

SKRIPSI

**APLIKASI PERSURATAN AKADEMIK UNIVERSITAS
FAJAR MAKASSAR BERBASIS *WEB***



MUHAMMAD RIFQY MURJANI
1520221042

PROGRAM STUDI TEKNIK ELKTRO
FAKULTAS TEKNIK
UNIVERSITAS FAJAR
MAKASSAR
2021

HALAMAN PENGESAHAN

Aplikasi Persuratan Akademik Universitas Fajar Makassar Berbasis *Web*

Disusun Oleh :

MUH. RIFQY MURJANI
1520221042

Telah diperiksa dan disetujui oleh Dosen Pembimbing

Makassar, 14 Februari 2022

Pembimbing I



Safaruddin, S.Si., MT
NIDN. 0909106901

Pembimbing II



Asmawaty Azis, ST., MT.
NIDN. 0905058504

Mengetahui,

Dekan Fakultas Teknik



Prof. Dr. Ir. Erniati, S.T., M.T
NIDN. 0906107701

Plt. Ketua Program Studi



Faris Jusawan, ST., MT.
NIDN. 0914038603

PERNYATAAN ORISINALITAS

Yang bertanda tangan dibawah ini :

Nama : Muh. Rifqy Murjani

Stambuk : 1520221042

Program Studi : Teknik Elektro

Menyatakan dengan sebenar – benarnya bahwa tugas akhir yang saya tulis ini benar – benar merupakan hasil karya saya sendiri, bukan merupakan pengabilan tulisan atau pemikiran orang lain. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan tugas akhir ini hasil karya orang lain, saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 30 November 2022

Yang Menyatakan,



Muh. Rifqy Murjani

KATA PENGANTAR

Segala puji syukur Alhamdulillah, dengan ini tidak putus-putusnya penulis panjatkan kepada ALLAH Subhanahu Wa Ta'ala yang telah melimpahkan segala rahmat serta hidayah-Nya, tak lupa Shalawat dan Salam, senantiasa tercurahkan kepada junjungan dari suri tauladan Nabi Muhammad Shallallahu 'Alaihi Wa Sallam, sehingga penulis dapat menyelesaikan Skripsi ini yang berjudul "**Aplikasi Persuratan Akademik Universitas Fajar Makassar Berbasis Web**". Laporan skripsi ini disusun sebagai salah satu syarat bagi mahasiswa Program Studi Teknik Elektro Universitas Fajar Makassar untuk dapat menyelesaikan studi Program Strata Satu (S1). Selama penyusunan tugas akhir ini saya banyak mendapat bantuan dari berbagai pihak, maka perkenankan saya menggunakan kesempatan ini untuk mengucapkan terima kasih serta penghargaan setinggi-tingginya kepada:

1. Allah Subhanahu Wa Ta'ala atas rahmat dan hidayah-Nya yang tidak terputus kepada kami.
2. Ayah dan ibu serta keluarga besar yang senantiasa mendoakan agar kami selalu diberi kekuatan dan kesabaran.
3. Ibu Dr. Erniati ST.,MT selaku Dekan Fakultas Teknik Universitas Fajar Makassar.
4. Ibu Asmawaty Azis ST.,MT selaku Ketua Prodi Teknik Elektro Universitas Fajar Makassar
5. Bapak Safaruddin S.Si.,MT selaku Dosen Pembimbing I.
6. Ibu Asmawaty Azis ST.,MT selaku Dosen Pembimbing II.
7. Dosen-dosen Program Studi Teknik Elektro Universitas Fajar Makassar
8. Rekan-rekan Mahasiswa Universitas Fajar di Fakultas Teknik, terkhusus Mahasiswa Program Studi Elektro Fakultas Teknik Universitas Fajar Makassar yang selalu memberikan bantuan dan masukan, serta semua pihak yang telah membantu sehingga Proposal Skripsi ini dapat terselesaikan.

Akhirnya, kepada Allah Subhanahu Wa Ta'ala juga penulis kembalikan permasalahan-permasalahan yang berada diluar kemampuan. Semoga Allah Subhanahu Wa Ta'ala senantiasa melimpahkan rahmat dan hidayah-Nya kepada kami atas usaha, perjuangan dan pengorbanan yang dilakukan.

Makassar, Oktober 2021

Penulis

ABSTRAK

Muhammad Rifqy Murjani“Aplikasi Persuratan Akademik Universitas Fajar Makassar Berbasis *WEB*”. Sistem informasi bisa diartikan menjadi sebuah sistem yang terintegrasi secara optimal serta berbasis komputer yang bisa menghimpun serta menyajikan berbagai jenis data yang seksama buat berbagai macam kebutuhan. Pengelolaan arsip memiliki peran penting bagi jalannya suatu organisasi maupun instansi, yaitu sebagai sumber informasi dan salah satu bagian penting dalam administrasi persuratan di suatu organisasi dan instansi, yang akan bermanfaat untuk bahan pengambilan keputusan atau penyusunan program pengembangan dari organisasi maupun instansi. Berdasarkan permasalahan tersebut maka dalam penelitian ini dibuat Aplikasi Persuratan Akademik Universitas Fajar Makassar Berbasis *WEB*, sehingga seluruh proses peminjaman dilakukan berbasis teknologi informasi. Aplikasi ini dibuat menggunakan *Framework Codeigniter 4* dengan bahasa pemrograman PHP dan Basis Data *Mysql*. Berdasarkan hasil pengujian *Whitebox*, semua logika program telah benar. Hasil pengujian *Blackbox* menyatakan semua fungsional aplikasi telah berfungsi, sedangkan berdasarkan hasil analisis efektifitas pengguna menyatakan berhasil dan layak diterapkan. Dengan adanya aplikasi ini maka memudahkan seluruh pihak dalam proses manajemen persuratan kampus Universitas Fajar Makassar.

Kata Kunci : Aplikasi, Sistem Informasi, Persuratan

ABSTRACT

Muhammad Rifqy Murjani“**Aplikasi Persuratan Akademik Universitas Fajar Makassar Berbasis WEB**”. *An information system can be interpreted as an optimally integrated and computerized system that can collect and present different types of accurate data for different needs. Records management plays an important role in the running of an organization or agency, namely as a source of information and one of the important parts in the management of correspondence in an organization and agency, which will be useful for decision-making materials or the preparation of development programs of organizations and agencies. Based on these problems, a WEB-based academic letter application was submitted in this study at Fajar Makassar University, so that the entire credit process was carried out using information technology. This application is created with the Codeigniter 4 framework with the programming language PHP and the MySQL database. Based on the results of the whitebox test, all of the program logic is correct. The results of the black box test showed that all functional applications had worked, while the results of the user effectiveness analysis determined that it was successful and actionable. With this application, it makes it easier for all parties to manage the Fajar Makassar University campus correspondence management process.*

Keywords: Applications, Information Systems, Mail

DAFTAR ISI

HALAMAN JUDUL	
HALAMAN PERSETUJUAN.....	i
KATA PENGANTAR	ii
ABSTRAK.....	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR	viii
DAFTAR TABEL.....	x
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	2
1.4 Batasan Masalah.....	2
1.5 Manfaat Penelitian.....	3
BAB 2 TINJAUAN PUSTAKA.....	4
2.1 Tinjauan Teori.....	4
2.1.1 Sistem Informasi	4
2.1.2 Perancangan Sistem	5
2.1.3 Informasi	5
2.1.4 Surat	5
2.1.5 <i>Website</i>	5
2.1.6 <i>Framework</i>	5
2.1.7 <i>Codeigniter</i>	5
2.1.8 Basis Data.....	6
2.1.9 XAMPP	6
2.1.10 HTML (<i>Hyper Text Markup Language</i>)	6
2.1.11 PHP (<i>Hypertext Preprocessor</i>)	7
2.1.12 MYSQL (<i>My Structured Query Language</i>)	7
2.1.13 <i>Flowchart</i>	8
2.1.14 UML (<i>Unified Modelling Language</i>).....	9
2.1.15 <i>The Waterfall Model</i>	14

2.1.16 Pengujian Sistem	15
2.2. Penelitian Terdahulu (<i>State of The Art</i>)	16
2.3. Kerangka pikir.....	20
BAB 3 METODE PENELITIAN	21
3.1 Prosedur Penelitian.....	21
3.2 Rancangan Penelitian/Sistem	23
3.3 Waktu dan Tempat Penelitian	33
3.4 Alat dan Bahan	34
3.5 Metode Pengumpulan Data	34
3.6 Metode Pengujian.....	35
BAB 4 HASIL DAN PEMBAHASAN	36
4.1 Hasil Penelitian.....	36
4.2 Pembahasan	45
BAB 5 PENUTUP	59
5.1 Kesimpulan.....	59
5.2 Saran.....	60
DAFTAR PUSTAKA	61

DAFTAR GAMBAR

Gambar 2.1 <i>Model Waterfall</i>	15
Gambar 3.1 Diagram Alir Penelitian.....	21
Gambar 3.2 <i>Use Case Admin</i>	23
Gambar 3.3 <i>Use Case Validator</i>	23
Gambar 3.4 <i>Use Case User</i>	23
Gambar 3.5 <i>Actifity Diagram</i> Buat Akun Baru	24
Gambar 3.6 <i>Actifity Diagram</i> Login.....	25
Gambar 3.7 <i>Actifity Diagram</i> Mengisi Data Persuratan.....	26
Gambar 3.8 <i>Actifity Diagram</i> Validasi.....	27
Gambar 3.9 <i>Sequence Diagram</i> Buat Akun Baru	27
Gambar 3.10 <i>Sequence Diagram</i> Login	28
Gambar 3.11 <i>Sequence Diagram</i> Mengisi Data Persuratan.....	28
Gambar 3.12 <i>Class Diagram</i>	29
Gambar 3.13 <i>Menu Login</i> Pada Sistem Informasi Persuratan	30
Gambar 3.14 Halaman Utama <i>Admin</i>	30
Gambar 3.15 Tampilan <i>Menu Home</i>	31
Gambar 3.16 Tampilan <i>Menu Surat</i>	31
Gambar 3.17 Contoh Tampilan Memilih <i>Menu Surat</i> Kerja Praktek	32
Gambar 3.18 Tampilan <i>Menu Validasi</i>	33
Gambar 3.19 Tampilan <i>Form Koreksi</i>	33
Gambar 4.1 Tampilan <i>Login</i>	36
Gambar 4.2 <i>Script Login</i>	37
Gambar 4.3 Tampilan Beranda/ <i>Dashboard</i>	37
Gambar 4.4 <i>Script</i> Tampilan Beranda/ <i>Dashboard</i>	38
Gambar 4.5 Tampilan Pengajuan Surat.....	38
Gambar 4.6 <i>Script</i> Pengajuan Surat	39
Gambar 4.7 Tampilan <i>Check</i> Status Surat.....	39
Gambar 4.8 <i>Script</i> <i>Check</i> Status Surat	40
Gambar 4.9 Tampilan Master Surat	40
Gambar 4.10 <i>Script</i> Master Surat.....	41
Gambar 4.11 Tampilan Manajemen Persuratan.....	41

Gambar 4.12 <i>Script</i> Manajemen Surat	42
Gambar 4.13 Tampilan Master Surat	42
Gambar 4.14 <i>Script</i> Master Surat.....	43
Gambar 4.15 Tampilan Manajemen <i>User</i>	43
Gambar 4.16 <i>Script</i> Manajemen <i>User</i>	44
Gambar 4.17 Tampilan <i>Setting</i>	44
Gambar 4.18 <i>Script Setting</i>	45
Gambar 4.19 Diagram <i>Flowchart Whitebox Testing Login</i>	49
Gambar 4.20 Diagram <i>Flowgraph Whitebox Testing Login</i>	49
Gambar 4.21 Diagram <i>Flowchart Whitebox Testing Login</i>	51
Gambar 4.22 Diagram <i>Flowgraph Whitebox Testing Login</i>	51
Gambar 4.23 Diagram <i>Flowchart Whitebox Testing</i> Tambah Data Format Surat.....	53
Gambar 4.24 Diagram <i>Flowgraph Whitebox Testing</i> Tambah Data Format Surat.....	53
Gambar 4.25 Diagram <i>Flowchart Whitebox Testing</i> Tambah <i>User</i>	55
Gambar 4.26 Diagram <i>Flowgraph Whitebox Testing</i> Tambah <i>User</i>	55
Gambar 4.27 Diagram <i>Flowchart Whitebox Testing</i> Tambah Pengajuan Surat.....	57
Gambar 4.28 Diagram <i>Flowgraph Whitebox Testing</i> Tambah Pengajuan Surat.....	57

DAFTAR TABEL

Tabel 2.1 Simbol-simbol dalam <i>Flowchart</i>	8
Tabel 2.2 Simbol-simbol <i>Use Case Diagram</i>	10
Tabel 2.3 Simbol-simbol <i>Actifity Diagram</i>	11
Tabel 2.4 Simbol-simbol <i>Sequence Diagram</i>	12
Tabel 2.5 Simbol-simbol <i>Class Diagram</i>	13
Tabel 2.6 Penelitian Terdahulu	16
Tabel 4.1 Instrumen Pengujian Fungsional Aplikasi.....	45
Tabel 4.2 <i>Test Case Login</i>	50
Tabel 4.3 <i>Test Case</i> Buat Akun Baru	52
Tabel 4.4 <i>Test Case</i> Tambah Data Format Surat	54
Tabel 4.5 <i>Test Case</i> Tambah <i>User</i>	56
Tabel 4.6 <i>Test Case</i> Tambah Pengajuan Surat	58

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Perkembangan dunia sistem informasi pada saat ini sudah sedemikian pesat dan merambah ke berbagai sisi kehidupan manusia. Perkembangan dapat terlihat dalam hal manajemen persuratan di beberapa instansi. Saat ini prosedur diterapkan masih secara manual dalam hal administrasi persuratan yakni pembuatan dan penyimpanan surat.

Pengelolaan file mempunyai kiprah penting bagi jalannya suatu organisasi maupun instansi, yaitu menjadi sumber informasi serta salah satu bagian krusial dalam administrasi persuratan pada suatu organisasi dan instansi, yang akan berguna buat bahan pengambilan keputusan atau penyusunan acara pengembangan dari organisasi juga instansi.

Begitu pula manajemen persuratan yang berlangsung di Fakultas Teknik Universitas Fajar Makassar masih secara manual dan menggunakan sistem konvensional. Dimana pembuatan surat masih menggunakan aplikasi *Microsoft Office Word*, staf administrasi masih mengetik nomor dan isi surat secara manual, kemudian surat yang telah dicetak diarsipkan.

Adapun penelitian yang dilakukan oleh Nirmasari dan Rahmat Firdaus mengacu pada sistem persuratan Universitas Fajar Makassar yang diperuntukkan untuk Program Studi Teknik Elektro. Penelitian tersebut bertujuan untuk surat-surat yang berkaitan dengan Program Studi Elektro dapat disampaikan melalui *softcopy*. Sistem kerja yang dilakukan ialah pengisian ataupun pembuatan surat akademik disampaikan melalui suatu yang berbasis *WEB* (sistem informasi persuratan Universitas Fajar Makassar).

Berdasarkan permasalahan yang dijelaskan, maka peneliti akan membuat sebuah sistem informasi persuratan yang berjudul “**Aplikasi Persuratan Akademik Universitas Fajar Makassar Berbasis WEB**”. Penelitian ini diharapkan mampu memberikan informasi secara cepat,

tepat dan akurat dalam hal administrasi persuratan Fakultas Teknik Universitas Fajar Makassar.

1.2 Rumusan Masalah

Dari uraian di atas maka dapat di rumuskan beberapa masalah yang timbul:

1. Bagaimana membuat dan merancang suatu sistem informasi persuratan berbasis *WEB* di Fakultas Teknik Universitas Fajar Makassar ?
2. Bagaimana menganalisa dan menguji aplikasi persuratan akademik di Fakultas Teknik Universitas Fajar berbasis *WEB* ?

1.3 Tujuan Penelitian

Adapun tujuan penelitian ini adalah

1. Membuat dan merancang sistem informasi persuratan berbasis *WEB* di Fakultas Teknik Universitas Fajar Makassar sehingga dapat memudahkan dalam pencarian, pendataan dan pembuatan arsip-arsip persuratan.
2. Menganalisa dan menguji aplikasi persuratan akademik di Fakultas Teknik Universitas Fajar berbasis *WEB*.

1.4 Batasan Masalah

Untuk menghindari meluasnya permasalahan, maka masalah dalam penelitian ini dibatasi. Adapun batasan masalah yang akan dibahas adalah:

1. Data persuratan dalam penelitian ini adalah data-data surat akademik yang ada pada Fakultas Teknik Universitas Fajar Makassar.
2. Aplikasi ini hanya untuk Mahasiswa Fakultas Teknik Universitas Fajar Makassar untuk mengelola persuratan akademik di Fakultas Teknik Universitas Fajar Makassar.

1.5 Manfaat Penelitian

Penelitian ini diharapkan dapat bermanfaat bagi penulis, mahasiswa, dan beberapa civitas akademika di Fakultas Teknik Universitas Fajar Makassar.

Adapun manfaat dari penelitian ini yaitu :

1. Bagi penulis bermanfaat sebagai penerapan ilmu dalam bidang informatika yang telah di dapat selama perkuliahan yang berhubungan dengan manfaat teknologi otomatis pada kehidupan sehari-hari.
2. Memudahkan mahasiswa untuk membuat persuratan akademik di Fakultas Teknik Universitas Fajar Makassar.
3. Memudahkan pihak administrasi persuratan akademik Fakultas Teknik Universitas Fajar Makassar dalam mengelolah laporan persuratan.

BAB II

LANDASAN TEORI

2.1. Tinjauan Teori

2.1.1 Sistem Informasi

Sistem informasi dapat diartikan sebagai sistem komputer yang terintegrasi secara optimal yang dapat mengumpulkan dan menyajikan berbagai jenis data yang akurat untuk kebutuhan yang berbeda (Juansyah, 2013)

2.1.2 Perancangan Sistem

Perancangan sistem merupakan fase setelah fase analisis sistem, yang menggambarkan bagaimana sistem dibangun, mendefinisikan kebutuhan fungsional dan bagaimana desain disiapkan untuk pembuatan program aplikasi (Jogiyanto H.M, 2001)

Dengan demikian perancangan sistem dapat diartikan sebagai berikut :

1. Tahap setelah analisis dari siklus pengembangan sistem.
2. Pendefinisian dari kebutuhan-kebutuhan fungsional.
3. Persiapan untuk rancang bangun implementasi.
4. Menggambarkan bagaimana suatu sistem dibentuk.
5. Dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi.

2.1.3 Informasi

Informasi adalah data yang telah diklasifikasikan, diproses atau ditafsirkan untuk pengambilan keputusan. Sistem penanganan informasi mengubah data menjadi informasi, atau lebih tepatnya, pengolahan data dari bentuk yang tidak berguna sehingga bermanfaat bagi penerimanya. Nilai informasi tergantung pada keputusan. Jika tidak ada pilihan atau solusi, informasi menjadi tidak perlu (Sutabri, 2005)

2.1.4 Surat

Surat adalah sarana atau alat untuk komunikasi dalam menyampaikan informasi dalam bentuk tulisan dari satu pihak yang ditujukan untuk pihak yang lainnya dalam kegiatan bisnis atau non bisnis (Djoko Purwanto, 2011)

2.1.5 Website

World Wide Web (WWW) adalah kumpulan halaman *web* domain yang berisi informasi. Sebuah situs *web* biasanya terdiri dari banyak halaman *web* yang bekerja sama. Tautan antara satu halaman *web* dengan halaman *web* lainnya disebut *hyperlink*, dan teks yang digunakan sebagai tautan disebut *hypertext* (Yuhefizar, 2009)

2.1.6 Framework

Framework secara sederhana dapat diartikan sebagai sekumpulan fungsi/prosedur dan kelas untuk tujuan tertentu yang siap digunakan untuk mempermudah dan mempercepat pekerjaan *programmer* tanpa harus membuat fungsi atau kelas dari awal.

Framework juga dapat diartikan sebagai rangkaian *script* (khusus kelas dan fungsi yang dapat membantu *developer/programmer* memecahkan berbagai masalah pemrograman seperti koneksi *database*, pemanggilan variabel dan *file*, sehingga *developer* lebih fokus dan dapat membangun aplikasi dengan lebih cepat (Wildan Mualim dan Gema Ulama Putra, 2017)

2.1.7 Codeigniter

Codeigniter adalah aplikasi *open source* sebagai *framework* atau kerangka kerja untuk membuat *website* dengan menggunakan bahasa pemrograman PHP. Tujuannya adalah untuk memungkinkan proyek berkembang lebih cepat daripada menulis kode dasar atau terstruktur dengan menyediakan banyak perpustakaan yang biasa digunakan dalam pengembangan. (Sulistiono, 2018)

2.1.8 Basis Data

Basis data adalah serangkaian *file* yang dihubungkan bersama, tautannya biasanya ditunjukkan dengan kunci dari setiap arsip yang ada. *Database* menyediakan formasi data yang digunakan dalam area informasi. *File* berisi *record* dengan tipe, ukuran, dan bentuk yang sama yang mewakili satu urutan entitas. Sebuah *record* terdiri dari *field-field* yang bekerja sama untuk menunjukkan bahwa *field* tersebut dalam arti penuh dari kata dan akan dicatat dalam *record*. Sistem manajemen basis data berisi satu set data terkait dan satu set program untuk mengakses data tersebut. Ini adalah sistem manajemen basis data dan serangkaian program manajemen untuk menambahkan data, menghapus data, mengekstrak data, dan membaca data (Mhd Bustanur Rahmad, Tedy Setiady, 2014)

2.1.9 XAMPP

XAMPP adalah *software web server Apache* yang terintegrasi melalui *MySQL* dan *PHPMyadmin*. XAMPP adalah singkatan dari X, *Apache Server*, *MySQL*, *PHPMyadmin* dan *Python*. X di bagian depan berarti XAMPP dapat diinstal pada berbagai macam sistem operasi (Dadan dan Kerendi Developers, 2015)

2.1.10 HTML (*Hyper Text Markup Language*)

HTML adalah singkatan dari *Hypertext Markup Language*, bahasa *web* standar yang dipelihara oleh pengguna *World Wide Web Consortium* (W3C) dalam bentuk *tag* yang membentuk setiap elemen situs *web*. HTML berperan dalam membangun struktur halaman *website*, menempatkan setiap elemen *website* sesuai dengan *layout* yang diinginkan. HTML biasanya disimpan dalam format *file* .html. Bentuk paling sederhana untuk memasukkan skrip HTML adalah dengan menggunakan *editor teks* seperti *Notepad* atau *editor teks* khusus yang mengenali setiap elemen skrip HTML dan menampilkannya dalam warna yang berbeda untuk keterbacaan yang lebih baik, seperti *Notepad++*, *Sublime Text* dan banyak lainnya. Program serupa (Abdulloh, 2013)

2.1.11 PHP (*Hypertext Preprocessor*)

Hypertext Preprocessor (PHP) adalah bahasa pemrograman untuk membuat situs *web* yang dinamis dan interaktif. Dinamis artinya sebuah *website* dapat mengubah tampilan dan isinya dalam kondisi tertentu. misalnya PHP dapat mengganti tanggal dan hari saat ini di situs *web*. Alat interaktif: PHP dapat memberikan umpan balik kepada pengguna (misalnya menunjukkan apa yang akan terjadi di lini produk) (Enterprise, 2018)

PHP adalah singkatan dari *Hypertext Preprocessor*, bahasa *scripting* tingkat tinggi yang diinstal ke dalam dokumen HTML. Sintaks PHP sebagian besar mirip dengan C, *Java*, dan *Perl*, tetapi PHP memiliki beberapa fungsi yang lebih spesifik. Namun, tujuan utama penggunaan bahasa ini adalah untuk menyediakan desain *web* dinamis yang bekerja secara otomatis.

Preprocessor hypertext bukanlah bahasa pemrograman yang perlu Anda gunakan untuk membuat halaman *web*. Kami hanya dapat membangun situs *web* dengan HTML. *Web* yang dibangun dengan HTML (dan CSS) dikenal sebagai situs *web* statis yang mengumpulkan konten dan halaman *web*. Sebagai perbandingan, situs *web* dinamis yang dapat dibuat dengan *preprosesor hypertext* adalah situs *web* yang dapat menyesuaikan tampilan konten tergantung pada situasinya. Situs *web* dinamis juga menyimpan data dalam *database*, membuat halaman yang berubah berdasarkan input pengguna, formulir proses, dan banyak lagi. Aneka ragam. Untuk membuat *website*, biasanya kode PHP disisipkan ke dalam dokumen HTML, karena PHP juga dikenal sebagai bahasa *scripting* atau bahasa pemrograman *script* berkat fitur ini (Setiawan, 2018)

2.1.12 MYSQL (*My Structured Query Language*)

MySQL adalah salah satu program yang menjalankan fungsi *Database Management System* (DBMS). MySQL digunakan untuk mengelola data *website*. MySQL dapat digunakan oleh banyak pengguna secara bersamaan dan dapat memproses *query* sederhana dengan cepat. Mendukung perintah

berupa *query* dan dapat menampilkan data melalui (*Application Programming Interface* (API) (Saputra dkk, 2018)

2.1.13 Flowchart

Flowchart adalah urutan langkah-langkah kerja dalam suatu proses yang digambarkan dengan simbol-simbol yang tersusun secara sistematis (Iswandy, 2015)

Secara formal, diagram blok didefinisikan sebagai representasi skema dari suatu algoritma atau proses. Tabel berikut menunjukkan simbol-simbol yang digunakan untuk membuat diagram alur/*flowchart* (Harumy, dkk, 2016)

Tabel 2.1 Simbol-simbol Dalam *Flowchart*

	Terminator Sebagai simbol ' <i>START</i> ' atau ' <i>END</i> ' untuk memulai atau mengakhiri <i>flowchart</i> .
	Input/Output Digunakan untuk menuliskan proses menerima data atau mengeluarkan data.
	Proses Digunakan untuk menuliskan proses yang diperlukan, misalnya operasi aritmatika.
	Conditional/Decision Digunakan untuk menyatakan proses yang membutuhkan keputusan.
	Preparation Digunakan untuk memberikan nilai awal.
	Arrow Sebagai penunjuk arah dan alur proses.
	Connector (On-page) Digunakan untuk menyatukan arrow.

	<p>Connector (Off-page) Digunakan untuk menghubungkan <i>flowchart</i> yang harus digambarkan pada halaman yang berbeda. Biasanya pada simbol ini diberi nomor sebagai penanda, misalnya angka 1.</p>
	<p>Display Digunakan untuk menampilkan data ke monitor.</p>

2.1.14 UML (*Unified Modelling Language*)

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

a) *Use Case Diagram*

Use Case Diagram merupakan model dari perilaku sistem informasi yang sedang dibangun. *Use case* bekerja dengan menggambarkan interaksi khas antara pengguna sistem dan sistem itu sendiri, dengan mengatakan bagaimana sistem digunakan.

b) *Class Diagram*

Class Diagram adalah ilustrasi struktur sistem berasal segi pendefinisian kelas-kelas yang akan didesain untuk membuat sistem. *class diagram* terdiri dari atribut serta operasi dengan tujuan membuat program bisa menghasilkan hubungan antara dokumentasi perancangan serta *software* yang sinkron.

c) *Sequence Diagram*

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Gambaran *sequence diagram* dibuat minimal sebanyak pendefinisian *use case* yang

memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *sequence diagram* sehingga semakin banyak *use case* yang didefinisikan, maka *sequence diagram* yang harus dibuat juga semakin banyak.

d) *Activity Diagram*

Activity Diagram merupakan diagram yang menggambarkan *workflow* atau aktivitas dari sebuah sistem yang ada pada perangkat lunak (Putra, 2019)

Adapun 4 jenis diagram yang digunakan yaitu : (Heriyanto, 2018)

1. *Use Case Diagram*

Use Case Diagram adalah sesuatu atau proses merepresentasikan hal-hal yang dapat dilakukan oleh aktor dalam menyelesaikan sebuah pekerjaan. Berikut beberapa simbol pada *use case*:

Tabel 2.2 Simbol *Use Case*

Simbol	Deskripsi
	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor, biasanya
	Menghubungkan <i>link</i> antar <i>element</i>
	Disebut juga <i>inheritance</i> (pewarisan), sebuah elemen <u>dapat merupakan</u> spesialisasi dari elemen lainnya.
	Disebut juga <i>inheritance</i> (pewarisan), sebuah elemen <u>dapat merupakan</u> spesialisasi dari elemen lainnya.
	Yaitu kelakuan yang harus terpenuhi agar sebuah <i>event</i> dapat terjadi, dimana pada kondisi ini sebuah <i>use case</i> adalah bagian dari <i>use case</i> lainnya

2. Activity Diagram

Activity Diagram mendeskripsikan *work flow* (sirkulasi kerja atau kegiatan dari sebuah sistem atau proses usaha. yang perlu diperhatikan disini ialah bahwa diagram aktivitas mendeskripsikan kegiatan sistem bukan apa yang dilakukan aktor, jadi aktivitas bisa dilakukan sistem. Berikut simbol-simbol yang terdapat di *activity diagram*:

Tabel 2.3 Simbol *Activity Diagram*

Simbol	Deskripsi
	Untuk menyatakan awal dari proses
	Untuk menyatakan akhir dari suatu proses

	Digunakan untuk menyatakan kondisi dari suatu proses
	Menyatakan aksi yang dilakukan dalam suatu arsitektur sistem

3. *Sequence Diagram*

Sequence Diagram ialah *tool* yang sangat terkenal pada pengembangan sistem informasi secara *object-oriented* untuk menampilkan hubungan antar objek. Berikut simbol-simbol yang terdapat di *sequence diagram*:

Tabel 2.4 Simbol *Sequence Diagram*

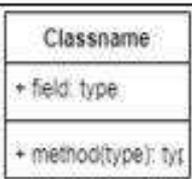
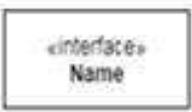
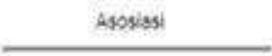
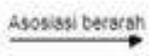
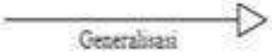
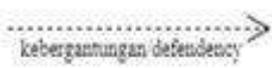
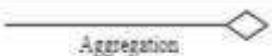
Simbol	Deskripsi
	Mengidentifikasi keberadaan sebuah objek dalam basis waktu notasi untuk <i>lifeline</i> adalah garis putus-putus <i>vertical</i> yang di Tarik dari sebuah objek
Activation 	Dinotasikan sebagai sebuah segiempat yang di gambar pada sebuah <i>lifeline</i> . Mengidentifikasi sebuah objek yang akan melaksanakan sebuah aksi
	Digambarkan dengan anak panah <i>horizontal</i> <u>antara</u> . <i>Message</i> mengindikasikan komunikasi antara objek-objek
	Adalah komponen berbentuk kotak yang mewakili sebuah <i>class</i> atau <i>object</i> . Mereka

	mendemonstrasikan bagaimana sebuah <i>object</i> berperilaku pada sebuah sistem
 Actor	Aktor juga dapat berkomunikasi

4. Class Diagram

Class Diagram mendeskripsikan struktur sistem dari segi pendefinisian kelas-kelas yang akan dirancang untuk membentuk sistem. Kelas mempunyai apa yang disebut atribut serta metode atau operasi. Atribut adalah variabel-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode merupakan fungsi-fungsi yang dimiliki oleh suatu kelas. Berikut simbol-simbol yang terdapat pada class diagram:

Tabel 2.5 Simbol *Class Diagram*

Simbol	Deskripsi
	Kelas pada struktur sistem
	Sama dengan konsep <i>interface</i> pada program berorientasi objek
	Kelas antar kelas dengan makna umum, asosiasi biasa di sertai dengan <i>multiplicity</i>
	Kelas antar kelas dengan makna kelas yang satu digunakan oleh kelas yang
	Kelas antar kelas dengan makna generalisasi spesialisasi (umum-khusus)
	Kelas antar kelas dengan makna kebergantungan antar kelas
	Kelas antar kelas dengan makna semua bagian

2.1.15 *The Waterfall Model*

Tahapan-tahapan *The Waterfall Model* (Pressman, 2010)

1) Tahapan Pemeriksaan

Dilakukan untuk mengetahui apakah ada masalah atau memungkinkan untuk mengembangkan sistem informasi. Pada tahap ini harus dilakukan studi kelayakan untuk menentukan apakah sistem informasi yang dikembangkan merupakan solusi yang *workable*.

2) Tahapan Analisis

Tujuannya adalah untuk memperjelas kebutuhan pengguna dan organisasi dan untuk menganalisis persyaratan yang ada (sebelum memperkenalkan sistem informasi baru).

3) Tahapan Desain

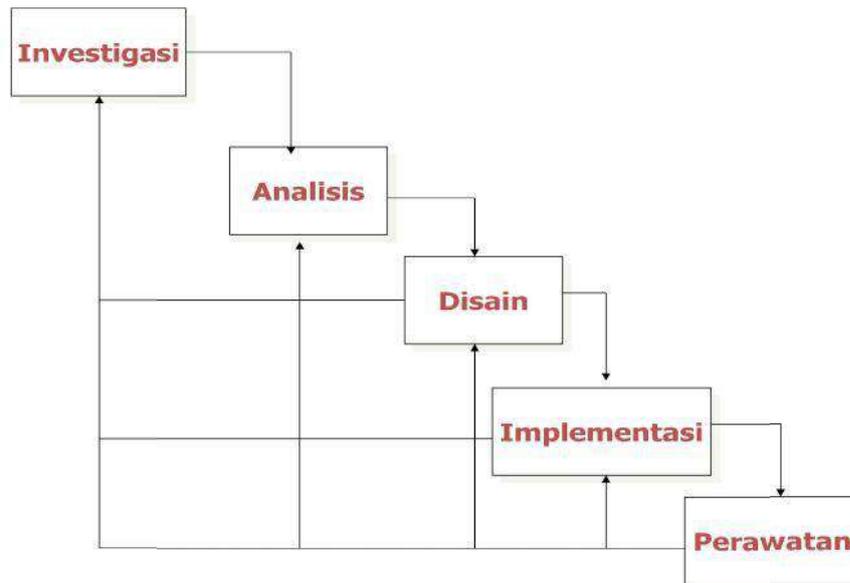
Tujuannya adalah untuk memilih spesifikasi yang lebih detail untuk komponen sistem informasi (manusia, perangkat keras, perangkat lunak, jaringan dan data) serta produk informasi yang disinkronkan dengan hasil tahap analisis.

4) Tahapan Implementasi

Ialah fase memperoleh atau mengembangkan peralatan dan aplikasi (program pengkodean), pengujian, pelatihan, dan transisi ke sistem baru.

5) Tahapan Perawatan

Dilakukan ketika sistem informasi telah dioperasikan



Gambar 2.1 Model Waterfall

2.1.16 Pengujian Sistem

Black Box Testing artinya teknik pengujian aplikasi yang berfokus pada spesifikasi fungsional dari aplikasi. *Black Box Testing* bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya difokuskan di info domain. *Black Box Testing* memungkinkan pengembang perangkat lunak untuk menghasilkan himpunan kondisi input yang akan melatih semua syarat-syarat fungsional suatu program.

Keuntungan penggunaan *Black Box Testing* ialah :

- a) Penguji tak perlu mempunyai pengetahuan perihal bahasa pemrograman tertentu
- b) Pengujian dilakukan berasal sudut pandang pengguna, ini membantu untuk menyampaikan ambiguitas atau inkonsistensi pada spesifikasi persyaratan
- c) Programmer serta tester keduanya saling bergantung satu sama lain

Kekurangan dari metode *Black Box Testing* merupakan :

- a) Uji perkara sulit disain tanpa spesifikasi yang jelas
- b) Kemungkinan mempunyai pengulangan tes yang telah dilakukan oleh programmer

- c) Beberapa bagian *back end* tidak diuji sama sekali. saat ini ada banyak metoda atau teknik untuk melaksanakan *Black Box Testing*, diantaranya : *Equivalence class Partitioning, Boundary Value Analysis, Error Guessy*

White Box Testing artinya pengujian yang memperhitungkan prosedur internal sistem atau komponen. *White Box Testing* pula dikenal sebagai *structural testing, clear-box testing, serta glass-box testing*.

Konotasi *clear-box* atau *glass box* menandakan bahwa kita mempunyai visibilitas penuh akan kerja internal *software*, terutama pada logika dan struktur kode. Adapun kelebihan metode ini artinya bisa mendeteksi kesalahan logika, kesalahan ketik pada kode sumber dan ketidaksesuaian asumsi. sementara kekurangannya ialah *software* yang tergolong besar , disebut sebagai metode yang boros sebab melibatkan sumber daya yang besar (Tri Sandhika Jaya, 2018)

II.2 Penelitian Terdahulu (*State of The Art*)

Tabel 2.6 Penelitian Terdahulu

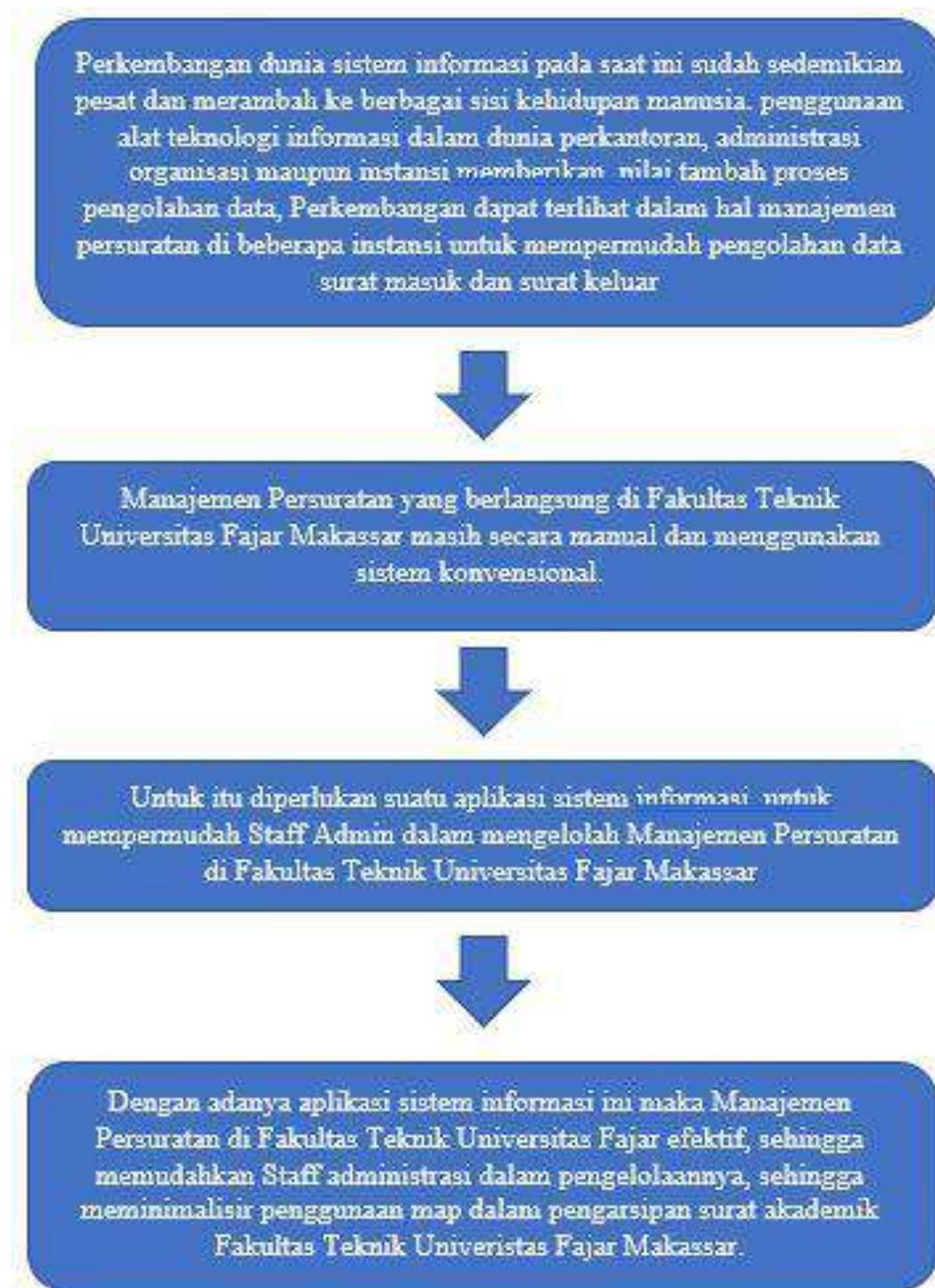
No	Nama Peneliti	Judul Penelitian	Metode	Hasil/Output
1.	Atika Fitriani (2016)	RANCANG BANGUN SISTEM ADMINISTRASI PERSURATAN DIGITAL MOBILE PADA FAKULTAS SAINS DAN TEKNOLOGI UIN ALAUDDIN MAKASSAR	Design and Creation	Merancang dan membangun Sistem Administrasi persuratan berbasis digital mobile untuk mempercepat penyelesaian surat dan mempermudah mahasiswa dalam mendapatkan

				informasi manajemen persuratan di Fakultas Sains dan Teknologi UIN Alauddin Makassar
2.	1. Wahyu Bagus Pratama 2. Ali Tarmuji (2015)	RANCANG BANGUN SISTEM INFORMASI PERSURATAN DAN PENGARSIPAN BERBASIS WEB PADA PG GONDANG BARU KLATEN	Prototyping	Proses implementasi dari rancangan sistem yang telah dibuat ke dalam perangkat lunak agar dapat digunakan oleh user. Tahapan ini melakukan komputerisasi dari sistem persuratan dan pengarsipan manual diubah dengan menggunakan bahasa pemrograman Framework CodeIgniter atau berbasis web

3.	Rita Lestari (2016)	RANCANG BANGUN SISTEM INFORMASI MANAJEMEN KEARSIPAN PADA BADAN PERIZINAN TERPADU DAN PENANAMAN MODAL KOTA MAKASSAR	Kualitatif	Merancang dan membangun sistem manajemen kearsipan berbasis web untuk mengefisienkan waktu dalam pencarian, pendataan, dan arsip-arsip surat di Badan Perizinan Terpadu dan Penanaman Modal kota Makassar.
4.	Edu Arif Rahman Hakim (2015)	RANCANG BANGUN SISTEM INFORMASI PERSURATAN PELANGGARAN KODE ETIK PENYELENGGARA PEMILU (STUDI KASUS : DEWAN KEHORMATAN PENYELENGGARA PEMILU (DKPP))	Pengembangan Sistem	Merancang dan membangun sistem persuratan berbasis web agar memberikan kemudahan dalam menyampaikan informasi persuratan dan penindak lanjutan permintaan persuratan kepada pihak

				tersangkut dari bagian Pengaduan dan atau bagian Persidangan pada Tata Usaha di Dewan Kehormatan Penyelenggara Pemilu (DKPP)
5.	Nurhayati (2011)	PENGEMBANGAN SISTEM INFORMASI LAYANAN PERSURATAN (STUDI KASUS KANTOR KELURAHAN BAMBU APUS KOTA TANGERANG SELATAN)	Analisis Sistem dan Desain Berorientasi Objek	Pengembangan Sstem penulis menggunakan metode Analisis dan Desain Berorientasi Objek (<i>Object Oriented Analysis and Design/OOAD</i>) dan alat yang digunakan yaitu UML (<i>Unified Modeling Language</i>) dengan <i>Use Case Diagram, Actiftiy Diagram, Class Diagram, Sequence Diagram</i>

II.3 Kerangka Pikir

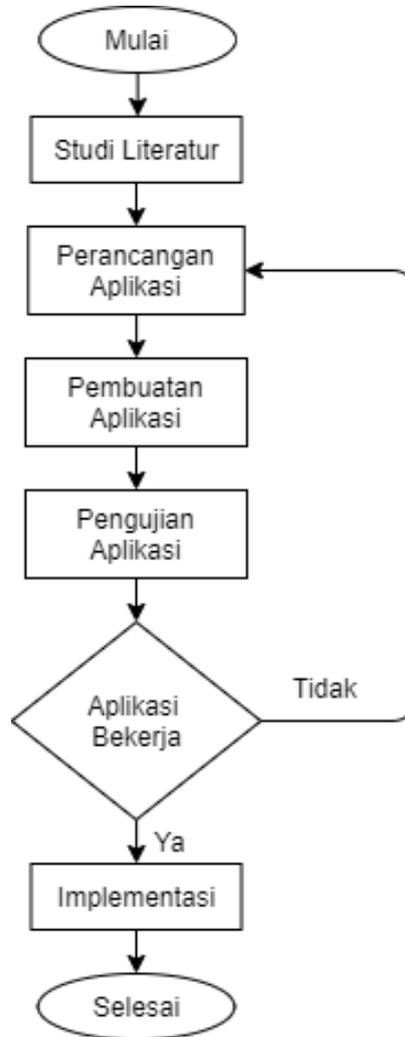


BAB III

METODOLOGI PENELITIAN

3.1 Prosedur Penelitian

Tahapan yang dilakukan dalam penelitian ini adalah sebagai berikut:



Gambar 3.1 Diagram Alir Penelitian

Dari diagram alir penelitian di atas, dijelaskan bahwa:

1. Mulai

Penelitian ini dimulai untuk memberi kemudahan bagi *Staff Admin* dalam mengelolah manajemen persuratsn Fakultas Teknik Universitas Fajar Makassar.

2. Studi Literatur

Tahap ini merupakan teknik penelusuran jurnal dan informasi-informasi di internet dalam rangka mencari keterangan-keterangan yang terkait dengan penelitian.

3. Perancangan Aplikasi

Pada tahap ini dilakukan perancangan yang akan dibuat melalui studi literatur dan sesuai data yang telah dirangkum untuk mewujudkan suatu rancangan yang sesuai dengan kebutuhan.

4. Pembuatan Aplikasi

Pada tahap ini dilakukan pembuatan aplikasi sesuai dengan perancangan yang telah dilakukan.

5. Pengujian Aplikasi

Pengujian aplikasi dilakukan setelah semua perangkat lunak diintegrasikan menjadi satu aplikasi. Pengujian ini dilakukan dengan menggunakan *Black Box Testing* dan *White Box Testing*.

6. Implementasi

Pada tahap ini sistem sudah siap digunakan apabila hasil dari semua pengujian menyatakan bahwa sistem yang telah dibangun berjalan sesuai fungsinya sehingga dapat digunakan di manajemen persuratan di Fakultas Teknik Universitas Fajar Makassar.

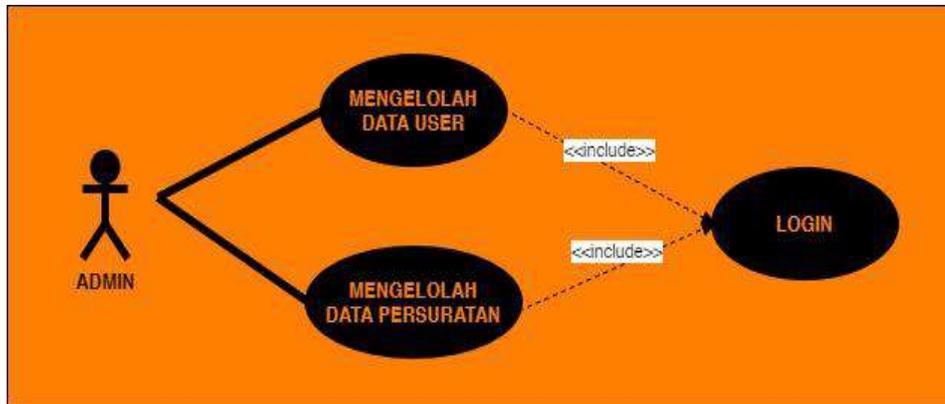
7. Selesai

Akhir dari penelitian dan pembuatan laporan akhir.

3.2 Rancangan Penelitian/Rancangan Sistem

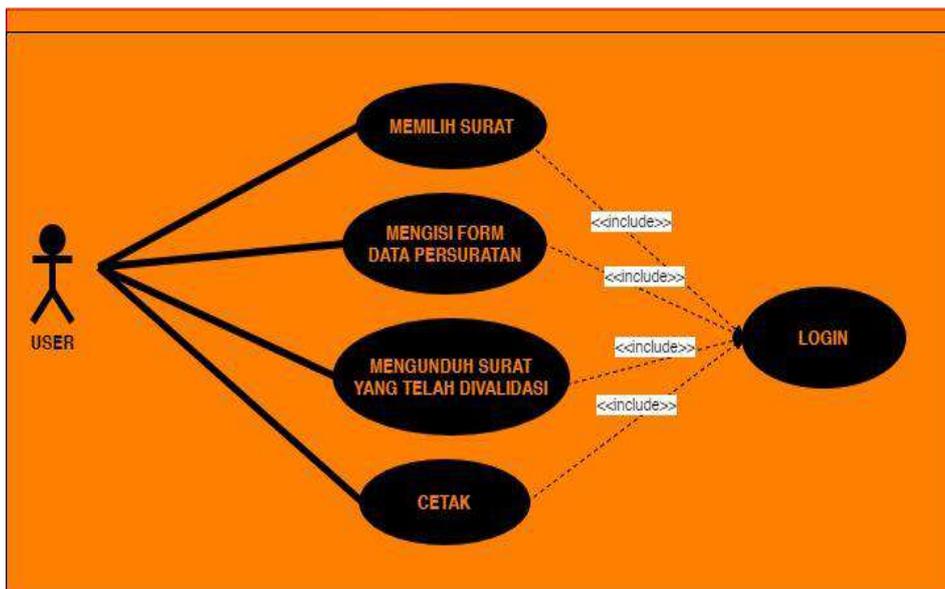
3.2.1 Unified Modelling Language (UML)

3.2.1.1 Use Case Diagram



Gambar 3.2 Use Case Admin

Gambar 3.3 Use Case Validator

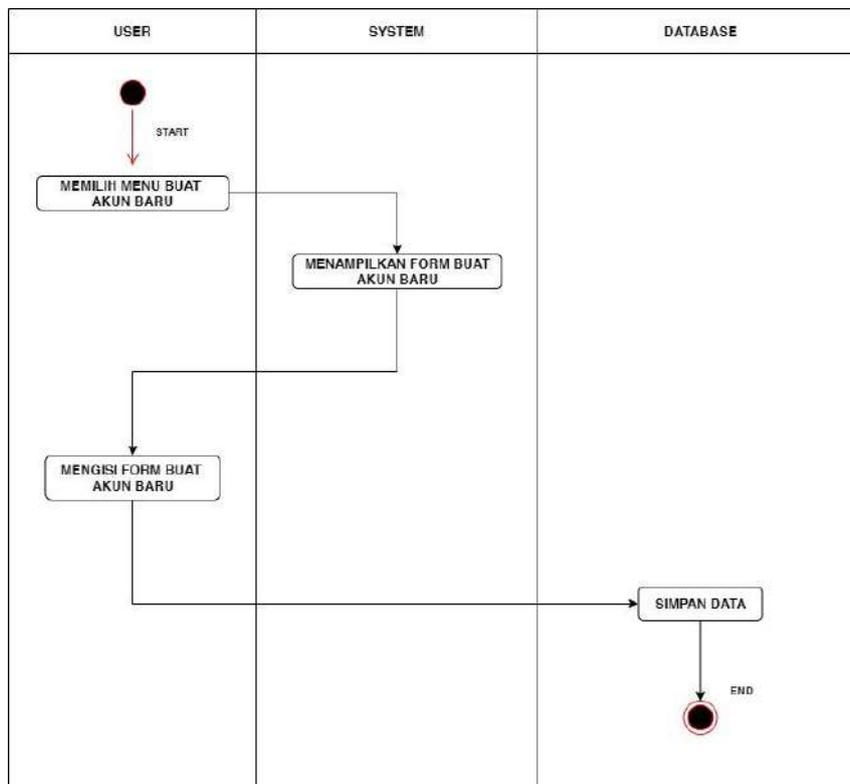


Gambar 3.4 Use Case User

Pada *Use Case* diatas, *actornya* ialah Admin, Dekan Fakultas, Prodi-prodi, sebagai *validator* dan Mahasiswa sebagai *user*. Tugas-tugas dari setiap *actor* berbeda-beda, dan dicantumkan pada *use case* yang ada. Tetapi, sebelum mereka bisa melakukan tugas tersebut ada *include* yang mengharuskan mereka untuk *login* kedalam *system*.

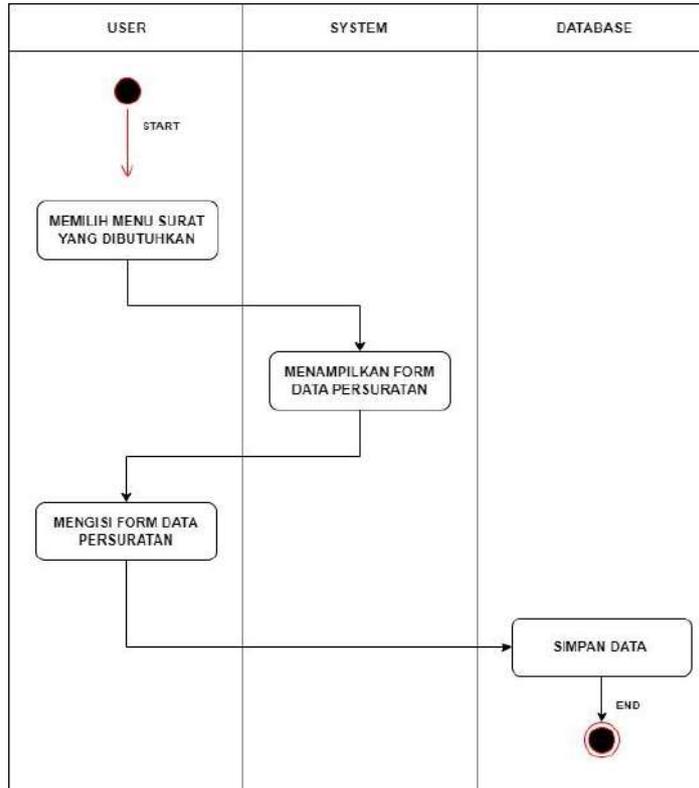
3.2.1.2 Activity Diagram

1. Activity Diagram Buat Akun Baru



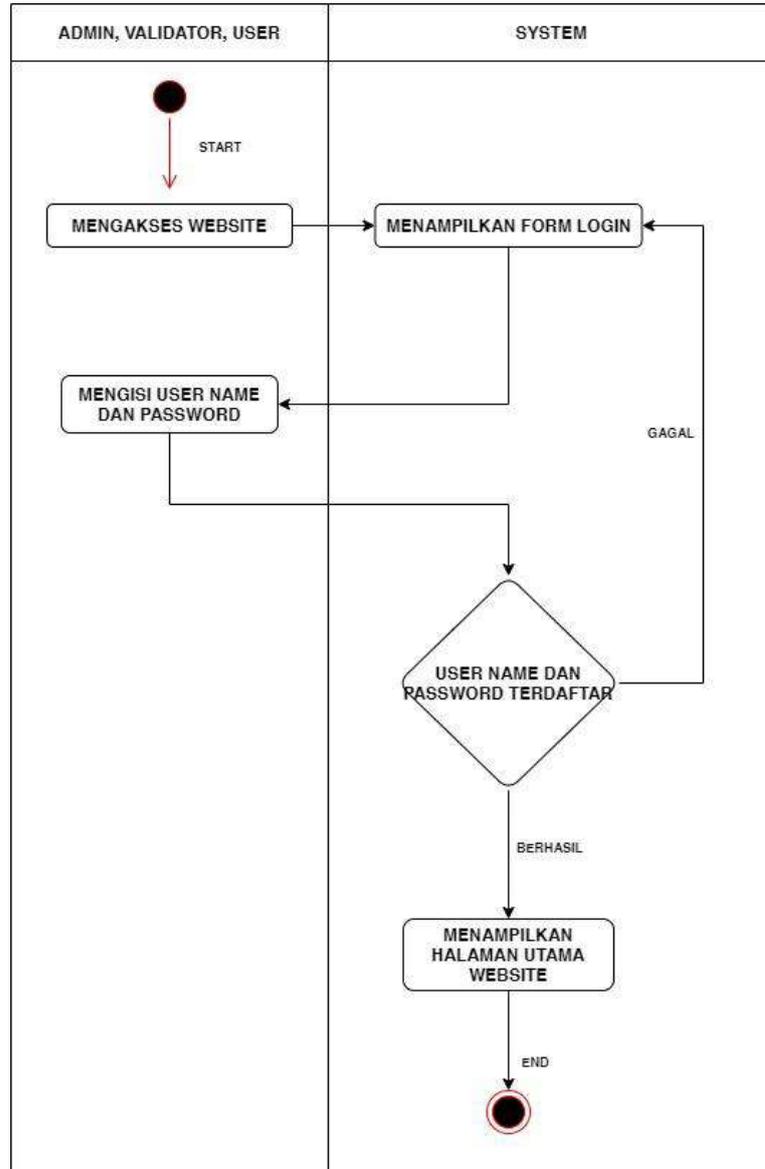
Gambar 3.5 Activity Diagram Buat Akun Baru

1. Activity Diagram Buat Mengisi Data Formulir*



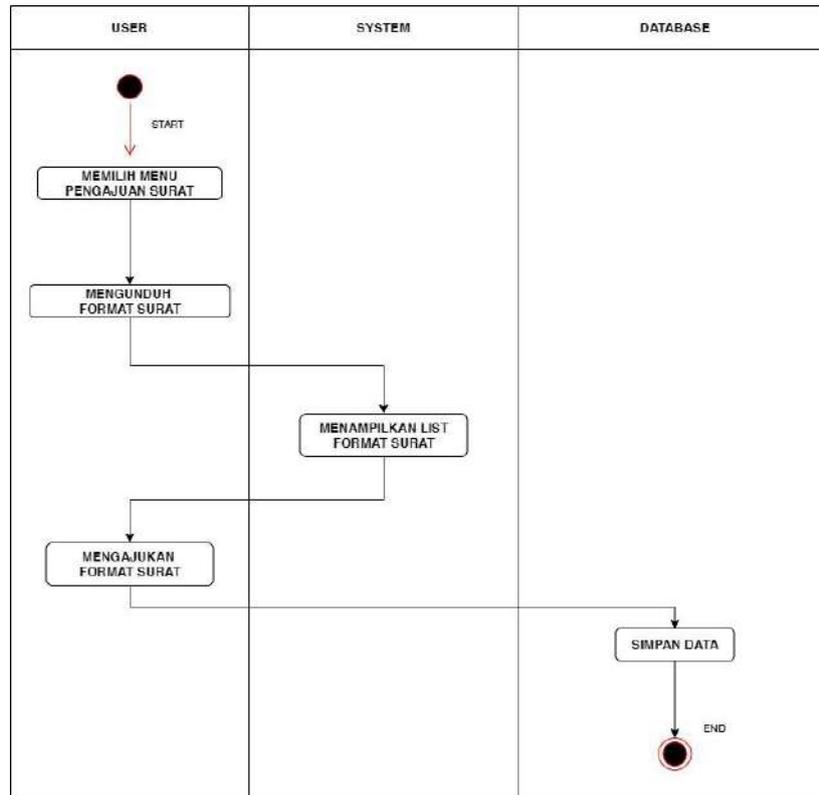
Gambar 3.5 Activity Diagram Buat Mengisi Data Formulir*

2. Activity Diagram Login



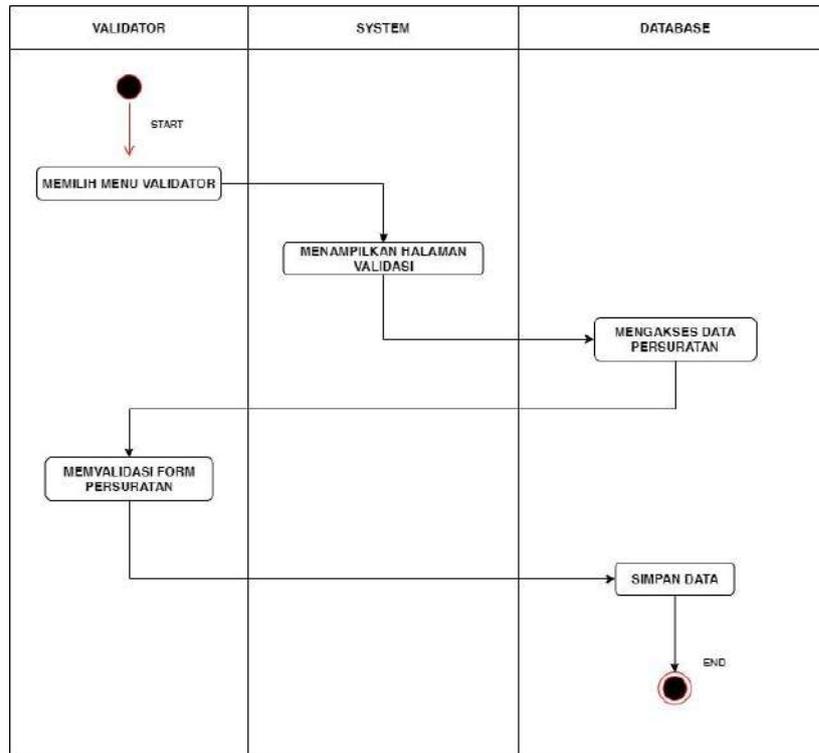
Gambar 3.6 Activity Diagram Login

3. Activity Diagram Mengisi Data Persuratan



Gambar 3.7 Activity Diagram Mengisi Data Persuratan

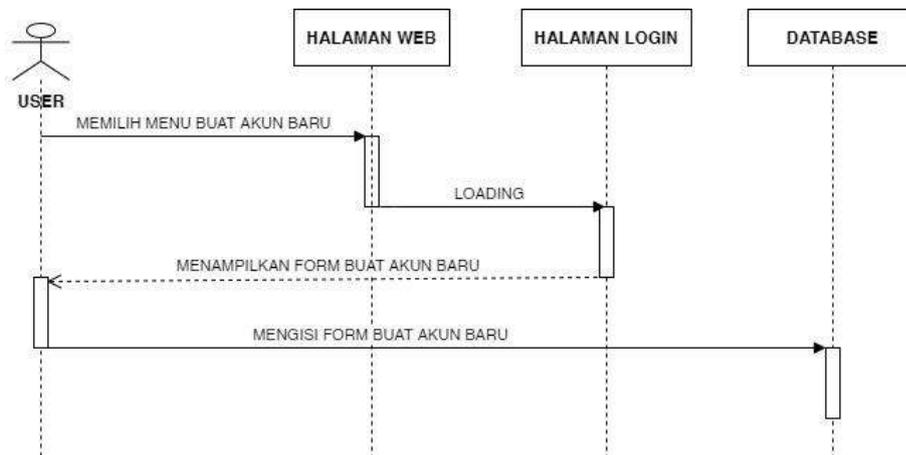
4. Activity Diagram Validasi



Gambar 3.8 Activity Diagram Validasi

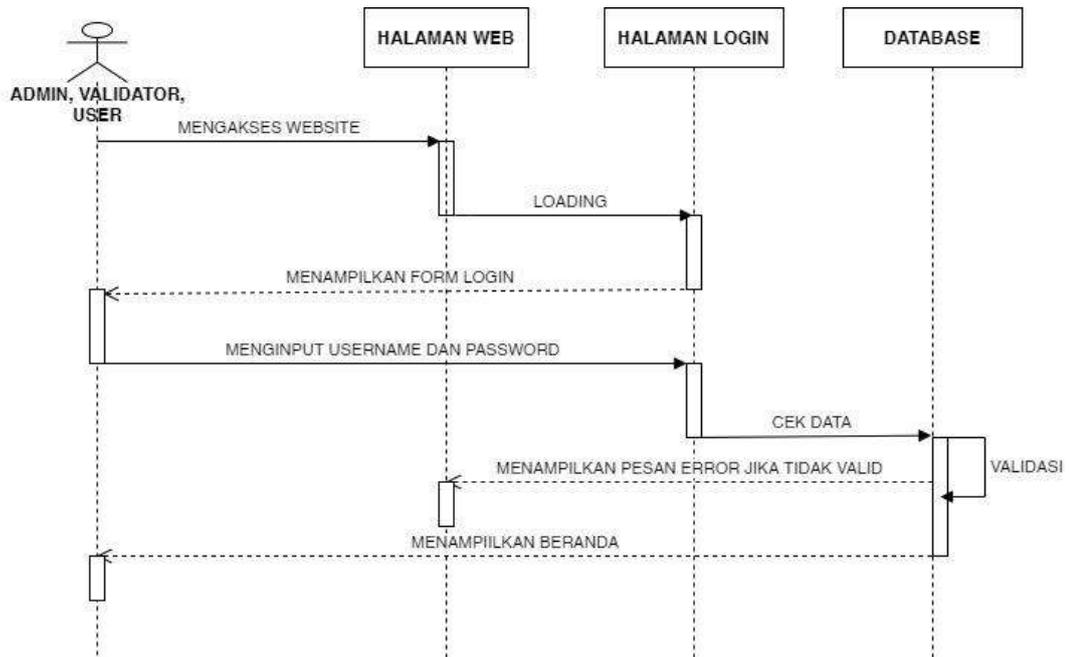
3.2.1.3 Sequence Diagram

1. Sequence Diagram Buat Akun Baru



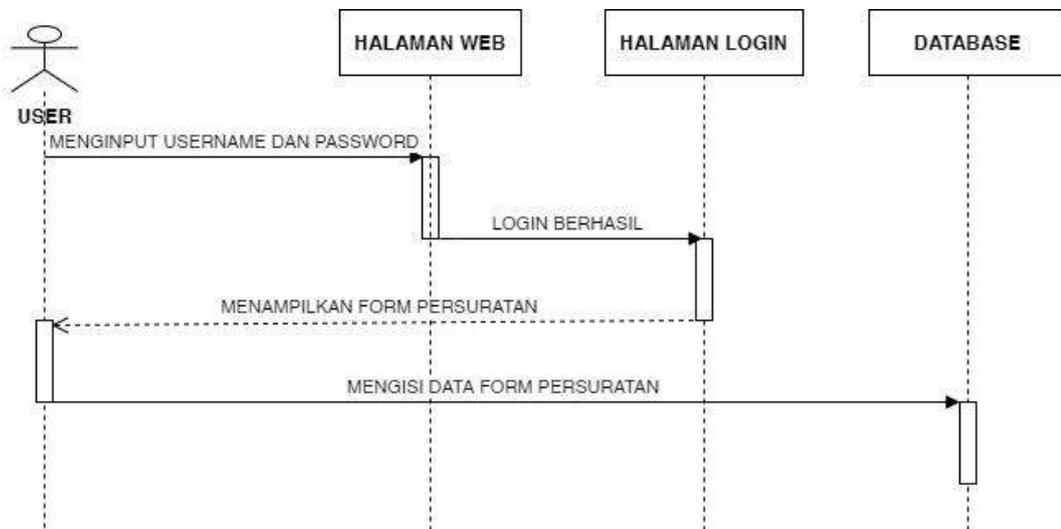
Gambar 3.9 Sequence Diagram Buat Akun Baru

2. Sequence Diagram Login



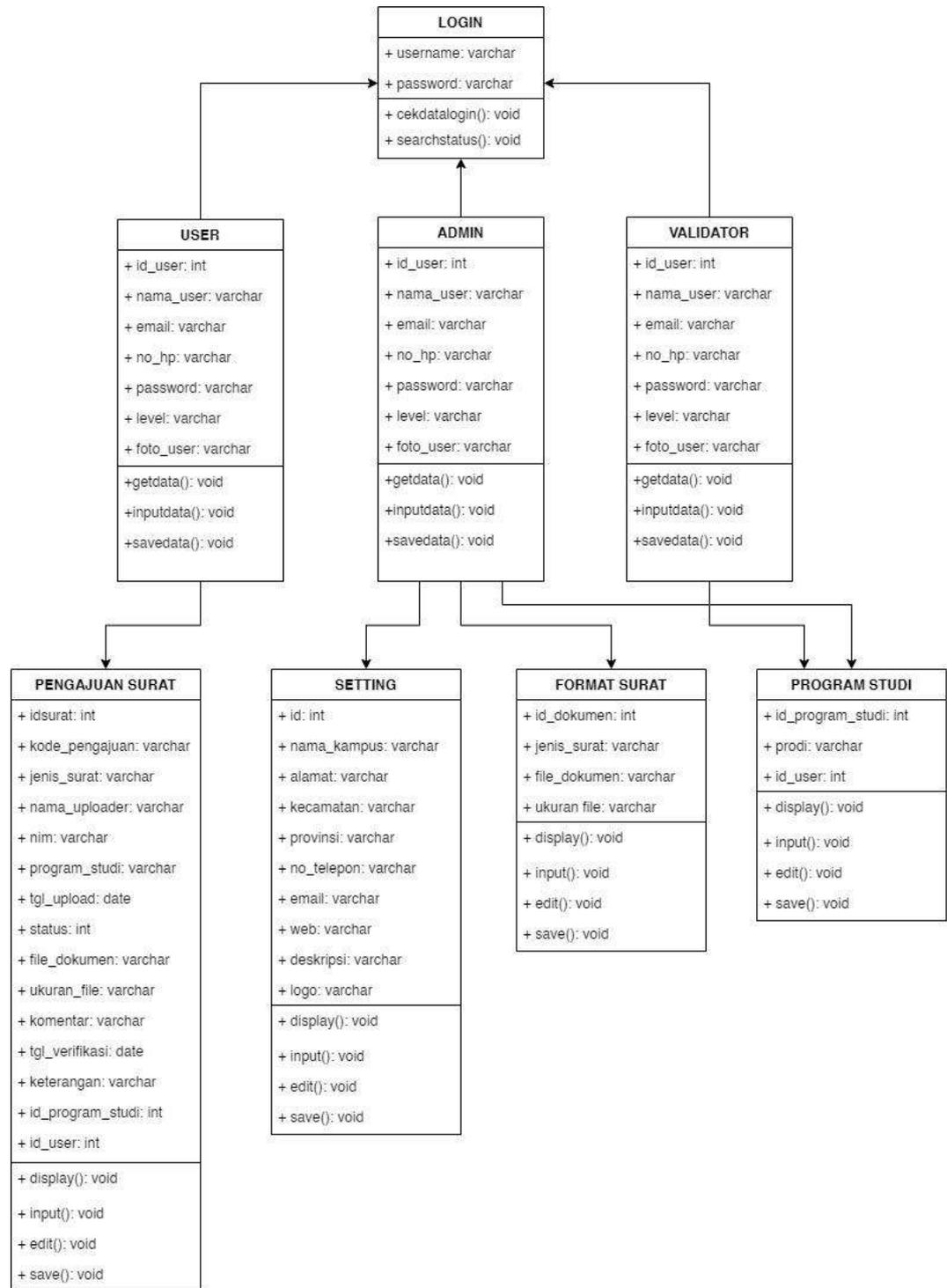
Gambar 3.10 Sequence Diagram Login

3. Sequence Diagram Mengisi Data Persuratan



Gambar 3.11 Sequence Diagram Mengisi Data Persuratan

3.2.1.4 Class Diagram



Gambar 3.12 Class Diagram

3.2.2 Desain Tampilan Awal

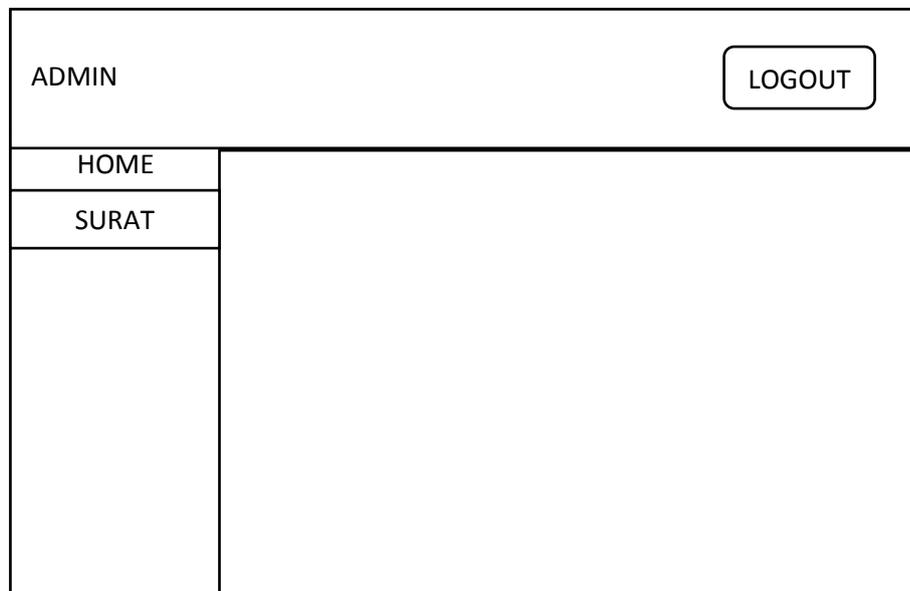


SISTEM INFORMASI PERSURATAN FAKULTAS TEKNIK
UNIVERSITAS FAJAR MAKASSAR

Username :

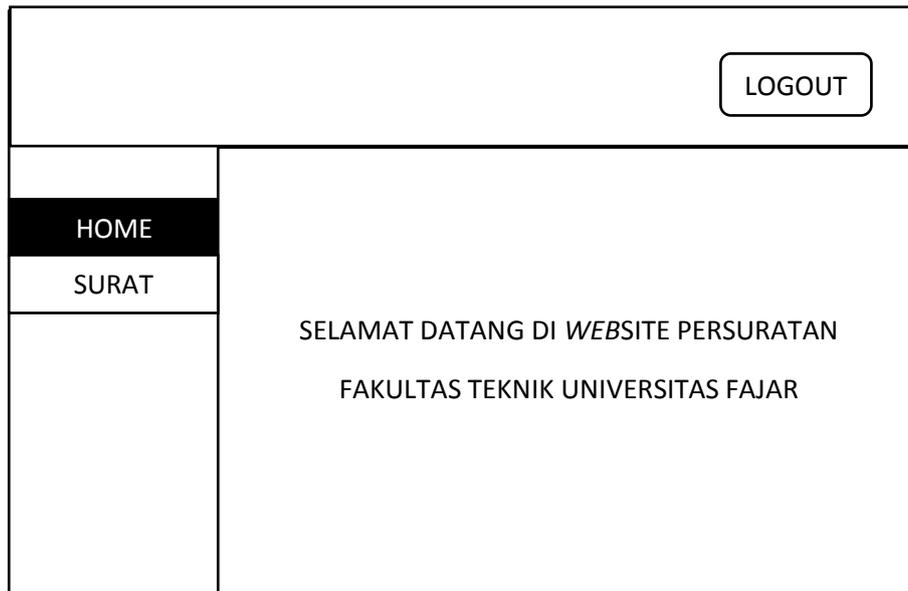
Password :

Gambar 3.13 Menu Login Pada Sistem Informasi Persuratan

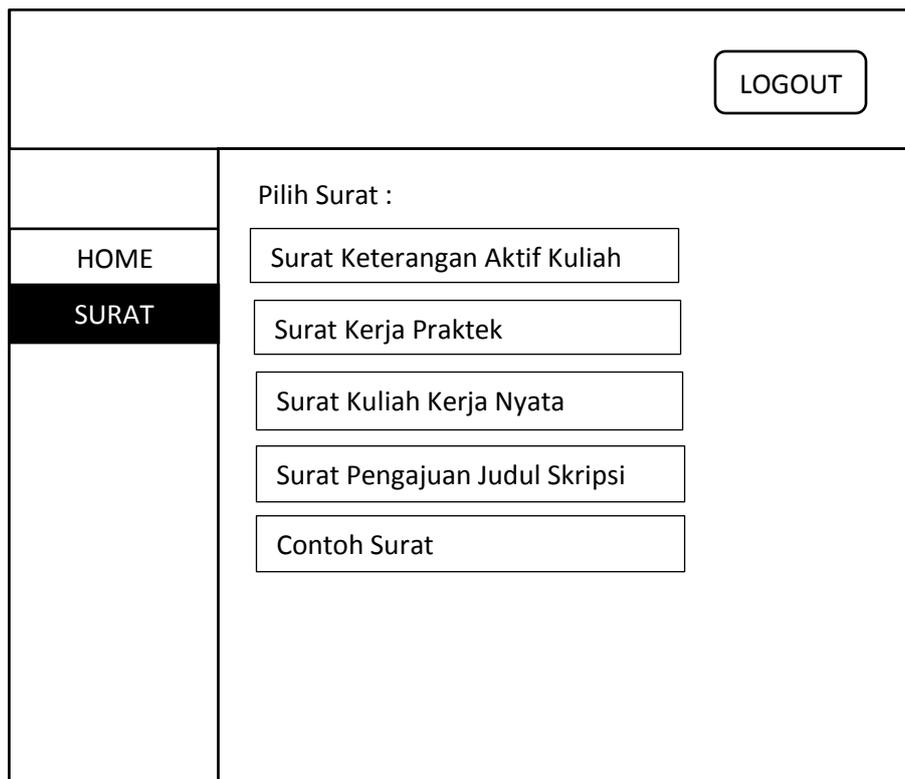


ADMIN		<input type="button" value="LOGOUT"/>
HOME		
SURAT		

Gambar 3.14 Halaman Utama Admin



Gambar 3.15 Tampilan *Menu Home*



Gambar 3.16 Tampilan *Menu Surat*

<input type="button" value="LOGOUT"/>	
	<div style="background-color: black; color: white; padding: 2px; display: inline-block;">Surat Kerja Praktek</div>
HOME	
SURAT	<p>Dengan ini menyetujui bahwa mahasiswa (i) yang tercantum namanya di bawah ini :</p> <p>Nama/NIM :/.....</p> <p>Program Studi/Konsentrasi :/.....</p> <p>Nomor Telepon :</p> <p>Akan Melaksanakan Kerja Praktek pada :</p> <p>Nama Perusahaan :</p> <p>Alamat Perusahaan :</p> <p>Waktu :</p> <p>Demikian Demikian form ini dibuat untuk dipergunakan sebagaimana mestinya.</p> <p style="text-align: right;">Makassar, 2020</p> <p style="text-align: center;">Mahasiswa(i) yang bermohon</p> <p style="text-align: center;">.....</p> <p style="text-align: center;">Menyetujui,</p> <p style="text-align: center;">Ketua Program Studi Teknik Elektro</p> <p style="text-align: center;">Asmawaty Azis, ST., MT.</p>

Gambar 3.17 Contoh Tampilan Memilih *Menu* Surat Kerja Praktek

<input type="button" value="LOGOUT"/>						
	Lihat Surat Masu					
HOME	NO.	NAMA MAHASISWA	STAMBUK	KODE SURAT	KETERANGAN	VALIDASI
VALIDASI	1.	Rifqy Murjani	1520221042	2/A	<input type="button" value="Lihat Surat"/>	<input checked="" type="checkbox"/>
	2.					
	3.					

Gambar 3.18 Tampilan *Menu Validasi*

<input type="button" value="LOGOUT"/>					
	Koreksi				
HOME					
VALIDASI	<table border="1" style="width: 100%;"> <tr> <td style="width: 50%;">Waktu</td> <td style="width: 50%;">.....</td> </tr> <tr> <td>Nomor Telepon</td> <td>.....</td> </tr> </table>	Waktu	Nomor Telepon
Waktu				
Nomor Telepon				

Gambar 3.19 Tampilan *Form Koreksi*

3.3 Waktu dan Tempat Penelitian

Penelitian tersebut dilaksanakan di Fakultas Teknik Universitas Fajar Makassar yang beralamat di jalan Abdurahman Basalamah No. 101 penelitian dilakukan pada tanggal 30 Juni – 30 September 2021.

3.4 Alat dan Bahan Penelitian

Berikut adalah alat dan bahan yang digunakan pada saat penelitian untuk membuat aplikasi persuratan, ialah:

3.4.1. Alat

- Laptop ASUS GL553VD Core i7 Gen 7
Spesifikasi : RAM 8GB

3.4.2. Bahan

- Microsoft Windows 10
- XAMPP Version 7.2.32-0
- PHP MyAdmin
- MySQL
- Visual Studio Code

3.5 Metode Pengumpulan Data

Pengumpulan data adalah salah satu hal yang penting dilakukan dalam memperoleh data yang diinginkan. Dengan adanya data yang diambil tersebut, sangat membantu dalam menghasilkan informasi yang diinginkan.

3.5.1 Data Primer

Pengumpulan data primer, yaitu data yang dikumpulkan dan diolah sendiri oleh peneliti langsung dari subjek atau objek penelitian dan dilakukan dengan cara observasi di lokasi penelitian dengan melakukan metode pengamatan langsung dan mengambil data tersebut. Observasi dilakukan di Universitas Fajar Makassar. Observasi yang dilakukan yaitu mengumpulkan data berupa format surat administrasi akademik Universitas Fajar Makassar.

3.5.2 Data Sekunder

Data sekunder, yaitu data yang didapatkan tidak secara langsung dari objek atau penelitian dengan mengolah data dari dokumen-dokumen yang sudah ada sebelumnya baik berasal dari jurnal, artikel, buku, internet dan beberapa sumber lain yang bersangkutan dengan masalah-masalah yang penulis teliti untuk memperoleh landasan teori yang dapat menunjang penelitian.

3.6 Metode Pengujian Sistem

Pada tahap ini, tahap analisa data atau metode pengujian yang digunakan yaitu metode *Black Box Testing* dan *White Box Testing*. Pengujian *Black Box* berfokus pada persyaratan fungsional dan suatu sistem. *White Box* adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan stuktur kontrol dari desain program secara prosedural untuk membagi pengujian ke dalam beberapa kasus pengujian.

BAB IV HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Berdasarkan penelitian yang telah dilakukan, berikut merupakan hasil yang diperoleh dari sistem yang telah dirancang. Adapun pengguna yang dapat mengakses sistem ini yaitu, pihak Administrasi Fakultas Teknik Universitas Fajar sebagai admin, Dekan Fakultas dan Program Studi Universitas Fajar Makassar sebagai validator, serta Mahasiswa sebagai pengguna. Berikut uraian penjelasan dan alur penggunaan aplikasi :

4.1.1. Tampilan Login



Gambar 4.1 Tampilan *Login*

Gambar 4.1 menunjukkan tampilan yang digunakan oleh Admin, Dekan Fakultas Teknik, Program Studi, dan User atau Mahasiswa yang menjadi gerbang awal untuk mengakses beranda aplikasi. Berikut merupakan *script* dari gambar 4.1 :


```

1 <div class="row">
2   <div class="col-lg-1">
3     <ul style="list-style-type: none;">
4       <li>Home</li>
5       <li>Pengajuan Surat</li>
6       <li>Check Status Surat</li>
7     </ul>
8   </div>
9   <div class="col-lg-2">
10    <div style="border: 1px solid #ccc; padding: 5px; text-align: center; font-weight: bold; font-size: 1.2em; margin-bottom: 10px;>
11      User: Rijal
12    </div>
13  </div>
14 </div>

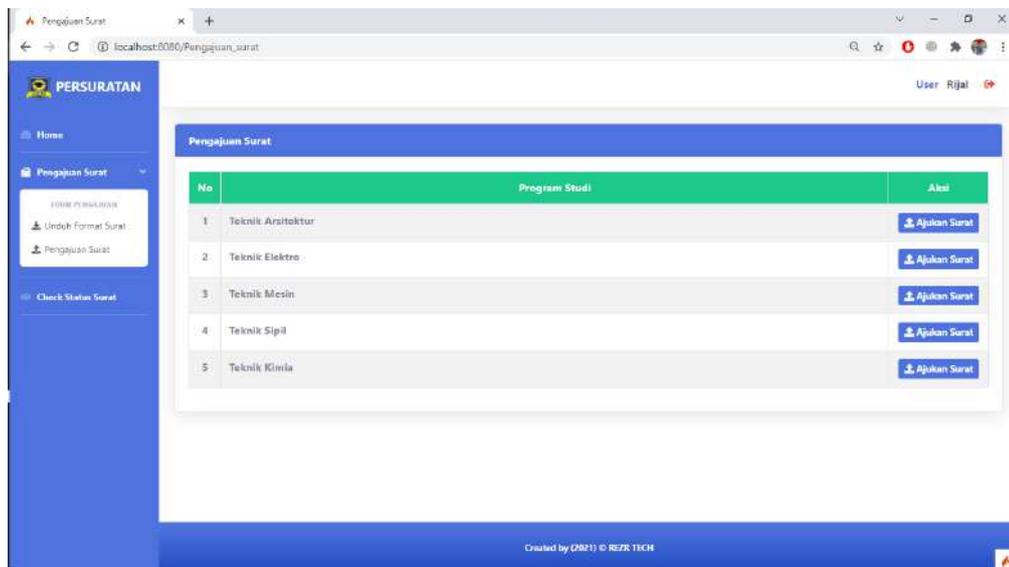
```

Gambar 4.4 Script Tampilan Beranda/Dashboard

4.1.3 Tampilan User

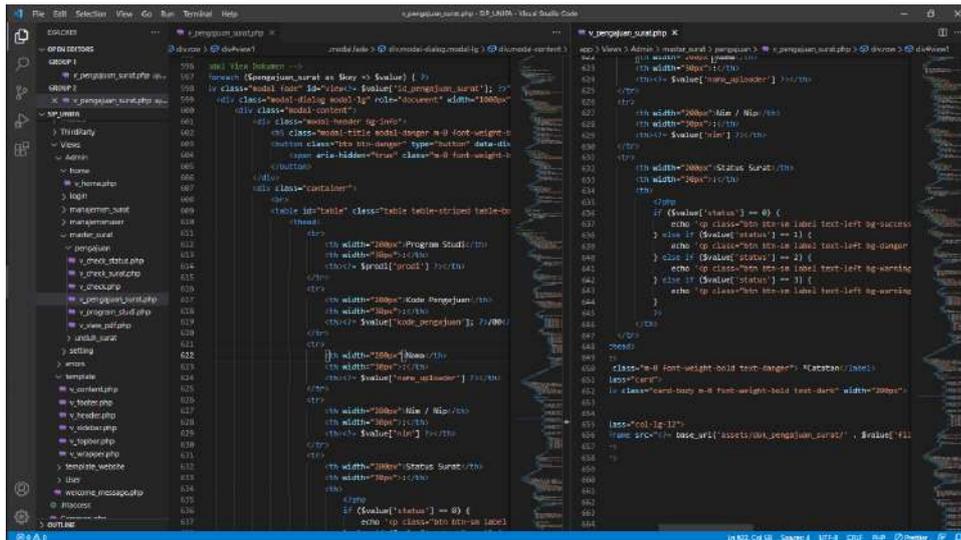
User mempunyai beberapa menu yang terdiri dari *Dashboard*, Pengajuan Surat dan *Check Status Surat*. Berikut tampilan-tampilan dari menu tersebut:

1) Pengajuan Surat



Gambar 4.5 Tampilan Pengajuan Surat

Pada gambar 4.5, menampilkan menu yang berfungsi untuk membuat surat yang diperlukan User atau Pengguna. Berikut *script* dari menu pengajuan surat:

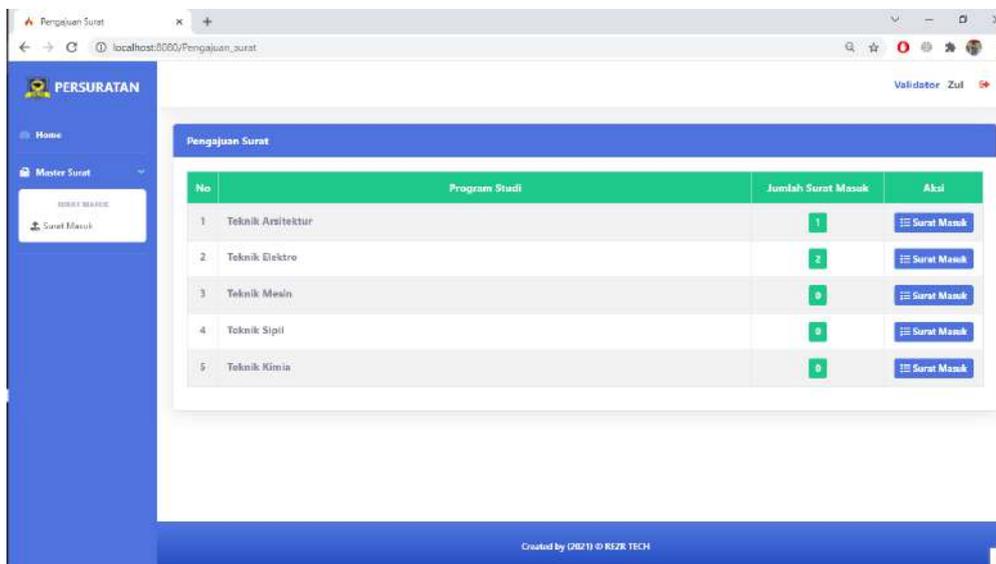


Gambar 4.8 Script Check Status Surat

4.1.4 Tampilan Validator

Validator terbagi atas 3, Validator 1 selaku pihak Admin Fakultas Teknik, Validator 2 selaku pihak Dekan Fakultas, Validator 3 selaku pihak Program Studi Fakultas Teknik. Berikut tampilan-tampilan dari menu tersebut:

1. Master Surat



Gambar 4.9 Tampilan Master Surat

Berikut *script* dari tampilan master surat:

```

<code>
</code>

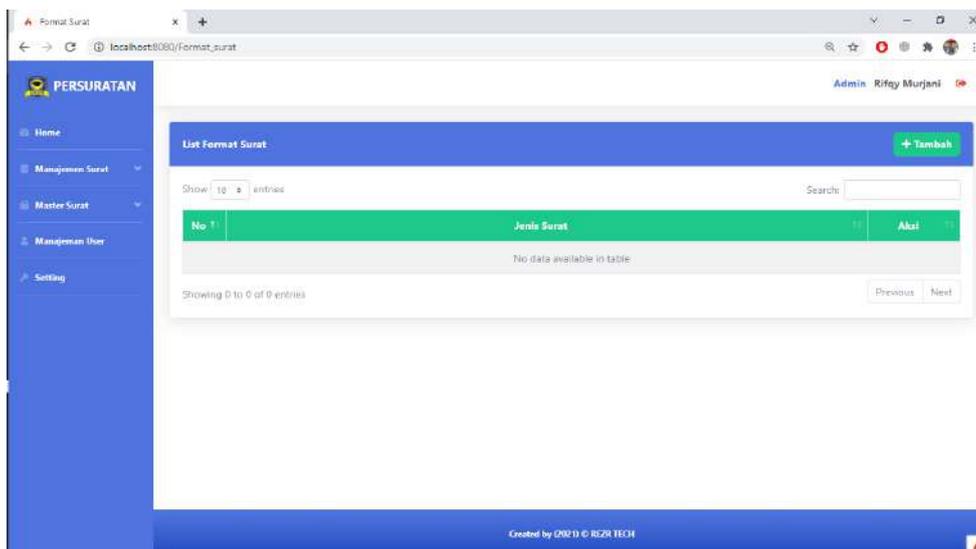
```

Gambar 4.10 *Script* Tampilan Master Surat

4.1.5 Tampilan Admin

Admin mempunyai beberapa menu yang terdiri dari *dashboard*, Manajemen Surat, Master Surat, Manajemen User, Setting. Berikut tampilan-tampilan dari menu tersebut:

1) Manajemen Surat



Gambar 4.11 Tampilan Manajemen Surat

Pada tampilan ini menunjukkan beberapa daftar Program Studi yang menyimpan beberapa arsip surat yang diperlukan oleh *User/Mahasiswa*. Berikut *script* dari menu Manajemen Surat

```

class DownloadFormatSurat
{
    public $model;
}

class DownloadFormatSuratController
{
    public function __construct()
    {
        $this->model = new DownloadFormatSurat();
    }

    public function index()
    {
        $data = array();
        $data['title'] = 'Manajemen Surat';
        $data['model'] = $this->model;
        $data['breadcrumb'] = array('Manajemen Surat', 'Manajemen Surat');
        return view('manajemen_surat/index', $data);
    }

    public function download($id)
    {
        $data = array();
        $data['title'] = 'Download Surat';
        $data['model'] = $this->model;
        $data['breadcrumb'] = array('Manajemen Surat', 'Manajemen Surat', 'Download Surat');
        echo view('manajemen_surat/download', $data);
    }
}
    
```

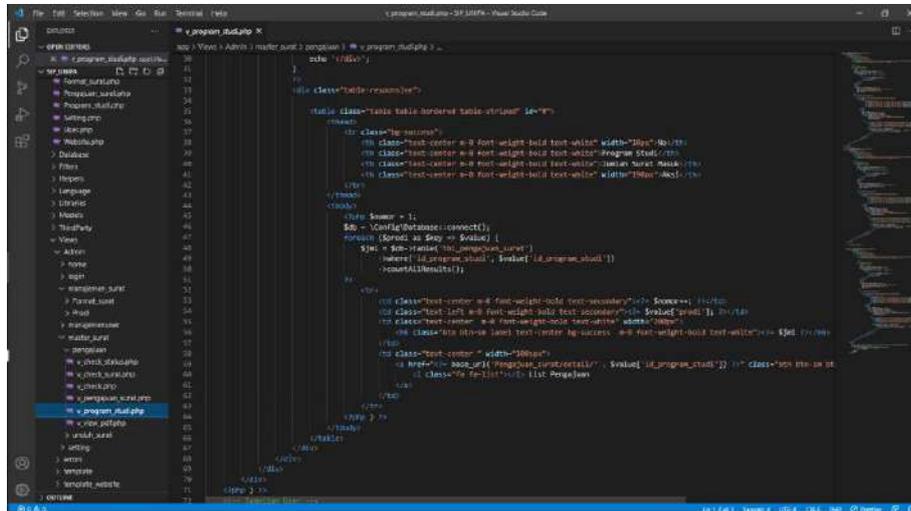
Gambar 4.12 *Script* Manajemen Surat

2) Master Surat

No	Program Studi	Jumlah Surat Masuk	Aksi
1	Teknik Arsitektur	1	List Pengajuan
2	Teknik Elektro	2	List Pengajuan
3	Teknik Mesin	0	List Pengajuan
4	Teknik Sipil	0	List Pengajuan
5	Teknik Kimia	0	List Pengajuan

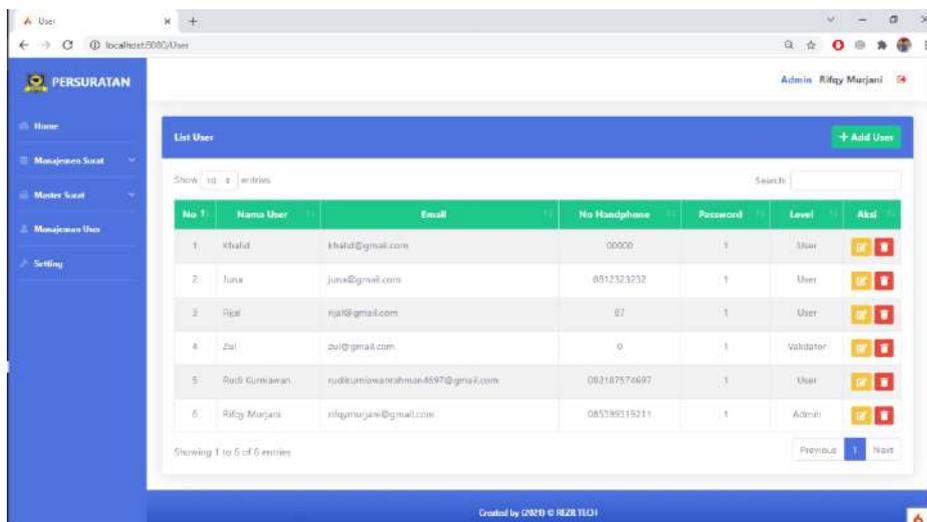
Gambar 4.13 Tampilan Master Surat

Pada gambar diatas, menunjukkan surat-surat yang telah di *upload* oleh Admin. Berikut *script* dari tampilan master surat:



Gambar 4.14 Script Master Surat

3) Manajemen User



Gambar 4.15 Tampilan Manajemen User

Pada gambar 4.15 menunjukkan *user-user* atau pengguna dari aplikasi persuratan ini. Berikut *script* tampilan manajemen *user*:

```

class BaseController
{
    public function __construct()
    {
        helper('Form');
        $this->ModelUser = new ModelUser();
    }

    public function index()
    {
        $data = [
            'title' => 'User',
            'user' => $this->ModelUser->getAllData(),
            'isi' => 'Admin/ManajemenUser/View/index_user',
        ];
        echo view('template/user_wrapper', $data);
    }

    public function add_user()
    {
        $data = [
            'title' => 'Add User',
            'isi' => 'Admin/ManajemenUser/View/add_user',
        ];
        echo view('template/user_wrapper', $data);
    }

    public function insert()
    {
        if ($this->validasi([
            'name_user' => [
                'label' => 'Nama User',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Isi Dan Tidak Boleh Kosong'
                ]
            ],
            'email' => [
                'label' => 'No-Idempotensi',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Isi Dan Tidak Boleh Kosong'
                ]
            ],
            'password' => [
                'label' => 'Password',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Isi Dan Tidak Boleh Kosong'
                ]
            ],
            'level' => [
                'label' => 'level',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Pilih Dan Tidak Boleh Kosong'
                ]
            ]
        ])) {
            // Logic for inserting user
        }
    }

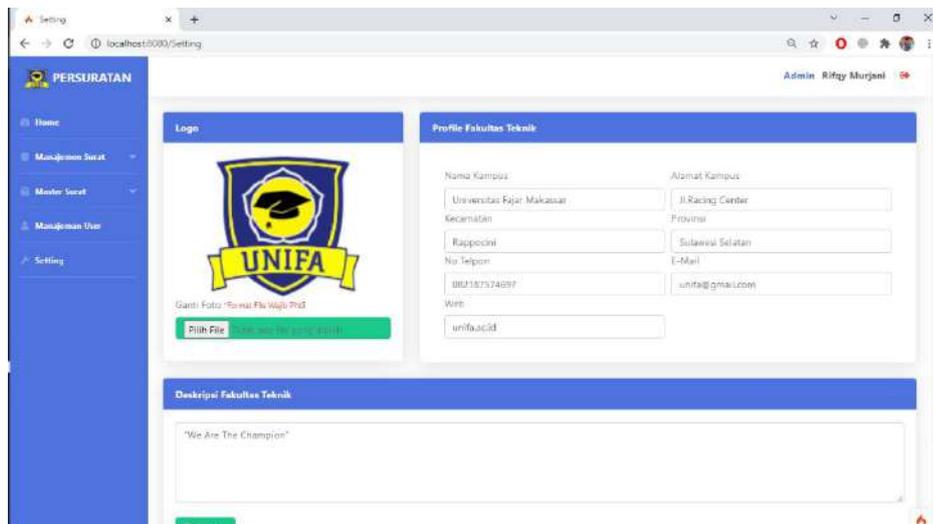
    public function edit_user($id_user)
    {
        $data = [
            'title' => 'Edit User',
            // Logic for editing user
        ];
        echo view('template/user_wrapper', $data);
    }

    public function update_user($id_user)
    {
        if ($this->validasi([
            'name_user' => [
                'label' => 'Nama User',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Isi Dan Tidak Boleh Kosong'
                ]
            ],
            'email' => [
                'label' => 'No-Idempotensi',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Isi Dan Tidak Boleh Kosong'
                ]
            ],
            'password' => [
                'label' => 'Password',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Isi Dan Tidak Boleh Kosong'
                ]
            ],
            'level' => [
                'label' => 'level',
                'rules' => 'required',
                'errors' => [
                    'required' => '(Field) Wajib Di Pilih Dan Tidak Boleh Kosong'
                ]
            ]
        ])) {
            // Logic for updating user
        }
    }
}

```

Gambar 4.16 Script Manajemen User

4) Setting



Gambar 4.17 Tampilan Setting

Pada gambar 4.17 menampilkan beberapa pengaturan-pengaturan pada aplikasi persuratan ini. Berikut *script* dari tampilan *setting*:

```

1 namespace App\Controllers;
2 class Setting extends BaseController
3 {
4     public function __construct()
5     {
6         helper('Form');
7         $this->Model_admin = new Model_admin();
8     }
9     public function index()
10    {
11        $data = array(
12            'title' => 'Setting',
13            'pengaturan' => $this->Model_admin->detailSetting(),
14            'isi' => 'admin/setting/v_setting',
15        );
16        return view('template/v_wrapper', $data);
17    }
18    public function setting()
19    {
20        $data = array(
21            'title' => 'Setting',
22            'pengaturan' => $this->Model_admin->detailSetting(),
23            'isi' => 'admin/setting/v_setting',
24        );
25        return view('template/v_wrapper', $data);
26    }
27    public function save_setting()
28    {
29        $file = $this->request->getFile('logo');
30        // jika gambar tidak diupload
31        if ($file->getError() == 4) {
32            $data = [
33                'nama_sesuai' => $this->request->getPost('nama_sesuai'),
34                'kewasit' => $this->request->getPost('kewasit'),
35                'keperawatan' => $this->request->getPost('keperawatan'),
36                'provinsi' => $this->request->getPost('provinsi'),
37                'no_telpom' => $this->request->getPost('no_telpom'),
38            ];
39        }
40    }
41 }

```

Gambar 4.18 Script Setting

4.2 Pembahasan

Berdasarkan hasil yang telah didapatkan dalam perancangan dan implementasi Aplikasi Persuratan Akademik Fakultas Teknik di Universitas Fajar Makassar Berbasis *WEB*, maka sistem ini telah dibuat sesuai dengan perancangan dan pengujian yang telah dilakukan sebelumnya.

Pengujian yang dilakukan berupa pengujian yang bertujuan untuk mengetahui hasil dari perancangan dan implementasi Aplikasi Persuratan Akademik Fakultas Teknik Universitas Fajar Berbasis *WEB*. Pengujian dilakukan dengan metode *blackbox testing* dan *whitebox testing* yang bertujuan untuk mengetahui apakah fungsi-fungsi pada sistem yang dibuat sudah berjalan sebagaimana mestinya.

4.2.1 Blackbox Testing

Tabel 4.1 Instrumen Pengujian Fungsional Aplikasi

Instrumen Pengujian <i>Black Box</i>					
No	Komponen yang diujikan	Skenario	Hasil yang diharapkan	Keterangan	
				Berfungsi	Tidak berfungsi

1.	Membuka Aplikasi	Klik Aplikasi Persuratan	Aplikasi akan menunjukkan halaman <i>Login</i>	Ya	
2.	<i>Login Admin</i>	Memasukkan <i>Username</i> dan <i>Password</i> dan klik <i>login</i>	Sistem akan <u>menampilkan</u> halaman beranda yang berisi menu (<i>Dashboard</i> , <i>Manajemen Surat</i> , <i>Master Surat</i> , <i>Manajemen User</i> , <i>Setting</i>)	Ya	
		Memasukkan <i>Username</i> dan <i>Password</i> salah dan klik <i>login</i>	Sistem menolak akses <i>login</i> dan tampilan tetap berada di halaman <i>login</i>	Ya	
3.	<i>Login Validator</i>	Memasukkan <i>Username</i> dan <i>Password</i> dan klik <i>login</i>	Sistem akan menampilkan halaman beranda yang berisi menu sidebar (<i>Dashboard</i> , <i>Master Surat</i>).	Ya	
		Memasukkan <i>Username</i> dan <i>Password</i>	Sistem menolak akses <i>login</i> dan tampilan tetap berada di halaman <i>login</i>		

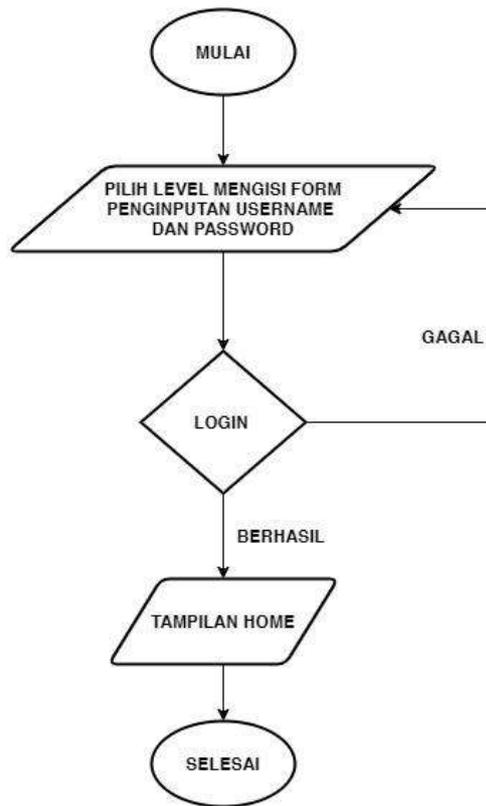
		salah dan klik <i>login</i>			
4.	<i>Login User</i>	Memasukkan <i>Username</i> dan <i>Password</i> dan klik <i>login</i>	Sistem akan menampilkan halaman beranda yang berisi menu (<i>Dashboard</i> , Pengajuan Surat, Check Status Surat).	Ya	
		Memasukkan <i>Username</i> dan <i>Password</i> salah dan klik <i>login</i>	Sistem menolak akses <i>login</i> dan tampilan tetap berada di halaman <i>login</i>	Ya	
5.	Menu Manajemen Surat	Klik menu Manajemen Surat pada <i>sidebar</i>	Sistem menerima akses dan menampilkan Program Studi, Tambah Surat	Ya	
6.	Menu Master Surat	Klik menu Master Surat pada <i>sidebar</i>	Sistem menerima akses dan menampilkan Pengajuan Surat, List Pengajuan Surat	Ya	
6.	Menu Manajemen User	Klik menu Manajemen User pada <i>sidebar</i>	Sistem menerima akses dan menampilkan form Menu Manajemen User	Ya	

		Klik tombol <i>Add User</i>	Sistem menerima akses dan Menampilkan form Input user.	Ya	
		Klik Tombol Simpan <i>User Baru</i>	Sistem Menerima akses dan Menyimpan data ke database.	Ya	
7.	Menu <i>Setting</i>	Klik menu Form Peminjaman pada <i>sidebar</i>	Sistem menerima akses dan menampilkan Form Permohonan Peminjaman.	Ya	
		Klik menu <i>Setting</i>	Sistem menerima akses dan menampilkan menu-menu pengaturan yang ada di aplikasi	Ya	
8.	Menu <i>Dashboard</i>	Klik menu <i>Dashboard</i> pada <i>sidebar</i>	Sistem menerima akses dan menampilkan Deskripsi Tentang Aplikasi.	Ya	

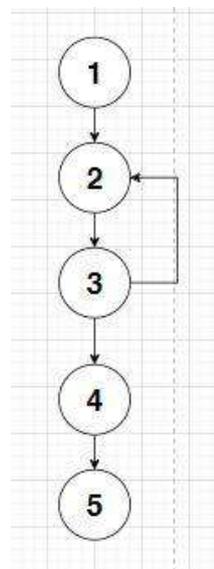
4.2.2 Whitebox Testing

1. Login

Login berfungsi untuk mengamankan akun, di sini ada 4 klasifikasi hak akses untuk mengakses aplikasi yaitu, Admin, Dekan Fakultas, Program Studi, User



Gambar 4.19 Diagram *Flowchart White Box Testing Login*



Gambar 4.20 Diagram *Flowgraph White Testing Login*

Dari gambar *Flow graph login* dapat dilakukan proses sebagai berikut:

Menghitung *cyclomatic complexity* dari *edge* dan *node*.
 Grafik alir terbagi 2 region, yaitu sebagai berikut:

1. $V(G) = 5\text{edge} - 5\text{node} + 2 = 2$
2. $V(G) = 1 \text{ perkiraan simpul} + 1 = 2$

Cyclomatic complexity dari Gambar 4.20 *flow graph login* ialah memiliki 2 jalur independennya, jalur independennya sebagai berikut:

Path 1 yaitu : 1-2-3-4-5

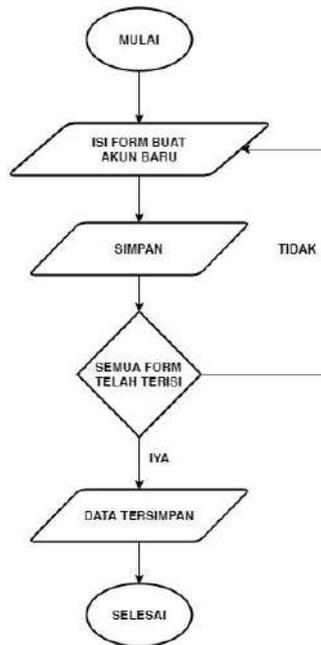
Path 2 yaitu : 1-2-3-2-3-4-5

Tabel 4.2 *Test Case Login*

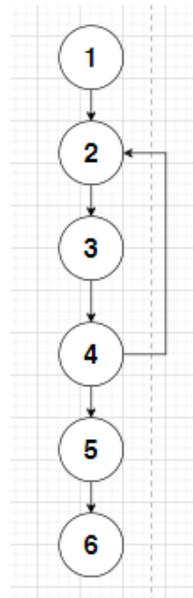
Path	1
Jalur	1-2-3-4-5
Skenario	-Mulai -Pilih level mengisi form penginputan username dan password -Login -Tampilan Home -Selesai
Hasil Pengujian	<i>Success</i>
Path	2
Jalur	1-2-3-2-3-4-5
Skenario	-Mulai -Pilih level mengisi form penginputan username dan password -Login -Login gagal karena user dan <u>password salah</u> -Mengisi ulang username dan password -Tampilan Home -Selesai
Hasil Pengujian	<i>Success</i>

2. Buat Akun Baru

Form buat akun baru berfungsi untuk membuat suatu akun baru bagi pengguna.



Gambar 4.21 Diagram *Flowchart White Box Testing* Buat Akun Baru



Gambar 4.22 Diagram *Flowgraph Whitebox Testing* Buat Akun Baru

Dari gambar *Flow graph login* dapat dilakukan proses sebagai berikut:

Menghitung *cyclomatic complexity* dari *edge* dan *node*.
 Grafik alir terbagi 2 region, yaitu sebagai berikut:

$$1. V(G) = 6\text{edge} - 6\text{node} + 2 = 2$$

$$2. V(G) = 1 \text{ perkiraan simpul} + 1 = 2$$

Cyclomatic complexity dari Gambar 4.22 *flow graph login* ialah memiliki 2 jalur independennya, jalur independennya sebagai berikut:

Path 1 : 1-2-3-4-5-6

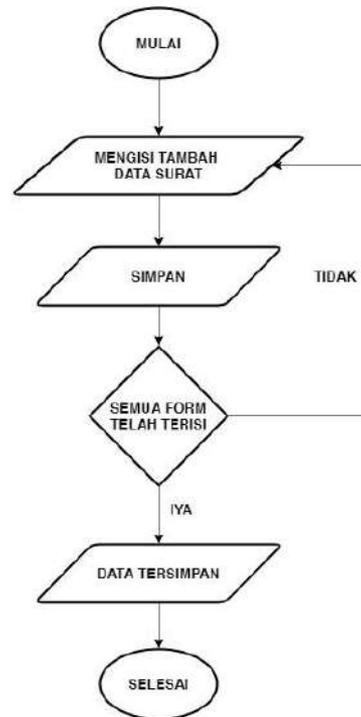
Path 2 : 1-2-3-4-2-3-4-5-6

Tabel 4.3 *Test Case* Buat Akun Baru

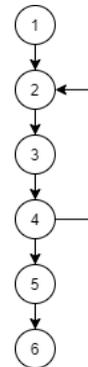
Path	1
Jalur	1-2-3-4-5-6
Skenario	-Mulai -Isi form buat akun baru -Simpan -Semua form telah terisi -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>
Path	2
Jalur	1-2-3-4-2-3-4-5-6
Skenario	-Mulai -Isi form buat akun baru -Simpan -Semua form telah terisi -Ada form yang belum terisi/ <i>error</i> -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>

3. Tambah Data Format Surat

Form tambah data format surat berfungsi untuk mengirim surat peminjaman fasilitas kampus.



Gambar 4.23 Diagram *Flowchart White Box Testing* Tambah Data Format Surat



Gambar 4.24 Diagram *Flowgraph White Box Testing* Tambah Data Format Surat

Dari gambar *Flow graph login* dapat dilakukan proses sebagai berikut:

Menghitung *cyclomatic complexity* dari *edge* dan *node*.
 Grafik alir terbagi 2 region, yaitu sebagai berikut:

1. $V(G) = 6\text{edge} - 6\text{node} + 2 = 2$
2. $V(G) = 1 \text{ perkiraan simpul} + 1 = 2$

Cyclomatic complexity dari Gambar 4.24 *flow graph login* ialah memiliki 2 jalur independennya, jalur independennya sebagai berikut:

Path 1 : 1-2-3-4-5-6

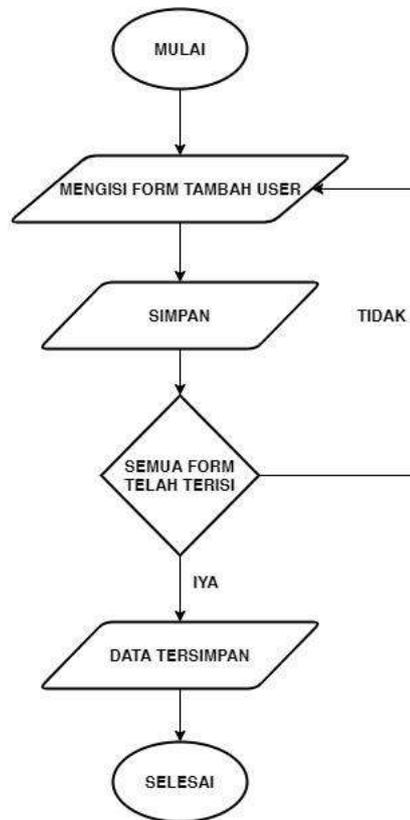
Path 2 : 1-2-3-4-2-3-4-5-6

Tabel 4.4 *Test Case* Tambah Data Format Surat

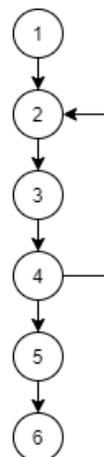
Path	1
Jalur	1-2-3-4-5-6
Skenario	-Mulai -Mengisi tambah data surat -Simpan -Semua form telah terisi -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>
Path	2
Jalur	1-2-3-4-2-3-4-5-6
Skenario	-Mulai -Mengisi tambah data surat -Simpan -Semua form telah terisi -Ada form yang belum terisi/ <i>error</i> -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>

4. Tambah User

Tambah user berfungsi untuk menambah pengguna aplikasi



Gambar 4.25 Diagram *Flowchart White Box Testing* Tambah User



Gambar 4.26 Diagram *Flowgraph White Box Testing* Tambah User

Dari gambar *Flow graph login* dapat dilakukan proses sebagai berikut:

Menghitung *cyclomatic complexity* dari *edge* dan *node*.
 Grafik alir terbagi 2 region, yaitu sebagai berikut:

1. $V(G) = 6 \text{ edge} - 6 \text{ node} + 2 = 2$
2. $V(G) = 1 \text{ perkiraan simpul} + 1 = 2$

Cyclomatic complexity dari Gambar 4.26 *flow graph login* ialah memiliki 2 jalur independennya, jalur independennya sebagai berikut:

Path 1 : 1-2-3-4-5-6

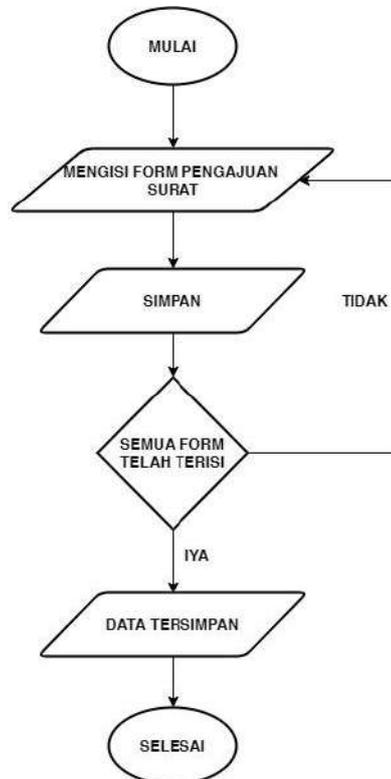
Path 2 : 1-2-3-4-2-3-4-5-6

Tabel 4.5 *Test Case Tambah User*

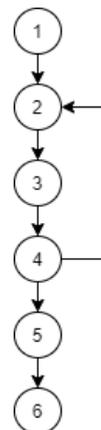
Path	1
Jalur	1-2-3-4-5-6
Skenario	-Mulai -Mengisi form tambah user -Simpan -Semua form penginputan terisi -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>
Path	2
Jalur	1-2-3-4-2-3-4-5-6
Skenario	-Mulai - Mengisi form tambah user -Simpan -Ada form yang belum terisi/ <i>error</i> -Mengisi ulang form tambah user -Semua form telah terisi -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>

5. Tambah Pengajuan Surat

Tambah Pengajuan Surat berfungsi untuk menambah atau *upload* surat yang diperlukan



Gambar 4.27 Diagram *Flowchart White Box Testing* Tambah Pengajuan Surat



Gambar 4.28 Diagram *Flowgraph White Box Testing* Tambah Pengajuan Surat

Dari gambar *Flow graph login* dapat dilakukan proses sebagai berikut:

Menghitung *cyclomatic complexity* dari *edge* dan *node*.
 Grafik alir terbagi 2 region, yaitu sebagai berikut:

1. $V(G) = 6 \text{ edge} - 6 \text{ node} + 2 = 2$
2. $V(G) = 1 \text{ perkiraan simpul} + 1 = 2$

Cyclomatic complexity dari Gambar 4.22 *flow graph login* ialah memiliki 2 jalur independennya, jalur independennya sebagai berikut:

Path 1 : 1-2-3-4-5-6

Path 2 : 1-2-3-4-2-3-4-5-6

Tabel 4.6 *Test Case* Tambah Pengajuan Surat

Path	1
Jalur	1-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> -Mulai -Mengisi form pengajuan surat -Simpan -Semua form telah terisi -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>
Path	2
Jalur	1-2-3-4-2-3-4-5-6
Skenario	<ul style="list-style-type: none"> -Mulai - Mengisi form pengajuan surat -Simpan -Ada form yang belum terisi/<i>error</i> -Mengisi ulang form pengajuan surat -Semua form telah terisi -Data tersimpan -Selesai
Hasil Pengujian	<i>Success</i>

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dari pembuatan dan perancangan Aplikasi Persuratan Akademik di Fakultas Teknik Universitas Fajar Makassar dapat diambil beberapa kesimpulan, sebagai berikut:

1. Peneliti telah berhasil merancang dan membuat Aplikasi Persuratan Akademik Universitas Fajar Makassar berbasis *WEB*. Hasil aplikasi persuratan akademik berbasis *web* ini menunjukkan hasil yang mendukung pengolahan *database* dan manajemen persuratan yang ada di Fakultas Teknik Universitas Fajar Makassar serta berhasil menampilkan data sesuai kebutuhan, sehingga aplikasi ini berhasil menampilkan apa yang diharapkan dan dapat diimplementasikan secara langsung. Berdasarkan penelitian ini, dapat disimpulkan bahwa aplikasi masih membutuhkan pengembangan yang lebih lanjut.
2. Aplikasi ini menggunakan metode pengujian *blackbox testing* dan *whitebox testing*. Berdasarkan pengujian fungsional dari metode *blackbox* yang beberapa kali melalui pengujian, bahwa fungsional aplikasi telah berfungsi dengan baik sesuai dengan rancangan, sedangkan berdasarkan hasil pengujian *whitebox* menyatakan bahwa semua logika program telah sesuai. Berdasarkan hasil analisis pengujian efektifitas oleh pengguna, aplikasi ini telah efektif sesuai dengan yang diharapkan.
3. Hasil dari penelitian ini sangat membantu untuk menunjang pengarsipan dan persuratan di lingkup Fakultas Teknik Universitas Fajar Makassar.

5.2 Saran

Penulis menyadari banyak kekurangan dalam penelitian ini, berikut beberapa saran yang dibutuhkan dalam pengembangan sistem pembuatan aplikasi ini pada penelitian terkait selanjutnya, yaitu :

1. Aplikasi ini perlu pengembangan dari segi pengguna, yaitu dosen maupun civitas akademika dari Fakultas Teknik Universitas Fajar Makassar itu sendiri.
2. Pada aplikasi ini diharapkan agar nantinya dilakukan pengembangan-pengembangan yang lebih baik dari segi konten, maupun tampilan.

Demikian saran penulis, semoga dapat dijadikan bahan evaluasi yang khususnya dapat bermanfaat bagi penulis dan pemngembang pada lazimnya.

DAFTAR PUSTAKA

- Abdulloh, rohi (2013), 7 in 1 Pemrograman *web* untuk pemula, Elex Media Komputindo Indonesia
- Dadan & Kerendi Developers. (2015). Membuat CMS Multifitur. Jakarta: PT. Elex Media Komputindo.
- Enterprise Jubilee (2018), HTML, PHP dan MySQL untuk pemula, Elex Media Komputindo Indonesia
- Hartono, Jogiyanto. (2001) "*Analisis dan Desain Sistem Informasi*": Pendekatan terstruktur teori dan praktek aplikasi bisnis, Andi : Yogyakarta.
- Harumy, T.Henny Febriana. Agus Perdana Windarto, dan Indri Sulistianingsih. (2016). *Belajar Dasar Algoritma & Pemrograman C++*. Medan: Pemula.
- Heriyanto, Y. (2018). Perancangan Sistem Informasi Rental Mobil Berbasis *Web* Pada PT.APM Rent Car. *Jurnal Intra-Tech*, 2(2), 64–77.
- Iswandy, Eka. (2015). *Sistem Penunjang Keputusan Untuk Menentukan Penerimaan Dana Santunan Sosial Anak Nagari dan Penyalurannya bagi Mahasiswa dan Pelajar Kurang Mampu di Kenagarian Barung-barung Balantai Timur*. Padang: Dosen Sekolah Tinggi Manajemen Informatika Komputer STMIK Jayanusa Padang.
- Juansyah, (2013). Pengertian Sistem Informasi, <http://juansyah.wordpress.com/2013/03/31/pengertian-sistem-informasi/diakses> pada tanggal 18 September 2015.
- Nurhadi. (2017). *Pondasi Dasar Pemrograman Website*. Surabaya: Garuda Mas Sejahtera
- Pratama, I Putu Agus Eka. (2014), Sistem Informasi dan Implementasinya. Bandung : Informatika,
- Pratama, Wahyu Bagus dan Ali, Tarmuji. (2015), "RANCANG BANGUN SISTEM INFORMASI PERSURATAN DAN PENGARSIPAN BERBASIS *WEB*

PADA PG GONDANG BARU KLATEN". Skripsi. Yogyakarta: Universitas Ahmad Dahlan, Informatika

Pressman, Roger S. (2010), "*Software Engineering a Practitioner's Approach*". New York, McGraw-Hill, Inc.

Purwanto Djoko (2011), *Komunikasi Bisnis*. Jakarta : Penerbit Erlangga.

Putra Dede Wira Trise dan Rahmi Andrian (2019), Unified Modelling Language (UML) dalam Perancangan Sistem Informasi Permohonan Pembayaran Restitusi SPPD, Fakultas Teknologi Industri, Institut Teknologi Padang

Rahmad, Mhd Bustanur dan Setiady, Tedy (2014) "PERANCANGAN SISTEM INFORMASI INVENTORY SPARE PART ELEKTRONIK BERBASIS WEB PHP (Studi CV. Human Global Service Yogyakarta)". Skripsi. Yogyakarta: Universitas Ahmad Dahlan, Informatika,

Saputra, Risang Suryadi, Hafidudin dan Dadan Nur Ramadan (2018), Aplikasi system monitoring jaringan berbasis *website*. Teknik Telekomunikasi, Fakultas Ilmu Terapan, Universitas Telkom.

Setiawan, Didik (2017), Buku sakti pemrograman *web*, Startup Yogyakarta

Sidik, Betha (2014) Pemrograman *Web* dengan PHP Revisi Kedua. Bandung : Informatika Bandung.

Sulistiono, Heru. (2018). Coding Mudah dengan CodeIgniter, jQuery, Bootstrap, dan Datatable. Jakarta: PT. Elex Media Komputindo.

Sutabri, (2005). Sistem Informasi Manajemen. Yogyakarta: Andi.

Tri Sandhika Jaya, (2018). "*Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis (Studi Kasus: Kantor Digital Politeknik Negeri Lampung)*", Program Studi Manajemen Informatika, Jurusan Ekonomi dan Bisnis, Politeknik Negeri Lampung

Wardana. (2010). Menjadi Master PHP dengan Framework CodeIgniter. Jakarta: PT. Elex Media Komputindo.

Yudhanto, Yudho, dan Helmi Adi Prasetyo. (2018). Panduan Mudah Belajar Framework Laravel. Jakarta: PT. Elex Media Komputindo.

Yuhefizar, HA Mooduto, Rahmat Hidayat. (2009). Cara Mudah Membangun Website Interaktif Menggunakan Content Management Sistem Joomla Edisi Revisi. Jakarta: PT. Elex Media Komputindo.

Yusdiardi, (2014), "Rancang Bangun Sistem Informasi Penjualan (Studi Kasus : PT I-Cube Creativindo), Jakarta: Universitas Islam Negeri Syarif Hidayatullah