layout, Bootstrap menyediakan fitur *grid* (12 *grid*). *Grid* ini ibarat kolom dalam sebuah tabel. Kita dapat membagi-bagi *layout* sebuah

halaman *website* menjadi beberapa bagian dengan mudah dan cepat. Selain kemampuan menghasilkan tampilan *website* yang *responsive*, *Bootstrap* juga menyediakan *Class-class* CSS yang sudah terintegrasi dengan *javascript* dan *jQuery*. Jadi, untuk merancang form, membuat *button* (tombol), *navigasi*, *dropdown* menu, model, *carousel*, ataupun *slider content*, dan lain sebagainya hanya dengan memanggil *class-class* yang sudah disediakan sehingga mempermudah kita untuk menciptakan *website* yang memiliki desain menarik (teknik), profesional, dan ringan ketika diakses.

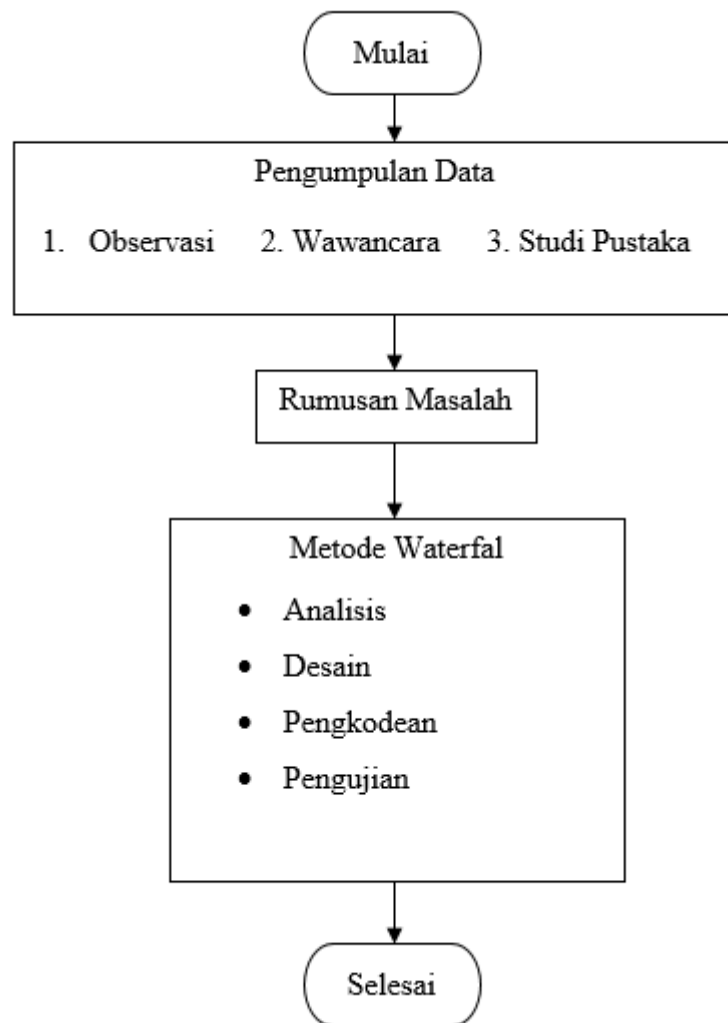
Salah satu sosial media yang menggunakan framework CSS *Bootstrap* adalah twitter. *Bootstrap* pertama kali dibuat oleh programmer Twitter, yaitu Mark Otto dan Jacob Thornton pada tahun 2011. Pada saat itu, antar sesama *programmer* Twitter menggunakan berbagai macam *tools* dan *library* secara terpisah dalam *men-developer project* sehingga tidak ada standarisasi dan sulit dalam pengelolaan. Hingga pada akhirnya , Otto dan Thornton membuat *framework* yang dapat digunakan secara bersama-sama dalam internet Twitter. Setelah keluar dari Twitter, mereka menjadikan *Bootstrap* menjadi *project open source* sehingga dapat digunakan oleh banyak *programmer* lain, hingga pada tahun 2014 *Bootstrap* menjadi proyek No. 1 di GitHub.

Cara menggunakan bootstrap. File CSS *Bootstrap* harus kita panggil terlebih dahulu di dalam dokumen HTML untuk dapat digunakan *Bootstrap* dan diletakan sebelum tag ‘head tutup’ ataupun </head> dan file javascript *jQuery* kita tempatkan sebelum tag ‘body tutup </body>’. Contoh penggunaan *Bootstrap* seperti kode HTML berikut ini. Tag </body>. Jadi, setiap menggunakan *Bootstrap* ketiga file tersebut, yaitu file CSS, *jQuery*, dan javascript *Bootstrap* harus kita sertakan dalam dokumen web.(Roberto Kaban. 2018)

BAB III

METODOLOGI PENELITIAN

3.1 Kerangka Pikir



Gambar 3.1 Kerangka Pikir

3.2 Deskripsi

2.2.1. Pengumpulan Data

- Observasi

Pada tahap ini penulis mengamati secara langsung permasalahan yang muncul di Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung dalam melakukan pengumpulan persyaratan pengajuan proposal skripsi. Setelah mengamati hasil dari pengamatan pengumpulan persyaratan pengajuan proposal skripsi masih dilakukan secara konvensional.

- Wawancara

Pada tahap ini wawancara dilakukan langsung kepada bagian Tata Usaha dan Mahasiswa Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung menghasilkan beberapa poin permasalahan diantaranya :

1. Bagaimana Mahasiswa agar lebih mudah dalam mengumpulkan persyaratan skripsi
2. Bagaimana prosedur yang tadinya panjang menjadi pendek
3. Bagaimana meng-implementasikan aplikasi agar dapat di akses oleh Mahasiswa dan Tata Usaha

- Studi Pustaka

Studi Pustaka dilakukan mengamati penelitian sebelumnya seperti jurnal-jurnal dan teori-teori yang terdapat di buku dan artikel lainnya yang digunakan sebagai referensi seperti yang tercantum pada BAB 2.

2.2.2. Rumusan Masalah

Rumusan masalah adalah bagian penting dalam makalah atau laporan. Dengan adanya rumusan masalah dalam sebuah makalah atau laporan maka akan menjawab permasalahan.

Pada tahap ini penulis merumuskan permasalahan yang akan di teliti di Program Studi Teknik Informatika Fakultas Teknologi Informasi dalam seleksi kelayakan pengajuan proposal skripsi karena masih menggunakan cara konvensional atau manual tentu itu tidak efektif karena akan menghabiskan waktu serta membantu tugas tata usaha agar lebih mudah begitu pula membantu mahasiswa dalam pengumpulan persyaratan skripsi , serta mengurangi ketidak efektifan pengumpulan persyaratan skripsi, serta mempermudah penyeleksian mahasiswa yang layak skripsi dilihat dari pengumpulan persyaratan skripsi.

2.2.3. Metode Waterfall

Pada tahap ini penulis menggunakan metode waterfall dalam membangun aplikasi yang akan dibuat dalam metode waterfall ini dibagi menjadi beberapa bagian diantaranya:

1. Analisis

Pada tahap ini analisis kebutuhan pada proses pengembangan sistem semua dokumen yang di butuhkan dalam pengumpulan persyaratan skripsi dikumpulkan serta di lakukan pendokumentasian serta melakukan prosedur konvensional. Analisis dilakukan dari mulai tahap yang awalnya dilakukan secara panjang serta merepotkan mahasiswa dalam pengumpulan persyaratan skripsi dan merepotkan bagian tata usaha, penulis menganalisis sistem yang sedang berjalan tersebut dan melakukan pengembangan atau pembangunan aplikasi agar melakukan pengumpulan persyaratan skripsi menjadi lebih singkat dan efisien.

2. Desain

Pada tahap ini design atau desain dilakukan untuk menggambarkan aplikasi yang akan dibangun, melalui Desain sistem gambaran alur kerja sistem dan tampilan (user Interface). Proses desain tersebut

digambarkan melalui mockup aplikasi sebagai rancangan aplikasi yang akan dibangun

3. Pengkodean

Setelah melakukan tahap analisis sistem yang sedang berjalan dan desain aplikasi sebagai gambaran aplikasi yang akan dibuat lalu penulis melakukan pengkodean agar sesuai dengan desain yang telah dirancang pada tahap sebelumnya, pada tahap ini penulis melakukan pengkodean mulai dari sistem yang sesuai dengan tampilan mockup aplikasi atau desain.

4. Pengujian

Pada tahap ini penulis melakukan pengujian setelah analisis, desain dan pengkodean lalu penulis melakukan pengujian apakah sistem pengkodean yang telah dilakukan sesuai dengan apa yang diharapkan dan dibutuhkan untuk membantu mahasiswa dalam melakukan pengumpulan persyaratan proposal skripsi.

BAB IV

ANALISIS DAN PERANCANGAN

4.1 Analisis

Analisis dilakukan sebagai langkah awal penelitian untuk mengetahui kebutuhan-kebutuhan yang diperlukan pengguna. Dalam analisis ini penyusun mempunyai instrumen penelitian dan melakukan beberapa tahapan yaitu, analisis sistem, analisis kebutuhan, *user interface*, fitur-fitur dan hasil analisis.

4.1.1 Instrumen Penelitian

Instrumen penelitian pada penelitian ini terdiri dari perangkat lunak, perangkat keras dan observasi pada studi pustaka.

A. Perangkat Lunak

Penyusun menggunakan beberapa perangkat lunak pada penelitian kali ini, yaitu :

1. Menggunakan Microsoft Windows 10 Home Single Language 64-bit
2. Menggunakan XAMPP sebagai web server yang berdiri sendiri (localhost)
3. *Code editor* menggunakan Sublime Text versi 3
4. Google Chrome untuk melihat hasil *compile* pembuatan web

B. Perangkat Keras

Penyusun menggunakan perangkat keras dengan spesifikasi berikut:

Tabel 4.1 Perangkat Keras Laptop

Deskripsi	Spesifikasi
Tipe Laptop	ASUSTek COMPUTER INC

Prosesor	Intel(R) Core (TM) i3-5005U CPU 2 GHz
RAM	4096 MB
HDD	500GB

C. Server

Penyusun menggunakan server hosting pada penelitian kali ini, dengan spesifikasi berikut :

Table 4.2 Server Hosting

Deskripsi	Spesifikasi
Core	1
RAM	512
Memory	2 GB
OS	Linux
PHP Version	7.4.6

D. Observasi

Observasi yang dilakukan oleh penyusun dalam penelitian ini yaitu observasi pada penelitian sebelumnya yang membahas topik serupa dengan penelitian ini yang mencakup tentang aplikasi Persyaratan dan pendaftaran.

4.1.2 Analisis Sistem

Analisis sistem yang sedang berjalan yaitu sistem pengumpulan persyaratan skripsi yang dilakukan secara manual oleh mahasiswa dan tata usaha dalam melakukan pendataan, pada analisis sistem yang sedang berjalan ini mahasiswa mengumpulkan persyaratan proposal skripsi diantaranya. Lembar penyerahan Laporan Kerja Praktek semester Ganjil, Lembar KRS skripsi, Sertifikat Toefl, Sertifikat dari BNSP, Cetak KHS Semester 1-7, Bukti Pembayaran Tahap 1 Skripsi, Proposal Skripsi, secara

manual, tentu itu tidak efisien dan memerlukan waktu yang cukup lama karena, dengan mahasiswa yang harus datang ke kampus dalam pengumpulan persyaratan skripsi, dan merepotkan tata usaha dalam melakukan pendataan persyaratan proposal skripsi karena mahasiswa yang cukup banyak.

Dari permasalahan diatas penulis melakukan analisis sistem untuk membantu dan memudahkan mahasiswa dalam pengumpulan persyaratan skripsi dan memudahkan bagian tata usaha dalam pendataan dengan proses waktu yang singkat agar tidak membutuhkan waktu cukup lama. Dalam perancangan aplikasi seleksi pengumpulan persyaratan skripsi. Di butuhkan kebutuhan sistem yang meliputi;

a. *Input*

User membuka aplikasi, selanjutnya user memasukan user name serta password dan setelah itu user melakukan pengumpulan persyaratan skripsi tahap pertama, serta user dapat mengirim pesan ke pada admin jika memiliki kendala dalam persyaratan skripsi.

b. *Proses*

User menunggu verifikasi admin sebelum melakukan pengumpulan persyaratan tahap selanjutnya. Serta Proses penambahan Mahasiswa maupun proses Kirim Pesan.

c. *Output*

Menampilkan hasil mahasiswa yang telah selesai melengkapi persyaratan pertama serta kedua serta menampilkan hasil mahasiswa yang baru mengisi persyaratan pertama dan mahasiswa yang belum melakukan pengumpulan persyaratan sama sekali.

4.1.3 Analisis Kebutuhan

Pada tahap ini yaitu menyiapkan kebutuhan-kebutuhan dari semua elemen sistem perangkat lunak yang akan di bangun. Pada tahap ini

dibentuk kebutuhan perangkat lunak dan fungsi perangkat lunak yang dibutuhkan.

1. Kebutuhan Antar Muka

Kebutuhan antar muka pada pembuatan aplikasi ini sebagai berikut :

- a) Aplikasi harus mampu menampilkan hasil data yang telah di input
- b) Aplikasi harus mampu melakukan interaksi dengan admin
- c) Aplikasi dapat menampilkan hasil dari mahasiswa yang belum melakukan pengumpulan persayratan
- d) Aplikasi dapat menampilkan hasil dari mahasiswa yang sudah melakukan pengumpulan persyaratn skripsi tahap pertama

2. Kebutuhan Data

Data yang diolah pada aplikasi ini antara lain :

- a) Data mahasiswa yaitu NIM
- b) Data persyaratan skripsi

3. Kebutuhan Fungsional

Penjelasan secara rinci dari setiap fungsi pada aplikasi. Fungsi-fungsi yang dimiliki aplikasi adalah :

1. User Mahasiswa

- a) Saat user melakukan pengumpulan peryaratan tahap pertama lalu setelah itu user menunggu verifikasi admin
- b) Setelah admin melakukan tahap verifikasi dengan memberikan centang biru maka mahasiswa bisa melakukan pengumpulan persyaratan tahap kedua
- c) Jika mahasiswa dalam pengumpulan persyaratan skripsi ada yang keliru maka mahasiswa bisa melakukan edit data
- d) Ketika user memiliki kendala maka user bisa melakukan verifikasi kepada admin dengan mengirim pesan.

4. User Admin

- a) Admin melakukan verifikasi dengan cara melakukan update status serta memeriksa berkas yang di kumpulkan oleh mahasiswa
- b) Admin memberikan informasi dengan mengupdate informasi akhir pengumpulan persyaratan skripsi
- c) Admin dapat menambahkan mahasiswa serta melakukan komunikasi melalui pesan
- d) Admin dapat melihat rekap mahasiswa yang telah melakukan pengumpulan persyaratan tahap pertama serta melihat mahasiswa yang belum mengumpulkan persyaratan tahap pertama dan admin dapat mencetak hasil dari rekap mahasiswa tersebut

4.1.4 Hasil Analisis

Hasil analisis yang didapat dalam penelitian ini yaitu untuk memudahkan mahasiswa dalam melakukan pengumpulan persyaratan skripsi serta memudahkan admin dalam melihat dan menentukan mahasiswa layak skripsi di lihat dari pengumpulan persyaratan skripsi, maka di buat lah aplikasi

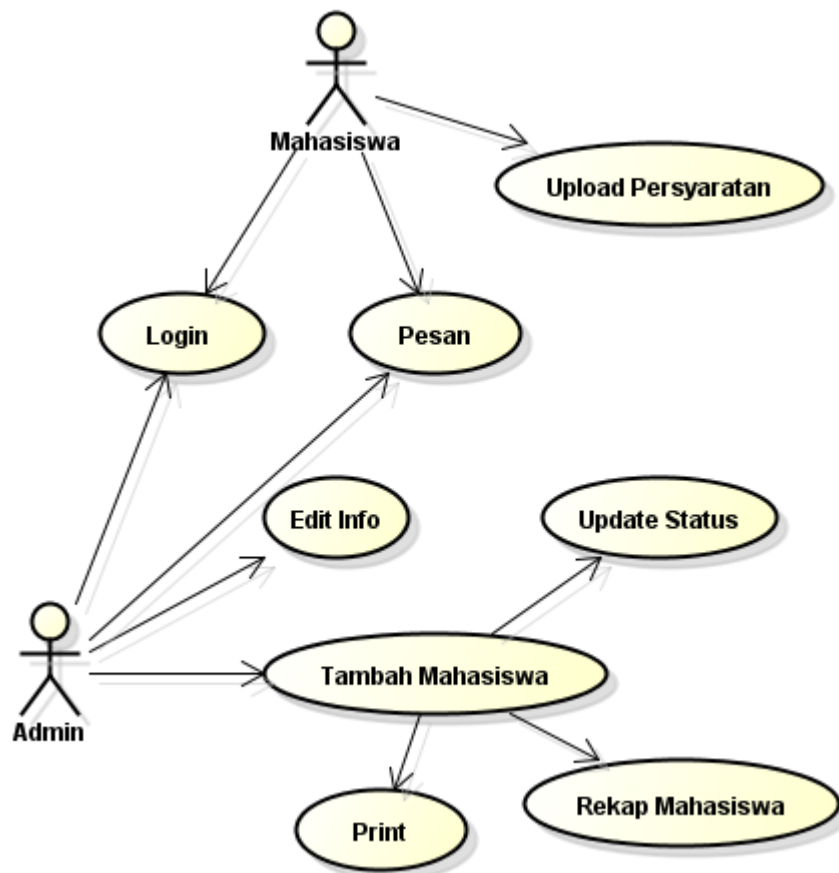
4.2 Perancangan

Sebelum masuk dalam pembuatan aplikasi dibutuhkan perancangan yaitu merancang *software* dalam bentuk UML yang terdiri dari *use case*, *activity diagram*, dan *class diagram* serta desain keseluruhan aplikasi

4.2.1 Diagram UML

1. Use Case Diagram

Usecase diagram menggambarkan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat untuk mengetahui fungsi apa saja yang ada didalam sistem dan siapa yang menggunakan fungsi tersebut. Berikut adalah *usecase* pada aplikasi pengumpulan persyaratan skripsi.



Gambar 4.1 Use Case Diagram

Gambar di atas adalah gambaran gambaran sistem yang digambarkan pada usecase diagram yang akan dibuat, pada gambar diatas kita bisa lihat bahawa mahasiswa hanya dapat mengakses beberapa menu aplikasi yang akan dirancang.

a. Skenario *Use Case*

Skenario *uses case* ini dibuat untuk menunjukkan scenario utama dari *use case* yang telah dibuat pada *use case* diagram. Pada skenario *use case* dijelaskan bagaimana urutan fungsionalitas berlangsung dari kondisi awal sampai kondisi akhir yang menunjukkan hasil akhir dari setiap *use case*.

Berikut adalah deskripsi pendefinisian tata usaha dan mahasiswa pada perancangan aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis WEB di Program Studi Teknik Informatika Universitas Bale Bandung:

Tabel 4.3 Deskripsi Tata Usaha dan Mahasiswa

No	Aktor	Deskripsi
1	Tata Usaha	Tata Usaha Bertugas Untuk mengelola data Mahasiswa yaitu menambah, memberikan perifikasi, edit informasi, menghapus mahasiswa, lihat berkas.
2	Mahasiswa	Mahasiswa atau pengguna aplikasi setelah <i>login</i> dapat mengumpulkan persyaratan tahap pertama, dan setelah tata usaha melakukan perifikasi lalu dapat melakukan pengumpulan persyaratan tahap kedua, liat berkas , dan edit berkas.

Tabel 4.4 *Use case* Tata Usaha

No	Use Case	Deskripsi
1	Login/Logout	Merupakan proses untuk melakukan <i>login</i> (masuk) dan <i>logout</i> (keluar)
2	Tambah mahasiswa	Merupakan menambah mahasiswa, <i>update</i> status mahasiswa, Rekap mahasiswa, <i>Print</i> dan menghapus mahasiswa.
3	Edit Informasi	Merupakan menambahkan informasi tentang <i>deatline</i> pengumpulan persyaratan
4	Lihat Persyaratan	Merupakan melihat persyaratan yang telah di kupulkan
5	Setting	Merupakan merubah password
6	Pesan	Merupakan mengirim pesan terhadap mahasiswa.

Tabel 4.5 *Use Case* Mahasiswa

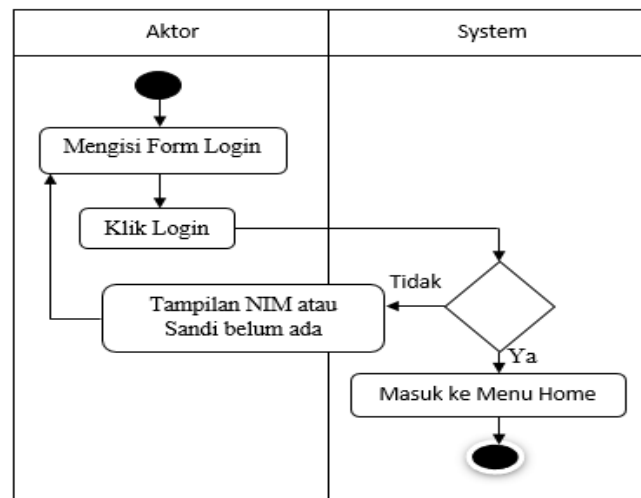
No	Use Case	Deskripsi
1	Login/Logout	Merupakan proses untuk melakukan Login (masuk) dan Logout (keluar)
2	Upload Persyaratan dan kedua	Merupakan upload persyaratan tahap pertama yang terdiri dari KRS, bukti pembayaran, Toefl, Sertifikasi, Proposal Skripsi, Bukti Telah menyelesaikan Kerja Praktek.
3	Edit Profil	Disini mhaasiswa hanya dpat mengupload atau edit profil poto sajah.
4	Pesan	Merupakan menu yang dapat di gunakan dalam melakukan kontak dengan Tata Usaha maupu Mahasiswa lain.

2. Activity Diagram

Activity Diagram menggambarkan urutan aktifitas proses pada sebuah sistem. Berikut adalah *activity* diagram mahasiswa dan admin pada Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis WEB:

1) Diagram Activity Mahasiswa

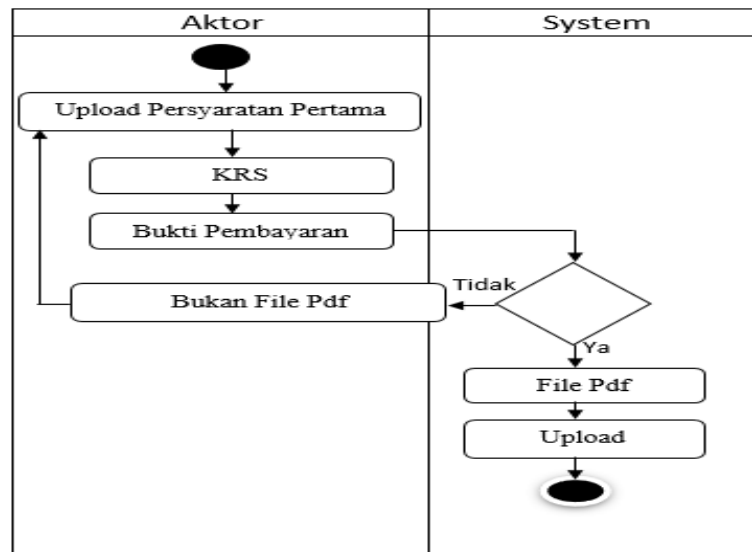
a. Diagram *Activity Login* Mahasiswa



Gambar 4.2 *Activity Login* Mahasiswa

Gambar di atas adalah gambar *login* mahasiswa yang di modelkan dengan diagram *activity*, aktor atau mahasiswa mengisi form data pribadi dan kemudian aktor akan masuk ke menu utama atau home jika sandi benar dan akan mengulangi mengisi form tersebut jika kata sandi salah.

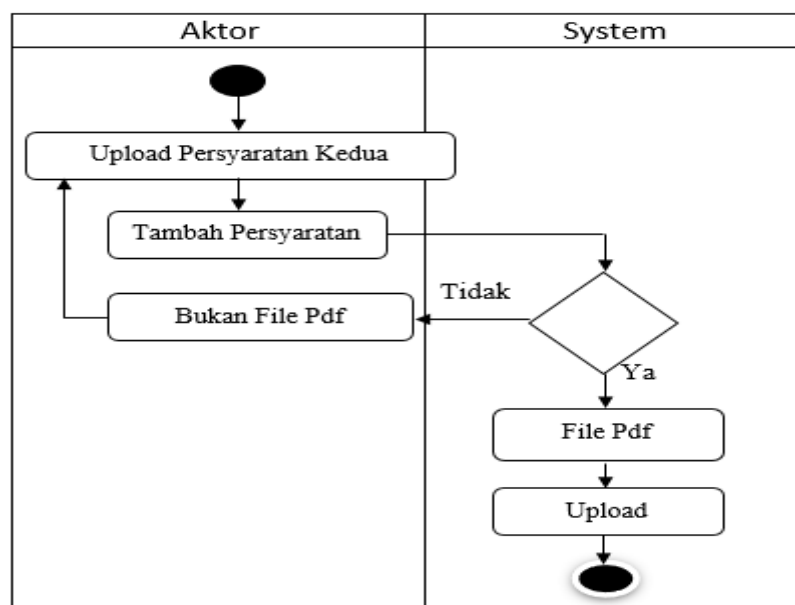
b. Diagram Activity Pengumpulan Persyaratan Pertama



Gambar 4.3 Activity Pengumpulan Persyaratan Pertama

Gambar di atas adalah gambar input persyaratan Skripsi tahap Pertama yang di modelkan dengan model diagram *Activity*, dari gambar di atas aktor menginput *file* dengan *format file* PDF dan jika *file* tersebut tidak berbentuk PDF maka aktor tidak dapat *mengupload file* tersebut jika *file* yang di *upload* sudah berbentuk PDF maka aktor mengklik tombol *Upload*.

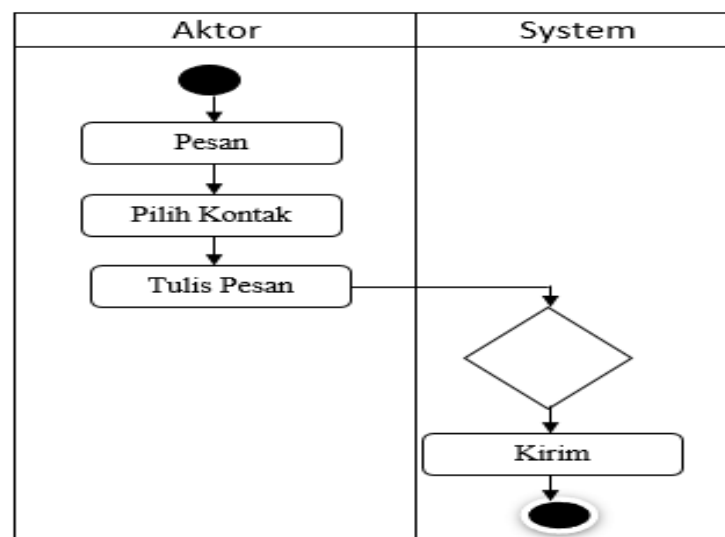
c. Diagram Activity Pengumpulan Persyaratan Kedua



Gambar. 4.4 Activity Pengumpulan Persyaratan kedua

Gambar di atas adalah gambar input persyaratan Skripsi tahap kedua yang di modelkan dengan model diagram Activity, dari gambar di atas aktor menginput *file* dengan format *file* PDF dan jika *file* tersebut tidak berbentuk Pdf maka aktor tidak dapat mengupload *file* tersebut jika file yang di upload sudah berbentuk Pdf maka aktor mengklik tombol *Upload*, dalam pengumpulan persyaratan skripsi kedua ini bisa dilakukan setelah admin melakukan verifikasi atau update status terhadap mahasiswa baru lah form seperti gambar di atas akan muncul dan mahasiswa melakukan pengumpulan skripsi tahap kedua.

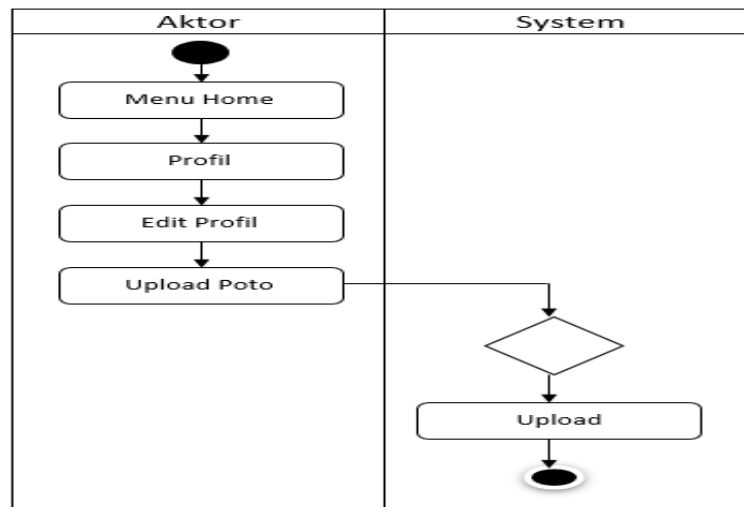
d. Diagram Activity Pesan



Gambar 4.5 Activity Pesan

Gambar Pesan diatas yang dimodelkan dengan diagram *activity*, diagram pesan yaitu diagram yang bisa dilakukan aktor saat terjadi kendala dalam pengumpulan persyaratan skripsi sehingga aktor dapat berkomunikasi dengan admin maupun dengan mahasiswa sekelasnya.

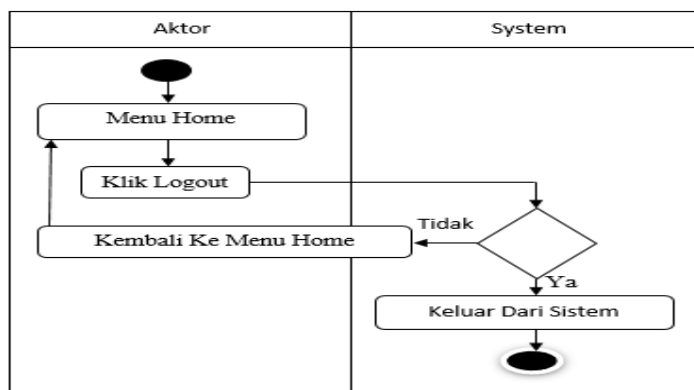
e. Edit Profil



Gambar 4.6 Activity Edit Profil

Gambar Pesan diatas yang dimodelkan dengan diagram activity, diagram Edit Profil yaitu diagram yang bisa dilakukan aktor jika ingin melakukan edit profil atau mengganti tampilan profil dengan poto aktor tersebut, namun disini aktor hanya dapat melakuakn edit Poto profil sajah tidak bisa melakukan edit Nama serta NIM.

f. Diagram Activity Logout



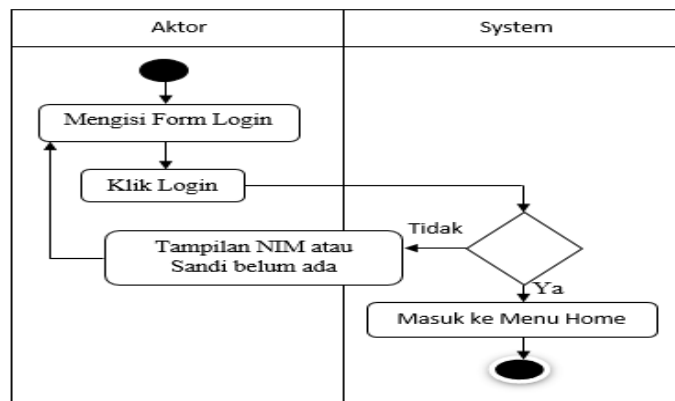
Gambar 4.7 Activity Logout Mahasiswa

Gambar di atas adalah model diagram *logout* yang di modelkan dengan model diagram activity. Dari gambar diatas jika aktor akan mengklik tombol logout maka aktor akan keluar dari aplikasi atau

kembali ke tampilan awal yaitu tampilan input user name dan password.

2) Diagram Activity Admin

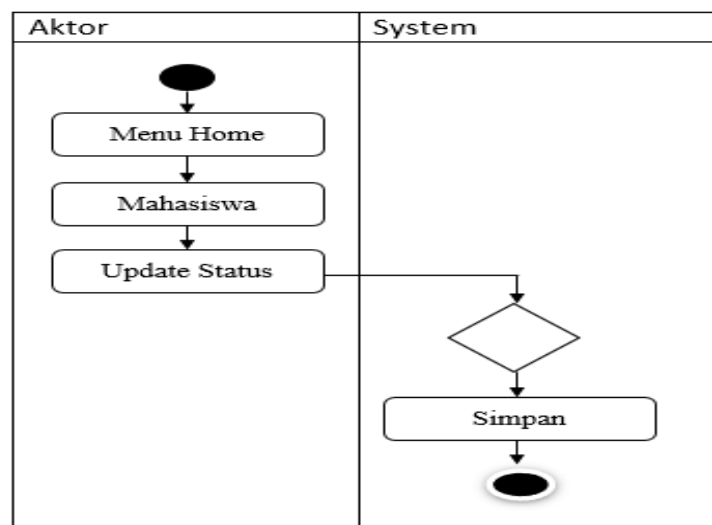
a. Diagram Activity Login Admin



Gambar 4.8 Activity Login Admin

Gambar di atas adalah gambar login mahasiswa yang di modelkan dengan diagram activity, aktor atau mahasiswa mengisi form data pribadi dan kemudian aktor akan masuk ke menu utama atau home jika sandi benar dan akan mengulangi mengisi form tersebut jika kata sandi salah.

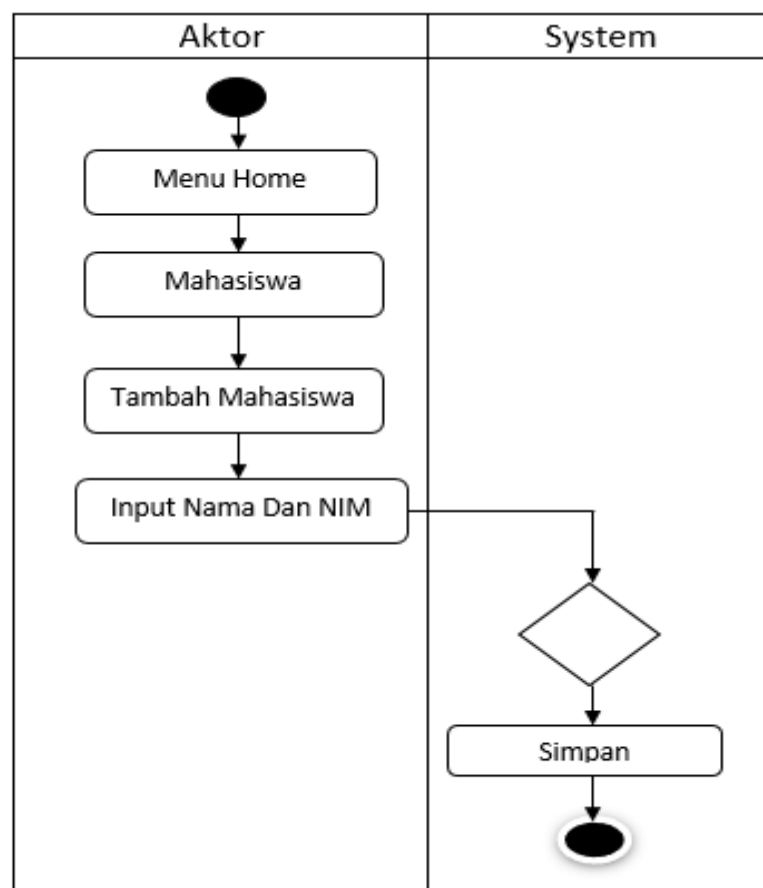
b. Diagram Activity Update Status



Gambar 4.9 Activity Update Status Mahasiswa

Gambar di atas adalah gambar *Update* status yang dilakukan admin terhadap mahasiswa yang di modelkan dengan diagram *activity*, aktor mengklik menu Mahasiswa setelah itu admin melakukan *update* status terhadap mahasiswa setelah admin melakukan *update status* setelah itu admin mengklik tombol simpan sehingga status mahasiswa tersebut telah berubah dan dapat melakukan pengumpulan persyaratan selanjutnya.

c. Diagram Activity Tambah Mahasiswa

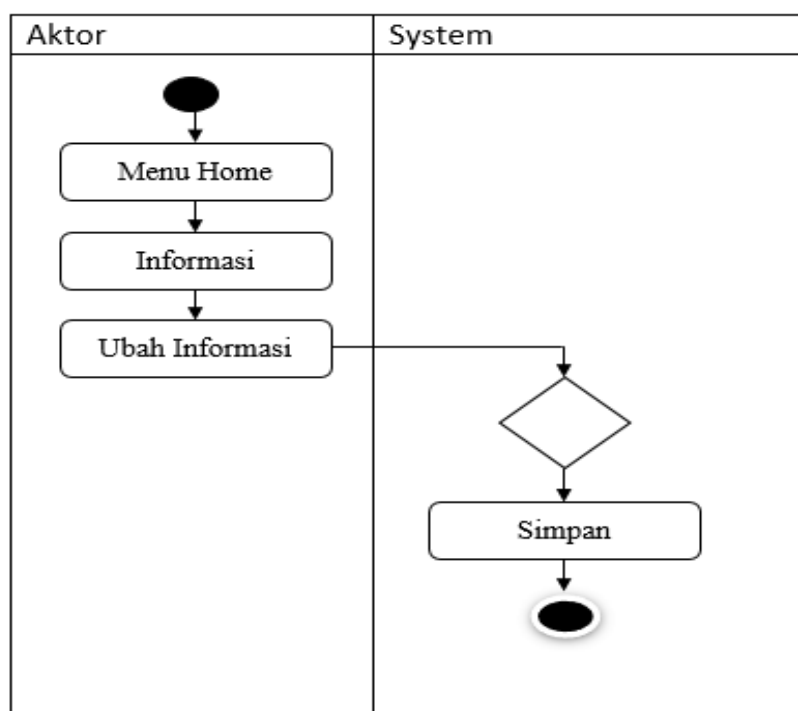


Gambar 4.10 Activity Tambah Mahasiswa

Gambar di atas adalah gambar tambah mahasiswa yang akan dilakukan oleh admin yang dimodelkan dengan diagram *activity*, diagram tambah mahasiswa yaitu diagram yang akan dilakukan oleh admin dalam menambahkan mahasiswa, admin akan memilih menu

mahasiswa setelah itu admin mengklik tambah mahasiswa setelah admin mengklik tombol tambah mahasiswa lalu admin akan diminta untuk melakukan pengisian data mahasiswa tersebut di antaranya NIM dan Nama Mahasiswa yang akan admin tambahkan setelah data tersebut telah di isi admin lalu mengklik tombol simpan maka mahasiswa tersebut telah di tambahkan.

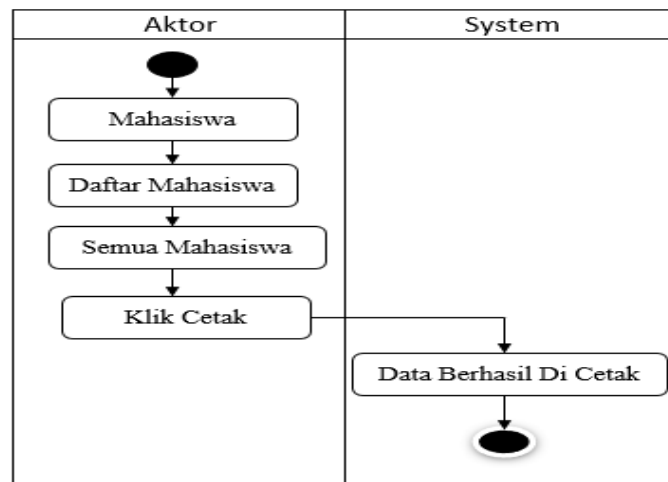
d. Diagram Activity Edit Informasi



Gambar 4.11 *Activity Edit Informasi*

Gambar di atas adalah gambar menu informasi yang yang di modelkan dalam dengan diagram *activity*, diagram informasi di atas dapat diberikan oleh *admin* terkait informasi mengenai batas waktu pengumpulan persyaratan skripsi ataupun informasi lainnya, admin akan memilih menu informasi yang ada dalam menu utama tersebut setelah itu admin akan di alihkan ke tampilah menu informasi berupa input informasi yang akan di berikan oleh admin terhadap semyua mahasiswa setelah admin melakukan input informasi lalu *admin* mengklik tombol simpan maka informasi yang yang di berikan dapat di tampilkan di halaman menu utama setiap mahasiswa.

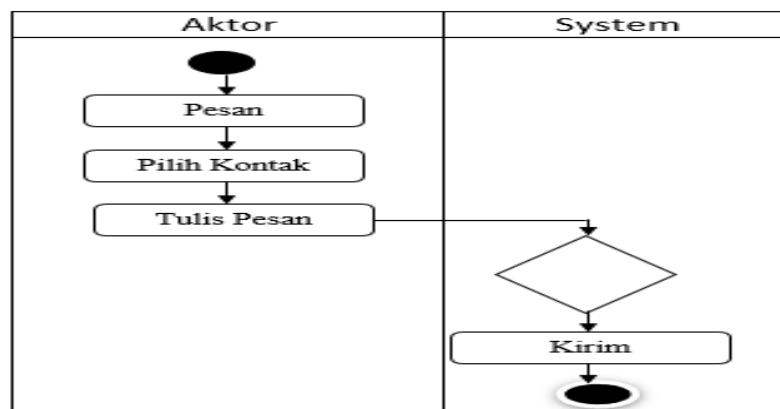
e. Diagram Activity Cetak Rekap Mahasiswa



Gambar 4.12 Activity Cetak Rekap Mahasiswa

Gambar di atas adalah gambar cetak mahasiswa yang modelkan dalam dengan diagram *activity*, diagram cetak mahasiswa yaitu diagram hasil dari diagram input persyaratan mahasiswa yang berisi data semua mahasiswa yang telah melakukan pengumpulan persyaratan dan mahasiswa yang belum melakukan pengumpulan persyaratan skripsi sehingga data tersebut bisa di *print*.

f. Diagram Activity Pesan

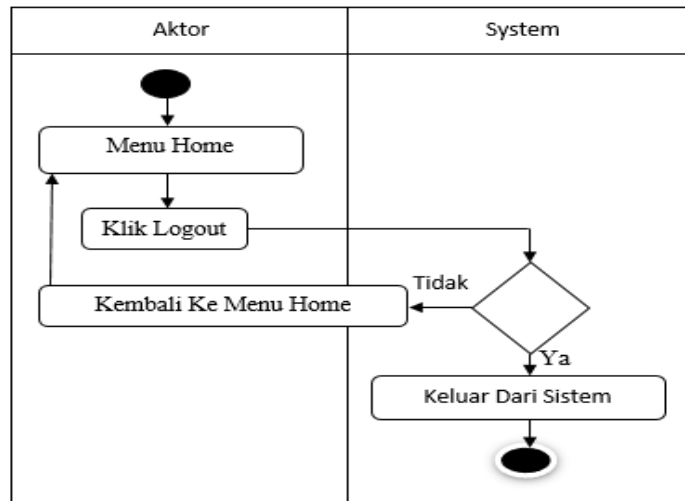


Gambar 4.13 Activity Pesan Admin

Gambar Pesan diatas yang dimodelkan dengan diagram *activity*, diagram pesan yaitu menu pesan yang admin dapat lakuka dalam

menerima pesan dari mahasiswa dan juga bias melakukan komunikasi dengan mahasiswa melalui pesan ini.

g. Diagram *Activity Logout*



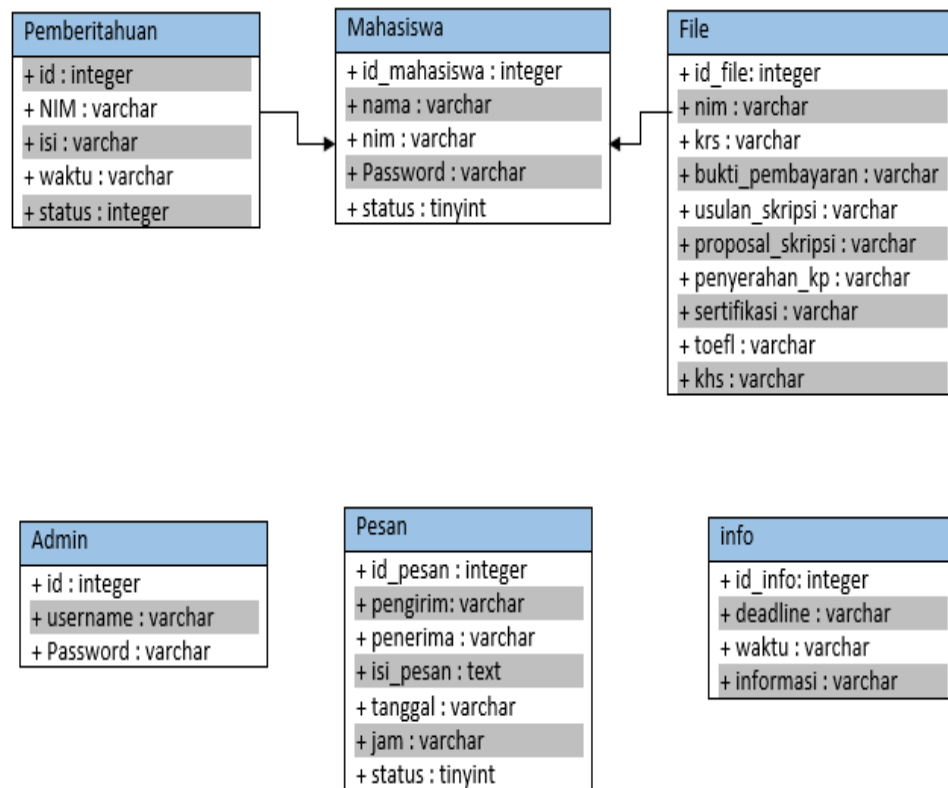
Gambar 4.14 Activity *Logout Admin*

Gambar di atas adalah model diagram *logout* yang di modelkan dengan model diagram *activity*. Dari gambar diatas jika aktor akan mengklik tombol *logout* maka aktor akan keluar dari aplikasi atau kembali ke tampilan awal yaitu tampilan *input user name* dan *password*.

3. Class Diagram

Class diagram adalah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Pada aplikasi Kelayakan Pengajuan Proposal Skripsi Berbasis Web terdapat 7 elemen penting, yaitu *Info*, *file*, *admin*, *file upload*, *pesan*, dan mahasiswa.

Berikut adalah *class* diagram pada aplikasi seleksi kelayakan pengajuan proposal skripsi berbasis WEB



Gambar 4.15 Class Diagram

a. Tabel Admin

Tabel 4.6 Tabel Admin

Name field	Type	Lenght	Keterangan
Id	Int	11	Primary Key. Auto Increment
Username	Varchar	10	Menampung username
Password	Varchar	126	Menampung password

b. Tabel File_Upload

Tabel 4.7 Tabel File_Upload

Name Field	Type	Lenght	Keterangan
Id_file_upload	Int	11	Primary Key. Auto Increment
Name_file	Varchar	50	Menampung nama file

c. Tabel File

Tabel 4.8 Tabel File

Name Field	Type	lenght	Keterangan
Id_file	Int	11	Primary Key. Auto Increment
Nim	Varchar	10	Menampung nim
Krs	Varchar	50	Menampung file krs
Bukti_Pembayaran	Varchar	50	Menampung file bukti pembayaran
Usulan_skripsi	Varchar	50	Menampung file usulan skripsi
Proposal_skripsi	Varchar	50	Manampung file proposal skripsi
Penyerahan_kp	Varchar	50	Menampung file penyerahan KP
Sertifikasi	Varchar	50	Menampung file sertifikasi
Toefl	Varchar	50	Menampung file toefl
Khs	Varchar	50	Menampung khs
Status	Int	1	Menampung status

d. Tabel Mahasiswa

Tabel 4.9 Tabel Mahasiswa

Name Field	Type	Lenght	Keterangan
Id_mahasiswa	Int	11	Primary Key. Auto Increment
Nama	Varchar	256	Menampung nama
Nim	Varchar	10	Menampung nim
Password	Varchar	256	Menampung password
Foto	Varchar	128	Menampung foto
Status	Tinyint	1	Menampung status

e. Tabel Pemberitahuan

Tabel 4.10 Tabel Pemberitahuan

Name Field	Type	Lenght	Keterangan
Id	Int	11	Primary Key. Auto Increment
Isi	Varchar	256	Menampung isi
Waktu	Varchar	20	Menampung waktu
Status	Int	1	Menampung status

f. Tabel Pesan

Tabel 4.11 Tabel Pesan

Name Field	Type	Lenght	Keterangan
Id_pesan	Int	11	Primary Key. Auto Increment
Pengirim	Varchar	50	Menampung pesan pengirim
Penerima	Varchar	50	Menampung Pesan Penerima
Isi_pesan	Text		Menampung Isi Pesan
Tanggal	Varchar	20	Menampung Tanggal
Jam	Varchar	10	Menampung Jam
status	Tinyint	1	Menampung Status

4. Desain

Desain tampilan aplikasi berbasis *web* merupakan sebuah rancangan awal dari membuat tampilan halaman aplikasinya. Untuk membuat sebuah aplikasi yang menarik dan mudah saat digunakan maka diperlukan perancangan tampilan yang bagus dan menarik serta mudah bagi orang lain yang mengoprasikannya.

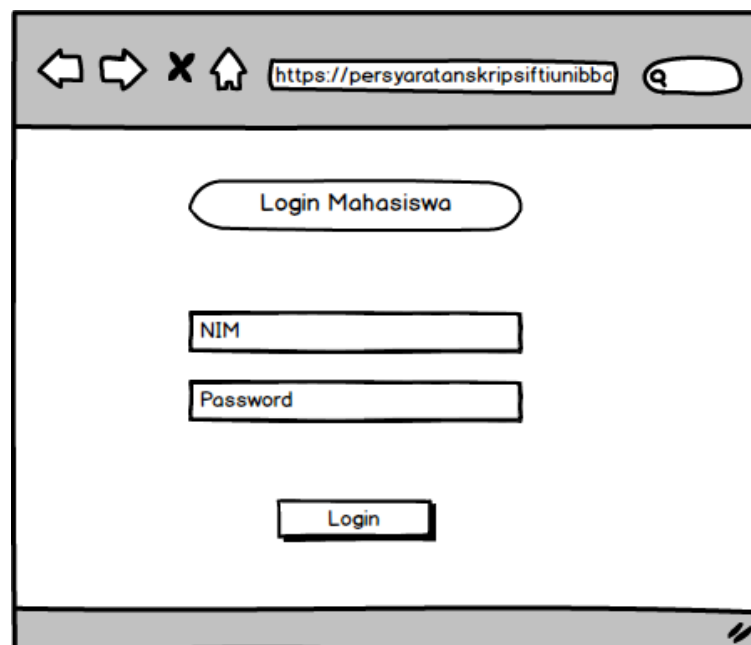
Sehingga penulis berusaha membuat tampilan yang bagus dan mendesain rancangan dengan *mockup* halaman aplikasi berbasis *web* agar mudah saat membuat tampilan *websitenya*. Dengan membuat *mockup* tampilan halaman aplikasinya dapat mempermudah saat membangun aplikasinya seperti menentukan fitur apa saja yang akan di sediakan di dalam halaman pada aplikasinya.

Hal ini lah yang menunjukan mockup termasuk salah satu hal yang perlu dipertimbangkan sejak awal sebelum memulai tahapan membangun aplikasi, Mockup juga bisa di artikan sebagai prototipe suatu halaman *website* atau gambar model yang dibuat secara menyeluruh dan mendetail.

Untuk perancangan desain antar muka dari aplikasi seleksi kelayakan pengajuan proposal skripsi berbasis web ini dibuat dalam bentuk *mockup* dengan menggunakan aplikasi *balasmiq mockup* hal ini bertujuan untuk memudahkan dalam pembuatan tampilan *user interface* di aplikasi yang akan di bangun.

1. MockUp Mahasiswa

a. User Interface Login Mahasiswa



Gambar 4.16 MockUp Login Mahasiswa

Gambar di atas adalah gambar menu *login* yang akan penulis bangun dengan menggambarkan tampilan *login* terlebih dahulu sebagai gambaran aplikasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram *activity* yang telah di rancang pada bagian sebelumnya.

b. User Interface Persyaratan Pertama

File	Keterangan	Aksi
KRS	-	
Bukti Pembayaran	-	
Usulan Skripsi	-	
Proposal Skripsi	-	
Laporan Penyerahan KP	-	
Sertifikasi	-	
Toefl	-	
KHS	-	

Gambar 4.17 MockUp Tampilan Upload Persyaratan Pertama

Gambar di atas adalah gambar menu tampilan utama serta tampilan *upload* persyaratan pertama yang harus mahasiswa lakukan yang akan penulis bangun dengan menggambarkan tampilan menu *home* terlebih dahulu sebagai gambaran aplikasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram *activity* yang telah di rancang pada bagian sebelumnya.

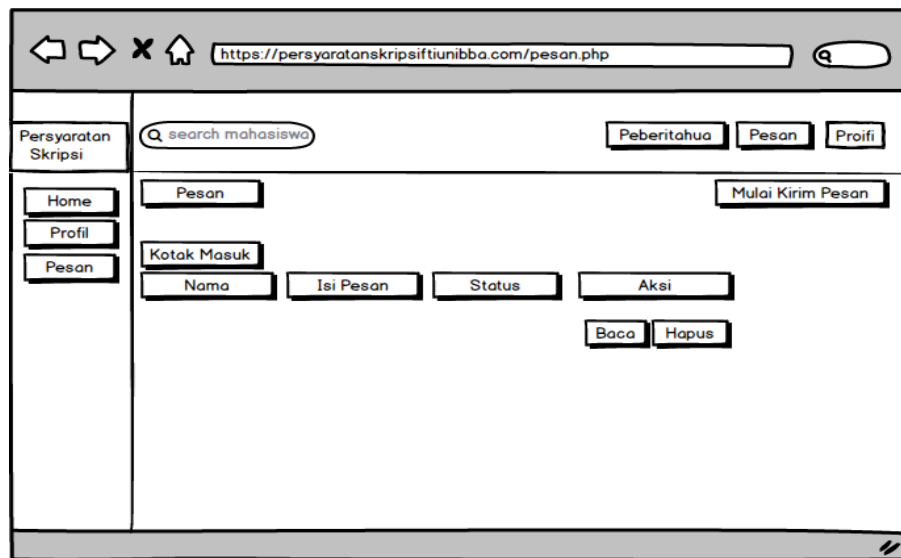
c. User Interface Persyaratan Kedua

File	Keterangan	Aksi
KRS	<input type="checkbox"/>	lihat hapus
Bukti Pembayaran	<input type="checkbox"/>	lihat hapus
Usulan Skripsi	<input type="checkbox"/>	lihat hapus
Proposal Skripsi	<input type="checkbox"/>	lihat hapus
Laporan Penyerahan KP	<input type="checkbox"/>	lihat hapus
Sertifikasi	<input type="checkbox"/>	lihat hapus
Toefl	<input type="checkbox"/>	lihat hapus
KHS	<input type="checkbox"/>	lihat hapus

Gambar 4.18 MockUp Mahasiswa Persyaratan Kedua

Gambar di atas adalah tampilan menu selanjutnya setelah mahasiswa melakukan pengumpulan persyaratan pertama dan setelah admin melakukan *update* status pada tahap pengumpulan persyaratan pertama yang penulis gambarkan dengan *user interface* seperti pada gambar di atas. rancangan diagram *activity* yang telah di rancang pada bagian sebelumnya.

d. User Interface Tampilan menu pesan mahasiswa

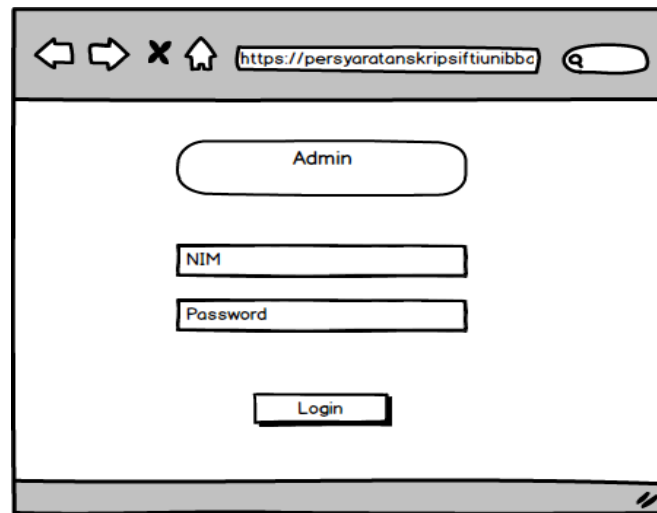


Gambar 4.19 MockUp Tampilan Pesan Mahasiswa

Gambar di atas adalah tampilan menu pesan yang dapat mahasiswa gunakan untuk berkomunikasi dengan admin dikala memiliki sebuah kendala dalam melakukan pengumpulan persyaratan skripsi juga bisa melakukan komunikasi dengan teman sejawatnya yang di gambarkan dengan *user interface* seperti gambar diatas.

2. MockUp Admin

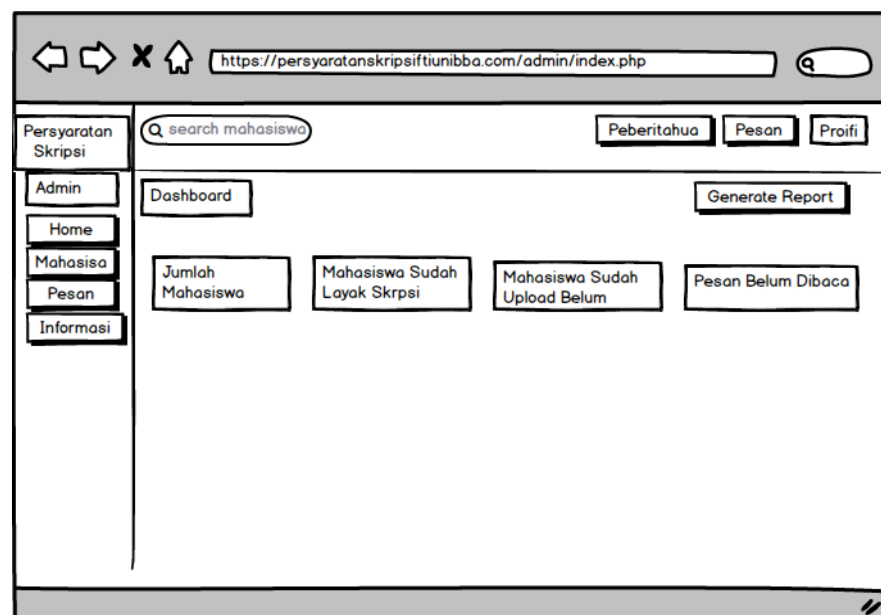
a. User Interface login admin



Gambar 4.20 MockUp Tampilan Menu Login

Gambar di atas adalah gambar menu *login* yang akan penulis bangun dengan menggambarkan tampilan *login* terlebih dahulu sebagai gambaran aplikasi yang akan di bangun dengan menggunakan *User Interface*. Yang mengacu pada rancangan diagram *activity* yang telah di rancang pada bagian sebelumnya.

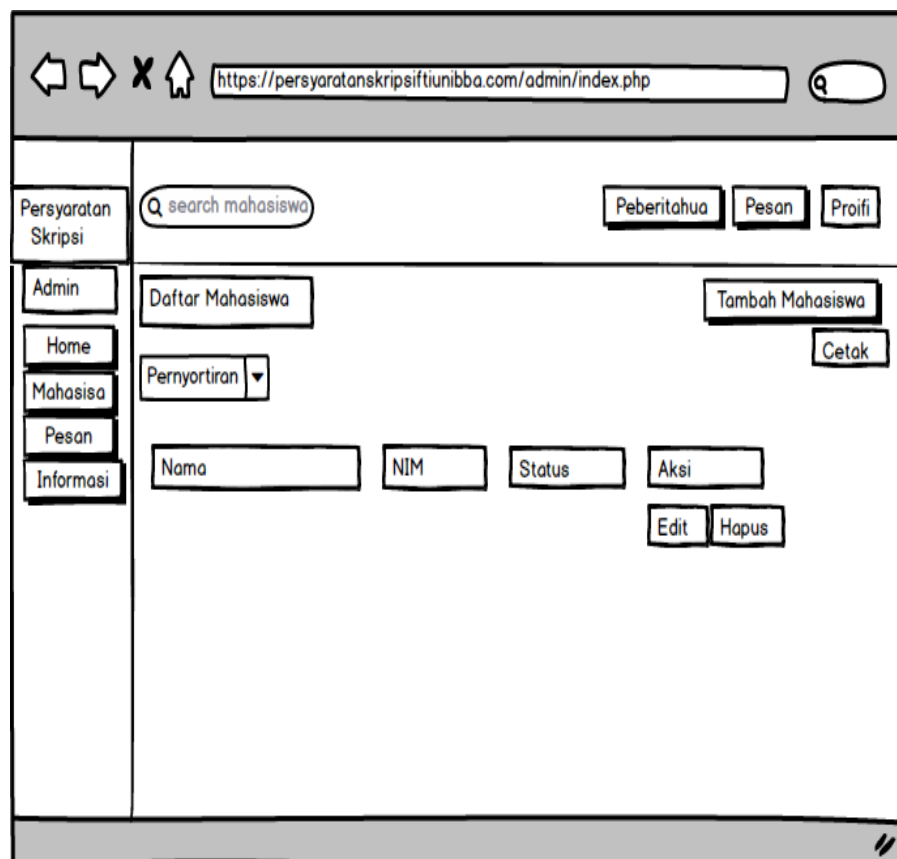
b. User Interface Menu Utama Admin



Gambar 4.21 MockUp Tampilan Menu Utama

Gambar di atas adalah gambar menu utama admin yang akan penulis bangun dengan menggambarkan, tampilan menu utama pada gambar di atas terlebih dahulu sebagai gambaran aplikasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram *activity* yang telah di rancang pada bagian sebelumnya.

c. User Interface Tampilan Menu mahasiswa



Gambar 4.22 MockUp tampilan Menu Mahasiswa

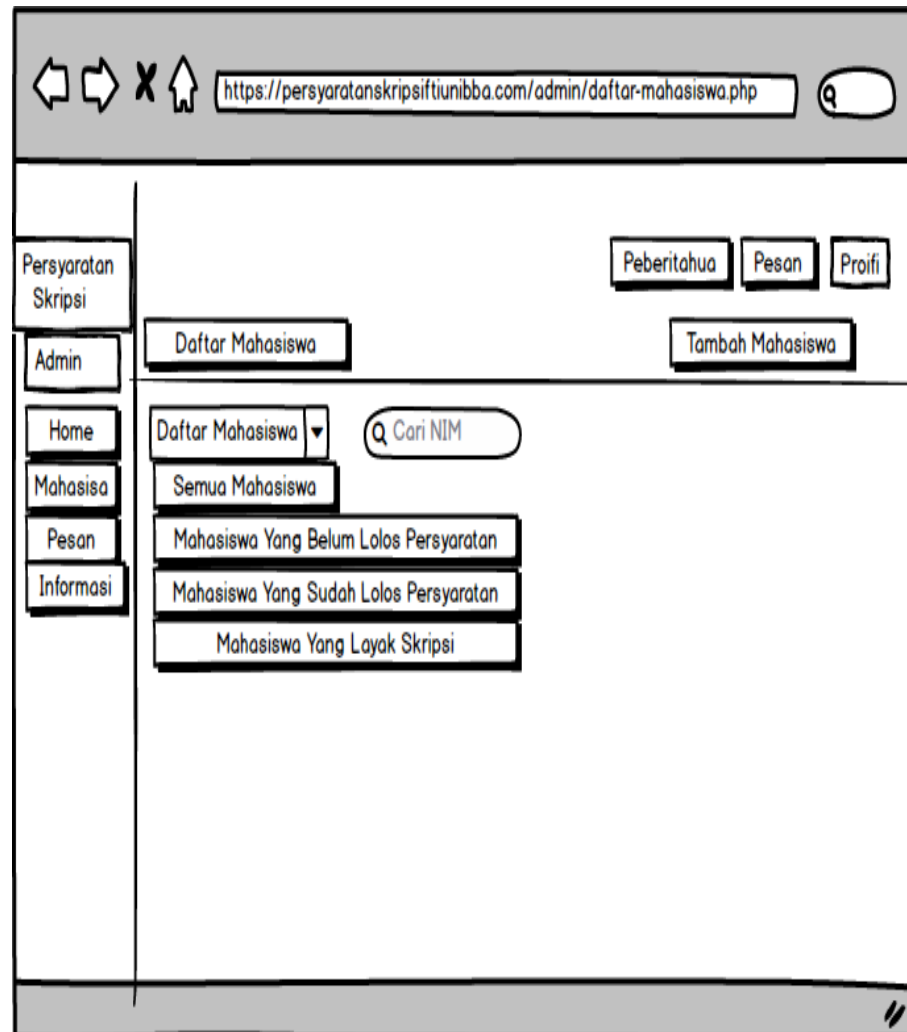
Gambar di atas adalah gambar menu mahasiswa admin yang akan penulis bangun dengan menggambarkan tampilan menu mahasiswa pada gambar di atas terlebih dahulu sebagai gambaran aplikasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram *activity* yang telah di rancang pada bagian sebelumnya. Pada menu ini admin dapat melihat semua mahasiswa serta menghapus mahasiswa tersebut

d. User Interface Tambah Mahasiswa

Gambar 4.23 MockUp Tambah Mahasiswa

Gambar di atas adalah gambar menu tambah mahasiswa pada menu mahasiswa admin yang akan penulis bangun dengan menggambarkan tampilan tambah mahasiswa pada gambar di atas terlebih dahulu sebagai gambaran tampilan aplikasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram activity yang telah di rancang pada bagian sebelumnya. Pada menu ini admin dapat menambahkan semua mahasiswa.

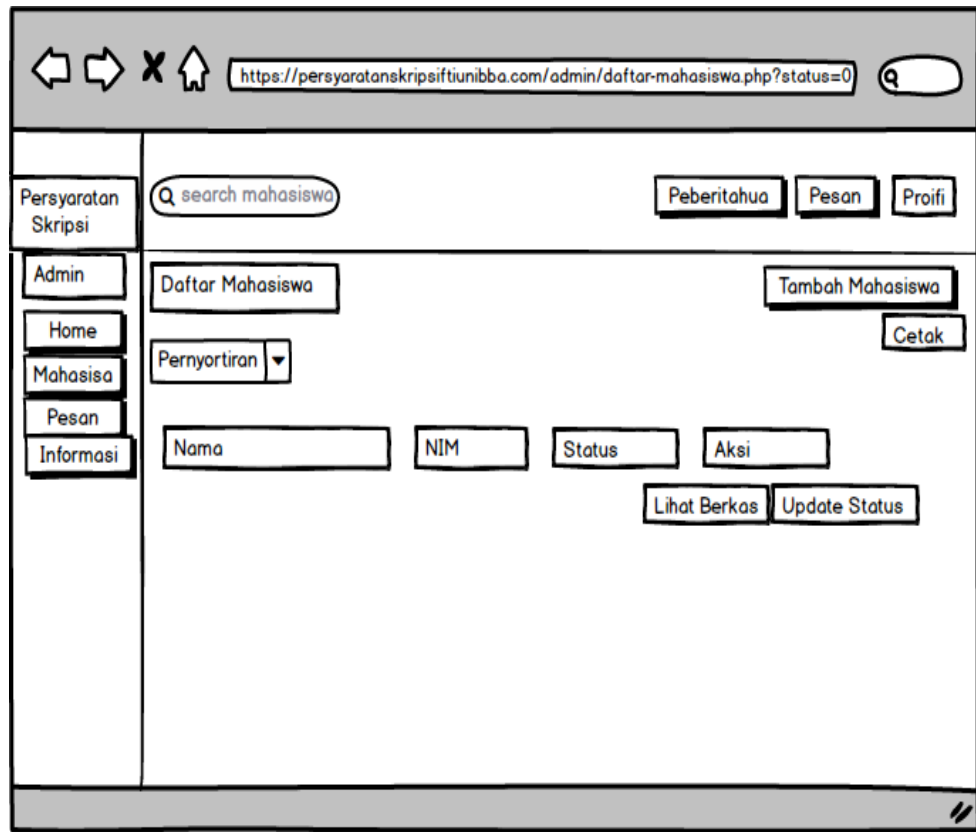
e. User Interface Tampilan Rekap Mahasiswa



Gambar 4.24 MockUp Tampilan rekap mahasiswa

Gambar di atas adalah gambar menu rekap mahasiswa pada menu mahasiswa admin yang akan penulis bangun dengan menggambarkan tampilan rekap mahasiswa pada gambar di atas terlebih dahulu sebagai gambaran tampilan aplikasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram activity yang telah di rancang pada bagian sebelumnya. Pada menu ini admin dapat melihat daftar mahasiswa yang telah melakukan pengumpulan persyaratan serta dapat melakukan pencetakan mahasiswa yang telah mengumpulkan persyaratan proposal skripsi maupun yang tidak.

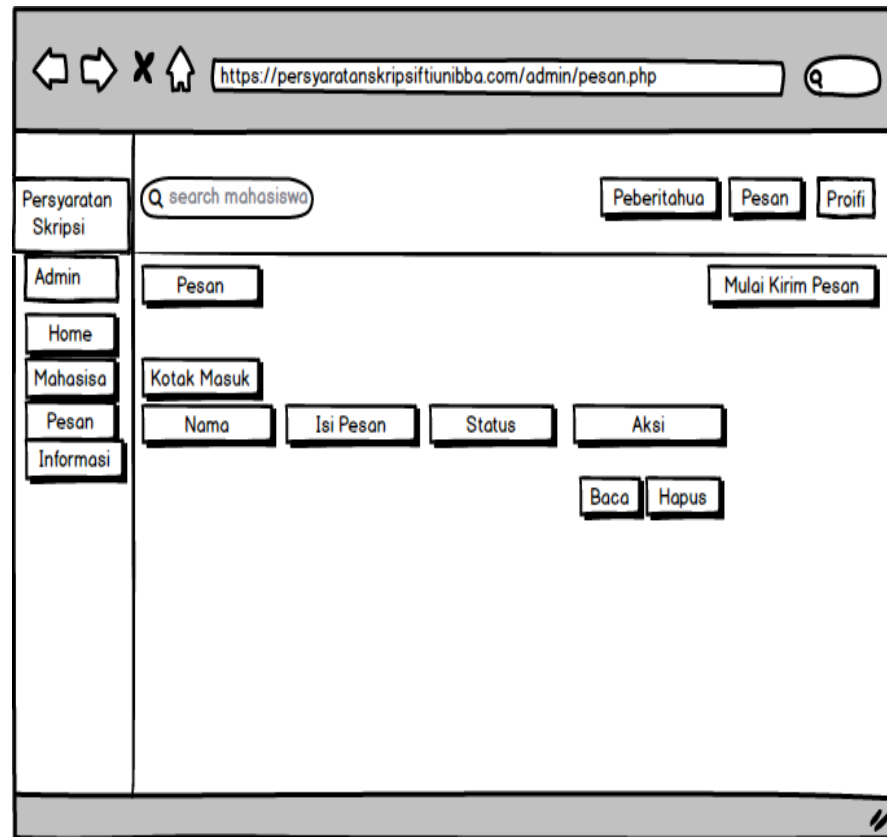
f. User Interface Tampilan Update status Admin



Gambar 4.25 MockUp Tampilan Update Status

Gambar di atas adalah gambar tampilan update status admin yang akan penulis bangun dengan menggambarkan tampilan update status setelah admin memilih menu mahasiswa yang telah mengumpulkan persyaratan pertama sebagai gambaran tampilan aplikasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram activity yang telah di rancang pada bagian sebelumnya. Pada menu ini admin dapat melihat meng *update* status mahasiswa setelah admin melihat berkas mahasiswa.

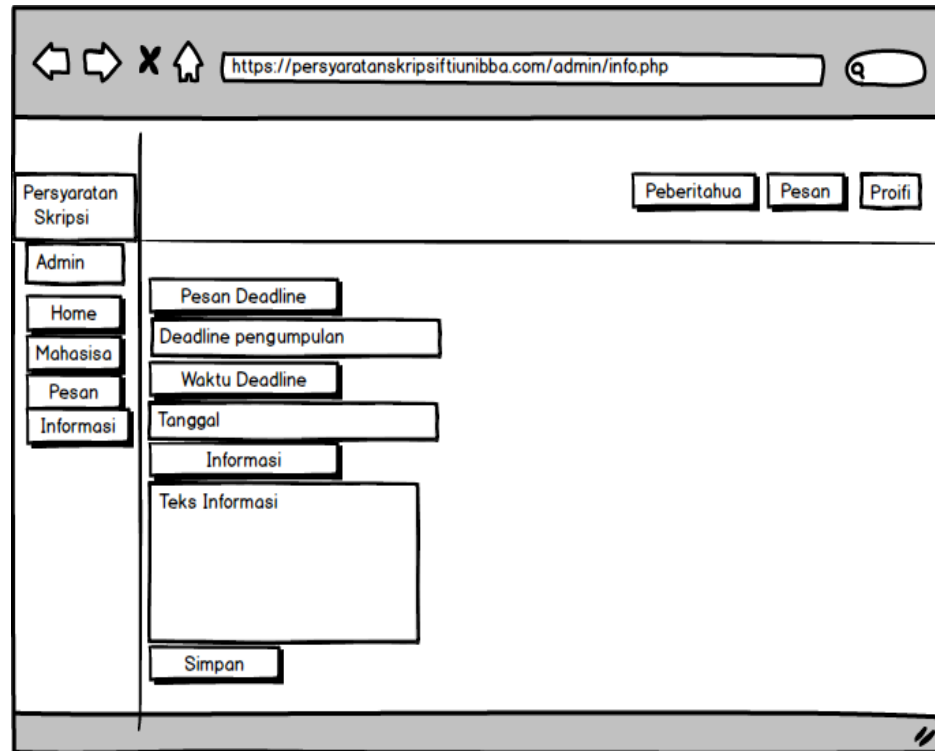
g. User Interface Tampilan menu pesan



Gambar 4.26 MockUp Tampilan Menu Pesan

Gambar di atas adalah gambar tampilan menu pesan admin yang akan penulis bangun dengan menggambarkan tampilan menu pesan setelah yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram *activity* yang telah di rancang pada bagian sebelumnya. Pada menu ini admin dapat melihat pesan masuk serta melakukan tulis pesan baru setelah itu memilih kontak lalu mengirimnya.

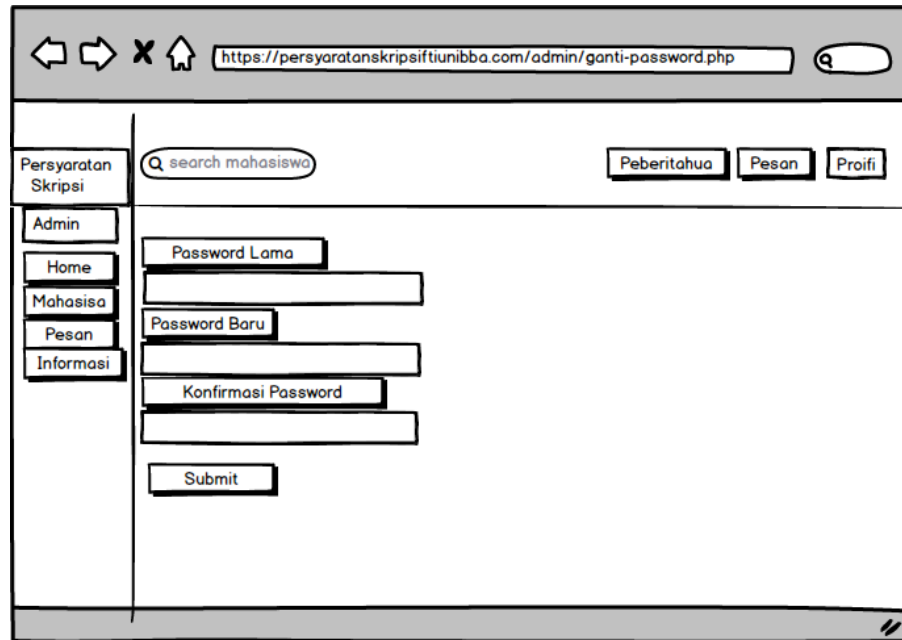
h. User Interface Tampilan Menu Informasi



Gambar 4.27 MockUp Tampilan Menu Informasi

Gambar di atas adalah gambar tampilan menu informasi admin yang akan penulis bangun dengan menggambarkan tampilan menu informasi yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram activity yang telah di rancang pada bagian sebelumnya. Pada menu ini admin dapat menulis informasi serta deatline pengumpulan persyaratan proposal skripsi serta melakukan edit infromasi yang akan di lihat oleh semua mahasiswa.

i. User Interface Tampilan Ganti Password Admin



The mockup shows a web browser window with the URL <https://persyaratanskripsiftiunibba.com/admin/ganti-password.php>. The page layout includes a sidebar on the left with a logo 'Persyaratan Skripsi' and a menu with items: Admin, Home, Mahasiswa, Pesan, and Informasi. The main content area at the top has a search bar labeled 'search mahasiswa' and three buttons: 'Peberitahua', 'Pesan', and 'Proifi'. The password change form itself consists of four input fields labeled 'Password Lama', 'Password Baru', and 'Konfirmasi Password', followed by a 'Submit' button.

Gambar 4.28 MockUp Tampilan Ganti Password admin

Gambar di atas adalah gambar tampilan ganti *password* admin yang akan penulis bangun dengan menggambarkan yang akan di bangun dengan menggunakan *user Interface*. Yang mengacu pada rancangan diagram activity yang telah di rancang pada bagian sebelumnya. Pada menu ini admin dapat mengganti *password* beda halnya dengan mahasiswa yang tidak dapat mengganti *password*.

BAB V

IMPLEMENTASI DAN PENGUJIAN

5.1 Implementasi

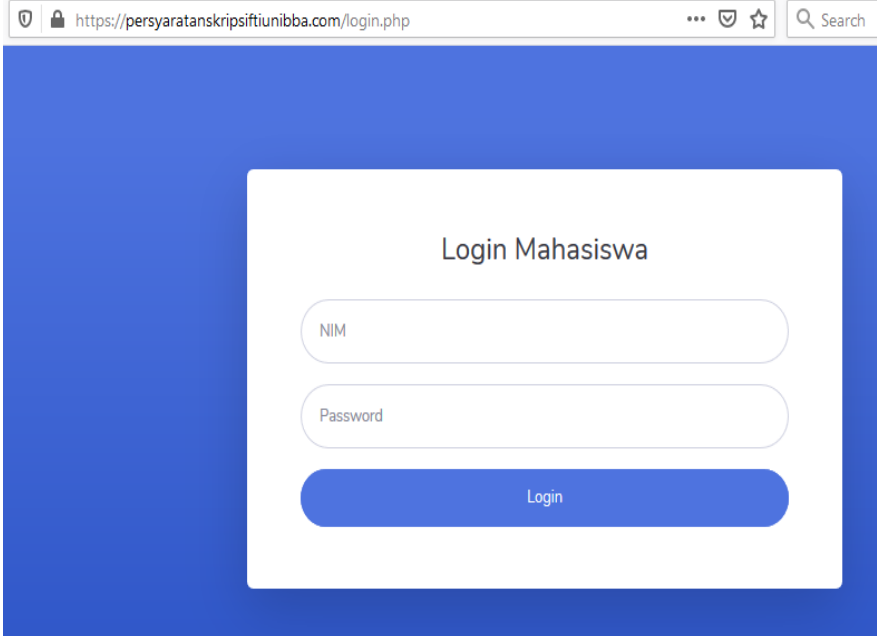
Implementasi adalah tahapan penerapan sistem berdasarkan hasil analisis dan perancangan yang dilakukan pada Bab IV. Pada bab V ini merupakan implementasi hasil rancangan menjadi Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi *Berbasis WEB* Untuk memudahkan mahasiswa dalam pengumpulan Persyaratan skripsi serta membantu tata usaha dalam melakukan pendataan terhadap mahasiswa dikarenakan cukup banyaknya Mahasiswa program Studi Teknik Informatika di Universitas Bale Bandung.

5.1.1 User Interface

Berikut adalah beberapa gambar hasil dari implementasi user interface :

1. Tampilan Mahasiswa

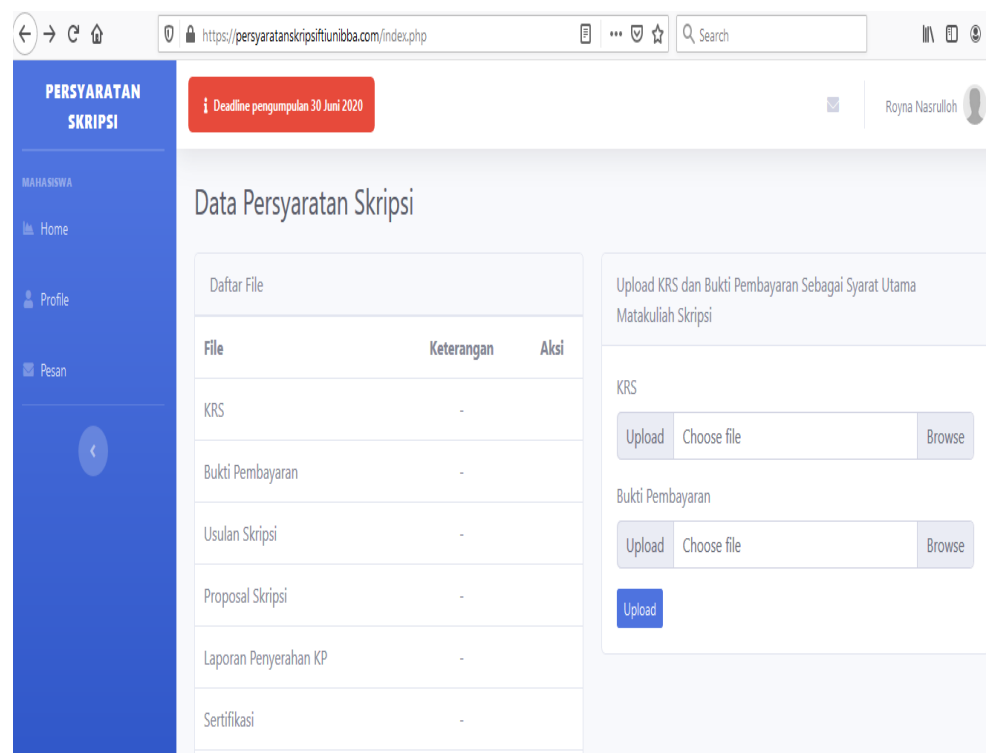
a. Tampilan Login Mahasiswa

A screenshot of a web browser displaying a login page for students. The browser's address bar shows the URL "https://persyaratanskripsitiunibba.com/login.php". The page has a solid blue background. In the center, there is a white rectangular box with rounded corners. Inside this box, the text "Login Mahasiswa" is centered at the top. Below the text, there are two input fields: the first is labeled "NIM" and the second is labeled "Password". Both fields have a light blue border and rounded ends. Below these fields is a blue button with rounded ends and the text "Login" in white. The overall design is clean and modern.

Gambar 5.1 *User Interface Login Admin*

Pada gambar diatas adalah halaman login mahasiswa. Yang di haruskan mahasiswa melakuak *input User name* dan *Password* menggunakan Identitas mahasiswa Berupa NIM sebelum bisa masuk ke halaman selanjutnya serta di sini mahasiswa tidak dapat mengganti *password* atau pun user name.

b. Tampilan halaman menu utama

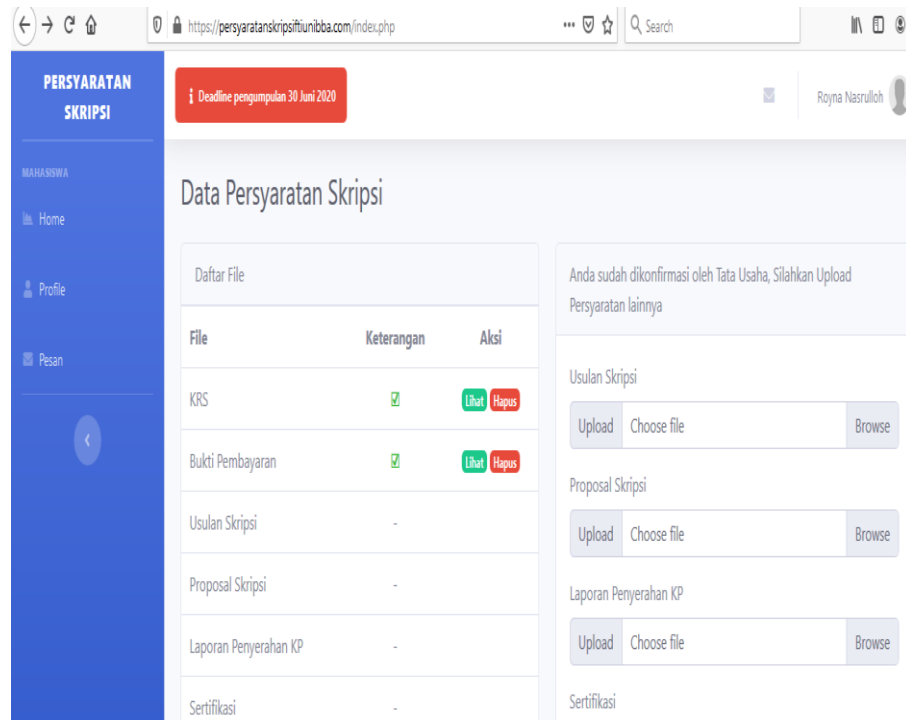


Gambar 5.2 User Interface Menu utama Mahasiswa

Pada gambar diatas adalah halaman utama pada saat mahasiswa telah melakukan *login* mahasiswa akan masuk ke halaman seperti gambar di atas serta mahasiswa dapat *mengupload* persyaratan skripsi tahap pertama serta mahasiswa dapat melihat informasi deatline pengumpulan persyaratan tahap pertama. Karena jika mahasiswa dalam mengumpulakn persyaratan skripsi lebih dari detaline yang telah ditentukan admin mahasiswa tidak dapat *mengupload* persyaratan, setelah mahasiswa telah melakukan *upload* persyaratan tahap pertama mahasiswa harus menunggu

perififikasi admin terlebih dahulu sebelum mahasiswa dapat meng-upload persyaratan selanjutnya.

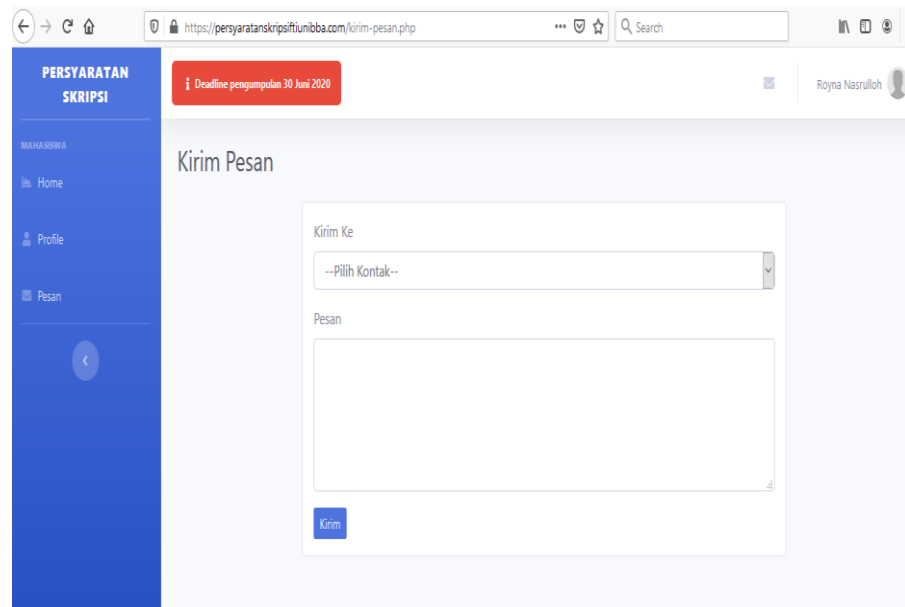
c. Tampilan Upload Persyaratan selanjutnya



Gambar 5.3 *User Interface Tampilan Upload Persyaratan Pertama*

Pada gambar diatas adalah tampilan halaman Pengumpulan Persyaratan Selanjutnya setelah mahasiswa sudah melakukan pengumpulan persyaratan tahap pertama dan admin telah melakukan pengecekan serta melakukan verifikasi terhadap persyartan tahap pertama baru lah mahasiswa akan bisa melakukan pengumpulan persyaratan tahap selanjutnya, dengan catatan *file* yang di *upload* harus berformatkan Pdf.

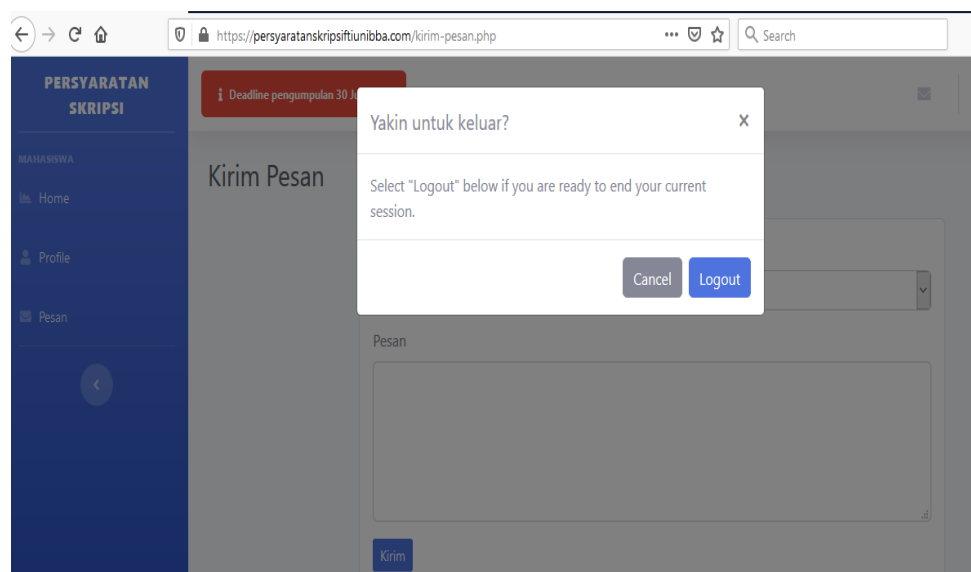
d. Tampilan tulis pesan



Gambar 5.4 *User Interface Tampilan Menu Pesan*

Pada gambar diatas adalah tampilan halaman Tulis Pesan yang bisa dilakukan mahasiswa jika ada terdapat sebuah kendala dan persyaratan skripsi, mahasiswa bisa bisa menulis pesan dan mengirim pesantersebut kepada admin dan menu pesan ini juggle bisa mahasiswa gunakan untuk melakukan komunikasi dengan teman seangkatannya.

e. Tampilan Logout

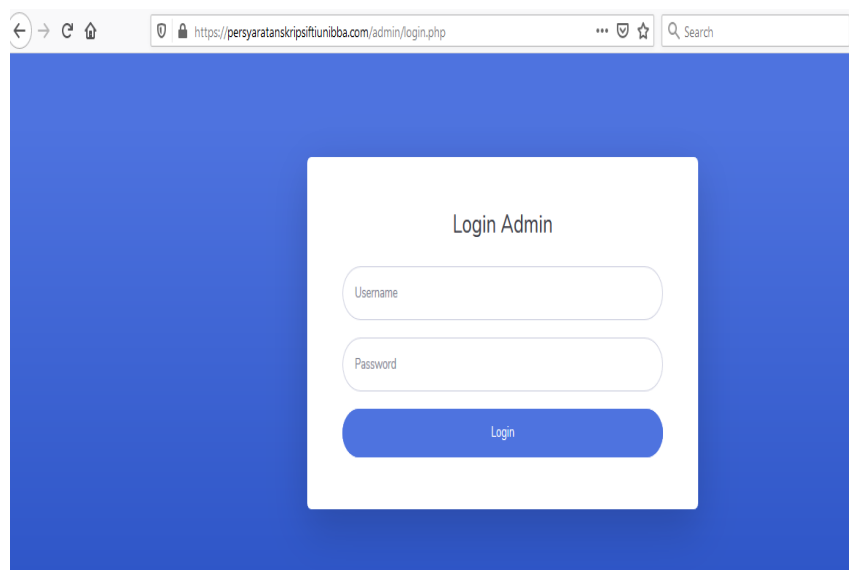


Gambar 5.5 *User Interface Tampilan Logout*

Pada gambar diatas adalah tampilan halaman menu *logout* yang bisa mahasiswa lakukan jika akan keluar dari aplikasi ini yaitu dengan mengklik *profil* yang ada di kanan atas lalu memilih *logout* atau keluar setelah mahasiswa mengklik tombol *logout*, setelah itu akan keluar pilihan seperti tampilan di atas lalu mahasiswa mengklik *logout* setelah itu mahasiswa akan keluar dari aplikasi tersebut.

2. Tampilan Admin

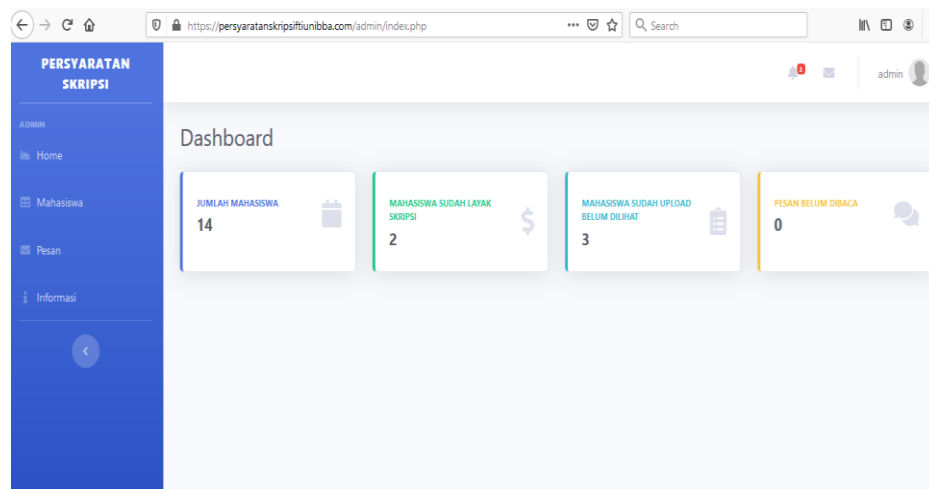
a. Tampilan Login Admin



Gambar 5.6 *User Interface* Tampilan Login Admin

Pada gambar diatas adalah tampilan halam *login* admin seperti halnya mahasiswa admin juga melakukan *login user name* dan *password* sebelum masuk ke halaman seanjutnya seperti halnya mahasiswa.

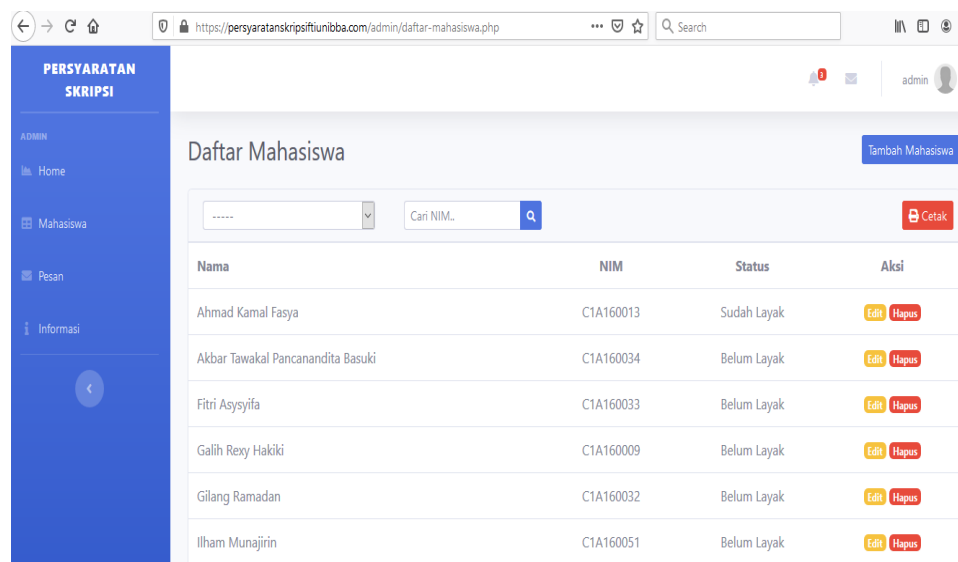
b. Tampilan Menu Utama Admin



Gambar 5.7 *User Interface* Tampilan Menu Utama Admin

Gambar di atas adalah tampilan menu utama admin disini admin dapat melihat jumlah keseluruhan mahasiswa, jumlah mahasiswa sudah *upload* persyaratan pertama, pesan masuk dan layak skripsi dilihat dari pengumpulan persyaratan skripsi.

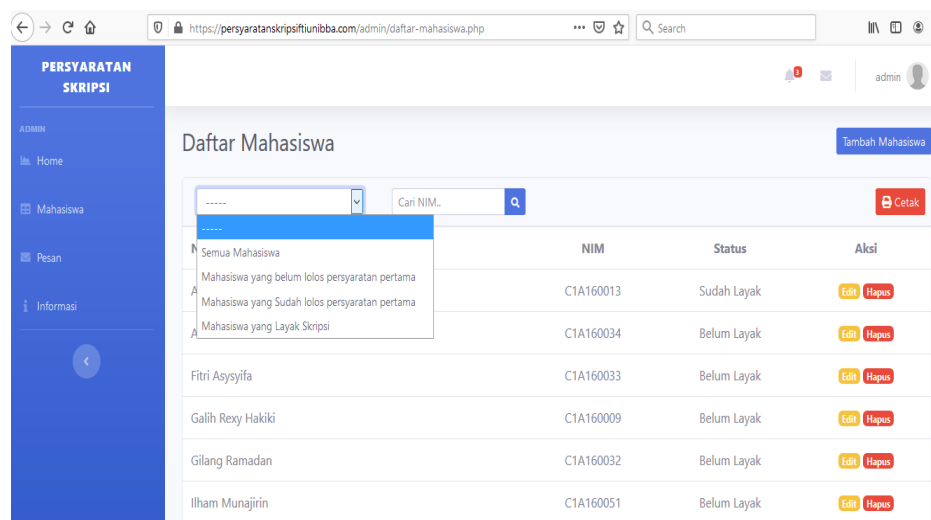
c. Tampilan Menu Mahasiswa



Gambar 5.8 *User Interface* Tampilan Menu Mahasiswa

Gambar di atas adalah Tampilah menu mahasiswa disini admin dapat melakukan hapus mahasiswa serta melakukan cetak atau *print* yang dapat admin lakukan serta tambah mahasiswa.

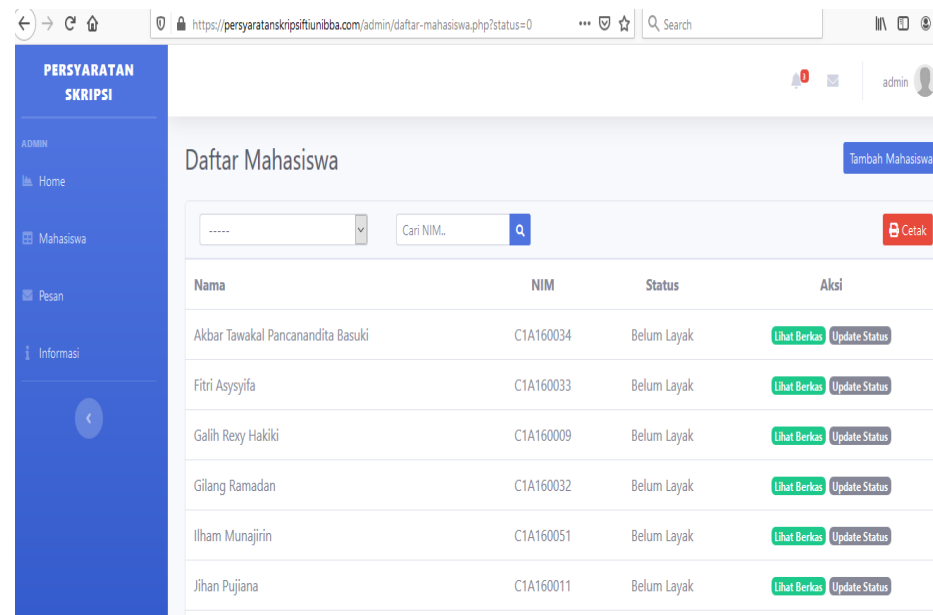
d. Tampilan Pilih Rekap Mahasiswa



Gambar 5.9 *User Interface* Tampilan Menu Pilih Rekap Mahasiswa

Gambar di atas adalah tampilan yang dapat admin pilih jika ingin melihat mahasiswa yang sudah *upload* persyaratan tahap pertama, mahasiswa belum lolos persyaratan tahap pertama, semua mahasiswa, dan mahasiswa layak skripsi, berikut adalah pilihan yang dapat admin pilih dikala admin hanya ingin mencetak mahasiswa yang belum melakukan pengumpulan persyaratan tahap pertama, jika admin mengklik cetak pada bagian kanan atas admin akan di lemparkan ke halaman Pdf dan setelah itu ,encetak berkas tersebut.

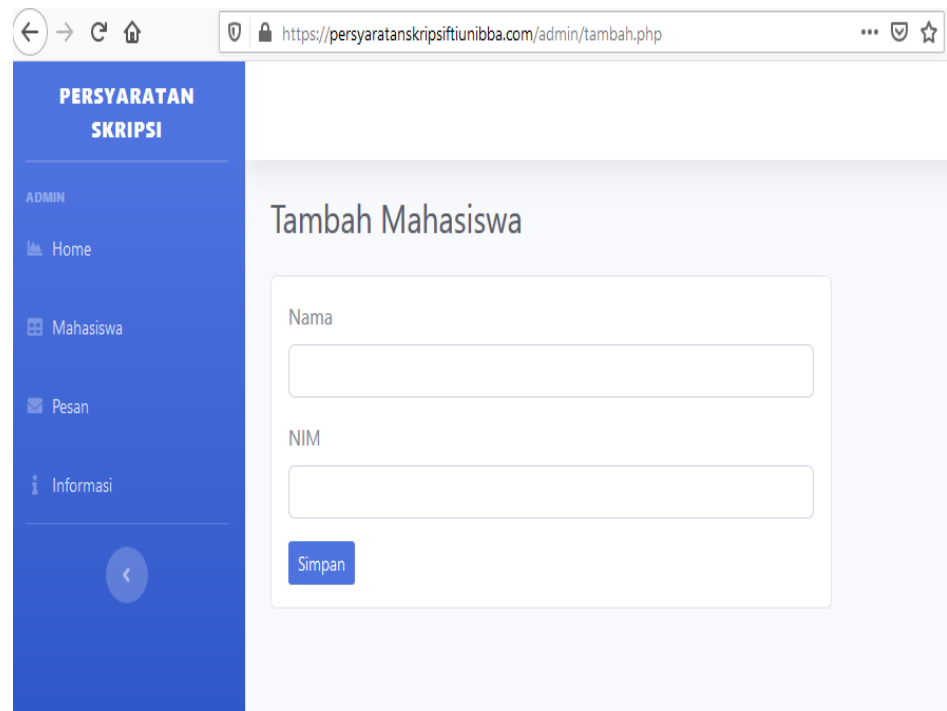
e. Tampilan Update status



Gambar 5.10 User Interface Tampilan Update Status

Gambar di atas adalah tampilan Update status mahasiswa yang dilakukan admin setelah mahasiswa melakukan pengumpulan persyaratan skripsi tahap pertama baru lah admin melakukan verifikasi mdengan cara meng-*update* status mahasiswa tersebut namun sebelum admin melakukan update status admin juga dapat melihat berkas atau *file* yang mahasiswa *upload* di takutkan mahasiswa salah meng *upload* persyaratan stersebut dan jika persyaratan yang mahasiswa lakukan telah benar maka admin melakukan *update status*.

f. Tampilan Tambah Mahasiswa

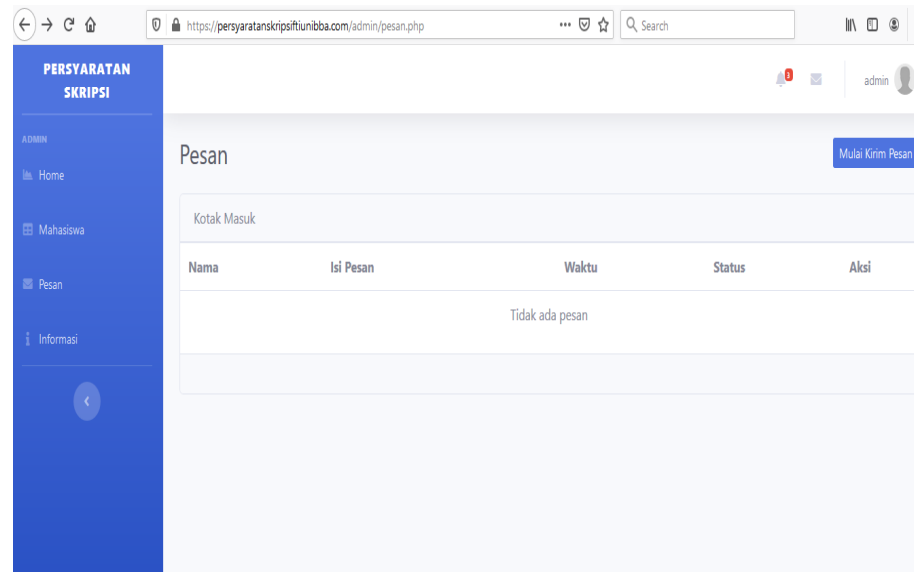


The screenshot displays a web application interface for adding a student. The browser's address bar indicates the URL: <https://persyaratanskripsiitiunibba.com/admin/tambah.php>. On the left, a blue sidebar features the header 'PERSYARATAN SKRIPSI' and an 'ADMIN' menu with options: Home, Mahasiswa, Pesan, and Informasi. The main content area, titled 'Tambah Mahasiswa', contains a form with two text input fields labeled 'Nama' and 'NIM', and a blue 'Simpan' button at the bottom.

Gambar 5.11 *User Interface* Tampilan Tambah Mahasiswa

Gambar di atas adalah menu tampilan tambah mahasiswa yang ada terletak di atas kanan disini admin melakukan input mahasiswa sesuai NIM dan nama mahasiswa yang nantinya NIM tersebut akan Mahasiswa gunakan untuk melakukan *login* ke aplikasi tersebut.

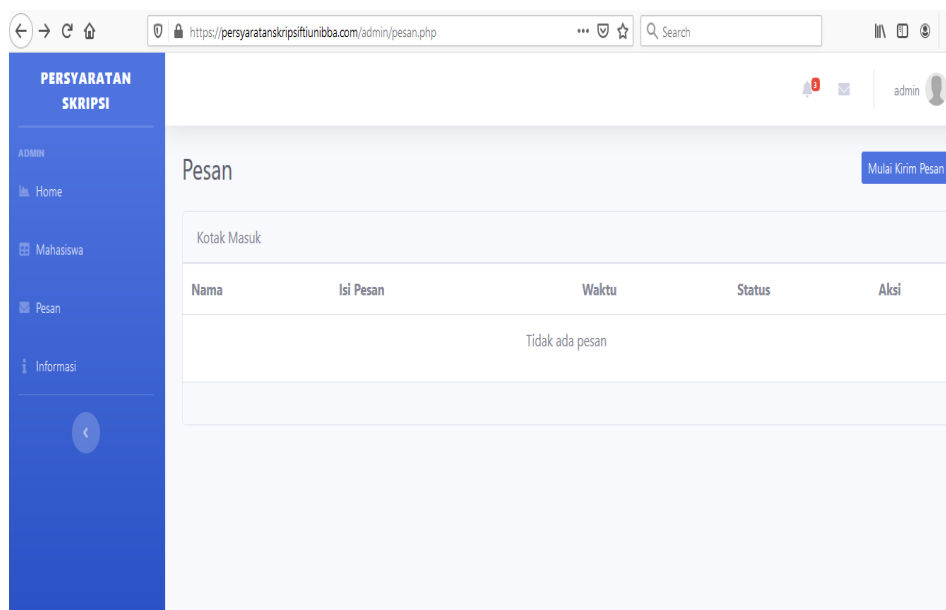
g. Tampilan Menu Pesan



Gambar 5.12 *User Interface* Tampilan menu Pesan

Gambar di atas adalah tampilan menu pesan disini admin dapat melihat pesan yang masuk dan dapat menulis pesan dengan mengklik *button* yang ada di sebelah kanan atas setelah admin mengklik *button* tersebut admin dapat menulis pesan dan memilih kontak atau memilih tujuan pesan yang akan di kirim.

h. Tampilan Menu Informasi



Gambar 5.13 *User Interface* Tampilan Menu Informasi

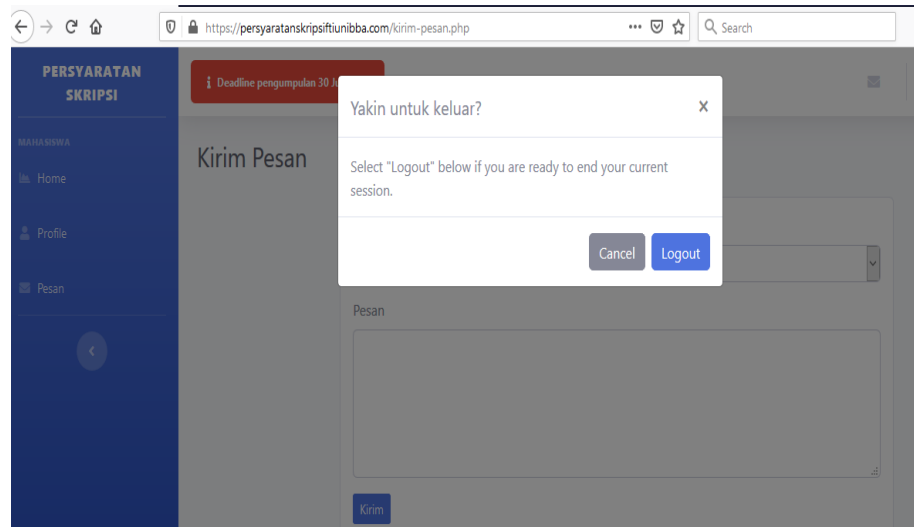
Gambar diatas adalah tampilan menu informasi dikala admin ingin mengubah atau menambahkan informasi untuk seluruh mahasiswa di sini melakukan input informasi serta waktu

i. Tampilan Ganti Password

Gambar 5.14 *User Interface* Tampilan menu Ganti Password

Beda lagi halnya dengan mahasiswa disini admin dapat mengubah *password* yang admin gunakan yaitu dengan mengklik nama admin yang ada di paling atas pojok atas lalu memilih pilihan ganti *password* setelah itu admin akan masuk ke halaman seperti pada gambar diatas dan melakukan pergantian *password* lama ke *password* baru.

j. Tampilan Logout



Gambar 5.15 User Interface Tampilan Logout

Seperti halnya mahasiswa admin juga melakukan *logout* jika ingin keluar dari aplikasi tersebut dengan meklik nama admin di atas kanan pojok atas dan memilih *logout* setelah itu akan muncul seperti gambar di atas lalu admin memilih *logout* jika memang ingin keluar dari aplikasi tersebut.

3. Tampilan Kesalahan Aplikasi

a. Tampilan Kesalahan Login

Gambar 5.16 Tampilan Kesalahan Login

Pada gambar diatas adalah tampilan halaman kesalahan login mahasiswa seperti *user name* dan *password* sebelum masuk ke menu utama jika belum terdaptar maha akan muncul kesalahan NIM belum ada.

b. Tampilan Kesalahan Upload File

PERSYARATAN SKRIPSI

MAHASISWA

Home

Profile

Pesan

Deadline pengumpulan 29 Juni 2020

Royana Nasrulloh

Data Persyaratan Skripsi

Format file harus PDF

File	Keterangan	Aksi
KRS	✓	Lihat Hapus
Bukti Pembayaran	✓	Lihat Hapus
Usulan Skripsi	-	
Proposal Skripsi	-	
Laporan Penyerahan KP	-	

Anda sudah dikonfirmasi oleh Tata Usaha, Silahkan Upload Persyaratan lainnya

Usulan Skripsi

Upload Choose file Browse

Proposal Skripsi

Upload Choose file Browse

Laporan Penyerahan KP

Upload Choose file Browse

Gambar 5.17 Tampilan Kesalahan *Upload File*

Pada gambar diatas adalah tampilan halaman kesalahan saat mahasiswa melakukan pengumpulan peryaratan skripsi akan muncul pemberitahuan *format file* harus PDF jika *file* yang mahasiswa kumpulkan bukan *file berformat* PDF.

5.2 Pengujian

Pengujian membangun aplikasi seleksi kelayakan pengumpulan persyaratan skripsi pada penelitian ini, mengumpulkan persyaratan proposal skripsi, data yang di kumpulkan dan langkah pengumpulan persyaratan proposal skripsi. Langkah yang di lakukan adalah sebagai berikut pada saat mahasiswa login dan mengumpulkan persyaratan tahap pertama setelah mahasiswa melakukan pengumpulan persyaratan, admin akan melakukan pengecekan terhadap berkas yang dikumpulkan serta melakukan update status,

setelah admin melakukan *update* status admin akan diminta kembali untuk melakukan pengumpulan persyaratan tahap kedua namun pada tahap pengumpulan persyaratan skripsi atahap pertama dan kedua *file* yang di *upload* harus berformatkan PDF karna jika bukan *file* PDF tidak bisa di *upload*. Dan jika pada saat melakukan pengumpulan persyaratan skripsi terdapat kendala mahka mahasiswa bisa melakukan komunikasi melalui menu pesan yang telah di sediakan, dalam aplikasi ini mahasiswa dapat melakukan komunikasi dengan adamin dan mahasiswa yang lainnya. Dalam pengumpulan persyaratan skripsi admin dapat memberikan informasi melalui menu informasi sekaligus memberikan *deatline* pengumpulan persyaratan skripsi, jika mahasiswa telat dalam melakukan pengumpulan persyaratan skripsi maka dalam tampilan mahasiswa tidak melakukan *upload* persyaratan skripsi.

5.2.1 Kegagalan Pengujian dengan Menggunakan Metode Black Box

sering disebut juga *glass-box testing*, merupakan metode *testing* yang menggunakan kontrol struktur dari rancangan prosedural untuk melakukan *test case* dan mengetahui *internal* dan *website*. *Design test* dijalankan pada semua *internal* dari *website* untuk melihat kegagalan yang ada pada aplikasi tersebut.

Tabel 5.1 Pengujian Kesalahan

NO	Fungsi Yang diuji	Cara Pengujian	Hasil Yang diharapkan	Hasil Pengujian
1	Login	Memasukan user name dengan kondisi (Salah)	Menampilkan kesalahan	Berhasil
		Memasukan passsword dengan kondisi salah	Menampilkan kesalahan	Berhasil
2	Upload data	Memasukan file pembayaran bukan Pdf	Menampilkan kesalahan	Berhasil

5.2.1 Hasil Pengujian dengan Menggunakan Metode Black Box

sering disebut juga *glass-box testing*, merupakan metode *testing* yang menggunakan kontrol struktur dari rancangan prosedural untuk melakukan *test case* dan mengetahui *internal* dan *website*. *Design test* dijalankan pada semua *internal* dari *website* untuk memastikan mereka beroperasi berdasarkan spesifikasi dan desain.

Tabel 5.2 Pengujian Perangkat admin

No	Fungsi yang diuji	Cara Pengujian	Hasil Yang diharapkan	Hasil Pengujian
1	Form login	Pada website pengumpulan persyaratan menampilkan form login untuk masuk ke dalam sistem admin	Menampilkan form login dan berhasil masuk ke dalam halaman website	OK
2	Form login	Admin dapat menginput user name dan password dan jika terdapat kesalahan dalam melakukan pengisian dalam user name dan password admin tidak dapat masuk kedalam sistem	Menampilkan kondisi dimana admin terdapat kesalahan dalam melakukan pengisian user name dan password	OK
3	Form Menu Utama	Memilih menu home mahasiswa, dan dapat menampilkan jumlah mahasiswa sudah layak skripsi, mahasiswa sudah	Menampilkan form menu utama	OK

		upload dan belum di lihat, dan pesan belum di baca		
4	Form Mahasiswa	Memilih menu Mahasiswa dan memilih Tambah mahasiswa kemudian dapat menambah, mengedit dan menghapus mahasiswa	Menampilkan form tambah mahasiswa dan dan mahasiswa dapat di tambah, edit dan hapus	OK
5	Form Mahasiswa	Memilih menu mahasiswa dan pilih daftar mahasiswa kemudian dapat mencetak semua mahasiswa	Menampilkan rekap mahasiswa dan menu print	OK
6	Form mahasiswa	Memilih menu mahasiswa yang belum lolos persyaratan pertama, kemudian dapat melihat berkas serta update status mahasiswa	Menampilkan menu mahasiswa yang belum lolos persyaratan pertama, lihat berkas, dan update status.	OK
7	Form Mahasiswa	Memilih menu rekap mahasiswa yang sudah lolos persyaratan pertama	Menampilkan hasil rekap mahasiswa yang sudah lolos persyaratan pertama	OK
8	Form Mahasiswa	Memilih menu rekap mahasiswa yang layak skripsi	Menampilkan hasil rekap mahasiswa yang layak skripsi	OK
9	Form Mahasiswa	Memilih pencarian mahasiswa berdasarkan NIM	Menampilkan hasil pencarian berdasarkan NIM	OK

10	Form Pesan	Memilih menu pesan kemudian dapat membaca, menghapus dan menulis pesan	Menampilkan form menu pesan dan dapat membaca, menghapus serta mengirim pesan	OK
11	Form Informasi	Memilih menu informasi kemudian dapat menulis informasi setelah itu memilih menu simpan	Menampilkan form informasi dan dapat menambahkan informasi	OK
12	Form Admin	Memilih menu profil dan dapat mengganti nama admin	Menampilkan form profil dan dapat mengganti nama	OK
13	Form Ganti Password	Menampilkan menu ganti Password dan dapat mengganti Password admin	Menampilkan menu Ganti Password dan dapat mengganti Password admin	OK
14	Form Logout	Menampilkan menu <i>Logout</i> dan menampilkan pilihan keluar atau tidak	Menampilkan menu <i>logout</i> dan memberikan pilihan ya atau tidak	OK

Tabel 5.3 Pengujian Perangkat Mahasiswa

No	Fungsi yang diuji	Cara Pengujian	Hasil Yang diharapkan	Hasil Pengujian
1	Form Login	Menampilkan menu login dan melakukan pengisian NIM dan Password dan jika mahasiswa belum terdaftar ada pemberitahuan mahasiswa belum terdaftar	Menampilkan form login dan menampilkan kondisi jika mahasiswa belum terdaftar	OK

2	Form Menu Utama	Menampilkan menu utama sekaligus pengumpulan persyaratan tahap pertama, serta informasi yang telah admin input	Menampilkan form menu utama sekaligus menu pengumpulan persyaratan tahap pertama serta informasi yang admin berikan	OK
3	Form menu utama	Melakukan upload persyaratan dan jika persyaratan bukan berformat pdf file tidak dapat di upload	Menampilkan form persyaratan tahap pertama dan upload file dalam format Pdf	OK
4	Form Profil	Menampilkan form profil dan hanya dapat mengganti atau upload foto saja tidak dapat mengganti Nama dan NIM	Menampilkan form profil mahasiswa dan hasil upload foto mahasiswa	OK
5	Form pesan	Menampilkan menu pesan dan dapat membaca hapus serta menulis pesan baru	Menampilkan form pesan	OK
6	Form Persyaratan Tahap kedua	Menampilkan form persyaratan tahap kedua dan jika file belum terpenuhi semuanya hanya user atau mahasiswa yang dapat melihat	Menampilkan form persyaratan tahap kedua	OK
7	Form logout	Menampilkan menu logout dan memberikan kondisi atau pilihan ya atau tidak untuk keluar dari sistem	Menampilkan menu logout	OK

BAB VI

KESIMPULAN DAN SARAN

8.1 Kesimpulan

Berdasarkan penelitian yang dilakukan penyusun melalui beberapa tahapan yang dilakukan pada bab-bab sebelumnya, maka penyusun dapat menyimpulkan bahwa:

1. Aplikasi Seleksi Kelayakan Pengajuan Proposal Skripsi Berbasis Web dapat membantu mahasiswa dalam pengumpulan persyaratan skripsi.
2. Dengan melakukan pengumpulan persyaratan skripsi secara online, membantu mahasiswa dengan mudah.
3. Aplikasi seleksi kelayakan pengajuan proposal skripsi dapat di akses oleh Mahasiswa dan Tata Usaha.

8.2 Saran

Berdasarkan hasil penelitian, maka penulis merekomendasikan atau menyarankan beberapa hal, yaitu :

1. Melakukan pengumpulan persyaratan skripsi dengan menggunakan aplikasi.
2. Melakukan input pengumpulan persyaratan dan melakukan komunikasi jika ada pengumpulan persyaratan skripsi yang keliru atau pun yang salah dalam input persyaratan melalui menu pesan yang dapat di gunakan oleh mahasiswa dan admin.
3. Aplikasi ini bisa digunakan oleh mahasiswa dan admin serta diharapkan dapat membantu mahasiswa dan admin serta tidak perlu memerlukan waktu lama

DAFTAR PUSTAKA

- Ambar T Sulistiyani dan Rosidah. 2003 “ *Manajemen Sumber Daya Manusia*”. Graha Ilmu: Yogyakarta.
- CSS <https://www.it-jurnal.com/pengertian-css-cascading-style-sheet/>
- Hasibuan Malayu <http://irrineayu.wordpress.com/2015/03/25/seleksi-sdm-definisi-tujuan-metode-proses-kendala-dan-faktor-yang-dipertimbangkan-dalam-proses-seleksi/>
- Hidayatullah Priyanto dan Khairul Jauhari Kawistara: 2017. “*Pemrograman Web*”. Informatika Bandung: Bandung.
- Kaban Roberto: 2018. “*Bootstrap CSS Framework*” . Penerbit Andi: Medan
- Madcoms dan Andi : 2016. “*Pemrograman PHP dan MySQL Untuk Pemula*”. Penerbit Andi: Yogyakarta.
- Permana, Diyan Agus dan Dewantara, Rizki Yudhi. 2018 “*Analisis Dan Perancangan Sistem Informasi Perekrutan Karyawan Berbasis Web*”, Jurnal Administrasi Bisnis (JAB), Vol. 56 No. 1.
- Pedoman Skripsi FTI UNIBBA.
- Pratama Putu Agus Eka: 2015. “*E-Commerce, E-Business dan Mobile Commerce*”. Informatika Bandung: Bandung.
- Raharjo dan Budi: 2018. “*Modul Pemrograman Web (HTML, PHP & MySQL/MariaDB) Edisi Keempat*”. Modula: Bandung.
- Rosa A.S dan M.Shalahuddin: 2019. “*Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Edisi Revisi*”. Informatika Bandung: Bandung.
- Yuniva, Ika dan Syafi'i, Ahmad. 2018 “*Pendekatan Model Waterfall Dalam Perancangan Web Sistem Informasi Penerimaan Siswa Baru*

Menggunakan Framework Bootstrap”, Paradigma, Vol. XX No. I, p-
ISSN: 1410-5063, e-ISSN: 2579-3500.

Zulkarnain Fera: 2015. “*Analisis Pengembangan Sistem Pendataan Siswa Berbasis Web Php dan Mysql Guna Mempermudah Pengelolaan Data Siswa*”,
Tugas Akhir Skripsi.

<http://www.maxmanroe.com/vid/teknologi/pengertian-aplikasi.html>

<http://brainly.co.id/tugas/2637786>

<http://bahasa.foresteract.com//skripsi/>

[http://www.martinrecords.com/technology/pengertian-web-browser-sejarah-
fungsi-serta-contohnya/](http://www.martinrecords.com/technology/pengertian-web-browser-sejarah-fungsi-serta-contohnya/)

DAFTAR LAMPIRAN

1. Koding Program Mahasiswa

A. Program Login

```
<?php
    require_once 'function/koneksi.php';
    session_start();
    if(isset($_SESSION['user'])) {
        header('Location: index.php');
    }
    $error = "";
    if(isset($_POST['submit'])) {
        $nim = $_POST['nim'];
        $password = $_POST['password'];
        //validasi atau logika
        if(!empty(trim($nim)) && !empty(trim($password))) {
            $query_read = "SELECT * FROM mahasiswa WHERE
nim='$nim'";
            $result_read = mysqli_query($koneksi, $query_read);
            if($user = mysqli_num_rows($result_read) != 0) {
                $row = mysqli_fetch_assoc($result_read);
                if(password_verify($password, $row['password'])) {
                    $_SESSION['user'] = $row['nim'];
                    header('Location: index.php');
                } else {
                    $error = 'nim atau password salah';
                }
            } else {
                $error = 'nim belum ada';
            }
        } else {
            $error = 'nim atau password salah';
        }
    }
}
```

```

        $error = 'Data tidak boleh kosong';
    }
}

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no">
    <meta name="description" content="">
    <meta name="author" content="">
    <title>Login Mahasiswa</title>
    <!-- Custom fonts for this template-->
    <link href="assets/vendor/fontawesome-free/css/all.min.css" rel="stylesheet"
type="text/css">
    <link
href="https://fonts.googleapis.com/css?family=Nunito:200,200i,300,300i,400,400
i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">
    <!-- Custom styles for this template-->
    <link href="assets/css/sb-admin-2.min.css" rel="stylesheet">
</head>
<body class="bg-gradient-primary">
    <div class="container">
        <!-- Outer Row -->
        <div class="row justify-content-center mt-5">
            <div class="col-lg-6">
                <div class="card o-hidden border-0 shadow-lg my-5">
                    <div class="card-body p-0">
                        <!-- Nested Row within Card Body -->
                        <div class="row">

```

```

<div class="col-lg">
  <div class="p-5">
    <div class="text-center">
      <h1 class="h4 text-gray-900 mb-4">Login Mahasiswa</h1>
    </div>
    <?php if($error) : ?>
      <div class="alert alert-danger">
        <?= $error; ?>
      </div>
    <?php endif; ?>
    <form class="user" method="POST" action="">
      <div class="form-group">
        <input type="text" class="form-control form-control-user"
placeholder="NIM" name="nim">
      </div>
      <div class="form-group">
        <input type="password" class="form-control form-control-user"
id="password" placeholder="Password" name="password">
      </div>
      <button type="submit" name="submit" class="btn btn-primary btn-
user btn-block">
        Login
      </button>
    </form>
  </div>
</div>
</div>
</div>
</div>
</div>

```

```

<script src="assets/vendor/jquery/jquery.min.js"></script>
<script src="assets/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
<script src="assets/vendor/jquery-easing/jquery.easing.min.js"></script>
<script src="assets/js/sb-admin-2.min.js"></script>
</body>
</html>

```

B. Halaman Utama Mahasiswa

```

<?php

    require_once 'function/koneksi.php';
    require_once 'templates/mahasiswa_header.php' ;
    require_once 'templates/sidebar.php' ;
    require_once 'templates/topbar.php' ;

    $nim = $_SESSION['user'];
    $sql = "SELECT * FROM mahasiswa WHERE nim = '$nim'";
    $query = mysqli_query($koneksi, $sql);
    $row = mysqli_fetch_assoc($query);
    $nama = $row['nama'];

?>
<div class="container-fluid">
    <div class="d-sm-flex align-items-center justify-content-between mb-4">
        <h1 class="h3 mb-0 text-gray-800">Data Persyaratan Skripsi</h1>
    </div>
    <?php if(isset($_SESSION['alert'])) : ?>
        <div class="alert alert-success">
            <?= $_SESSION['alert']; ?>
        </div>
        <?php unset($_SESSION['alert']); ?>
    <?php endif; ?>

```



```

<?php if(isset($_SESSION['error'])) : ?>
    <div class="alert alert-danger">
        <?= $_SESSION['error']; ?>
    </div>
    <?php unset($_SESSION['error']); ?>
<?php endif; ?>
<div class="row">
<div class="col-md-6">
    <div class="card">
<div class="card-header">Daftar File</div>
<?php
$sql_file = "SELECT * FROM file WHERE nim ='$nim'"; $query_file =
mysqli_query($koneksi, $sql_file);
$row_file = mysqli_fetch_assoc($query_file);
?>
<div class="table-responsive">
<table class="table">
    <tr>
<th>File</th>
<th class="text-center">Keterangan</th>
<th class="text-center">Aksi</th>
</tr>
<tr>
<td>KRS</td>
<td class="text-center"><?= ($row_file['krs'] != null) ? '&#9745;': '-'; ?></td>
<td class="text-center">
<?php if($row_file['krs'] != null): ?>
<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?= $row_file['krs']; ?>"
class="badge badge-success" target="_blank">Lihat</a>
<a class="badge badge-danger" onclick="return confirm('Apa anda yakin?')"
href="hapus-file.php?nim=<?= $nim; ?>&file=<?= $row_file['krs'];
?>&jenis_file=krs">Hapus</a>

```

```

<?php endif; ?>
</td>
</tr>
<tr>
<td>Bukti Pembayaran</td>
<td class="text-center"><?=( $row_file['bukti_pembayaran'] != null) ?
'&#9745;' : '-'; ?></td>
<td class="text-center">
<?php if($row_file['bukti_pembayaran'] != null): ?>
<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?=
$row_file['bukti_pembayaran']; ?>" class="badge badge-success"
target="_blank">Lihat</a><a class="badge badge-danger" onclick="return
confirm('Apa anda yakin?')" href="hapus-file.php?nim=<?= $nim;
?>&file=<?= $row_file['bukti_pembayaran'];
?>&jenis_file=bukti_pembayaran">Hapus</a>
<?php endif; ?>
</td>
</tr>
<tr>
<td>Usulan Skripsi</td>
<td class="text-center"><?=( $row_file['usulan_skripsi'] != null) ? '&#9745;' :
'-'; ?></td>
<td class="text-center">
<?php if($row_file['usulan_skripsi'] != null): ?>
<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?=
$row_file['usulan_skripsi']; ?>" class="badge badge-success"
target="_blank">Lihat</a><a class="badge badge-danger" onclick="return
confirm('Apa anda yakin?')" href="hapus-file.php?nim=<?= $nim;
?>&file=<?= $row_file['usulan_skripsi'];
?>&jenis_file=usulan_skripsi">Hapus</a>
<?php endif; ?>
</td>

```

```

</tr>

<tr>

<td>Proposal Skripsi</td>

<td class="text-center"><?=( $row_file['proposal_skripsi'] != null) ? '&#9745;'
: '-'; ?></td>

<td class="text-center">

<?php if($row_file['proposal_skripsi'] != null): ?>

<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?=
$row_file['proposal_skripsi']; ?>" class="badge badge-success"
target="_blank">Lihat</a><a class="badge badge-danger" onclick="return
confirm('Apa anda yakin?')" href="hapus-file.php?nim=<?= $nim;
?>&file=<?= $row_file['proposal_skripsi'];
?>&jenis_file=proposal_skripsi">Hapus</a>

<?php endif; ?>

</td>

</tr>

<tr>

<td>Laporan Penyerahan KP</td>

<td class="text-center"><?=( $row_file['penyerahan_kp'] != null) ? '&#9745;' :
'-'; ?></td>

<td class="text-center">

<?php if($row_file['penyerahan_kp'] != null): ?>

<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?= $row_file['penyerahan_kp'];
?>" class="badge badge-success" target="_blank">Lihat</a><a class="badge
badge-danger" onclick="return confirm('Apa anda yakin?')" href="hapus-
file.php?nim=<?= $nim; ?>&file=<?= $row_file['penyerahan_kp'];
?>&jenis_file=penyerahan_kp">Hapus</a>

<?php endif; ?>

</td>

</tr>

<tr>

<td>Sertifikasi</td>

```

```

<td class="text-center"><?=( $row_file['sertifikasi'] != null) ? '&#9745;' : '-';
?></td>

<td class="text-center">
<?php if($row_file['sertifikasi'] != null): ?>
<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?= $row_file['sertifikasi']; ?>"
class="badge badge-success" target="_blank">Lihat</a><a class="badge badge-
danger" onclick="return confirm('Apa anda yakin?')" href="hapus-
file.php?nim=<?= $nim; ?>&file=<?= $row_file['sertifikasi'];
?>&jenis_file=sertifikasi">Hapus</a>
<?php endif; ?>
</td>
</tr>

<tr>
<td>TOEFL</td>
<td class="text-center"><?=( $row_file['toefl'] != null) ? '&#9745;' : '-'; ?></td>
<td class="text-center">
<?php if($row_file['toefl'] != null): ?>
<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?= $row_file['toefl']; ?>"
class="badge badge-success" target="_blank">Lihat</a>
<a class="badge badge-danger" onclick="return confirm('Apa anda yakin?')"
href="hapus-file.php?nim=<?= $nim; ?>&file=<?= $row_file['toefl'];
?>&jenis_file=toefl">Hapus</a>
<?php endif; ?>
</td>
</tr>

<tr>
<td>KHS</td>
<td class="text-center"><?=( $row_file['khs'] != null) ? '&#9745;' : '-'; ?></td>
<td class="text-center">
<?php if($row_file['khs'] != null): ?>
<a href="lihat-file.php?nim=<?= $nim; ?>&file=<?= $row_file['khs']; ?>"
class="badge badge-success" target="_blank">Lihat</a>

```

```

<a class="badge badge-danger" onclick="return confirm('Apa anda yakin?')"
href="hapus-file.php?nim=<?= $nim; ?>&file=<?= $row_file['khs'];
?>&jenis_file=khs">Hapus</a>
<?php endif; ?>
</td>
</tr>
</table>
</div>
</div>
</div>
<?php if(date('d M Y') < date('d M Y', strtotime($row_info['waktu']))) : ?>
<?php if($row['status'] == 0) : ?>
<div class="col-md-6">
<div class="card">
<div class="card-header">Upload KRS dan Bukti Pembayaran Sebagai Syarat
Utama Matakuliah Skripsi</div>
<div class="card-body">
form method="POST" action="proses_upload.php" enctype="multipart/form-
data">
<input type="hidden" name="nama" value="<?= $nama; ?>">
<input type="hidden" name="nim" value="<?= $nim; ?>">
<div class="form-group">
<label>KRS</label>
<div class="input-group mb-3">
<div class="input-group-prepend">
<span class="input-group-text" id="inputGroupFileAddon01">Upload</span>
</div>
<div class="custom-file">
<input type="file" class="custom-file-input" name="file[]"
accept="application/pdf" id="inputGroupFile01" aria-
describedby="inputGroupFileAddon01" required=""><label class="custom-file-
label" for="inputGroupFile01">Choose file</label>

```

```

</div>
</div>
</div>
<div class="form-group">    <label>Bukti Pembayaran</label>
    <div class="input-group mb-3">
        <div class="input-group-prepend">
            <span class="input-group-text" id="inputGroupFileAddon01">Upload</span>
        </div>
        <div class="custom-file"><input type="file" class="custom-file-input"
            name="file[]" accept="application/pdf" id="inputGroupFile01" aria-
            describedby="inputGroupFileAddon01" required="">
            <label class="custom-file-label" for="inputGroupFile01">Choose file</label>
        </div>
    </div>
    <div>
        <button name="submit" class="btn btn-primary btn-sm btn-
            submit">Upload</button>
    </div>
</form>
</div>
</div>
</div>
<?php else : ?>
<div class="col-md-6">
<div class="card">
<div class="card-header">Anda sudah dikonfirmasi oleh Tata Usaha, Silahkan
Upload Persyaratan lainnya</div>
<div class="card-body">
<form method="POST" action="proses_upload.php" enctype="multipart/form-
data">
<input type="hidden" name="nama" value="<?= $nama; ?>">
<input type="hidden" name="nim" value="<?= $nim; ?>">
<div class="form-group">

```

```

<label>Usulan Skripsi</label>
<div class="input-group mb-3">
  <div class="input-group-prepend">
    <span class="input-group-text" id="inputGroupFileAddon01">Upload</span>
  </div>
  <div class="custom-file">
    <input type="file" class="custom-file-input" name="file[]"
    accept="application/pdf" id="inputGroupFile01" aria-
    describedby="inputGroupFileAddon01">
    <label class="custom-file-label" for="inputGroupFile01">Choose file</label>
  </div>
</div>
</div>
</div>
<div class="form-group">
  <label>Proposal Skripsi</label>
  <div class="input-group mb-3">
    <div class="input-group-prepend">
      <span class="input-group-text" id="inputGroupFileAddon02">Upload</span>
    </div>
    <div class="custom-file">
      <input type="file" class="custom-file-input" name="file[]"
      accept="application/pdf" id="inputGroupFile02" aria-
      describedby="inputGroupFileAddon02">
      <label class="custom-file-label" for="inputGroupFile02">Choose file</label>
    </div>
  </div>
</div>
</div>
<div class="form-group">
  <label>Laporan Penyerahan KP</label>
  <div class="input-group mb-3">

```

```

<div class="input-group-prepend">
<span class="input-group-text" id="inputGroupFileAddon03">Upload</span>
</div>
<div class="custom-file">
<input type="file" class="custom-file-input" name="file[]"
accept="application/pdf" id="inputGroupFile03" aria-
describedby="inputGroupFileAddon03">
<label class="custom-file-label" for="inputGroupFile03">Choose file</label>
</div>
</div>
</div>
<div class="form-group">
<label>Sertifikasi</label>
<div class="input-group mb-3">
<div class="input-group-prepend">
<span class="input-group-text" id="inputGroupFileAddon04">Upload</span>
</div>
<div class="custom-file">
<input type="file" class="custom-file-input" name="file[]"
accept="application/pdf" id="inputGroupFile04" aria-
describedby="inputGroupFileAddon04">
<label class="custom-file-label" for="inputGroupFile04">Choose file</label>
</div>
</div>
</div>
<div class="form-group">
<label>TOEFL</label>
<div class="input-group mb-3">
<div class="input-group-prepend">
<span class="input-group-text" id="inputGroupFileAddon05">Upload</span>
</div>
<div class="custom-file">

```



```

<input type="file" class="custom-file-input" name="file[]"
accept="application/pdf" id="inputGroupFile05" aria-
describedby="inputGroupFileAddon05">

<label class="custom-file-label" for="inputGroupFile05">Choose file</label>
</div>
</div>
</div>
<div class="form-group">
<label>KHS</label>
div class="input-group mb-3">
<div class="input-group-prepend">
<span class="input-group-text"
id="inputGroupFileAddon06">Upload</span></div>
<div class="custom-file"><input type="file" class="custom-file-input"
name="file[]" accept="application/pdf" id="inputGroupFile06" aria-
describedby="inputGroupFileAddon06">
<label class="custom-file-label" for="inputGroupFile06">Choose file</label>
</div>
</div>
</div>

<button name="submit" class="btn btn-primary btn-sm">Upload</button>

</form>
</div>
</div>
</div>
<?php endif; ?>
<?php endif; ?>
</div>
<br><br>
</div>

```

</div>

<?php require_once 'templates/footer.php'; ?>

C. Proses Upload

```
<?php
    require_once 'function/koneksi.php';
    timezone_open("Asia/Bangkok");
    session_start();
    $error = [];
    $countfiles = count($_FILES['file']['name']);
    for($i=0;$i<$countfiles;$i++){
        if($_FILES['file']['type'][$i] != 'application/pdf' &&
$_FILES['file']['name'][$i] != "){
            $error[] = 'error';
        }
    }
    if(count($error) > 0){
        $_SESSION['error'] = 'Format file harus PDF';
        header('Location: index.php');
        exit;
    }
    $nim = $_POST['nim'];
    if(!file_exists('assets/Data-Mahasiswa/'.$nim)){
        mkdir('assets/Data-Mahasiswa/'.$nim);
    }
    //query mahasiswa
    $sql_mahasiswa = "SELECT * FROM mahasiswa WHERE nim = '$nim'";
    $query_mahasiswa = mysqli_query($koneksi, $sql_mahasiswa);
    $mahasiswa = mysqli_fetch_assoc($query_mahasiswa);
    //query file
    $sql_file = "SELECT usulan_skripsi, proposal_skripsi, penyerahan_kp,
sertifikasi, toefl, khs FROM file WHERE nim = '$nim'";
```

```

$query_file = mysqli_query($koneksi, $sql_file);
$row_file = mysqli_fetch_array($query_file);
// Looping upload files
for($i=0;$i<$countfiles;$i++){
    $filename = $_FILES['file']['name'][$i];
    move_uploaded_file($_FILES['file']['tmp_name'][$i], 'assets/Data-
Mahasiswa/'.$nim.'/'.$_FILES["file"]["name"][$i]);
}
if($mahasiswa['status'] == 0){
    $krs = $_FILES['file']['name'][0];
    $bukti_pembayaran = $_FILES['file']['name'][1];
    if(mysqli_num_rows($query_file) == 0){
        $sql = "INSERT INTO file(nim, krs, bukti_pembayaran)
VALUES('$nim', '$krs', '$bukti_pembayaran')";
    }else{
        $sql = "UPDATE file SET krs = '$krs',
bukti_pembayaran =
'$bukti_pembayaran'
WHERE nim =
'$nim'";
    }
    $query = mysqli_query($koneksi, $sql);
    $isi = $mahasiswa['nama'].' mengupload persyaratan skripsi tahap
1';

    $waktu = time();
    $sql_notif = "INSERT INTO pemberitahuan(nim, isi, waktu,
status) VALUES('$nim', '$isi', '$waktu', 0)";
    mysqli_query($koneksi, $sql_notif);
    $_SESSION['alert'] = 'File berhasil diupload, tunggu konfirmasi
dari admin';
    header('Location: index.php');
}else{

```

```

for($i=0;$i<$countfiles;$i++){
    if($_FILES['file']['name'][$i]){
        $file[] = $_FILES['file']['name'][$i];
        $filename = 'Data-
Mahasiswa/'.$nim.'/'. $row_file[$i];
        unlink($filename);
    }else{
        $file[] = $row_file[$i];
    }
}
$sql = "UPDATE file SET usulan_skripsi = '$file[0]',
proposal_skripsi =
'$file[1]',
penyerahan_kp =
'$file[2]',
sertifikasi = '$file[3]',
toefl = '$file[4]',
khs = '$file[5]'
WHERE nim =
'$nim'";

$query = mysqli_query($koneksi, $sql);

$sql_status_file = "UPDATE file SET status = 2 WHERE
usulan_skripsi != " AND penyerahan_kp != " AND sertifikasi != " AND toefl
!= " AND khs != " AND nim = '$nim'";

if(mysqli_query($koneksi, $sql_status_file)){
    $isi = $mahasiswa['nama'].' mengupload persyaratan skripsi
tahap 2';

    $waktu = time();
    $sql_notif = "INSERT INTO pemberitahuan(nim, isi,
waktu, status) VALUES('$nim', '$isi', '$waktu', 0)";
    mysqli_query($koneksi, $sql_notif);
}

```

```

        $_SESSION['alert'] = 'File berhasil diupload, Silahkan upload file
lain secepatnya jika belum semua';
        header('Location: index.php');
    }
?>

```

2. Koding Program Admin

A. Login Admin

```

<?php

require_once '../function/koneksi.php';
session_start();
if(isset($_SESSION['admin'])){
    header('Location: index.php');
}
$error = "";
if(isset($_POST['submit'])){
    $username = $_POST['username'];
    $password = $_POST['password'];
    //validasi atau logika
    if(!empty(trim($username)) && !empty(trim($password))){
        $query_read = "SELECT * FROM admin WHERE
username='$username'";

        $result_read = mysqli_query($koneksi, $query_read);
        if($user = mysqli_num_rows($result_read) != 0){
            $row = mysqli_fetch_assoc($result_read);
            if(password_verify($password, $row['password'])){
                $_SESSION['admin'] = $username;
                header('Location: index.php');
            }else{
                $error = 'Username atau password salah';
            }
        }
    }
}

```

```

        }else{
            $error = 'username belum ada';
        }
    }else{
        $error = 'Data tidak boleh kosong';
    }
}

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-
to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">
    <title>Login Admin</title>
    <!-- Custom fonts for this template-->
    <link href=" ../assets/vendor/fontawesome-free/css/all.min.css" rel="stylesheet"
type="text/css">
    <link
href="https://fonts.googleapis.com/css?family=Nunito:200,200i,300,300i,400,4
00i,600,600i,700,700i,800,800i,900,900i" rel="stylesheet">
    <!-- Custom styles for this template-->
    <link href=" ../assets/css/sb-admin-2.min.css" rel="stylesheet">
</head>
<body class="bg-gradient-primary">
    <div class="container">
        <!-- Outer Row -->
        <div class="row justify-content-center mt-5">
            <div class="col-lg-6">

```

```

<div class="card o-hidden border-0 shadow-lg my-5">
  <div class="card-body p-0">
    <!-- Nested Row within Card Body -->
    <div class="row">
      <div class="col-lg">
        <div class="p-5">
          <div class="text-center">
            <h1 class="h4 text-gray-900 mb-4">Login Admin</h1>
          </div>
          <?php if($error) : ?>
            <div class="alert alert-danger">
              <?= $error; ?>
            </div>
          <?php endif; ?>
          <form class="user" method="POST" action="">
            <div class="form-group">
              <input type="text" class="form-control form-control-user"
placeholder="Username" name="username">
            </div>
            <div class="form-group">
              <input type="password" class="form-control form-control-user"
id="password" placeholder="Password" name="password">
            </div>
            <button type="submit" name="submit" class="btn btn-primary btn-
user btn-block">
              Login
            </button>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>

```

```

        </div>
    </div>
</div>
</div>
<!-- Bootstrap core JavaScript-->
<script src="../../assets/vendor/jquery/jquery.min.js"></script>
<script src="../../assets/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
<script src="../../assets/vendor/jquery-easing/jquery.easing.min.js"></script>
<script src="../../assets/js/sb-admin-2.min.js"></script>
</body>
</html>

```

B. Halaman Menu Utama Admin

```

<?php require_once '../function/koneksi.php' ;?>
<?php require_once '../templates/header.php' ;?>
<?php require_once '../templates/sidebar.php' ;?>
<?php require_once '../templates/topbar.php' ;?>
<?php

$query_mhs = mysqli_query($koneksi, "SELECT * FROM mahasiswa");
$jumlah_mhs = mysqli_num_rows($query_mhs);

$query_mhs_layak = mysqli_query($koneksi, "SELECT * FROM mahasiswa
WHERE status = '2'");
$jumlah_mhs_layak = mysqli_num_rows($query_mhs_layak);

$query_pesan = mysqli_query($koneksi, "SELECT * FROM pesan WHERE
penerima = 'Admin Tata Usaha FTI IF UNIBBA' AND status = 0");
$jumlah_pesan = mysqli_num_rows($query_pesan);

$query_notif = mysqli_query($koneksi, "SELECT * FROM pemberitahuan
WHERE status = 0");
$jumlah_notif = mysqli_num_rows($query_notif);
?>
<div class="container-fluid">

```



```

<div class="d-sm-flex align-items-center justify-content-between mb-4">
  <h1 class="h3 mb-0 text-gray-800">Dashboard</h1>
  <!-- <a href="#" class="d-none d-sm-inline-block btn btn-sm btn-primary
shadow-sm"><i class="fas fa-download fa-sm text-white-50"></i> Generate
Report</a> -->
</div>
<div class="row">
  <!-- Earnings (Monthly) Card Example -->
  <div class="col-xl-3 col-md-6 mb-4">
    <div class="card border-left-primary shadow h-100 py-2">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-primary text-uppercase mb-
1">Jumlah Mahasiswa</div>
            <div class="h5 mb-0 font-weight-bold text-gray-800"><?=
$jumlah_mhs; ?></div>
          </div>
          <div class="col-auto">
            <i class="fas fa-calendar fa-2x text-gray-300"></i>
          </div>
        </div>
      </div>
    </div>
  </div>
  <!-- Earnings (Monthly) Card Example -->
  <div class="col-xl-3 col-md-6 mb-4">
    <div class="card border-left-success shadow h-100 py-2">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">

```

```

        <div class="text-xs font-weight-bold text-success text-uppercase mb-1">Mahasiswa Sudah Layak Skripsi</div>
        <div class="h5 mb-0 font-weight-bold text-gray-800"><?=$jumlah_mhs_layak; ?></div>
    </div>
    <div class="col-auto">
        <i class="fas fa-dollar-sign fa-2x text-gray-300"></i>
    </div>
</div>
</div>
</div>
</div>
</div>
<!-- Earnings (Monthly) Card Example -->
<div class="col-xl-3 col-md-6 mb-4">
    <div class="card border-left-info shadow h-100 py-2">
        <div class="card-body">
            <div class="row no-gutters align-items-center">
                <div class="col mr-2">
                    <div class="text-xs font-weight-bold text-info text-uppercase mb-1">Mahasiswa sudah upload belum dilihat</div>
                    <div class="row no-gutters align-items-center">
                        <div class="col-auto">
                            <div class="h5 mb-0 mr-3 font-weight-bold text-gray-800"><?=$jumlah_notif; ?></div>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
    <div class="col-auto">
        <i class="fas fa-clipboard-list fa-2x text-gray-300"></i>
    </div>
</div>
</div>

```

```

</div>
</div>
<!-- Pending Requests Card Example -->
<div class="col-xl-3 col-md-6 mb-4">
  <div class="card border-left-warning shadow h-100 py-2">
    <div class="card-body">
      <div class="row no-gutters align-items-center">
        <div class="col mr-2">
          <div class="text-xs font-weight-bold text-warning text-uppercase mb-1">Pesan belum dibaca</div>
          <div class="h5 mb-0 font-weight-bold text-gray-800"><?=$jumlah_pesan; ?></div>
        </div>
        <div class="col-auto">
          <i class="fas fa-comments fa-2x text-gray-300"></i>
        </div>
      </div>
    </div>
  </div>
</div>
</div>
</div>
</div>
<?php require_once '../templates/footer.php' ;?>

```

C. Daftar Mahasiswa

```

<?php
  require_once '../function/koneksi.php';
  require_once '../templates/header.php';
  require_once '../templates/sidebar.php';
  require_once '../templates/topbar.php' ;
  $status = isset($_GET['status']) ? $_GET['status'] : "";

```

```

$halaman = 10;
$page = isset($_GET["halaman"]) ? (int)$_GET["halaman"] : 1;
$mulai = ($page > 1) ? ($page * $halaman) - $halaman : 0;
$result_halaman = mysqli_query($koneksi, "SELECT * FROM
mahasiswa");

$total = mysqli_num_rows($result_halaman);
$pages = ceil($total/$halaman);
$no = $mulai+1;
?>
<!-- Begin Page Content -->
<div class="container-fluid">
<!-- Page Heading -->
<!-- Page Heading -->
<div class="d-sm-flex align-items-center justify-content-between mb-4">
    <h1 class="h3 mb-0 text-gray-800">Daftar Mahasiswa</h1>
    <a href="tambah.php" class="btn btn-primary btn-sm">Tambah
Mahasiswa</a>
</div>
<?php if(isset($_SESSION['alert'])) : ?>
    <div class="alert alert-success">
        <?= $_SESSION['alert']; ?>
    </div>
    <?php unset($_SESSION['alert']); ?>
<?php endif; ?>
<!-- Content Row -->
<div class="card">
    <div class="card-header">
        <div class="row">
            <div class="col-md-3">
                <select class="form-control form-control-sm"
onChange="document.location.href=this.options[this.selectedIndex].value;">
                    <option value="#">-----</option>

```

```

        <option value="daftar-
mahasiswa.php">Semua Mahasiswa</option>
        <option value="daftar-
mahasiswa.php?status=0">Mahasiswa yang belum lolos persyaratan
pertama</option>
        <option value="daftar-
mahasiswa.php?status=1">Mahasiswa yang Sudah lolos persyaratan
pertama</option>
        <option value="daftar-
mahasiswa.php?status=2">Mahasiswa yang Layak Skripsi</option>
    </select>
</div>
<div class="col-md-3">
    <form action="" method="GET" class="d-none d-sm-
inline-block form-inline mr-auto ml-md-3 my-2 my-md-0 mw-100 navbar-
search">
        <div class="input-group">
            <input type="text" name="nim" class="form-control
form-control-sm border-1 small" placeholder="Cari NIM.." aria-
label="Search" aria-describedby="basic-addon2">
            <div class="input-group-append">
                <button class="btn btn-primary btn-sm"
type="submit">
                    <i class="fas fa-search fa-sm"></i>
                </button>
            </div>
        </div>
    </form>
</div>
<div class="ml-auto">
    <?php if($status != "") : ?>

```

```

        <a href="cetak.php?status=<?= $status; ?>"
target="_blank" class="btn btn-danger btn-sm"><i class="fa fa-print"></i>
Cetak</a>

        <?php else : ?>
            <a href="cetak.php" target="_blank"
class="btn btn-danger btn-sm"><i class="fa fa-print"></i> Cetak</a>
        <?php endif; ?>
    </div>
</div>
</select>
</div>
<div class="table-responsive">
    <table class="table">
        <tr>
            <th class="text-left">Nama</th>
            <th class="text-center">NIM</th>
            <th class="text-center">Status</th>
            <th class="text-center">Aksi</th>
        </tr>
        <?php
            //cari berdasarkan status
            if($status == ""){
                $sql = "SELECT * FROM mahasiswa
ORDER BY nama ASC LIMIT $mulai, $halaman";
            }elseif($status == 0){
                $sql = "SELECT * FROM mahasiswa
WHERE status = 0 ORDER BY nama ASC LIMIT $mulai, $halaman";
            }elseif($status == 1){
                $sql = "SELECT * FROM mahasiswa
WHERE status = 1 ORDER BY nama ASC LIMIT $mulai, $halaman";
            }else{

```

```

        $sql = "SELECT * FROM mahasiswa
WHERE status = 2 ORDER BY nama ASC LIMIT $mulai, $halaman";
    }

    //cari berdasarkan nim
    $cari = isset($_GET['nim']) ? $_GET['nim'] : false;
    if($cari){
        $sql = $sql = "SELECT * FROM mahasiswa
WHERE nim = '$cari'";
    }

    $query = mysqli_query($koneksi, $sql);
    ?>
    <?php if(mysqli_num_rows($query) == 0) : ?>
        <td>Tidak ada mahasiswa</td>
    <?php endif; ?>
    <?php while($row = mysqli_fetch_assoc($query)) : ?>
        <tr>
            <td class="text-left"><?= $row['nama'];
?></td>
            <td class="text-center"><?= $row['nim'];
?></td>
            <td class="text-center"><?= ($row['status']
== '2') ? 'Sudah Layak' : 'Belum Layak' ; ?></td>
            <td class="text-center">
                <?php if($status != "") : ?>
                    <a href="lihat-
berkas.php?nim=<?= $row['nim']; ?> " class="badge badge-success">Lihat
Berkas</a>
                    <a onclick="return
confirm('Apa anda yakin?')" href="update-status.php?nim=<?= $row['nim'];
?>" class="badge badge-secondary">Update Status</a>

```

```

                <?php else : ?>
                    <a href="edit.php?nim=<?=$row['nim']; ?>" class="badge badge-warning">Edit</a>
                    <a onclick="return
confirm('Apa anda yakin?')" href="hapus.php?nim=<?=$row['nim']; ?>"
class="badge badge-danger">Hapus</a>
                <?php endif; ?>
            </td>
        </tr>
    <?php endwhile; ?>
</table>
</div>
<div class="card-footer">
    <nav aria-label="...">
        <ul class="pagination pagination-sm justify-content-center">
            <!-- <li class="page-item disabled">
                <a class="page-link" href="#" tabindex="-1" aria-
disabled="true"><</a>
            </li> -->
            <?php for ($i=1; $i<=$pages; $i++) : ?>
                <?php if($page == $i) : ?>
                    <?php if(isset($_GET['status'])) : ?>
                        <li class="page-item active"><a
class="page-link" href="?status=<?=$_GET['status']; ?>&halaman=<?=$i;
?>"><?=$i; ?></a></li>
                    <?php else : ?>
                        <li class="page-item active"><a
class="page-link" href="?halaman=<?=$i; ?>"><?=$i; ?></a></li>
                    <?php endif; ?>
                <?php else : ?>
                    <?php if(isset($_GET['status'])) : ?>

```



```

        <li class="page-item"><a class="page-link"
href="?status=<?= $_GET['status']; ?>&halaman=<?= $i; ?>"><?= $i;
?></a></li>

        <?php else : ?>
            <li class="page-item"><a class="page-link"
href="?halaman=<?= $i; ?>"><?= $i; ?></a></li>
        <?php endif; ?>
    <?php endif; ?>
<?php endfor; ?>
<!-- <li class="page-item">
    <a class="page-link" href="#">></a>
</li> -->
</ul>
</nav>
</div>
</div>
</div>
</div>
<?php require_once '../templates/footer.php' ;?>

```

D. Lihat File

```

<?php

    // Store the file name into variable
    $filename = '../assets/Data-Mahasiswa/'.$_GET['nim'].'/'.$_GET['file'];
    $pdf = file_get_contents($filename);
    header('Content-Type: application/pdf');
    header('Cache-Control: public, must-revalidate, max-age=0'); // HTTP/1.1
    header('Pragma: public');
    header('Expires: Sat, 26 Jul 1997 05:00:00 GMT'); // Date in the past
    header('Last-Modified: '.gmdate('D, d M Y H:i:s'). ' GMT');
    header('Content-Length: '.strlen($pdf));
    header('Content-Disposition: inline; filename="'.basename($filename).'.pdf');
    ob_clean();

```

```
flush();
echo $pdf;
?>
```

E. Pdf

```
<?php

require_once 'dompdf/autoload.inc.php';
use Dompdf\Dompdf;
class Pdf extends Dompdf {
    public function __construct() {
        parent::__construct();
    }
}
?>
```

F. Update Status

```
<?php

require_once '../function/koneksi.php';
session_start();
$nim = $_GET['nim'];
$sql = "UPDATE mahasiswa SET status=status+1 WHERE nim = '$nim'";
$query = mysqli_query($koneksi, $sql);
$_SESSION['alert'] = "Status Mahasiswa berhasil diupdate";
header('Location: daftar-mahasiswa.php');
?>
```