

DESAIN DAN IMPLEMENTASI APLIKASI MANAJEMEN *REPOSITORY* SURAT BERBASIS WEB

(STUDI KASUS: FAKULTAS TEKNIK UNIVERSITAS BENGKULU)

Mochammad Yusa¹, Funny Farady Coastera^{2*}, Wahyu Syahputra³, Stefani Tasya Hallatu⁴

^{1,2,3,4} Program Studi Infomatika, Fakultas Teknik, Universitas Bengkulu.
Jl. WR. Supratman Kandang Limun Bengkulu 38371A INDONESIA
(telp: 0736-341022; fax: 0736-341022)

¹ mohammad.yusa@unib.ac.id

^{2*} faradyc@unib.ac.id

³ syahputrawahyu61@gmail.com

⁴ fhallatu08@gmail.com

*Corresponding Author

Abstrak: Surat merupakan salah satu media informasi yang sering digunakan dalam menyampaikan sebuah informasi kepada orang lain. Pada lingkungan Fakultas Teknik ada beberapa kendala di bidang surat menyurat khususnya Surat Keputusan dan Surat Tugas. Pada Lingkungan Fakultas Teknik Universitas Bengkulu, pendistribusian Surat Keputusan dan Surat Tugas masih dilakukan secara konvensional dengan menyerahkan langsung terhadap yang bersangkutan dalam bentuk *hardcopy*. Selain itu, SK dan Surat Tugas yang menumpuk juga sangat sulit untuk diurutkan berdasarkan tanggalnya sesuai dengan kebutuhan - kebutuhan penggunaan SK dan Surat Tugas tersebut. Penelusuran terkait resmi atau tidaknya SK dan Surat Tugas yang dikeluarkan juga menjadi kendala lainnya meskipun potensinya kecil namun tidak menutup kemungkinan hal ini dapat terjadi. Sehingga dilakukan penelitian ini yang bertujuan untuk merancang sistem informasi yang memungkinkan dapat mengarsipkan data histori surat keputusan dan surat tugas yang dikeluarkan dari Fakultas Teknik Universitas Bengkulu. Pembuatan sistem informasi ini menggunakan metode pengumpulan data dengan mengumpulkan data dari dokumen atau catatan yang ada, dan juga menggunakan metode studi literatur dengan menelusuri sumber – sumber tulisan lainnya. Untuk mencapai hasil penelitian ini dilakukan analisis masalah, kebutuhan, dan lainnya sehingga di dapatkan hasil perancangan UML dan perancangan sistem informasi beserta dengan pengujiannya.

Kata Kunci: Surat, Surat Keputusan, Surat Tugas, Sistem Informasi, Manajemen surat, Laravel.

Abstract: *Letters are a medium of information that is often used in conveying information to others. In the Faculty of Engineering environment, there are several obstacles in the field of correspondence, especially decrees and assignments. In the Faculty of Engineering, University of Bengkulu, the distribution of Decree and Assignment Letters is still done conventionally by handing over to the person concerned in hardcopy. In addition, the accumulated SK and Assignment Letters are also very difficult to sort by date according to the needs of the SK and Assignment Letter. Investigating whether or not the SK and Assignment Letter were issued is another obstacle, although the potential is small, it does not rule out the possibility of this happening. So this research was carried out which aims to design an information system that allows historical data archiving of decisions and assignments issued from the Faculty of Engineering, University of Bengkulu. Making this information system uses the data collection method by collecting data from existing documents or notes, and also using the literature study method by tracing other sources of deafness. To achieve the results of this research, analysis of problems, needs, and others was carried out so that the results of UML design and information system design were obtained along with their testing.*

Keyword: *Letters, Decree, Letter of Assignment, Information Systems, Mail management, Laravel.*

I. PENDAHULUAN

Surat Keputusan (SK) adalah surat yang berisi suatu keputusan yang dibuat oleh pimpinan suatu organisasi atau lembaga pemerintahan berkaitan dengan kebijakan organisasi atau lembaga tersebut. Surat Keputusan digunakan untuk memberikan disposisi pekerjaan terkait dengan penyelenggaraan kegiatan yang menunjang aktivitas dari civitas

akademik di lingkungan Fakultas Teknik Universitas Bengkulu.

Selain menunjang kegiatan aktivitas akademik, SK juga menjadi dokumen yang penting bagi individu civitas akademisi khususnya bagi tenaga pendidik (dosen). Menurut Peraturan Pemerintah (PP) No 37 Tahun 2009 tentang dosen menyatakan bahwa beban kerja minimal seorang dosen adalah melaksanakan tridharma perguruan tinggi setara dengan 12 (dua belas) sks dan maksimal setara 16 (enam belas) sks pada setiap semester sesuai dengan kualifikasi akademik [1]. Dalam 12 (dua belas) sks tersebut, salah satunya adalah unsur penunjang yaitu kegiatan yang dilakukan dosen di luar yang dibuktikan dengan Surat Keputusan (SK). Selain menunjang aktivitas pelaporan Beban Kerja Dosen (BKD), SK juga memiliki peranan penting dalam perhitungan angka kredit dosen yang yang dikonversikan dalam perhitungan daftar usulan angka kredit untuk kenaikan jabatan fungsional dosen. Dalam beberapa kasus, menurut Surat Keputusan Rektor Universitas Bengkulu No.2143/UN30/HK/2019, SK menjadi dokumen yang dapat membuktikan bahwa dosen tersebut mendapatkan tugas tambahan yang nantinya dapat dikonversikan menjadi tunjangan remunerasi baik untuk tenaga pendidik (dosen) maupun tenaga kependidikan [2]. Sehingga SK memiliki peranan yang penting untuk individu civitas akademika Universitas Bengkulu.

Saat ini, Manajemen surat keputusan yang ada di Fakultas Teknik Universitas Bengkulu sendiri masih mengalami beberapa kendala khususnya pendistribusian SK terhadap individu yang ditugas. Saat ini sistem pendistribusian masih dilakukan secara konvensional dengan menyerahkan langsung terhadap yang bersangkutan dalam bentuk *hardcopy*. Pendistribusian yang manual dan konvensional memiliki beberapa kelemahan diantara tidak fleksibel dan tidak *paperless* serta

memakan waktu yang cukup lama dalam pendistribusiannya. Selain itu, SK yang menumpuk juga sangat sulit untuk diurutkan berdasarkan tanggalnya sesuai dengan kebutuhan-kebutuhan penggunaan SK tersebut. Penelusuran terkait resmi atau tidaknya SK yang dikeluarkan juga menjadi kendala lainnya meskipun potensinya kecil namun tidak menutup kemungkinan hal ini dapat terjadi. Berdasarkan kendala-kendala tersebut, manajemen SK perlu ditindaklanjuti sebagai solusi untuk mengatasi masalah-masalah kearsipan SK di lingkungan Fakultas Teknik Universitas Bengkulu.

Berkembangnya teknologi informasi menciptakan solusi baru dalam proses manajemen yang disebut Sistem Informasi. Sistem Informasi (SI) merupakan kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen [3]. Sistem Informasi menjadi salah satu terobosan yang dapat mendukung proses bisnis dalam sebuah organisasi. Penelitian [4] mengembangkan sistem informasi surat keputusan pada Universitas Udayana Bali. Dalam penelitian tersebut dinyatakan bahwa sistem informasi dibangun harus berdasarkan kebutuhan fungsionalitas dari setiap organisasi sehingga perlu dilakukan analisis yang menyesuaikan dengan kebutuhan manajerial organisasi. Penelitian lain [5] menyebutkan bahwa institusi pendidikan seperti universitas sangat penting untuk dibangun sebuah repository center yang dapat memudahkan civitas akademisi dalam mencapai tujuan. Kemudian institusi selain universitas yang penting untuk dibangun pengelolaan sistem informasi manajemen surat keputusan seperti yang dilaksanakan oleh [6]- [7]. Dalam beberapa penelitian tersebut disebutkan bahwa sistem informasi merupakan salah satu solusi yang efektif dalam mendukung proses hilirisasi proses dalam sebuah organisasi.

Sistem informasi manajemen surat merupakan hal yang penting sehingga dalam beberapa penelitian sudah ditelaah. Namun model organisasi yang berbasiskan pada sumber daya membentuk *customization* dalam setiap organisasi. Sehingga untuk memodelkan sistem kearsipan surat dibutuhkan analisis berdasarkan struktur dan prosedur organisasi sesuai dengan keadaan dan sistem yang berjalan di organisasi tersebut. Mengingat pentingnya pengelolaan sistem informasi repository Surat keputusan di lingkungan Fakultas Teknik Universitas Bengkulu maka penting untuk dilakukan kajian dan perancangan sistem informasi kearsipan surat keputusan di lingkungan FT Universitas Bengkulu.

II. PENELITIAN TERKAIT

Pengembangan sistem informasi repository kearsipan surat pernah dilakukan di berbagai instansi pemerintahan maupun pendidikan. Penelitian [7] yang bertujuan untuk membuat sebuah sistem pengarsipan surat masuk dan keluar di intruksi pendidikan sekolah menengah atas di Jepara menunjukkan bahwa dengan mengimplementasi sistem informasi untuk manajemen surat keluar dan masuk dapat mengoptimalkan waktu dan membantu organisasi dalam mencapai visi dan misi organisasi. Penelitian [6] yang bertujuan untuk mengembangkan sistem informasi pada Sub Bagian Organisasi Tata Laksana dan Perundang-undangan UIN Syarif Hidayatullah dengan metode pengembangan *Rapid Agile Development* (RAD). Metode ini memiliki keunggulan dalam mempercepat proses pembuatan namun dalam proses perawatan dan maintenance metodologi ini masih cukup ditelusuri khususnya ketika terdapat kesalahan atau error dalam barisan code. Metode pengembangan yang akan dilakukan dalam penelitian ini adalah metode waterfall dimana metode tersebut lebih terstruktur khususnya dalam pengembangan kedepannya. Kemudian

penelitian yang dilakukan oleh [8] yaitu mengembangkan sistem manajemen surat yang termonitoring di lingkungan fakultas. Sistem yang dikembangkan menggunakan web service yang akan diadopsi dalam penelitian yang dilakukan namun sistem yang akan dikembangkan juga mempertimbangkan hak akses user sesuai dengan Sumber Daya Manusia (SDM) yang ada di lingkungan Fakultas Teknik Universitas Bengkulu.

Penelitian lainnya adalah yang dilakukan oleh [9], yang mengembangkan proses pembuatan surat berdasarkan kebutuhan di bagian kepegawaian UIN Syarif Hidayatullah. Hasil penelitian menunjukkan bahwa pengembangan sistem memiliki manfaat yang besar khususnya dalam proses pengolahan administrasi tentang kepegawaian di lingkungan universitas. Penelitian selanjutnya adalah penelitian yang dilakukan oleh [10] untuk merancang dan mengembangkan sistem kearsipan terintegrasi pada pemerintahan provinsi Nusa Tenggara Barat. Dalam penelitian ini disimpulkan bahwa perancangan yang baik akan mengoptimalkan proses manajerial dalam sebuah tatanan yang terstruktur. Penelitian [11] menyebutkan bahwa kustomisasi perangkat lunak sistem informasi akan lebih baik jika disesuaikan dengan sumber daya yang ada. Dalam penelitian tersebut sistem yang sudah dikembangkan mampu meng-cover 11 (sebelas) kebutuhan fungsionalitas dalam lingkungan institusinya dan akan dikembangkan ke sistem multi-rapat. Penelitian [12] dalam penelitiannya menyebutkan bahwa sistem informasi yang terstruktur akan membantu proses administrasi berjalan dengan lebih cepat dan terukur. Berdasarkan penelitian-penelitian tersebut di atas, maka penelitian yang akan dilakukan adalah perancangan sistem informasi yang fleksibel sesuai dengan sumber daya akan meningkatkan proses pelayanan baik dalam proses manajerial maupun proses administrasi sehingga penelitian ini

akan sangat bermanfaat jika dapat direalisasikan di lingkungan Universitas Bengkulu.

III. METODE PENELITIAN

Dalam penelitian ini penulis mengumpulkan data dengan menggunakan metode:

a. Pengumpulan Data

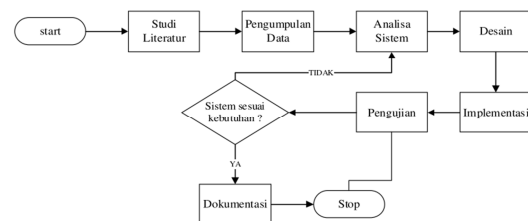
Dalam penelitian ini, pengumpulan data diambil dari dokumen - dokumen yang ada atau catatan - catatan yang tersimpan, seperti data surat, data dosen, data *staff*, dan lainnya.

b. Studi Literatur

Dalam penelitian ini Perangkat Lunak ini penelusuran sumber – sumber tulisan yang sudah pernah di buat sebelumnya untuk menyelesaikan masalah pada penyelesaian penelitian ini.

c. Desain penelitian

Secara umum alur penelitian yang digunakan dalam usulan ini dapat digambarkan melalui ilustrasi diagram pada Gambar 1.



Gambar 1. Bagan alur penelitian

Gambar 1 merupakan rancangan desain penelitian yang akan dilaksanakan. Langkah pertama adalah Studi Literatur. Studi literatur mempelajari hal-hal yang terkait dengan pengembangan sistem. Perancangan sistem informasi pengelolaan surat dan kearsipan menggunakan metode waterfall, framework Codeigniter dan twitter bootstrap dalam membangun sistem. Langkah kedua adalah Pengumpulan Data. Pengumpulan data dilakukan system manajemen konvensional kearsipan surat yang telah dilaksanakan pada saat ini dengan berupa metode diantaranya metode wawancara dan

observasi. Wawancara dilakukan kepada staff bagian kearsipan dan wakil dekan bagian sumber daya serta sekretaris dekan FT untuk mendapatkan informasi mengenai sistem pengelolaan surat dan arsip yang sedang berjalan dan informasi mengenai sistem yang diharapkan. Observasi atau melakukan peninjauan secara langsung pada Sub Bagian Arsip untuk mengetahui proses sistem yang sedang berjalan. Langkah selanjutnya adalah analisis Sistem. Analisis kebutuhan sistem yang digunakan untuk membangun sistem informasi pengelolaan surat dan kearsipan yaitu Desain Sistem yang terdiri dari *Use Case*, *Class Diagram*, *Entities Relationship Diagram*, dan *Interface design*. Tahap design merupakan tahap penggambaran suatu sistem yang akan dikembangkan. Use case diagram sistem informasi pengelolaan surat dan kearsipan digunakan untuk menjelaskan kegiatan dari aktor yang dapat dilakukan oleh petugas Ekspedisi dan Arsip dan pegawai Fakultas Teknik Universitas Bengkulu. *Class diagram* merupakan diagram yang menunjukkan *class-class* yang ada di sistem dan hubungannya secara logic. *Entity Relationship Diagram* (ERD) merupakan model yang menjelaskan hubungan antar data dalam database berdasarkan objek-objek dasar data. Perancangan antarmuka dibuat sebagai gambaran mengenai user interface yang akan dibangun. Langkah terakhir

adalah implementasi dan disain dari system yang sudah dianalisis dan dirancang menjadi teknologi tepat guna yang siap diimplementasikan dan dijalankan kepada public selingkung FT UNIB.

IV. ANALISIS DAN PERANCANGAN

4.1 Analisis Sistem Yang Sedang Berjalan

Sistem manajemen repository surat di Fakultas Teknik Universitas Bengkulu saat ini masih dilakukan secara manual. Dimana pengelolaan yang dilakukan meliputi pendataan surat yang masih manual, serta pengambilan surat bagi para dosen masih harus di ambil secara manual. Sehingga manajemen repository surat di Fakultas Teknik Universitas Bengkulu saat ini masih belum tertata dan terkelola dengan baik.

4.2 Analisis Masalah

Berdasarkan analisis masalah yang di lakukan terdapat beberapa kendala yang ada di dalam manajemen repository surat di Fakultas Teknik Universitas Bengkulu. Analisis masalah tersebut di lakukan berdasarkan beberapa kriteria yaitu dari segi *Performance*, *Information*, *Economic*, *Control*, *Efeciensy*, dan *Service*. Metode ini dikenal dengan PIECES analisis (*Performance*, *Information*, *Economic*, *Control*, *Efeciensy*, *Service*) [13]. Analisis masalah ini akan dijelaskan pada Tabel 4.1 Analisis Masalah.

Table 1. Analisis Masalah

Faktor	Sistem lama	Sistem baru
<i>Performance</i>	Proses pengelolaan surat masih dilakukan dengan cara manual secara satu per satu dimana masih kurang efisien dalam pengelolaan surat.	Proses pengolahan surat pada sistem baru menggunakan <i>framework</i> yaitu <i>Laravel</i> , menghasilkan kinerja system dalam pengelolaan surat menjadi lebih baik sehingga pengelolaan surat lebih efisien.
<i>Information</i>	Informasi yang diberikan kurang tersampaikan dengan baik karena informasi pemberitahuan adanya surat masih di lakukan secara manual	sistem yang baru dapat menyampaikan informasi surat dengan mengirimkan data surat secara langsung ke dalam data surat dari pengguna.
<i>Economy</i>	Pencetakan surat yang menggunakan banyak kertas sehingga biaya yang di keluarkan cukup besar	Sistem yang baru dapat di akses melalui website serta pencetakan surat dapat dilakukan oleh dosen sendiri dan dapat dipilih sehingga mampu mengurangi biaya.
<i>Control</i>	Pada sistem lama pengelolaan surat masih dilakukan secara manual dengan pencatatan surat secara manual oleh <i>staff</i> sehingga masih kurangnya <i>control</i> pada pengelolaan surat	Pada sistem baru pengelolaan surat langsung di inputkan ke dalam website sehingga data – data surat lebih mudah di <i>control</i> dan dipertanggung jawabkan.
<i>Efficiency</i>	Waktu dan biaya yang dikeluarkan lebih lama.	Lebih efektif, efisien dan hemat biaya.

Faktor	Sistem lama	Sistem baru
Service	Pencatatan dan penyimpanan data surat dilakukan secara manual secara satu per satu oleh <i>staff</i> sehingga adanya kemungkinan surat rusak ataupun hilang	Pencatatan dan penyimpanan data surat dilakukan secara <i>online</i> sehingga data tersimpan lebih aman

4.3 Analisis Kebutuhan

Pada analisis kebutuhan sistem terdapat dua kebutuhan sistem yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Dimana dilakukannya analisis kebutuhan digunakan sebagai batasan dari sebuah sistem dengan menentukan kemampuan dan fungsi penggunaan yang sesuai.

4.3.1 Kebutuhan Fungsional

Kebutuhan fungsional adalah kebutuhan dari fasilitas yang di butuhkan serta aktivitas yang dilakukan pada sistem secara umum. Adapun kebutuhan fungsional pada sistem manajemen repository surat Fakultas Teknik Universitas Bengkulu akan di jelaskan pada Tabel Elisitias di bawah ini.

Tabel 2. Tabel Kebutuhan Fungsional Sistem

No	Kebutuhan Sistem
1.	Sistem yang dapat diakses dalam komputer melalui website
2.	Sistem menampilkan menu login
3.	Sistem menampilkan verifikasi login benar
4.	Sistem menampilkan verifikasi login salah
5.	Sistem menampilkan data Surat Tugas
6.	Sistem menampilkan data surat keterangan
7.	Sistem menampilkan data <i>user</i>
8.	Sistem dapat menampilkan link Pengunduhan Surat
9.	Sistem menampilkan file surat
10.	Sistem bisa melakukan tambah data
11.	Sistem bisa melakukan edit data
12.	Sistem bisa melakukan hapus data
13.	Sistem bisa melakukan unduh data
14.	Sistem terdapat fitur pencarian data
15.	Sistem terdapat fitur pengurutan data
16.	Sistem dapat menambahkan data berupa file berformat PDF
17.	Menggunakan MySql Sebagai Database
18.	Interface Sistem yang mudah digunakan dan dipahami <i>User</i> .

4.3.2 Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional merupakan paparan analisis mengenai *hardware* dan *software* yang di butuhkan pada sebuah sistem. Berikut adalah kebutuhan non-fungsional yang di butuhkan. Sub bab berikut merupakan analisis

kebutuhan fungsional yang terdiri dari Hardware dan Software.

4.3.2.1 Kebutuhan Perangkat Keras (*Hardware*)

Adapun Perangkat keras (*hardware*) Yang mendukung dakam pembuatan sistem ini adalah sebagai berikut :

1. *Processor* : Intel core i7
2. RAM 8192 MB
3. Kapasitas Penyimpanan : 1 TB HDD
4. Tipe Grafis : Nvidia GeForce GT940M 2 GB
5. Resolusi Layar : HD (1366 x 768 piksel)

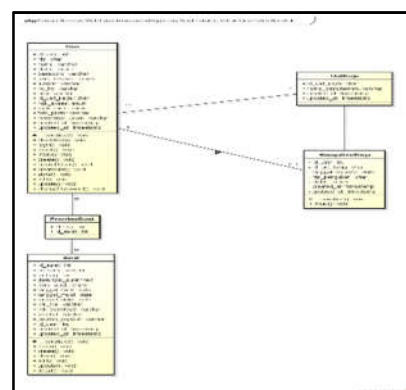
4.3.2.2 Kebutuhan Perangkat Lunak (*Software*)

Adapun Perangkat lunak (*software*) Yang mendukung dakam pembuatan sistem ini adalah sebagai berikut :

1. Visual Studio Code / Sublime Text 3
2. Web Browser : Google Form dan Mozilla Firefox
3. Desain *interface* : Adobe XD
4. Diagram Perancangan : Astah, dan Visio
5. *Framework* : Laravel
6. *Bootstrap*
7. Bahasa Pemrograman : PHP dan JavaScript
8. MySQL
9. Sistem Operasi Windows 10

4. Perancangan Sistem

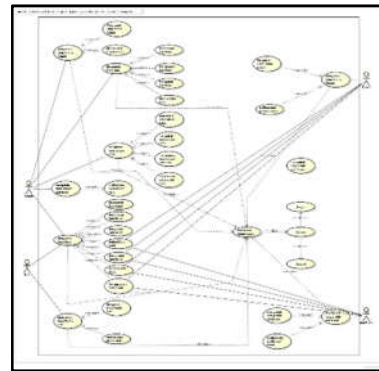
4.4.1 Class Diagram



Gambar 2. Class Diagram Aplikasi

Pada Gambar 2 di atas merupakan relasi yang terjadi di dalam *class diagram* aplikasi yang secara umum diambil dari pendefinisian database. Pada *class diagram* ini memiliki 5 kelas yaitu terdapat class *user*, surat, penerima surat, riwayat unit kerja, dan unit kerja. *User* saling berelasi dengan unit kerja, riwayat unit kerja, dan penerima surat. Penerima surat berelasi dengan *user* dan surat. *Unit* kerja berelasi dengan riwayat unit kerja, dan *user*. Riwayat *unit* kerja juga berelasi dengan *user* dan unit kerja.

Usecase Diagram



Gambar 3. Usecase Diagram Aplikasi

Table 3. Deskripsi Pendefinisian Aktor Pada Usecase Diagram

No.	Aktor	Deskripsi
1.	Admin	Admin adalah orang yang bertugas dan memiliki hak akses yang dapat digunakan untuk melakukan operasi pengelolaan data <i>profie</i> admin, data <i>user</i> , data satuan kerja, data riwayat unit kerja dan data surat
2.	Staff	Staff adalah orang yang di perbolehkan mengelola data surat, dan data <i>profie</i> staff sesuai dengan hak aksesnya.
3.	Dosen	Dosen adalah orang yang dapat mengelola data <i>profie</i> dosen, dan mengelola data surat sesuai dengan hak aksesnya yang hanya dapat melihat data surat, mengunduh data surat, mencari data surat, dan mengurutkan data surat.
4.	Staff Biasa	Staff biasa adalah orang yang dapat mengelola data <i>profie</i> staff biasa, dan mengelola data surat sesuai dengan hak aksesnya yang hanya dapat melihat data surat, mengunduh data surat, mencari data surat, dan mengurutkan data surat.

Table 4. Deskripsi Usecase Diagram

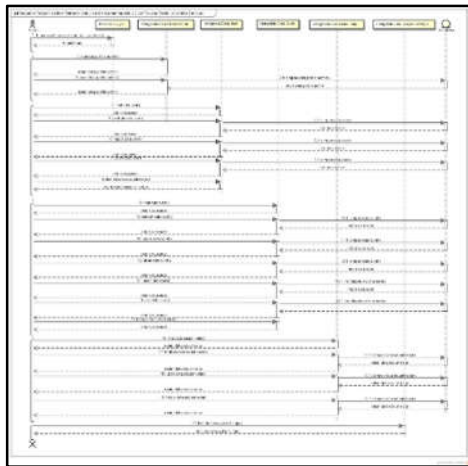
NO.	Use Case	Aktor	Deskripsi
1.	Mengelola Data <i>Profie</i> Admin	Admin	Pada pengelolaan data admin, admin dapat melakukan perubahan terhadap data dari <i>profie</i> admin itu sendiri, dan melihat data <i>profie</i> admin
2.	Mengelola Data <i>User</i>	Admin	Pada pengelolaan data <i>user</i> , admin dapat melakukan penambahan data <i>user</i> , penghapusan data <i>user</i> , pengubahan data <i>user</i> , melihat data <i>user</i> , dan melihat riwayat <i>unit</i> kerja. Data <i>user</i> terkait dengan data admin, data <i>staff</i> , <i>staff</i> biasa dan data dosen.
3.	Mengelola Data Surat	Admin	Pada pengelolaan data surat, admin dapat melakukan penambahan data surat, penghapusan data surat, pengubahan data surat, melihat data surat, mengunduh data surat, mencari data surat, dan mengurutkan data surat sesuai dengan hak aksesnya.
		Staff	Pada pengelolaan data surat, <i>staff</i> dapat melakukan penambahan data surat, penghapusan data surat, pengubahan data surat, melihat data surat, mengunduh data surat, mencari data surat, dan mengurutkan data surat sesuai dengan hak aksesnya.
		Dosen	Pada pengelolaan data surat, dosen dapat melakukan melihat data surat, mengunduh data surat, mencari data surat, dan mengurutkan data surat sesuai dengan hak aksesnya.
		Staff biasa	Pada pengelolaan data surat, <i>staff</i> biasa dapat melakukan melihat data surat, mengunduh data surat, mencari data surat, dan mengurutkan data surat sesuai dengan hak aksesnya.
4.	Mengelola Data <i>Profie</i> Staff	Staff	Pada pengelolaan data <i>profie</i> , <i>staff</i> dapat melakukan melihat data <i>profie</i> dan mengubah data <i>profie</i> .
5.	Mengelola Data <i>Profie</i> Staff biasa	Staff biasa	Pada pengelolaan data <i>profie</i> , <i>Staff</i> biasa dapat melihat data <i>profie</i> dan mengubah data <i>profie</i> .
6.	Mengelola Data <i>Profie</i> Dosen	Dosen	Pada pengelolaan data <i>profie</i> , Dosen dapat melihat data <i>profie</i> dan mengubah data <i>profie</i> .

Tabel 4 merupakan deskripsi pendefinisian aktor pada Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik

4.4.2 Sequence Diagram

Universitas Bengkulu. Tabel 4.4 adalah deskripsi *usecase* pada Aplikasi yang akan dikembangkan:

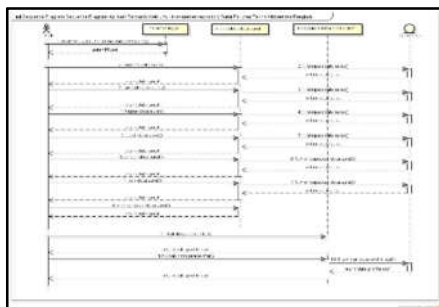
4.4.3.1 Sequence Diagram Admin



Gambar 4. Sequence diagram Admin Aplikasi

Pada gambar 4 merupakan *sequence diagram* dari admin pada “Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu”. Admin bertindak sebagai aktor yang melakukan interaksi pada setiap manajemen yang ada seperti manajemen data *profie* admin, data *user*, dan data surat.

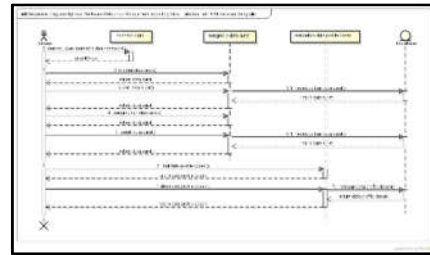
4.4.3.2 Sequence Diagram Staff



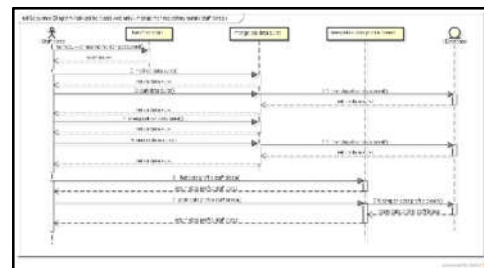
Gambar 5. Sequence diagram Staff

Pada gambar 5 merupakan *sequence diagram* dari *staff* pada “Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu”. *Staff* bertindak sebagai aktor yang melakukan interaksi pada setiap manajemen yang ada seperti manajemen data surat, dan data *profie staff*.

4.4.3.3 Sequence Diagram Dosen dan Staff Biasa



Gambar 6. Sequence diagram Dosen Aplikasi

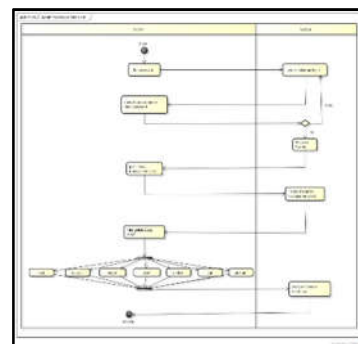


Gambar 7 Sequence diagram Staff Biasa

Pada gambar 6 dan 7 merupakan *sequence diagram* dari dosen dan *staff* biasa pada “Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu”. Dimana dosen dan *staff* biasa bertindak sebagai aktor yang melakukan interaksi pada setiap manajemen yang ada seperti manajemen data surat, dan data *profie* dosen dan *staff* biasa.

4.4.3 Activity Diagram

4.4.4.1 Activity Diagram Admin

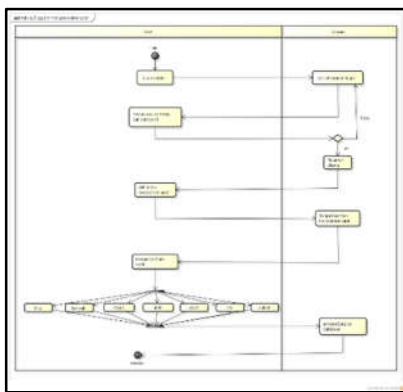


Gambar 8. Activity Diagram Mengelola Data Surat Oleh Admin

Pada gambar 8 *activity diagram* diatas adalah alur kegiatan dari admin dan sistem pada pengelolaan data surat. Pada pengelolaan data surat, pertama admin akan membuka sistem dan sistem akan menampilkan halaman login. Kemudian admin akan melakukan login dengan

memasukkan *username* dan *password*. Kemudian sistem akan mengautentifikasi *username* dan *password* yang di masukkan. Jika *username* dan *password* benar maka akan tampil halaman utama. Jika salah maka akan kembali ke halaman login. Setelah masuk ke halaman utama admin dapat memilih menu manajemen surat dan sistem akan menampilkan halaman manajemen surat. Selanjutnya admin dapat melakukan pengelolaan data surat seperti tambah, hapus, ubah, lihat, unduh, cari, dan mengurutkan. Dan perubahan data yang dilakukan akan di simpan ke dalam *database* oleh sistem.

4.4.4.2 Activity Diagram Staff

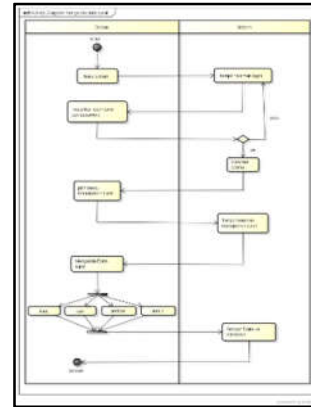


Gambar 9. Activity Diagram Mengelola Data Surat Oleh Staff

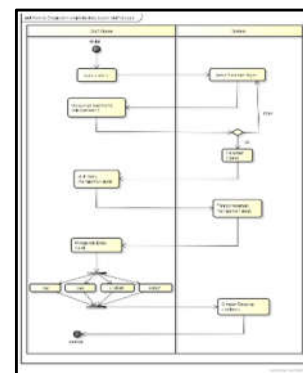
Pada gambar 9 Pada *activity diagram* diatas adalah alur kegiatan dari *staff* dan sistem pada pengelolaan data surat. Pada pengelolaan data surat, pertama *staff* akan membuka sistem dan sistem akan menampilkan halaman login. Kemudian *staff* akan melakukan login dengan memasukkan *username* dan *password*. Kemudian sistem akan mengautentifikasi *username* dan *password* yang di masukkan. Jika *username* dan *password* benar maka akan tampil halaman utama. Jika salah maka akan kembali ke halaman login. Setelah masuk ke halaman utama *staff* dapat memilih menu manajemen surat dan sistem akan menampilkan halaman manajemen surat. Selanjutnya *staff* dapat melakukan pengelolaan data surat seperti tambah, hapus, ubah, lihat, unduh,

cari, dan mengurutkan. Dan perubahan data yang dilakukan akan di simpan ke dalam *database* oleh sistem.

4.4.4.3 Activity Diafram Dosen dan Staff Biasa



Gambar 10. Activity Diagram Mengelola Data Surat Oleh Dosen

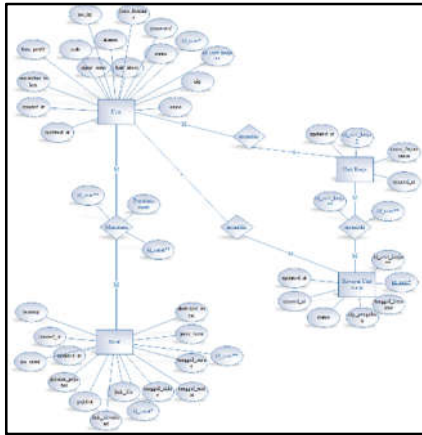


Gambar 11. Activity Diagram Mengelola Data Surat Oleh Staff Biasa

Pada gambar 10 dan 11, Pada *activity diagram* diatas adalah alur kegiatan dari dosen dan *staff* biasa dan sistem pada pengelolaan data surat. Pada pengelolaan data surat, pertama dosen dan *staff* biasa akan membuka sistem dan sistem akan menampilkan halaman login. Kemudian dosen dan *staff* biasa akan melakukan login dengan memasukkan *username* dan *password*. Kemudian sistem akan mengautentifikasi *username* dan *password* yang di masukkan. Jika *username* dan *password* benar maka akan tampil halaman utama. Jika salah maka akan kembali ke halaman login. Setelah masuk ke halaman utama dosen dan *staff* biasa dapat memilih menu manajemen surat dan

sistem akan menampilkan halaman manajemen surat. Selanjutnya dosen dan *staff* biasa dapat melakukan pengelolaan data surat seperti lihat, unduh, cari, dan mengurutkan. Dan perubahan data yang dilakukan akan di simpan ke dalam *database* oleh sistem.

4.4 ERD (Entity Relationship Diagram)



Gambar 12. ERD (Entity Relationship Diagram) Aplikasi

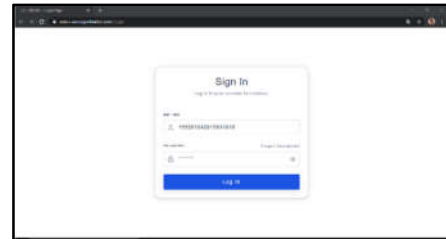
Pada Gambar 12 adalah ERD (*Entity Relationship Diagram*) dari “ Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu ”. Penggunaan ERD diperlukan untuk menggambarkan hubungan antar objek data yang memiliki relasi sehingga memudahkan dalam pembuatan *database*. Penggunaan ERD ini digunakan karena belum adanya sistem manajemen surat ini di lingkungan Fakultas Teknik Universitas Bengkulu. Pada ERD ini memiliki objek data atau entitas berupa *user*, surat, riwayat unit kerja, dan riwayat unit kerja. Pada entitas *user* dan surat memiliki relasi *many to many* sehingga menghasilkan entitas baru berupa penerima surat. Pada entitas *user* dan unit kerja relasi yang digunakan adalah *many to one*. Pada entitas *user* dan riwayat unit kerja relasi yang digunakan adalah *one to many*. Dan Pada entitas unit kerja dan riwayat unit kerja relasi yang digunakan adalah *many to many*.

V. HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan mengenai hasil dan pembahasan dari sistem yang telah dibangun, Penjelasan pada bab ini antara lain terdiri dari implementasi antar muka dan pengujian *black box*.

5.1 Implementasi Antar Muka

(1) Menu Login

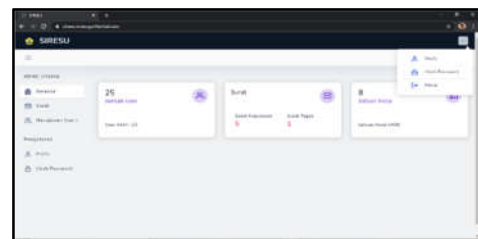


Gambar 13. Tampilan Halaman Login

Gambar 13 merupakan tampilan halaman *login*. Pada halaman *login* yang dapat masuk adalah admin, *staff*, dosen, *staff* biasa. Pada halaman ini *user* dapat mengisi *field* NIP / NIK dan mengisi *field password*. Jika NIP / NIK atau *password* tidak sesuai atau salah maka akan muncul keterangan “ **Error! These credentials do not match our records** “. Jika NIP / NIK dan *password* yang di masukkan benar maka akan lanjut ke halaman *dashboard*.

5.1.1 Implementasi *Interface* Sistem (Admin)

(1) Halaman Utama

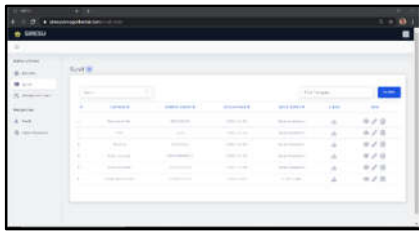


Gambar 14. Tampilan halaman Dashboard (Admin)

Gambar 14 merupakan tampilan dari halaman *dashboard*. Pada tampilan awal halaman *dashboard* terdapat informasi jumlah *user*, jumlah surat, dan jumlah satuan kerja. Pada halaman *dashboard* ini juga terdapat beberapa menu manajemen yang dapat di akses oleh admin seperti manajemen surat, manajemen *user*, manajemen *profil*, dan manajemen ubah *password*. Pada manajemen *user*

terdiri dari manajemen data *user*, manajemen satuan kerja, dan manajemen riwayat *unit* kerja.

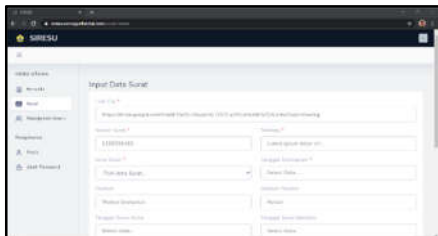
(2) Halaman Surat



Gambar 15. Tampilan Halaman Surat

Gambar 15 merupakan tampilan dari halaman surat. Pada tampilan halaman surat terdapat informasi surat seperti tentang surat, nomor surat, waktu ditetapkan, jenis surat, serta *file* surat yang sudah di tambahkan. Pada halaman surat ini, admin dapat melakukan pengelolaan tambah data surat baru, hapus data surat, ubah data surat yang sudah ada, melihat *file* surat dimana jika kita mengklik *icon* melihat *file* surat maka admin akan di arahkan ke *google drive* dimana tempat *file* surat di simpan, dan pada halaman surat admin juga dapat melakukan pengunduhan *file* surat.

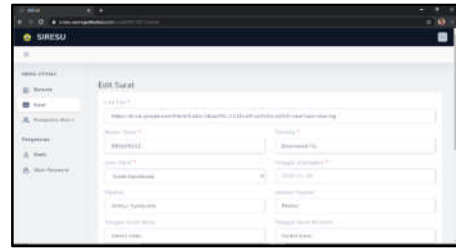
(3) Halaman Input Surat



Gambar 16. Tampilan halaman Input Surat (Admin)

Gambar 16 adalah tampilan dari halaman *input* surat. Pada tampilan halaman *input* surat akan tampil tampilan berupa *form*. Pada *form* ini terdapat beberapa *field* yang dapat di isi seperti *link file*, nomor surat, tentang surat, jenis surat, tanggal ditetapkan, pejabat penandatangan, jabatan pejabat, tanggal surat mulai, tanggal surat berakhir, penerima surat, dan deskripsi surat.

(4) Halaman Edit Surat

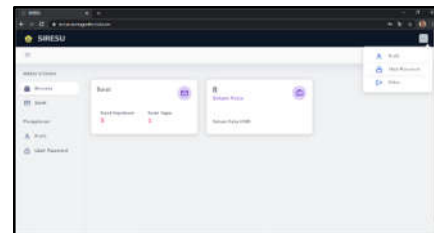


Gambar 17. Tampilan Halaman Edit Surat (Admin)

Gambar 17 adalah tampilan dari halaman *edit* surat. Pada tampilan halaman *edit* surat akan tampil tampilan berupa *form* seperti pada halaman *input*. Admin dapat mengubah isi *field* yang sudah terisi sehingga data berubah.

5.1.2 Implementasi Interface Sistem (Staff)

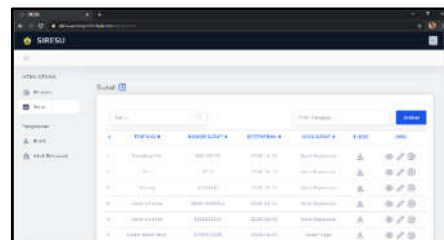
(1) Halaman Utama



Gambar 18. Tampilan Halaman Dashboard (Staff)

Gambar 18 merupakan tampilan dari halaman *dashboard*. Pada tampilan awal halaman *dashboard* terdapat informasi jumlah surat, dan jumlah satuan kerja. Pada halaman *dashboard* ini juga terdapat beberapa menu manajemen yang dapat di akses oleh *staff* seperti manajemen surat, manajemen *profil*, dan manajemen ubah *password*. Pada manajemen – manajemen tersebut *staff* dapat melakukan pengelolaan data.

(2) Halaman Surat

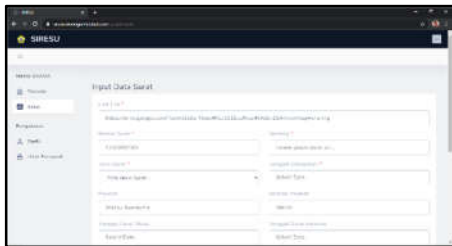


Gambar 19. Tampilan Halaman Surat (Staff)

Gambar 19 merupakan tampilan dari halaman surat. Pada tampilan halaman surat terdapat

informasi surat seperti tentang surat, nomor surat, waktu ditetapkan, jenis surat, serta *file* surat yang sudah di tambahkan. Pada halaman surat ini, *staff* dapat melakukan pengelolaan tambah data surat baru, hapus data surat, ubah data surat yang sudah ada, melihat *file* surat dimana jika kita mengklik *icon* melihat *file* surat maka *staff* akan di arahkan ke *google drive* dimana tempat *file* surat di simpan, dan pada halaman surat *staff* juga dapat melakukan pengunduhan *file* surat.

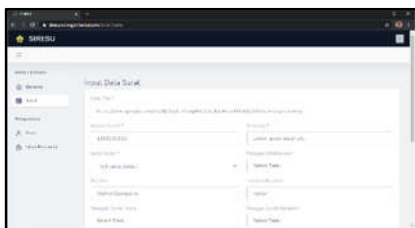
(3) Halaman *Input* Surat



Gambar 20. Tampilan Halaman *Input* Surat (*Staff*)

Gambar 20 adalah tampilan dari halaman *input* surat. Pada tampilan halaman *input* surat akan tampil tampilan berupa *form*. Pada *form* ini terdapat beberapa *field* yang dapat di isi seperti *link file*, nomor surat, tentang surat, jenis surat, tanggal ditetapkan, pejabat penandatanganan, jabatan pejabat, tanggal surat mulai, tanggal surat berakhir, penerima surat, dan deskripsi surat.

(4) Halaman *Input* Surat

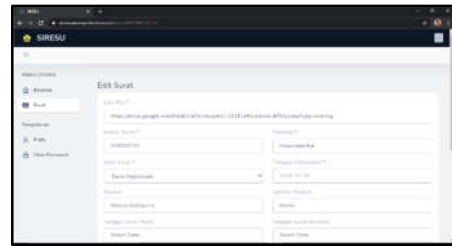


Gambar 21. Tampilan Halaman *Input* Surat (*Staff*)

Gambar 21 adalah tampilan dari halaman *input* surat. Pada tampilan halaman *input* surat akan tampil tampilan berupa *form*. Pada *form* ini terdapat beberapa *field* yang dapat di isi seperti *link file*, nomor surat, tentang surat, jenis surat, tanggal ditetapkan, pejabat penandatanganan, jabatan pejabat,

tanggal surat mulai, tanggal surat berakhir, penerima surat, dan deskripsi surat.

(5) Halaman *Edit* Surat

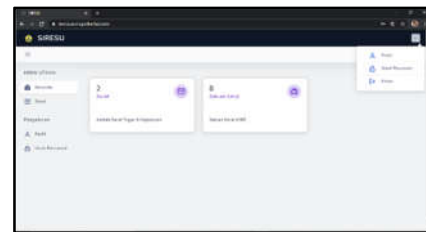


Gambar 22. Tampilan Halaman *Edit* Surat (*Staff*)

Gambar 22 adalah tampilan dari halaman *edit* surat. *Staff* dapat mengubah isi *field* yang ada sehingga data diperbaharui.

5.1.3 Implementasi *Interface* Sistem (Dosen dan *Staff* Biasa)

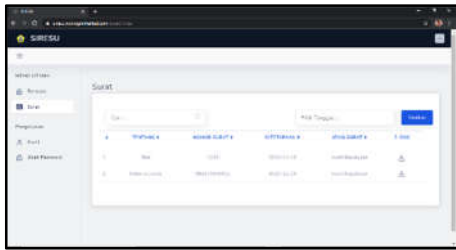
(1) Halaman Utama



Gambar 23. Tampilan Halaman Dashboard (*Staff* Biasa dan Dosen)

Gambar 23 merupakan tampilan dari halaman *dashboard*. Pada tampilan awal halaman *dashboard* terdapat informasi jumlah surat, dan jumlah satuan kerja. Pada halaman *dashboard* ini juga terdapat beberapa menu manajemen yang dapat di akses oleh *staff* biasa dan dosen seperti manajemen surat, manajemen *profil*, dan manajemen ubah *password*. Pada manajemen – manajemen tersebut *staff* biasa dan dosen dapat melakukan pengelolaan data.

(2) Halaman Surat



Gambar 24. Tampilan Halaman Surat (Staff Biasa dan Dosen)

Gambar 24 merupakan tampilan dari halaman surat. Pada tampilan halaman surat terdapat informasi surat seperti tentang surat, nomor surat, waktu ditetapkan, jenis surat, serta *file* surat yang sudah di tambahkan. Pada halaman surat ini, *staff* biasa dan dosen dapat melakukan pengelolaan melihat data surat dan pengunduhan *file* surat.

5.2 Pengujian *Black-Box Testing*

Dalam pembuatan sebuah sistem atau aplikasi diperlukannya adanya *software testing* untuk memeriksa apakah sistem atau aplikasi sudah berjalan dengan baik dan benar sehingga sistem

atau aplikasimemenuhi kriteria yang sudah ditentukan dan yang dibutuhkan oleh *user*. Dalam pembuatan sistem “ Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu ” menggunakan teknik pengujian yaitu *black-box testing*, dimana pengujian dilakukan didasarkan dengan tampilan aplikasi, fungsi – fungsi pada aplikasi, dan alur sistem seperti apa yang diinginkan. Pada sistem ini dilakukan pengujian dengan teknik *black-box testing* melalui beberapa aktor yaitu admin, *staff*, dosen, dan *staff* biasa.

5.2.1 Pengujian *Black-Box Testing* (Admin)

Pada Aplikasi yang sudah dibangun dilakukan pengujian sistem dengan teknik *black-box testing*, dimana pengujian dilakukan pada fungsi – fungsi yang memiliki prioritas tinggi saja. Pada pengujian sistem sebagai aktor admin menggunakan teknik *black-box testing* dijelaskan pada Tabel 5 sebagai berikut :

Table 5. Black-Box Testing Admin

No.	Pengujian	Skenario	Hasil Yang Di Harapkan	Hasil Pengujian
1.	Login	Memasukkan <i>username</i> dan <i>password</i> dengan benar	Sistem akan menerima <i>username</i> dan <i>password</i> kemudian menampilkan halaman utama	Berhasil
		Memasukkan <i>username</i> dan <i>password</i> dengan tidak benar	Sistem akan menolak akses <i>login</i> dengan menampilkan suatu keterangan “ Error! These credentials do not match our records“	Berhasil
		Memasukkan salah satu <i>username</i> atau <i>password</i> dengan tidak benar	Sistem akan menolak akses <i>login</i> dengan menampilkan suatu keterangan “ Error! These credentials do not match our records“	Berhasil
		Mengosongkan <i>username</i> atau <i>password</i>	Sistem akan menolak akses <i>login</i>	Berhasil
		Mengosongkan salah satu <i>username</i> atau <i>password</i>	Sistem akan menolak akses <i>login</i>	Berhasil
2.	Manajemen Surat	Tambah surat yang berisi <i>field – field</i> yang harus di isi sesuai dengan ketentuan pengisian untuk penambahan data surat	Sistem berhasil menambahkan data dengan menampilkan notifikasi bahwa surat berhasil dimasukkan	Berhasil
		Menghapus data surat	Sistem akan menampilkan notifikasi konfirmasi penghapusan data dan kemudian sistem berhasil menghapus data dengan menampilkan notifikasi surat berhasil dihapus	Berhasil
		Mengubah data surat	Sistem berhasil mengubah data dengan menampilkan notifikasi surat berhasil diubah	Berhasil
		Mengunduh data surat	Sistem berhasil mengunduh data dengan menampilkan bahwa data surat terunduh	Berhasil
3.	Manajemen <i>user</i>	Tambah data <i>user</i> yang berisi <i>field – field</i> yang harus di isi sesuai dengan ketentuan pengisian untuk	Sistem berhasil menambahkan data dengan menampilkan notifikasi bahwa data <i>user</i> berhasil ditambahkan	Berhasil

No.	Pengujian	Skenario	Hasil Yang Di Harapkan	Hasil Pengujian
		penambahan data <i>user</i>		
		Menonaktifkan data <i>user</i>	Sistem akan menampilkan notifikasi konfirmasi menonaktifkan data <i>user</i> dan kemudian sistem berhasil menonaktifkan data dengan menampilkan notifikasi data <i>user</i> berhasil dinonaktifkan	Berhasil
		Mengubah data <i>user</i>	Sistem berhasil mengubah data dengan menampilkan notifikasi data <i>user</i> berhasil diubah	Berhasil
4.	Manajemen Satuan Kerja	Tambah data satuan yang berisi <i>field – field</i> yang harus di isi sesuai dengan ketentuan pengisian untuk penambahan data satuan kerja	Sistem berhasil menampilkan form penambahan data satuan kerja dan berhasil menambahkan data dengan menampilkan notifikasi bahwa data berhasil ditambahkan	Berhasil
		Menghapus data satuan kerja	Sistem akan menampilkan notifikasi konfirmasi menghapus data satuan kerja dan kemudian sistem berhasil menghapus data dengan menampilkan notifikasi data berhasil dihapus	Berhasil
		Mengubah data satuan kerja	Sistem akan menampilkan notifikasi konfirmasi mengubah data satuan kerja dan kemudian sistem berhasil mengubah data dengan menampilkan notifikasi data berhasil diedit	Berhasil
5.	Manajemen profie	Mengubah data profie	sistem berhasil mengubah data dengan menampilkan notifikasi data berhasil diedit	Berhasil
6.	Manajemen ubah password	Mengubah password	sistem berhasil mengubah password dengan menampilkan notifikasi password berhasil diubah	Berhasil

5.2.2 Pengujian *Black-Box Testing* (*Staff*)

Dalam phase pengujian, Aplikasi Berbasis Web Untuk Manajemen Repository Surat juga diuji dengan dengan teknik *black-box testing*, dimana

pengujian dilakukan pada fungsi – fungsi yang memiliki prioritas tinggi saja. Pada pengujian sistem sebagai aktor *staff* menggunakan teknik *black-box testing* dijelaskan pada Tabel 6 sebagai berikut :

Table 6. Black-Box Testing Staff

No.	Pengujian	Skenario	Hasil Yang Di Harapkan	Hasil
1.	Login	Memasukkan username dan password dengan benar	Sistem akan menerima username dan password kemudian menampilkan halaman utama	Berhasil
		Memasukkan username dan password dengan tidak benar	Sistem akan menolak akses login dengan menampilkan suatu keterangan “Error! These credentials do not match our records“	Berhasil
		Memasukkan salah satu username atau password dengan tidak benar	Sistem akan menolak akses login dengan menampilkan suatu keterangan “Error! These credentials do not match our records“	Berhasil
		Mengosongkan username atau password	Sistem akan menolak akses login	Berhasil
		Mengosongkan salah satu username atau password	Sistem akan menolak akses login	Berhasil
2.	Manajemen Surat	Tambah surat yang berisi <i>field – field</i> yang harus di isi sesuai dengan ketentuan pengisian untuk penambahan data surat	Sistem berhasil menambahkan data dengan menampilkan notifikasi bahwa surat berhasil dimasukkan	Berhasil
		Menghapus data surat	Sistem akan menampilkan notifikasi konfirmasi penghapusan data dan kemudian sistem berhasil menghapus data dengan menampilkan notifikasi surat berhasil dihapus	Berhasil
		Mengubah data surat	Sistem berhasil mengubah data dengan menampilkan notifikasi surat berhasil diubah	Berhasil
		Mengunduh data surat	Sistem berhasil mengunduh data dengan menampilkan bahwa data surat terunduh	Berhasil
3.	Manajemen profie	Mengubah data profie	sistem berhasil mengubah data dengan menampilkan notifikasi data berhasil diedit	Berhasil

No.	Pengujian	Skenario	Hasil Yang Di Harapkan	Hasil
4.	Manajemen ubah password	Mengubah password	sistem berhasil mengubah password dengan menampilkan notifikasi password berhasil diubah	Berhasil

5.2.3 Pengujian *Black-Box Testing* (Dosen dan *Staff* Biasa)

Pada “Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu ” dilakukan pengujian sistem dengan teknik *black-box testing*, dimana pengujian

dilakukan pada fungsi – fungsi yang memiliki prioritas tinggi saja. Pada pengujian sistem sebagai aktor dosen dan *staff* biasa menggunakan teknik *black-box testing* dijelaskan pada Tabel 7 sebagai berikut :

Table 7. Black-Box Testing Dosen dan Staff Biasa

No.	Pengujian	Skenario	Hasil Yang Di Harapkan	Hasil
1.	Login	Memasukkan <i>username</i> dan <i>password</i> dengan benar	Sistem akan menerima <i>username</i> dan <i>password</i> kemudian menampilkan halaman utama	Berhasil
		Memasukkan <i>username</i> dan <i>password</i> dengan tidak benar	Sistem akan menolak akses <i>login</i> dengam menampilkan suatu keterangan “ Error! These credentials do not match our records“	Berhasil
		Memasukkan salah satu <i>username</i> atau <i>password</i> dengan tidak benar	Sistem akan menolak akses <i>login</i> dengam menampilkan suatu keterangan “ Error! These credentials do not match our records“	Berhasil
		Mengosongkan <i>username</i> atau <i>password</i>	Sistem akan menolak akses <i>login</i>	Berhasil
		Mengosongkan salah satu <i>username</i> atau <i>password</i>	Sistem akan menolak akses <i>login</i>	Berhasil
2.	Manajemen surat	Mengunduh data surat	Sistem berhasil mengunduh data dengan menampilkan bahwa data surat terunduh	Berhasil
3.	Manajemen <i>profie</i>	Mengubah data <i>profie</i>	sistem berhasil mengubah data dengan menampilkan notifikasi data berhasil diedit	Berhasil
4.	Manajemen ubah <i>password</i>	Mengubah <i>password</i>	sistem berhasil mengubah <i>password</i> dengan menampilkan notifikasi <i>password</i> berhasil diubah	Berhasil

VI. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari kegiatan penelitian dan pembahasan yang sudah dilakukan, diperoleh kesimpulan bahwa:

1. Telah dibangun sistem Aplikasi Berbasis Web Untuk Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu yang memberikan kemudahan dalam pendistribusian surat terhadap individu yang akan ditugaskan sehingga pendistribusian surat dapat dilakukan dalam bentuk *hardcopy* maupun *softcopy* di Fakultas Teknik Universitas Bengkulu, dari manual ke sistem yang sudah terkomputerisasi.
2. Sistem yang telah dibuat mampu menampilkan data data berupa data – data

pengguna seperti admin, *staff*, *staff* biasa, dan dosen, serta data surat.

B. Saran

Dari hasil analisis dan pengimplementasian Aplikasi Manajemen Repository Surat Fakultas Teknik Universitas Bengkulu ini, dapat di simpulkan bahwa sistem ini masih mempunyai kekurangan untuk menjadi saran pengembangan selanjutnya, yaitu :

1. Perlu adanya pemeliharaan dan pengembangan terhadap sistem agar dapat dipergunakan lebih baik lagi dan dipergunakan sesuai dengan kebutuhannya.
2. Perlunya dikembangkan sistem keamanan, agar web yang sudah disediakan dapat terjaga dari hal-hal yang tidak diinginkan.

REFERENSI

- [1] *Peraturan Pemerintah No 5 Tahun 2009 Tentang Dosen*, Republik Indonesia, 2009.
- [2] *Surat Keputusan Rektor Universitas Bengkulu No.2143/UN30/HK/2019*, Bengkulu: Universitas Bengkulu, 2019.
- [3] A. J. O'Brien and G. M. Marakas, *Analisis Sistem Informasi*, Tata Sutabri, Ed, 1, 2016.
- [4] G. A. A. Putri, "Rancang Bangun Sistem Informasi Persuratan Dan Kearsipan Universitas Udayana Menggunakan Paradigma Pemrograman Berorientasi Objek," *Majalah Ilmiah Teknik Elektro*, vol. 4, no. 2, p. 35-41, 2012.
- [5] N. W. Wisswani, I. G. N. B. C. Bawa and I. W. C. Winetra, "RECORD CENTER ARSIP SURAT KEPUTUSAN (SK) UNTUK UNIVERSITAS," *Jurnal Integrasi*, vol. 11, no. 1, pp. 1-8, 2019.
- [6] F. Rahman, "Rancang bangun sistem repository surat keputusan (SK) dengan memanfaatkan zkoss Framework (studi kasus: sub bagian organisasi tata laksana dan perundang-undangan Universitas Islam Negeri Syarif Hidayatullah Jakarta)," Universitas Islam Negeri Syarif Hidayatullah, Jakarta, 2011.
- [7] M. Khoirul, T. I. Tjendrowasono and B. K. Riasti, "Aplikasi pengelolaan data kearsipan pada Sekolah Menengah Atas Negeri 1 Mlonggo Jepara berbasis multiuser," *IJNS-Indonesian Journal on Networking and Security*, p. 2-11, 2013.
- [8] A. E. Ahmad, M. H. Koniyo and L. Hadjaratie, "Optimalisasi Pengelolaan Surat Keputusan Pada Fakultas Teknik Universitas Negeri Gorontalo," *Jurnal Teknik*, vol. 16, no. 1, p. 33-48, 2018.
- [9] Sarika, "Pengembangan sistem informasi manajemen surat keputusan studi kasus: subbagian administrasi kepegawaian pusat," UIN Syarif Hidayatullah, Jakarta, 2011.
- [10] E. A. K. Lestari, S. E. Anjarwani and N. Agitha, "Rancang Bangun Sistem Informasi Surat Perintah Perjalanan Dinas Pada Sekretariat Daerah Provinsi Berbasis Web," *Journal of Computer Science and Informatics Engineering (J-Cosine)*, vol. 2, no. 1, p. 7-16, 2018.
- [11] S. Lolong, "Sistem Pengarsipan Surat Keputusan Di Universitas Klabat," *CogITO Smart Journal*, vol. 4, no. 1, p. 11, 2018.
- [12] R. Zubaedah and N. P. Putra, "Website Pengarsipan Dokumen Dan Surat Pada Jurusan Sistem Informasi Universitas Musamus," *Mustek Anim Ha*, vol. 8, no. 1, p. 41-48, 2019.
- [13] H. O. L. Wijaya, "Implementasi Metode PIECES Pada Analisis Website Kantor Penanaman Modal Kota Lubuklinggau," *JUSIM (Jurnal Sistem Informasi Musirawas)*, p. 50, 2018.
- [14] H. O. L. Wijaya, "Implementasi Metode PIECES Pada Analisis Website Kantor Penanaman Modal Kota Lubuklinggau," *JUSIM (Jurnal Sistem Informasi Musirawas)*, p. 50, 2018.