

**PENGEMBANGAN SISTEM INFORMASI ARSIP SURAT
DENGAN METODE *PROTOTYPE* BERBASIS *WEBSITE*
STUDI KASUS: PRODI TEKNOLOGI INFORMASI**

TUGAS AKHIR

Diajukan Oleh:

**M. YOGA AGUSTIRANDA
NIM. 190705097
Mahasiswa Fakultas Sains dan Teknologi
Prodi Teknologi Informasi**



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI AR-RANIRY
BANDA ACEH
1444 H/2023**

LEMBAR PERSETUJUAN

**PENGEMBANGAN SISTEM INFORMASI ARSIP SURAT
DENGAN METODE *PROTOTYPE* BERBASIS *WEBSITE*
STUDI KASUS: PRODI TEKNOLOGI INFORMASI**

TUGAS AKHIR

Diajukan Kepada Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Ar-Raniry Banda Aceh
Sebagai Salah Satu Beban Studi Memperoleh Gelar Sarjana (S1)
dalam Ilmu Teknologi Informasi

Oleh:

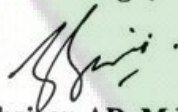
M.Yoga Agustiranda

NIM. 190705097

**Mahasiswa Fakultas Sains dan Teknologi
Program Studi Teknologi Informasi**

Disetujui untuk Dimunaqasyahkan Oleh:

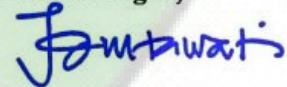
Pembimbing I,



Khairan AR, M.Kom

NIP. 198607042014031001

Pembimbing II,



Ima Dwitawati, M.B.A

NIP. 198210132014032002

Mengetahui

Ketua Program Studi Teknologi Informasi



Ima Dwitawati, M.B.A

NIP. 198210132014032002

LEMBAR PENGESAHAN

**PENGEMBANGAN SISTEM INFORMASI ARSIP SURAT
DENGAN METODE *PROTOTYPE* BERBASIS *WEBSITE*
STUDI KASUS: PRODI TEKNOLOGI INFORMASI**

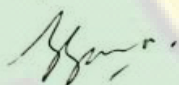
TUGAS AKHIR

Telah Diuji Oleh Panitia Ujian Munaqasah Tugas Akhir
Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh dan Dinyatakan
Lulus
Serta Diterima Sebagai Salah Satu Beban Studi Program Sarjana (S-1)
Dalam Prodi Teknologi Informasi

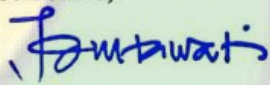
Pada Hari/Tanggal : Selasa, 10 Oktober 2023
24 Rabiul Awal 1445H

di Darussalam, Banda Aceh
Panitia Ujian Munaqasah Tugas Akhir


Ketua,


Khairan AR, M.Kom
NIP. 198607042014031001

Sekretaris,


Ima Dwitawati, M.B.A
NIP. 198210132014032002

Penguji I,

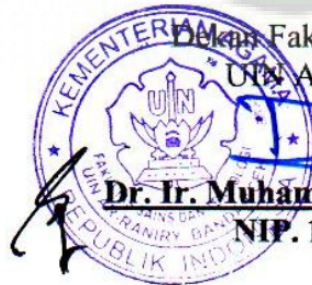

Mulkan Fadhli, S.T., M.T.
NIP. 198811282020121006

Penguji II,


Ridwah, S.ST., M.T.
NIP. 198402242019031004

Mengetahui:

Dekan Fakultas Sains dan Teknologi
UIN Ar-Raniry Banda Aceh,




Dr. Ir. Muhammad Dirhamsyah, M.T., IPU
NIP. 196210021988111001

LEMBAR PERNYATAN KEASLIAN

Yang bertanda tangan di bawah ini :

Nama : M.Yoga Agustiranda
NIM : 190705097
Program studi : Teknologi Informasi
Fakultas : Sains dan Teknologi
Judul : Pengembangan Sistem Informasi Arsip Surat Dengan metode *Prototype* Berbasis *Website* Studi Kasus : Prodi Teknologi Informasi

Dengan ini menyatakan bahwa dalam penulisan skripsi ini, saya:

1. Tidak menggunakan ide orang lain tanpa mampu mengembangkan dan mempertanggungjawabkan;
2. Tidak melakukan plagiasi terhadap naskah karya orang lain;
3. Tidak menggunakan karya orang lain tanpa menyebutkan sumber asli atau tanpa izin pemilik karya;
4. Tidak memanipulasi dan memalsukan data;
5. Mengerjakan sendiri karya ini dan mampu bertanggung jawab atas karya ini.

Bila dikemudian hari ada tuntutan dari pihak lain atas karya saya, dan telah melalui pembuktian yang dapat dipertanggungjawabkan dan ternyata memang ditemukan bukti bahwa saya telah melanggar pernyataan ini, maka saya siap dikenai sanksi berdasarkan aturan yang berlaku di Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh.

Demikian pernyataan ini saya buat dengan sesungguhnya dan tanpa paksaan dari pihak manapun.

Banda Aceh, 5 Oktober 2023

Yang Menyatakan,



M.Yoga Agustiranda

ABSTRAK

Nama : M.Yoga Agustiranda
NIM : 190705097
Program studi : Teknologi Informasi
Judul : Pengembangan Sistem Informasi Arsip Surat Dengan metode *Prototype Berbasis Website* Studi Kasus : Prodi Teknologi Informasi
Pembimbing I : Khairan AR, M.Kom.
Pembimbing II : Ima Dwitawati, M.B.A
Kata Kunci : Teknologi Informasi, Sistem Pengarsipan Surat Dengan Laravel, Website.

Program Studi Teknologi Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Ar-Raniry, Banda Aceh, saat ini menghadapi beberapa tantangan dalam pengelolaan arsip surat. Pengarsipan surat yang masih menggunakan metode konvensional telah menyebabkan sejumlah masalah, seperti kesulitan dalam mencari surat yang telah diarsipkan, penomoran surat keluar yang dilakukan secara manual memerlukan waktu lebih banyak serta proses pembuatan surat yang menyebabkan terjadi kesalahan dalam pengetikan. Selain itu, proses persetujuan dan tanda tangan surat oleh ketua program studi selama ini juga masih berlangsung secara manual dalam hal tersebut memerlukan ruang dan waktu. Untuk mengatasi permasalahan ini, tugas akhir ini bertujuan untuk mengembangkan sebuah *aplikasi pengarsipan surat berbasis website dengan menggunakan metode pengembangan aplikasi prototype*. Aplikasi ini dibangun dengan memanfaatkan framework *Laravel* versi 10, *HTML*, *CSS*, dan *JavaScript*, guna memastikan pengelolaan arsip surat yang lebih efisien. Selain itu, aplikasi ini juga memecahkan masalah saat staf ingin meminta tanda tangan dengan menambahkan fitur *QRcode*. Aplikasi di uji menggunakan metode *black box testing* untuk memastikan tingkat akurasi yang valid. Hasilnya diperoleh pada bagian halaman yang terdiri dari login, dashboard, transaksi surat, laporan surat, gallery surat, dan user profile.

Kata Kunci : Teknologi Informasi, Sistem Pengarsipan Surat Dengan Laravel, Website

KATA PENGANTAR

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ

Puji Syukur saya panjatkan kehadiran Allah SWT dan junjungan Nabi Besar Muhammad saw, karena dengan rahmat, *ridha*, pengetahuan, dan penerangan alam, sehingga saya dapat menyusun tugas akhir. Shalawat dan salam semoga tercurahkan kepada junjungan Nabi Besar Muhammad SAW beserta keluarganya, para sahabatnya dan seluruh umatnya yang selalu istiqamah hingga akhir zaman.

Tugas akhir ini menjelaskan tentang perancangan dan “pengembangan sistem informasi arsip surat dengan menggunakan metode *prototype* berbasis *website* studi kasus prodi teknologi informasi”. Tugas akhir ini merupakan salah satu persyaratan dalam menyelesaikan tugas akhir bagi mahasiswa/i Program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Ar-Raniry.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu penulis dalam penyelesaian tugas akhir, penulis juga mendapatkan banyak pengetahuan dan wawasan baru yang sangat berarti. Oleh karena itu, penulis tidak lupa mengucapkan terima kasih kepada :

1. Pertama, terimakasih untuk diri sendiri karena telah mampu berusaha keras dan berjuang sejauh ini. Mampu mengendalikan diri dari berbagai tekanan diluar keadaan dan tak pernah memutuskan menyerah sesulit apapun proses penyusunan tugas akhir ini dengan menyelesaikan sebaik dan semaksimal mungkin, ini merupakan pencapaian yang patut di banggakan untuk diri sendiri.
2. Ansari Daud selaku ayah dan Yuniar selaku ibu dan keluarga penulis, yang senantiasa mendoakan dan memberi support kepada penulis dalam menyelesaikan tugas akhir ini.
3. Ibu Ima Dwitawati, M.B.A. selaku Ketua Program Studi Teknologi Informasi.
4. Bapak Khairan Ar, M.kom dan ibu Ima Dwitawati, M.B.A. selaku pembimbing tugas akhir Prodi Teknologi Informasi UIN Ar-Raniry. Yang telah meluangkan waktunya untuk membimbing dan mengarahkan penulis dalam membuat tugas akhir ini sehingga saya dapat menyelesaikan penulisan tugas akhir.

5. Bapak Mulkan Fadhli, M.T selaku dosen wali akademik saya. Banyak ilmu, arahan dan dukungan selama saya menulis tugas akhir ini.
6. Bu Cut Ida Rahmadiana, S. Si, selaku *staff* Prodi Teknologi Informasi yang telah membantu dalam administrasi dan surat-surat untuk keperluan menyelesaikan tugas akhir.
7. Ucapan terima kasih juga disampaikan kepada sahabat-sahabat saya, khususnya kepada tim BBT (Bagi Bagi Tugas) yang terdiri dari saya sendiri, Khairun Ardiasyah S.Kom, Teuku Rovid Nina S.Kom, Maulidil Fatahsya S.Kom, dan 22 anggota lainnya. Mereka telah memberikan banyak bantuan dan masukan yang berharga dalam penelitian ini.
8. Semua pihak yang telah membantu penulis dalam menyelesaikan tugas akhir ini.

Akhir kata semoga laporan ini dapat dinilai dengan sebaik-baiknya dan menjadi dokumen yang dapat memberikan masukan dan penilaian dalam sistem perguruan tinggi serta masukan bagi dunia pendidikan. Dan mohon maaf jika terdapat kesalahan dan kekurangan dalam tugas akhir ini.

Banda Aceh, 5 Oktober 2023
Penulis,

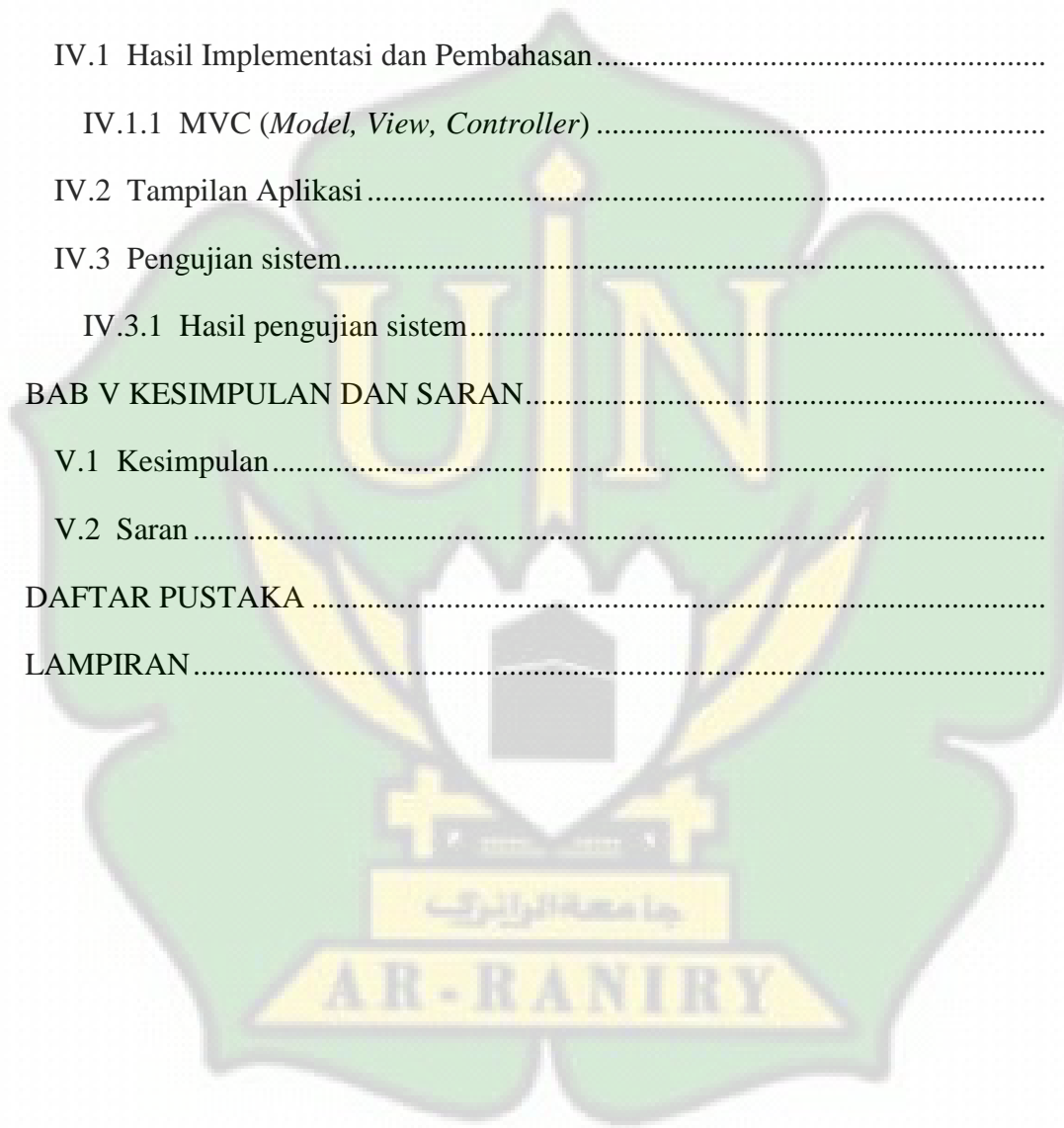
M. Yoga Agustiranda

DAFTAR ISI

LEMBAR PERSETUJUAN.....	i
LEMBAR PENGESAHAN	ii
LEMBAR PERNYATAN KEASLIAN.....	iii
ABSTRAK.....	iv
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xiii
LAMPIRAN.....	xiv
BAB I PENDAHULUAN.....	1
I.1 Latar Belakang	1
I.2 Rumusan Masalah.....	3
I.3 Tujuan Penelitian	3
I.4 Batasan Masalah	3
I.5 Manfaat Penelitian	4
BAB II TINJAUAN PUSTAKA.....	5
II.1 Landasan teori dasar	5
II.1.1 Sistem.....	5
II.1.2 <i>Website</i>	5
II.1.3 Web Browser.....	6
II.1.4 Arsip.....	6
II.1.5 Surat	6

II.2.1 <i>Prototype</i>	6
II.2.2 <i>User Interface (UI)</i>	8
II.2.3 <i>Flowchart</i>	9
II.2.4 Bahasa Pemograman	11
II.2.5 <i>Database</i>	12
II.2.6 UML (<i>Unifed Modelling Language</i>).....	14
II.2.7 Aplikasi server dan text editor	15
II.3 Penelitian Terdahulu	16
BAB III METODOLOGI PENELITIAN.....	18
III.1 Tahapan penelitian	18
III.2 Teknik Pengumpulan data	22
III.2.1 Observasi	23
III.2.2 Metode Wawancara	23
III.2.3 Studi Literatur	27
III.3 Analisis Kebutuhan Sistem	27
III.3.1 Kebutuhan Input	27
III.3.2 Kebutuhan Proses	27
III.3.3 Kebutuhan Output.....	28
III.3.4 Kebutuhan Perangkat Lunak (<i>Software</i>).....	28
III.3.5 Kebutuhan Perangkat Keras (<i>Hardware</i>)	28
III.4 Perancangan Sistem.....	29
III.4.1 <i>Use Case Diagram</i>	29
III.4.2 <i>Activity Diagram</i>	31
III.4.3 <i>Sequence Diagram</i>	42

III.4.4 <i>Entity Relationship Diagram</i>	49
III.4.5 <i>Class Diagram</i>	50
III.4.6 <i>Database</i>	51
BAB IV HASIL DAN PEMBAHASAN	56
IV.1 Hasil Implementasi dan Pembahasan	56
IV.1.1 MVC (<i>Model, View, Controller</i>)	56
IV.2 Tampilan Aplikasi	58
IV.3 Pengujian sistem	75
IV.3.1 Hasil pengujian sistem	76
BAB V KESIMPULAN DAN SARAN	85
V.1 Kesimpulan	85
V.2 Saran	85
DAFTAR PUSTAKA	87
LAMPIRAN	89

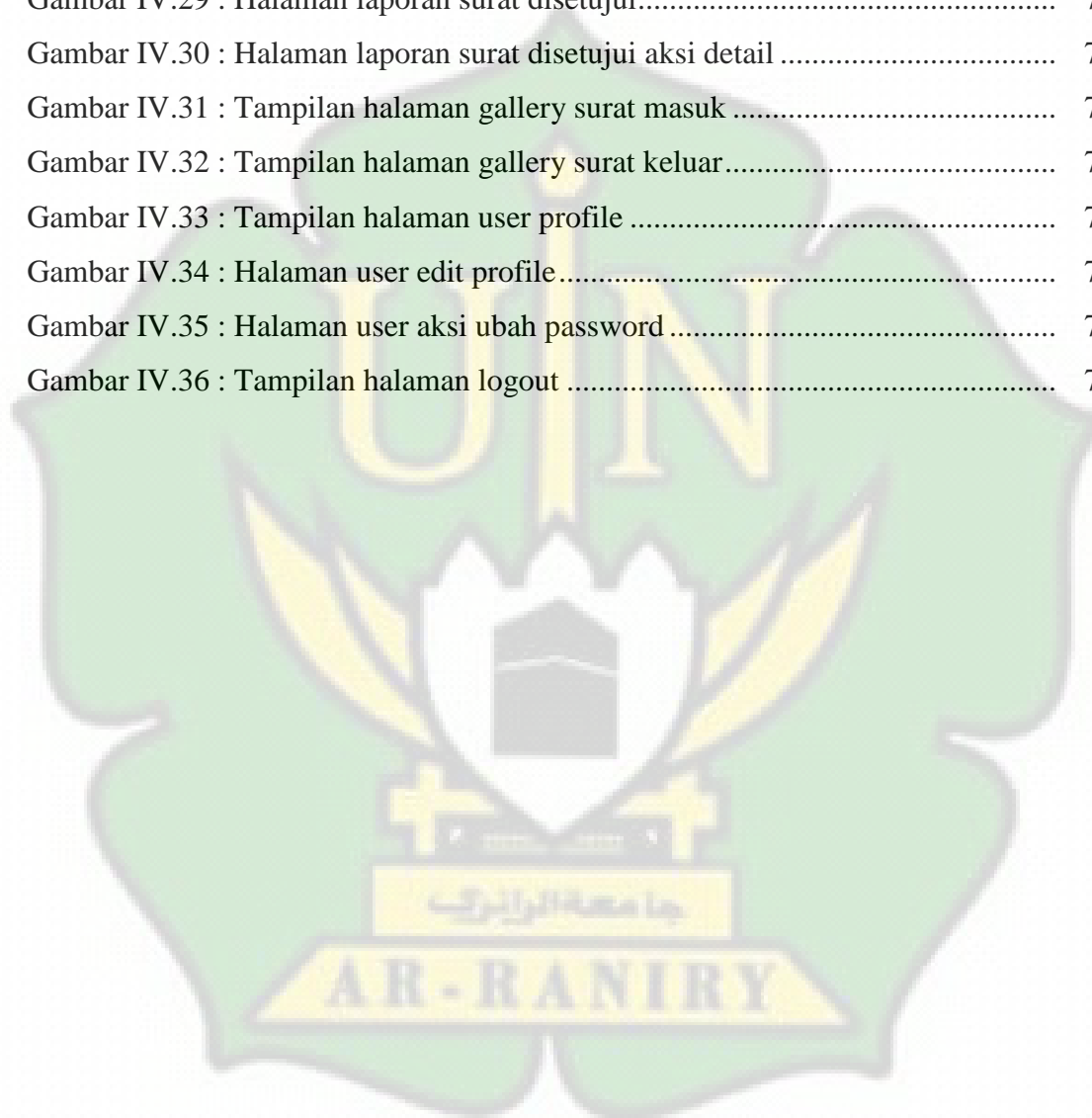


DAFTAR GAMBAR

Gambar II.1 : Tahapan Prototype.....	7
Gambar III.1 : Tahapan Penelitian	18
Gambar III.2 : Flowchart surat masuk	19
Gambar III.3 : Flowchart surat keluar.....	21
Gambar III.4 : Format catatan surat masuk.....	26
Gambar III.5 : Format catatan surat keluar	26
Gambar III.6 : Use Case Diagram.....	30
Gambar III.7 : Activity Diagram Login	31
Gambar III.8 : Activity diagram transaksi surat masuk	32
Gambar III.9 : Activity diagram transaksi surat keluar.....	33
Gambar III.10 : Activity diagram laporan surat masuk	34
Gambar III.11 : Activity diagram laporan surat keluar.....	35
Gambar III.12 : Activity diagram laporan surat disposisi.....	36
Gambar III.13 : Activity diagram laporan surat disetujui	37
Gambar III.14 : Activity diagram gallery file surat masuk.....	38
Gambar III.15 : Activity diagram gallery file surat keluar	39
Gambar III.16 : Activity diagram user profile	40
Gambar III.17 : Activity diagram ubah password.....	41
Gambar III.18 : Activity diagram logout	42
Gambar III.19 : Sequence diagram login	43
Gambar III.20 : Sequence diagram pengaturan.....	43
Gambar III.21 : Sequence diagram transaksi surat masuk	44
Gambar III.22 : Sequence diagram transaksi surat keluar	44
Gambar III.23 : Sequence diagram laporan surat masuk	45
Gambar III.24 : Sequence diagram laporan surat keluar.....	45
Gambar III.25 : Sequence diagram laporan surat disposisi.....	46
Gambar III.26 : Sequence diagram laporan surat disetujui.....	46
Gambar III.27 : Sequence diagram gallery file surat masuk.....	47

Gambar III.28 : Sequence diagram gallery file surat keluar	47
Gambar III.29 : Sequence diagram user profile	48
Gambar III.30 : Sequence diagram ubah password	48
Gambar III 31 : Entity Relationship Diagram surat	49
Gambar III.32 : Class diagram surat	50
Gambar III.33 : Rancangan database surat	51
Gambar IV.1 : Alur MVC	57
Gambar IV.2 : Halaman Login.....	58
Gambar IV.3 : Halaman Dashboard.....	58
Gambar IV.4 : Halaman pengaturan	59
Gambar IV.5 : Halaman pengaturan aksi tambah user	59
Gambar IV.6 : Halaman pengaturan aksi edit user	60
Gambar IV.7 : Halaman pengaturan aksi hapus user.....	60
Gambar IV.8 : Halaman tampilan transaksi surat masuk.....	61
Gambar IV.9 : Halaman transaksi surat masuk aksi tambah	61
Gambar IV.10 : Halaman transaksi surat masuk aksi edit	62
Gambar IV.11 : Halaman transaksi surat masuk aksi detail	62
Gambar IV.12 : Halaman transaksi surat masuk aksi hapus.....	63
Gambar IV.13 : Halaman tampilan transaksi surat keluar	63
Gambar IV.14 : Halaman transaksi surat keluar aksi tambah.....	64
Gambar IV.15 : Halaman transaksi surat keluar aksi edit.....	64
Gambar IV.16 : Halaman transaksi surat keluar aksi detail.....	65
Gambar IV.17 : Halaman transaksi surat keluar aksi hapus	65
Gambar IV.18 : Halaman laporan tampilan surat masuk	66
Gambar IV.19 : Halaman laporan surat masuk aksi filter.....	66
Gambar IV.20 : Halaman laporan surat masuk aksi disposisi	67
Gambar IV.21 : Halaman laporan surat masuk aksi detail.....	67
Gambar IV.22 : Tampilan halaman laporan surat keluar.....	68
Gambar IV.23 : Halaman laporan surat keluar aksi filter	68
Gambar IV.24 : Halaman laporan surat keluar aksi setuju surat	69

Gambar IV.25 : Halaman laporan surat keluar aksi view pdf.....	69
Gambar IV.26 : Halaman laporan surat keluar aksi detail	70
Gambar IV.27 : Tampilan halaman laporan surat disposisi.....	70
Gambar IV.28 : Halaman laporan surat disposisi aksi detail	71
Gambar IV.29 : Halaman laporan surat disetujui.....	71
Gambar IV.30 : Halaman laporan surat disetujui aksi detail	72
Gambar IV.31 : Tampilan halaman gallery surat masuk	72
Gambar IV.32 : Tampilan halaman gallery surat keluar.....	73
Gambar IV.33 : Tampilan halaman user profile	73
Gambar IV.34 : Halaman user edit profile.....	74
Gambar IV.35 : Halaman user aksi ubah password	74
Gambar IV.36 : Tampilan halaman logout	75

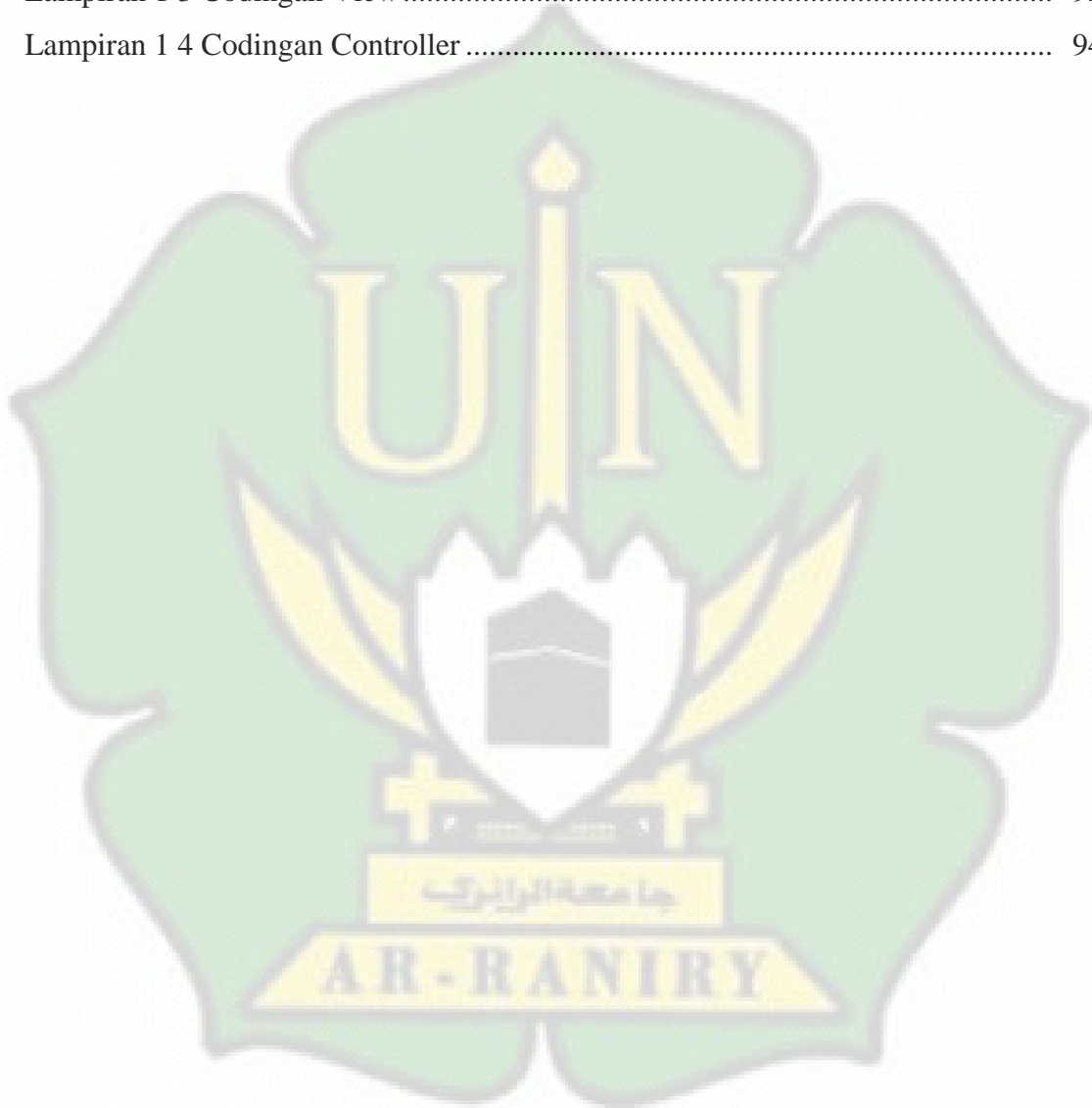


DAFTAR TABEL

Tabel II.1 : Simbol-Simbol pada Flowchart.....	9
Tabel II.2 : Notasi ERD	13
Tabel II.3 : Notasi Unified Modeling Language.....	14
Tabel III.1 : Tabel user.....	52
Tabel III.2 : Tabel surat masuk	52
Tabel III.3 : Tabel surat keluar.....	53
Tabel III.4 : Tabel mahasiswa.....	54
Tabel III.5 : Tabel dosen	54
Tabel III.6 : Tabel surat keterangan	54
Tabel III.7 : Tabel surat permohonan.....	55
Tabel III.8 : Tabel surat pemberitahuan.....	55
Tabel IV.1 : Pengujian halaman login.....	76
Tabel IV.2 : Pengujian halaman dashboard	76
Tabel IV.3 : Pengujian halaman pengaturan role.....	77
Tabel IV.4 : Pengujian halaman transaksi surat masuk	78
Tabel IV.5 : Pengujian halaman transaksi surat keluar.....	79
Tabel IV.6 : Pengujian halaman laporan surat masuk.....	80
Tabel IV.7 : Pengujian halaman laporan surat keluar	81
Tabel IV.8 : Pengujian halaman laporan surat disposisi	82
Tabel IV.9 : Pengujian halaman laporan surat disetujui	82
Tabel IV.10 : Pengujian halaman gallery surat masuk	83
Tabel IV.11 : Pengujian halaman gallery surat keluar	83
Tabel IV.12 : Pengujian halaman <i>user profile</i>	83
Tabel IV.13 : Pengujian halaman <i>logout</i>	84

LAMPIRAN

Lampiran 1 1 Codingan <i>file Routes</i> (web)	89
Lampiran 1 2 Codingan Model	90
Lampiran 1 3 Codingan View	92
Lampiran 1 4 Codingan Controller	94



BAB I

PENDAHULUAN

I.1 Latar Belakang

Surat merupakan instrumen yang digunakan oleh suatu instansi untuk berkomunikasi, sehingga penanganan dan pengelolaan surat masuk dan keluar harus di proses dengan cepat dan lebih efisien. Pertukaran informasi melalui surat di anggap berhasil apabila surat tersebut jelas mengenai isi dan tujuannya sehingga tidak menimbulkan kesalahpahaman antara penerima atau pengguna. Arsip surat merupakan sebuah kumpulan-kumpulan surat atau dokumen-dokumen yang disimpan secara teratur dan sistematis sebagai bukti aktivitas organisasi atau perusahaan. Arsip surat harus dilakukan untuk beberapa alasan di antaranya, untuk menjaga dokumentasi, kepatuhan hukum, dan memudahkan pengambilan keputusan. Pengelolaan arsip surat melibatkan pengumpulan, klasifikasi, penyimpanan, pemeliharaan dan penggunaan untuk keperluan administrasi dan kepentingan jangka panjang sebuah instansi atau organisasi.

Berdasarkan pengamatan dan observasi yang dilakukan penulis pada bagian staf prodi teknologi informasi, penulis menemukan bahwa masih terdapat banyak ketidakefisienan pada bagian pengarsipan surat baik itu surat masuk ataupun surat keluar yang masih dilakukan dengan cara konvensional, yaitu pendataannya masih menggunakan sebuah buku agenda, yang dapat menyebabkan sulitnya mencari surat yang sebelumnya sudah di arsipkan dikarenakan tidak adanya pendataan di mana surat tersebut disimpan. Terkadang pada penomoran surat keluar juga keliru nomor berapa yang sudah di keluarkan karena memakai pendataan yang secara konvensional.

Masalah lain yang juga ditemukan adalah ketika bagian *staff* ingin meminta tanda tangan dari ketua prodi, para bagian *staff* harus menitipkan lembar tersebut di mejanya. Hal ini dapat menimbulkan kendala dan keterlambatan dalam proses persetujuan dan tanda tangan. Dan untuk pembuatan surat yang diminta dari pihak-pihak, baik itu mahasiswa atau instansi eksternal, juga menghadapi kendala. Saat ini,

staff prodi harus membuka *file template Word* terlebih dahulu untuk menghasilkan surat yang sesuai dengan format yang ditentukan. Hal ini dapat memakan waktu dan menghambat efisiensi proses pembuatan surat.

Pada penelitian (Lubis & Samsudin, 2022) juga ditemukan permasalahan yang serupa dimana surat yang masuk dan keluar masih diarsipkan secara manual yaitu dengan menuliskannya ke buku besar dan menyimpannya dalam map yang kemudian dimasukkan ke dalam lemari. Hal ini membutuhkan banyak waktu dan usaha yang cukup lama dan kemungkinan terjadinya surat hilang dan kerusakan pada surat tersebut.

Prodi teknologi informasi membutuhkan sistem pengarsipan surat untuk mengoptimal administrasi persuratannya. Menurut (Saifudin & Setiaji, 2019), sistem informasi pengarsipan surat ini bertujuan untuk mengubah cara penyimpanan surat dengan cara mengurangi penggunaan kertas (*paperless*).

Berdasarkan permasalahan tersebut, penulis memiliki niat untuk melakukan penelitian dengan judul “Pengembangan Sistem Informasi Arsip Surat dengan Metode *Prototype* Berbasis Website Studi Kasus : Prodi Teknologi Informasi”. Penelitian ini bertujuan untuk mengembangkan sistem informasi arsip surat yang mengintegrasikan solusi-solusi yang telah disebutkan sebelumnya. Dalam penelitian ini, penulis ingin mengimplementasikan metode *prototype* untuk mengembangkan sistem informasi arsip surat yang berbasis *website*. Metode *prototype* akan memungkinkan penulis untuk membangun versi awal dari sistem, menguji dan mengevaluasi fungsionalitasnya, serta memperoleh umpan balik dari pengguna terkait fitur-fitur yang dibutuhkan dan perbaikan yang mungkin diperlukan.

Sistem informasi pengarsipan surat yang dikembangkan mencakup beberapa fitur seperti: pengumpulan surat secara elektronik, klasifikasi surat berdasarkan kriteria tertentu, penyimpanan yang aman dan terstruktur dari surat, pemeliharaan dan penyimpanan arsip, serta kemudahan akses dan pencarian surat. Fitur-fitur tersebut akan berkontribusi untuk meningkatkan efisiensi dalam pemrosesan surat masuk dan keluar, pemrosesan persetujuan/tanda tangan, dan pembuatan surat menggunakan *template* yang telah disiapkan sebelumnya.

I.2 Rumusan Masalah

Berdasarkan penjelasan yang telah di uraikan diatas, rumusan masalah dapat dirincikan sebagai berikut:

1. Bagaimana menerapkan metode *prototype* pada rancang bangun Sistem Informasi pengarsipan surat yang efisien berbasis *website* pada prodi teknologi informasi?''.
2. Bagaimana sistem ini dapat meningkatkan efisiensi dalam pengelolaan surat masuk dan keluar, serta proses persetujuan/tanda tangan surat?
3. Bagaimana sistem ini dapat memudahkan pembuatan surat dengan menggunakan template yang telah disiapkan sebelumnya?

I.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah mengembangkan Sistem Informasi Arsip Surat berbasis *website* dengan menggunakan metode *prototype* dan bantuan *framework laravel* untuk Prodi Teknologi Informasi. Tujuan ini mencakup beberapa aspek, yaitu:

1. Meningkatkan efisiensi dalam pengelolaan dan pemrosesan surat masuk dan keluar.
2. Membuat proses pembuatan surat lebih efisien dengan menggunakan template yang telah disiapkan sebelumnya.
3. Mempermudah proses persetujuan/tanda tangan surat.

I.4 Batasan Masalah

Agar pembahasan sesuai dengan tema judul yang telah ditentukan dan latar belakang yang telah jelaskan, maka penulis membatasi masalah yang mencakup:

1. Penelitian ini hanya berfokus pada pengembangan Sistem Informasi Arsip Surat untuk Prodi Teknologi Informasi. dengan bahasa pemograman web yaitu Html, Css, *javascript* dan *framework laravel*. Bantuan *software* yang akan digunakan ialah *Visual Studio Code* sebagai IDE (*Integrated Development Environment*) teks editornya dan *XAMPP Control Panel* sebagai *server* databasenya.
2. Penelitian ini akan memanfaatkan metode *prototype* sebagai pendekatan pengembangan.
3. Hanya surat masuk dan keluar yang akan dikelola dan diarsipkan oleh sistem ini.

I.5 Manfaat Penelitian

Berdasarkan latar belakang, rumusan masalah, dan tujuan penelitian yang telah penulis uraikan, adapun manfaat dari penelitian ini adalah:

1. Bagi penulis, diharapkan penelitian ini dapat menambah wawasan serta pengalaman penulis dalam membangun sebuah sistem dengan menerapkan metode *prototype*.
2. Bagi instansi, diharapkan hasil dari penelitian ini mudah mudahan dapat membantu staf prodi dalam pengelolaan surat.



BAB II

TINJAUAN PUSTAKA

II.1 Landasan teori dasar

Adapun landasan teori dasar yang mendukung dalam penelitian ini adalah sebagai berikut :

II.1.1 Sistem

Komponen yang bekerja untuk mencapai suatu tujuan di sebut dengan sistem. Sistem memiliki fungsi utamanya ialah untuk menerima masukan(input), melakukan pengolahan(proses), dan menghasilkan keluaran(output) sesuai dengan tujuan yang ingin di capai.

II.1.2 Website

Website bisa dimaksud sebagai sebuah domain atau halaman yang berisikan data atau informasi yang digunakan oleh perusahaan atau institusi untuk memperkenalkan produk dan layanannya, disediakan lewat jalan internet sehingga bisa diakses di seluruh dunia. Secara garis besar menurut (Saifudin & Setiaji, 2019) mengelompokkan web menjadi beberapa bagian yaitu:

1. *Website Statis*

Website statis ialah halaman web yang mempunyai halaman yang tetap dan tidak pernah berubah.

2. *Website Dinamis*

Website dinamis ialah sebuah halaman web yang dirancang supaya bisa melakukan perubahan isi dari halaman secara terstruktural pada web tersebut.

3. *Website Interaktif*

Website Interaktif merupakan web yang di mana pengguna bisa berinteraksi dan beradu pendapat mengenai ide dari masing-masing pengguna.

II.1.3 Web Browser

Untuk dapat mengakses sebuah halaman website diperlukan sebuah *software* yang disebut *browser*. *Browser* ialah sebuah *software* yang digunakan untuk menjalankan program atau *script* pada sebuah *website*. Contoh *browser* adalah *google chrome*, *safari*, *Microsoft edge*, *Mozilla FireFox* dan lain-lain nya.

II.1.4 Arsip

Arsip adalah data yang disimpan dalam bermacam bentuk baik itu rekaman ataupun tulisan, supaya disaat dibutuhkan dapat ditemui kembali. Arsip memuat informasi yang disimpan dalam bentuk tulisan atau rekaman sehingga bisa digunakan sebagai acuan dalam pengambilan keputusan. Arsip merupakan suatu aksi untuk menempatkan dokumen ke dalam tempat penyimpanan yang teratur dan sesuai dengan peraturan yang telah ditetapkan sebelumnya. Tujuannya adalah agar setiap dokumen dapat dengan mudah ditemukan kembali dan cepat pada saat dibutuhkan.

II.1.5 Surat

Surat ialah sarana komunikasi yang digunakan oleh pihak tertentu guna menyampaikan informasi tertulis dari satu pihak kepada pihak lainnya. Surat memegang peranan penting sebagai suatu komunikasi yang sangat penting bagi sebuah lembaga, baik dalam berhubungan dengan pihak luar ataupun dalam organisasi itu sendiri. Pada prinsipnya, dalam satu organisasi, surat terbagi menjadi dua jenis, yaitu surat yang diterima (surat masuk) dan surat yang dikirimkan (surat keluar).

II.2 Teori pengembangan sistem

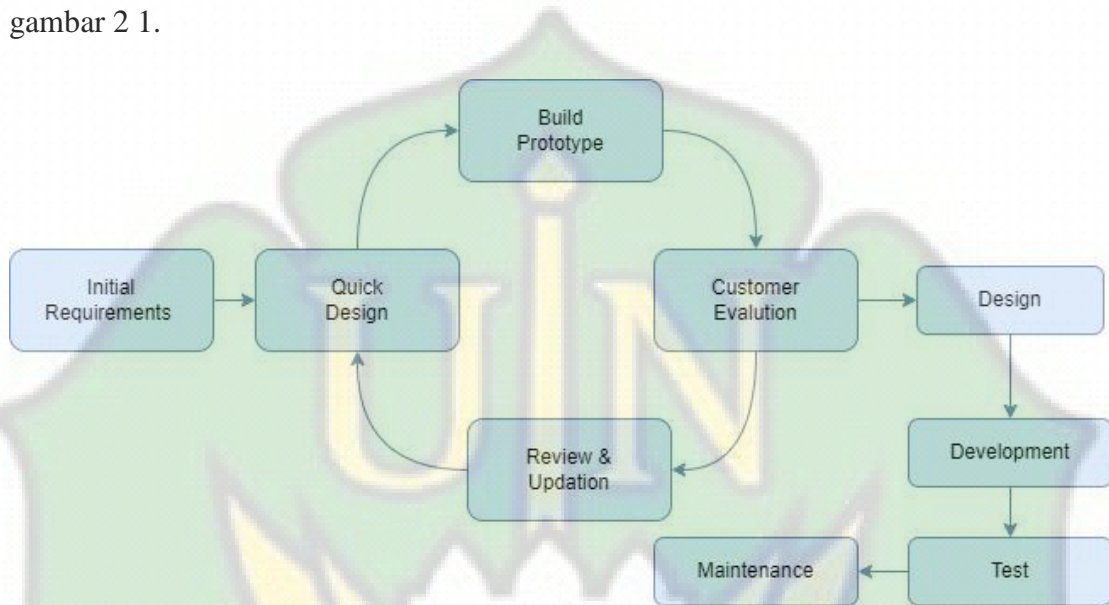
Adapun beberapa teori pendukung dalam pengembangan sistem penelitian ini adalah sebagai berikut :

II.2.1 Prototype

Menurut (Yoko et al., 2019), Model *prototype* merupakan sebuah metode yang diwajibkan pengembang *software* atau perangkat lunak untuk membuat sebuah model dari aplikasi atau yang biasa di sebut *mockup*, perilaku ini cocok pada saat kondisi di mana pada saat sipengguna tidak bisa memberikan informasi secara detail tentang

kebutuhan yang sesuai dengan keinginannya. Model metode prototype menghasilkan sebuah *mockup* yang menjadi acuan untuk desain yang akan digunakan dalam pelatihan, penilaian desain, presentasi, promosi, atau tujuan lainnya.

Metode *prototyping* terdiri dari delapan 8 tahapan sebagaimana yang tercantum pada gambar 2 1.



Gambar II.1 : Tahapan *Prototype*

- *Initial Requirements* / Pengumpulan kebutuhan pengguna dan *developer* bekerja sama untuk mendefinisikan keseluruhan dari software yang akan di kembangkan, menentukan kebutuhan dari perangkat lunak, dan menggambarkan garis besar sistem yang akan dibuat.
- *Design*
Kebutuhan yang telah didapatkan berdasarkan diskusi dari langkah awal akan dibuatkan gambaran arsitektur sistem untuk menjelaskan alur proses sebagai dasar dalam pembuatan *prototyping*.
- *Membangun Prototyping*
Proses pembuatan *prototype* melibatkan pembuatan desain sementara yang difokuskan pada presentasi kepada pelanggan, contohnya membuat format *input* dan format *output*.

- *Customer Evaluation*

Evaluasi *prototyping* akan dilakukan oleh pelanggan untuk mengetahui apakah *prototyping* yang telah dibangun sesuai dengan harapan mereka atau belum. Jika *prototyping* telah selesai dan sesuai, maka akan meneruskan langkah selanjutnya. Namun, jika belum sesuai, *prototyping* akan direvisi terlebih dahulu dengan mengulang langkah yang sebelumnya.

- *Review and Updation*

Setelah dilakukan evaluasi oleh pengguna, hasilnya direview dan jika terdapat beberapa kekurangan berdasarkan hasil evaluasi tersebut, akan segera dilakukan pembaruan yang digunakan sebagai dasar untuk memperbaiki desain sistem.

- *Development / pengkodean sistem*

Pada tahap ini gambaran dari *prototype* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yaitu dengan melakukan pengkodean program program yang sesuai.

- *Testing*

Setelah perangkat lunak sudah siap pakai kemudian perangkat lunak harus melewati proses pengujian pada sistem. Pengujian ini dilakukan dengan *White Box, Black Box*.

- *Maintenance System*

Sistem yang sudah di pakai akan dilakukan pemulihan dan perawatan selama digunakan oleh pengguna agar menjamin perangkat lunak akan terus bisa berjalan dengan baik.

II.2.2 User Interface (UI)



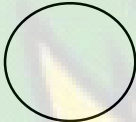

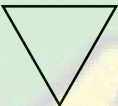
User interface ialah ilmu yang mempelajari tentang tata letak grafis suatu tampilan dari *interface* sebuah sistem. Pengelompokan yang termasuk UI yaitu seperti tombol, gambar, teks, dan semua item yang berinteraksi dengan penggunanya. Adanya UI dalam pengembangan produk digital dapat lebih memudahkan para *developer* dalam merancang produknya sebelum dilakukan pengkodean dan di publikasikan ke user (Ariawan et al., 2020).

II.2.3 Flowchart

Menurut (Codingstudio.id, 2023) *Flowchart* merupakan representasi untuk menunjukkan langkah dan keputusan pada proses sebuah program dari sebuah prosedur atau algoritma untuk menyelesaikan suatu masalah, Menggunakan *flowchart* dapat memudahkan *user* melakukan pengecekan bagian yang sudah terlewat dalam analisa masalah. *Flowchart* memudahkan dalam memahami tahapan logika yang panjang dan rumit. Simbol simbol *flowchart* dapat di dilihat pada tabel 2.1.

Tabel II.1 : Simbol-Simbol pada *Flowchart*

No	Simbol	Nama	Keterangan
1		Terminal	Simbol untuk memulai dan mengakhiri dalam sebuah proses program.
2		<i>Input</i> dan <i>output</i>	Simbol yang menyatakan pemasukan data dan pengeluaran data (input/output).
3		Proses komputer	Fungsi pemrosesan yang dilakukan oleh komputer untuk menghasilkan data informasi.
4		<i>Flow</i> / Aras dokumen	Flow arah normal adalah kekanan dan kebawah.
5		Keputusan	Simbol keputusan digunakan dalam diagram alir program komputer untuk menunjukkan dua kemungkinan yang mungkin diambil, yaitu <i>true</i> dan <i>false</i>

6		Operasi Manual	Simbol yang menyatakan proses dilakukan secara manual.
7		Penyimpanan	Simbol yang menyatakan akses secara langsung perangkat penyimpanan.
8		Off-page reference	Simbol ini digunakan untuk menghubungkan bagan alur di halaman yang berbeda dalam diagram alur program komputer
9		Penghubung dalam sebuah halaman	Simbol ini digunakan untuk menghubungkan bagan alur di halaman yang sama dalam diagram alur program komputer
10		Dokumen	Simbol yang menyatakan bahwasanya input berasal dari dokumen fisik, atau output yang perlu di cetak.
11		Arsip	Arsip dokumen ini menunjukkan bahwa data akan di simpan secara sementara.

II.2.4 Bahasa Pemograman

a. *Hypertext Markup Language (HTML)*

Web browser seperti *google chrome* dan *browser* lainnya akan membaca dokumen HTML yang merupakan bahasa pengformatan teks yang berisi tag-tag yang digunakan untuk membuat kerangka web *pages* pada sebuah jaringan komputer yang dikenal sebagai *World Wide Web (www)* atau biasa disebut dengan *web* saja.

b. *Cascading Style Sheet (CSS)*

Menurut (I. P. Sari et al., 2022), *Cascading Style Sheet (CSS)* merupakan bahasa komputer yang berisikan kode untuk menambahkan *design* yang saling terhubung guna mengatur format tampilan pada suatu halaman dokumen HTML.

c. *JavaScript (JS)*

Menurut (Rhomdani, 2022), *JavaScript* merupakan *script* yang paling populer saat ini dan banyak digunakan dalam pemrograman web pada saat ini. Adanya *JavaScript* sangat berguna bagi sebuah web akan lebih menjadi interaktif dan tampil lebih menawan dengan sebuah animasi.

d. *PHP Hypertext Preprocessor (PHP)*

Menurut (Suryadi, 2019), *PHP Hypertext Preprocessor* ialah bahasa pemograman *back end* yang banyak digunakan dalam pembuatan *website*, bahasa PHP merupakan bahasa pemograman yang berbentuk skrip yang bekerja di dalam server web.

e. *Framework Laravel*

Laravel merupakan sebuah *framework* aplikasi web berbasis bahasa pemograman PHP yang memiliki fitur dan alat yang dapat membantu dalam pengembangan aplikasi dengan mudah. Laravel di desain dengan konsep model-view-controller (MVC) yang dapat memisahkan logika aplikasi dari tampilan pengguna.

II.2.5 Database

a. Mysql

MySQL merupakan sebuah perangkat lunak yang sistem manajemennya berbasis data *Structure Query Language* (SQL) atau *Database Management System* (DBMS) yang *multi thread* dan *multi user*, dengan sekitar kurang lebih enam juta instalasi di seluruh dunia.

b. ERD (Entity-Relation Diagram)

Entity-Relationship Diagram (ERD) merupakan sebuah cara yang digunakan untuk menjelaskan data atau objek-objek yang dibuat berdasarkan dari entitas serta relasi antar entitas tersebut menggunakan beberapa notasi. Diagram yang berbentuk notasi grafis yang berada dalam pembuatan *database* yang membuat *database* lebih jelas. ERD berfungsi sebagai alat bantu dalam proses pembuatan database dan memberikan gambaran bagaimana database yang dibuat akan bekerja (khoulah afifah el al., 2022).

ERD dibagi dalam 3 elemen, yaitu sebagai berikut:

1) Entitas

Entitas adalah objek yang ada dalam suatu *database*, entitas dapat berupa manusia, benda, atau kondisi mengenai sebuah data. Simbol dari entitas ialah persegi panjang.

2) Atribut

Penggunaan atribut untuk menggambarkan informasi yang terdapat dalam entitas. Atribut biasanya terletak dalam table entitas atau terpisah dari tabel. Simbol atribut berbentuk elips.

3) Relasi



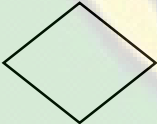

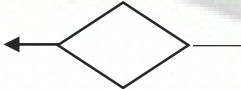
Penggunaan relasi untuk dapat menghubungkan antara entitas entitas dua atau lebih.

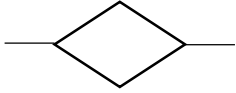
Ada beberapa macam relasi yang dimiliki oleh ERD, yaitu:

- *One to One* : Relasi dengan hubungan dimana setiap satu baris data pada tabel pertama hanya terhubung dengan satu baris pada data tabel kedua.

- *One to Many* : Relasi dengan hubungan dimana setiap satu baris data pada tabel pertama berhubungan lebih dari satu baris pada data tabel kedua.
- *Many to Many* : Relasi dengan hubungan dimana setiap lebih dari satu baris data pada tabel pertama berhubungan dengan lebih dari satu baris pada data tabel kedua.

Tabel II.2 : Notasi ERD

Notasi	Nama	Keterangan
	Entitas	Mewakili suatu objek yang berwujud nyata dan dapat dibedakan dengan objek lainnya.
	Atribut / field	Property dimiliki sebuah entitas, berguna untuk mendeskripsikan karakteristik dari sebuah entitas tersebut.
	Relasi	Menunjukkan koneksi antara beberapa entitas yang berbeda.
	Relasi 1 : 1	Hubungan yang diberikan menunjukkan koneksi setiap entitas himpunan pertama dapat berelasi paling banyak satu entitas pada himpunan entitas kedua, demikian sebaliknya.
	Relasi 1 : N	Hubungan yang diberikan menunjukkan koneksi setiap entitas himpunan pertama dapat berelasi banyak entitas pada himpunan entitas kedua, tetapi tidak sebaliknya.

	Relasi N : N	Hubungan yang diberikan menunjukkan koneksi pada entitas himpunan pertama dapat berelasi banyak entitas pada himpunan entitas kedua.
---	--------------	--

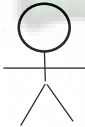

II.2.6 UML (*Unified Modelling Language*)

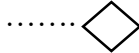
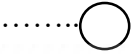







UML (*Unified Modeling Language*) ialah bahasa yang standar pada perancangan dan pendokumentasian dari sistem. Bahasa pemodelan visual UML digunakan untuk berkomunikasi dan menggunakan diagram untuk memodelkan sistem sekaligus teks-teks pendukung. UML digunakan sebagai pemodelan (Jantce TJ Sitinjak et al., 2020).

Adapun diagram UML yang digunakan adalah sebagai berikut:

1. *Activity* diagram untuk menggambarkan proses urutan aktifitas dalam sebuah sistem yang akan dibuat.
2. *Use case* diagram menggambarkan interaksi antara sistem dengan user (pengguna). Juga merupakan tahap awal dalam proses perancangan sistem.
3. *Class* diagram menggambarkan sekumpulan kelas dalam sebuah sistem, interface dan kolaborasi terhubungnya antara satu dengan yang lain antara masing-masing bagian.
4. *Sequence* diagram ialah interaksi proses yang terjadi didalam perangkat lunak yang diatur dalam urutannya.

Tabel II.3 : Notasi *Unified Modeling Language*

Simbol	Nama	Keterangan
	<i>Actor</i>	Menunjukkan peran orang, alat ketika berhubungan dengan <i>use case</i>
	<i>Use case</i>	Abstraksi dari penghubung antar actor dengan <i>use case</i>

	<i>Generalization</i>	Menyatakan spesialisasi actor agar dapat berpartisipasi dengan <i>use case</i>
	<i>Interface</i>	Kumpulan operasi tanpa implementasi dari sebuah class
	<i>Relization</i>	Hubungan elemen yang bertanda panah akan merelasikan ke elemen yang ada bagian tanda panah
	<i>Interaction</i>	Digunakan untuk alir pesan antar objek
	<i>Association</i>	Abstraksi dari penghubung antar actor dengan <i>use case</i>
	<i>Dependency</i>	Hubungan yang memperlihatkan perubahan pada suatu elemen yang memberi pengaruh pada elemen lainnya
	<i>Note</i>	Elemen fisik pada saat program dijalankan mencerminkan sumber daya yang berkomputasi
	<i>Class</i>	Kumpulan objek-objek yang mempunyai atribut sekaligus operasi
	<i>Package</i>	Sebuah tempat yang berguna saat mengelompokkan elemen dari suatu sistem yang bangun.

II.2.7 Aplikasi server dan text editor

a. XAMPP

XAMPP adalah sebuah *software* yang berpaket didalamnya mengandung *Web Apache Server*, *database MySQL*, *PhpMyAdmin*, dan *interpreter PHP*” dan lain-lainnya. Dengan menggunakan *xampp* anda tidak perlu menginstall aplikasi tersebut satu persatu (Putra, 2021).

b. *Visual Studio Code*

Visual Studio Code adalah sebuah IDE (*Integrated Development Environment*) *software text editor* yang bermanfaat bagi kalangan, khususnya bagi seorang *developer*. Dikarenakan *text editor* ini mendukung banyak ekstensi atau plugin yang sangat berguna bagi *developer* dalam membuat sebuah program.

II.3 Penelitian Terdahulu

Penelitian terdahulu merupakan serangkaian sumber-sumber penelitian terdahulu yang di kumpulkan dan dimuat untuk mendukung penelitian serta sebagai acuan penulis dalam meneliti dan menghindari kesamaan hasil dari penelitian yang di lakukan. Selain itu, juga sebagai sumber inspirasi dalam penelitian penelitian lain yang ditulis agar dapat menghasilkan suatu penemuan yang baru, atau diperbarui dengan berbagai metode dan cara yang ilmiah.

Penelitian yang dilakukan oleh (Putri et al., 2020) dalam penelitiannya yang berjudul “Implementasi Metode *Prototyping* pada Perancangan Aplikasi *Electronic Ticket* (E-Ticket) berbasis Android”. Penelitian ini menghasilkan sebuah sistem aplikasi e-tiket yang memudahkan sipembeli tiket dalam melakukan pemesanan tiket dan dapat membantu pengguna dalam pencarian informasi tentang jadwal keberangkatan dan juga mengetahui harga tiket secara langsung tanpa harus ke terminal terlebih dahulu.

Pengembangan yang dilakukan oleh (Ardiyansah et al., 2021) dalam penelitiannya yang berjudul “Implementasi Metode *Prototype* pada sistem Informasi Pengadaan barang cetakan Berbasis web”. Penelitian ini menghasilkan sebuah sistem pengadaan barang cetakan yang memudahkan dalam memonitoring data pengadaan barang cetakan serta lebih aman dan terciptanya efisiensi.

Pengembangan yang dilakukan oleh (Nuryani, 2021) dalam penelitiannya yang berjudul “Pengembangan Aplikasi *Mobile Booking Online* perawatan gigi dengan metode *Prototype* studi kasus di Klinik Gigi Budiono, Drg. Kota Bandung” Penelitian ini menghasilkan sebuah aplikasi berbasis android yang membantu klinik gigi Budiono dalam manajemen antrian pasien yang sebelumnya kurang efisien dikarenakan

pelayanan yang terjadi di klinik tersebut meliputi pendaftaran di meja resepsionis, pasien yang berkonsultasi hingga melakukan perjanjian untuk bertemu dokter semua dilakukan booking melalui telepon. Dengan adanya aplikasi ini, sekarang pasien dapat melakukan interaksi melalui aplikasi untuk pendaftaran tanpa harus telepon dan antri ditempat klinik dengan waktu antrian yang lama.

Pengembangan yang dilakukan oleh (Yoko et al., 2019) yang judul penelitiannya “Penerapan Metode Prototype dalam Perancangan Aplikasi SIPINJAM Berbasis Website pada Credit Union Canaga Antutn”. Pada penelitian ini menghasilkan sistem informasi berbasis web yang dapat membantu meningkatkan performa dalam pengelolaan data simpan dan pinjam seperti pengelolaan data simpanan, anggota, pinjaman seperti angsuran dan lain sebagainya. Semua data transaksi bisnis terakam dan tersimpan secara terstruktur menggunakan *database* sehingga dapat digunakan kembali serta mudah dalam melakukan pencarian data.

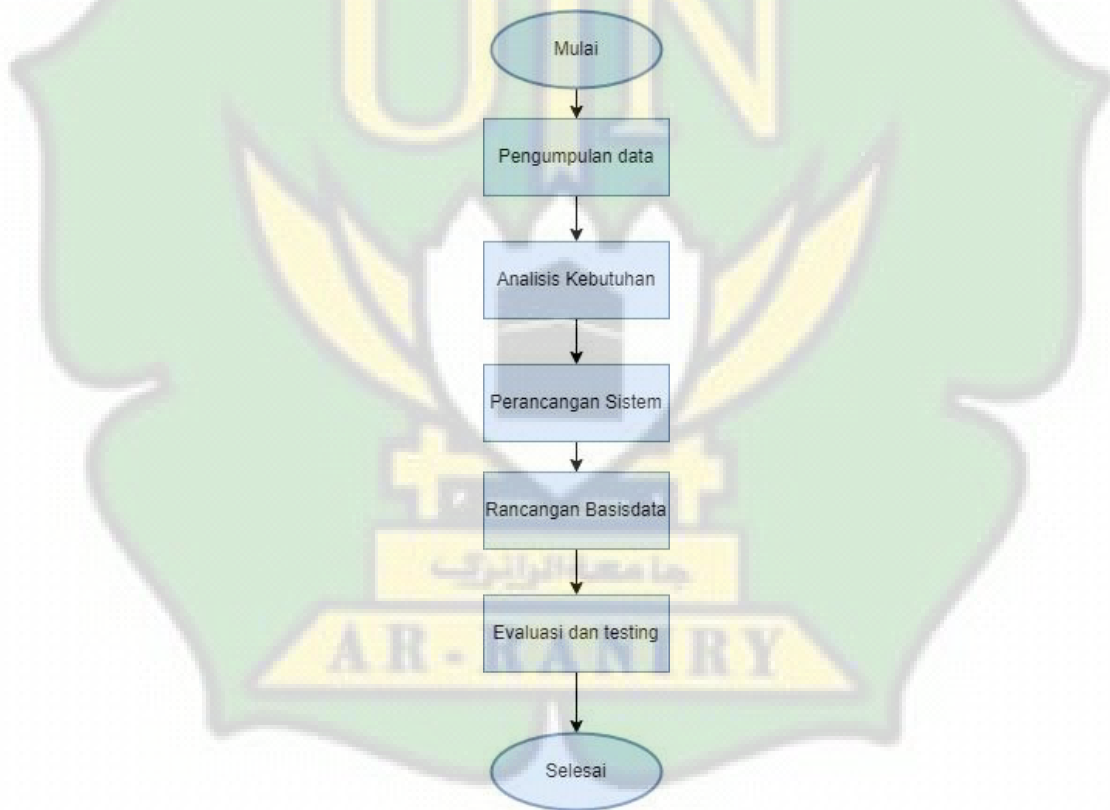
Pengembangan yang dilakukan oleh (A. M. Sari et al., 2021) yang judul penelitiannya “Implementasi Aplikasi Mobile Peta NKRI (Negara Kesatuan Republik Indonesia) Berbasis Android Menggunakan Metode Prototype”. Penelitian ini menghasilkan sebuah aplikasi berbasis android yang dapat mempermudah seseorang yang ingin mengetahui serta belajar tentang betapa indah dan kaya negara ini. Aplikasi ini mempermudah user untuk melihat peta dari kepulauan Negara Indonesia yang terdiri dari 34 provinsi

BAB III

METODOLOGI PENELITIAN

III.1 Tahapan penelitian

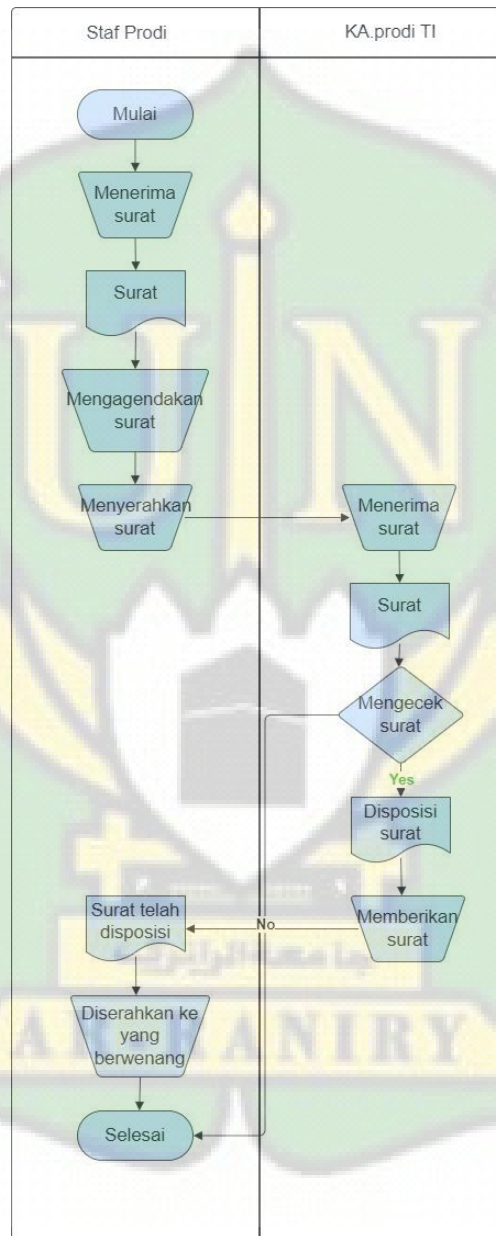
Dalam proses perancangan dan pembangunan sistem, peneliti membuat kerangka kerja yang berbentuk skema untuk memudahkan penulis dalam menyelesaikan rancang bangun sistem informasi arsip surat dengan skema tersebut dapat menjadi pedoman dalam melakukan tahapan-tahapan proses perancangan dan membangun sistem informasi arsip surat. Adapun tahap-tahap yang disusun sebagai berikut:



Gambar III.1 : Tahapan Penelitian

Adapun *flowchart* yang sedang berjalan pada kearsipan surat masuk Prodi TI adalah sebagai berikut.

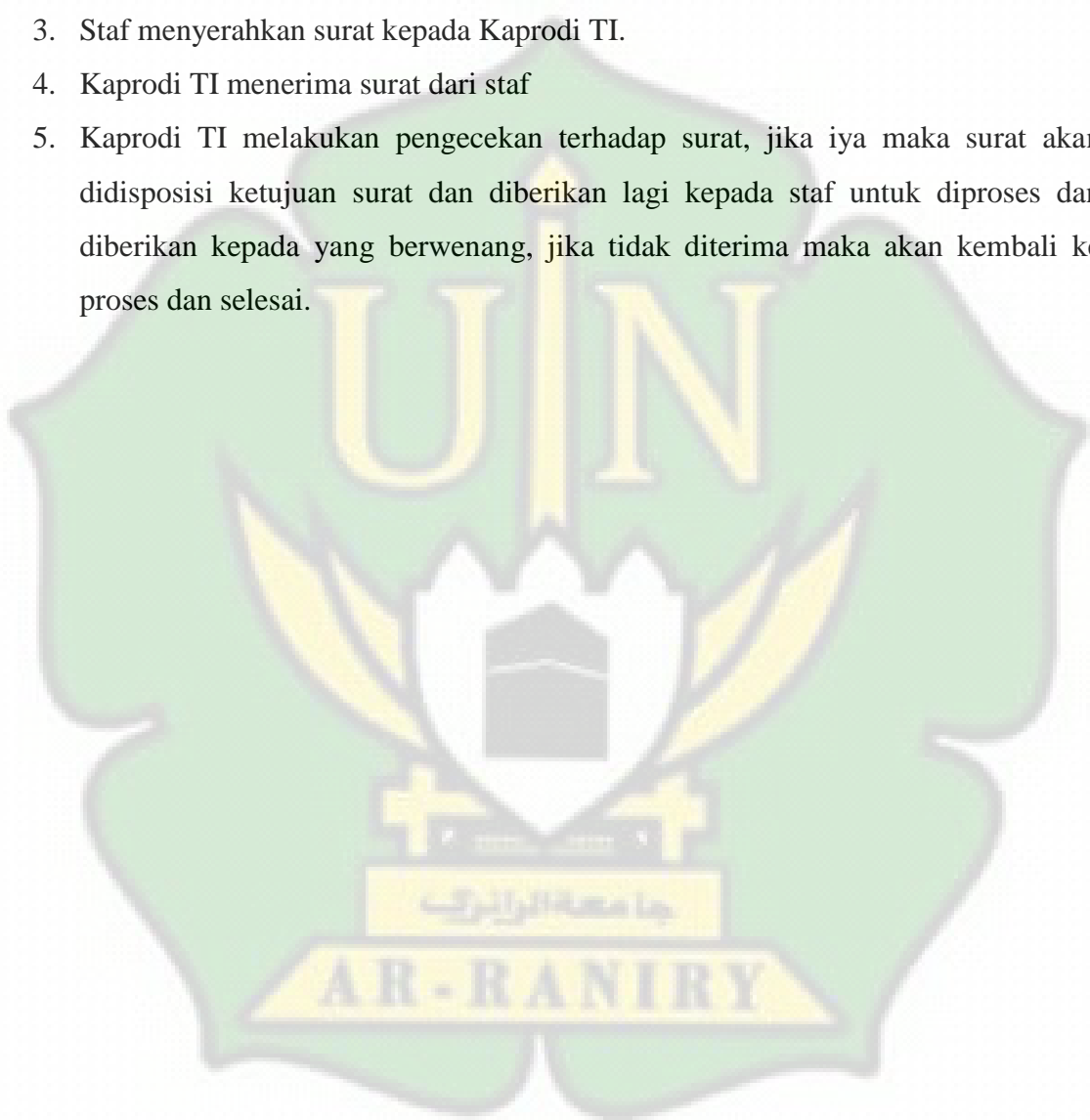
a. *Flowchart* surat masuk



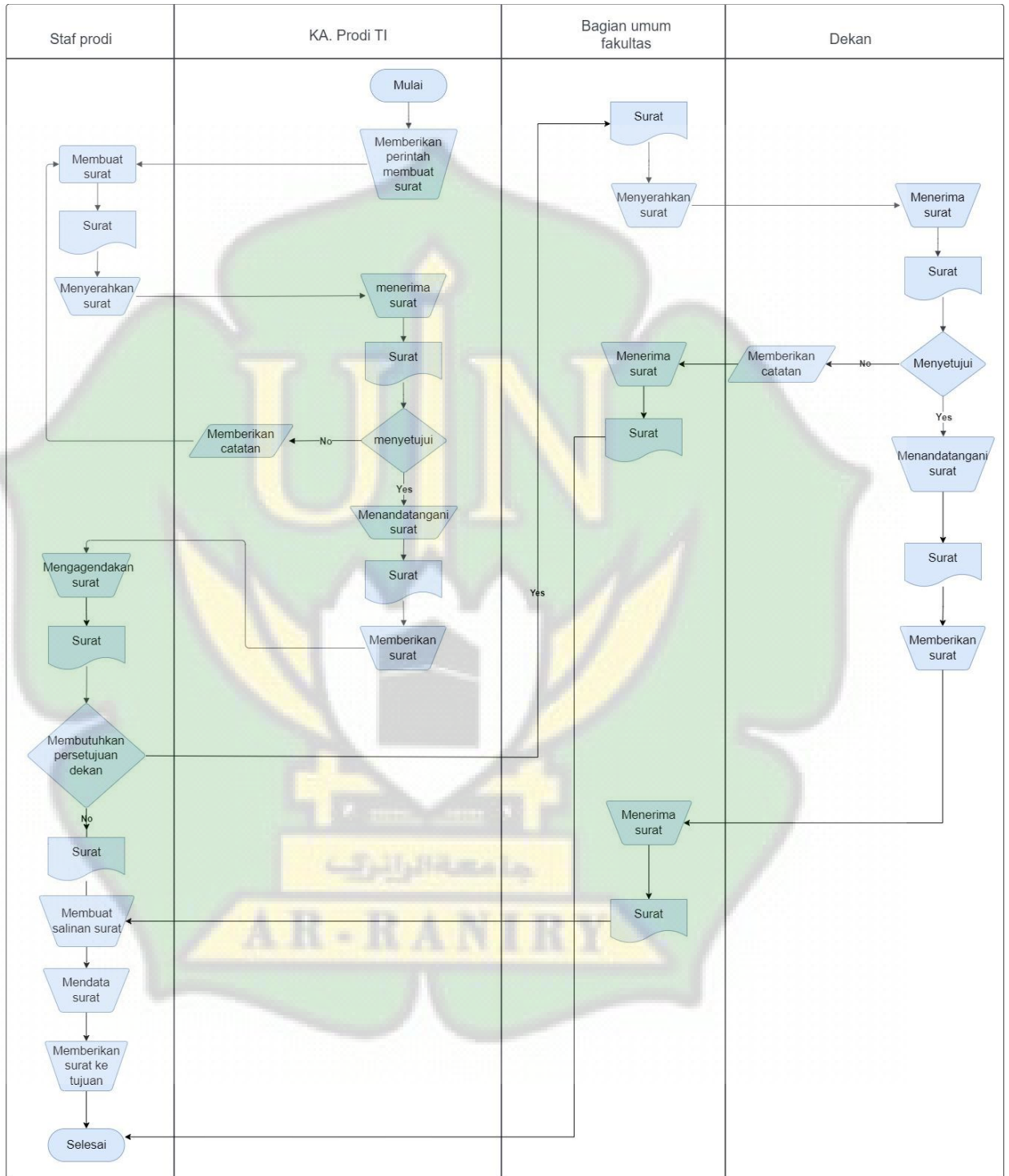
Gambar III.2 : *Flowchart* surat masuk

Berdasarkan gambaran *flowchart* yang sedang berjalan pada surat masuk dapat dijelaskan sebagai berikut :

1. Staf menerima surat dari instansi, himpunan atau dari pihak lainnya.
2. Staf mendata surat dibuku agenda.
3. Staf menyerahkan surat kepada Kaprodi TI.
4. Kaprodi TI menerima surat dari staf
5. Kaprodi TI melakukan pengecekan terhadap surat, jika iya maka surat akan didisposisi ketujuan surat dan diberikan lagi kepada staf untuk diproses dan diberikan kepada yang berwenang, jika tidak diterima maka akan kembali ke proses dan selesai.



b. *Flowchart surat keluar*



Gambar III.3 : *Flowchart surat keluar*

Berdasarkan gambaran *flowchart* yang berjalan pada surat keluar dapat dijelaskan sebagai berikut :

1. Kaprodi TI memberikan perintah untuk membuat surat keluar kepada staf.
2. Staf membuat surat yang telah di perintah, lalu menyerahkan surat keluar kepada Kaprodi TI.
3. Kaprodi TI menerima surat yang telah dibuat oleh staf dan melakukan pengecekan jika menyetujui maka surat keluar akan ditanda tangani dan memberikan surat kepada staf, jika tidak di setujui maka surat akan diberikan catatan dan diberikan kepada staf untuk pembuatan ulang.
4. Staf melakukan pengecekan apakah membutuhkan persetujuan dekan atau tidak, jika iya maka surat akan diberikan oleh staf kepada bagian umum fakultas, jika tidak maka surat akan dibuatkan salinan dan akan didata surat di buku agenda lalu memberikan surat kepada pihak yang berwenang.
5. Bagian umum fakultas menerima surat dari staf dan memberikan surat kepada dekan.
6. Dekan menerima surat dan melakukan pengecekan, jika surat keluar disetujui maka dekan akan menanda tangani dan surat akan di berikan kepada bagian umum fakultas, dibagian umum fakultas akan memberikan surat keluar kembali kepada staf, lalu staf membuat salinan, mendata surat dan menyerahkan surat kepada pihak yang berwenang, jika tidak di setujui maka dekan akan memberikan catatan dan surat akan diberikan kepada bagian umum fakultas, lalu bagian umum fakultas mengembalikan surat kepada staf Prodi TI dan selesai.

III.2 Teknik Pengumpulan data

Pengumpulan data ialah suatu metode wajib dipenuhi dalam melakukan suatu penelitian. Tujuannya adalah untuk mendapatkan data yang valid dan benar, maka penulis menggunakan beberapa metode untuk pengumpulan data sebagai berikut:

III.2.1 Observasi

Tahapan pengumpulan informasi dan data-data awal, peneliti melakukan observasi untuk mengumpulkan informasi yang diperlukan untuk mengembangkan sistem pengarsipan surat yang masuk dan keluar pada Prodi TI. Observasi dilakukan pada prodi TI bertepatan pada tanggal 5 April 2023. Adapun hasilnya sebagai berikut: cara staf dalam melakukan pendataan dan pengarsipan surat masih secara manual yang masih dilakukan pendataan pada buku agenda yang akan di masukkan kedalam map lalu disimpan kedalam lemari arsip.

III.2.2 Metode Wawancara

Pada tahapan ini melibatkan proses wawancara yang dilakukan kepada calon user yang berupa admin Prodi Teknologi Informasi yang nantinya akan menggunakan sistem yang telah di rancang. Pada Prodi Teknologi Informasi memiliki satu orang admin untuk mengelola sistem tersebut. Hasil dari wawancara akan disusun dan digunakan untuk acuan dalam perancangan sistem pengarsipan surat yang masuk dan keluar.

Tabel III.1 : Instrumen wawancara

No	Pertanyaan
1	Darimana sumber arsip berasal ?
2	Bagaimana proses pengelolaan arsip dinamis pada prodi TI ?
3	Apa saja catatan yang dibuat dalam prodi TI ?
4	Bagaimana bentuk format catatan yang dibuat ?
5	Bagaimana sistem menemukan kembali catatan pengelolaan arsip dinamis di prodi TI ?
6	Bagaimana sistem menemukan kembali catatan pengelolaan arsip dinamis di prodi TI ?

Wawancara dilakukan pada tanggal 5 April 2023 pada prodi TI dengan staf prodi yaitu ibuk Cut Ida Rahmadiana, S.Pd, adapun hasil wawancara seperti pada tabel dibawah sebagai berikut :

Tabel III.2 : Hasil Wawancara

No	Pewawancara (Penulis)	Staf Prodi TI
1	Darimana sumber arsip berasal ?	Untuk surat yang masuk ke prodi terdiri dari dua sumbernya, dari bagian umum fakultas dan mahasiswa, selain itu tidak ada yang lain, semua surat yang masuk dari luar, itu semua melalui bagian umum fakultas nanti baru di berikan kepada prodi. Sedangkan untuk surat keluar ya dari prodi sendiri baik itu surat perintah atau lain nya.
2	Bagaimana proses pengelolaan arsip dinamis pada prodi TI ?	untuk arsip surat masuk, setelah diberikan surat oleh pihak bagian umum fakultas lalu saya mendata surat tersebut di buku agenda warna hijau, setelah di catat maka surat diserahkan ke sekretaris prodi pak khairan, lalu pak khairan menyerahkan kepada ketua prodi buk ima, setelah sampai surat itu dan di ketahui oleh buk ima, surat itu di serahkan lagi kepada saya untuk saya proses, setelah di proses surat itu di masukkan kedalam map file dan disimpan kedalam lemari arsip, nah untuk surat keluarnya setelah saya membuat surat saya menyerahkan nya ke ketua prodi atau sekretaris prodi untuk di setuju dan di tanda tangan, setelah itu saya memproses surat tersebut lalu mendata surat keluar itu di buku agenda, lalu baru saya masukkan kedalam map file dan saya simpan di lemari arsip.
3	Apa saja catatan yang dibuat dalam prodi TI ?	Catatan untuk surat keluar dan surat masuk nya seperti yang ada di buku agenda tersebut, untuk surat masuk ada nomor berturut, sipengirim, nomor surat dan tanggal, isi ringkasan dan hubungan dengan nomor agenda berikutnya. Untuk surat keluar ada isi ringkasan, alamat dan tanggal, hubungan dengan nomor agenda berikutnya dan keterangan.
4	Bagaimana bentuk format catatan yang dibuat ?	Untuk format catatan nya seperti yang ada di buku agenda sendiri, buku yang warna hijau.

5	Bagaimana sistem menemukan kembali catatan pengelolaan arsip dinamis di prodi TI ?	Untuk mendapatkan surat yang telah di arsip, dengan cara membaca catatan lagi agar tau surat tersebut di tanggal berapa nomor berapa, setelah informasinya tau maka di cari surat nya dalam map file yang di simpan di lemari arsip tadi.
6	Apa yang menjadi kendala atau hambatan yang terdapat pada pengelolaan arsip sekarang	kendala yang pertama itu susah dalam proses menemukan kembali surat2 yang telah di simpan di lemari arsip, untuk menemukan nomor surat keluar pada beberapa bulan kebelakang juga susah dan sesekali surat masuk di agendakan 2 kali.

Berdasarkan wawancara di atas yang dilakukan pada tanggal 5 April 2023 pada prodi TI dengan staf prodi yaitu saudari Cut Ida Rahmadiana, S.Pd, dapat disimpulkan sebagai berikut :

1. Sumber arsip prodi TI berasal dari bagian umum fakultas sains dan teknologi dan mahasiswa teknologi informasi.
2. Proses pengelolaan arsip surat masuk pertama dicatat pada buku agenda, lalu diserahkan pada sekretaris prodi, lalu sekretaris prodi menyerahkan pada ketua program studi, setelah ketua program studi mengetahui surat tersebut maka surat dikembalikan kepada staf prodi untuk di proses, setelah di proses surat tersebut diarsipkan di dalam map file. Sedangkan surat keluar proses arsipnya, setelah staf membuat surat lalu memberikan kepada ketua program studi atau sekretaris prodi

untuk disetujui, lalu staf memproses surat keluar tersebut dan mencatat pada buku agenda setelah itu baru melakukan pegarsipan pada map file surat keluar.

3. Catatan yang dibuat pada surat masuk antara lain, nomor berturut, sipengirim, nomor dan tanggal, isi ringkas dan hubungan dengan nomor agenda berikutnya. Adapun untuk surat keluar antara lain, isi ringkasan, alamat dan tanggal, hubungan dengan nomor agenda yang berikut dan keterangan.
4. Adapun format catatan yang dibuat untuk surat yang masuk seperti pada gambar 3.4 dan surat yang keluar seperti pada gambar 3.5 berikut :

SURAT - SURAT MASUK				Hubungan dengan nomor agenda yg berikut
Nomor Berturut	Sipengirim	Nomor dan Tanggal	ISI RINGKAS	
		7724/Un.08/	Pembentahan Perpanjangan	
		B.1/Kp.04.1/	Pemataphiran Data My SAPK	
		11/2021		

Gambar III.4 : Format catatan surat masuk

SURAT - SURAT KELUAR		Hubungan dengan nomor agenda yg berikut	KETERANGAN
ISI RINGKAS	Alamat dan Tanggal		
Undangan Sempro An. Syauki Ikramillah			
Undangan Sempro An. Fathul Faiz			

Gambar III.5 : Format catatan surat keluar

5. Proses menemukan kembali catatan pengelolaan arsip dengan cara membaca kembali catatan di buku arsip lalu mencari surat yang masuk dan keluar pada map file.

6. Kendala yang dialami pada proses pengarsipan surat antara lain, kesusahan dalam menemukan surat keluar dan surat masuk yang telah di arsip beberapa bulan sebelumnya, kesusahan dalam menemukan no surat keluar pada beberapa bulan sebelumnya dan ada kalanya surat masuk diagendakan dua kali.

III.2.3 Studi Literatur

Metode pengumpulan yang dilakukan penulis untuk mencari berbagai data-data mengenai sistem pengarsipan surat. Informasi yang digunakan sebagai referensi yang didapatkan dari membaca dan mempelajari buku-buku, jurnal, artikel penelitian dan berbagai sumber lainnya yang berhubungan dengan sistem informasi yang diangkat penulis. Hasil dari studi literatur dapat dilihat pada BAB II LANDASAN TEORI proposal tugas akhir.

III.3 Analisis Kebutuhan Sistem

Analisa untuk kebutuhan sistem adalah langkah awal untuk pengembangan perangkat lunak yang tujuannya memperoleh informasi dan spesifikasi tentang perangkat lunak yang sesuai dengan *user* atau klien. Kedua belah pihak, yakni *user* dan sipembuat perangkat lunak sangat terlibat pada tahapan ini. Informasi dari *user* terlibat akan menjadi rujukan dalam melakukan pendesain dari sebuah perangkat lunak.

Adapun beberapa kebutuhan yang dibutuhkan oleh sistem untuk analisis yaitu seperti menganalisis kebutuhan input, proses, output, perangkat lunak (*Software*) dan analisis kebutuhan perangkat keras (*hardware*). Langkah-langkah yang dibutuhkan oleh sistem sebagai berikut:

III.3.1 Kebutuhan Input

Sistem informasi pengarsipan surat ini dibutuhkan beberapa inputan yang akan dimasukkan oleh admin yaitu data-data surat seperti: nomor agenda, nomor, tanggal, asal, sifat dan perihal tentang yang tertera pada surat.

III.3.2 Kebutuhan Proses

Sistem informasi juga memiliki proses yang dapat dilakukan oleh admin dan user terhadap sistem informasi pengarsipan ini. Adapun proses yang dilakukan oleh

admin ialah untuk mengola data surat seperti: menambah data surat yang masuk dan keluar, mengubah data surat yang masuk dan keluar, menghapus data surat yang masuk dan keluar, dan juga menampilkan laporan data surat yang masuk dan keluar untuk si user. Proses yang dilakukan oleh user adalah mengecek data surat yang masuk dan keluar untuk memberikan perizinan terhadap surat sehingga dapat didisposisi ketujuan surat untuk diberikan kepada yang berwenang.

III.3.3 Kebutuhan Output

Menganalisis kebutuhan output ada beberapa output yang bisa didapatkan dari hasil input dan proses sebelumnya. Adapun output yang ini dihasilkan ialah sebuah laporan surat masuk dan keluar.

III.3.4 Kebutuhan Perangkat Lunak (*Software*)

Software yang telah digunakan pada pengembangan sistem pengarsipan surat pada penelitian ini yaitu sebagai berikut:

- a. *Xampp Control Panel*
- b. *Visual Studio Code*
- c. *Mysql*
- d. *PHP, Html, Css, Javascript*

III.3.5 Kebutuhan Perangkat Keras (*Hardware*)

Hardware atau spesifikasi dari perangkat keras yang digunakan pada pengembangan sistem informasi pengarsipan surat pada penelitian ini adalah sebagai berikut:

Laptop	:	Asus
Processor	:	Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz 1.99 GHz
RAM	:	8 GB
Penyimpanan	:	500 GB
Monitor	:	14 inch

III.4 Perancangan Sistem

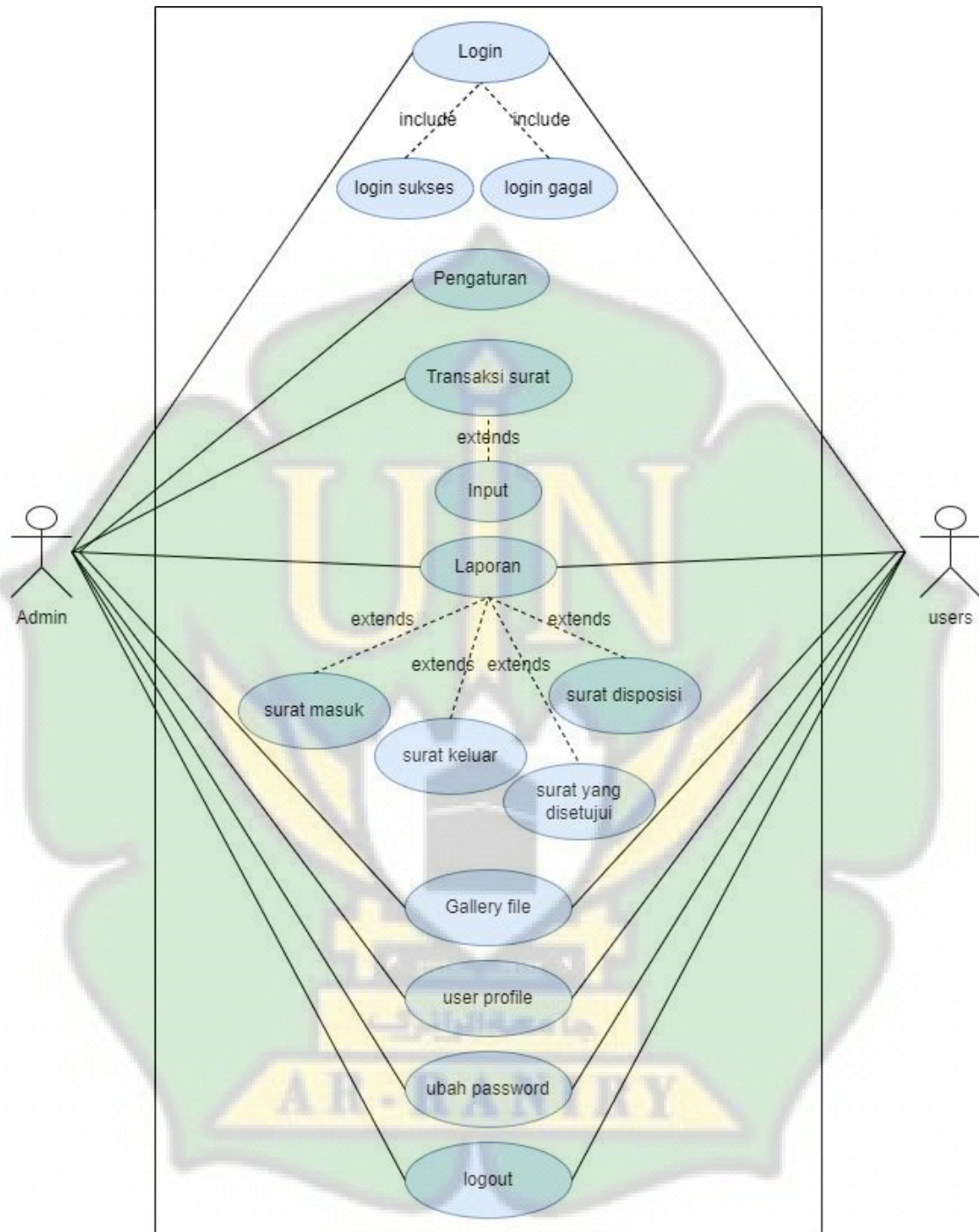
Perancangan sistem ialah proses yang dilakukan setelah terjadinya komunikasi antara *client* dalam pembuatan sebuah sistem. Ada beberapa hal yang diperlukan untuk merancang sistem informasi dengan menggunakan diagram UML, dan rancangan *database*.

III.4.1 Use Case Diagram

Use case ini sebagai gambaran proses yang dirancang pada sistem. Sistem akan melakukan interaksi dengan actor dan user untuk menjelaskan tentang keseluruhan sistem. Pada gambar *use case* dibawah ini aktor disebut sebagai administrator. Ada beberapa interaksi yang dapat aktor lakukan pada sistem yaitu, *login* ke sistem, melakukan transaksi surat, laporan surat, user profile dan mengelola data surat yang masuk dan keluar, *use case* diagramnya dapat dilihat sebagai berikut.

Berdasarkan *use case* diagram pada gambar 3.6 dibawah ini, sistem pengarsipan surat memiliki beberapa kegiatan, yaitu sebagai berikut :

1. Sistem mencakup semua fitur yang ada pada sistem pengarsipan surat masuk dan keluar.
2. Satu orang admin yang bisa mengakses seluruh fitur yang ada pada sistem pengarsipan surat masuk dan keluar.
3. User bisa mengakses login, dashboard, laporan surat, laporan surat masuk dan keluar, laporan surat disposisi, laporan surat disetujui, user profile, ubah password, gallery file dan logout.



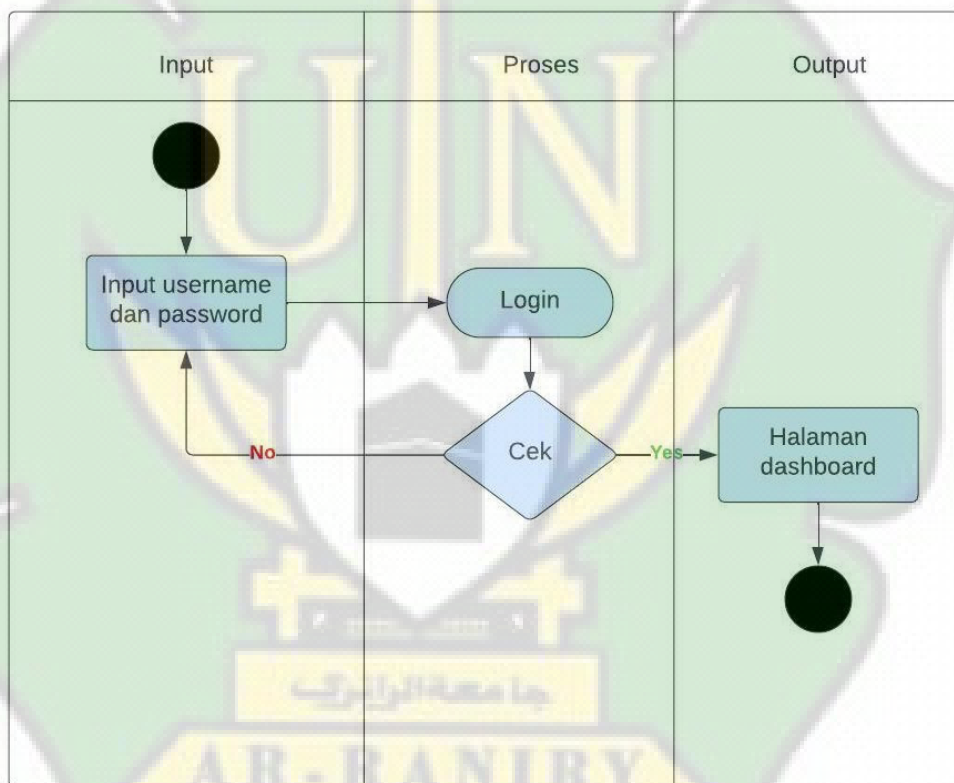
Gambar III.6 : Use Case Diagram

III.4.2 Activity Diagram

Activity diagram ini berguna sebagai contoh gambaran aktivitas yang akan dikerjakan oleh *actor* kepada sistem. Adapun *activity diagram* sistem pengembangan sistem informasi pengarsipan surat, yaitu:

a. *Activity diagram login*

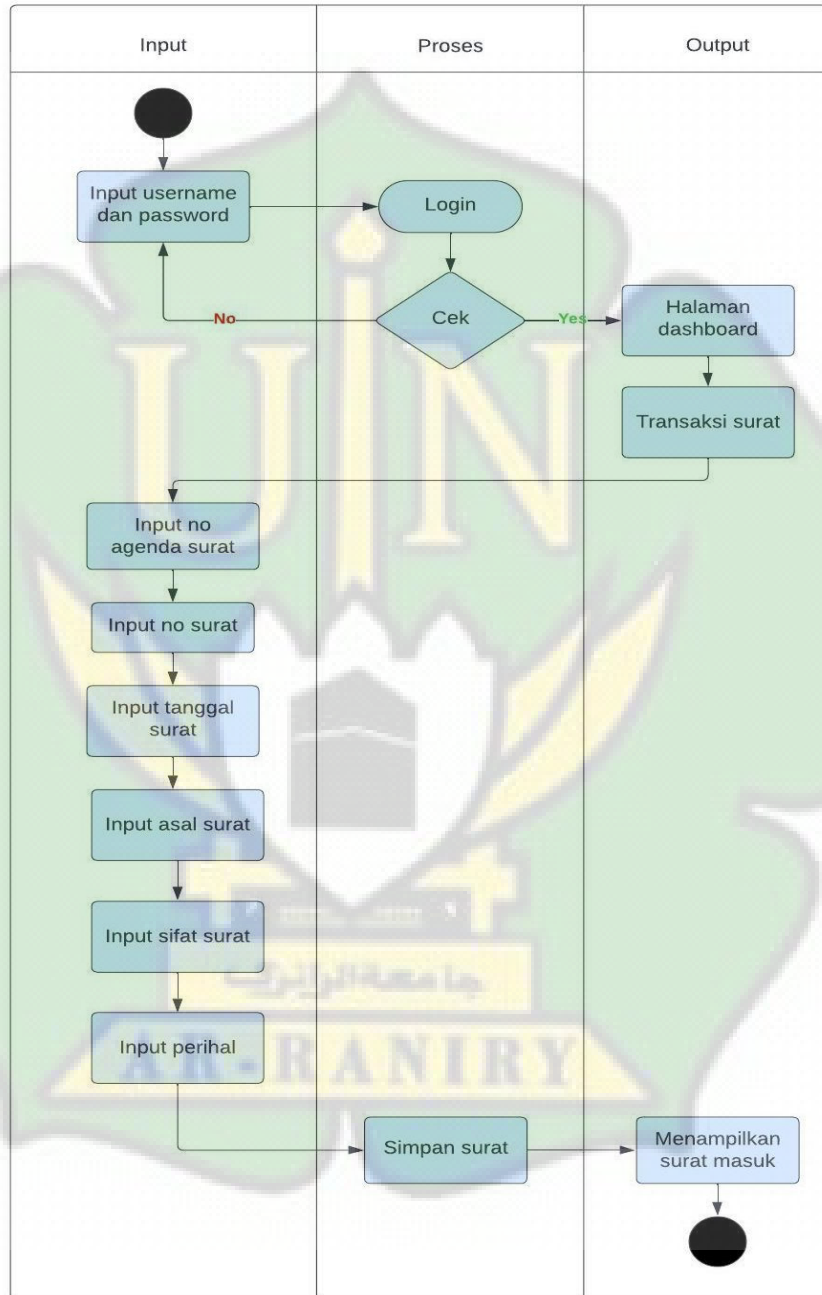
Administrator dan *user* memasukkan *username* dan *password* sehingga administrator dan *user* bisa masuk ke sistem. Setelah validasi data berhasil akan masuk ke menu utama sistem atau ke *dashboard*.



Gambar III.7 : *Activity Diagram Login*

b. *Activity diagram* transaksi surat masuk

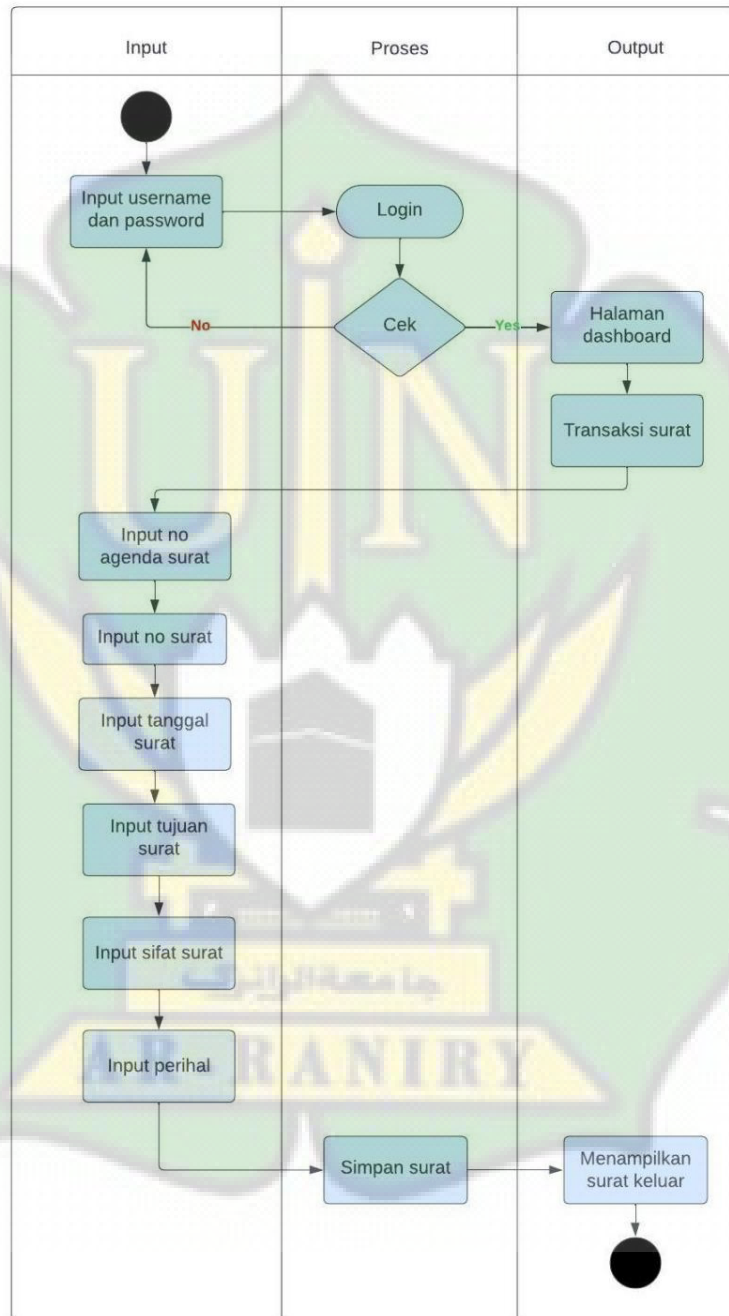
Setelah masuk pada halaman sistem, aktivitas yang dapat dilakukan oleh administrator ialah, menambahkan data surat yang masuk.



Gambar III.8 : *Activity diagram* transaksi surat masuk

c. *Activity diagram* menu transaksi data surat keluar

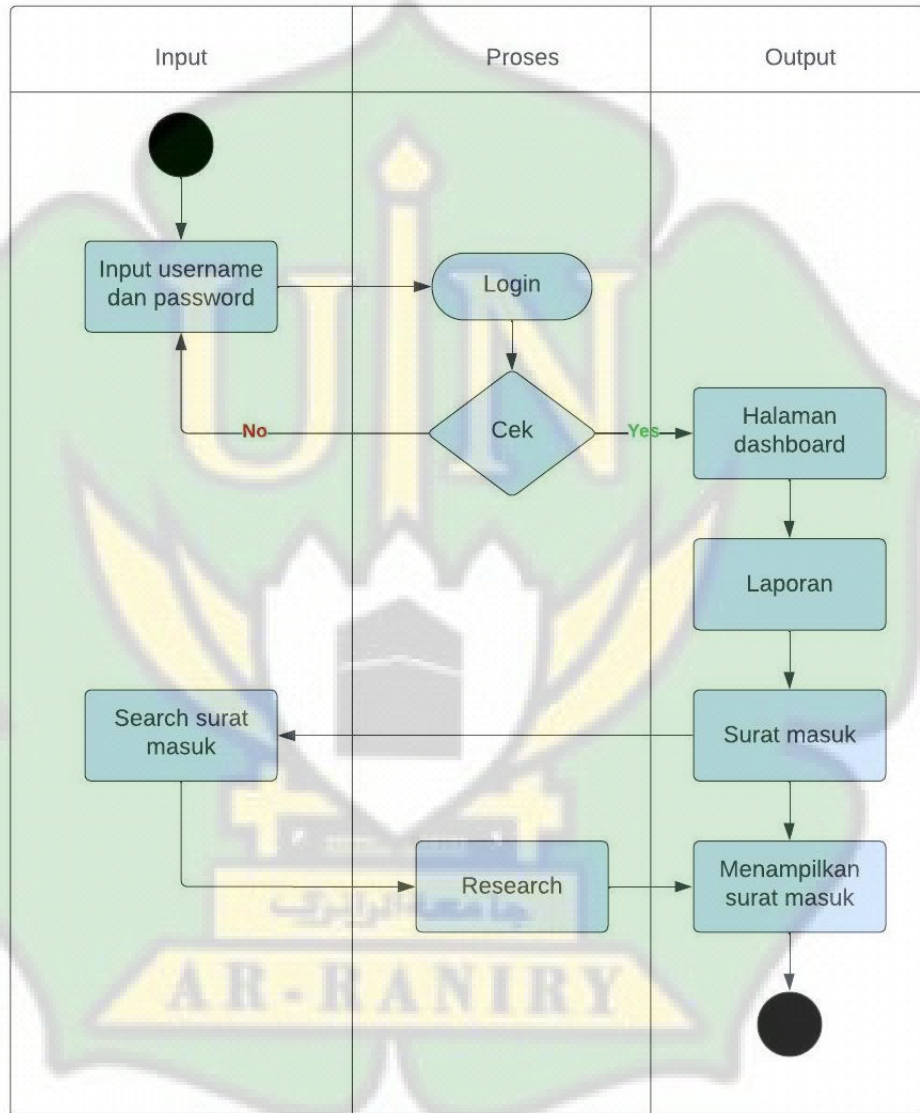
Setelah masuk pada halaman sistem, aktivitas yang dapat dilakukan oleh administrator ialah, menambahkan data surat yang keluar.



Gambar III.9 : *Activity diagram* transaksi surat keluar

d. *Activity diagram* menu laporan surat masuk

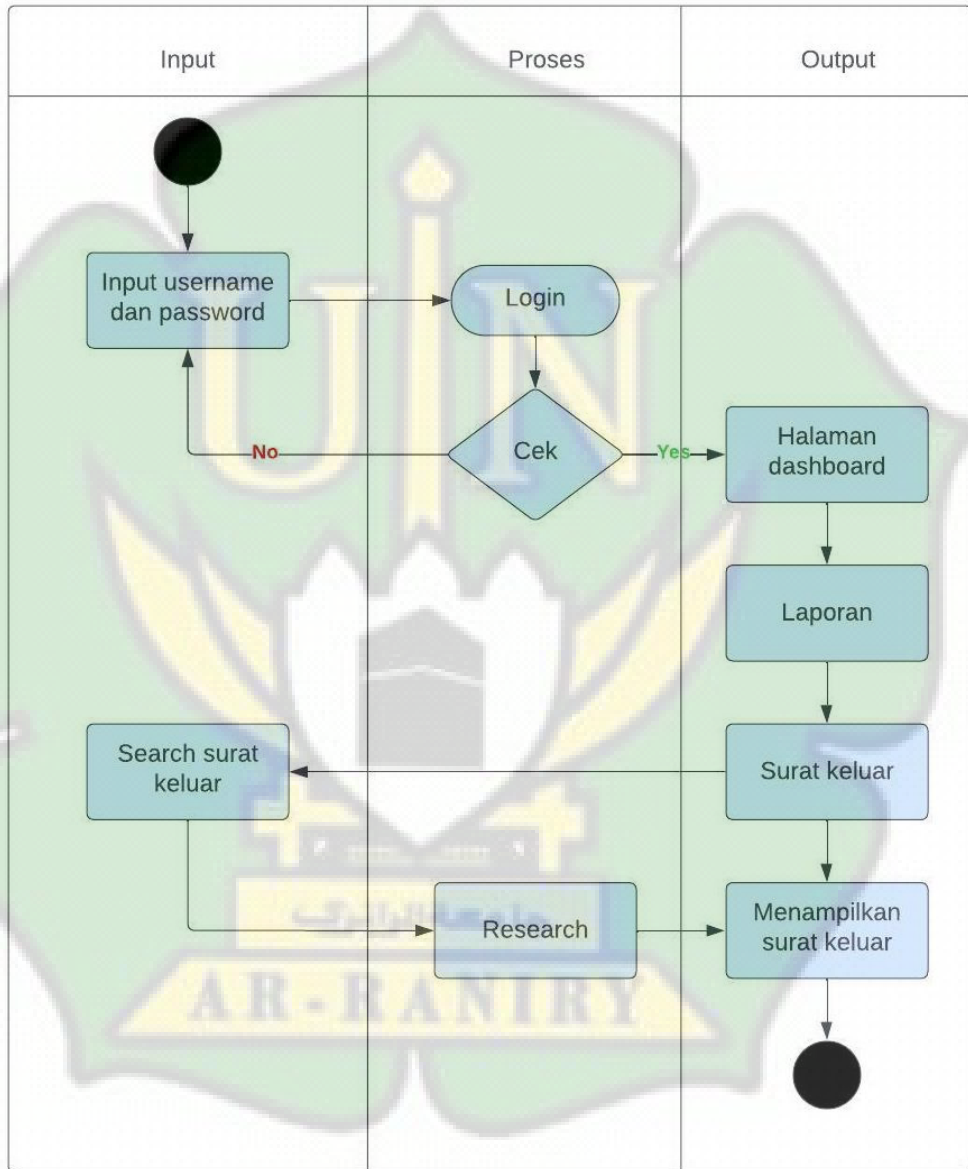
Aktivitas yang dilakukan administrator dan user untuk melihat semua laporan surat yang masuk dan telah di input oleh admin ke dalam sistem. Bisa mencari surat yang di perlukan dan melihat *detail* dari surat yang diinginkan.



Gambar III.10 : *Activity diagram* laporan surat masuk

e. *Activity diagram* menu laporan surat keluar

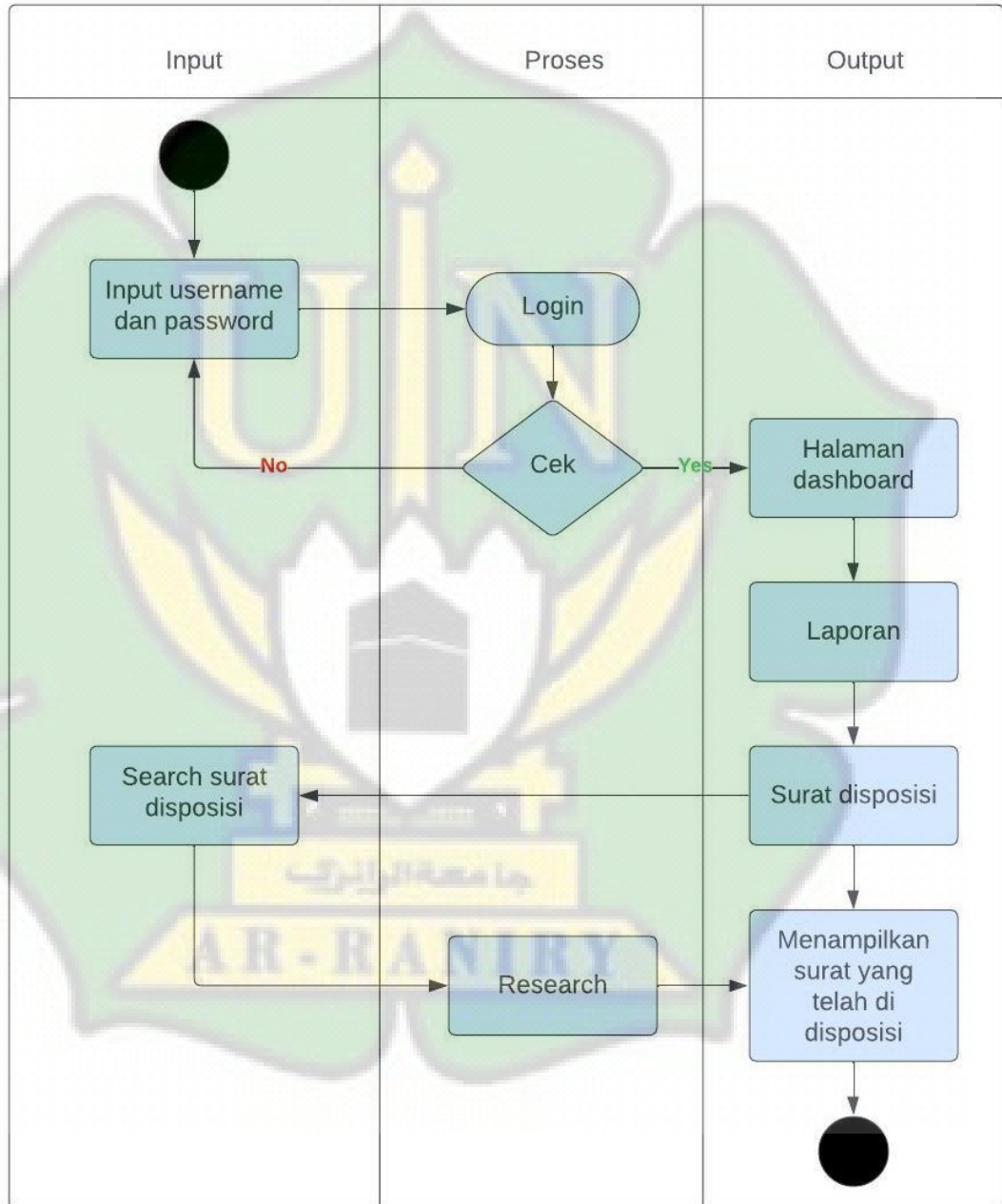
Aktivitas yang dilakukan administrator dan *user* untuk melihat semua laporan surat yang keluar dan telah di input oleh admin ke dalam sistem. Bisa mencari surat yang di perlukan dan melihat *detail* dari surat yang diinginkan.



Gambar III.11 : *Activity diagram* laporan surat keluar

f. *Activity diagram* menu menu laporan surat disposisi

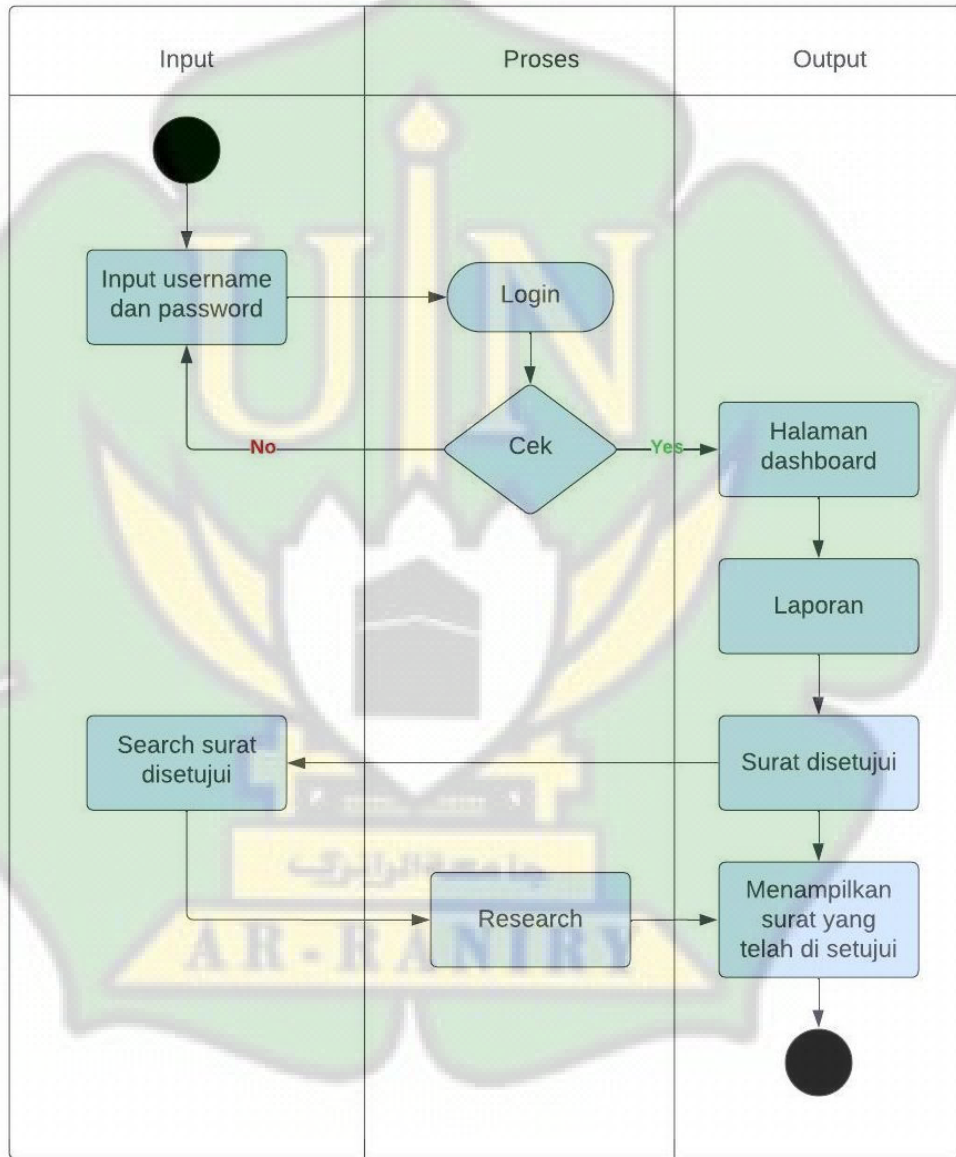
Aktivitas yang dilakukan administrator dan user untuk melihat semua laporan surat disposisi yang telah di input oleh admin ke dalam sistem. Bisa mencari surat yang di perlukan dan melihat *detail* dari surat yang diinginkan.



Gambar III.12 : *Activity diagram* laporan surat disposisi

g. *Activity diagram* menu menu laporan surat disetujui

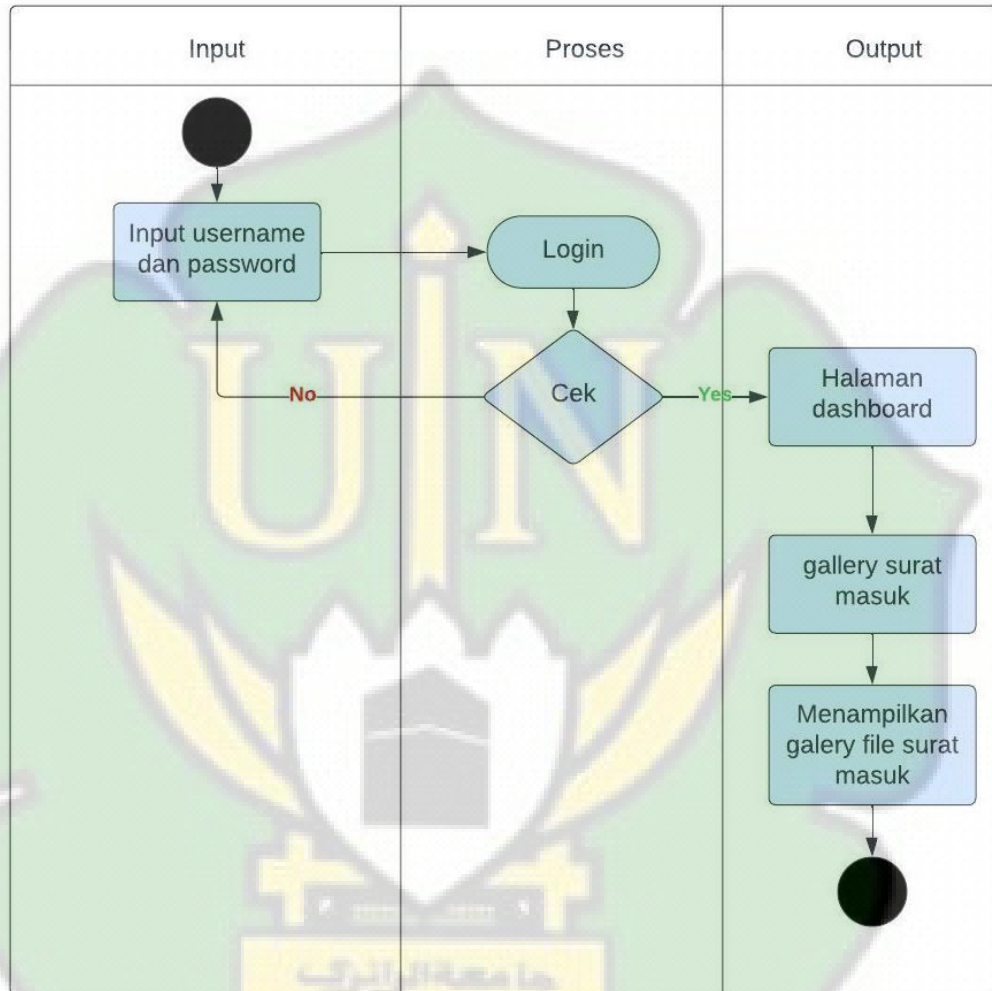
Aktivitas yang dilakukan administrator dan user untuk melihat semua laporan surat yang disetujui yang telah di input oleh admin ke dalam sistem. Bisa mencari surat yang di perlukan dan melihat *detail* dari surat yang diinginkan.



Gambar III.13 : *Activity diagram* laporan surat disetujui

h. *Activity diagram* gallery file surat masuk

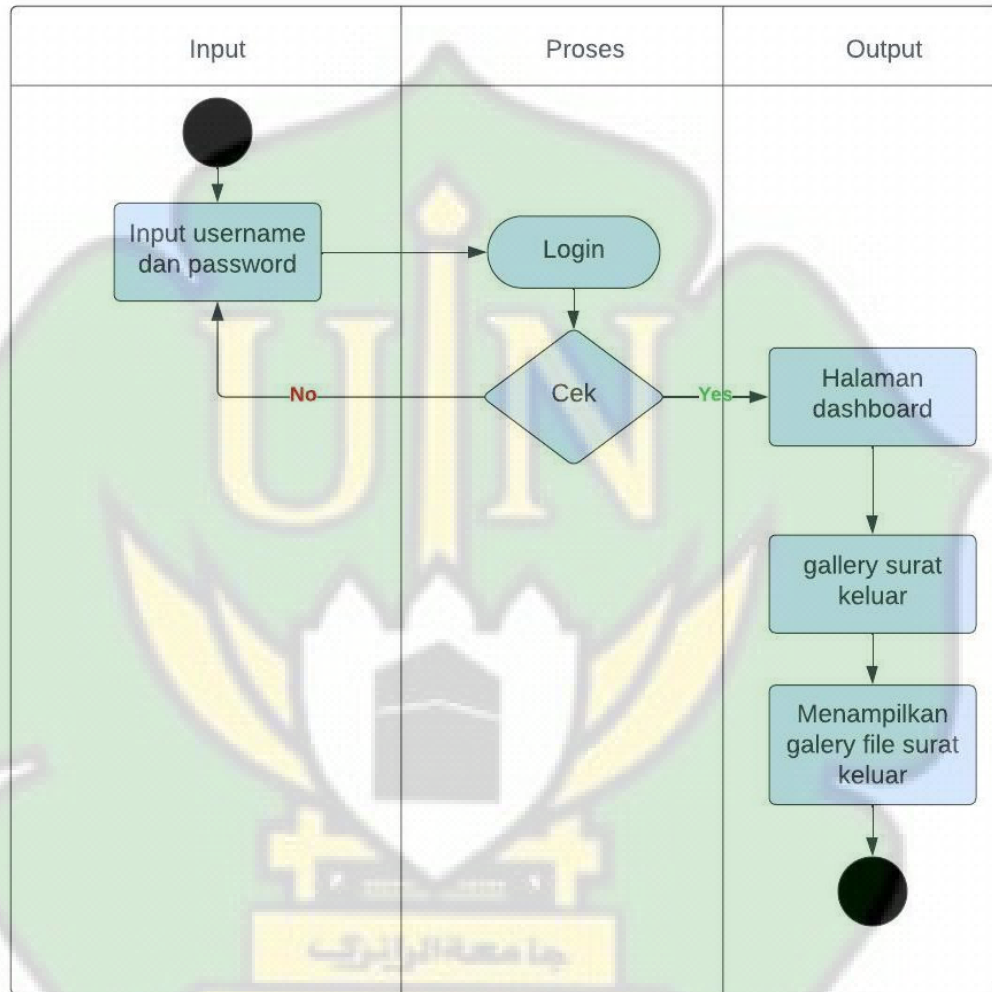
Aktivitas yang dilakukan administrator dan user untuk melihat foto dari file surat masuk yang sudah di setuju dan disposisi.



Gambar III.14 : *Activity diagram* gallery file surat masuk

i. *Activity diagram gallery file surat keluar*

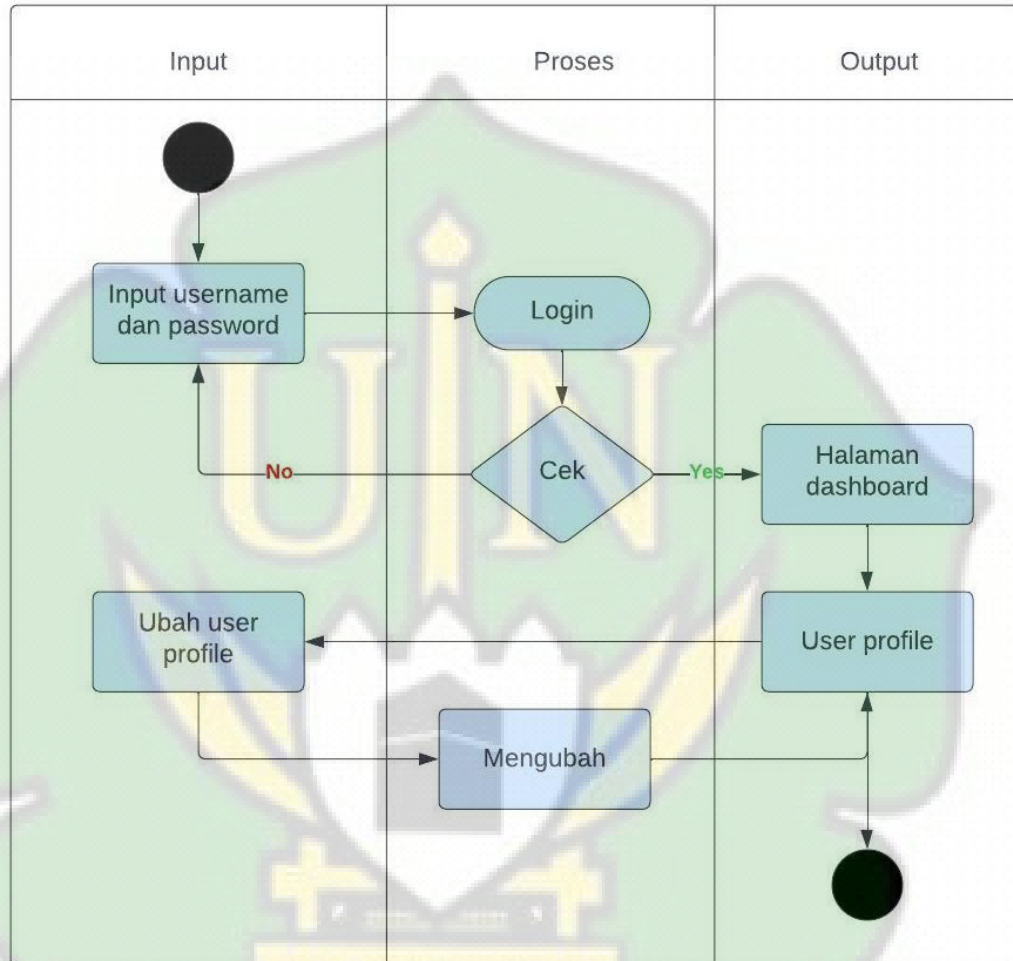
Aktivitas yang dilakukan administrator dan user untuk melihat foto dari file surat keluar yang sudah di setujui dan disposisi.



Gambar III.15 : *Activity diagram gallery file surat keluar*

j. *Activity diagram* menu user profile

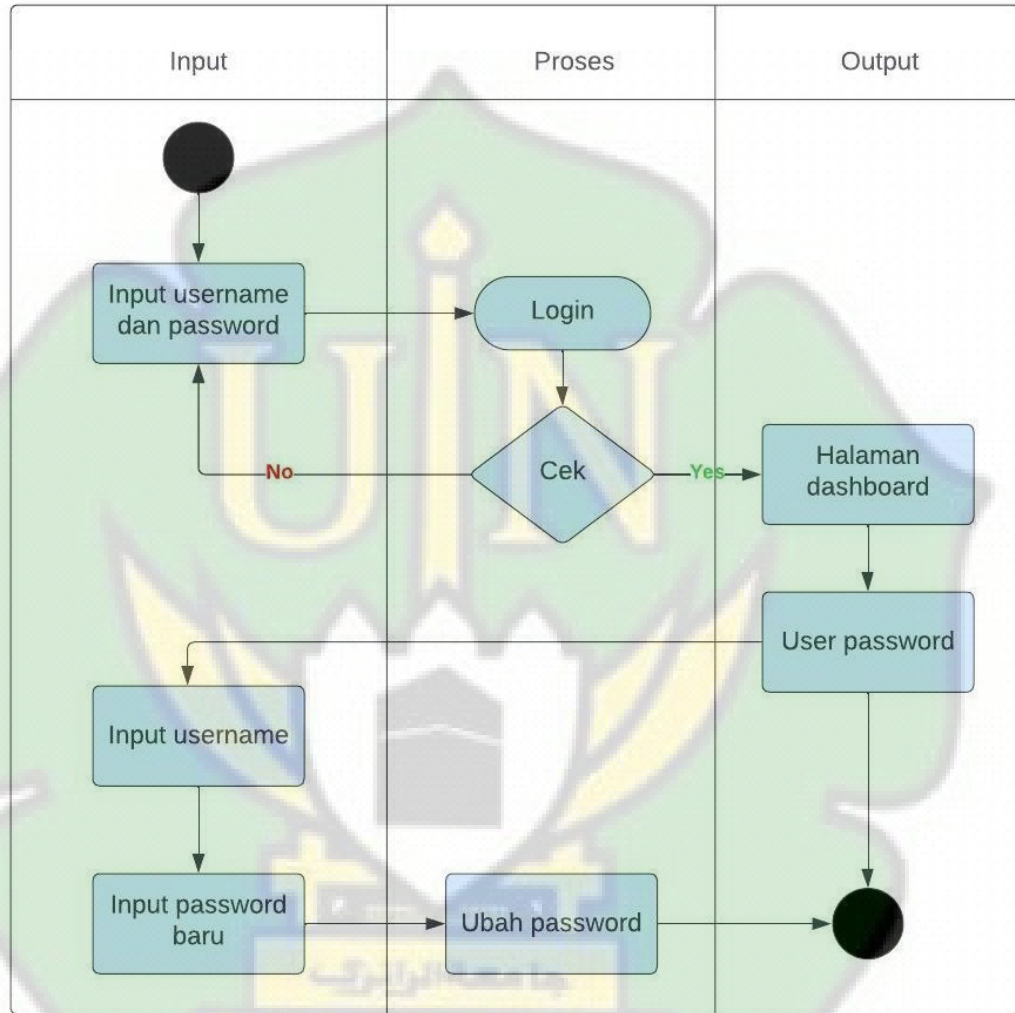
Aktivitas yang dilakukan administrator dan *user* untuk melihat detail dari profile sendiri dan dapat mengubah profile yang telah ada.



Gambar III.16 : *Activity diagram* user profile

k. *Activity diagram menu ubah password*

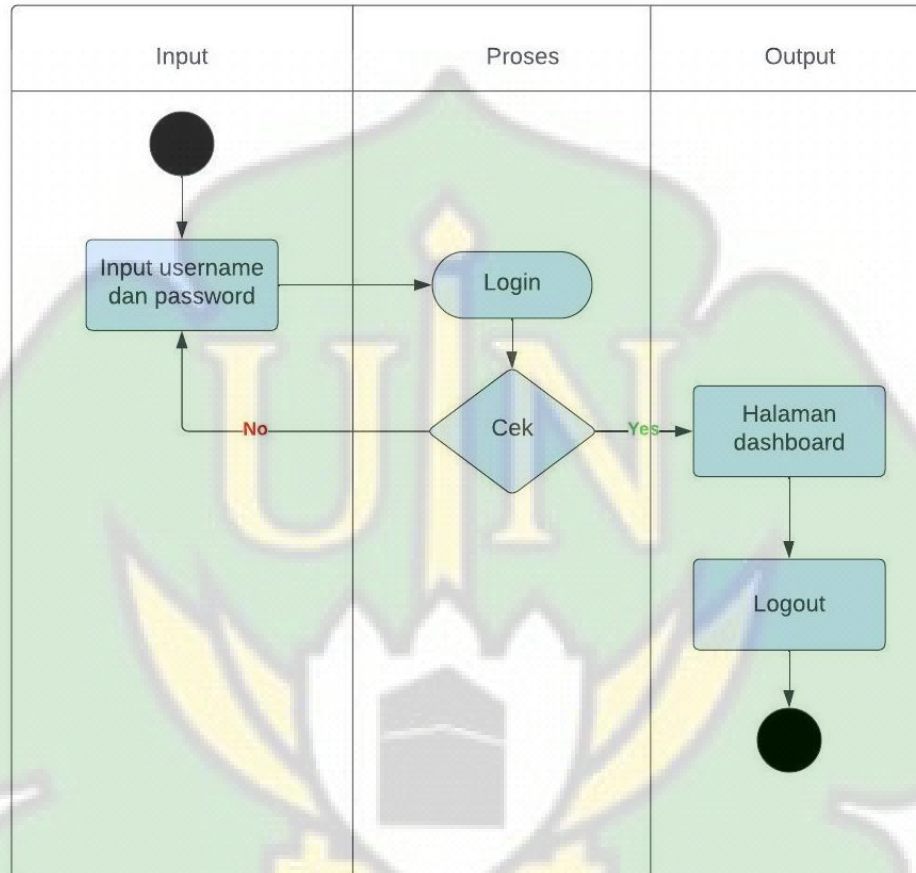
Aktivitas yang dilakukan administrator dan *user* untuk mengubah *password* dari sistem.



Gambar III.17 : *Activity diagram ubah password*

1. *Activity diagram menu logout*

Aktivitas yang dilakukan administrator dan *user* untuk keluar dari sistem



Gambar III.18 : *Activity diagram logout*

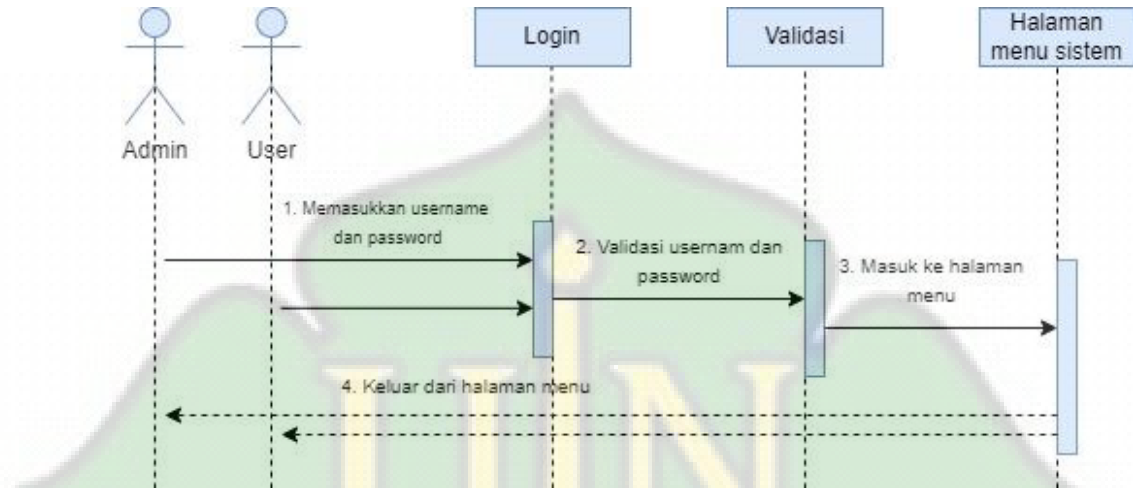
III.4.3 Sequence Diagram

Sequence digram merupakan sebuah gambaran yang menunjukkan interaksi proses yang akan aktor lakukan terhadap sistem.

a. *Sequence diagram login*

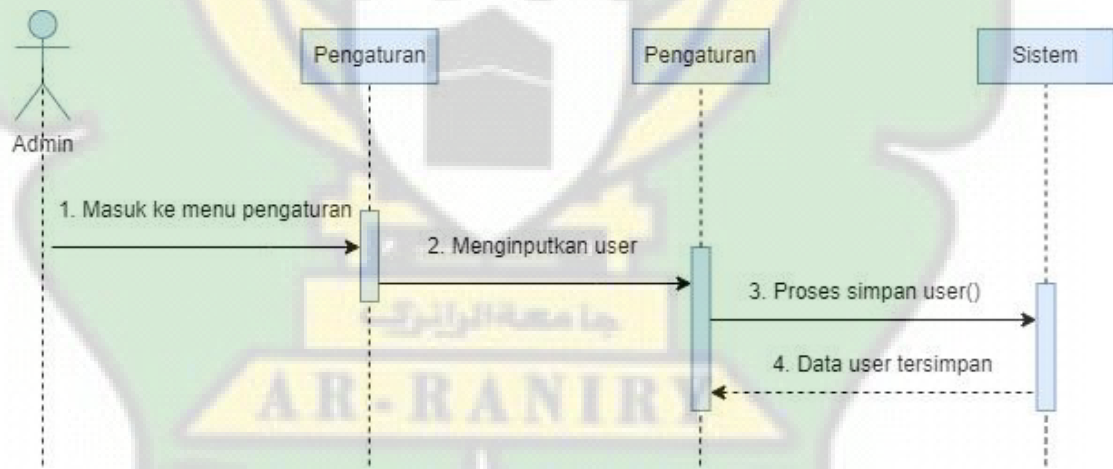
Langkah pertama yang harus administrator dan *user* lakukan ialah menginput atau memasukkan *username* dan *password* dalm field input lalu sistem akan membaca apakah data administrator yang dimasukkan benar atau salah, setelah

validasi data sistem akan menampilkan halaman menu *home* atau tampilan *dashboard* yang dapat dilihat dan dikelola oleh *administrator* dan *user*.



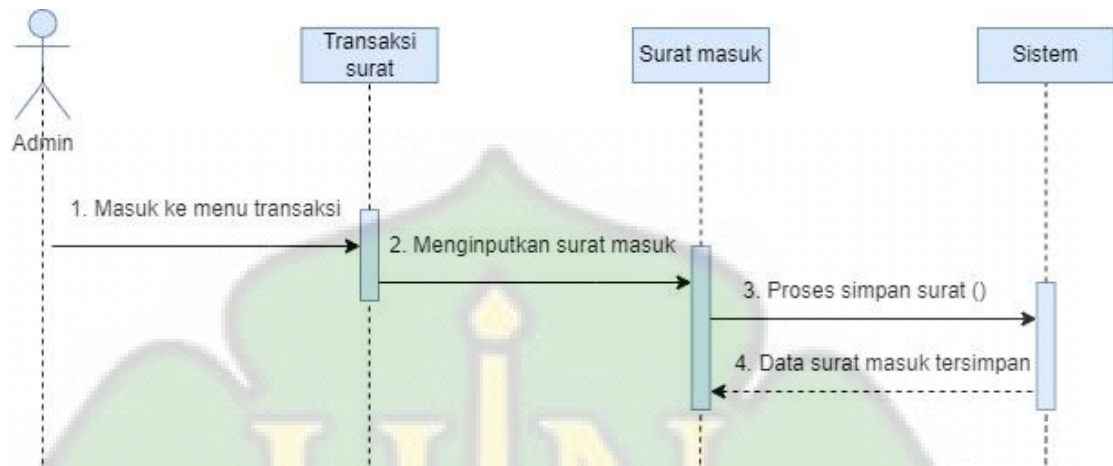
Gambar III.19 : *Sequence diagram* login

b. *Sequence diagram* pengaturan



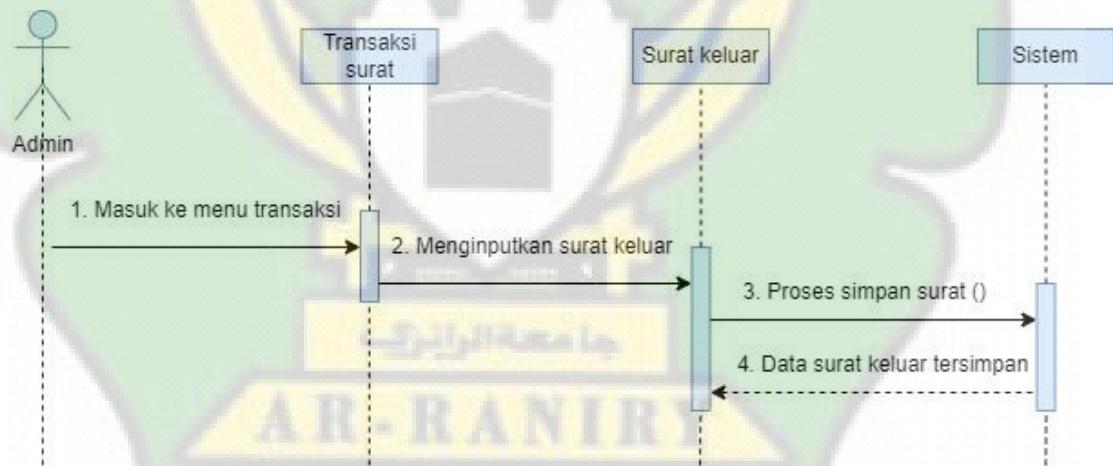
Gambar III.20 : *Sequence diagram* pengaturan

c. *Sequence diagram* transaksi surat masuk



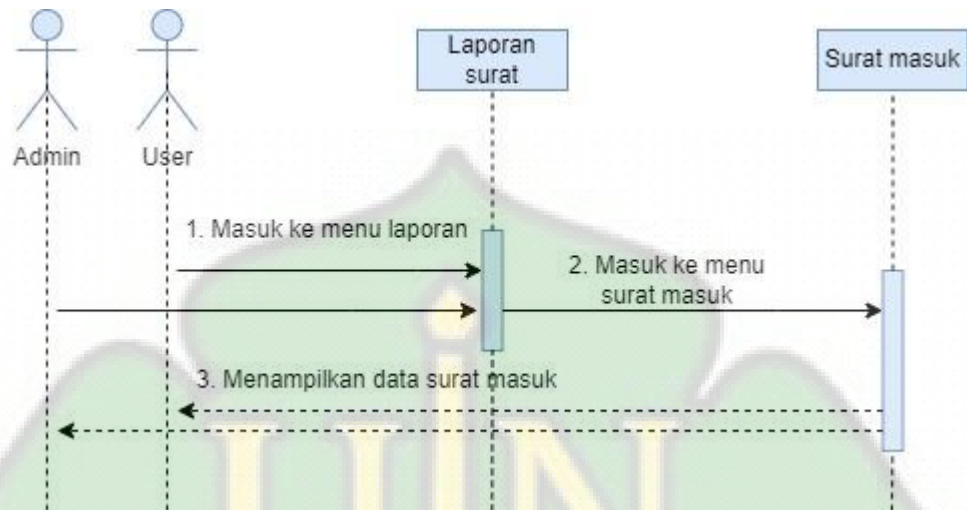
Gambar III.21 : *Sequence diagram* transaksi surat masuk

d. *Sequence diagram* transaksi surat surat keluar



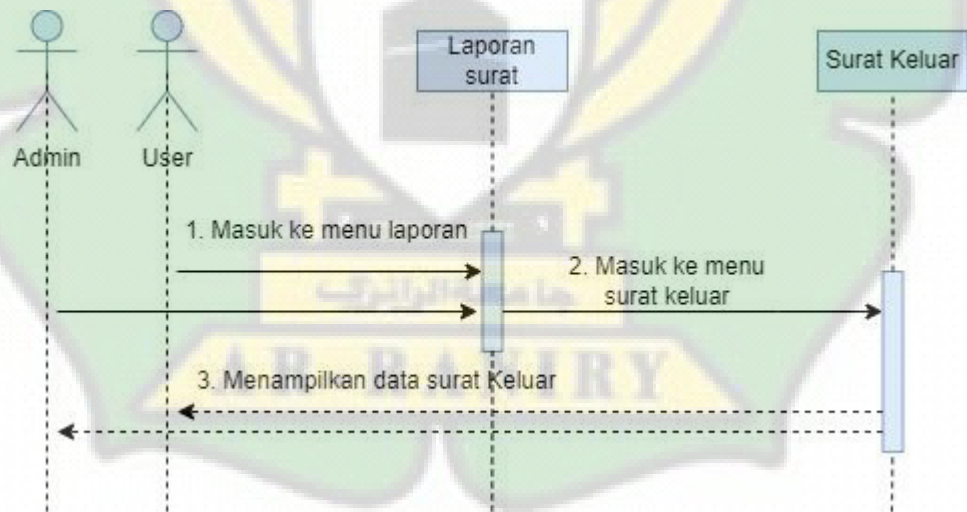
Gambar III.22 : *Sequence diagram* transaksi surat keluar

e. *Sequence diagram* laporan surat masuk



Gambar III.23 : *Sequence diagram* laporan surat masuk

f. *Sequence diagram* laporan surat keluar



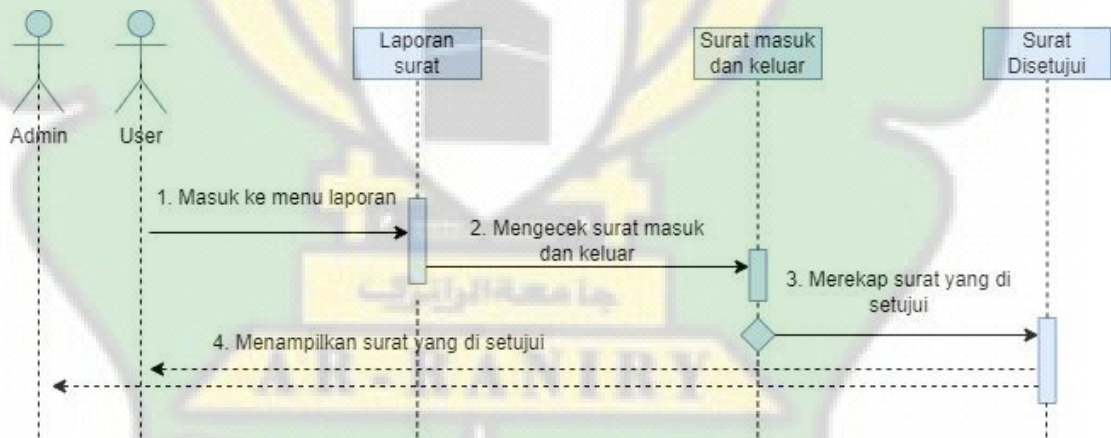
Gambar III.24 : *Sequence diagram* laporan surat keluar

g. *Sequence diagram* laporan surat disposisi



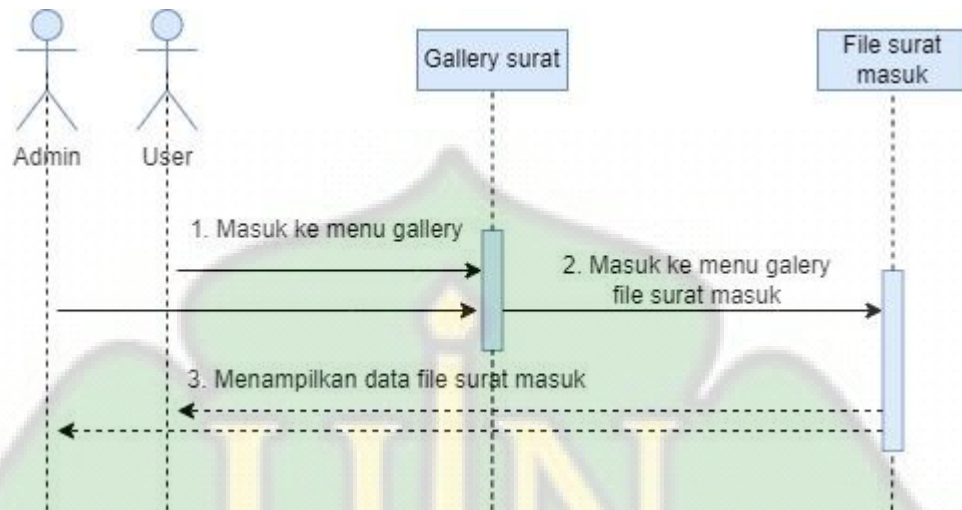
Gambar III.25 : *Sequence diagram* laporan surat disposisi

h. *Sequence diagram* laporan surat di setuju



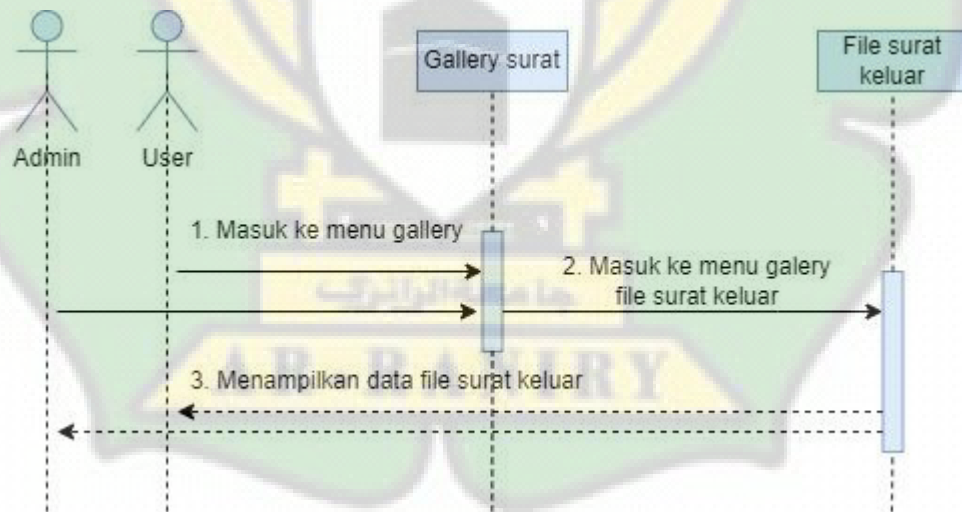
Gambar III.26 : *Sequence diagram* laporan surat di setuju

i. *Sequence diagram* gallery file surat masuk



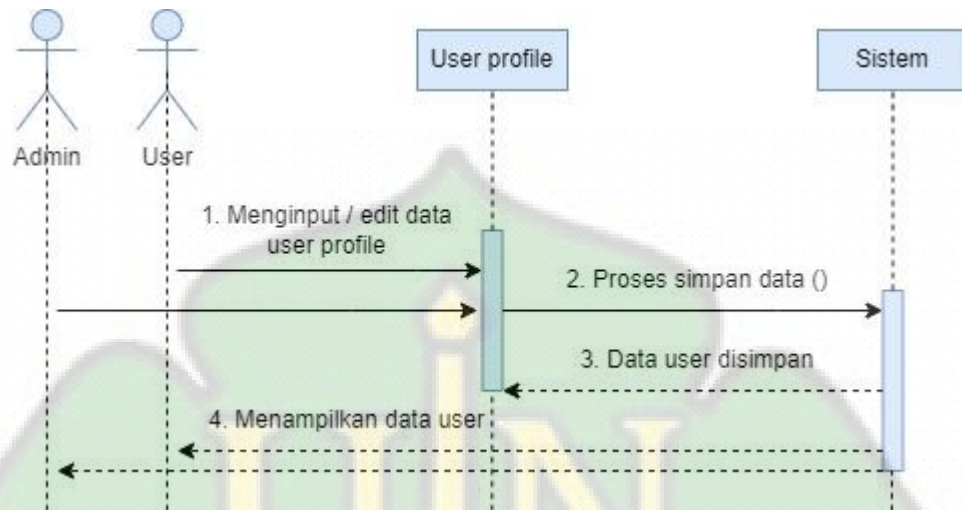
Gambar III.27 : *Sequence diagram* gallery file surat masuk

j. *Sequence diagram* gallery file surat keluar



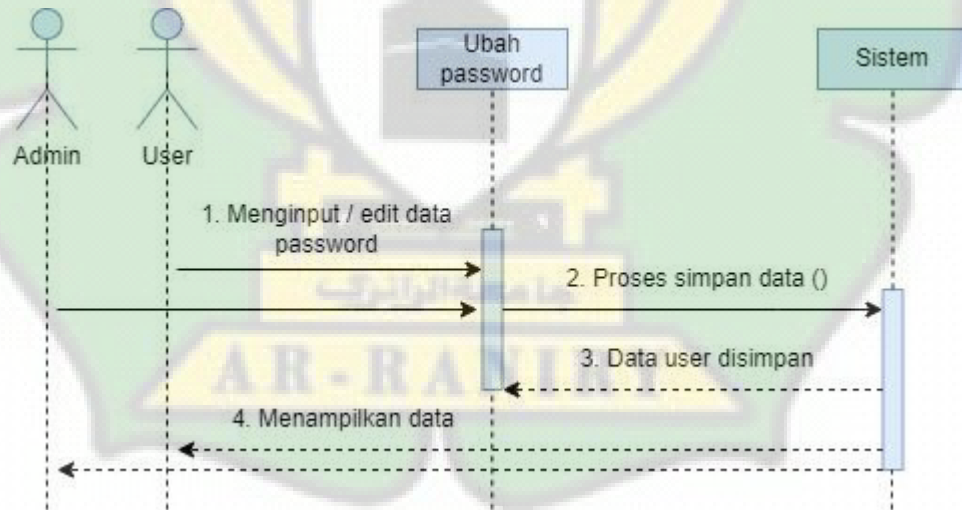
Gambar III.28 : *Sequence diagram* gallery file surat keluar

k. *Sequence diagram user profile*



Gambar III.29 : *Sequence diagram user profile*

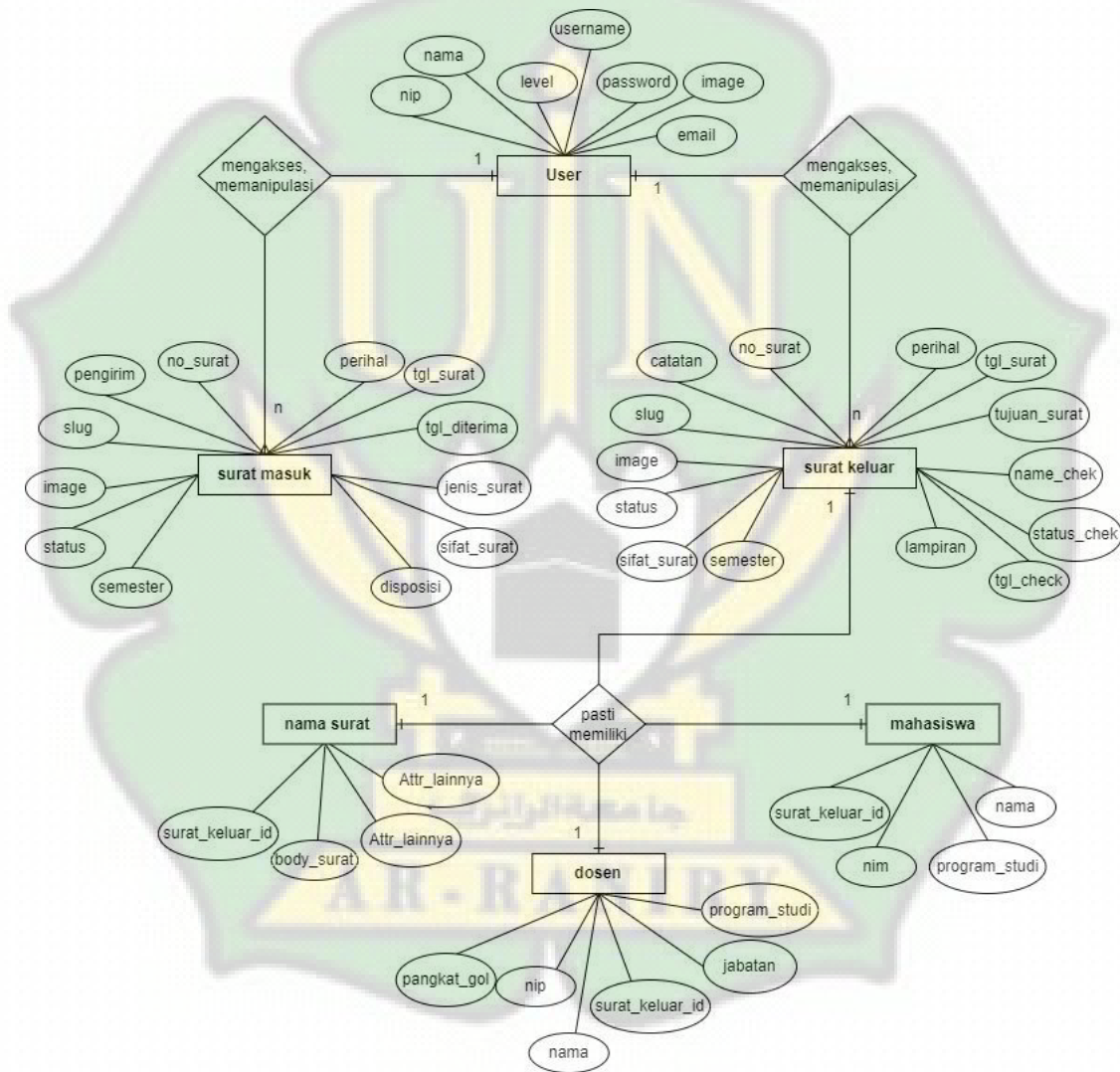
1. *Sequence diagram ubah password*



Gambar III.30 : *Sequence diagram ubah password*

III.4.4 Entity Relationship Diagram

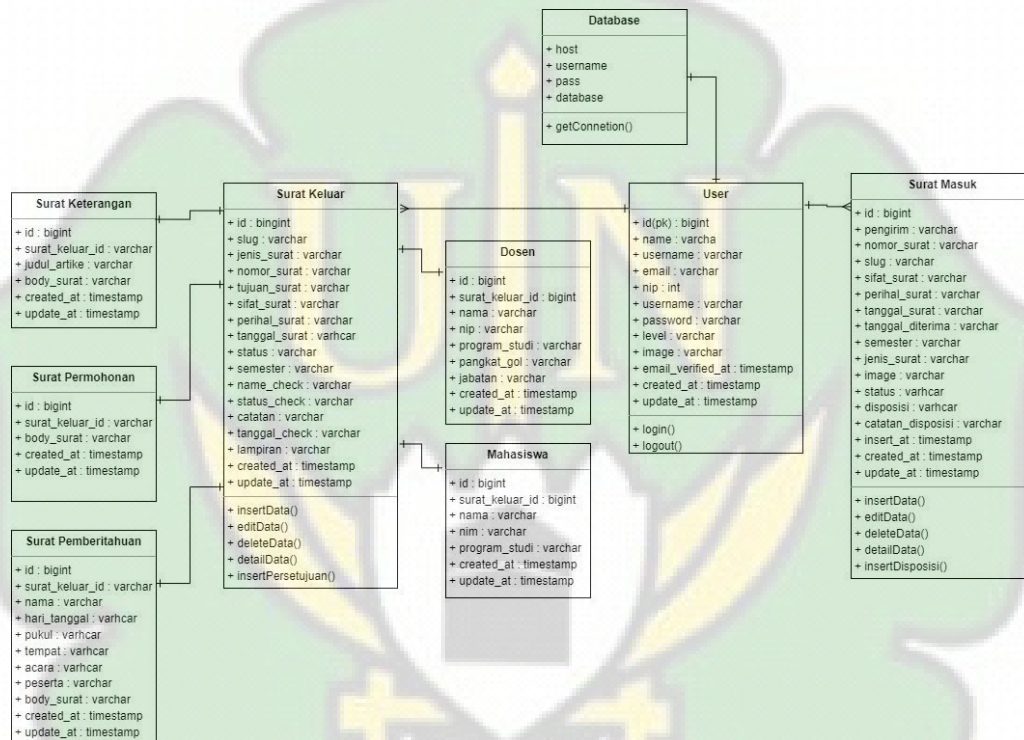
ERD (*Entity Relationship Diagram*) atau diagram hubungan entitas merupakan diagram yang digunakan untuk merancang sebuah *database* dengan menunjukkan koneksi atau hubungan antara objek dan entitas beserta atribut-atributnya secara detail. Dengan menggunakan ERD, sistem *database* yang sedang dibentuk dapat digambarkan dengan lebih terstruktur dan terlihat rapi.



Gambar III 31 : Entity Relationship Diagram surat

III.4.5 Class Diagram

Terlihat pada gambar yang tertera di bawah ini, sebuah class diagram yang dirancang. Class diagram ini memiliki beberapa objek, seperti objek database, user, surat masuk, surat keluar, dosen, mahasiswa, surat keterangan, surat permohonan dan surat pemberitahuan. Setiap objek memiliki atribut dan metode atau operasi yang dapat diterapkan pada objek tersebut."



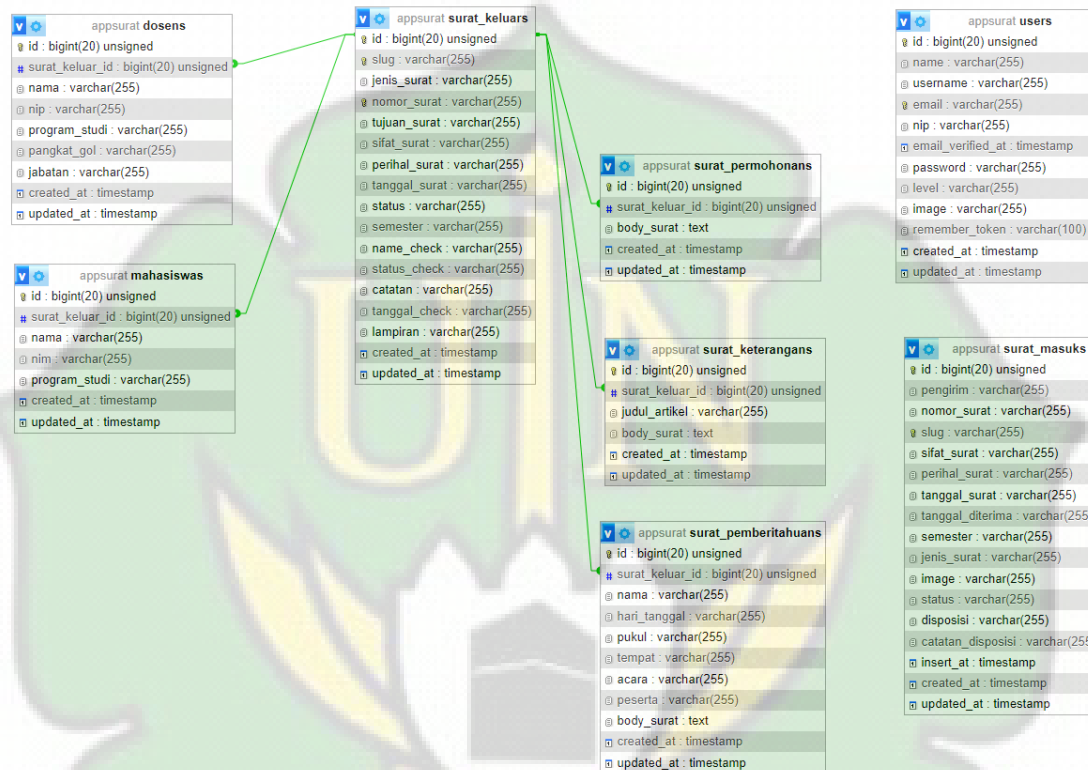
Gambar III.32 : Class diagram surat

Relasi kardinalitas :

1. User memiliki relasi kardinalitas *one to many* ke tabel surat masuk
2. User memiliki relasi kardinalitas *one to many* ke tabel surat keluar
3. Surat keluar memiliki relasi kardinalitas *one to one* ke tabel dosen, mahasiswa, surat keterangan, surat permohonan dan surat pemberitahuan, meninjau dari penggunaan framework laravel untuk kardinalitas ini disebutkan sebagai (*hasOne*) dan *inverse* nya (*belongsTo*).

III.4.6 Database

Perancangan *database* atau basis data dibuat untuk memudahkan penyimpanan, perubahan dan pembacaan data. Pada gambar di bawah ini dapat dilihat rancangan *database* yang di buat pada *phpmyadmin* untuk sistem informasi pengarsipan surat.



Gambar III.33 : Rancangan database surat

Dapat dilihat dari hasil perancangan *database* di atas didapatkan delapan table, yaitu tiga tabel utama yang meliputi tabel user, surat masuk, surat keluar dan tabel yang berelasi dengan surat keluar diantaranya tabel mahasiswa, dosen, surat keterangan, surat permohonan dan surat pemberitahuan.

Tabel III.1 : Tabel user

Field name	Data type	length	Description
id	bigint	20	<i>Primary key</i>
name	varchar	30	
username	varchar	15	
email	varchar	35	
nip	varchar	17	
password	varchar	255	Enkripsi
level	varchar	8	
image	varchar		
created_at	timestamp		
updated_at	timestamp		

Tabel III.2 : Tabel surat masuk

Field name	Data type	length	Description
id	bigint	20	<i>Primary key</i>
pengirim	varchar	150	
nomor_surat	varchar	35	
sifat_surat	varchar	10	
slug	varchar	150	
perihal_surat	varchar	255	
tanggal_surat	varchar	10	
tanggal_diterima	varchar	10	
semester	varchar	7	
jenis_surat	varchar	30	
image	varchar	255	
Status	varchar	20	
disposisi	varchar	100	
catatan_disposisi	varchar	150	

insert_at	timestamp		
created_at	timestamp		
updated_at	timestamp		

Tabel III.3 : Tabel surat keluar

Field name	Data type	length	Description
id	bigint	20	<i>Primary key</i>
slug	varchar	150	
jenis_surat	varchar	30	
nomor_surat	varchar	35	
tujuan_surat	varchar	150	
sifat_surat	varchar	10	
perihal_surat	varchar	255	
tanggal_surat	varchar	10	
status	varchar	15	
semester	varchar	7	
name_check	varchar	35	
status_check	varchar	15	
catatan	varchar	50	
tanggal_check	varchar	10	
lampiran	varchar	2	
created_at	timestamp		
updated_at	timestamp		

Tabel III.4 : Tabel mahasiswa

Field name	Data type	length	Description
Id	bigint	20	Primary key
surat_keluar_id	bigint	20	Foreign key
nama	varchar	35	
nim	varchar	12	
program_studi	varchar	30	
created_at	timestamp		
updated_at	timestamp		

Tabel III.5 : Tabel dosen

Field name	Data type	length	Description
id	bigint	20	Primary key
id_surat_keluar	bigint	20	Foreign key
nama	varchar	35	
nip	varchar	17	
program_studi	varchar	30	
pangkat_gol	varchar	30	
jabatan	varchar	35	
created_at	timestamp		
updated_at	timestamp		

Tabel III.6 : Tabel surat keterangan

Field name	Data type	length	Description
id	bigint	20	Primary key
surat_keluar_id	bigint	20	Foreign key
body_surat	text		

created_at	timestamp		
updated_at	timestamp		

Tabel III.7 : Tabel surat permohonan

Field name	Data type	length	Description
id	bigint	20	<i>Primary key</i>
surat_keluar_id	bigint	20	Foreign key
body_surat	text		
created_at	timestamp		
updated_at	timestamp		

Tabel III.8 : Tabel surat pemberitahuan

Field name	Data type	length	Description
id	bigint	20	<i>Primary key</i>
surat_keluar_id	bigint	20	Foreign key
hari_tanggal	varchar	25	
pukul	varchar	25	
tempat	varchar	100	
acara	varchar	150	
peserta	varchar	50	
body_surat	text		
created_at	timestamp		
updated_at	timestamp		

BAB IV

HASIL DAN PEMBAHASAN

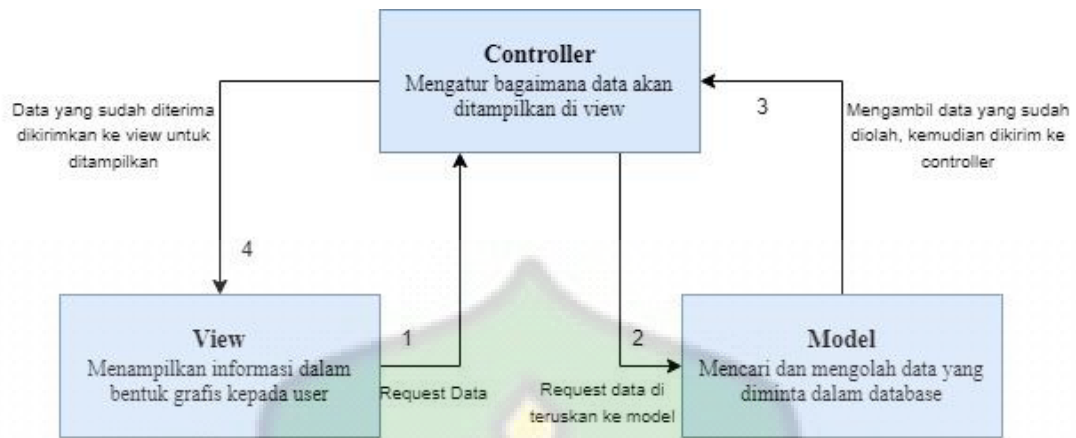
IV.1 Hasil Implementasi dan Pembahasan

Implementasi dan uji coba dilakukan pada sistem informasi surat keluar masuk dengan tujuan untuk mengetahui kekurangan yang ada pada sistem sebelum sistem digunakan oleh pengguna. Hasil dari pengujian jika ditemukan kelemahan pada bagian sistem maka pengguna bisa mengembangkan pada bagian sistem tersebut yang dianggap lemah. Tahap implementasi sistem merupakan proses membangun sistem sesuai dengan rancangan dan desain sistem yang telah dipersiapkan atau dibuat sebelumnya sekaligus pengujian bagi sistem yang baru untuk dioperasikan pada keadaan yang sebenarnya sehingga efektifitas sistem akan diketahui secara pasti. Tahapan ini dapat dibagi menjadi beberapa bagian sesuai dengan fungsi-fungsi yang dibuat.

IV.1.1 MVC (*Model, View, Controller*)

MVC (*Model, View, Controller*) merupakan metode yang digunakan untuk merancang sebuah aplikasi dengan cara memisahkan tiga bagian, Model, View, dan Controller. walaupun dipisahkan tiga bagian ini tetap berkaitan satu sama lain hal ini jadi memiliki banyak keuntungan pada saat pengembangan dan pemeliharaan aplikasi.

- Model, Bagian yang bertugas untuk mengatur, menyiapkan, memanipulasi dan mengorganisasikan data yang terdapat dalam sebuah database.
- View, Bagian yang bertugas untuk menampilkan informasi atau mengatur tampilan dalam bentuk GUI (Graphical User Interface)
- Controller, Bagian yang bertugas untuk menghubungkan serta mengatur model dan view agar dapat saling terhubung.



Gambar IV.1 : Alur MVC

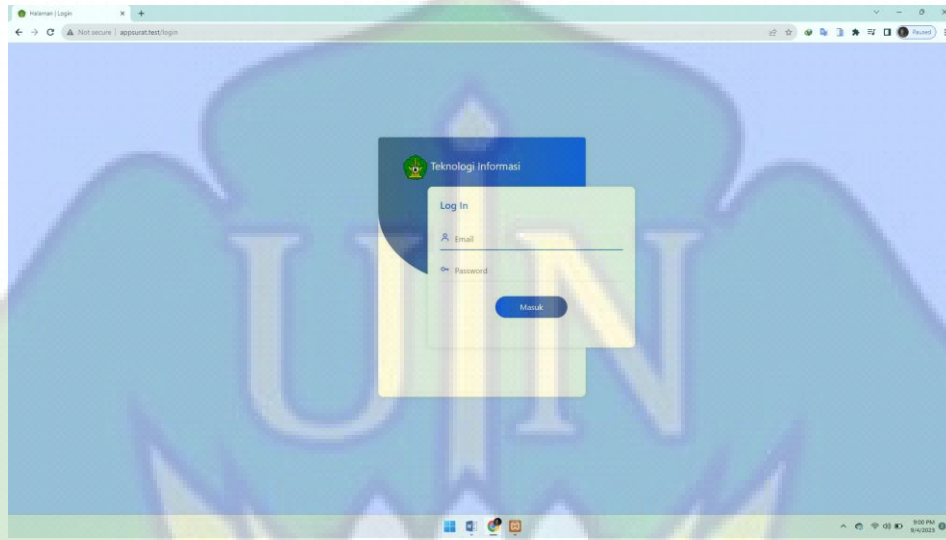
Berdasarkan gambar IV.1 diatas, dapat dijelaskan alur dari MVC adalah sebagai berikut:

1. View akan meminta data ke controller untuk ditampilkan dalam bentuk grafis kepada pengguna.
2. Proses tersebut akan diterima oleh controller dan akan diteruskan ke model untuk diproses datanya.
3. Model akan mengolah data yang diminta oleh controller dalam databasenya.
4. Setelah data sudah diolah, model akan mengirimkan data tersebut ke controller.
5. Controller akan mengambil data yang sudah di olah kemudian mengaturnya di bagian view untuk diperlihatkan kepada pengguna.

IV.2 Tampilan Aplikasi

a. Tampilan halaman login

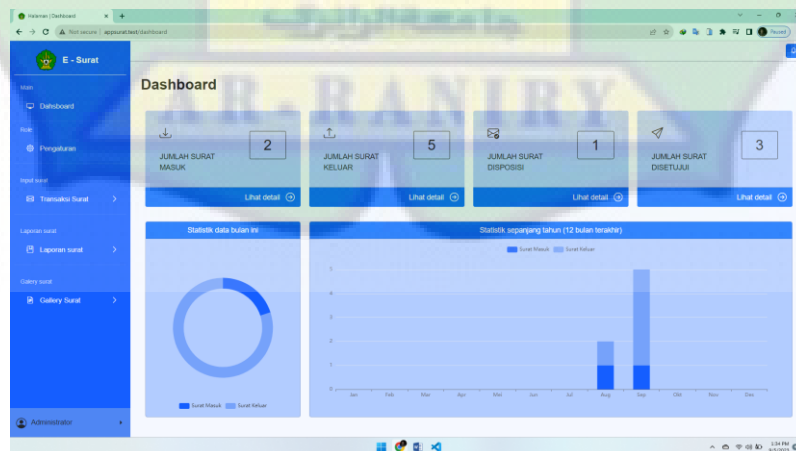
Halaman login untuk pengarsipan surat masuk dan surat keluar digunakan untuk memberikan akses terbatas ke sistem pengarsipan surat kepada pengguna, baik mereka adalah administrator (admin) atau pengguna lain (user).



Gambar IV.2 : Halaman Login

b. Tampilan halaman dashboard

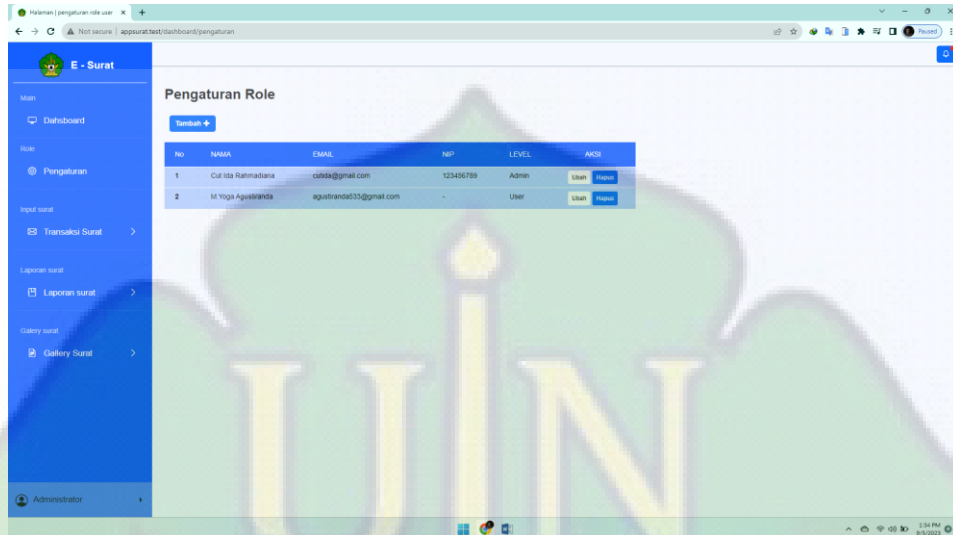
Halaman dashboard, setelah *user* melakukan login maka akan dialihkan ke halaman dashboard ini. Pada halaman ini terdapat beberapa bagian yaitu menu dan konten. Menu terdapat pada bagian kiri dari *website*.



Gambar IV.3 : Halaman Dashboard

c. Tampilan halaman pengaturan

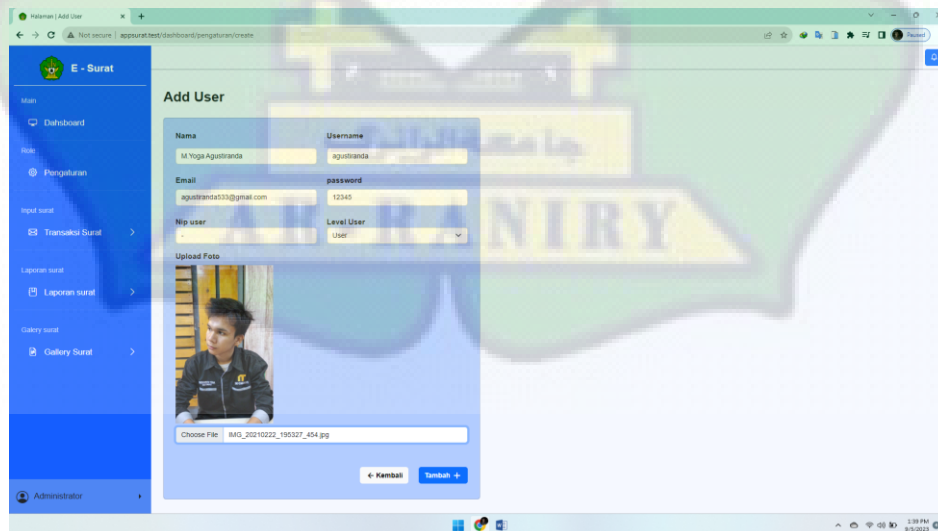
Halaman pengaturan ini hanya dapat diakses oleh admin *website* sendiri, halaman ini terdapat informasi yang terdaftar di dalam *website*.



Gambar IV.4 : Halaman pengaturan

- Halaman pengaturan aksi tambah

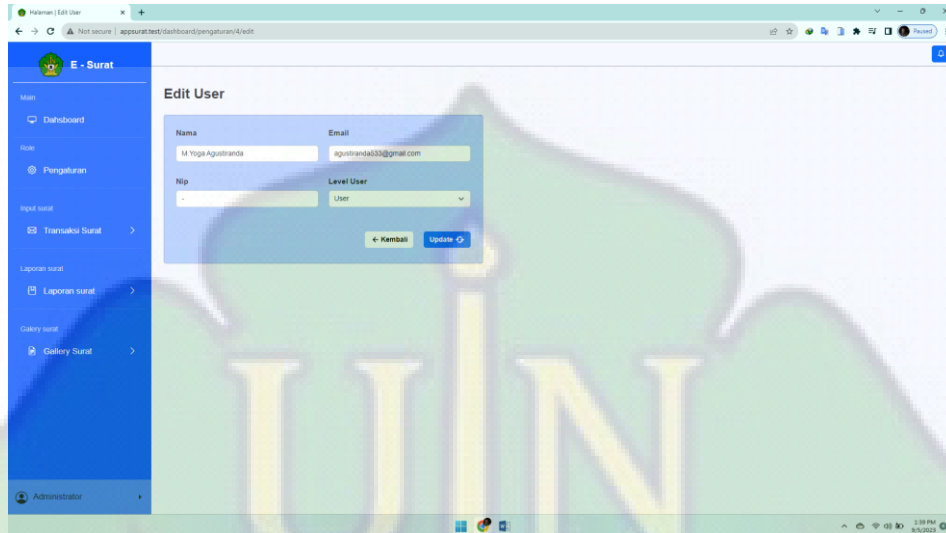
Halaman tambah pengguna (*add user*) juga hanya dapat diakses oleh admin, halaman muncul jika admin menekan tombol tambah pada halaman sebelumnya.



Gambar IV.5 : Halaman pengaturan aksi tambah user

- Halaman pengaturan aksi ubah

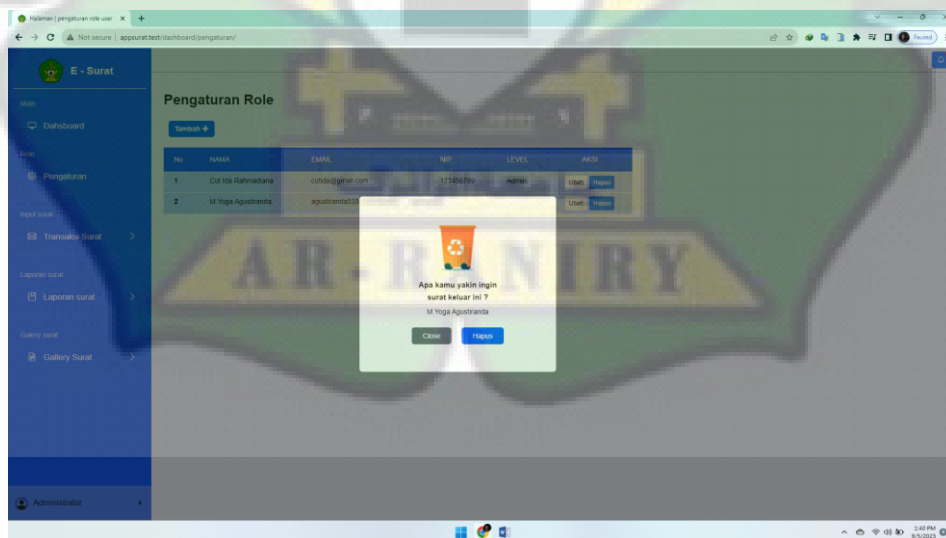
Halaman edit pengguna, fungsinya mengedit hak akses pengguna yang terdaftar di dalam *website*, mengedit pengguna hanya dapat dilakukan oleh admin.



Gambar IV.6 : Halaman pengaturan aksi edit user

- Halaman pengaturan aksi hapus

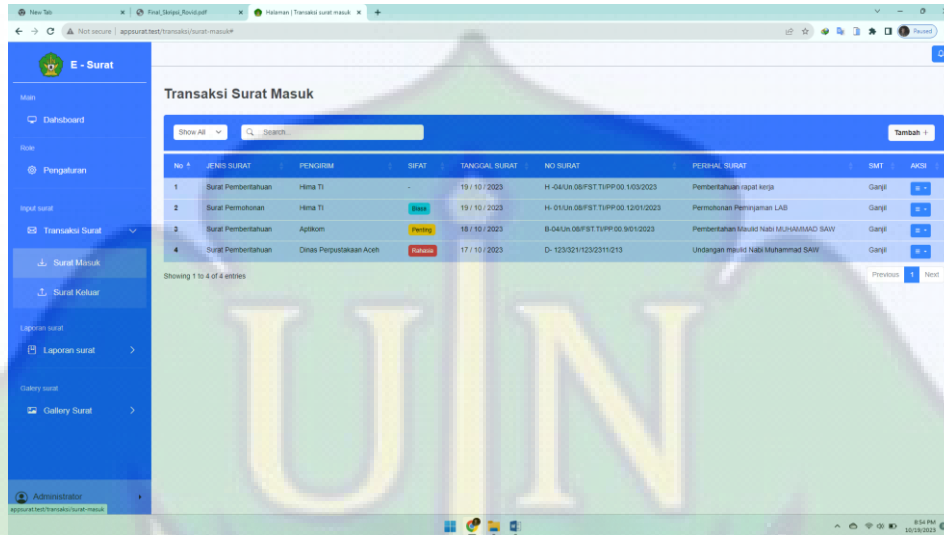
Halaman hapus pengguna, fungsinya menghapus pengguna yang terdaftar di dalam *website*, menghapus pengguna hanya dapat dilakukan oleh admin.



Gambar IV.7 : Halaman pengaturan aksi hapus user

d. Tampilan halaman transaksi surat masuk

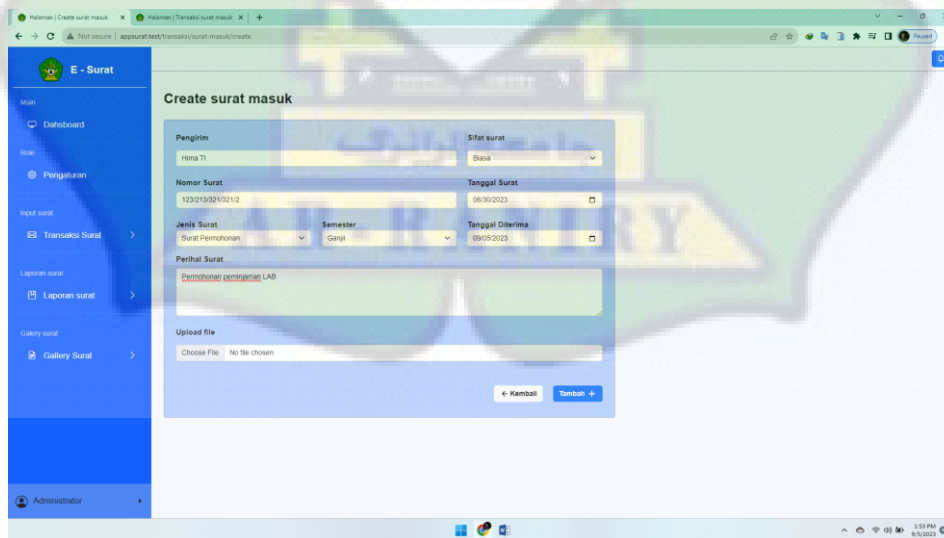
Halaman ini terdapat beberapa informasi keseluruhan surat masuk yang diinput oleh admin ke dalam *website*, admin juga bisa melakukan sorting surat, dan mencari surat.



Gambar IV.8 : Halaman tampilan transaksi surat masuk

- Halaman transaksi surat masuk aksi tambah

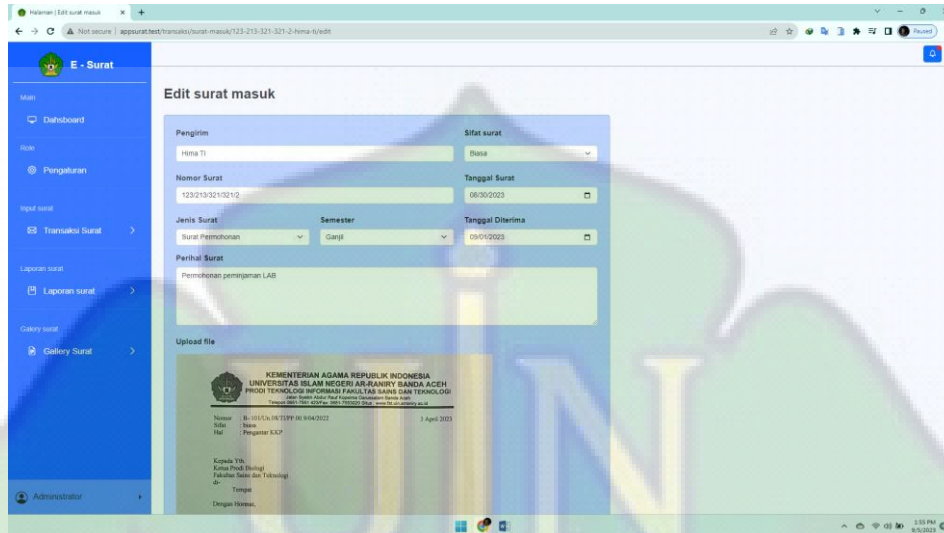
Halaman tambah surat masuk hanya dapat diakses oleh admin, halaman ini akan muncul jika admin menekan tombol tambah pada halaman sebelumnya.



Gambar IV.9 : Halaman transaksi surat masuk aksi tambah

- Halaman transaksi surat masuk aksi edit

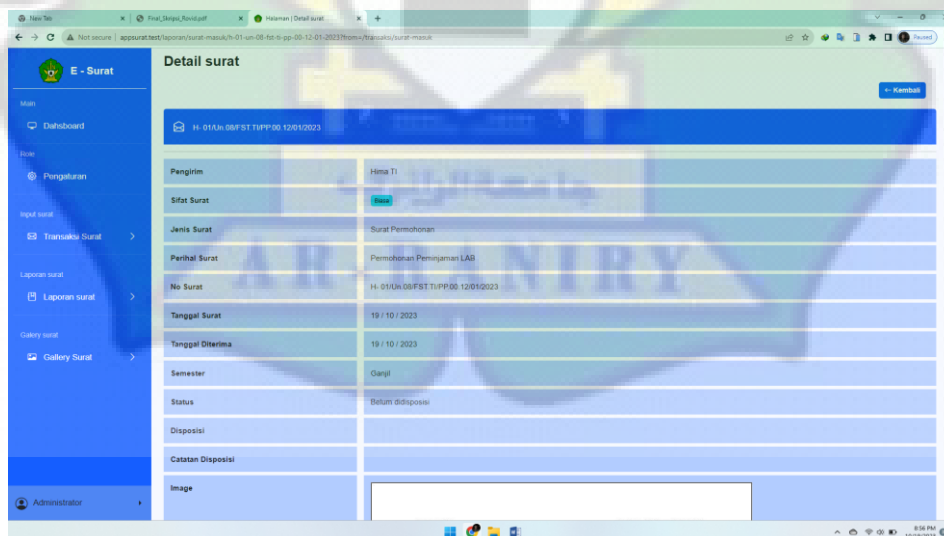
Halaman edit surat masuk hanya dapat diakses oleh admin, halaman ini akan muncul jika admin menekan tombol edit pada halaman sebelumnya.



Gambar IV.10 : Halaman transaksi surat masuk aksi edit

- Halaman transaksi surat masuk aksi detail

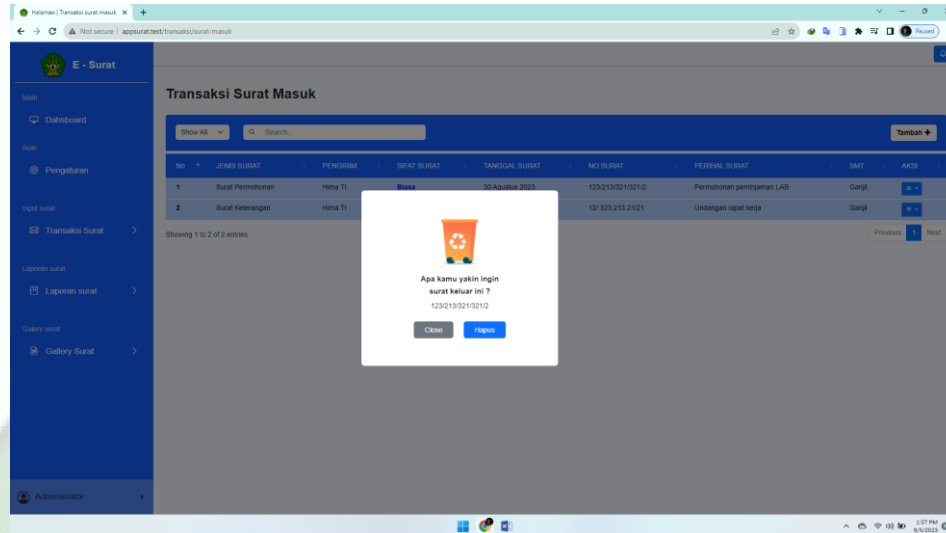
Halaman detail surat masuk, halaman ini akan muncul jika admin menekan tombol detail, halaman ini menampilkan informasi detail dari surat masuk.



Gambar IV.11 : Halaman transaksi surat masuk aksi detail

- Halaman transaksi surat masuk aksi hapus

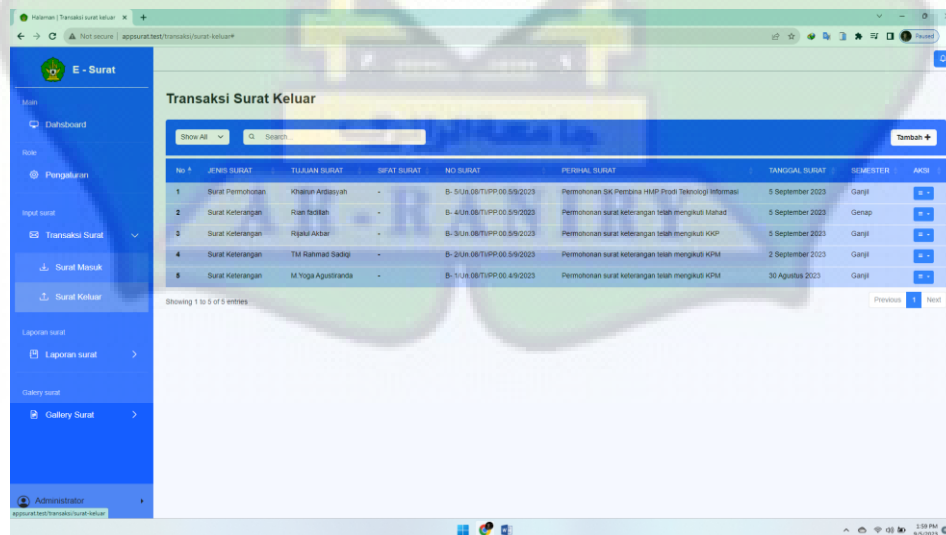
Halaman hapus surat masuk, halaman ini akan muncul jika admin menekan tombol hapus pada aksi yang ada pada halaman surat masuk sebelumnya.



Gambar IV.12 : Halaman transaksi surat masuk aksi hapus

- e. Tampilan halaman transaksi surat keluar

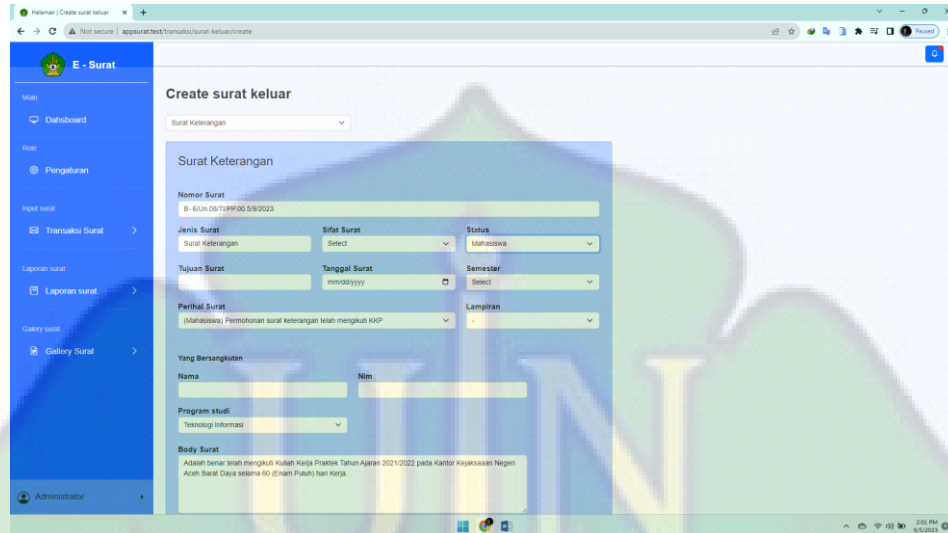
Halaman ini terdapat beberapa informasi keseluruhan surat keluar yang diinput oleh admin ke dalam *website*, admin juga bisa melakukan sorting surat, dan mencari surat.



Gambar IV.13 : Halaman tampilan transaksi surat keluar

- Halaman transaksi surat keluar aksi tambah

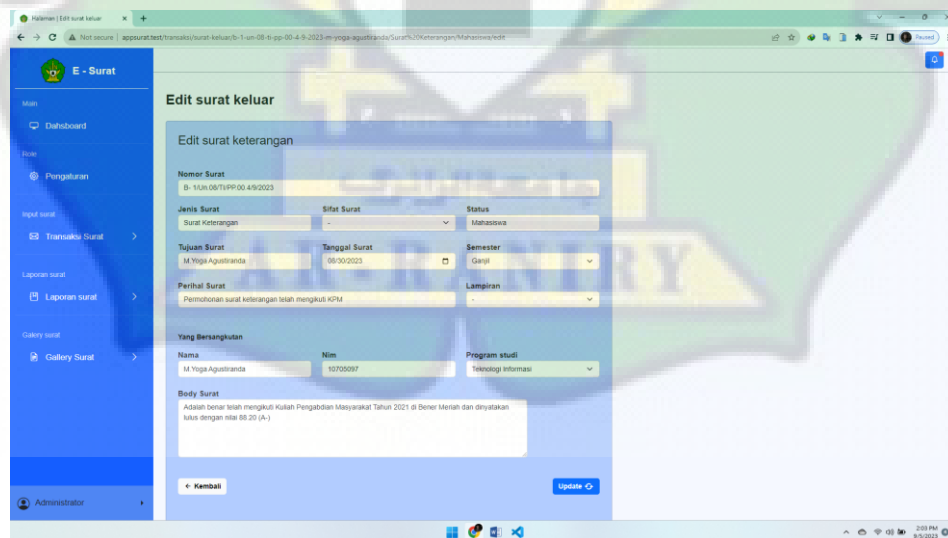
Halaman tambah surat keluar hanya dapat diakses oleh admin, halaman ini akan muncul jika admin menekan tombol tambah pada halaman sebelumnya.



Gambar IV.14 : Halaman transaksi surat keluar aksi tambah

- Halaman transaksi surat keluar aksi edit

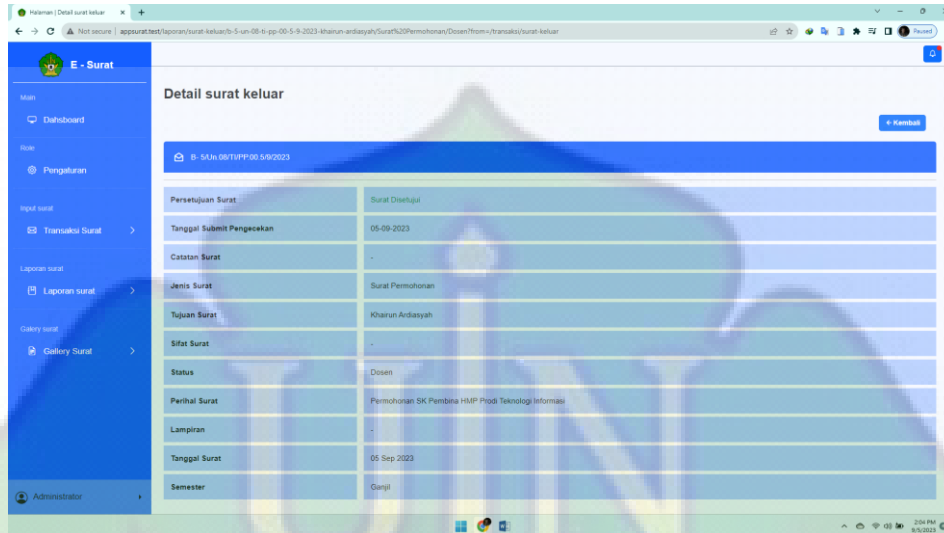
Halaman edit surat keluar hanya dapat diakses oleh admin, halaman ini akan muncul jika admin menekan tombol edit pada halaman sebelumnya.



Gambar IV.15 : Halaman transaksi surat keluar aksi edit

- Halaman transaksi surat keluar aksi detail

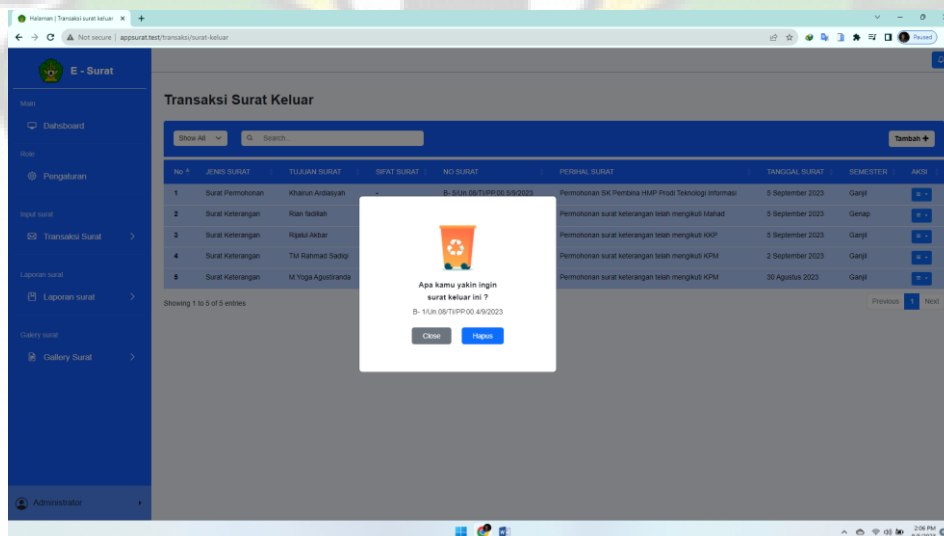
Halaman detail surat keluar, halaman ini akan muncul jika admin menekan tombol detail, halaman ini menampilkan informasi detail dari surat keluar.



Gambar IV.16 : Halaman transaksi surat keluar aksi detail

- Halaman transaksi surat keluar aksi hapus

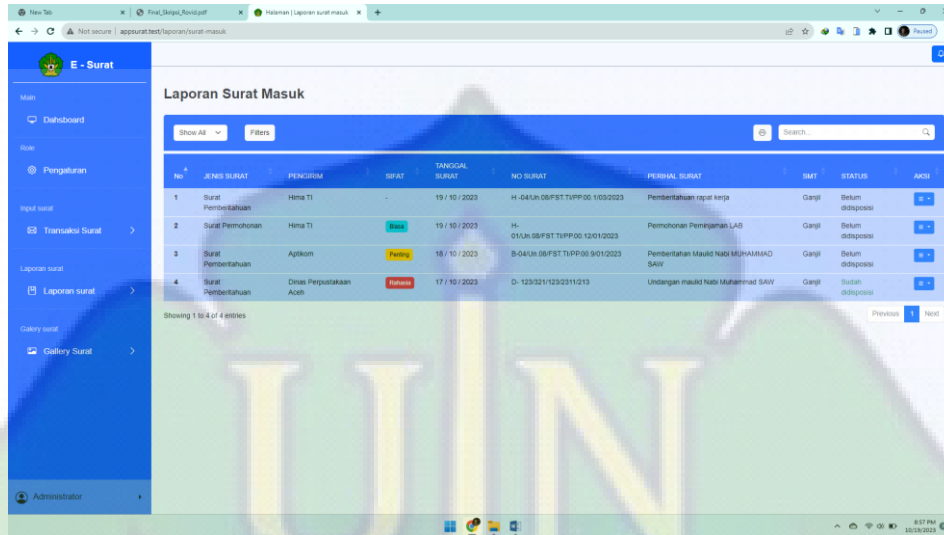
Halaman hapus surat keluar, halaman ini akan muncul jika admin menekan tombol hapus pada aksi yang ada pada halaman surat keluar.



Gambar IV.17 : Halaman transaksi surat keluar aksi hapus

f. Tampilan halaman laporan surat masuk

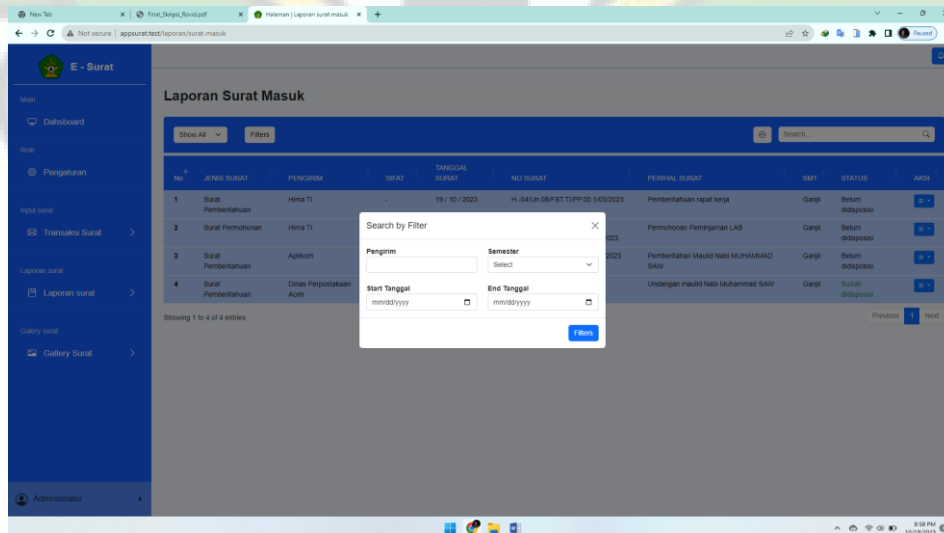
Halaman laporan surat masuk ini akan menampilkan list surat masuk yang sudah dan belum didisposisi oleh *user* atau kaprodi TI.



Gambar IV.18 : Halaman laporan tampilan surat masuk

- Halaman laporan surat masuk aksi filter surat

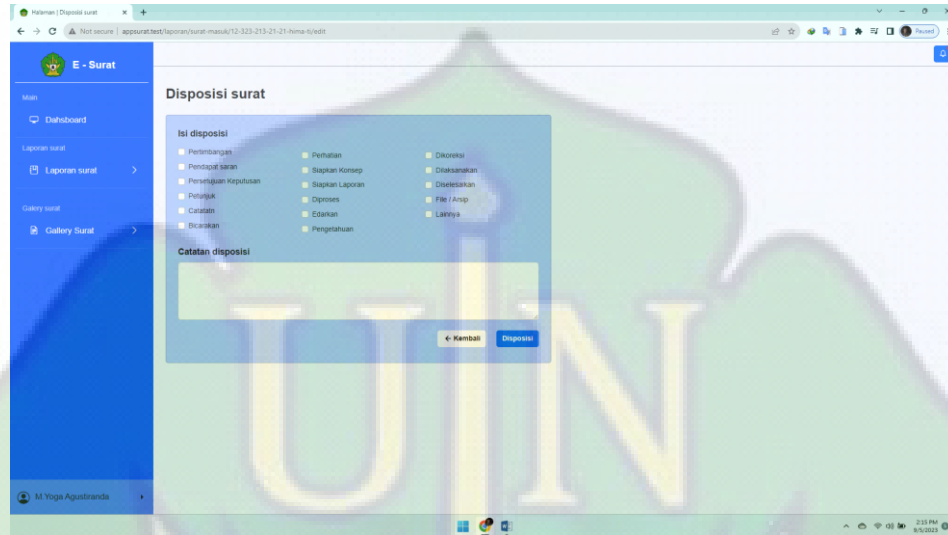
Halaman laporan surat masuk aksi filter ini akan menampilkan *list* surat masuk yang di *searching* sesuai dengan range inputan pencarian.



Gambar IV.19 : Halaman laporan surat masuk aksi filter

- Halaman laporan surat masuk aksi disposisi

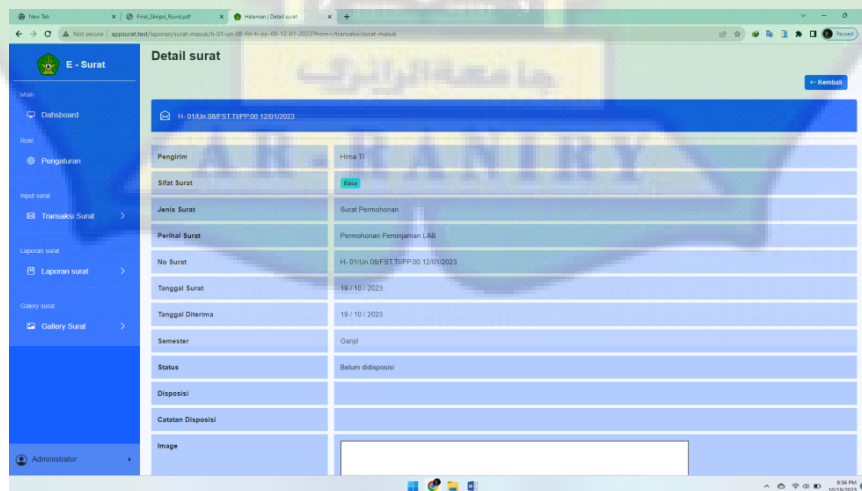
Halaman disposisi surat masuk hanya dapat diakses oleh kaprodi Ti (*user*), halaman ini akan muncul jika *user* menekan tombol disposisi pada halaman sebelumnya.



Gambar IV.20 : Halaman laporan surat masuk aksi disposisi

- Halaman laporan surat masuk aksi detail

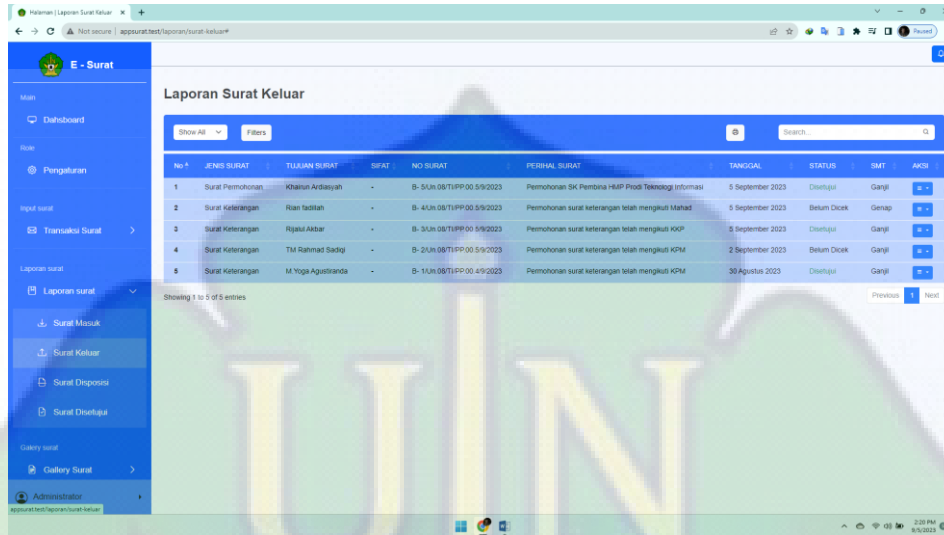
Halaman detail laporan surat masuk, halaman ini akan muncul jika admin menekan tombol detail, halaman ini menampilkan informasi detail dari surat masuk.



Gambar IV.21 : Halaman laporan surat masuk aksi detail

g. Tampilan halaman laporan surat keluar

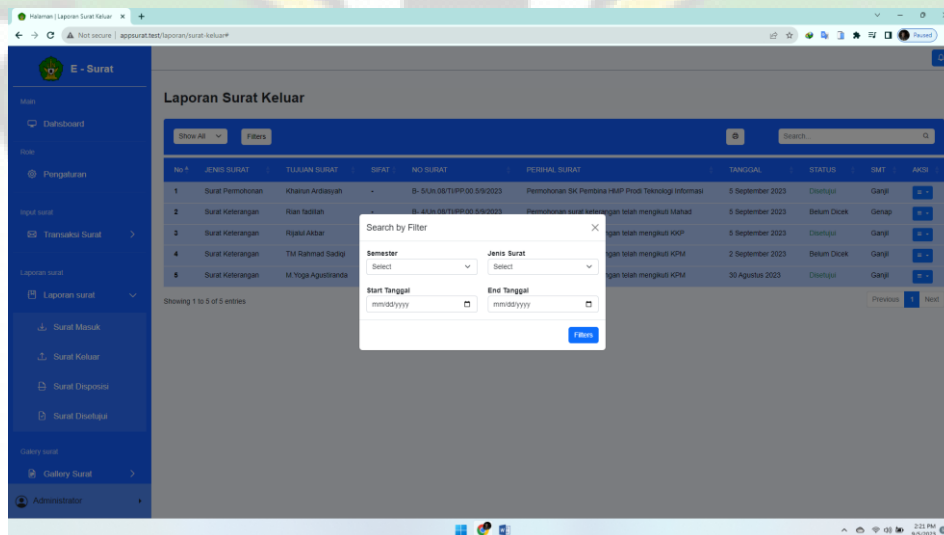
Halaman laporan surat keluar ini akan menampilkan *list* surat keluar yang sudah dan belum disetujui oleh user atau kaprodi TI.



Gambar IV.22 : Tampilan halaman laporan surat keluar

- Halaman laporan surat keluar aksi filter

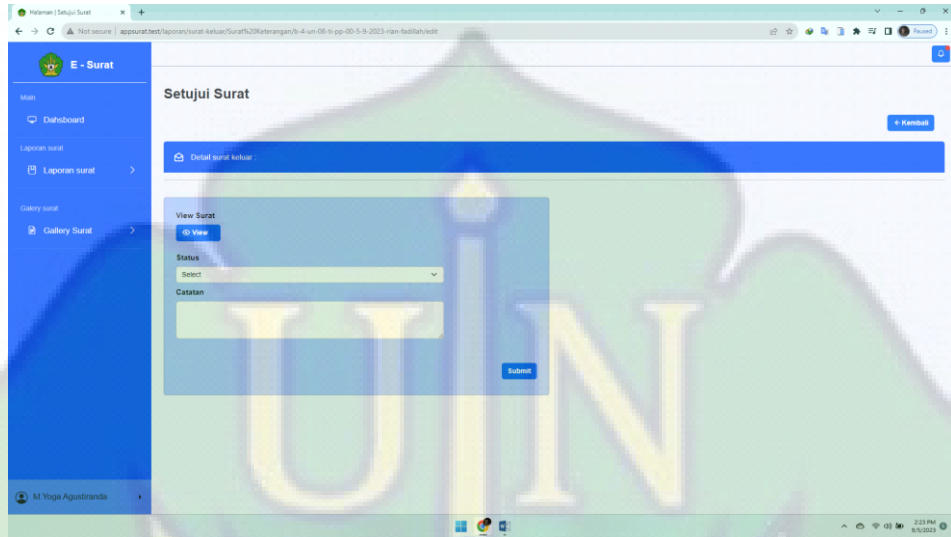
Halaman laporan surat keluar aksi filter ini akan menampilkan list surat keluar yang di *searching* sesuai dengan *range* inputan pencarian.



Gambar IV.23 : Halaman laporan surat keluar aksi filter

- Halaman laporan surat keluar aksi setuju surat

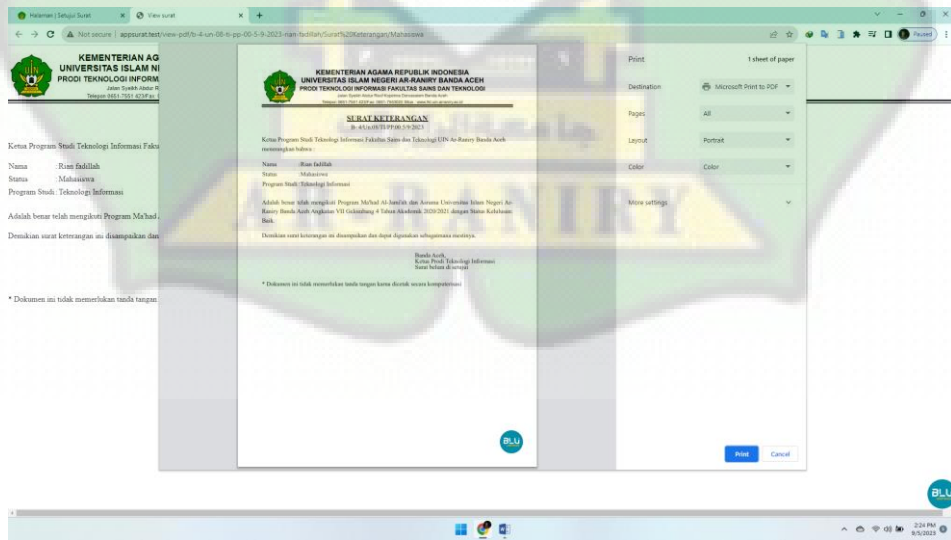
Halaman setuju surat keluar hanya dapat diakses oleh kaprodi Ti (*user*), halaman ini akan muncul jika *user* menekan tombol setuju pada halaman sebelumnya.



Gambar IV.24 : Halaman laporan surat keluar aksi setuju surat

- Halaman laporan surat keluar aksi *view pdf* surat disetujui

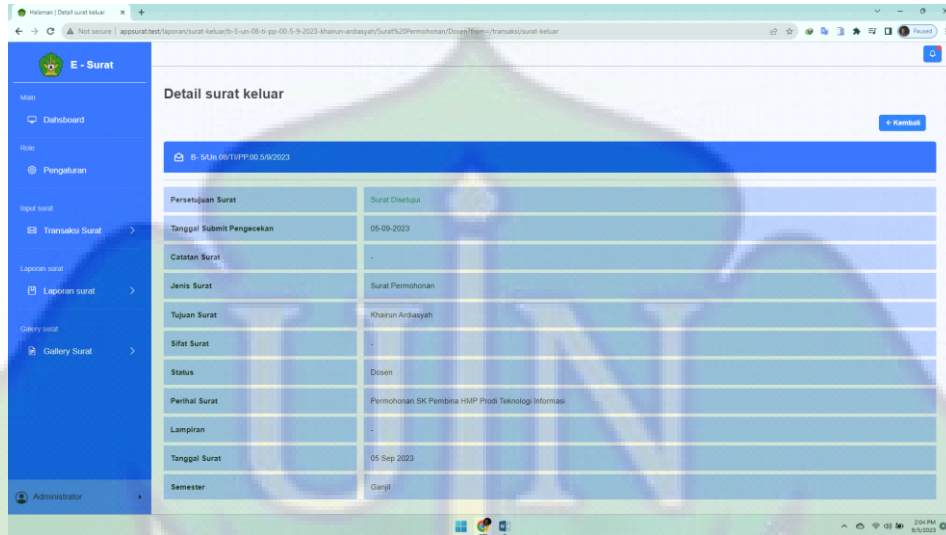
Halaman ini akan muncul jika admin menekan tombol untuk melihat surat dalam bentuk pdf, halaman ini menampilkan informasi dalam bentuk format print.



Gambar IV.25 : Halaman laporan surat keluar aksi *view pdf*

- Halaman laporan surat keluar aksi detail

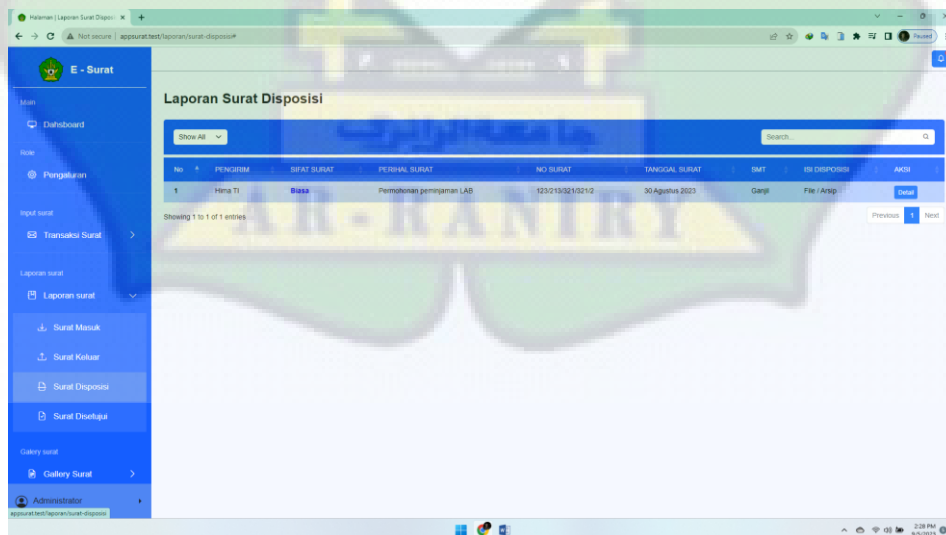
Halaman detail laporan surat keluar, halaman ini akan muncul jika admin menekan tombol detail, halaman ini menampilkan informasi detail dari surat keluar.



Gambar IV.26 : Halaman laporan surat keluar aksi detail

- h. Tampilan halaman laporan surat disposisi

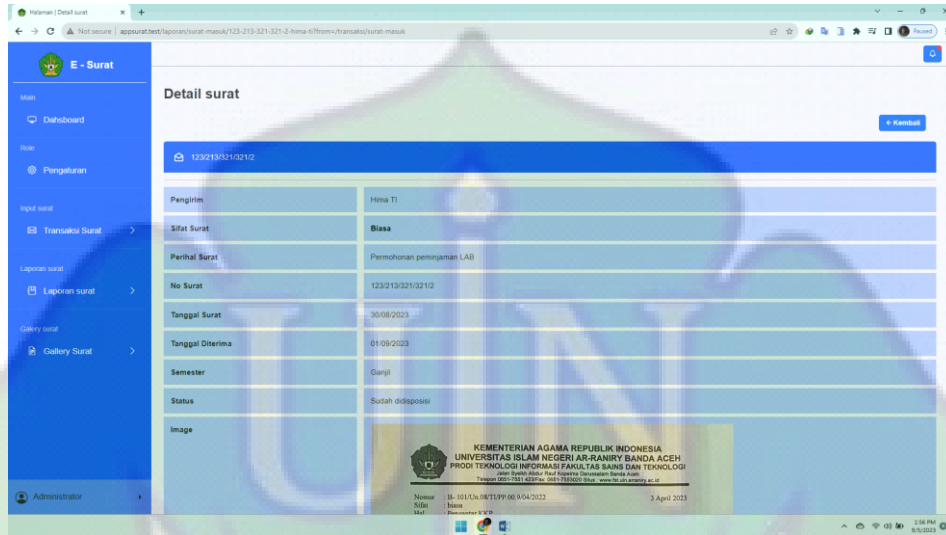
Halaman laporan surat disposisi ini akan menampilkan *list* surat masuk yang sudah disposisi oleh *user* atau kaprodi TI.



Gambar IV.27 : Tampilan halaman laporan surat disposisi

- Halaman laporan surat disposisi aksi detail

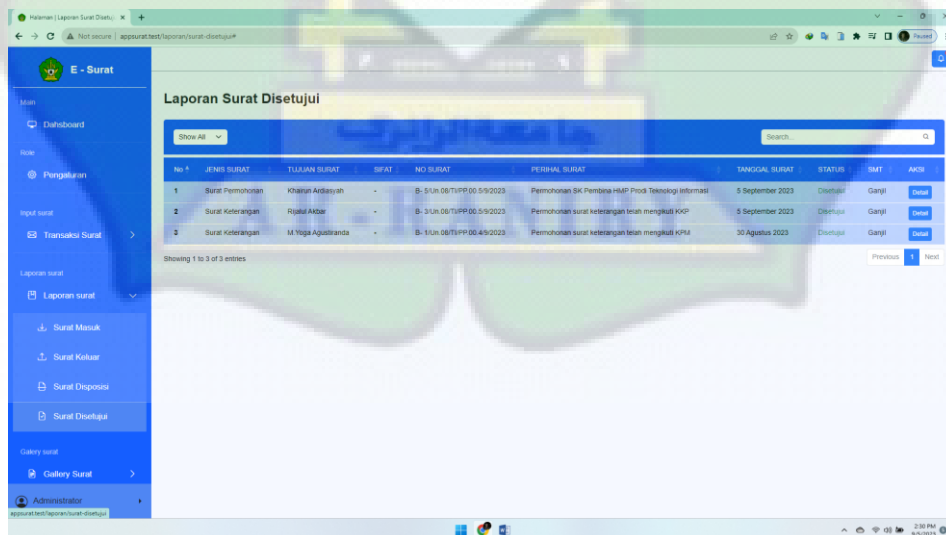
Halaman detail laporan surat disposisi, halaman ini akan muncul jika menekan tombol detail, halaman ini menampilkan informasi detail dari surat disposisi.



Gambar IV.28 : Halaman laporan surat disposisi aksi detail

- i. Tampilan halaman laporan surat disetujui

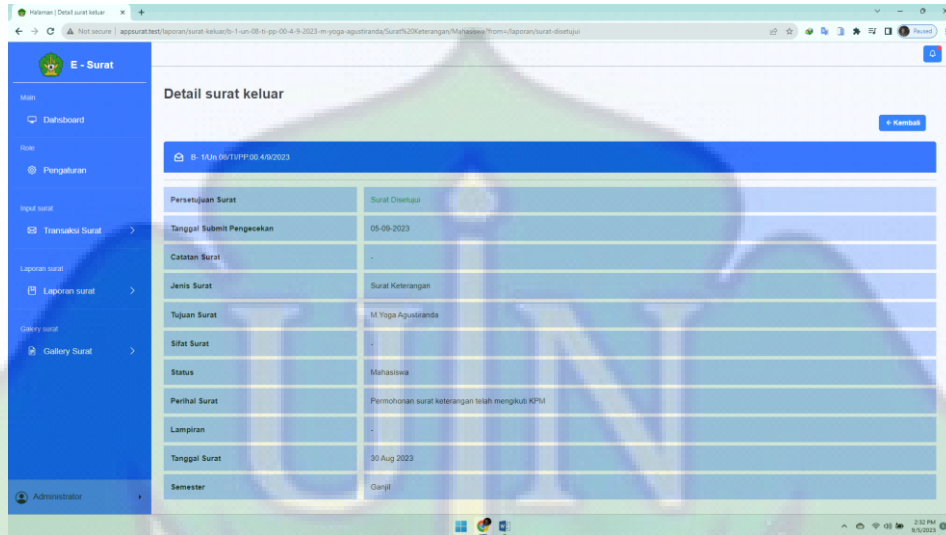
Halaman laporan surat disetujui ini akan menampilkan list surat keluar yang sudah disetujui oleh *user* atau kaprodi TI.



Gambar IV.29 : Halaman laporan surat disetujui

- Halaman laporan surat disetujui aksi detail

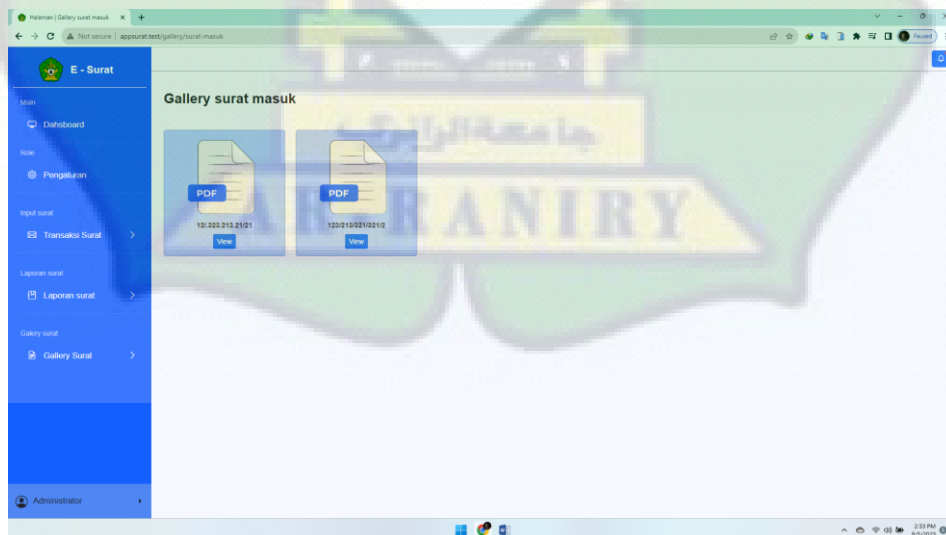
Halaman detail laporan surat disetujui, halaman ini akan muncul jika menekan tombol detail, halaman ini menampilkan informasi detail dari surat disetujui.



Gambar IV.30 : Halaman laporan surat disetujui aksi detail

- j. Tampilan halaman gallery surat masuk

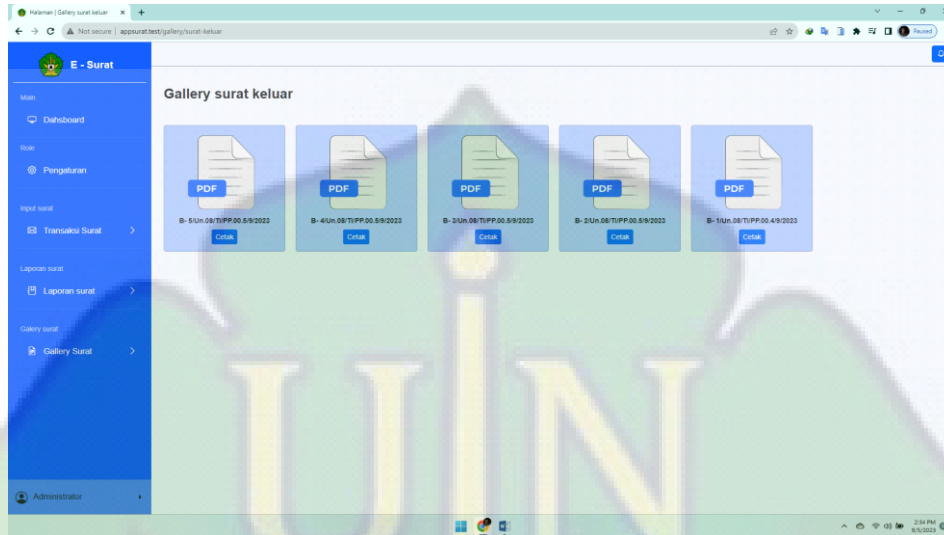
Halaman galeri *file* surat masuk, halaman ini menampilkan *file* dari surat masuk yang telah disposisi.



Gambar IV.31 : Tampilan halaman gallery surat masuk

k. Tampilan halaman gallery surat keluar

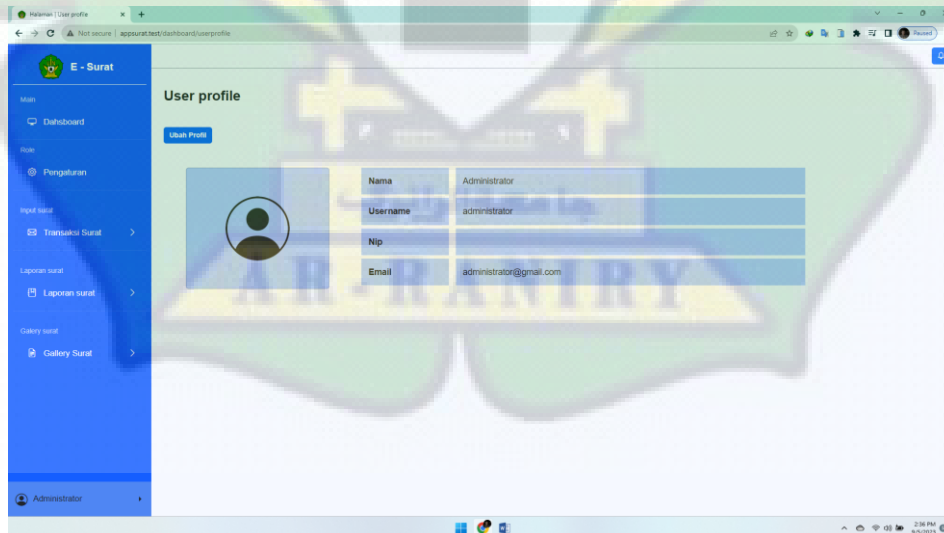
Halaman galeri *file* surat keluar, halaman ini menampilkan *file* dari surat masuk yang telah disetujui.



Gambar IV.32 : Tampilan halaman gallery surat keluar

l. Tampilan halaman user profile

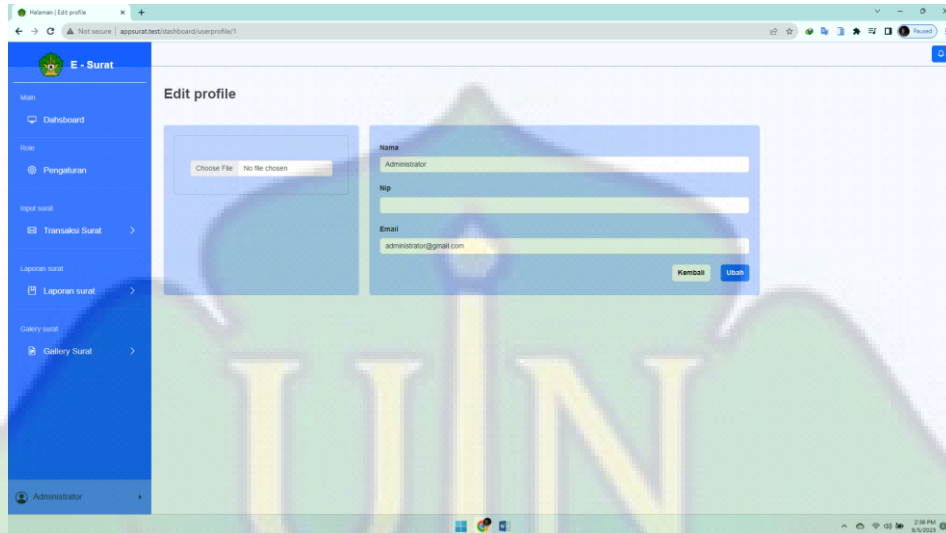
Halaman profil, halaman ini menampilkan informasi pengguna, halaman ini bisa diakses oleh admin dan *user*.



Gambar IV.33 : Tampilan halaman *user profile*

- Halaman user aksi edit *profile*

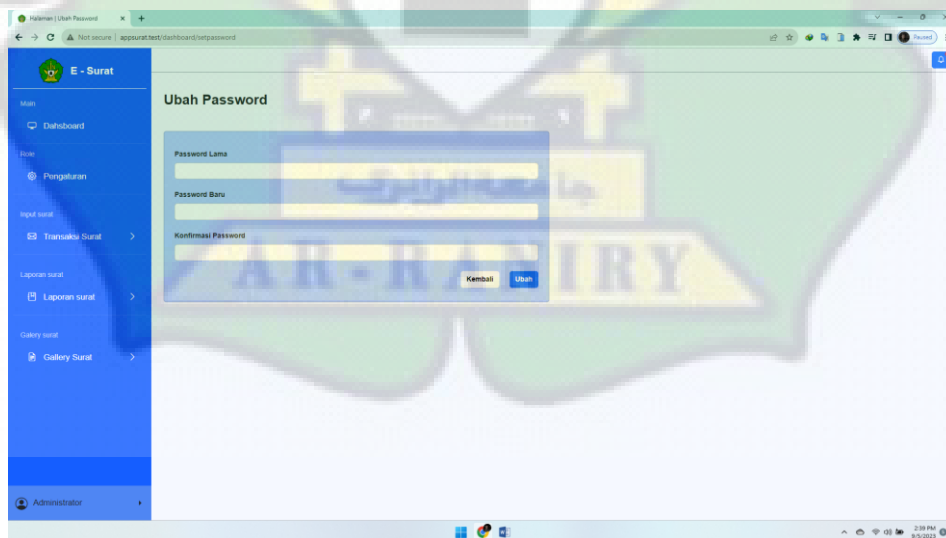
Halaman ubah profil, halaman ini untuk melakukan perubahan dari profil pengguna, halaman ini bisa diakses oleh admin dan *user*.



Gambar IV.34 : Halaman user edit *profile*

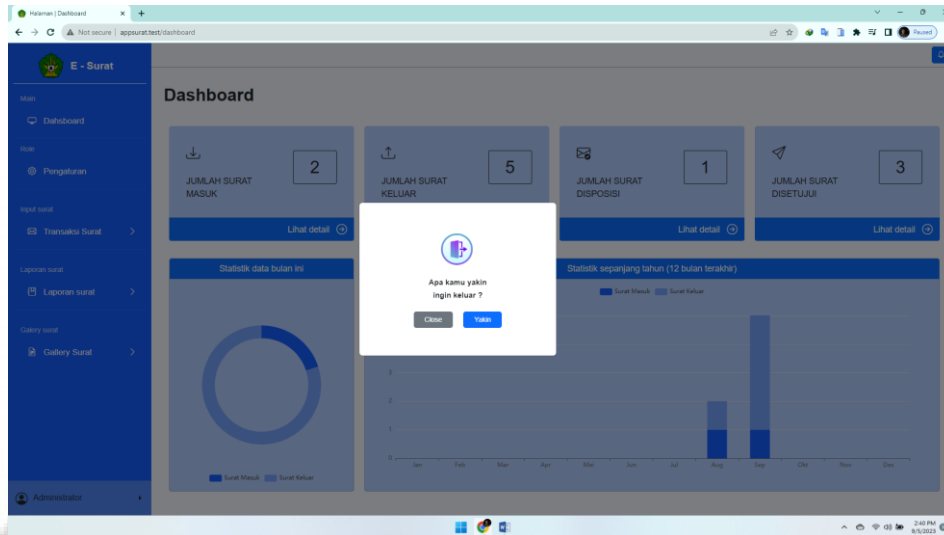
- Halaman user aksi ubah *password*

Halaman ubah *password*, halaman ini untuk melakukan perubahan dari *password* pengguna, halaman ini bisa diakses oleh admin dan *user*.



Gambar IV.35 : Halaman *user* aksi ubah *password*

m. Tampilan halaman *logout*



Gambar IV.36 : Tampilan halaman *logout*

IV.3 Pengujian sistem

Pengujian sistem merupakan proses menjalankan aplikasi yang telah dibuat untuk menentukan apakah aplikasi tersebut cocok dengan spesifikasi aplikasi yang di rencanakan dan berjalan di lingkungan yang diinginkan. Pengujian aplikasi sering diasosiasikan dengan pencarian bug, ketidak sempurnaan aplikasi, kesalahan pada aplikasi yang menyebabkan kegagalan pada eksekusi system perangkat lunak. Pengujian dilakukan dengan cara menguji setiap proses yang ada di dalam aplikasi.

Adapun pengujian aplikasi yang digunakan adalah Black box. Pengujian Black box yaitu menguji aplikasi dari segi spesifikasi fungsional tanpa menguji desain dan kode program (Parlika et al., 2020). Pengujian dimaksudkan untuk mengetahui apakah fungsi - fungsi, masukan dan keluaran dari aplikasi sesuai dengan spesifikasi yang dibutuhkan.

IV.3.1 Hasil pengujian sistem

Tabel IV.1 : Pengujian halaman login

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Mengosongkan email dan <i>password</i> lalu klik tombol login	Muncul <i>tooltips required input field</i>	Muncul <i>tooltips required input field</i>	Valid
Mengosongkan salah satu <i>field</i> , baik email ataupun <i>password</i>	Muncul <i>tooltips required input field</i>	Muncul <i>tooltips required input field</i>	Valid
Memasukkan email atau <i>password</i> yang salah	Muncul <i>alert login failed!</i>	Muncul <i>alert login failed!</i>	Valid
Memasukkan email dan <i>password</i> yang benar	Login berhasil dan <i>redirect</i> ke halaman dashboard	Login berhasil dan <i>redirect</i> ke halaman dashboard	Valid

Tabel IV.2 : Pengujian halaman dashboard

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan notifikasi untuk jumlah surat yang belum didisposisikan, belum dicek dan tidak disetujui	Muncul notifikasi jumlah surat belum didisposisikan, belum cek dan tidak disetujui	Muncul notifikasi jumlah surat belum didisposisikan, belum cek dan tidak disetujui	Valid
Tampilkan <i>card</i> untuk jumlah surat masuk, keluar, disposisi dan disetujui	Muncul jumlah surat berdasarkan <i>card</i> surat masuk, keluar, disposisi dan disetujui.	Muncul jumlah surat berdasarkan <i>card</i> surat masuk, keluar, disposisi dan disetujui.	Valid
Tampilkan grafik <i>circle</i> untuk data jumlah surat yang masuk dan keluar bulan ini	Muncul grafik <i>circle</i> perbandingan jumlah surat masuk dan keluar bulan ini	Muncul jumlah perbandingan jumlah surat masuk dan keluar bulan ini	Valid
Tampilkan grafik <i>bar</i> untuk data jumlah surat yang masuk dan keluar tahun ini	Muncul grafik <i>bar</i> jumlah surat masuk dan keluar tahun ini	Muncul grafik <i>bar</i> jumlah surat masuk dan keluar tahun ini	Valid

Tabel IV.3 : Pengujian halaman pengaturan role

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan formulir tambah <i>user</i> pada saat tombol tambah di klik	Muncul halaman formulir untuk tambah <i>user</i>	Muncul halaman formulir untuk tambah <i>user</i>	Valid
<i>Redirect</i> ke halaman pengaturan pada saat tombol kembali di klik dalam formulir tambah <i>user</i>	<i>Redirect</i> ke halaman pengaturan role	<i>Redirect</i> ke halaman pengaturan role	Valid
Mengosongkan salah satu field untuk inputan <i>user</i>	Muncul <i>tooltips required input field</i>	Muncul <i>tooltips required input field</i>	Valid
Memilih gambar <i>user</i> yang lebih dari 2MB	Muncul pesan di bawah input <i>field</i> , batas gambar 2048KB.	Muncul pesan di bawah <i>input field</i> , batas gambar 2048KB.	Valid
Memilih <i>level user</i> sebagai is <i>user</i> , dan jika <i>user</i> sudah ada maka muncul <i>alert user</i> sudah ada	Muncul <i>alert user</i> dengan level is user sudah ada	Muncul <i>alert user</i> dengan level is user sudah ada	Valid
Menginputkan semua <i>field</i> sesuai dengan sistem	Muncul <i>alert user has been added!</i>	Muncul <i>alert user has been added!</i>	Valid
Tampilkan formulir edit <i>user</i> pada saat tombol ubah di klik	Muncul halaman formulir untuk ubah <i>user</i>	Muncul halaman formulir untuk ubah <i>user</i>	Valid
Mengedit bagian <i>field</i> yang diperlukan sesuai dengan sistem dan pada saat di klik	Muncul <i>alert user has been update!</i>	Muncul <i>alert user has been update!</i>	Valid
Tampilkan <i>modal popup</i> serta nama <i>user</i> pada saat tombol hapus di klik	Muncul <i>modal popup</i> hapus	Muncul <i>modal popup</i> hapus	Valid
Hapus <i>user</i> pada saat tombol hapus di klik	<i>User</i> terhapus	<i>User</i> terhapus	Valid

Tabel IV.4 : Pengujian halaman transaksi surat masuk

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan data surat masuk sesuai inputan dalam field search secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Valid
Tampilkan formulir tambah surat masuk pada saat tombol tambah di klik	Muncul halaman formulir untuk tambah surat masuk	Muncul halaman formulir untuk tambah surat masuk	Valid
<i>Redirect</i> ke halaman surat masuk pada saat tombol kembali di klik dalam formulir tambah surat masuk	<i>Redirect</i> ke halaman surat masuk	<i>Redirect</i> ke halaman surat masuk	Valid
Mengosongkan salah satu field untuk inputan surat masuk	Muncul <i>tooltips required input field</i>	Muncul <i>tooltips required input field</i>	Valid
Memilih gambar surat masuk yang lebih dari 1MB	Muncul pesan di bawah input field, batas gambar 1024KB.	Muncul pesan di bawah input field, batas gambar 1028KB.	Valid
Menginputkan semua field sesuai dengan sistem	Muncul <i>alert</i> surat masuk <i>has been added!</i>	Muncul <i>alert</i> surat masuk <i>has been added!</i>	Valid
Tampilkan formulir edit surat masuk pada saat tombol ubah di klik	Muncul halaman formulir untuk ubah surat masuk	Muncul halaman formulir untuk ubah surat masuk	Valid
Mengedit bagian yang diperlukan sesuai dengan sistem dan pada saat di klik	Muncul <i>alert</i> surat masuk <i>has been update!</i>	Muncul <i>alert</i> surat masuk <i>has been update!</i>	Valid
Tampilkan formulir untuk melihat detail surat masuk pada saat tombol detail di klik	Muncul halaman formulir detail untuk surat masuk	Muncul halaman formulir detail untuk surat masuk	Valid
Tampilkan <i>modal popup</i> serta nomor suratnya pada saat tombol hapus di klik	Muncul <i>modal popup</i> hapus	Muncul <i>modal popup</i> hapus	Valid
Hapus surat masuk pada saat tombol hapus di klik	Surat masuk terhapus	Surat masuk terhapus	Valid

Tabel IV.5 : Pengujian halaman transaksi surat keluar

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan data surat keluar sesuai inputan dalam field search secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Valid
Tampilkan formulir tambah surat keluar pada saat tombol tambah di klik	Muncul halaman formulir untuk tambah surat keluar	Muncul halaman formulir untuk tambah surat keluar	Valid
<i>Redirect</i> ke halaman surat keluar pada saat tombol kembali di klik dalam formulir tambah surat masuk	<i>Redirect</i> ke halaman surat keluar	<i>Redirect</i> ke halaman surat keluar	Valid
Bagian nomor surat terisi secara otomatis (<i>auto number</i>)	Nomor surat berlanjut sesuai dengan nomor terakhir di database	Nomor surat berlanjut sesuai dengan nomor terakhir di database	Valid
Jika nomor surat sebelumnya sudah tersedia di database dan pada saat tombol tambah diklik	Muncul <i>alert</i> nomor surat sudah tersedia di database	Muncul <i>alert</i> nomor surat sudah tersedia di database	Valid
Mengosongkan salah satu field untuk inputan surat keluar	Muncul <i>tooltips required input field</i>	Muncul <i>tooltips required input field</i>	Valid
Menampilkan field inputan yang bersangkutan sesuai dengan status yang di pilih	Menampilkan field yang sesuai dengan status yang di pilih	Menampilkan field yang sesuai dengan status yang di pilih	Valid
Menginputkan semua field sesuai dengan sistem	Muncul <i>alert</i> surat keluar <i>has been added!</i>	Muncul <i>alert</i> surat keluar <i>has been added!</i>	Valid
Tampilkan formulir edit surat keluar pada saat tombol ubah di klik	Muncul halaman formulir untuk ubah surat keluar	Muncul halaman formulir untuk ubah surat keluar	Valid
Mengedit bagian yang diperlukan sesuai dengan sistem dan pada saat di klik	Muncul <i>alert</i> surat keluar <i>has been update!</i>	Muncul <i>alert</i> surat keluar <i>has been update!</i>	Valid
Tampilkan formulir untuk melihat detail surat keluar pada saat tombol detail di klik	Muncul halaman formulir detail untuk surat keluar	Muncul halaman formulir detail untuk surat keluar	Valid

Buka tab terbaru pada saat tombol view dari surat keluar di klik	Tab terbaru terbuka dan meuncul surat untuk di print	Tab terbaru terbuka dan meuncul surat untuk di print	Valid
Tampilkan <i>QRcode</i> di pdf untuk di print pada saat status surat telah di setuju	<i>QRcode</i> muncul pada bagian tanda tangan	<i>QRcode</i> muncul pada bagian tanda tangan	Valid
Tampilkan <i>modal popup</i> serta nomor suratnya pada saat tombol hapus di klik	Muncul <i>modal popup</i> hapus	Muncul <i>modal popup</i> hapus	Valid
Hapus surat keluar pada saat tombol hapus di klik	Surat keluar terhapus	Surat keluar terhapus	Valid

Tabel IV.6 : Pengujian halaman laporan surat masuk

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan data surat masuk sesuai inputan dalam field search secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Valid
Tampilkan modal <i>popup</i> pada saat tombol filter di klik	Muncul <i>popup</i> untuk mengfilter surat masuk	Muncul <i>popup</i> untuk mengfilter surat masuk	Valid
Jalankan fungsi filter pada saat salah satu atau semua field di isi	Menampilkan data yang sesuai dengan difilterkan	Menampilkan data yang sesuai dengan difilterkan	Valid
Tombol print di klik dan beberapa parameter seperti tanggal mulai, akhir dan semester tidak terpenuhi	Tampilkan <i>alert</i> dengan pesan beberapa parameter tidak di isi	Tampilkan <i>alert</i> dengan pesan beberapa parameter tidak di isi	Valid
Buka tab terbaru pada saat tombol print dari surat masuk di klik jika syarat parameter pencarian memenuhi	Tab terbaru terbuka dan meuncul jumlah surat yang sesuai dengan filter	Tab terbaru terbuka dan meuncul jumlah surat yang sesuai dengan filter	Valid
Tampilkan formulir untuk inputan surat masuk pada saat tombol disposisi di klik	Muncul halaman formulir disposisi untuk surat masuk	Muncul halaman formulir disposisi untuk surat masuk	Valid

Redirect ke halaman surat masuk pada saat tombol kembali di klik dalam formulir disposisi surat masuk	Redirect ke halaman surat masuk	Redirect ke halaman surat masuk	Valid
Pada saat tombol disposisi diklik di formulir disposisi surat	Muncul popup surat <i>disposition mail has been added!</i>	Muncul popup surat <i>disposition mail has been added!</i>	Valid
Tampilkan formulir untuk melihat detail surat masuk pada saat tombol detail di klik	Muncul halaman formulir detail untuk surat masuk	Muncul halaman formulir detail untuk surat masuk	Valid

Tabel IV.7 : Pengujian halaman laporan surat keluar

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan data surat keluar sesuai inputan dalam field search secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Valid
Tampilkan modal <i>popup</i> pada saat tombol filter di klik	Muncul <i>popup</i> untuk mengfilter surat keluar	Muncul <i>popup</i> untuk mengfilter surat keluar	Valid
Jalankan fungsi filter pada saat salah satu atau semua field di isi	Menampilkan data yang sesuai dengan difilterkan	Menampilkan data yang sesuai dengan difilterkan	Valid
Tombol print di klik dan beberapa parameter seperti tanggal mulai, akhir dan semester tidak terpenuhi	Tampilkan <i>alert</i> dengan pesan beberapa parameter tidak di isi	Tampilkan <i>alert</i> dengan pesan beberapa parameter tidak di isi	Valid
Buka tab terbaru pada saat tombol print dari surat keluar di klik jika syarat parameter pencarian memenuhi	Tab terbaru terbuka dan meuncul jumlah surat yang sesuai dengan filter	Tab terbaru terbuka dan meuncul jumlah surat yang sesuai dengan filter	Valid
Tampilkan formulir untuk inputan surat keluar pada saat tombol setuju surat di klik	Muncul halaman formulir setuju untuk surat keluar	Muncul halaman formulir setuju untuk surat keluar	Valid

Redirect ke halaman surat keluar pada saat tombol kembali di klik dalam formulir setuju surat keluar	Redirect ke halaman surat keluar	Redirect ke halaman surat keluar	Valid
Buka tab terbaru pada saat tombol view dari surat keluar di klik	Tab terbaru terbuka dan muncul surat untuk di print	Tab terbaru terbuka dan muncul surat untuk di print	Valid
Pada saat tombol setuju surat diklik di formulir setuju surat	Muncul popup <i>new mail has been submit</i>	Muncul popup <i>new mail has been submit</i>	Valid

Tabel IV.8 : Pengujian halaman laporan surat disposisi

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan data surat disposisi sesuai inputan dalam field <i>search</i> secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Muncul data yang di cari sesuai dengan column bagian surat secara <i>live searching</i>	Valid
Tampilkan formulir untuk melihat detail surat masuk pada saat tombol detail di klik	Muncul halaman formulir detail untuk surat masuk	Muncul halaman formulir detail untuk surat masuk	Valid
Redirect ke halaman surat disposisi pada saat tombol kembali di klik dalam formulir detail surat disposisi	Redirect ke halaman laporan surat disposisi	Redirect ke halaman laporan surat disposisi	Valid

Tabel IV.9 : Pengujian halaman laporan surat disetujui

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan data surat disetujui sesuai inputan dalam field <i>search</i> secara <i>live searching</i>	Muncul data yang di cari sesuai dengan kolom bagian surat secara <i>live searching</i>	Muncul data yang di cari sesuai dengan kolom bagian surat secara <i>live searching</i>	Valid
Tampilkan formulir untuk melihat detail surat keluar pada saat tombol detail di klik	Muncul halaman formulir detail untuk surat masuk	Muncul halaman formulir detail untuk surat masuk	Valid

<i>Redirect</i> ke halaman surat disetujui pada saat tombol kembali di klik dalam formulir detail surat disposisi	<i>Redirect</i> ke halaman laporan surat disetujui	<i>Redirect</i> ke halaman laporan surat disetujui	Valid
---	--	--	-------

Tabel IV.10 : Pengujian halaman gallery surat masuk

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan <i>card</i> yang berisikan nomor surat untuk gambar surat masuk	Menampilkan <i>card</i> dan tombol untuk <i>view</i> gambar surat masuk	Menampilkan <i>card</i> dan tombol untuk <i>view</i> gambar surat masuk	Valid
Buka tab terbaru pada saat tombol <i>view</i> di klik	Tab terbaru terbuka dan menampilkan gambar dari surat	Tab terbaru terbuka dan menampilkan gambar dari surat	Valid

Tabel IV.11 : Pengujian halaman gallery surat keluar

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan <i>card</i> yang berisikan nomor surat untuk cetak surat keluar	Menampilkan <i>card</i> dan tombol untuk cetak surat keluar	Menampilkan <i>card</i> dan tombol untuk cetak surat keluar	Valid
Buka tab terbaru pada saat tombol cetak di klik	Tab terbaru terbuka dan menampilkan <i>print</i> dari surat keluar	Tab terbaru terbuka dan menampilkan <i>print</i> dari surat keluar	Valid

Tabel IV.12 : Pengujian halaman *user profile*

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan formulir edit <i>user profile</i> pada saat tombol ubah di klik	Muncul halaman formulir untuk ubah <i>user profile</i>	Muncul halaman formulir untuk ubah <i>user profile</i>	Valid
<i>Redirect</i> ke halaman <i>user profile</i> pada saat tombol kembali di klik dalam formulir ubah <i>user profile</i> dan ganti <i>password</i>	<i>Redirect</i> ke halaman <i>user profile</i>	<i>Redirect</i> ke halaman <i>user profile</i>	Valid

Mengedit bagian yang diperlukan sesuai dengan sistem dan pada saat di klik	Muncul <i>alert user profile has been update!</i>	Muncul <i>alert user profile has been update!</i>	Valid
Tampilkan formulir ubah <i>password user profile</i> pada saat tombol ubah <i>password</i> di klik	Muncul halaman formulir untuk ubah <i>password</i> user	Muncul halaman formulir untuk ubah <i>password</i> user	Valid
<i>Password</i> lama salah pada saat <i>password</i> baru di di ganti	Muncul <i>popup alert password</i> lama anda salah	Muncul <i>popup alert password</i> lama anda salah	Valid
Konfirmasi <i>password</i> berbeda dari isi field <i>password</i> baru	Muncul pesan kesalahan dibawah field konfirmasi <i>password</i>	Muncul pesan kesalahan dibawah field konfirmasi <i>password</i>	Valid
Inputan sesuai dengan permintaan sistem dan pada saat tombol ubah diklik	Muncul <i>alert password</i> telah di ganti	Muncul <i>alert password</i> telah di ganti	Valid

Tabel IV.13 : Pengujian halaman *logout*

Skenario Pengujian	Hasil yang Diharapkan	Hasil pengujian	Kesimpulan
Tampilkan modal <i>logout</i> pada saat tombol <i>logout</i> di klik	Muncul modal <i>popup logout</i>	Muncul modal <i>popup logout</i>	Valid
<i>Redirect</i> sekaligus keluar dari aplikasi pada saat tombol <i>logout</i> di klik	<i>Redirect</i> ke halaman login	<i>Redirect</i> ke halaman login	Valid

BAB V

KESIMPULAN DAN SARAN

V.1 Kesimpulan

Berdasarkan hasil dari pengembangan website pengarsipan surat masuk dan surat keluar dengan menggunakan metode *prototype* berbasis *website*, maka didapatkan kesimpulan sebagai berikut :

1. Penelitian ini menghasilkan aplikasi berbasis website yang pengembangannya menggunakan pemograman *Html*, *Css*, *Javascript* dan *framework laravel* versi 10, dan juga menggunakan metode pengembangan aplikasi *prototype*.
2. Aplikasi yang di hasilkan memberikan beberapa kemudahan seperti penomoran surat keluar otomatis, tanda tangan surat dengan *QRcode*, dan dokumentasi yang disediakan untuk surat masuk dan surat keluar setiap tahun, memungkinkan pengguna untuk lebih mudah memantau dan mencari surat-surat yang diperlukan dalam arsip.
3. Hasil pengujian *blackbox* keseluruhan fungsi dari implementasi metode *prototype* pada aplikasi Sistem Informasi Arsip Surat menggunakan Framework Laravel 10 dan beberapa bahasa *web* lain adalah valid dan berjalan dengan baik.

V.2 Saran

Dari hasil Implementasi Metode *prototype* pada Pembuatan Aplikasi Sistem Informasi Arsip Surat Berbasis *Website* menggunakan *Framework laravel* 10 dan beberapa bahasa web lain dalam penelitian ini, maka saran yang dapat di berikan oleh peneliti untuk selanjutnya adalah :

1. Pengujian lanjutan

Melakukan pengujian lebih lanjut terhadap sistem yang telah dibangun. Hal ini mencakup pengujian fungsional, pengujian performa, dan pengujian keamanan. Pengujian yang lebih mendalam akan membantu menemukan potensi bug atau masalah yang belum terdeteksi.

2. Pengembangan fitur tambahan

Berdasarkan umpan balik dari pengguna, pertimbangkan untuk menambahkan fitur-fitur tambahan yang dapat meningkatkan fungsionalitas sistem. Pastikan fitur tambahan ini tetap relevan dengan tujuan utama sistem.

3. Optimasi kinerja

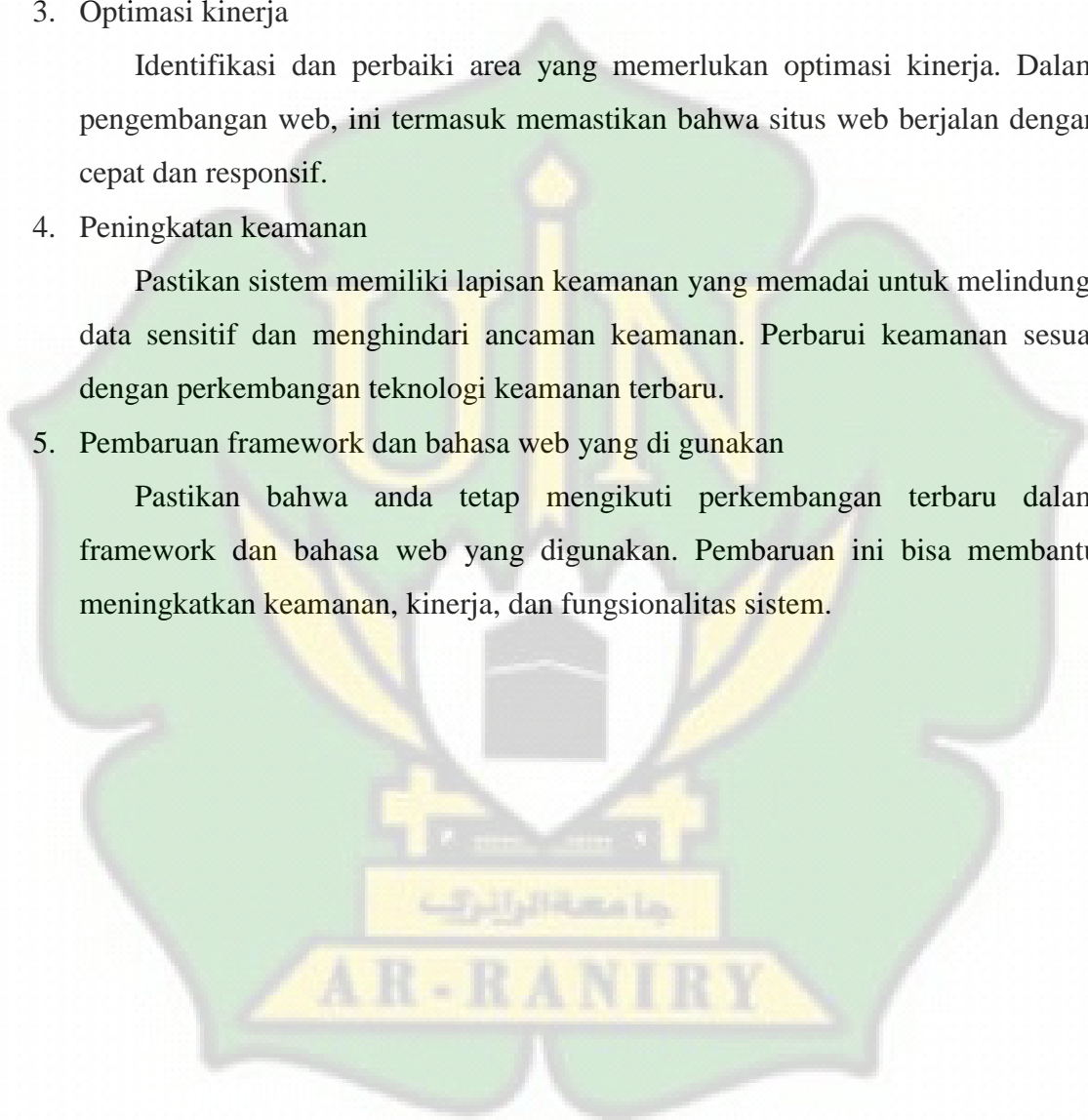
Identifikasi dan perbaiki area yang memerlukan optimasi kinerja. Dalam pengembangan web, ini termasuk memastikan bahwa situs web berjalan dengan cepat dan responsif.

4. Peningkatan keamanan

Pastikan sistem memiliki lapisan keamanan yang memadai untuk melindungi data sensitif dan menghindari ancaman keamanan. Perbarui keamanan sesuai dengan perkembangan teknologi keamanan terbaru.

5. Pembaruan framework dan bahasa web yang di gunakan

Pastikan bahwa anda tetap mengikuti perkembangan terbaru dalam framework dan bahasa web yang digunakan. Pembaruan ini bisa membantu meningkatkan keamanan, kinerja, dan fungsionalitas sistem.



DAFTAR PUSTAKA

- Ardiyansah, D., Pahlevi, O., & Santoso, T. (2021). Implementasi Metode Prototyping Pada Sistem Informasi. *Jurnal Teknik Dan Sains*, 2(2), 17–22. <http://jurnal.uts.ac.id/index.php/hexagon/article/view/1083%0Ahttps://jurnal.uts.ac.id/index.php/hexagon/article/download/1083/701>
- Ariawan, M. D., Triayudi, A., & Sholihati, I. D. (2020). Perancangan User Interface Design dan User Experience Mobile Responsive Pada Website Perusahaan. *Jurnal Media Informatika Budidarma*, 4(1), 161. <https://doi.org/10.30865/mib.v4i1.1896>
- Codingstudio.id 2023. “Flowchart Adalah: Tujuan, Simbol, dan Manfaatnya”. [Online]. Available: <https://codingstudio.id/blog/flowchart-adalah/>. [Accessed: 03-Sep-2023].
- Datatables.net. 2023. "DataTables". [Online]. Available: <https://datatables.net/>. [Accessed: 15-Agu-2023].
- Dicoding.com 2023 “Memahami ERD, Model data, dan Komponenya”. [Online]. Available: <https://www.dicoding.com/blog/memahami-erd/>. [Accessed: 14-Okt-2023].
- Getbootstrap.com. 2023. "Getting Started". [Online]. Available: <https://getbootstrap.com/docs/5.2/getting-started/introduction/>. [Accessed: 20-Jul-2023].
- Jantce TJ Sitinjak, D. D., Maman, ., & Suwita, J. (2020). Analisa Dan Perancangan Sistem Informasi Administrasi Kursus Bahasa Inggris Pada Intensive English Course Di Ciledug Tangerang. *Insan Pembangunan Sistem Informasi Dan Komputer (IPSIKOM)*, 8(1). <https://doi.org/10.58217/ipsikom.v8i1.164>
- Jquery.com. 2023. [Online]. Available: <https://getbootstrap.com/docs/5.2/getting-started/introduction/>. [Accessed: 20-Jul-2023].
- Laravel.com. 2023. "Getting Started". [Online]. Available: <https://laravel.com/docs/10.x>. [Accessed: 03-Agu-2023].
- Lubis, R. A., & Samsudin. (2022). *Perancangan Aplikasi Arsip Surat Berbasis Web Di Dinas*. 17, 28–35.
- Nuryani, S. (2021). Pengembangan Aplikasi Mobile Booking Online Perawatan Gigi Dengan Metode Prototype Studi Kasus di Klinik Gigi Budiono, drg. Kota Bandung. *Intelektiva : Jurnal Ekonomi, Sosial & Humaniora*, 2(6), 18–28.
- Parlika, R., Nisaa', T. A., Ningrum, S. M., & Haque, B. A. (2020). Studi Literatur Kekurangan Dan Kelebihan Pengujian Black Box. *Teknomatika*, 10(02), 131–140.

- Putra, A. S. (2021). Sistem Manajemen Pelayanan Pelanggan Menggunakan PHP Dan MySQL (Studi Kasus pada Toko Surya). *Tekinfo: Jurnal Bidang Teknik Industri Dan Teknik Informatika*, 22(1), 100–116. <https://doi.org/10.37817/tekinfo.v22i1.1190>
- Putri, N., Agung Prabowo, N., & Widyanto, R. A. (2020). Implementasi Metode Prototyping pada Perancangan Aplikasi Electronic Ticket (E-Ticket) berbasis Android. *Jurnal Komtika (Komputasi Dan Informatika)*, 3(2), 62–68. <https://doi.org/10.31603/komtika.v3i2.3474>
- Rhomdani, R. W. (2022). Algoritma Modulo Berpangkat Menggunakan Teorema Binomial Newton Dan Phi Euler Dengan Javascript. *Teorema: Teori Dan Riset Matematika*, 7(2), 403. <https://doi.org/10.25157/teorema.v7i2.7707>
- Saifudin, S., & Setiaji, A. Y. (2019). Sistem Informasi Arsip Surat (Sinau) Berbasis Web Pada Kantor Desa Karangsalam Kecamatan Baturraden. *EVOLUSI : Jurnal Sains Dan Manajemen*, 7(2), 15–21. <https://doi.org/10.31294/evolusi.v7i2.6751>
- Sari, A. M., Yani, D., & Suryani, D. (2021). Implementasi Aplikasi Mobile Peta NKRI (Negara Kesatuan Republik Indonesia) Berbasis Android Menggunakan Metode Prototype. *Journal of Information System Research (JOSH)*, 2(4), 288–292. <https://doi.org/10.47065/josh.v2i4.793>
- Sari, I. P., Azzahrah, A., Qathrunada, I. F., Lubis, N., & Anggraini, T. (2022). Perancangan Sistem Absensi Pegawai Kantoran Secara Online pada Website Berbasis HTML dan CSS. *Blend Sains Jurnal Teknik*, 1(1), 8–15. <https://doi.org/10.56211/blendsains.v1i1.66>
- Suryadi, A. (2019). Rancang Bangun Sistem Pengelolaan Arsip Surat Berbasis Web Menggunakan Metode Waterfall (Studi Kasus : Kantor Desa Karangrau Banyumas). *Jurnal Khatulistiwa Informatika*, 7(1), 13–21. <https://doi.org/10.31294/jki.v7i1.36>
- Yoko, P., Adwiya, R., & Nugraha, W. (2019). Penerapan Metode Prototype dalam Perancangan Aplikasi SIPINJAM Berbasis Website pada Credit Union Canaga Antutn. *Jurnal Ilmiah Merpati (Menara Penelitian Akademika Teknologi Informasi)*, 7(3), 212. <https://doi.org/10.24843/jim.2019.v07.i03.p05>
- Wikipedia.org 2023 “Arsip”. [Online]. Available: <https://id.wikipedia.org/wiki/Sistem>. [Accessed: 03-Sep-2023].
- Wikipedia.org 2023 “Sistem”. [Online]. Available: <https://id.wikipedia.org/wiki/Arsip>. [Accessed: 03-Sep-2023].

LAMPIRAN

Lampiran codingan *Model View Controller* (MVC)

Lampiran 1 1 Codingan *file Routes* (web)

```
1 Route::get('/', function () {
2     return redirect('/dashboard');
3 });
4
5 Route::get('/login', [LoginController::class, 'index']->name('login')->middleware('guest');
6 Route::post('/login', [LoginController::class, 'authenticate']);
7 Route::post('/logout', [LoginController::class, 'logout']);
8
9 Route::get('/dashboard', [DashboardController::class, 'index']->middleware('auth');
10 Route::get('/get-data-by-month-and-year/{bulan}/{tahun}', [DashboardController::class, 'getDataByMonthAndYear']);
11
12 // Role Pengaturan
13 Route::resource('/dashboard/pengaturan', UserController::class)->except('show')->middleware('is_admin');
14 Route::resource('/dashboard/pengaturan/sifat surat', SifatSuratController::class)->except(['show', 'edit', 'update'])->middleware('is_admin');
15 Route::resource('/dashboard/pengaturan/klasifikasi', KlasifikasiController::class)->middleware('is_admin');
16
17 // Transaksi Surat
18 Route::resource('/transaksi/surat-masuk', TransaksiSuratMasukController::class)->middleware('is_admin');
19 Route::resource('/transaksi/surat-keluar', TransaksiSuratKeluarController::class)->middleware('is_admin');
20
21 // Laporan Surat
22 Route::resource('/laporan/surat-masuk', LaporanSuratMasukController::class)->only(['index', 'show', 'edit', 'update'])->middleware('auth');
23 Route::resource('/laporan/surat-keluar', LaporanSuratKeluarController::class)->only(['index', 'edit', 'update', 'show'])->middleware('auth');
24 Route::resource('/laporan/surat-disposisi', LaporanSuratDisposisiController::class)->only('index')->middleware('auth');
25 Route::resource('/laporan/surat-disetujui', LaporanSuratDisetujuiController::class)->only('index')->middleware('auth');
26
27 // gallery Surat
28 Route::resource('/gallery/surat-masuk', GallerySuratMasukController::class)->only(['index', 'show'])->middleware('auth');
29 Route::resource('/gallery/surat-keluar', GallerySuratKeluarController::class)->only('index')->middleware('auth');
30
31 // User profile
32 Route::get('/dashboard/userprofile', [UserProfileController::class, 'index']->name('userprofile');
33 Route::get('/dashboard/userprofile/{id}', [UserProfileController::class, 'edit']->name('edit');
34 Route::put('/dashboard/userprofile/{id}', [UserProfileController::class, 'update']->name('updateuserprofile');
35
36 // User edit password
37 Route::get('/dashboard/setpassword', [UserProfileController::class, 'setpassword']->name('setpassword');
38 Route::put('/dashboard/setpassword/{id}', [UserProfileController::class, 'updatepassword']->name('updatepassword');
39
40 // Route untuk menangani aksi edit surat di menu transaksi surat
41 Route::get('/transaksi/surat-keluar/{slug}/{jenis}/{status}/edit', [TransaksiSuratKeluarController::class, 'edit']->name('edit-surat-keluar')->middleware('auth');
42
43 // Route untuk menangani aksi setuju surat di menu Laporan surat
44 Route::get('/laporan/surat-keluar/{jenis}/{slug}/edit', [LaporanSuratKeluarController::class, 'edit']->name('edit-laporan-surat-keluar')->middleware('auth');
45
46 // Route untuk menangani aksi detail surat keluar
47 Route::get('/laporan/surat-keluar/{slug}/{jenis}/{status}', [LaporanSuratKeluarController::class, 'show']->name('show-detail-laporan')->middleware('auth');
48
49 // Check Slug
50 Route::get('/transaksi/suratmasuk/checkslug', [TransaksiSuratMasukController::class, 'checkslug']);
51 Route::get('/transaksi/suratkeluar/checkslug', [TransaksiSuratKeluarController::class, 'checkslug']);
52
53 // view PDF
54 Route::get('/view-pdf/{slug}/{jenis}/{status}', [ViewPDFController::class, 'viewpdf']->name('viewpdf');
55 Route::get('/print-pdf/laporan-surat-masuk', [ViewPDFController::class, 'printSuratMasuk']->name('printSuratMasuk');
56 Route::get('/print-pdf/laporan-surat-keluar', [ViewPDFController::class, 'printSuratKeluar']->name('printSuratKeluar');
57
```

Lampiran 1 2 Codingan Model

1. Model surat masuk

```
1 class SuratMasuk extends Model
2 {
3     use HasFactory;
4     use Sluggable;
5
6     protected $guarded = ['id'];
7
8     public function sluggable(): array
9     {
10        return [
11            'slug' => [
12                'source' => 'nomor_surat'
13            ]
14        ];
15    }
16
17    // Ganti yang biasa dikirim parameter ID ke SLUG
18    public function getRouteKeyName(): string
19    {
20        return 'slug';
21    }
22
23    public function scopeFilter($query, array $filters)
24    {
25        $query->when($filters['pengirim'] ?? false, function ($query, $pengirim) {
26            return $query->where('pengirim', 'like', '% ' . $pengirim . '%');
27        });
28
29        $query->when($filters['semester'] ?? false, function ($query, $semester) {
30            return $query->where('semester', 'like', '% ' . $semester . '%');
31        });
32
33        $query->when($filters['start_date'] ?? false, function ($query, $start_date) use ($filters) {
34            $query->whereDate('tanggal_surat', '>=', $start_date);
35
36            // Harusnya bisa pakek metode null coalescing ??
37            // tetapi kasusnya kita mengecek, apakah end date terinput atau tidak ?
38            // Makanya menggunakan isset()
39            if (isset($filters['end_date'])) {
40                $query->whereDate('tanggal_surat', '<=', $filters['end_date']);
41            }
42        });
43    }
44 }
45
```

2. Model surat keluar

```
1 class SuratKeluar extends Model
2 {
3     use HasFactory;
4     use Sluggable;
5
6     protected $guarded = ['id'];
7
8     // Create Slug
9     public function sluggable(): array
10    {
11        return [
12            'slug' => [
13                'source' => 'nomor_surat'
14            ]
15        ];
16    }
17
18    // Ganti yang biasa dikirim parameter ID ke SLUG
19    public function getRouteKeyName(): string
20    {
21        return 'slug';
22    }
23
24    // Generate Nomor Surat
25    public static function generateNomorSurat()
26    {
27        $lastSurat = SuratKeluar::latest('id')->first();
28
29        if ($lastSurat) {
30            $lastNumber = explode('-', $lastSurat->nomor_surat)[1];
31            $nextNumber = intval($lastNumber) + 1;
32        } else {
33            $nextNumber = 1;
34        }
35
36        $now = Carbon::now();
37        $tanggal = $now->day;
38        $bulan = $now->month;
39        $tahun = $now->year;
40
41        return "B- $nextNumber/Un.08/II/PP.00.$tanggal/$bulan/$tahun";
42    }
43
44    // Search Scope Filters
45    public function scopeFilter($query, array $filters)
46    {
47        $query->when($filters['jenis_surat'] ?? false, function ($query, $jenis_surat) {
48            return $query->where('jenis_surat', 'like', '%'. $jenis_surat . '%');
49        });
50
51        $query->when($filters['semester'] ?? false, function ($query, $semester) {
52            return $query->where('semester', 'like', '%'. $semester . '%');
53        });
54
55        $query->when($filters['start_date'] ?? false, function ($query, $start_date) use ($filters) {
56            $query->whereDate('tanggal_surat', '>=', $start_date);
57            if (isset($filters['end_date'])) {
58                $query->whereDate('tanggal_surat', '<=', $filters['end_date']);
59            }
60        });
61    }
62 }
63 }
64 }
```

Lampiran 1.3 Codingsan View

1. View surat masuk

```
1 @extends('dashboard.layouts.main')
2
3 @section('content')
4
5 <div class="main-title mb-1">
6     <span>Transaksi Surat Masuk</span>
7 </div>
8
9 @if (session()-has('success'))
10 <div class="alert alert-warning alert-dismissible fade show col-lg-5 d-flex align-items-center role="alert">
11     <i class="bi bi-envelope-open me-3 fs-5"></i>
12     {{ session('success') }}
13     <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
14 </div>
15 @endif
16
17 <div class="content-header mt-4 rounded-top">
18     <div class="row gap-2">
19         <div class="col-1">
20             <select id="nomoruruts" class="form select border-0">
21                 <option selected value="100">Show All</option>
22                 <option value="10">10</option>
23                 <option value="25">25</option>
24                 <option value="50">50</option>
25                 <option value="100">100</option>
26             </select>
27         </div>
28         <div class="col-sm-3">
29             <div class="input-group">
30                 <span class="input-group-text border-0 bg-white id="search"><i class="bi bi-search"></i></span>
31                 <input type="text" class="form-control border-0 name="search" id="searchInput" placeholder="Search..." aria-label="Search" aria-describedby="search" autocomplete="off">
32             </div>
33         </div>
34         <div class="col">
35             <div class="d-flex justify-content-end">
36                 <a href="/transaksi/surat-masuk/create" role="button" class="btn btn-light fw-bold">Tambah <i class="bi bi-plus-lg"></i></a>
37             </div>
38         </div>
39     </div>
40 </div>
41
42 <div class="table-responsive">
43     <table class="table text-white border-light" id="dataTable" style="width: 100%;>
44         <thead class="header-table">
45             <tr>
46                 <td class="py-3 px-4">No</td>
47                 <td class="py-3 px-4">JENIS SURAT</td>
48                 <td class="py-3 px-4">PENGIRIM</td>
49                 <td class="py-3 px-4">SIFAT</td>
50                 <td class="py-3 px-4">TANGGAL SURAT</td>
51                 <td class="py-3 px-4">NO SURAT</td>
52                 <td class="py-3 px-4">PERIHAL SURAT</td>
53                 <td class="py-3 px-4">SMT</td>
54                 <td class="py-3 px-4">AKSI</td>
55             </tr>
56         </thead>
57
58         <tbody class="body-table">
59             <@foreach ($surats as $data)>
60                 <tr>
61                     <td class="py-2 px-4 fw-sembold">{{ $loop->iteration }}</td>
62                     <td class="py-2 px-4">{{ $data->klasifikasi->jenis_surat }}</td>
63                     <td class="py-2 px-4">{{ $data->pengirim }}</td>
64                     <td class="py-2 px-4">
65                         <span class="sifat-surat fw-normal">{{ isset($data->sifat_surat->sifat_surat) ? $data->sifat_surat->sifat_surat : '-' }}</span>
66                     </td>
67                     <td class="py-2 px-4">{{ \Carbon\Carbon::createFromFormat('Y-m-d', $data->tanggal_surat->locale('id')->isoformat('DD / MM / YYYY')) }}</td>
68                     <td class="py-2 px-4">{{ $data->nomor_surat }}</td>
69                     <td class="py-2 px-4">{{ $data->perihal_surat }}</td>
70                     <td class="py-2 px-4">{{ $data->semester }}</td>
71                     <td class="py-2 px-4">
72                         <div class="btn-group">
73                             <button type="button" class="btn btn-sm btn-primary dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
74                                 <i class="bi bi-list btn-menu"></i>
75                             </button>
76                             <ul class="dropdown-menu">
77                                 <li>
78                                     <a class="dropdown-item" href="/transaksi/surat-masuk/{{ $data->slug }}/edit">
79                                         <i class="btn btn-success bi bi-pencil"></i>
80                                     </a>
81                                 </li>
82                                 <li>
83                                     <button type="button" class="dropdown-item deletedModalSuratHasuk" data-slug="/transaksi/surat-masuk/{{ $data->slug }}" data-nomor="{{ $data->nomor_surat }}">
84                                         <i class="btn btn-danger bi bi-trash"></i>
85                                     </button>
86                                 </li>
87                                 <li>
88                                     <a class="dropdown-item" href="/laporan/surat-masuk/{{ $data->slug }}">{{ $data->slug }}</a>
89                                     <i class="btn btn-warning bi bi-card-list"></i>
90                                 </li>
91                             </ul>
92                         </div>
93                     </td>
94                 </tr>
95             </tbody>
96         </tbody>
97     </table>
98 </div>
99 @endsection
```

2. View surat keluar

```
1 @extends('dashboard.layouts.main')
2
3 @section('content')
4
5 <div class="main-little mb-1">
6     <span>Transaksi Surat Keluar</span>
7 </div>
8
9 @if (session()>has('success'))
10 <div class="alert alert-warning alert-dismissible fade show col-lg-5 d-flex align-items-center" role="alert">
11     <i class="bi bi-envelope-open me-1 fs-5"></i>
12     {{ session('success') }}
13     <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
14 </div>
15 @endif
16
17 <div class="content-header mt-4 rounded-top">
18     <div class="row gsp-2">
19         <div class="col-1">
20             <select id="showEntries" class="form-select border-0">
21                 <option selected value="100">Show All</option>
22                 <option value="10">10</option>
23                 <option value="25">25</option>
24                 <option value="50">50</option>
25                 <option value="100">100</option>
26             </select>
27         </div>
28         <div class="col-sm-3">
29             <div class="input-group">
30                 <span class="input-group-text border-0 bg-white" id="search"><i class="bi bi-search"></i></span>
31                 <input type="text" class="form-control border-0" name="search" id="searchinput" placeholder="Search..." aria-label="Search" aria-describedby="search" autocomplete="off">
32             </div>
33         </div>
34         <div class="col">
35             <div class="d-flex justify-content-end">
36                 <a href="/transaksi/surat_keluar/create" role="button" class="btn btn-light fw bold">Tambah
37                 <i class="bi bi-plus-lg"></i>
38             </a>
39         </div>
40     </div>
41 </div>
42 </div>
43
44 <div class="table responsive">
45     <table class="table text-white border-light" id="datatable" style="width: 100%;>
46         <thead class="thead">
47             <tr>
48                 <th></th>
49                 <th class="py-3 px-4">No</th>
50                 <th class="py-3 px-4">JENIS SURAT</th>
51                 <th class="py-3 px-4">TUJUAN SURAT</th>
52                 <th class="py-3 px-4">SIKAF</th>
53                 <th class="py-3 px-4">NO SURAT</th>
54                 <th class="py-3 px-4">PERHAL SURAT</th>
55                 <th class="py-3 px-4">TANGGAL SURAT</th>
56                 <th class="py-3 px-4">SEMESTER</th>
57                 <th class="py-3 px-4">AKSI</th>
58             </tr>
59         </thead>
60         <tbody class="tbody">
61             <tr>
62                 <th colspan="8" class="text-white">{{ $data }}
63             </tr>
64             <tr>
65                 <th class="py-2 px-4 fw-semibold">{{ $loop->iteration }}</th>
66                 <th class="py-2 px-4">{{ $data->jenis_surat }}</th>
67                 <th class="py-2 px-4">{{ $data->tujuan_surat }}</th>
68                 <th class="py-2 px-4">{{ $data->sifat_surat }}</th>
69                 <th class="py-2 px-4">{{ $data->no_surat }}</th>
70                 <th class="py-2 px-4">{{ $data->perihal_surat }}</th>
71                 <th class="py-2 px-4">{{ Carbon\Carbon::createFromFormat('Y-m-d', $data->tanggal_surat)->locale('id')->isoformat('DD / MM / YYYY') }}</th>
72                 <th class="py-2 px-4">{{ $data->semester }}</th>
73                 <th class="py-2 px-4">{{ $data->aksi }}</th>
74             </tr>
75             <tr>
76                 <td colspan="8" class="text-white">
77                     <div class="btn-group">
78                         <button type="button" class="btn btn-sm btn-primary dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">
79                             <i class="bi bi-list btn-menu"></i>
80                         </button>
81                         <ul class="dropdown-menu">
82                             <li>
83                                 <a class="dropdown-item href="{{ route('edit-surat-keluar', ['slug' => $data->slug, 'jenis' => $data->jenis_surat, 'status' => $data->status ]) }}">
84                                     <i class="btn btn-success bi bi-pencil"></i>
85                                 </a>
86                             </li>
87                             <li>
88                                 <a class="dropdown-item href="{{ route('show-detail-laporan', ['slug' => $data->slug, 'jenis' => $data->jenis_surat, 'status' => $data->status ]) }}">
89                                     <i class="btn btn-danger bi bi-trash"></i>
90                                 </a>
91                             </li>
92                             <li>
93                                 <a class="dropdown-item href="{{ route('show-detail-laporan', ['slug' => $data->slug, 'jenis' => $data->jenis_surat, 'status' => $data->status ]) }}">
94                                     <i class="btn btn-warning bi bi-card-list"></i>
95                                 </a>
96                             </li>
97                         </ul>
98                     </div>
99                 </td>
100             </tr>
101         </tbody>
102     </table>
103 </div>
104 @endsection
```

Lampiran 1 4 Codingan *Controller*

1. Controller surat masuk

```
1 class TransaksiSuratMasukController extends Controller
2 {
3     /**
4      * Display a listing of the resource.
5      */
6     public function index()
7     {
8         return view('dashboard.transaksi.surat_masuk.index', [
9             'title' => 'Transaksi surat masuk',
10            'surats' => SuratMasuk::latest()->get(),
11        ]);
12    }
13
14    /**
15     * Show the form for creating a new resource.
16     */
17    public function create()
18    {
19        return view('dashboard.transaksi.surat_masuk.create', [
20            'title' => 'Tambah surat masuk',
21            'sifatsurats' => SifatSurat::all(),
22            'klasifikasis' => Klasifikasi::all()
23        ]);
24    }
25
26    /**
27     * Store a newly created resource in storage.
28     */
29    public function store(Request $request)
30    {
31
32        $validateData = $request->validate([
33            'sifat_surat_id' => 'required',
34            'klasifikasi_id' => 'required|max:30',
35            'pengirim' => 'required|max:150',
36            'nomor_surat' => 'required|max:35',
37            'slug' => 'required|unique:surat_masuks',
38            'tanggal_surat' => 'required|max:10',
39            'perihal_surat' => 'required|max:255',
40            'file' => 'file|mimes:pdf,jpeg,png|max:1024',
41            'tanggal_diterima' => 'required|max:10',
42            'semester' => 'required|max:7',
43        ]);
44
45        // $validateData['image'] = $request->file('image')->store('suratmasuk-images');
46        $file = Str::random(100) . '.' . $request->file('file')->getClientOriginalExtension();
47        $request->file('file')->move(public_path() . '/suratmasuk-files/', $file);
48        $validateData['file'] = $file;
49
50        SuratMasuk::create($validateData);
51
52        // Set data dalam email
53        $dataMail = $validateData;
54        $emailUser = User::where('level', 'is_user')->get('email');
55        Mail::to($emailUser)->send(new SendEmailSuratMasuk($dataMail));
56
57        return redirect('/transaksi/surat-masuk')
58            ->with('success', 'New incoming mail has been added!');
59    }
}
```

```

1 public function edit(SuratMasuk $suratMasuk)
2 {
3     // ddd($suratMasuk);
4     return view('dashboard.transaksi.surat_masuk.edit', [
5         'title' => 'Edit surat masuk',
6         'surat' => $suratMasuk,
7         'sifat-surats' => SifatSurat::all(),
8         'klasifikasis' => Klasifikasi::all()
9     ]);
10 }
11
12 /**
13  * Update the specified resource in storage.
14  */
15 public function update(Request $request, SuratMasuk $suratMasuk)
16 {
17     $rules = [
18         'sifat_surat_id' => 'required',
19         'klasifikasi_id' => 'required|max:30',
20         'pengirim' => 'required|max:150',
21         'nomor_surat' => 'required|max:35',
22         'tanggal_surat' => 'required|max:10',
23         'perihal_surat' => 'required|max:255',
24         'file' => 'file|mimes:pdf,jpeg,png|max:1024',
25         'tanggal_diterima' => 'required|max:10',
26         'semester' => 'required|max:10',
27     ];
28
29     // Cek apakah ada slug baru
30     if ($request->slug != $suratMasuk->slug) {
31         $rules['slug'] = 'required|unique:surat_masuks';
32     }
33
34     $validateData = $request->validate($rules);
35
36     // Cek image
37     if ($request->file('file')) {
38         // jika ada image baru, hapus image lama di folder storage
39         if ($request->oldfile) {
40             File::delete(public_path() . '/suratmasuk-files/' . $request->oldfile);
41         }
42         $file = Str::random(100) . '.' . $request->file('file')->getClientOriginalExtension();
43         $request->file('file')->move(public_path() . '/suratmasuk-files/', $file);
44         $validateData['file'] = $file;
45     }
46
47     SuratMasuk::where('id', $suratMasuk->id)
48         ->update($validateData);
49
50     return redirect('/transaksi/surat-masuk')
51         ->with('success', 'Mail has been Update!');
52 }
53
54 /**
55  * Remove the specified resource from storage.
56  */
57 public function destroy(SuratMasuk $suratMasuk)
58 {
59     // Delete beserta gambar
60     if ($suratMasuk->file) {
61         File::delete(public_path() . '/suratmasuk-files/' . $suratMasuk->file);
62     }
63     SuratMasuk::destroy($suratMasuk->id);
64     return redirect('/transaksi/surat-masuk')
65         ->with('success', 'Mail has been deleted!');
66 }
67
68 // Slug
69 public function checkSlug(Request $request)
70 {
71     $slug = SlugService::createSlug(SuratMasuk::class, 'slug', $request->nomor_surat);
72     return response()->json(['slug' => $slug]);
73 }
74 }
75 }
76

```


2. Controller surat keluar

```
1 class TransaksiSuratKeluarController extends Controller
2 {
3     /**
4      * Display a listing of the resource.
5      */
6     public function index()
7     {
8         return view('dashboard.transaksi.surat_keluar.index', [
9             'title' => 'Transaksi surat keluar',
10            'surats' => SuratKeluar::with('suratKeterangan', 'suratpermohonan', 'suratpemberitahuan')->latest()->get(),
11        ]);
12    }
13
14    /**
15     * Show the form for creating a new resource.
16     */
17    public function create()
18    {
19        $nextNomorSurat = SuratKeluar::generateNomorSurat();
20
21        return view('dashboard.transaksi.surat_keluar.create', [
22            'title' => 'Tambah surat keluar',
23            'nextNomorSurat' => $nextNomorSurat,
24            'sifatsurats' => SifatSurat::all()
25        ]);
26    }
27
28    /**
29     * Store a newly created resource in storage.
30     */
31
32    public function store(Request $request)
33    {
34        $dataSuratKeluar = $request->validate([
35            'slug' => 'required',
36            'nomor_surat' => 'required|max:35',
37            'jenis_surat' => 'required|max:30',
38            'sifat_surat_id' => 'required',
39            'status' => 'required|max:15',
40            'tujuan_surat' => 'required|max:100',
41            'tanggal_surat' => 'required|max:10',
42            'semester' => 'required|max:7',
43            'perihal_surat' => 'required|max:255',
44            'lampiran' => 'required|max:2',
45        ]);
46
47        // Memeriksa apakah nomor surat sudah ada dalam database
48        if (SuratKeluar::where('nomor_surat', $dataSuratKeluar['nomor_surat'])->exists()) {
49            return redirect('/transaksi/surat-keluar/create')
50                ->with('error', $dataSuratKeluar['nomor_surat'] . ' Nomor surat sudah tersedia di database');
51        }
52
53        $suratKeluar = SuratKeluar::create($dataSuratKeluar);
54    }
55 }
```

```

1  public function edit($slug, $jenis, $status)
2  {
3
4      if ($jenis == 'Surat Keterangan') {
5          $jenis = 'surat_keterangan';
6          $data = SuratKeluar::with('suratketerangan', 'suratpermohonan')->where('slug', $slug)->first();
7          return view("dashboard.transaksi.surat_keluar.view.$jenis.edit", [
8              'title' => 'Edit surat keluar',
9              'surat' => $data,
10             'sifatsurats' => SifatSurat::all()
11         ]);
12     } else if ($jenis == 'Surat Permohonan') {
13         $jenis = 'surat_permohonan';
14         $data = SuratKeluar::with('suratpermohonan')->where('slug', $slug)->first();
15         return view("dashboard.transaksi.surat_keluar.view.$jenis.edit", [
16             'title' => 'Edit surat keluar',
17             'surat' => $data,
18             'sifatsurats' => SifatSurat::all()
19         ]);
20     } else if ($jenis == 'Surat Pemberitahuan') {
21         $jenis = 'surat_pemberitahuan';
22         $data = SuratKeluar::with('suratpemberitahuan')->where('slug', $slug)->first();
23         return view("dashboard.transaksi.surat_keluar.view.$jenis.edit", [
24             'title' => 'Edit surat keluar',
25             'surat' => $data,
26             'sifatsurats' => SifatSurat::all()
27         ]);
28     } else {
29         abort(404, 'Not Found.');
```

```

30     }
31 }
32
33
34 /**
35  * Update the specified resource in storage.
36  */
37
38 public function update(Request $request, SuratKeluar $suratKeluar)
39 {
40     $rulesSuratKeluar = [
41         'nomor_surat' => 'required|max:35',
42         'jenis_surat' => 'required|max:30',
43         'sifat_surat_id' => 'required',
44         'status' => 'required|max:15',
45         'tujuan_surat' => 'required|max:100',
46         'tanggal_surat' => 'required|max:10',
47         'semester' => 'required|max:7',
48         'perihal_surat' => 'required|max:255',
49         'lampiran' => 'required|max:2',
50     ];
51
52     if ($request->slug != $suratKeluar->slug) {
53         $rulesSuratKeluar['slug'] = 'required|unique:surat_keluars';
54     }
55     $validateData = $request->validate($rulesSuratKeluar);
56

```

Akses aplikasi sepenuhnya di github.com :