BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Manusia sangat membutuhkan teknologi pada zaman dulu hingga zaman sekarang. Pekerjaan manusia semakin mudah dengan adanya teknologi. Untuk mengikuti perkembangan teknologi manusia dituntut untuk menimba ilmu setinggi-tingginya. Sebagaimana dalam agama Islam, umatnya diperintahkan untuk menuntut ilmu, Allah berfirman:

قُلْ هَلْ يَسْتَوِي ٱلَّذِينَ يَعْلَمُونَ وَٱلَّذِينَ لَا يَعْلَمُونَۗ إِنَّمَا يَتَذَكَّرُ أُوْلُواْ ٱلْأَلْبَٰبِ ٩

Artinya, *"Katakanlah: "Adakah sama orang-orang yang mengetahui dengan orang-orang yang tidak mengetahui?" Sesungguhnya orang yang berakallah yang dapat menerima pelajaran."* (QS Az Zumar [39]:9) [1]

Dalam ayat di atas Allah berfirman "Katakanlah: Adakah sama orang-orang yang mengetahui dengan orang-orang yang tidak mengetahui?" Yaitu, apakah orang ini sama dengan orang yang menjadikan tandingan-tandingan bagi Allah untuk menyesatkan (manusia) dari jalan-Nya? "Sesungguhnya orang yang berakallah yang dapat menerima pelajaran." Yaitu, yang mengetahui perbedaan antara orang ini dengan orang itu hanyalah orang yang memiliki inti pemikiran, yaitu akal [1].

Berdasarkan ayat dan tafsir di atas penulis menyimpulkan bahwa orang-orang yang berilmu dan berakal mempunyai kedudukan yang lebih mulia di mata Allah dan tidaklah disamakan kedudukannya dengan orang yang tidak berilmu dan tidak berakal. Ayat tersebut juga menggambarkan bahwa

pentingnya ilmu untuk kita sebagai manusia dalam menjalani kehidupan baik di dunia maupun untuk akhirat dalam mencapai ketaatan kepada Allah. Sehingga dapat diambil kesimpulan bahwa Islam sangat menjunjung tinggi kedudukan ilmu.

Salah satu yang membedakan Islam dengan yang lainnya adalah penekanannya terhadap masalah ilmu. Al-qur'an dan hadits mengajak umat Islam untuk mencari dan mendapatkan ilmu dan kearifan, serta menempatkan orang-orang berilmu pada derajat yang tinggi. Ada beberapa istilah yang dipakai untuk menyebutkan ilmu pengetahuan ilmu, pengetahuan, *al-'ilm* dan *sains*. Dalam Al-Qur'an, kata *ilm* ternyata memang banyak disebut, yaitu sebanyak 105 kali, dengan kata jadiannya disebut tak kurang dari 744 kali.[2] Hal ini dapat membuktikan bahwa pentingnya ilmu pengetahuan bagi kita, dan salah satu upaya untuk mendapatkan ilmu pengetahuan yaitu dengan pendidikan.

Aktualisasi pendidikan di negara Indonesia telah tertuang dalam undang-undang Sistem pendidikan nasional tahun 2003 pasal 1 ayat 1 yang menegaskan bahwa pendidikan adalah usaha sadar dan terencana untuk mewujudkan suasana belajar dan proses pembelajaran agar peserta didik secara aktif mengembangkan potensi dirinya untuk memiliki kekuatan spiritual keagamaan, pengendalian diri, kepribadian, kecerdasan, akhlak mulia, serta keterampilan yang diperlukan dirinya, masyarakat, bangsa dan negara. [3]

Pengertian di atas menjelaskan bahwa pendidikan adalah upaya dalam meningkatkan, memperbaiki, mengubah pengetahuan, keterampilan, serta perilaku seseorang sebagai usaha mencerdaskan manusia melalui usaha-usaha kependidikan. Salah satu bentuk usaha tersebut ialah dengan adanya Lembaga-lembaga pendidikan yang memiliki tingkatan-tingkatan pendidikan tertentu. Salah satu lembaga pendidikan tersebut ialah Perguruan Tinggi.

Perguruan Tinggi adalah lembaga formal tertinggi tempat masyarakat menimba pendidikan. Lembaga ini menerapkan program pendidikan dengan melibatkan berbagai komponen yang terdapat di dalamnya. Salah satu komponen yang terpenting dalam sebuah lembaga perguruan tinggi adalah mahasiswa. Berhasil atau tidaknya suatu lembaga ini dapat dilihat dari kuantitas dan kualitas mahasiswa yang menjadi ujung tombak dari perguruan tinggi tersebut.[4] Salah satu lembaga perguruan tinggi tersebut ialah Universitas Islam Negeri (UIN) Bukittinggi.

UIN Bukittinggi merupakan salah satu Perguruan Tinggi Negeri Islam yang sedang berkembang di kota Bukittinggi dimana saat ini telah memiliki ribuan orang mahasiswa yang berkembang pesat dari tahun ketahun. Terdiri dari beberapa fakultas, salah satunya Fakultas Ekonomi dan Bisnis Islam (FEBI). Setiap fakultas memiliki layanan yang dapat memenuhi keperluan administrasi mahasiswa, salah satu nya dalam pelayanan pengurusan surat.

Administrasi berasal dari Bahasa Latin Administrare yang memiliki arti membantu atau melayani. Dalam bahasa Inggris perkataan administrasi

berasal dari kata *administration*, yang artinya melayani, mengendalikan, atau mengelola suatu organisasi dalam mencapai tujuannya secara intensif[5]. Administrasi memiliki peranan yang sangat penting dalam pendidikan untuk mengelola dan mengatur serta mengalokasikan sumber daya, administrasi juga merupakan alat untuk menyatukan dan menyelaraskan peranan seluruh sumber daya yang dimiliki agar tercapainya tujuan pendidikan itu sendiri.

Pentingnya administrasi juga sudah diatur dalam Peraturan Pemerintah Nomor 4 Tahun 2017 Tentang Penyelenggaraan dan Pengelolaan Perguruan Tinggi, yang mana pada peraturan ini dijelaskan bahwa perguruan tinggi adalah pengatur, perencana, pengawas, pemantau, dan evaluasi serta pembinaan dan koordinasi pelaksana jalur, jenjang, dan jenis Pendidikan Tinggi oleh menteri untuk mencapai tujuan Peguruan Tinggi[6]. Dari penjelasan ayat tersebut dapat diartikan bahwa peguruan tinggi berperan sebagai penyelenggaran pendidikan sekaligus pengelola.

Dalam sebuah perguruan tinggi kegiatan kemahasiswaan tidak dapat dipisahkan dari administrasi, mulai dari pendaftaran mahasiswa baru sampai kepada proses kegiatan perkuliahan, pengurusan pengajuan dan penerbitan surat juga merupakan bagian dari administrasi akademik yang harus ditangani oleh staff di masing – masing fakultas.

Surat merupakan salah satu media komunikasi tertulis untuk menyampaikan informasi pernyataan, pesan kepada pihak lain yang mempunyai keperluan kegiatan dengan berbagai pihak tertentu, alat

komunikasi tertulis yang berasal dari satu pihak dan ditujukan kepada pihak lain untuk menyampaikan berita, atau sejenis warkat yang dipergunakan sebagai sarana komunikasi tertulis antara pihak pertama dengan pihak lain dengan mempergunakan kertas berukuran tertentu. Surat menyurat memiliki beberapa jenis baik dilihat dari sisi dan tujuan maupun wujud dan sifatnya[7]. Dalam arti sempit surat merupakan sarana kounikasi yang digunakan untuk menyampaikan informasi tertulis oleh satu pihak ke pikah lain.

Pada umumnya mahasiswa diharuskan untuk datang ke fakultas untuk pengurusan administrasi surat, kemudian mengikuti prosedur yang sudah ditetapkan mulai dari pengajuan kemudian surat diproses dan bisa diterima kembali oleh mahasiswa setelah diterbitkan. Dengan perkembangan teknologi semua prosedur tersebut mulai digantikan dengan sistem komputerisasi yang disebut sistem informasi untuk menangani administrasi surat mahasiswa dengan tujuan agar proses dalan pengajuan dan penerbitan surat tersebut bisa di selesaikan dengan lebih cepat.

Hasil wawancara penulis bersama kepala subbagian umum dan administrasi FEBI Bapak Elmin Pasaribu pada 22 April 2020, pengurusan surat mahasiswa sudah dialihkan ke media sosial Whatsapp kemudian staff yang berada di fakultas akan menginput data mahasiswa kedalam sistem e – campus lalu memproses surat tersebut hingga diterbitkan. Kemudian mahasiswa datang ke fakultas untuk menjemput surat yang sudah diterbitkan, surat yang narasumber menyampaikan bahwa surat yang diterbitkan per hari nya mencapai 150 lembar surat, akan tetapi tetap terjadi penumpukan

mahasiswa di fakultas ketika mengambil surat, hal ini terjadi karena banyaknya jumlah mahasiswa tidak sebandingan dengan kapasitas staff yang bekerja di fakultas, ditambahkan lagi ketika ada informasi mengenai beasiswa dikampus maupun luar kampus, pengajuan surat mahasiswa menjadi lebih banyak dari biasanya, hal ini dinilai sangat tidak efektif karena staff yang bekerja hanya 2 orang di fakultas saat penelitian ini dilakukan. Penulis menyimpulkan bahwa pengurusan surat menggunakan sistem yang sekarang masih kurang efektif dan masih banyak kendala baik itu dari SDM maupun sistem itu sendiri, beberapa hal mungkin terjadi seperti lonjakan pengurusan surat seperti yang dijelaskan, kemudian tidak adanya sistem khusus untuk administrasi surat tersebut, kemudian yang terjadi di media sosial WhatsApp adalah penumpukan chat yang masuk sehingga chat atau data yang masuk tidak terbaca lagi oleh staff, terjadi antrian panjang ketika pengambilan surat di fakultas, dan belum adanya sistem pengarsipan elektronik khusus.

Kemudian tahun 2022 ketika penelitian selanjutnya dilakukan, Ibuk Syafnidar sebagai kepala subbagian yang menggantikan Bapak Elmin Pasaribu menjelaskan bahwa sekarang prosedur pengurusan surat mahasiswa sudah dialihkan ke media e-mail yang kemudian mahasiswa juga akan menerima e-mail balasan berupa surat yang telah di setuju oleh fakultas, beberapa surat yang dimaksud seperti; surat keterangan aktif kuliah mahasiswa, surat izin penelitian atau observasi.

Prosedurnya tidak jauh berbeda dengan yang sebelumnya, hanya saja media WhatsApp digantikan menggunakan e-mail. Pengumuman resmi

dikirimkan ke wesite resmi Fakultas Ekonomi dan Bisnis Islam tentang prosedur penerbitan surat aktif kuliah, surat izin penelitian, surat observasi, dan SK pembimbing. Kemudian mahasiswa mengirim data lengkap dengan persyaratan yang dibutuhkan untuk pengurusan surat, ketika syarat sudah lengkap surat akan di proses dan dikembalikan ke alamat email pengirim, jika belum maka akan diberitahu untuk memenuhi persyaratan terlebih dahulu. Narasumber juga menjelaskan untuk penomoran surat sendiri tetap menggunakan format yang ada didalam sistem tata persuratan di e-campus.

Berdasarkan wawancara tersebut penulis menyimpulkan bahwa sistem pengurusan surat yang ada sekarang masih belum praktis, efektif, dan efisien. Dikarenakan masih banyak terdapat kendala – kendala yang ditimbulkan. Maka dari itu penulis memberikan solusi bahwa perlu adanya sistem yang terkomputerisasi yang dapat mengatur dan membantu kerja dari beberapa pihak seperti mahasiswa dan staff yang bekerja menganani masalah pengurusan surat tersebut.

Salah satu sistem yang dapat membantu proses tersebut adalah sistem administrasi pengajuan dan penerbitan surat. Sistem ini berupa web yang menggunakan bahasa pemograman PHP dan database MySQL, serta berbasis *online* internet supaya bisa diakses dimana saja. Dengan adanya sebuah sistem yang digunakan, maka mahasiswa yang akan melakukan pengurusan surat tersebut tinggal mengakses via internet baik itu PC maupun smartphone. Berdasarkan latar belakang masalah di atas, maka penelitian ini penulis beri judul:

“SISTEM ADMINISTRASI PENGAJUAN DAN PENERBITAN SURAT DI FAKULTAS EKONOMI DAN BISNIS ISLAM UIN BUKITTINGGI”

B. Identifikasi Masalah

Identifikasi masalah berdasarkan latar belakang masalah yang telah dikemukakan di atas, penulis menyimpulkan bahwa pengajuan dan penerbitan surat selama ini belum menggunakan sistem informasi khusus dalam pelayanannya sehingga:

a. Pengurusan surat masih dikerjakan secara manual/semi – otomatis.
b. Jumlah surat yang diterbitkan masih terbatas.
c. Terjadi antrian panjang ketika mahasiswa hendak mengambil surat di fakultas
d. Mahasiswa harus datang ke fakultas untuk memastikan surat sudah diterbitkan atau belum.
e. Belum adanya sistem informasi khusus yang menganani masalah administrasi pengajuan dan penerbitan surat di Fakultas Ekonomi dan Bisnis Islam (FEBI).
f. Belum adanya sistem informasi untuk pengarsipan elektronik.

C. Batasan Masalah

Agar pembahasan menjadi lebih rinci dan terarah, maka penulis menerapkan batasan masalah dengan memfokuskan pada hal: Sistem Administrasi Pengajuan dan Penerbitan Surat di Fakultas Ekonomi dan Bisnis Islam UIN Bukittinggi menggunakan bahasa pemograman PHP dan database MySQL Berbasis *Online* internet.

D. Rumusan Masalah

Bertitik tolak dari batasan masalah di atas, maka dapat dirumuskan permasalahan dalam penelitian ini yaitu: "Bagaimana Sistem Administrasi Pengajuan dan Penerbitan Surat di Fakultas Ekonomi dan Bisnis Islam UIN Bukittinggi menggunakan bahasa pemograman PHP dan *database* MySQL Berbasis *Online* internet yang valid, praktis, dan efektif?"

E. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk: Untuk menghasilkan Sistem Administrasi Pengajuan dan Penerbitan Surat di Fakultas Ekonomi dan Bisnis Islam UIN Bukittinggi menggunakan Bahasa pemograman Laravel dan *database* MySQL berbasis *Online* internet yang valid, praktis, dan efektif.

F. Manfaat Penelitian

Hasil penelitian diharapkan memberi manfaat untuk:

1. Sistem administrasi pengajuan dan penerbitan surat ini dapat menjadi media pilihan baru bagi mahasiswa untuk pengurusan surat tanpa harus ke fakultas
2. Sistem ini dapat membantu dalam pelayanan pengajuan dan penerbitan mahasiswa agar lebih praktis
3. Membantu dalam memanajemen pengajuan dan penerbitan surat sekaligus pengarsipan surat tersebut secara elektronik
4. Memenuhi salah satu syarat untuk mendapatkan gelar kesarjanaan pada program studi PTIK di UIN Bukittinggi.

G. Penjelasan Judul

Untuk memudahkan dalam membaca skripsi ini dan untuk menghindari kesalahpahaman terhadap judul skripsi ini, maka penulis perlu menjelaskan beberapa istilah yang terdapat pada judul tersebut yaitu:

Sistem	:	Kumpulan atau himpunan dari unsur atau komponen atau variabel yang terorganisir, saling tergantung satu sama lain, dan terpadu [8].
Administrasi	:	Seluruh kegiatan dari pengaturan hingga pengurusan sekelompok orang yang memiliki diferensiasi pekerjaan untuk mencapai suatu tujuan bersama[9].

Surat : Surat merupakan kertas dan sebagainya yang tertulis atau secarik kertas sebagai keterangan atas sesuatu yang tertulis[10].

A. Sistematika Penulisan

Untuk mempermudah dalam penulisan skripsi ini, penulis menjabarkan beberapa bagian dalam penulisan ini, seperti yang dibawah ini :

BAB 1 Pendahuluan

Dalam bab ini penulis akan mendeskripsikan secara umum permasalahan yang ditemukan saat wawancara dilapangan yang di mulai dari penulisan latar belakang masalah, identifikasi masalah, rumusan masalah, tujuan dan manfaatnya, penjelasan judul dan sistematika penulisan.

BAB 2 Landasan Teori

Dalam bab ini penulis menjelaskan pembahasan tentang landasan teori yang berisikan materi tentang sistem informasi dan pengelolan aset sekolah.

BAB 3 Metodelogi Penelitian

Dalam bab ini penulis menjelaskan pembahasan tentang tentang waktu dan tempat penelitian, metode penelitian, model pengembagan sistem, tahapan penelitian, dan teknik pengujian produk.

BAB 4 Hasil dan Pembahasan

Dalam bab ini penulis menjelaskan pembahasan tentang hasil penelitian.

BAB 5 Penutup

Dalam bab ini berisi tentang kesimpulan dan saran-sara

BAB II

LANDASAN TEORI

A. Sistem Informasi Administrasi

Sistem berasal dari kata yunani yang artinya kesatuan. Suatu sistem terdiri dair eleme yang saling berinteraksi untuk mencapai tujuan tertentu. Sistem adalah jaringan kerja yang terdiri dari prosedur yang saling berhubunga, berkumpul sama – sama untuk melakukan suatu kegiatan atau menyelesaikan tujuan atau sasaran tertentu[10]. Sistem adalah kumpulan atau himpunan dari unsur atau komponen atau variabel yang terorganisir, saling berinteraksi, saling tergantung sama lain, dan terpadu. Dari penjelasan diatas dapat diartikan bahwa sistem bekerja dalam suatu jaringan kerja yang saling terkait dan berinteraksi satu sama lain untuk menyelesaikan suatu tujuan atau sasaran yang dimaksud.

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan informasi yang diperlukan untuk pengambilan keputusan[11]. Sistem informasi juga dapat diartikan sebagai kerangka kerja yang mengkoordinasikan sumber daya untuk mengubah masukan data menjadi informasi untuk mencapai tujuan.

Administrasi adalah suatu atau kegiatan yang berkaitan dengan penyelenggaraan kebijakan untuk mencapai tujuan atau keseluruhan proses kerjasama antara dua orang manusia atau lebih yang didasarkan atas rasionalitas tertentu untuk mencapai tujuan yang telah ditentukan[11].

Sedangkan sistem informasi administrasi merupakan sebuah sistem atau kerangka kerja dalam sebuah organisasi yang dapat mengintegrasikan sumber daya dan data, lalu memproses data tersebut mejadi sebuah informasi yang berguna dalam sebuah organisasi penyelenggara kebijakan yang bertujuan untuk mencapai suatu sasaran yang telah ditentukan sebelumnya.

B. Pengajuan dan Penerbitan Surat

1. Pengertian Surat

Surat merupakan alat atau sarana untuk menyampaikan pernyataan atau informasi secara tertulis dari pihak satu kepada pihak lainnya. Informasi tersebut dapat berupa pemberitahuan, pernyataan, permintaan, laporan, sanggahan, pemikiran, dan sebagainya. Pada umunya surat digunakan oleh perusahaan atau organisasi karena surat berperan sebagai alat pengingat dan sebahai bahan dokumentasi bagi setiap proses baik internal maupun eksternal[9].

2. Jenis Surat

Surat dapat dibagi dalam beberapa jenis berdasarkan tujuan, sofat, wujud, dan kriteria lainnya. Agar memudahkan kita mengetahui jenis surat, berikut ini adalah macam-macam surat tersebut :

a. Jenis Surat Berdasarkan Tujuannya

1. Surat pemberitahuan, biasanya berisi pemberitahuan kepada banyak pihak yang merupakan bagian dari suatu instansi
2. Surat perintah, surat yang diberikan kepada pihak tertentu agar melakukan sesuatu yang tertera dalam surat tersebut
3. Surat permintaan, surat yang isinya meminta informasi lebih detail mengenai barang atau jasa yang ditawarkan
4. Surat peringatan, surat ini bertujuan untuk mengingatkan kesalahan seseorang di suatu organisasi
5. Surat panggilan, bertujuan untuk memanggil pihak lain baik secara personal maupun organisasi
6. Surat susulan, merupakan penegasan dari surat yang pernah dikirim sebelumnya

7. Surat keputusan, surat yang berisi keputusan yang diambil oleh suartu organisasi atas dasar pertimbangan
8. Surat laporan, surat yang berisi laporan kerja kepada orang lain
9. Surat penawaran, umumnya surat ini berhubungan dengan jual-beli barang atau jasa yang isinya mengenai penawaran spesifikasi dan harga

b. Jenis Surat Berdasarkan Wujudnya

1. Kartu pos, surat yang isinya ditulis pada secarik kertas dan dikirim melalui pos. Biasanya berukuran 10x15 cm
2. Warkat pos, surat tertutup yang dibuat dari sehelai kertas cetakan dan dapat dilihat seperti amplop
3. Surat bersampul, surat yang dibuat dari dalam beberapa helaian kertas terpisah dimana untuk mengirimnya harus menggunakan amplop.
4. Memorandum dan Nota, surat yang digunakan untuk keperluan internal suatu organisasi yang bertujuan untuk memberi/meminta informasi antar pejabat kantor

5. Telegram, pesan singkat yang dikirim dari jarak jauh dalam waktu cepat dengan bantuan pesawat telegram

c. Jenis Surat Menurut Sifat Isi dan Asalnya

1. Surat dinas, digunakan untuk kepentingan tugas dinas oleh instansi-instansi tertentu, baik instansi sipil maupun militer
2. Surat Niaga/Bisnis, surat yang dipakai untuk keperluan bisnis oleh lembaga-lembaga usaha.
3. Surat Pribadi, surat yang sifatnya pribadi yang dikirimkan kepada orang lain atau petugas lembaga

d. Jenis Surat Menurut Jumlah Penerima

1. Surat biasa, umumnya ditujukan kepada satu orang atau petugas lembaga.
2. Surat edaran, ditujukan kepada banyak orang atau banyak lembaga, bahkan kepada seluruh masyarakat.
3. Surat pengumuman, umumnya ditujukan kepada sekelompok orang atau lembaga.

e. Jenis Surat Menurut Kerahasiannya

1. Surat biasa, surat yang isinya tidak terdapat hal-hal yang perlu dirahasiakan
2. Surat rahasia, surat yang isinya hanya boleh diketahui oleh orang tertentu yang berkepntingan saja.

3. Surat sangat rahasia, surat yang kerahasiaan isinya dijaga ketat agar tidak diketahui oleh orang lain.

f. Jenis Surat Berdasarkan Jangkauannya

1. Surat Internal, yaitu surat yang asalnya dari dalam suatu organisasi/perusahaan dan ditujukan ke divisi lain dalam perusahaan yang sama.
2. Surat Eksternal, yaitu surat yang asalnya dari luar organisasi/perusahaan dan ditujukan untuk organisasi atau perusahaan yang lain[12].

3. Pengajuan dan Penerbitan Surat

Pengajuan atau mengajukan menurut KKBI merupakan mengemukakan (usul, permintaan, sanggahan, protes, dan sebagainya). Pegajuan juga dapat diartikan sebagai kegiatan mengemukakan atau mengusulkan sesuatu dan sebagainya. Pengajuan surat dapat diartikan sebagai kegiatan mengusulkan surat permintaan/permohonan untuk hal tertentu yang berkaitan dengan organisasi tertentu yang dapat digunakan untuk menyampaikan informasi ke pihak tertentu.

Dalam akademik hal ini berkaitan dengan kegiatan pengajuan surat mahasiswa seperti :

- Surat Observasi
- Surat Keterangan Lulus
- Surat Rekomendasi Beasiswa

- Surat Praktek Mata Kuliah
- Surat Penunjukan Promotor
- Surat Keterangan telah Selesaiu Penelitian
- Surat Uji Coba Angket Penelitian
- SK Pembimbing
- Surat Keterangan Aktif Kuliah
- Surat Izin Penelitian

Surat tersebut akan di kelola oleh sistem yang merupakan produk dari penelitian ini.

4. Penerbitan Surat

Penerbitan dalam KBBI berasal dari kata penerbit yang artinya orang yang menerbitkan atau perusahaan yang menerbitkan buku dan sebagainya[13]. Penerbitan surat merupakan suatu kegiatan menerbitkan surat dari suatu organisasi yang berwenang dan daat bertanggung jawab atas surat tersebut. Surat tersebut ditujukan untuk pihak atau organisasi lain yang ditujukan.

C. Alat Bantu Perancangan

1.UML

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Languag*e (UML). UML adalah Bahasa visual untuk pemodelan dan komunikasi

mengenai sebuah *system* dengan menggunakan diagram dan teks-teks pendukung [14].

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari *system* perangkat lunak. Jadi UML hanya berfungsi untuk melakukan pemodelan. UML merupakan gabungan dari beberapa konsep yaitu seperti konsep *Object Modelling Technique* (OMT), konsep *The Classes Responsibilities Collaborators* (CRC), konsep pemikiran Ivar Jacobson, dan beberapa konsep lainnya yang bergabung dalam sebuah perusahaan sehingga menghasilkan bahasa yang disebut dengan *Unified Modeling Languag*e (UML).

Uml terdiri dari beberapa macam diantaranya yaitu [14]:

a. *Use case Diagram*

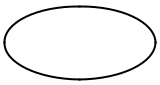
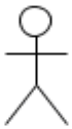
Use case atau *diagram use case* adalah pemodelan untuk kelakuakan (*behavior*) sistem informasi yang akan dibuat.*use case* mendiskripsikan sebuah interaksi antara satu atau lebih aktor dengan system informasi yang akan dibuat. Syarat penamamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu:

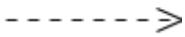
1. Aktor merupakan orang, proses, atau sistem lain yang berintekrasi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.

2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor.

Berikut adalah simbol-simbol yang ada pada *diagram use case* pada tabel 2.1:

Tabel 2.1 Simbol-simbol *Use case* [14]

No.	Simbol	Nama	Deskripsi
1.		*Use case*	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesanantar unit
			atau aktor biasanya dinyatakan meggunakan kata kerja di awal *frase* nama *use case.*
2.		*Actor*	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

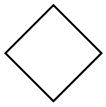
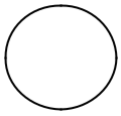
A	B	C	D
3.		*Association*	Komunikasi antara aktor dan *use case* yang berpartisispasi pada *use case* atau *use case* memiliki interaksi dengan aktor.
4.		*Extend*	Relasi *use case* dimana *use case* yang ditambahkan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan.
5.		*Generalisasi*	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6.		*Include*	Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditanbahkan memerlukan *use case* ini untuk menjalankan atau sebagai syarat dijalankan *use case* ini.

b. *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarakan *workflay* (aliran kerja) atau aktivitas dari sebuah sistem atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut adalah simbol-simbol yang ada pada diagram aktivitas terdapat pada tabel 2.2:

Tabel 2.2 Simbol-simbol *Activity Diagr*am [14]

No.	Simbol	Nama	Deskripsi
1.		*Status awal*	merupakan tanda yang berada pada sudut kiri atas yang merupakan tanda dari awal aktivitas
2.		*Status Akhir*	tanda yang diposisikan di paling akhir sebagai tanda akhir dari aktivitas

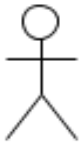
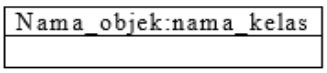
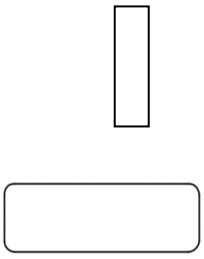
A	B	C	D
3.		*Decision*	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
4.		*Penggabungan*	Asosiasi dimana penggabungan dilakukan lebih dari satu aktivitas digabungkan menjadi satu.
5.		*Status akhir*	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

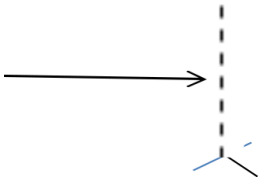
c. *Sequence Diagram*

Diagram sequence menggambarkan kelakuan objek pada *use case* dengan mendiskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *diagram sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-motode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas terdapat pada tabel 2.3:

Tabel 2.3 Simbol-simbol *Sequence Dia*gram [14]

No.	Nama dan Simbol	Deskripsi
1.	Aktor 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2.	l*ifeline*	Menyatakan kehidupan suatu objek.
3.	Objek 	Menyatakan objek yang berinteraksi pesan.
4.	Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktuaktif ini adalah sebuah tahapan yang dilakukan di dalamnya.

A	B	C
5.	Pesan tipe *create* <<create>> 	Menyatakan suatu objek membuata objek yang lain, arah panah mengarah pada objek yang dibuat.
6.	Pesan tipe *call* 1: nama_metode() 	Menyatakan suatau objek memanggil operasi atau metode yang ada pada objek atau dirinya sendiri.
7.	Pesan tipe *send* 1: masukan 	Menyatakan bahwa suatu objek Mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi
8.	Pesan tipe *return* 1: keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan sesuatu operasi atau metode menghasilakan suatu kembalian ke objek tertentu.
9.	Pesan tipe *destroy* 	Menyatakan suatu objek mengakhiri objek yang lain, arah panah mangarah pada objek yang diakhiri, sebaiknya jika ada *create* maka ada *destroy*

d. *Class diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefenisian kelas-kelas yang akan dibuat untuk membangaun sistem kelas memiliki apa yang disebut atribut dan metode atau operasi. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas sebagai berikut [14]:

1. Kelas main

   Memiliki fungsi awal untuk dieksekusi ketika sistem dijalankan.

2. Kelas tampilan sistem (*view*)

   Berfungsi untuk mendefinisikan dan mengatur tampilan.

3. Kelas pendiefinisian *use case* (*controller*)

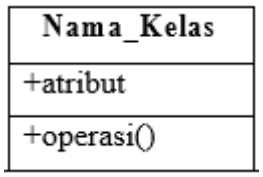
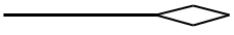
   Menangani fungsi-fungsi yang harus diambil dari pendefinisian *use case* tertentu.

4. Kelas pendefinisian data (model)

   Berfungsi untuk memegang dan membungkus data menjadi suatu kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut adalah simbol-simbol yang ada pada *class diagram* terdapat pada table 2.4:

Tabel 2.4 Simbol-simbol *Class diagram* [14]

No.	Nama dan Simbol	Deskripsi
1.	Kelas 	Kelas pada struktur sistem
2.	Antarmuka / *interface* 	Sama dengan konsep *interface* dalam pemrograman berorientasi objek.
3.	*Association*	Relasi antarkelas dengan makna umum
4.	Asosiasi berarah	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
5.	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6.	Kebergantungan 	Relasi antarkelas dengan makna kebergantungan antarkelas.
7.	Agregasi 	Relasi antarkelas dengan makna semua-bagian (*whole-part*)

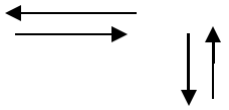
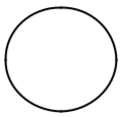
2. Bagan Alur (*Flowchart*)

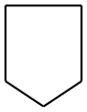
Flowchart adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. Jadi *flowchart* merupakan cara penyajian dari suatu algoritma [15]. Adapun simbol-simbol pada *flowchart* sebagai berikut:

a. *Flow direction symbols*

Simbol yang digunakan untuk menghubungkan antara simbol satu dengan simbol yang lainnya. Adapun simbol-simbolnya sebagai berikut:

Tabel 2.5 Simbol-simbol *Flow direction symbols* [15]

No.	Simbol	Nama	Fungsi
1.		*Flow (simbol arus)*	Untuk menyatakan jalannya arus sebuah proses
2.		*Communication link*	Menyatakan adanya transisi suatu data atau informasi dari satu lokasi ke lokasi lainnya.
3.		*Connector*	Menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembar yang sama.

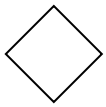
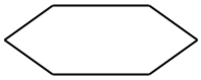
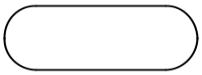
A	B	C	D
4.		*Offline Connector*	Sambungan dari satu proses ke proses lainnya dalam halaman atau lembar yang berbeda

b. *Processing symbols*

Yaitu simbol yang menunjukkan operasi pengolahan sebuah proses atau prosedur. Simbol-simbolnya sebagai berikut:

Table 2.6 Simbol-simbol *Processing symbols* [15]

No.	Simbol	Nama	Fungsi
1.		*Offline Connector*	Untuk sambungan dari satu proses ke proses lainnya dalam halaman atau lembar yang berbeda.
2.		*Manual*	Untuk suatu tindakan (proses) yang tidak dilakukan oleh komputer.

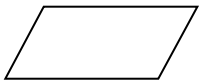
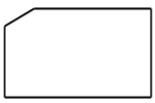
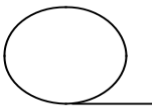
A	B	C	D
3.		*Decision (logika)*	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban (ya atau tidak).
4.		*Predefined Proses*	Penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5.		*Terminal*	Menunjukkan permulaan atau akhir dari suatu program.
6.		*Keying Operation*	Menunjukkan semua jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
7.		*Off-line storage*	Menyatakan bahwa data dalam symbol ini akan disimpan ke suatu media tertentu.

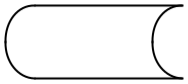
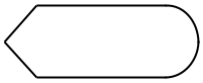
A	B	C	D
8.		*Manual input*	Memasukkan data secara manual dengan menggunakan *online keyboard.*

c. *Input-output symbols*

Simbol yang menunjukkan jenis atribut yang digunakan untuk media *input* atau *output*. Adapun simbol-simbolnya sebagai berikut:

Tabel 2.7 Simbol-simbol *Input-output symbols* [15]

No.	Simbol	Nama	Fungsi
1.		*Input-output*	Untuk proses input dan output tanpa tergantung dengan jenis peralatannya.
2.		*Punched Card*	Menunjukkan input berasal dari kartu atau output ditulis ke kartu.
3.		*Magnetic-tape unit*	Menunjukkan input berasal dari pita magnetic atau output disimpan ke pita magnec.

A	B	C	D
4.		*Disk storage*	Untuk menunjukkan input berasal dari disk atau output disimpan ke disk.
5.		*Document*	Mencetak laporan ke printer
6.		*Display*	Menunjukkan peralatan output yang digunakan berupa layar (video, komputer).

D. Sofware Perancangan Aplikasi

1. Pengertian PHP (Hypertext Preprocessor)

Syafi'i berpendapat bahwa PHP bahasa pemrograman yang berfungsi untuk membuat *website* dinamis maupun aplikasi web. Berbeda dengan HTML yang hanya bisa menampilkan konten statis, PHP bisa berinteraksi dengan *database*, *file* dan *folder*, contohnya Blog, Toko *Online*, CMS, Forum, dan *Website Social Networking*. PHP adalah bahasa *scripting*, bukan bahasa *tag-based* seperti HTML. PHP termasuk

bahasa *cross-platform*, ini artinya PHP bisa berjalan di sistem operasi yang berbeda-beda (Windows, Linux, ataupun MAC).

"Untuk dapat berjalan, PHP membutuhkan web *server*, yang bertugas untuk memproses file php dan mengirimkan hasil pemrosesan yang akan ditampilkan di *browser client*. Oleh karena itu, PHP termasuk *server-side scripting* (script yang diproses di *server*). Web *server* sendiri adalah *software* yang diinstal di komputer lokal ataupun komputer lain yang berada di jaringan intranet/internet yang berfungsi untuk melayani permintaan-permintaan web dari *client*. Web *server* yang paling digunakan saat ini untuk PHP adalah "Apache". Untuk media penyimpanan datanya (*database server*), PHP biasa menggunakan MySQL.

"Untuk menginstall dan mengkonfigurasikan ketiga *software* tersebut (Apache, PHP, MySQL) agar dapat berjalan dan selalu terhubung, memang cukup sulit. Maka dari itu dibuatlah paket *software* LAMP, XAMPP, MAMP, WAMP yang tinggal kita install dalam satu kali instalasi. Dalam satu kali instalasi, sudah mencakup ketiga *software* tersebut dan sudah dikonfigurasikan untuk keperluan lingkungan pengembangan aplikasi web"[16].

2. XAMPP

Menurut MADCOMS (2016) "Xampp adalah sebuah paket kumpulan software yang terdiri dari *Apache*, *MySQL*, *PhpMyAdmin*, *PHP*, *Perl*, *Filezilla*, dan lain." Xampp berfungsi untuk memudahkan

instalasi lingkungan PHP, di mana biasanya lingkungan pengembangan web memerlukan *PHP, Apache, MySQL* dan *PhpMyAdmin*[17].

3. MySQL

MySQL adalah sebuah server database open source yang terkenal yang digunakan berbagai aplikasi terutama untuk server atau membuat WEB. Mysql berfungsi sebagai SQL *(Structured Query Langguage*) yang dimiliki sendiri dan sudah diperluas oleh MySQL umumnya digunakan bersamaan dengan PHP untuk membuat aplikasi server yang dinamis dan powerfull. Tidak sama dengan proyek-proyek seperti apache, dimana perangkat lunak dikembangkan oleh komunikasi umum, dan hak cipta untuk kode sumber dimiliki oleh penulisanya masing-masing. Mysql dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang finlandia yang mendirikan MySQL AB adalah David Axmark, Allan Larson dan Mhicael “Monty”Widenius[16].

MySQL adalah sebuah implementasi dari sistem menajemen basisdata relasional (RDMBS) yang didistribusikan secara gratis dibawah lisensi GPL *(General Public License).* Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL *(Structured Query*

Langguage). SQL adalah sebuah konsep pengoperasian basisdata, Terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Kehandalan suatu sistem basis data (DBMS) dapat diketahui dari cara kerja pengoptimasi-nya dalam melakukan proses perintah-perintah memanfaatkan nya. Sebagai peladen basis data, MySQL mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional. Pada modus operasi non-transaksional, MySQL dapat dikatakan unggul dalam hak untuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya. Namun demikian pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan. Karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis web (wordpress), CMS, dan sejenisnya[16].

E. Penelitian Relevan

1. **A. Rachmatsyah, D. Merlin, *Program Studi Sistem Informasi, SMTIK Atma Luhur,* (Perancangan Sistem Informasi Administrasi Surat Berbasis Dekstop).** Penelitian ini bertujuan untuk merancang sistem informasi administrasi surat berbasis desktop pada kantor notaris. Metode pengembangan yang digunakan dalam merancang sistem ini adalah menggunakan siklus hidup pengembangan sistem

dengan model *waterfall*. Metode *waterfall* dibagi menjadi 4 fase yaitu *planning, modeling, construction, deployment.*

2. **Fauzan Masykur, Ibnu Makruf Pandu Atmaja, *Program Studi Teknik Informatika, Universitas Muhammadiyah Ponorogo,* (Sistem Administrasi Pengelolaan Arsip Surat Masuk dan Surat Keluar).** Penilitian ini bertujuan membangun sisten administrasi pengelollan arsip surat masuk dan surat keluar pada SMK Negeri 1 Magetan, dengan Sistem tersebut akan memberi kemudahan dalam pengelolaan surat masuk dan surat keluar, sekaligus memberi kemudahan dalam pencarian arsip surat masuk dan surat keluar. Metode pengembangan yang di gunakan dalam membangun sistem informasi ini adalah Metode Pengumpulan data yang dibagi dalam 2 fase yaitu, *observation,* dan *documentation.*
3. **Muhammad Fajri Ramadhan, Lusi Fajarita, *Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Budi Luhur* (Pembuatan Aplikasi Sistem Informasi Administrasi Surat Menyurat Studi Kasus : Kesekretariatan Badan Eksekutif Mahasiswa Universitas Budi Luhur).** Penelitian ini bertujuan membangun aplikasi sistem informasi administrasi surat menyurat pada kesekretariatan badan eksekutif mahasiswa Universitas Budi Luhur. Sistem ini akan memberi kemudahan dalam pengarsipan surat masuk dan surat keluar, pembuatan laporan, proposal dan sebagainya. Metode pengembangan yang digunakan dalam membuat aplikasi ini

adalah Metode pengumpulan data yang dibagi dalam beberapa tahap yaitu, wawancara, observasi, studi pustaka, studi literatur.

BAB III
METODOLOGI PENELITIAN

A. Waktu dan Tempat Penelitian

Penelitian akan dilaksanakan di Fakultas Ekonomi dan Bisnis Islam UIN Bukittinggi. Pemilihan tempat penelitian ini dikarenakan atas dasar pertimbangan bahwa kondisi yang memungkinkan untuk sistem informasi administrasi pengajuan dan penerbitan surat seperti yang telah diuraikan dalam latar belakang masalah di Fakultas Ekonomi dan Bisnis Islam UIN Bukittinggi, dalam proses pengelolan dan informasi kehadiran siswa tidak perlu dilakukan secara manual lagi.

Penelitian akan dilaksanakan pada bulan Oktober 2020. Dari penelitian ini, penulis akan mengumpulkan data-data yang dibutuhkan untuk kemudian diolah sebagai bahan dalam perancangan notifikasi untuk memantau infomasi presensi siswa untuk membantu penelitian dalam menyelesaikan penelitian.

B. Metode Penelitian

Dalam pengembangan program ini, diperlukan data yang nantinya akan diolah sebagai bahan dalam pembuatan program. Maka dari itu, diperlukan suatu metode untuk memperoleh data tersebut. Dalam penelitian ini, metode yang digunakan adalah metode Penelitian dan Pengembangan (*Research and Development*). Secara sederhana, *Research and Development* atau lebih dikenal dengan R&D didefinisikan sebagai metode penelitian yang secara sengaja, sistematis, bertujuan atau

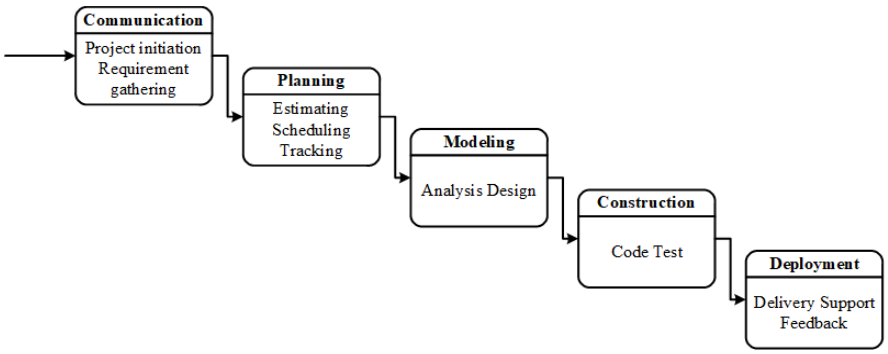
diarahkan untuk mencari temukan, merumuskan memperbaiki, mengembangkan, menghasilkan, menguji keefektifan produk, model, metode, jasa, prosedur tertentu yang lebih unggul, baru,efektif, efisien, produktif dan bermakna[18].

C. Model Pengembangan Sistem

Model pengembangan sistem yang diterapkan dalam penelitian ini adalah model *System Development Life Cycle* (SDLC) atau dalam bahasa Indonesia dikenal dengan Siklus Hidup Pengembangan Sistem. SDLC adalah proses mengembangkan atau mengubah suatu sistem *software* dengan menggunakan model-model dan metodologi yang digunakan untuk mengembangkan sistem-sistem *software* sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik). SDLC ini juga memiliki model-model yang digunakan dalam proses pengembangan sistem. Dalam penelitian ini, akan digunakan model *waterfall*.

Model SDLC *waterfall* (air terjun) sering juga disebut dengan model sekuensial linier atau alur hidup klasik. Model ini menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung. Menurut Pressman dalam Jurnal Permadi Setiawan, dalam model *waterfall* tersirat pendekatan yang sistematis dan berurutan pada pengembangan sistem. Pengembangan sistem dimulai dari penentuan spesifikasi kebutuhan dan berlanjut melalui tahapan-tahapan perencanaan (*planning*), pemodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem

ke para pelanggan atau pengguna (*deployment*), yang diakhiri dengan dukungan berkelanjutan pada sistem yang dihasilkan[19].



Gambar 3.1. Model *Waterfall*

Gambar diatas menunjukkan tahapan umum dari model proses *waterfall*. Model ini disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Akan tetapi, Pressman memecah model ini meskipun secara garis besar sama dengan tahapan-tahapan model *waterfall* pada umumnya. Model ini merupakan model yang paling banyak dipakai dalam *Software Engineering*. Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap *Communication*, *Planning*, *Modeling*, *Construction*, dan *Deployment*. Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam Model Waterfall menurut Pressman[20]:

a. *Communication*

Tahapan pertama ini merupakan tahapan yang penting, karena pada tahapan ini akan dikumpulkan data – data mengenai kebutuhan pengguna serta mendefinisikan spesifikasi sistem secara rinci.

b. *Planning*

Setelah proses *communication* dilakukan, tahapan selanjutnya adalah merencanakan pengerjaan sistem yang meliputi tugas-tugas teknis yang akan dilakukan, resiko yang mungkin terjadi, sumber daya dan data yang dibutuhkan, jadwal pengerjaan serta hasil yang akan dibuat.

c. *Modeling*

Pada tahap ini, rencana yang telah ditetapkan sebelumnya diterjemahkan ke sebuah perancangan sistem sebelum proses pembuatannya. Proses ini berfokus pada perancangan struktur data, arsitektur sistem yang akan dibuat, tampilan (*interfaces*), serta detail prosedur yang ada dalam sistem.

d. *Construction*

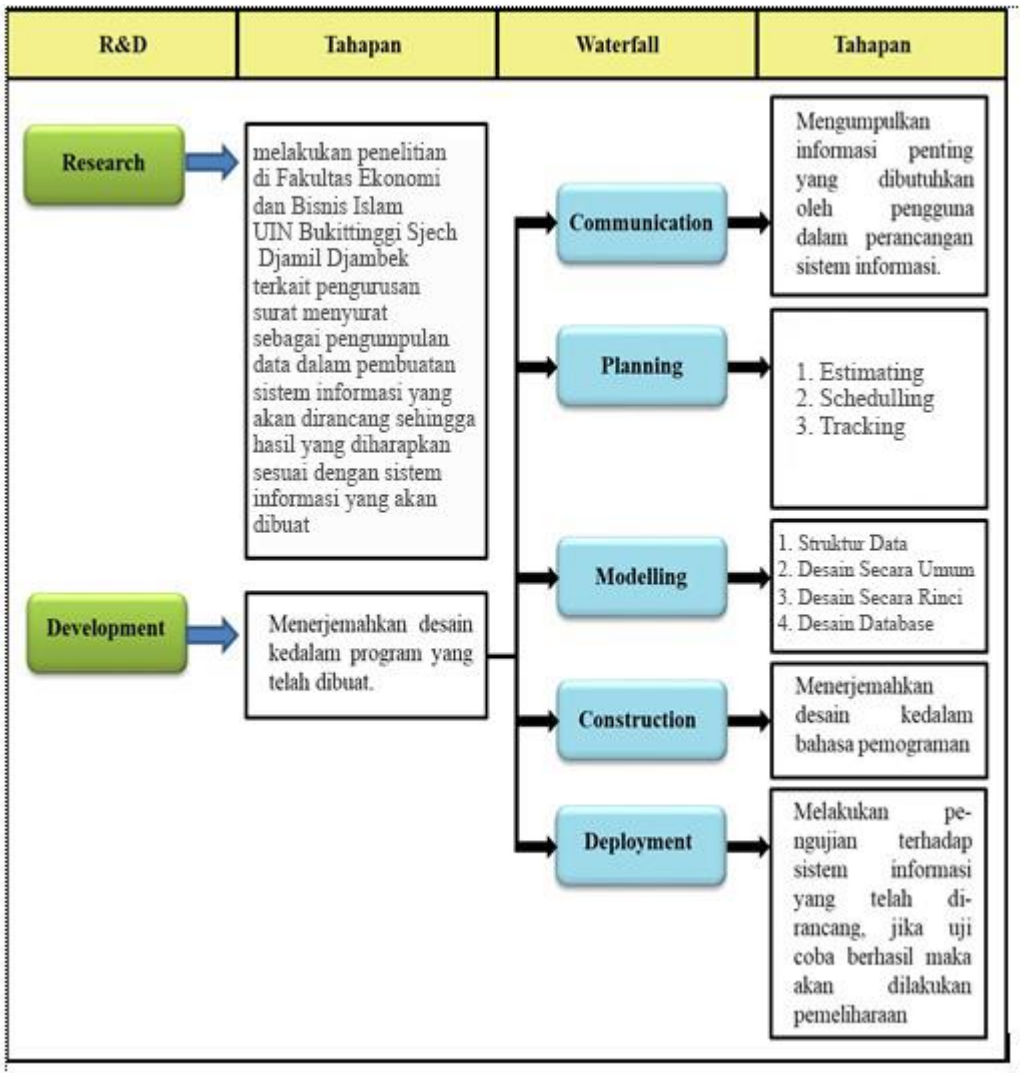
Dalam tahap *construction*, proses *coding* akan dilakukan. Tahap ini merupakan tahapan nyata dalam mengerjakan suatu sistem. Artinya, penggunaan komputer akan dimaksimalkan dalam tahap ini. Setelah proses *coding* selesai, maka akan dilakukan uji coba untuk menemukan kesalahan (*bug*) terhadap sistem tersebut untuk kemudian bisa diperbaiki.

e. *Deployment*

Deployment merupakan tahapan akhir (final) dalam proses pengembangan sistem. Setelah proses pembuatan dan analisis terhadap sistem selesai, maka selanjutnya akan disalurkan dan digunakan oleh pengguna. Sistem yang telah dibuat akan dilakukan pemeliharaan (*maintenance*) secara berkala sesuai dengan kebutuhan.

D. Tahapan Penelitian

Dalam melaksanakan penelitian, tahapan-tahapan yang dilakukan penulis sesuaikan dengan model penelitian tipe ADDIE serta model pengembangan sistem tipe *waterfall* seperti yang telah dijelaskan sebelumnya. Tahapan tersebut diantaranya:



Gambar 3.2 Tahapan Penelitian

E. Uji Produk

Uji produk merupakan langkah-langkah yang diterapkan untuk menguji suatu produk yang dihasilkan dengan tujuan menilai apakah produk tersebut sesuai dengan standar yang telah ditetapkan. Uji produk terdiri dari uji validitas, uji praktikalitas, dan uji efektivitas.

1. Uji Validitas

Aspek yang pertama kali diuji untuk menentukan kualitas dari suatu produk adalah validitas. Uji validitas dilakukan untuk menentukan apakah suatu produk yang dihasilkan dapat berjalan sesuai dengan fungsi dan tujuan yang telah ditetapkan di awal. Dalam artian, uji validitas ini dimaksudkan untuk menilai apakah suatu alat ukur dapat mengukur yang seharusnya diukur.

Uji validitas mengacu pada rumus Statistik Aiken's V, yaitu[21]:

$$\sum \frac{s}{[n(c-1)]}$$

Keterangan:

s: r - lo

lo: Angka penilaian validitas yang terendah

c: Angka penilaian validitas yang tertinggi

r: Angka yang diberikan oleh seorang penilai

n: Jumlah penilai

Selain itu, juga terdapat kategori dalam penentuan validitas dari formula tersebut. Nilai validitas "V" yang diperoleh dari formula berada pada rentang angka 0,00 sampai 1,00. Kategori penentuan validitas dari formula Aiken menyatakan bahwa suatu produk valid apabila berada pada rentang angka 0,60 sampai 1,00 dan akan tidak valid apabila berada di bawah angka 0,60.

2. Uji Praktikalitas

Apabila sebuah produk bersifat praktis dan mudah dalam pengadministrasiannya maka dikatakan bahwa produk tersebut memiliki praktikalitas tinggi. Sebaliknya, produk yang rumit dan sukar pengadmisitrasiannya dikatakan sebagai produk yang praktikalitasnya rendah.

Kepraktisan mengacu pada tingkat bahwa pengguna mempertimbangkan intervensi dapat digunakan dan disukai dalam kondisi normal. Dalam kerja berkaitan dengan pengembangan materi pembelajaran, dapat disinyalir bahwa mengukur tingkat kepraktisan dilihat dari pertimbangan apakah aplikasi mudah dan dapat digunakan oleh pegawai T.U.

Boslaugh Instrument penelitian berupa angket yang terdiri dari lembar validitas dan lembar praktikalitas. Data yang diperoleh dianalisis menggunakan momen kappa Boslaugh.

Dengan mangacu rumus Statistik Momen Kappa, yaitu sebagai berikut:

$$K = \frac{\rho - \rho e}{1 - \rho e}$$

Keterangan:

K : Moment kappa yang menunjukkan efektifitas produk.

P : Proporsi yang terealisasi, dihitung dengan cara jumlah nilai yang diberi oleh penguji dibagi jumlah nilai maksimal.

ρe : Proporsi yang tidak terealisasi, dihitung dengan cara jumlah nilai maksimal dikurangi dengan jumlah nilai total yang diberi penguji dengan jumlah nilai maksimal[22].

Tabel 3.4 Kategori Keputusan Berdasarkan Momen Kappa (k)

Interval	Kategori
0,81 – 1,00	Sangat tinggi
0,61 – 0,80	Tinggi
0,41 – 0,60	Sedang
0,21 – 0,40	Rendah
0,01 – 0,20	Sangat rendah
0,00	Tidak efektif

(sumber Boslaugh, 2008)

3. Uji Efektivitas

Uji efektivitas dimaksudkan untuk melihat seberapa efektif sistem dalam menyelesaikan masalah yang ada. Suatu sistem dapat dikatakan efektif apabila dapat memenuhi semua tujuan dan fungsi yang telah ditetapkan dengan baik dan benar. Rumus yang digunakan dalam

penentuan efektivitas ini mengacu pada rumus statistik Richard R. Hake (G-Score), yaitu[21]:

$$G = \frac{Sf - Si}{100\% - Si}$$

Keterangan:

G: G-Score

Sf: Score akhir

Si: Score awal

Kategori dari uji efektivitas ini dijelaskan dalam tabel 3.2.

Persentase (%)	**Kriteria**
< 29	Kurang efektif
30 – 69	Cukup efektif
70 – 100	Sangat efektif

Tabel 3.5 Kategori uji efektivitas

Sumber: Dimodifikasi dari Richard R. Hake. *Analyzing Change/Gain Scores*. (Jurnal American Educational Research Association's Division D, 1999), hal. 1

BAB IV
HASIL DAN PEMBAHASAN

A. Hasil Penelitian

Tahapan ini merupakan tahapan bagi penulis dalam menjabarkan proses-proses yang dilakukan dalam membuat produk. Tahapan ini sesuai dengan model penelitian tipe ADDIE yang digunakan serta dipadukan dengan model pengembangan sistem tipe *waterfall* (air terjun). Proses dimulai dari *analyze*, *design*, *develop*, *implement*, dan *evaluate*. Dalam proses *develop*, akan dipadukan dengan proses dari *waterfall* yakni, *communication*, *planning*, *modeling*, *construction* dan *deployment*.

Hasil penelitian yang didapatkan dari sistem informasi administrasi pengajuan dan penerbitan surat (E-LETTER) Fakultas Ekonomi dan Bisnis Islam berbasis php *mysqlI* adalah sebuah *system* informasi yang dapat difungsikan oleh pengguna atau fakultas dalam mengelola proses surat menyurat yang ada. dan memberikan kemudahan kepada mahasiswa dalam proses pengajuan dan mendapatkan surat yang dibutuhkan, karena sistem ini bersifat online dan bisa diakases pada *smarthpone*.

B. Pembahasan

1. *Communication*

a. *Project innitation*

Sebelum melakukan perancangan sistem, penulis melakukan wawancara di Fakultas Ekonomi dan Bisnis Islam dan penulis mendapatkan beberapa masalah seperti pengurusan surat masih

dikerjakan secara manual atau semi-otomatis. Jumlah surat yang diterbitkan masih terbatas. Terjadi antrian panjang ketika mahasiswa hendak mengambil surat di fakultas. Mahasiswa harus datang ke fakultas untuk memastikan surat sudah diterbitkan atau belum. Belum adanya sistem informasi khusus yang menganani masalah administrasi pengajuan dan penerbitan surat di Fakultas Ekonomi dan Bisnis Islam. Belum adanya sistem informasi untuk pengarsipan elektronik.

Dengan adanya permasalahan – permasalahan yang dijelaskan seperti diatas, penulis menyimpulkan untuk membuat sistem administrasi khusus untuk membantu mempermudah pengajuan dan penerbitan bagi mahasiswa di Fakultas Ekonomi dan Bisnis Islam.

b. *Requirements Ghatering*

a. Kebutuhan *User*

Kebutuhan user dari sistem informasi yang dirancang terdiri dari :

1. Super Admin

Super admin di Fakultas Ekonomi dan Bisnis Islam adalah tenaga IT atau operator yang bertugas menjalankan sistem informasi. Sehingga sistem dapat berjalan dengan baik. Adapun proses yang dapat dilakukan admin adalah mengelola data admin (Staff/TU) dan mengelola data

mahasiswa, beberapa kebutuhan user ini dijabarkan sebagai berikut :

a) Membutuhkan fitur Create, Read, Update, Delete (CRUD) untuk mengelola user ditingkat admin dan user mahasiswa.
b) Membutuhkan fitur download dan upload data data jumlah besar yang nanti nya akan di masukkan kedalam database.
c) Membutuhkan fitur export import data dalam bentuk excell untuk rekapan data membuat laporan.

2. Admin

Admin pada sistem ini adalah TU atau staf yang ada pada di FEBI. Adapun kebutuhan yang diperlukan oleh admin adalah adanya sistem informasi yang bisa membantu mempermudah dalam hal membuat surat yang telah diajukan dan menginformasikan surat yang telah dibuat, user tingkat ini nanti nya akan melakukan proses peninjauan surat dan syarat yang dibutuhkan melalui sistem, jika surat permohonan dan syarat sudah lengkap surat kemudian di disposisikan dan di validasi, setelah itu admin mengupload kembali surat yang diajukan oleh user biasa/mahasiswa, surat yang dapat dikelola seperti SK pembimbing, aktif kuliah, izin penelitian, observasi, surat

praktek mata kuliah, rekomendasi beasiswa, surat penunjukan promotor, surat keterangan lulus, surat keterangan telah selesai penelitian, dan surat uji coba angket. Kebutuhan user admin dapat di jabarkan sebagai berikut :

a) Membutuhkan fitur untuk menampilkan status dari proses sebuah surat (diproses, ditolak, dan selesai).
b) Membutuhkan tombol Create, Read, Update, Delete (CRUD) untuk pengelolaan surat.
c) Membutuhkan fitur download dan upload surat yang telah selesai diproses.
d) Membutuhkan fitur untuk menampilkan laporan surat yang telah selesai di proses dengan pilihan laporan bulanan dan tahunan untuk rekapan data.

3. *User* biasa

User biasa adalah para mahasiswa dari Fakultas Ekonomi dan Bisnis Islam. Sistem infomasi ini dibuat bertujuan untuk mempermudah mahasiswa dalam pengajuan surat. Mahasiswa tidak perlu lagi melakukan antrian dalam pengambilan surat, dan efisiensi waktu untuk mahasiswa itu sendiri.

Pada sistem ini mahasiswa dapat melakukan pengajuan surat dan mendapatkan surat yang telah

diterbitkan. Surat yang dimaksud disini seperti SK pembimbing, aktif kuliah, izin penelitian, observasi, surat praktek mata kuliah, rekomendasi beasiswa, surat penunjukan promotor, surat keterangan lulus, surat keterangan telah selesai penelitian, dan surat uji coba angket. Kebutuhan mahasiswa ditingkat ini bisa dijabarkan sebagai berikut :

a) Membutuhkan fitur yang dapat membuat surat permohonan untuk pengajuan surat secara otomatis
b) Membutuhkan fitur *form/textarea/field* untuk pengisian data selain dari data mahasiswa (NIM, Nama, Fakultas).
c) Membutuhkan fitur download surat yang telah diproses dan upload file/berkas tambahan jika surat yang diajukan memiliki persyaratan berkas tambahan selain dari surat permohonan.
d) Membutuhkan fitur untuk memantau proses dari surat yang telah diajukan.

b. Kebutuhan Non Fungsional

Tahap ini untuk menjalankan sistem bagi super admin, admin, mahasiswa. aplikasi yang di rancang membutuhkan dua komponen teknologi informasi yaitu *hardware* dan *software.*

1) *Hardware*

Hardware yang digunakan untuk menjalankan E-LETTER bagi super admin, admin, dan mahasiswa adalah :

a) Laptop atau PC, untuk menjalan sistem administasi ini minimal mempunyai spesifikasi prosesor Inter Dual Core@ 2.0 Ghz, RAM 4 GB.

b) *Mouse*

c) *Keyboard*

d) *Monitor*

e) Media penyimpanan

f) *Smarthphone*, prosesor Octa Core, GPU Adreno 505, RAM 2GB

2) *Software*

Software yang digunakan untuk bisa menjalankan E-LETTER bagi super admin, admin, dan mahasiswa adalah :

a) Microsoft Windows XP

b) Android versi 6.0

c) Browser, *Goggle Chrome, Microsoft Edge, Internet Exploerer, Safari*

d) *PDF reader*

2. *Planning (Estimating, Schedulling, Tracking)*

a) *Estimating (*Perkiraan Tugas)

Tahapan ini menjelaskan mengenai bagian kerja yang dilakukan dalam membangun sistem penjadwalan kuliah ini, diantaranya:

1) Tahap pertama yang penulis lakukan adalah mengumpulkan data-data yang diperlukan untuk membangun sistem ini. Data tersebut berupa data jenis surat, prosedur pengajuan surat mahasiswa, persyaratan pengajuan surat untuk tiap jenis surat hingga surat diterbitkan. Data ini penulis dapatkan langsung dari kasubbag umum Ibuk Syafnidar, SE., M.Si dan juga kabag TU Ibuk Sasmiarti, S.Ag,. S.IP., MH.
2) Selanjutnya, penulis merancang logika-logika yang diperlukan pada sistem seperti logika input dan output serta logika pengajuan dan surat.
3) Menuangkan rancangan logika yang telah dibuat ke dalam bahasa pemrograman PHP menggunakan *framework Laravel*.
4) Membangun tampilan yang ditampilkan pada saat user mengakses sistem.
5) Melakukan uji coba terhadap sistem apakah sesuai dengan yang diharapkan.
6) Melakukan *maintenance* atau perbaikan terhadap beberapa kesalahan yang tidak terdeteksi saat membangun sistem.

b) *Schedulling*

Nama Proyek : Sistem Informasi Administrasi Pengajuan Dan Penerbitan Surat Fakultas Di Fakultas Ekonomi dan Bisnis Islam UIN Bukittinggi

Manager proyek : Haikal

Dibuat oleh : Haikal

Tanggal : September 2022

Proses membangun sistem dilakukan selama lebih kurang dua bulan. Dimulai pada bulan September hingga Oktober 2022. Berikut ini penulis jabarkan jadwal dalam bentuk tabel pada pembuatan sistem mulai dari pengumpulan data sampai ke perbaikan (*maintenance*). Berikut ini adalah jadwal dalam pembuatan system :

No	Proses	Bulan / Tanggal													
		September													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Pengumpulan data														
2	Merancang logika														

No	Proses	Bulan / Tanggal															
		September															
		15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
1	Menuangkan logika ke bahasa pemrograman																	

No	Proses	Bulan / Tanggal																		
		Oktober																		
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Menuangkan logika ke bahasa pemrograman																				

No	Proses	Bulan / Tanggal									
		April									
		21	22	23	24	25	26	27	28	29	30
1	Membangun tampilan										
2	Uji coba kasar										
3	Memperbaiki kesalahan										

a) Tahap pengumpulan data penulis lakukan pada awal bulan September selama satu minggu tepatnya pada minggu pertama dari tanggal 1 sampai tanggal 7 September 2022.

b) Tahapan merancang logika penulis lakukan selama seminggu mulai tanggal 8 sampai tanggal 14 September 2022.

c) Menuangkan logika ke bahasa pemrograman merupakan tahapan paling lama dikarenakan dalam proses *coding* harus mencari dan menyesuaikan dengan logika yang telah dibuat. Tahap ini dilakukan lebih kurang selama satu bulan lebih yang dimulai dari tanggal 15 September sampai dengan tanggal 20 Oktober 2022.

d) Proses membangun tampilan penulis lakukan selama 5 hari. Proses ini tergolong cepat karena mengandalkan *framework* Bootstrap sehingga hanya perlu menyesuaikan tampilan sesuai

dengan yang diinginkan. Proses ini dilakukan dari tanggal 21 sampai 25 Oktober 2022.

e) Uji coba secara kasa penulis lakukan selama 2 hari , tanggal 26 Oktober dan 27 Oktober 2022.

f) Tahapan terakhir dalam memperbaiki kesalahan penulis lakukan selama 3 hari mulai tanggal 28 sampai 30 Oktober 2022.

3) *Tracking* (Pelacakan)

Pelacakan yang penulis lakukan adalah dengan melihat dari beberapa referensi jurnal dengan judul yang memiliki kesamaan dengan penulis, sistem administrasi pengajuan dan penerbitan surat, perancangan sistem administrasi surat masuk dan keluar, dan beberapa lainnya. Selain itu, penulis juga mengambil referensi dari aplikasi yang sudah ada. Akan tetapi, dalam aplikasi tersebut masih terdapat beberapa kekurangan dimana terdapat perbedaan beberapa seperti alur atau prosedur pengajuan surat yang berbeda dengan studi kasus penulis. Sehingga dalam aplikasi tersebut masih harus dilakukan penyempurnaan. Pengerjaan sistem informasi ini, penulis melalui beberapa tahapan. Membuat aplikasi dengan menggunakan bahasa pemrograman laravel dan *framework front end* CSS *bootstrap* 4. Sistem informasi yang telah selesai akan di hosting melalui situs ....., dan penulis membuat URL dari sistem ini yaitu http//…….....com

3. Modelling

a. Struktur Data

Struktur data pada pembuatan E-LETTER adalah dengan menggunakan laravel dan *framework front end* CSS *bootstrap* 4. Adapun spesifikasi struktur datanya sebagai berikut :

1) Penyimpanan Data

Penyimpanan data pada sistem ini menggunakan *database*, dengan menggunakan file yang berisikan koneksi ke *database*, berikut adalah koding koneksi :

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=e_letter
DB_USERNAME=root
DB_PASSWORD=
```

2) Penyusunan Data

Penyusunan data pada *framework* laravel sudah menyediakan sistem dengan metode MVC (*Model, View, Controller*). Pada codeigniter ke tiga bagian tersebut dipisah kedalam 3 file yaiut data (*model*), tampilan (*view*), proses (*controller*). *View* berisikan koding tampilan dari sistem yang dibuat, *model* berisikan koding yang berhubungan dengan proses yang berkaitan dengan database yaitu CRUD (*Create,*

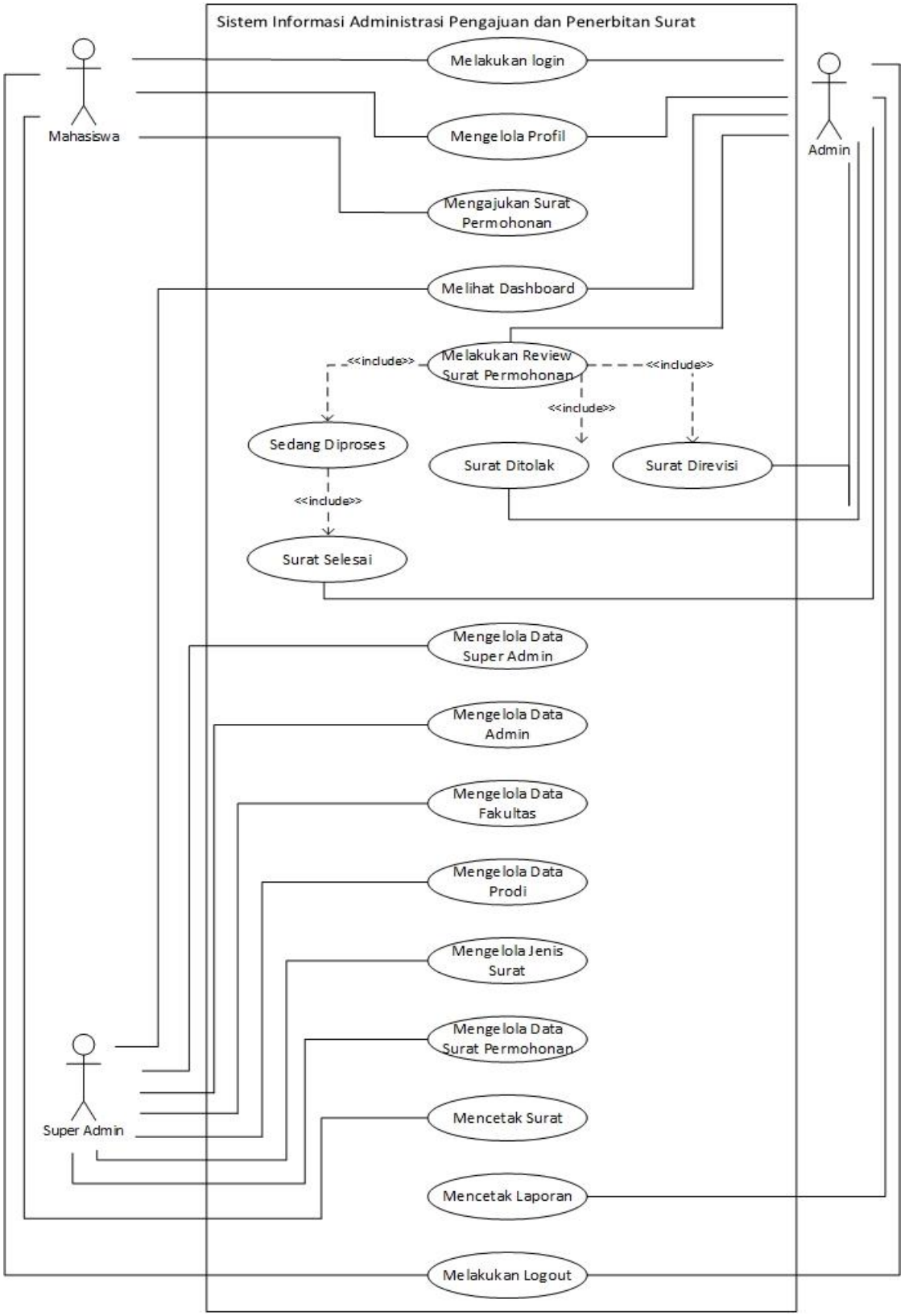
Read, Update, Delete), dan *controller* berisikan koding yang berfungsi untuk menjembatani antara *view* dan *model*.

b. Desain Secara Umum

Desain secara umum berisikan hubungan yang terjadi antara sistem dan pengguna, sistematika yang ada pada sistem, dan aktifitas yang pengguna lakukan terhadapa sistem yang penulis buat. Hal-hal tersebut digambarkan melalui *use case diagram, activity diagram, sequence diagram*, dan *class diagram.*

1) *Use Case Diagram*

Use case diagram yang terdapat pada E-LETTER menggambarkan fungsionalitas yang di harapkan dari sebuah sistem, diagram yang menggambarkan interaksi antara sistem dengan sistem eksternal dan pengguna. Berikut ini merupakan *use case* diagram dari E-LETTER.



Gambar 3.3 *Use case* E-Letter

Tabel 4.1 ***Actor Use Case***

NO	AKTOR	DESKRIPSI
1	Mahasiswa	Mahasiswa melakukan login, setelah login mahasiswa dapat melakukan pengajuan surat permohonan dengan mengisi form pada menu permohonan surat dan melampirkan dokumen tertentu jika dibutuhkan. Setelah diajukan mahasiswa dapat memantau proses surat yang telah diajukan pada menu surat permohonan. Setelah surat di review dan dirilis kembali, mahasiswa dapat mengunduh surat yang telah disetujui oleh fakultas.
2	Admin	Admin melakukan login, kemudian memantau surat yang masuk pada menu surat permohonan, kemudian melakukan review surat permohonan yang masuk, apabila surat sudah memenuhi syarat surat diproses, di disposisikan sampai disetujui kemudian diupload kembali ke sistem agar dapat di download oleh mahasiswa, jika surat yang diajukan belum memenuhi syarat, surat akan di tolak dan diberi catatan khusus. Admin juga dapat mencetak laporan surat, dan melakukan rekap data surat.
3	Super Admin	Super Admin melakukan login, super admin dapat melakukan penambahan user untuk tiap level user, mengelola fakultas, program studi, surat, surat permohonan, dan laporan.

Tabel 4.2 ***Narative Use Case***

NO	USE CASE	DESKRIPSI
1	Melakukan Login	*User* melakukan login ke dalam sistem dengan id dan password yang sesuai agar dapat menggunakan fitur yang ada dalam sistem sesuai dengan level user
2	Mengelola profil	*User* dapat mengelola profil menggunakan menu ubah profil lalu menyimpan kembali dengan mengklik menu profil.
3	Melihat dashboard	Super Admin dan Admin dapat memantau surat yang menunggu Direview, menunggu Direvisi, Selesai/Dirilis, dan Ditolak.

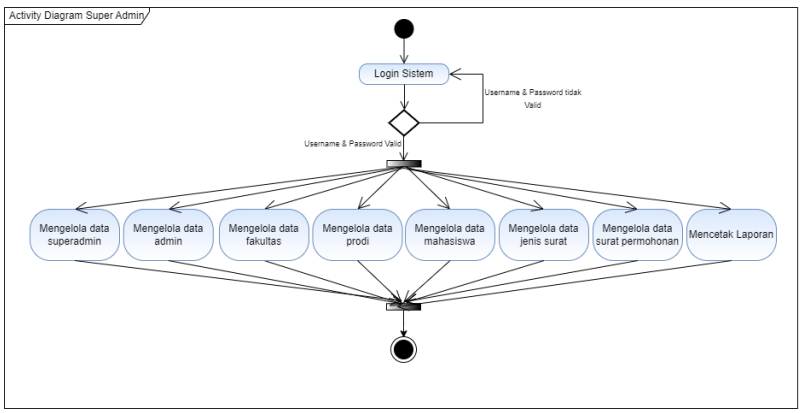
A	B	C
4	Mengajukan Surat Permohonan	Mahasiswa melakukan pengajuan surat permohonan, dengan mengisi form dan melampirkan dokumen yang dibutuhkan, yang kemudian di review oleh Admin/Staff.
5	Mengelola Data Super Admin	Super Admin dapat mengelola data super admin seperti menambah, mengedit, dan menghapus data super admin pada menu super admin
6	Mengelola Data Admin	Super Admin dapat mengelola data admin seperti menambah, mengedit, dan menghapus data user pada menu super admin
7	Mengelola Data Mahasiswa	Super Admin dapat mengelola data mahasiswa seperti menambah, mengedit, dan menghapus data mahasiswa pada menu super admin
8	Mengelola Data Fakultas	Super Admin dapat mengelola data fakultas seperti menambah, mengedit, dan menghapus data fakultas pada menu super admin
9	Mengelola Data Prodi	Super Admin dapat mengelola data program studi seperti menambah, mengedit, dan menghapus data program studi pada menu super admin
10	Mengelola Data Surat	Super Admin dapat mengelola data surat seperti menambah, mengedit, dan menghapus jenis surat pada menu super admin
11	Melakukan Review Surat	Admin melakukan review terhadap surat yang telah diajukan oleh mahasiswa, apabila surat di tolak akan diupdate status nya “Ditolak” pada halaman permohonan surat dengan catatan khusus, apabila di terima, status surat akan menjadi “Sedang Diproses” dan diproses hingga surat di rilis dan diupload kembali ke sistem oleh admin, status surat akan berubah menjadi “Selesai”
12	Surat Di Proses	Surat yang telah direview dan telah lengkap dengan persyarantannya akan langsung diproses, status akan berubah menjadi “Sedang Diproses”
13	Surat Selesai	Surat yang telah selesai di proses akan di upload oleh Admin/Staff TU kemudian status akan berubah menjadi “Selesai”

A	B	C
14	Surat Direvisi	Surat yang belum lengkap data dan persyaratannya akan ditolak dengan memberikan catatan khusus, status akan berubah menjadi "Menunggu Revisi"
	Surat Ditolak	Surat yang belum lengkap data dan persyaratannya akan ditolak dengan memberikan catatan khusus, status akan berubah menjadi "Ditolak"
12	Mencetak Surat	Mahasiswa mengunduh surat yang telah dirilis atau telah selesai di proses oleh Admin/Staff TU pada faultas yang di upload dengan format PDF
13	Mencetak Laporan	Admin mencetak laporan dengan beberapa pilihan dan periode atau rentang waktu tertentu yang tersedia pada sistem dengan format PDF
14	Melakukan Logout	User melakukan logout untuk keluar dari sistem

2) *Activity Diagram*

Activity diagram menggambarkan perilaku dinamis dari suatu sistem atau suatu komponen sistem melalui aliran kontrol di antara aksi-aksi yang dilakukan oleh sistem. Berikut *Activity diagram* dari E-LETTER.

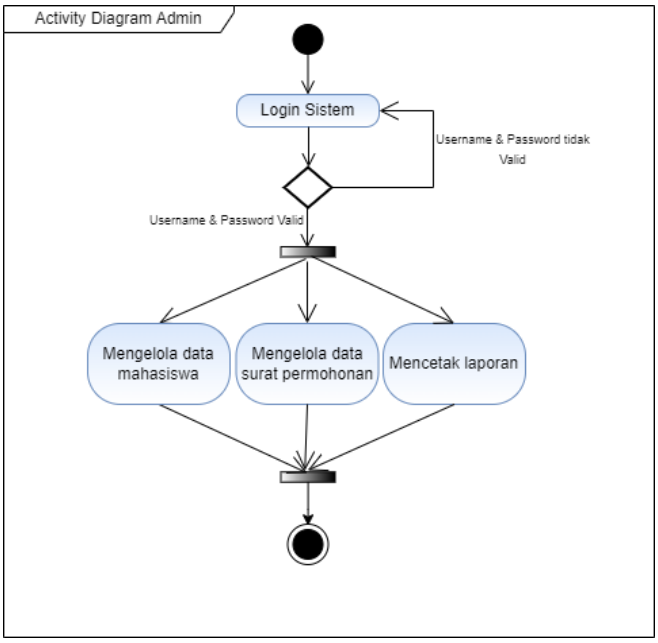
a) *Activity Diagram* Superadmin



Gambar 3.4 *Activity* Diagram Super Admin

Berdasarkan *activity diagram* super admin pada gambar 4.2 dapat kita lihat aktifitas yang dapat dilakukan super admin. Jika *user* login sebagai admin makan kegiatan yang dilakuannya yaitu mengelola data super admin, mengelola data admin, mengelola data fakultas, mengelola data prodi, mengelola data mahasiswa, mengelola data jenis surat, mengelola data surat permohonan, mencetak laporan, dan melakukan logout.

b) *Activity Diagram* Admin

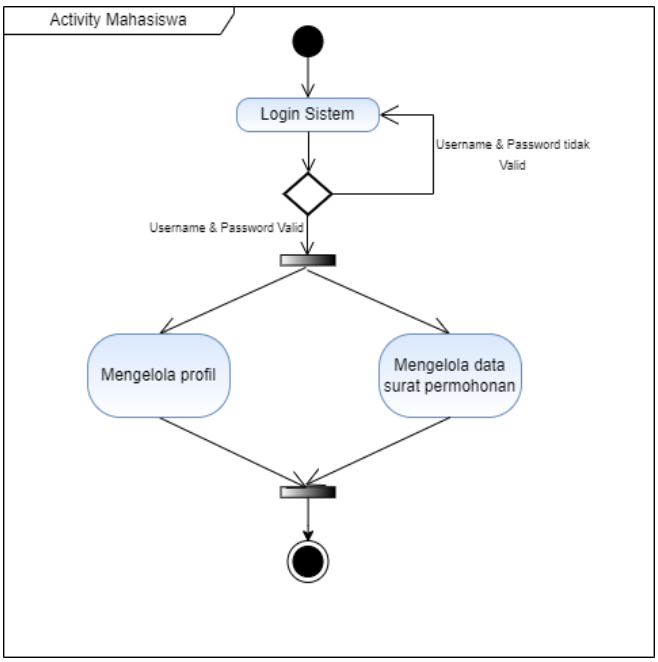


Gambar 3.5 *Activity Diagram* admin

Berdasarkan *activity diagram* admin pada gambar 4.3 dapat kita lihat aktifitas yang dapat dilakukan admin. Jika *user* login sebagai admin makan kegiatan yang

dilakuannya yaitu mengelola data mahasiswa, mengelola data surat permohonan, dan mencetak laporan.

c) *Activity Diagram* Mahasiswa



Gambar 3.6 *Activity Diagram* Mahasiswa

Berdasarkan *activity diagram* mahasiswa pada gambar 3.6 dapat kita lihat aktifitas yang dapat dilakukan mahasiswa. Jika *user* login sebagai mahasiswa maka kegiatan yang dilakuannya yaitu mengelola data profil dan mengelola data surat permohonan.

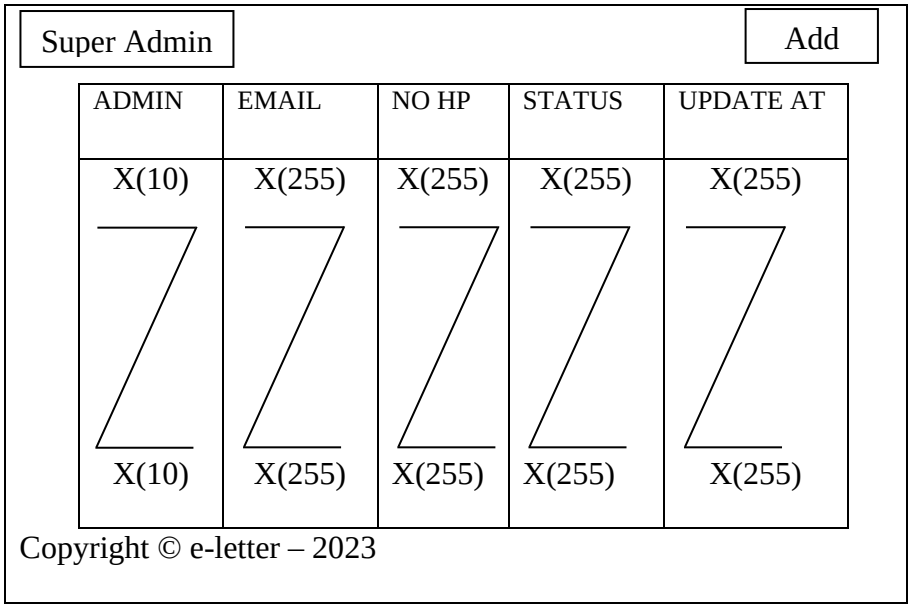
c. Desain Secara Rinci

1) Desain Output

a) Desain *output* data super admin.

Output data super admin akan menampilkan daftar admin yang ada pada sistem. Desain tampilan data admin sebagai berikut :

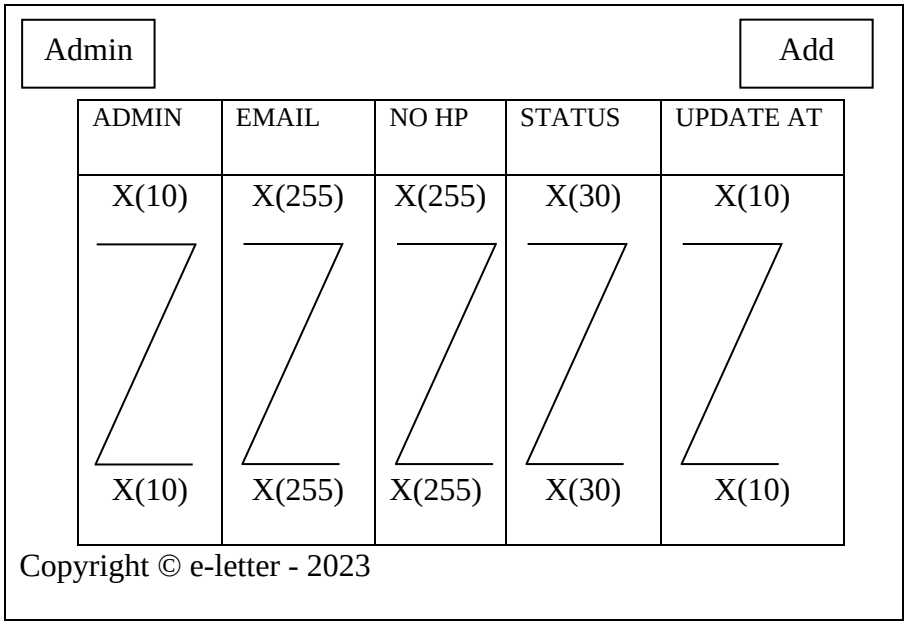
Gambar 3.7 Desain *output* data super admin



b) Desain *output* data admin.

Output data admin akan menampilkan daftar admin yang ada pada sistem. Desain tampilan data admin sebagai berikut :

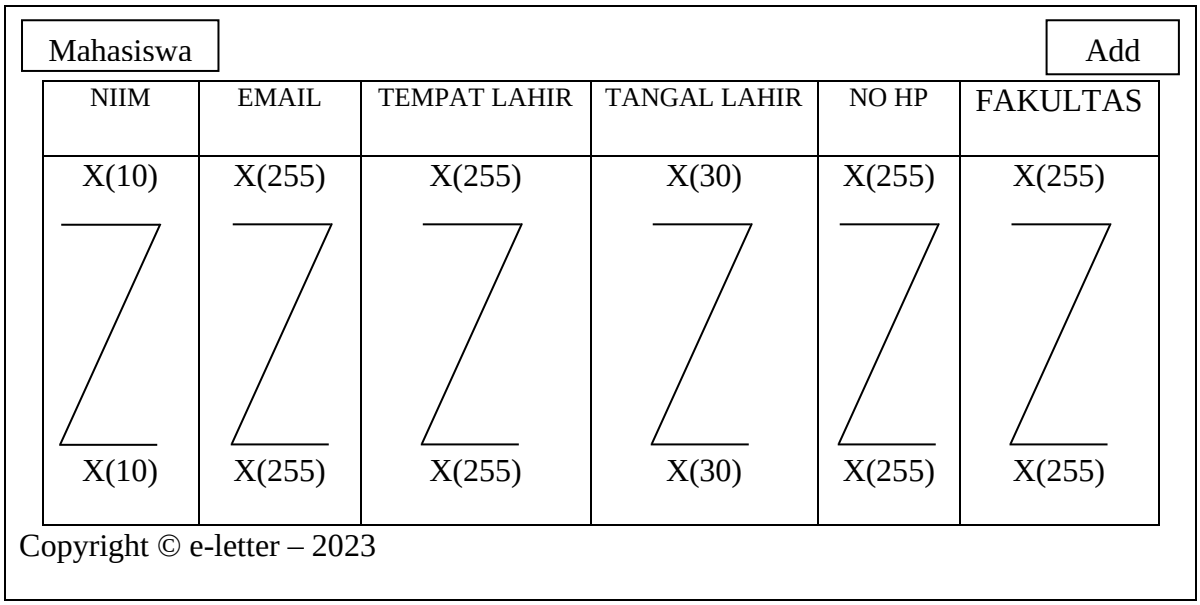
Gambar 3.8 Desain *output* data admin



c) Desain *output* data mahasiswa.

Output data super admin akan menampilkan daftar admin yang ada pada sistem. Desain tampilan data mahasiswa berikut :

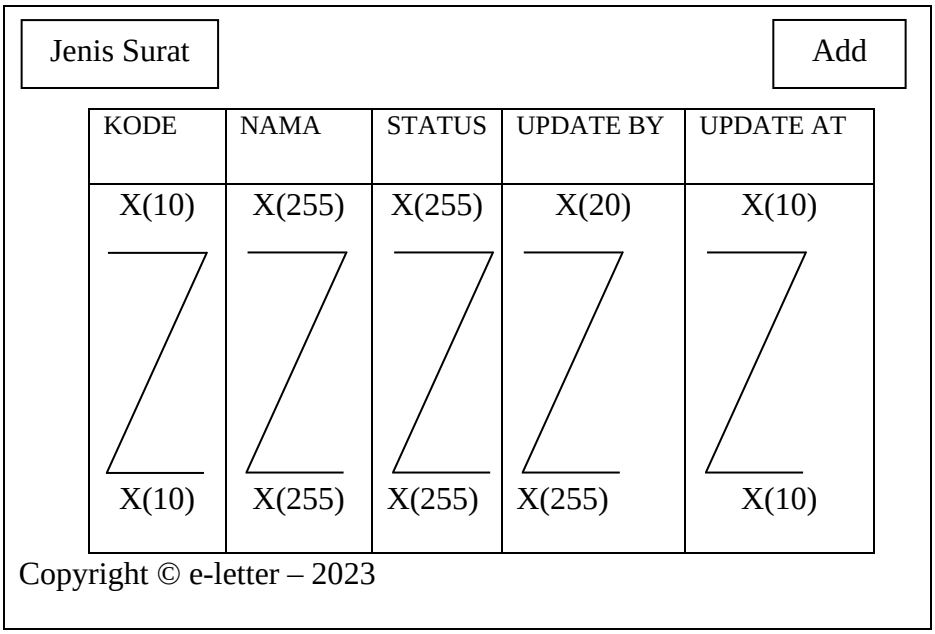
Gambar 3.9 Desain *output* data mahasiswa



d) Desain *output* data surat.

Output data surat akan menampilkan daftar admin yang ada pada sistem. Desain tampilan data surat berikut :

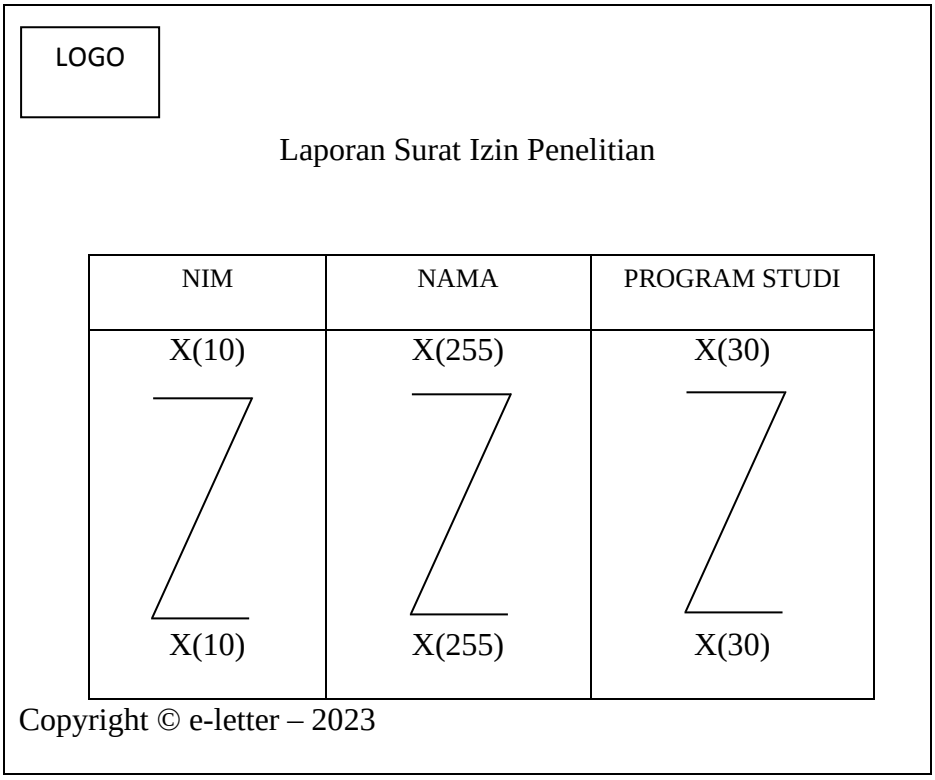
Gambar 4.0 Desain *output* data surat



e) Desain *output* surat keluar

Output surat keluar akan menampilkan desain surat keluar dari sistem ini. Desain tampilan output surat keluar berikut :

Gambar 4.1 Desain *output* cetak laporan

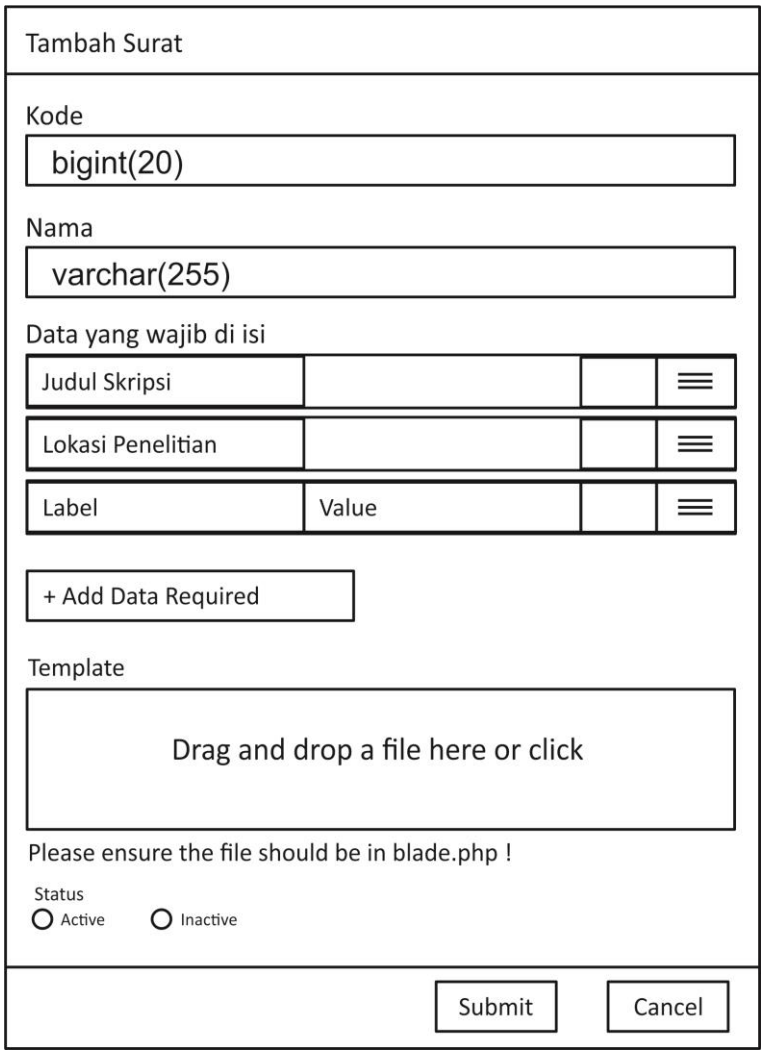


2) Desain Input

Desain *Input* (masukan) merupakan tampilan sebagai data pendukung yang nanti nya dimasukkan de dalam sistem.

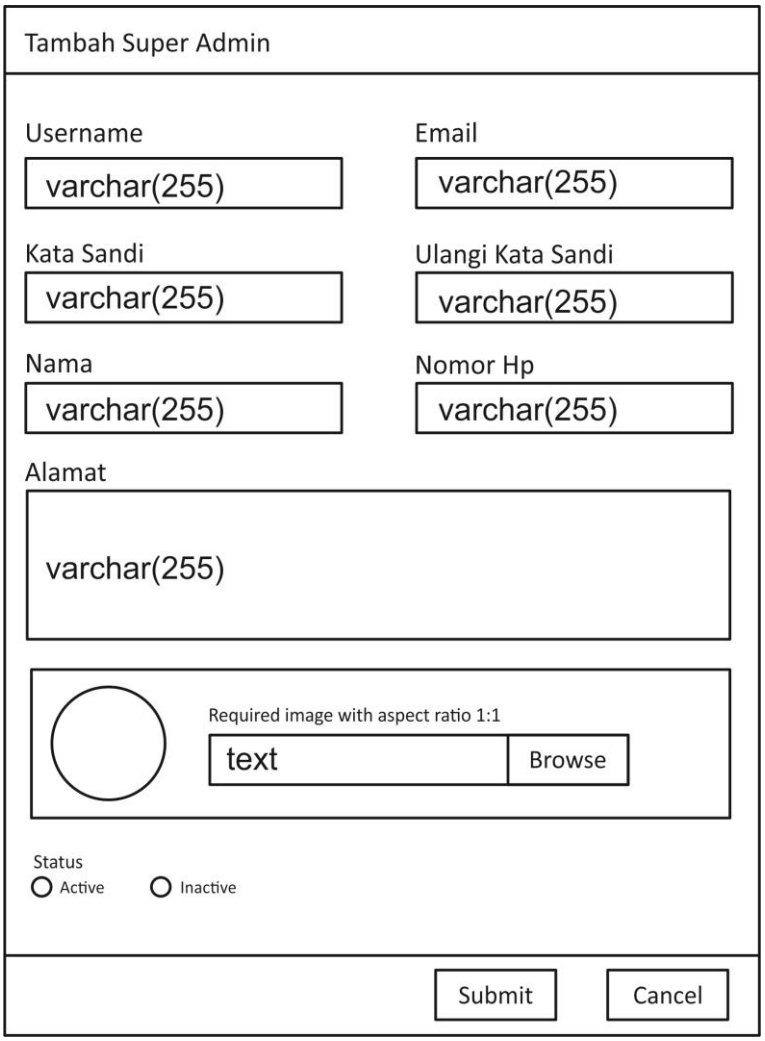
Hasil input nantinya akan diolah menjadi sebuah *output* (keluaran) berupan informasi yang dapat dipahami oleh *user*. Berikut adalah desain input yang digunakan dalam sistem :

a) Desain Input Surat



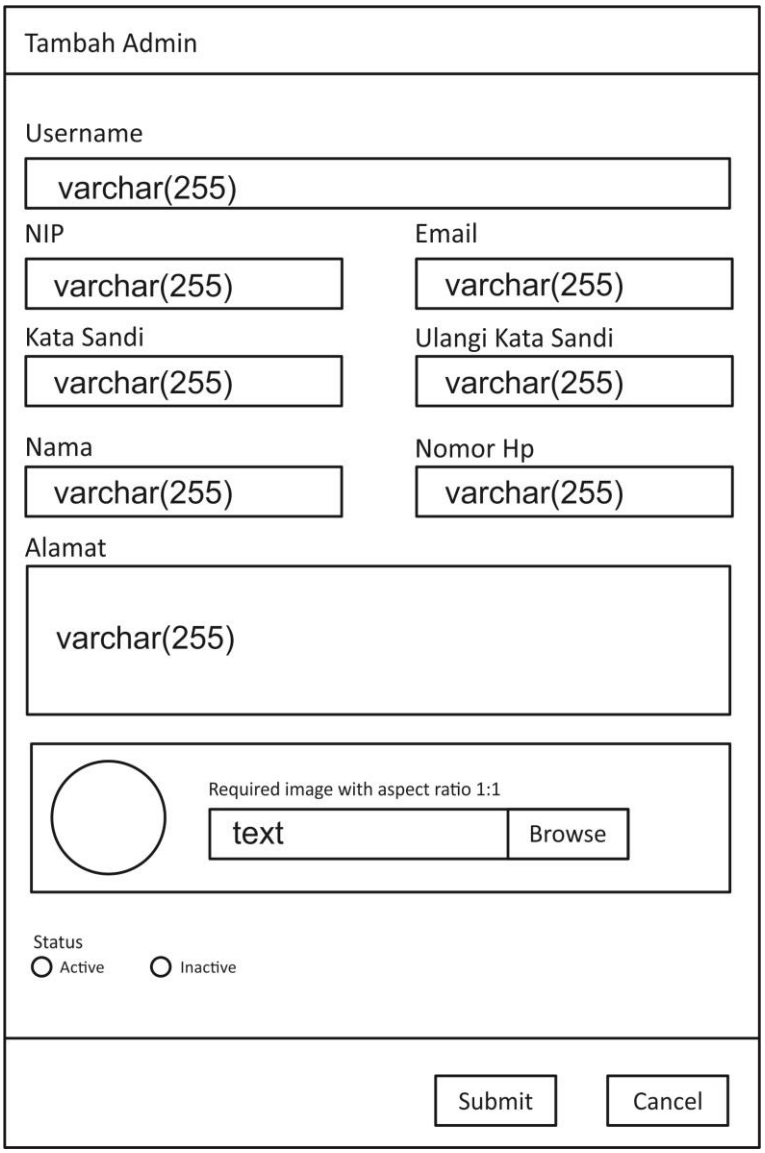
Gambar 4.2 Desain *Input* Surat.

b) Desain Input Super Admin



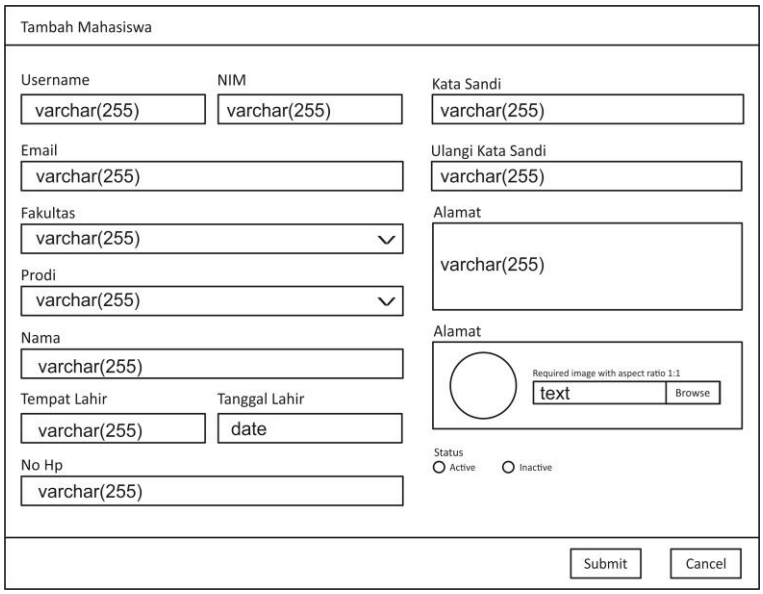
Gambar 4.3 Desain *Input* Super Admin

c) Desain Input Admin



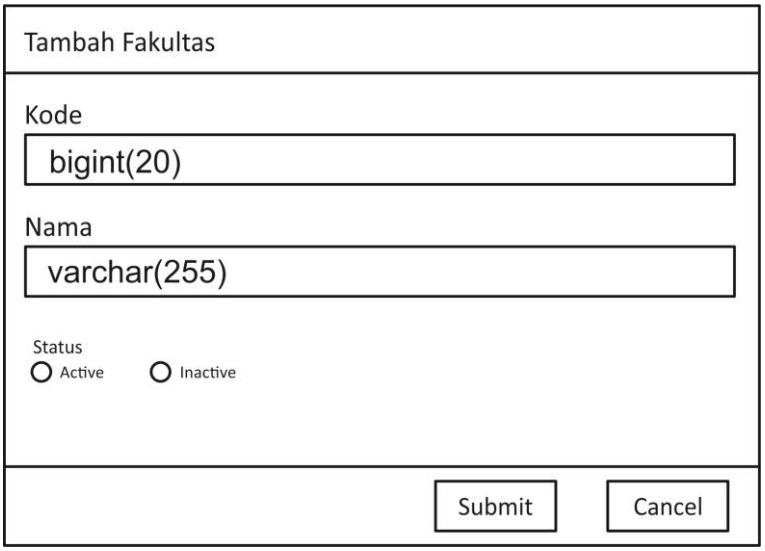
Gambar 4.4 Desain *Input* Admin

d) Desain Input Mahasiswa



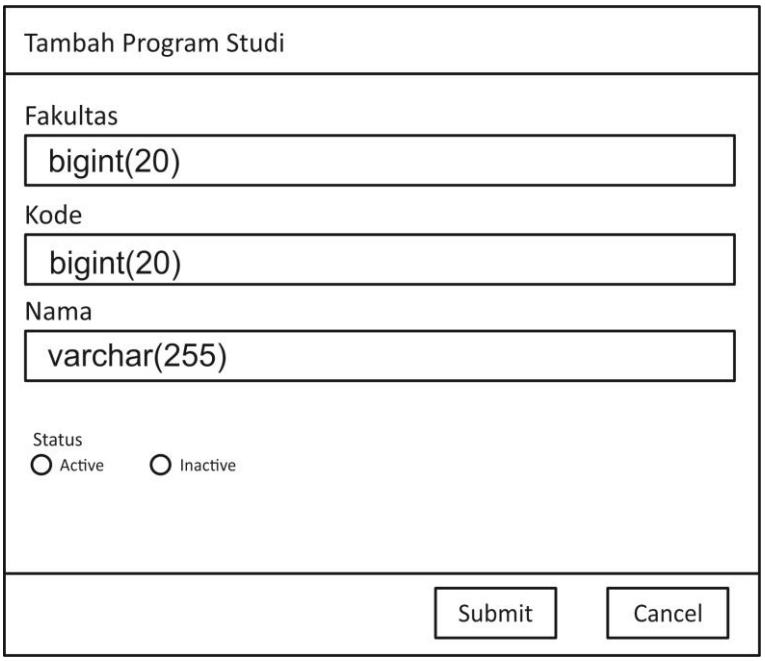
Gambar 4.5 Desain *Input* Mahasiswa

e) Desain Input Fakultas



Gambar 4.6 Desain *Input* Fakultas

f) Desain Input Prodi



Gambar 4.7 Desain *Input* Program Studi

d. Desain Database

1) Tabel Program Studi

Desain tabel program studi ini merupakan tabel yang berisi data atribut program studi dalam *database*.

Tabel 4.7 Tabel Program Studi

Nama *database* : e_letter

Nama tabel : t_program_study

Field key : Id

No	Field Name	Tipe	Width	Keterangan
1	id	bigint	20	ID user

A	B	C	D	E
2	auth_id	bigint	20	Autentifikasi id login untuk menetukan level user
3	faculty_id	bigint	20	Id fakultas
4	code	varchar	255	Kode program studi
5	name	varchar	255	Nama program studi
6	remember_token	varchar	100	
7	state	smallint	6	status
8	created_at	timestamp		
9	updated_at	timestamp		

2) Tabel Super Admin

Desain tabel ini merupakan tabel yang berisi data atribut dari super admin dalam *database*.

Tabel 4.8 Tabel Super Admin

Nama *database* : e_letter

Nama tabel : t_user_super_admin

Field key : id

No	Field Name	Tipe	Width	Keterangan
1	Id	bigint	20	ID user
2	Email	bigint	20	Email user
3	name	varchar	255	Nama user
4	No_hp	varchar	255	No hp user
5	address	text		Alamat user
6	photos	text		Foto user
7	Email_verified_at	timestamp		Verifikasi email user
8	created_at	timestamp		
9	updated_at	timestamp		

3) Tabel Admin

Desain tabel ini merupakan tabel yang berisi data atribut dari super admin dalam *database*.

Tabel 4.9 Tabel User Admin

Nama *database* : e_letter

Nama tabel : t_user_admin

Field key : id

No	Field Name	Tipe	Width	Keterangan
1	id	bigint	20	ID user
2	email	bigint	20	Email user

A	B	C	D	E
3	name	varchar	255	Nama user
4	No_hp	varchar	255	No hp user
5	address	text		Alamat user
6	photos	text		Foto user
7	Email_verified_at	timestamp		Verifikasi email user
8	created_at	timestamp		
9	updated_at	timestamp		

4) Tabel Mahasiswa

Desain tabel ini merupakan tabel yang berisi data atribut dari super admin dalam *database*.

Tabel 4.10 Tabel User Mahasiswa

Nama *database* : e_letter

Nama tabel : t_user_student

Field key : id

No	Field Name	Tipe	Width	Keterangan
1	id	bigint	20	ID user
2	auth_id	bigint	20	id login untuk menetukan level user

A	B	C	D	E
3	program_study_id	bigint	20	Id program studi
4	nim	varchar	255	
5	email	varchar	255	Email user
6	name	varchar	255	Nama user
7	no_hp	varchar	255	No hp user
8	date_of_birth	date		
9	place_of_birth	varchar	255	
10	address	varchar	255	Alamat user
11	photos	text		Foto user
13	created_at	timestamp		
14	updated_at	timestamp		

5) Tabel Surat

Desain tabel ini merupakan tabel yang berisi data atribut dari surat dalam *database*.

Tabel 4.11 Tabel Surat

Nama *database* : e_letter

Nama tabel : t_letter

Field key : Id

No	Field Name	Tipe	Width	Keterangan
1	id	bigint	20	Id surat
2	owner_id	bigint	20	Id mahasiswa
3	editor_id	bigint	20	Id editor/admin
4	type_id	bigint	20	
5	student_id	bigint	20	Id mahasiswa
6	submitted_date	datetime		Tanggal di ajukan
7	data	Text		
8	comment	Text		
9	state	smallint	6	
10	created_at	timestamp		
11	updated_at	timestamp		

4. *Construction*

a) *Coding*

1) Routes

```
<?php
use Illuminate\Support\Facades\Route;

Route::get('/', function () {
  return view('welcome');
});
```

```
Route::middleware(['session'])-
>prefix('/{locale}/auth')->name('auth.')-
>group(function () {
  Route::get('/sign-in', 'Pages\Auth@sign_in')-
>name('sign_in');
  Route::get('/sign-up', 'Pages\Auth@sign_up')-
>name('sign_up');
  Route::middleware(['token.web'])->get('/sign-
out', 'Pages\Auth@sign_out')->name('sign_out');
  Route::get('/forgot-password',
'Pages\Auth@forgot_password')-
>name('forgot_password');
  Route::get('/reset-password',
'Pages\Auth@reset_password')-
>name('reset_password');

  Route::post('/sign-in',
'AuthController@sign_in')->name('sign_in');
  Route::post('/sign-up',
'AuthController@sign_up')->name('sign_up');
});
Route::middleware(['language', 'session',
'token.web', 'superadmin'])-
>prefix('/{locale}/manage')->name('superadmin.')-
>group(function () {
  Route::get('/dashboard',
'Pages\SuperAdmin@dashboard')->name('dashboard');
  Route::get('/faculty',
'Pages\SuperAdmin@faculty')->name('faculty');
  Route::get('/program-study',
'Pages\SuperAdmin@program_study')-
>name('program_study');
  Route::get('/superadmin',
'Pages\SuperAdmin@superadmin')->name('superadmin');
  Route::get('/admin', 'Pages\SuperAdmin@admin')-
>name('admin');
  Route::get('/student',
'Pages\SuperAdmin@student')->name('student');
  Route::get('/category',
'Pages\SuperAdmin@letter_type')->name('category');
  Route::get('/letter', 'Pages\SuperAdmin@letter')-
>name('letter');
  Route::get('/report', 'Pages\SuperAdmin@report')-
>name('report');
});

Route::middleware(['language', 'session',
'token.web', 'admin'])->prefix('/{locale}/admin')-
>name('admin.')->group(function () {
  Route::get('/dashboard',
'Pages\Admin@dashboard')->name('dashboard');
```

```
  Route::get('/student', 'Pages\Admin@student')-
>name('student');
  Route::get('/letter', 'Pages\Admin@letter')-
>name('letter');
  Route::get('/report', 'Pages\Admin@report')-
>name('report');
});

Route::middleware(['language', 'session',
'token.web', 'student'])-
>prefix('/{locale}/student')->name('student.')-
>group(function () {
  Route::get('/dashboard',
'Pages\Student@dashboard')->name('dashboard');
  Route::get('/letter', 'Pages\Student@letter')-
>name('letter');
  Route::get('/report', 'Pages\Student@report')-
>name('report');
});

Route::middleware(['language', 'session',
'token.web'])->prefix('/{locale}/download')-
>name('download.')->group(function () {
  Route::get('/letter', 'Pages\Download@letter')-
>name('letter');
  Route::get('/template/letter-type',
'Pages\Download@letter_type_template')-
>name('letter_type.template');
});
```

2) View

```
<?php
<div class="modal fade"
id="modal_form_letter_student" tabindex="-1"
role="dialog" aria-
labelledby="modal_form_letter_student_title" aria-
hidden="true">
  <div class="modal-dialog modal-lg" role="
document">
    <form id="form_letter_student"
class="form_validation">
      <div class="modal-content">
        <div class="modal-header">
          <h5 class="modal-title" id="modal-
title">Add Letter</h5>
          <button type="button" class="close"
data-dismiss="modal" aria-label="Close">
            <span aria-
hidden="true">&times;</span>
```

```
          </button>
        </div>
        <div class="modal-body">
          <div class="alert alert-danger mt-1"
id="letter_type_note" role="alert">
            <div class="alert-body"> Please ensure
the file should be in <strong>.pdf!</strong></div>
          </div>
          <div class="row">
            <div class="col-7">
              <div class="form-group">
                <label class="form-
label">@lang('label.letter_type')</label>
                <select class="form-control
select2" name="type_id"
onchange="generate_dynamic_form()">
                  @foreach ($list_letter_type as
$a)
                    <option value="{{ $a->id
}}">{{ $a->name }}</option>
                  @endforeach
                </select>
              </div>
              <div id="container_dynamic_form">
              </div>
              <blockquote class="blockquote pl-1
border-left-primary border-left-3" id="comment">
                <p class="mb-0 comment_text">
                  Sometimes when you innovate, you
make mistakes. It is best to admit them quickly,
and get on with improving
                  your other innovations.
                </p>
                <footer class="blockquote-footer
comment_editor">
                  Steve Jobs
                  <cite title="Source
Title">Entrepreneur</cite>
                </footer>
              </blockquote>
            </div>
            <div class="col-5">
              <div class="form-group">
                <label class="form-
label">@lang('label.status')</label>
                <select class="form-control
select2" name="state">
                  <option
value="0">@lang('label.save_as_draft')</option>
                  <option
value="1">@lang('label.submit_for_review')</option
>
```

```
                </select>
              </div>
              <div class="form-group"
id="container_download">
              </div>
              <div class="form-group"
id="container_attachment">
                <label class="form-
label">@lang('label.attachment')</label>
                <input id="attachment"
class="dropify" data-allowed-file-extensions="pdf"
data-max-file-size="10M" id="file" name="file"
type="file">
              </div>
            </div>
          </div>

        </div>
        <div class="modal-footer">
          <button type="submit" class="btn btn-
primary waves-effect waves-float waves-light"
id="submit">Submit</button>
          <button type="reset" class="btn btn-
outline-secondary" data-
dismiss="modal">Cancel</button>
        </div>
      </div>
    </form>
  </div>
</div>
```

3) Model

```
<?php

namespace App\Models\User;

use DateTimeInterface;
use
Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\Storage;

class SuperAdminModel extends Model
{
  use HasFactory;
  protected $table = 't_user_super_admin';
  protected $fillable = [
    'auth_id',  'email', 'name', 'no_hp',
'address', 'photo', 'state',
    'created_at', 'updated_at',
  ];
```

```
  protected $hidden = [
    'auth_id', 'created_at'
  ];
  protected $casts = [
    // 'updated_at' => 'datetime:d F Y, H:i',
  ];
  protected $appends = [
    'photo_path', 'photo_url'
  ];
  protected function
serializeDate(DateTimeInterface $date)
  {
    return $date->format('Y-m-d H:i:s');
  }
  public function auth()
  {
    return $this->belongsTo(AuthModel::class,
'auth_id', 'id');
  }
  public function getPhotoPathAttribute()
  {
    if (!empty($this->attributes['photo'])) {
      $filePath = 'public/superadmin' . '/' .
$this->attributes['photo'];
      return $filePath;
    } else {
      return null;
    }
  }
  public function getPhotoUrlAttribute()
  {
    if (!empty($this->attributes['photo'])) {
      if (Storage::disk('public')-
>exists('superadmin/' . $this-
>attributes['photo'])) {
        return asset(Storage::url('superadmin/' .
$this->attributes['photo']), env('FORCE_HTTPS'));
      } else {
        return asset('app-
assets/images/avatars/user.png',
env('FORCE_HTTPS'));
      }
    } else {
      return asset('app-
assets/images/avatars/user.png',
env('FORCE_HTTPS'));
    }
  }
}
```

A) Integrasi dan Pengujian Sistem

2) *Testing*

Testing adalah tahap pengujian dari sistem informasi yang telah rampung dibuat. Setelah program dilakukan *testing* apabila terdapat kesalahan maka program akan diperbaiki, dan apabila sistem informasi tidak terdapat kesalahan maka program akan dicobakan secara langsung ke lapangan.

4) Pengujian dengan Metode *Black Box*

Tabel 4.12 *Blackbox Testing*

No	Rancangan Dan Proses	Yang diharapkan	Ket
1	**2**	**3**	**4**
1	Buka halaman login	Tampil halaman login admin	Berhasil
2	Buka menu utama	Tampil halaman menu utama	Berhasil
3	Klik menu data super admin	Tampil halaman super admin	Berhasil
4	Klik form input super admin	Tampil form input super admin	Berhasil
5	Klik menu data admin	Tampil halaman admin	Berhasil
6	Klik form input admin	Tampil form input admin	Berhasil
7	Klik menu data fakultas	Tampil halaman fakultas	Berhasil
8	Klik form input fakultas	Tampil form input fakultas	Berhasil
9	menu data prodi	Tampil halaman prodi	Berhasil

A	B	C	D
10	Klik form input prodi	Tampil form input prodi	Berhasil
11	Klik menu data mahasiswa	Tampil halaman data mahasiswa	Berhasil
12	Klik form input mahasiswa	Tampil form input data mahasiswa	Berhasil
13	Klik menu jenis surat	Tampil halaman jenis surat	Berhasil
14	Klik menu input jenis surat	Tampil input jenis surat	Berhasil
15	Klik menu permohonan	Tampil halaman permohonan	Berhasil
16	Klik menu input permohonan	Tampil input permohonan	Berhasil

5. *Deployment*

a. *Delivery*

Pengiriman produk ke *user* dilakukan dengan cara membagikan *link* dari sistem informasi melalui sosial media kepada mahasiswa, dan untuk web admin dapat di akses oleh fakultas melalui jaringan dengan mengakses url yang telah diberikan.

b. *Support*

Memiliki manfaat untuk fakultas Ekonomi dan Bisnis Islam, sistem berjalan dengan baik, sistem mudah digunakan, menarik, efisien, dan inspiratif. Sistem dapat digunakan dimana saja dan kapan saja.

c. *Feedback*

Dalam tahap ini penulis telah melakukan pembaharuan terhadap sistem dan koreksi dari berbagai kekurangan yang telah melalui tahap pengujian sistem.

C. Uji Porduk

1. Uji Validitas

Uji Validitas dilakukan dengan tujuan untuk mendapatkan hasil yang berkualitas dan siap digunakan. Uji validitas juga bertujuan untuk melihat ketepatan dari isi produk yang dihasilkan. Uji validitas produk dilaksanakan dengan cara mempresentasikan dan meminta penilaian dari para ahli dibidang komputer. Tahap pengujian validitas ini peneliti tujukan kepada beberapa ahli sistem *computer*. Penulis melakukan dua jenis validitas yaitu validitas konstruk program dan konten program yaitu :

Tabel 4.13 Validitas Produk

No	**Validator**						**Nilai/V**
	Bapak Riri Okra, M.Kom		**Ibu Gusnita Darmawati, M.Kom**		**Ibu Dr. Liza Efriyanti, S.Si, M.Kom**		
	Skor / r	**S**	**Skor / r**	**S**	**Skor / r**	**S**	
Item 1	5	**4**	5	**4**	5	**4**	**1,0**
Item 2	5	**4**	5	**4**	5	**4**	**1,0**
Item 3	5	**4**	5	**4**	5	**4**	**0,9**
Item 4	4	**3**	5	**4**	5	**4**	**0,8**

Item 5	4	**3**	5	**4**	5	**4**	**1,0**
Item 6	5	**4**	4	**3**	5	**4**	**0,9**
Item 7	4	**3**	5	**4**	5	**4**	**0,9**
Item 8	4	**3**	5	**4**	5	**4**	**1,0**
Item 9	5	**4**	5	**4**	5	**4**	**1,0**
Item 10	5	**4**	5	**4**	5	**4**	**1,0**
Item 11	4	**3**	5	**4**	5	**4**	**0,9**
Item 12	4	**3**	5	**4**	5	**4**	**0,9**
Item 13	5	**4**	5	**4**	5	**4**	**0,8**
Item 14	4	**4**	4	**3**	5	**4**	**0,9**
Item 15	5	**4**	4	**3**	5	**4**	**0,8**
Item 16	5	**4**	4	**3**	5	**4**	**0,8**
Item 17	5	**4**	5	**4**	5	**4**	**0,8**
Item 18	5	**4**	5	**4**	5	**4**	**0,8**
Item 19	4	**3**	5	**4**	5	**4**	**0,6**
Jumlah							**17,2**
Rata-rata							**0,9**

Berdasarkan hasil uji validitas yang telah dilaksanakan, dan dilakuakn perhitunga dengan rumus yang ada maka didapatkan nilai validitas 0,9 yang artinya produk ini valid.

2. Uji Praktikalitas

Uji praktikalitas dilakukan dengan tujuan untuk mendapatkan apakah produk yang dihasilkan mudah digunakan atau tidak. Kepraktisan dari produk yang dibuat dapat diukur dari pendapat pihak sekolah cara mengisi angket praktikalitas yang telah penulis buat, untuk uji praktikalitas produk penulis tujukan kepada staf yang ada di Fakultas Ekonomi dan Bisnis Islam. adapun tabel hasil penilaian dari uji praktikalitas yaitu :

Tabel 4.14 Uji Praktikalitas

No	Nama Penguji praktikalitas	Skor					Skor Maks	*po*	*Pe*	K
		Item1	Item2	Item3	Item4	Item5				
1	Wilda Sri Sartika	4	5	4	5	5	5	0,92	0,08	0,91
2	Sasmiarti, S.Ag, SIP, M.H	5	5	5	5	5	5	1	0	1
Jumlah										**1,91**
Rata-rata										**0,96**

Berdasarkan hasil uji praktikalitas yang telah dilakukan kepada staf yang ada di FEBI maka, didapatkan hasil dari aspek evaluasi diperoleh rata-rata yaitu 0,96 dengan kategori sangat praktis.

3. Uji Efektifitas

Uji efektifitas dari E-LETTER ini diperoleh dari lembar efektifitas yang diisi oleh beberapa mahasiswa. adapun tabel hasil uji efektifitas yaitu :

Tabel 4.15 Hasil Efektifitas Produk

No	Validator	Sebelum (*si*)	Sesudah (*sf*)
1	Nurhidayati	25	90
2	Nanda Tasya	45	90
3	Winda Anggraini	50	95
4	Warman Sakri	50	90
5	Rahmat Ibrahim	50	95
6	Rani Septiawan	30	90
7	Nadin Efendi	50	100
8	Linda Wulandari	45	90
9	Tasya Putri	30	95
10	Popy Oktaniani	30	90
11	Irvan Afandi	40	90
Jumlah		445	1015
Rata-rata		40,45	92,27

$$G = \frac{Sf - Si}{100 - Si}$$

$$G = \frac{92,27 - 40,45}{100 - 40,45}$$

$$G = \frac{51,81}{59,54}$$

G = 0,87

Berdasarkan tabel hasil uji efektifitas dari beberapa Mahasiswa di FEBI dapat dilihat bahwa hasil dari aspek evaluasi diperoleh rata-rata yaitu 0,87 dengan kategori efektifitas tinggi.

BAB V

PENUTUP

A. Kesimpulan

Pembahasan yang telah penulis kemukakan pada bab-bab sebelumnya berdasarkan dari hasil penelitian yang telah dilakukan, yaitu mengenai sistem E-Letter menggunakan *framework Laravel* dan *bootstrap,* maka dapat disimpulkan hal-hal sebagai berikut :

1. Penulis menggunakan software *Visual Studio Code* serta di dukung oleh software pendukung lainnya maka dapat di rancang sebuah sistem E-Letter.
2. E-Letter dapat mempermudah FEBI dalam proses adminstrasi surat menyurat sehingga berjalan lebih efektif dan efisien, dapat mempermudah mahasiswa dalam melakukan admnistrasi surat menyurat di FEBI seperti pengajuan surat dan mendapatkan surat yang diterbitkan.
3. E-Letter yang dirancang mendapat nilai validasi akhir 0,9 nilai praktikalitas akhir 0,96 dan nilai efektifitas akhir 0,87 dapat disimpulkan bahwa sistem sistem E-Letter valid, sangat praktis, dan efektif di gunakan.

B. Saran

Berdasarkan kesimpulan di atas dan setelah melihat hasil penelitian yang telah dilakukan, maka penulis dapat mengemukakan beberapa saran yang dapat menjadi bahan pertimbangan lebih lanjut dalam rangka meningkatkan sistem E-Letter :

Adapun saran-saran sebagai berikut :

1. Perubahan dari sistem Surat menyurat secara manual, ke sistem yang baru membutuhkan waktu untuk penyesuaian bagi staf.
2. Sistem informasi ini dibutuhkan tenaga kerja yang terampil dan ahli agar sistem dapat berjalan dengan baik.

DAFTAR PUSTAKA

[1] D. A. bin Muhammad, *Tafsir Ibnu Katsir Jilid 7*. Bogor: Pustaka Imam Asy-Syafi'i, 2004.

[2] Iswantir, "Paradigma Lembaga Pendidikan." AURA (CV. Anugrah Utama Raharja), Bandar Lampung, p. 20, 2019.

[3] Z. Sesmiarni, "Efektivitas Pembelajaran Berbasis Cara Kerja Otak pada Mata Pelajaran IPA," *J. Teknol. Pendidik.*, vol. 14, no. 2, pp. 110–120, 2012.

[4] Z. Sesmiarni, "Efikasi Diri Mahasiswa Di Perguruan Tinggi," *Anal. J. Pendidik.*, vol. 11, no. 2, p. 115, 2014.

[5] Y. Hadiwijaya, *Administrasi Pendidikan*. Medan: Perdana Publishing.

[6] Peraturan Pemerintah Republik Indonesia Nomor 4 Tahun 2014 Tentang Penyelanggaraan Pendidikan Tinggi dan Pengelolaan Perguruan Tinggi, "PP Nomor 04 Tahun 2014," 2014.

[7] M. F. Ramadhan *et al.*, "Administrasi Surat Menyurat Studi Kasus : Kesekretariatan Badan Eksekutif," vol. 1, no. 1, 2018.

[8] E. Maiyana and T. Mengkasrinal, "Pengembangan Sistem Informasi Surat Keterangan Pendamping Ijazah Berbasis Web dan Mobile Android," *Pros. SISFOTEK 2017*, vol. 1, no. 1, pp. 7–16, 2017.

[9] A. D. Rachmatsyah and D. Merlini, "Perancangan Sistem Informasi Administrasi Surat Berbasis Desktop," *J. SISFOKOM*, vol. 06, no. 2, pp. 130–136, 2017.

[10] F. Masykur, I. Makruf, and P. Atmaja, "Sistem Administrasi Pengelolaan Arsip Surat Masuk Dan Surat Keluar Berbasis Web," *Indones. J. Netw. Secur.*, vol. 4, no. 3, pp. 1–7, 2015.

[11] H. Antonio and N. Safriadi, "Rancang Bangun Sistem Informasi Administrasi Informatika (SI-ADIF)," vol. 4, no. 2, pp. 12–15, 2012.

[12] R. N. 'Ichsan, M. 'Yusuf, and A. 'Karim, *Komunikasi Bisnis*, Komunikasi Bisnis., vol. 275. Medan: CV. Manhaji, 2019.

[13] Pusat Bahasa Departemen Pendidikan Nasional, *Kamus Bahasa Indonesia*. Jakarta: Pusat Bahasa Departemen Pendidikan Nasional, 2008.

[14] R. Shalahuddin, "Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek." Informatika Bandung, Bandung, pp. 137–166, 2014.

[15] Ladjamudin bin Al-bahra, “Analisis dan Desain Sistem Informasi.” Graha Ilmu, Yogyakarta, pp. 263–268, 2005.

[16] W. Istiono, “Pengembangan Sistem Aplikasi Penilaian dengan Pendekatan MVC dan Menggunakan Bahasa PHP dengan Framework Codeigniter dan Database MYSQL pada Pahoa College Indonesia,” pp. 53–59, 2016.

[17] L. Pkl, P. Devisi, and H. Pt, “2) 1,2,” vol. 2, no. 2, pp. 12–26, 2018.

[18] I.D.Id, *Framework Codeigniter Sebuah Panduan dan Best Practice*. Pekanaru: Kode.web.id-Kumpulan Tutorial Komputer, 2011.

[19] and K. P. I. M. Tegeh, I. N. Jampel, *Model Penelitian Pengembangan*. yogyakarta: Graha Ilmu, 2014.

[20] P. ; S. Setiawan, “Rancang Bangun Aplikasi Pengolahan Data Evaluasi Proses Belajar Mengajar Berbasis Web,” *Jsika*, vol. 4, no. 2, pp. 1–6, 2015.

[21] S. Derta, “Perancangan Sistem Informasi Pengelolaan Praktik Kerja Lapangan (PKL) di SMK Negeri 2 Bukittinggi,” *J. Sains, Inform. dan Ekon.*, vol. 2, no. 2, pp. 79–86, 2019.

[22] R. R. Fadila, W. Aprison, and H. A. Musril, “Perancangan Perizinan Santri Menggunakan Bahasa Pemograman PHP / MySQL Di SMP Nurul Ikhlas,” vol. 11, no. 2, pp. 94–105, 2019.

[23] Anhar, *Panduan Menguasai PHP &MySQL Secara Otodidak*, 1st ed. Jakarta: mediakita, 2010. [Online]. Available: https://www.google.co.id/books/edition/PHP_MySql_Secara_Otodidak/J711efbP9LYC?hl=id&gbpv=1

[24] W. Istiono, H. Hijrah, and S. Sutarya, “Pengembangan Sistem Aplikasi Penilaian dengan Pendekatan MVC dan Menggunakan Bahasa PHP dengan Framework Codeigniter dan Database MYSQL pada Pahoa College Indonesia,” *J. TICOM*, vol. 5, no. 1, pp. 53–59, 2016, [Online]. Available: https://media.neliti.com/media/publications/93757-ID-pengembangan-sistem-aplikasi-penilaian-d.pdf

[25] R. Abdulloh, *7 Materi Pemograman Web untuk Pemula 5: Laravel & MariaDB*, 1st ed. Jakarta: PT Elex Media Komputindo, 2022.

[26] Y. Yudhanto and H. A. Prasetyo, *Panduan Mudah Belajar Framework Laravel*, 7th ed. Jakarta: PT Elex Media Komputindo, 2018.

[27] S. M. Ibnu, *Otodidak Web Programming: Membuat Website Edutaintment*, 6th ed. Jakarta: PT Elex Media Komputindo, 2020.