

**RANCANG BANGUN APLIKASI WEB SISTEM  
INFORMASI SKRIPSI INTERAKTIF  
MENGUNAKAN GOOGLE DIALOGFLOW  
(Di Universitas Pamulang)**

**Skripsi**



**FAJAR AL HAKIM  
2016142116**

**TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS PAMULANG  
TANGERANG  
2020**

**RANCANG BANGUN APLIKASI WEB SISTEM  
INFORMASI SKRIPSI INTERAKTIF  
MENGUNAKAN GOOGLE DIALOGFLOW  
(Di Universitas Pamulang)**

**Skripsi**

**Diajukan Untuk Melengkapi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer**



**FAJAR AL HAKIM  
2016142116**

**TEKNIK INFORMATIKA  
FAKULTAS TEKNIK  
UNIVERSITAS PAMULANG  
TANGERANG  
2020**

## **LEMBAR PERNYATAAN**

Yang bertanda tangan di bawah ini:

Nama : FAJAR AL HAKIM  
NIM : 2016142116  
Program Studi : TEKNIK INFORMATIKA  
Fakultas : TEKNIK  
Jenjang Pendidikan : STRATA 1

Menyatakan bahwa skripsi yang saya buat dengan judul:

**RANCANG BANGUN APLIKASI WEB SISTEM INFORMASI SKRIPSI  
INTERAKTIF MENGGUNAKAN GOOGLE DIALOGFLOW (Di  
Universitas Pamulang)**

1. Merupakan hasil karya tulis ilmiah sendiri, bukan merupakan karya yang pernah diajukan untuk memperoleh gelar akademik oleh pihak lain, dan bukan merupakan hasil plagiat.
2. Saya ijin untuk dikelola oleh Universitas Pamulang sesuai dengan norma hukum dan etika yang berlaku.

Pernyataan ini saya buat dengan penuh tanggung jawab dan saya bersedia menerima konsekuensi apapun sesuai aturan yang berlaku apabila dikemudian hari pernyataan ini tidak benar.

Pamulang, 14 Agustus 2020

(FAJAR AL HAKIM)

## LEMBAR PERSETUJUAN

NIM : 2016142116  
Nama : FAJAR AL HAKIM  
Program Studi : TEKNIK INFORMATIKA  
Jenjang Pendidikan : STRATA 1  
Judul Skripsi : **RANCANG BANGUN APLIKASI WEB SISTEM  
INFORMASI SKRIPSI INTERAKTIF  
MENGUNAKAN GOOGLE DIALOGFLOW  
(Di Universitas Pamulang)**

Skripsi ini telah diperiksa dan disetujui oleh pembimbing untuk persyaratan sidang skripsi.

Pamulang, 14 Agustus 2020

Pembimbing

Munawaroh S.Kom.,M.Kom  
NIDN. 0402089004

Mengetahui  
Kaprodi Teknik Informatika

Dr. Ir. Sewaka, M.M  
NIDN. 8842760018

## LEMBAR PERSETUJUAN

NIM : 2016142116  
Nama : FAJAR AL HAKIM  
Program Studi : TEKNIK INFORMATIKA  
Jenjang Pendidikan : STRATA 1  
Judul Skripsi : **RANCANG BANGUN APLIKASI WEB SISTEM  
INFORMASI SKRIPSI INTERAKTIF  
MENGUNAKAN GOOGLE DIALOGFLOW  
(Di Universitas Pamulang)**

Skripsi ini telah dipertahankan dihadapan dewan penguji ujian skripsi Fakultas Teknik, Program studi Teknik Informatika dan dinyatakan LULUS.

Pamulang, 14 Agustus 2020

Penguji I

Penguji II

Ari Syaripudin, S.Kom., M.Kom.  
NIDN. 0406028802

Yono Cahyono, S.Kom., M.Kom.  
NIDN. 0420028403

Pembimbing

Munawaroh S.Kom.,M.Kom.  
NIDN. 0402089004

Mengetahui  
Kaprodi Teknik Informatika

Dr. Ir. Sewaka, M.M  
NIDN. 8842760018

## DAFTAR RIWAYAT HIDUP

### Data Pribadi

Nama : FAJAR AL HAKIM  
Tempat, Tanggal Lahir : Grobogan, 8 Juni 1998  
Kebangsaan : Indonesia  
Alamat : Kp. Sanggrahan No. 6 RT.001/006  
Kel. Meruya Utara, Kec. Kembangan  
Jakarta Barat 11620  
Jenis Kelamin : Laki - laki  
Status : Single  
Agama : Islam  
No. Handphone : 085777774107  
Email : fajaral\_id@programmer.net

### Pendidikan Formal

2016 – 2020	Universitas Pamulang, Tangerang Selatan
2013 – 2016	SMA Negeri 1 Gubug, Kab. Grobogan
2010 – 2013	SMP Negeri 1 Tanggunharjo, Kab. Grobogan
2004 – 2010	SD Negeri 2 Sugihmanik, Kab. Grobogan

## ABSTRACT

*Thesis subjects become compulsory subjects in the Department of Informatics at the University of Pamulang whose implementation does not follow the information on KRS during the key-in, therefore any information about all procedures from registration to the issuance of the value of these courses students must find information independently.*

*The problem often faced by students is the lack of information provided on socialization, guidelines, websites, etc., making students have to find alternative sources of information. The source of information that students get is through friends, which is not necessarily valid information and the responses given can be very long. The solution offered in this study for the problem that occurs is to make an information system application into an interactive service (chatbot) that all students can access anytime and anywhere. The developed application will be web-based so that students can access it without having to have a platform or device operating system limitation used.*

*The purpose of this research is to make an information system application an interactive service (chatbot) for students majoring in Informatics Engineering at Pamulang University. The flow of work in this study is observation, interviews, and data collection. The observation phase is to retrieve data on any student's problem when he wants to find information about the Thesis procedure and from any source of information that students get apart from guides, socialization, websites, and so on. Interviews were conducted in this study by searching for any data needed from guides, students and lecturers. The design phase is to design the application interface, design the system architecture and design a knowledge base. The implementation and testing phase is to make everything that has been designed and tested it. This data collection users will provide feedback on the use of the application and will be evaluated based on the results of testing in the previous stage.*

*Keywords: thesis, information system, chatbot, application.*

## ABSTRAK

Mata kuliah Skripsi menjadi mata kuliah wajib pada jurusan Teknik Informatika di Universitas Pamulang yang pelaksanaannya tidak mengikuti informasi pada KRS saat *key-in*, maka dari itu setiap informasi tentang seluruh prosedur mulai dari pendaftaran hingga keluarnya nilai mata kuliah tersebut mahasiswa harus mencari informasi secara mandiri.

Masalah yang sering dihadapi oleh mahasiswa adalah kurangnya informasi yang diberikan pada sosialisasi, panduan, *website*, dan lain – lain, membuat mahasiswa harus mencari sumber informasi alternatif. Sumber informasi yang mahasiswa dapatkan adalah melalui teman, yang belum tentu *valid* informasinya dan respon yang diberikan bisa jadi sangat lama. Solusi yang ditawarkan pada penelitian ini untuk masalah yang terjadi adalah membuat sebuah aplikasi sistem informasi menjadi layanan interaktif (*chatbot*) yang dapat semua mahasiswa akses kapan saja dan dimana saja. Aplikasi yang dikembangkan akan berbasis *web* agar mahasiswa dapat mengaksesnya tanpa harus ada batasan *platform* atau sistem operasi gawai yang digunakan.

Tujuan pada penelitian ini adalah membuat aplikasi sistem informasi menjadi layanan interaktif (*chatbot*) untuk mahasiswa jurusan Teknik Informatika di Universitas Pamulang. Alur pengerjaan pada penelitian ini adalah observasi, wawancara, dan pengumpulan data. Tahap observasi untuk mengambil data apa saja masalah mahasiswa ketika ingin mencari informasi seputar prosedur Skripsi dan dari mana saja sumber informasi yang mahasiswa dapatkan selain dari panduan, sosialisai, *website*, dan lain – lain. Wawancara yang dilakukan pada penelitian ini dengan cara mencari setiap data yang diperlukan dari panduan, mahasiswa dan dosen. Tahap perancangan untuk mendesain antarmuka aplikasi, merancang arsitektur sistem dan merancang *knowledge base*. Tahap implementasi dan pengujian adalah untuk membuat segala hal yang telah dirancang dan mengujinya. Pengumpulan data ini pengguna akan memberikan umpan balik penggunaan aplikasi dan akan dilakukan evaluasi berdasarkan hasil pengujian pada tahap sebelumnya.

Kata kunci: skripsi, sistem informasi, chatbot, aplikasi.



## KATA PENGANTAR

Puji syukur Alhamdulillah kehadiran Allah SWT yang telah melimpahkan segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan laporan skripsi yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada program studi Teknik Informatika di Universitas Pamulang.

Penulis menyadari skripsi ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa penulis terima dengan senang hati. Dengan segala keterbatasan, penulis menyadari pula bahwa skripsi ini takkan terwujud tanpa bantuan, bimbingan, dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada:

1. Allah SWT yang telah memberikan begitu banyak nikmat dan karunia diantaranya iman dan Islam serta sehat dan umur panjang sehingga bisa menyelesaikan studi ini.
2. Bapak Dr. (HC) H. Darsono, selaku ketua Yayasan Sasmita Jaya yang telah memberikan tempat untuk mencari ilmu.
3. Bapak Dr. H. Dayat Hidayat, MM., selaku Rektor Universitas Pamulang.
4. Bapak Ir. Dadang Kurnia, MM., selaku Dekan Fakultas Teknik Universitas Pamulang.
5. Bapak Dr. Ir. Sewaka M.M, selaku Ketua Program Studi Teknik Informatika Universitas Pamulang.
6. Ibu Munawaroh S.Kom,.M.Kom, selaku Dosen Pembimbing yang sudah membimbing dan memberikan motivasi serta petunjuk dan arahan kepada penulis dalam menyelesaikan Tugas Akhir ini.
7. Bapak dan Ibu tercinta serta keluarga yang telah mendoakan, dan memberikan semangat untuk menyelesaikan tugas akhir ini.
8. Terimakasih kepada para staff, karyawan, dan dosen Universitas Pamulang yang secara langsung atau tidak langsung telah memberikan dukungan.

9. Terima kasih untuk kawan-kawan angkatan 2016 khususnya Teknik Informatika kelas 08TPLM008 dan mahasiswa di Forum Ketua Kelas Teknik Informatika Universitas Pamulang yang senantiasa memberikan dukungan untuk menyelesaikan tugas akhir ini.
10. Serta semua pihak yang tidak dapat saya sebutkan satu per satu, terima kasih atas kontribusinya.

Akhirnya mengingat kemampuan penulis yang dirasakan masih banyak kekurangan, maka apabila terdapat kekurangan dalam penyusunan tugas akhir ini dapat kiranya dimaafkan, semoga tugas akhir ini dapat bermanfaat. Penulis sangat berterima kasih sekali apabila pembaca memberikan saran dan kritik sehingga laporan tugas akhir ini mudah-mudahan bisa memberikan nilai yang lebih, khususnya bagi penulis dan umumnya bagi pembaca.

Semoga Allah SWT melimpahkan kasih dan rahmat-Nya kepada kita semua dan memudahkan segala urusan perkuliahan kita dan menunjukan jalan yang lebih terbuka dan berpeluang memperoleh prestasi dan kesuksesan.

Pamulang, 14 Agustus 2020

(Fajar Al Hakim)

## DAFTAR ISI

LEMBAR PERNYATAAN .....	iii
LEMBAR PERSETUJUAN.....	iv
DAFTAR RIWAYAT HIDUP.....	vi
<i>ABSTRACT</i> .....	vii
ABSTRAK .....	viii
KATA PENGANTAR .....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xv
DAFTAR TABEL .....	xvii
DAFTAR LAMPIRAN.....	xviii
BAB I PENDAHULUAN .....	1
1.1    Latar Belakang .....	1
1.2    Identifikasi Masalah .....	2
1.3    Rumusan Masalah .....	2
1.4    Batasan Penelitian .....	3
1.5    Tujuan Penelitian.....	3
1.6    Manfaat Penelitian.....	3
1.7    Metode Penelitian.....	3
1.7.1    Metode Pengumpulan Data.....	4
1.7.2    Metode Pengembangan Perangkat Lunak.....	4
1.7.3    Metode Pengujian Perangkat Lunak .....	5
BAB II LANDASAN TEORI .....	7
2.1    Tinjauan Pustaka .....	7
2.2    Tinjauan Teori .....	10
2.2.1    Pengertian Perancangan.....	10

2.2.2	Pengertian Aplikasi.....	11
2.2.3	Pengertian Perancangan Aplikasi .....	11
2.2.4	Pengertian Sistem .....	11
2.2.5	Pengertian Informasi.....	13
2.2.6	Pengertian <i>Web</i> .....	15
2.2.7	Google Dialogflow .....	15
2.3	Aplikasi Pendukung .....	17
2.3.1	XAMPP.....	17
2.3.2	PHP .....	18
2.3.3	MySQL .....	18
2.3.4	Codeigniter.....	19
2.3.5	Bootstrap.....	19
2.3.6	Sublime Text.....	19
2.4	Pengertian Metode <i>Waterfall</i> .....	20
2.5	Teori Perancangan Basis Data.....	21
2.5.1	ERD ( <i>Entity Relationship Diagram</i> ).....	21
2.5.2	LRS ( <i>Logical Record Structure</i> ).....	22
2.6	<i>Unified Modelling Language</i> (UML) .....	22
2.6.1	<i>Activity Diagram</i> .....	23
2.6.2	<i>Use Case Diagram</i> .....	24
2.6.3	<i>Class Diagram</i> .....	25
2.6.4	<i>Sequence Diagram</i> .....	26
2.7	Notasi UML.....	27
2.8.1	<i>Black Box Testing</i> .....	30
BAB III ANALISA DAN PERANCANGAN .....		33
3.1	Analisa Sistem.....	33
3.1.1	Analisa Sistem Berjalan.....	33
3.1.2	Analisa Sistem Usulan .....	35
3.2	Perancangan Basis Data .....	36
3.2.1	<i>Entity Relationship Diagram</i> (ERD).....	36
3.2.2	Transformasi ERD Ke <i>Logical Record Structure</i> (LRS).....	37
3.2.3	<i>Logical Record Structure</i> (LRS).....	38

3.2.4	Spesifikasi Basis Data.....	38
3.3	Pengajuan <i>Unified Modelling Language</i> (UML) .....	40
3.3.1	<i>Use Case</i> Diagram .....	40
3.3.2	<i>Activity</i> Diagram .....	41
3.3.3	<i>Class</i> Diagram.....	49
3.3.4	<i>Sequence</i> Diagram .....	50
3.4	Perancangan Antar Muka ( <i>User Interface</i> ) .....	55
3.4.1	Halaman <i>Login</i> .....	56
3.4.2	Halaman <i>Dashboard</i> .....	56
3.4.3	Halaman <i>Post</i> Info .....	57
3.4.4	Halaman <i>Post</i> Dosen.....	58
3.4.5	Halaman <i>Post</i> Eprints .....	58
3.4.6	Halaman <i>Post</i> Judul .....	59
3.4.7	Halaman Utama .....	60
3.4.8	Halaman Simulasi .....	60
3.4.9	Halaman Informasi .....	61
3.4.10	Halaman Judul .....	62
3.4.12	Halaman <i>Form</i> Eprints.....	63
BAB IV IMPLEMENTASI DAN PENGUJIAN.....		65
4.1	Spesifikasi Sistem.....	65
4.1.1	Spesifikasi Perangkat Keras.....	65
4.1.2	Implementasi Perangkat Lunak .....	65
4.2	Implementasi Antarmuka ( <i>Interface</i> ) .....	66
4.2.1	Halaman Utama .....	66
4.2.2	Halaman <i>Login</i> .....	67
4.2.3	Halaman <i>Dashboard</i> Admin.....	67
4.2.4	Halaman <i>Post</i> Info .....	68
4.2.5	Halaman <i>Post</i> Dosen.....	68
4.2.6	Halaman <i>Post</i> Eprints .....	69
4.2.7	Halaman <i>Post</i> Judul .....	69
4.2.8	Halaman Skripsi.....	70
4.2.9	Halaman Informasi .....	70

4.2.10 Halaman Judul .....	71
4.2.11 Halaman Dosen.....	71
4.2.12 Halaman Eprints .....	72
4.2.13 Halaman <i>Chatbot</i> .....	72
4.3 Pengujian .....	73
4.3.1 Pengujian <i>Black Box</i> .....	73
BAB V PENUTUPAN .....	78
5.1 Kesimpulan.....	78
5.2 Saran .....	78
DAFTAR PUSTAKA .....	79

## DAFTAR GAMBAR

Gambar 1. 1 Skema Model <i>Waterfall</i> .....	5
Gambar 2. 1 Alur Kerja <i>Dialogflow Agent</i> .....	16
Gambar 2. 2 Metode <i>Waterfall</i> .....	20
Gambar 3. 1 <i>Activity Diagram</i> Sistem Yang Berjalan .....	34
Gambar 3. 2 <i>Activity Diagram</i> Sistem Yang Diusulkan .....	35
Gambar 3. 3 <i>Entity Relationship Diagram</i> .....	36
Gambar 3. 4 ERD ke <i>Logical Record Structure</i> .....	37
Gambar 3. 5 <i>Logical Record Structure</i> .....	38
Gambar 3. 6 <i>Use Case Diagram</i> .....	41
Gambar 3. 7 <i>Activity Diagram Login</i> .....	42
Gambar 3. 8 <i>Activity Diagram Bot</i> .....	43
Gambar 3. 9 <i>Activity Diagram</i> Kontak Dosen .....	44
Gambar 3. 10 <i>Activity Diagram</i> Judul .....	45
Gambar 3. 11 <i>Activity Diagram</i> Eprints .....	46
Gambar 3. 12 <i>Activity Diagram Post</i> Informasi .....	47
Gambar 3. 13 <i>Activity Diagram Post</i> Dosen .....	48
Gambar 3. 14 <i>Activity Diagram Update</i> Data Eprints .....	49
Gambar 3. 15 <i>Class Diagram Bot</i> .....	50
Gambar 3. 16 <i>Sequence Diagram Login</i> Admin .....	50
Gambar 3. 17 <i>Sequence Diagram Post</i> Informasi .....	51
Gambar 3. 18 <i>Sequence Diagram Post</i> Dosen .....	52
Gambar 3. 19 <i>Sequence Diagram Post</i> Eprints .....	53
Gambar 3. 20 <i>Sequence Diagram</i> Informasi .....	53
Gambar 3. 21 <i>Sequence Diagram</i> Judul .....	54
Gambar 3. 22 <i>Sequence Diagram</i> Dosen .....	54
Gambar 3. 23 <i>Sequence Diagram</i> Eprints .....	55
Gambar 3. 24 <i>Sequence Diagram Chatbot</i> .....	55
Gambar 3. 25 Rancangan Tampilan Halaman <i>Login</i> .....	56
Gambar 3. 26 Rancangan Tampilan Halaman <i>Dashboard</i> .....	56

Gambar 3. 27 Rancangan Tampilan Halaman <i>Post Info</i> .....	57
Gambar 3. 28 Rancangan Tampilan Halaman <i>Post Dosen</i> .....	58
Gambar 3. 29 Rancangan Tampilan Halaman <i>Post Eprints</i> .....	58
Gambar 3. 30 Rancangan Tampilan Halaman <i>Post Judul</i> .....	59
Gambar 3. 31 Rancangan Tampilan Halaman <i>Utama</i> .....	60
Gambar 3. 32 Rancangan Tampilan Halaman <i>Simulasi</i> .....	60
Gambar 3. 33 Rancangan Tampilan Halaman <i>Informasi</i> .....	61
Gambar 3. 34 Rancangan Tampilan Halaman <i>Judul</i> .....	62
Gambar 3. 35 Rancangan Tampilan Halaman <i>Chatbot</i> .....	62
Gambar 3. 36 Rancangan Tampilan Halaman <i>Form Eprints</i> .....	63
Gambar 3. 37 Rancangan Tampilan Halaman <i>Kontak Dosen</i> .....	64
Gambar 4. 1 Tampilan Halaman <i>Utama</i> .....	66
Gambar 4. 2 Tampilan Halaman <i>Login Admin</i> .....	67
Gambar 4. 3 Tampilan Halaman <i>Dashboard</i> .....	67
Gambar 4. 4 Tampilan Halaman <i>Post Info</i> .....	68
Gambar 4. 5 Tampilan Halaman <i>Post Dosen</i> .....	68
Gambar 4. 6 Tampilan Halaman <i>Post Eprints</i> .....	69
Gambar 4. 7 Tampilan Halaman <i>Post Judul</i> .....	69
Gambar 4. 8 Tampilan Halaman <i>Skripsi</i> .....	70
Gambar 4. 9 Tampilan Halaman <i>Informasi</i> .....	70
Gambar 4. 10 Tampilan Halaman <i>Judul</i> .....	71
Gambar 4. 11 Tampilan Halaman <i>Dosen</i> .....	71
Gambar 4. 12 Tampilan Halaman <i>Eprints</i> .....	72
Gambar 4. 13 Tampilan Halaman <i>Chatbot</i> .....	72



## DAFTAR TABEL

Tabel 2. 1 ERD ( <i>Entity Relationship Diagram</i> ) .....	22
Tabel 2. 2 Simbol <i>Activity Diagram</i> .....	24
Tabel 2. 3 Simbol <i>Use Case Diagram</i> .....	25
Tabel 2. 4 Simbol <i>Class Diagram</i> .....	26
Tabel 2. 5 Simbol <i>Sequence Diagram</i> .....	27
Tabel 3. 1 Tabel user .....	39
Tabel 3. 2 Tabel <i>post_dosen</i> .....	39
Tabel 3. 3 Tabel <i>post_acrticle</i> .....	39
Tabel 3. 4 Tabel <i>post_eprints</i> .....	39
Tabel 3. 5 Tabel <i>post_judul</i> .....	40
Tabel 4. 1 <i>Blackbox Login Admin</i> .....	73
Tabel 4. 2 <i>Blackbox Post Info</i> .....	74
Tabel 4. 3 <i>Blackbox Post Dosen</i> .....	75
Tabel 4. 4 <i>Blackbox Post Eprints</i> .....	75
Tabel 4. 5 <i>Blackbox Judul</i> .....	76
Tabel 4. 6 <i>Blackbox Utama</i> .....	76
Tabel 4. 7 <i>Blackbox Info Dosen</i> .....	76
Tabel 4. 8 <i>Blackbox Form Eprints</i> .....	77

## **DAFTAR LAMPIRAN**

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Skripsi menjadi karya tulis ilmiah yang berdasarkan hasil penelitian lapangan dan atau studi kepustakaan yang disusun mahasiswa sesuai dengan bidang studinya sebagai tugas akhir dalam studi formalnya di Program Studi Teknik Informatika Universitas Pamulang. Prosedur untuk program Skripsi berbeda dengan mata kuliah wajib pada umumnya. Berbedanya prosedur dan pembayaran pada program Skripsi dibanding mata kuliah yang lain, menjadikan sebagian mahasiswa harus mencari informasi lebih lanjut ketika ingin mengikuti atau mengambil mata kuliah tersebut. Informasi tentang Skripsi telah disediakan oleh Universitas Pamulang pada situs resminya ([www.my.unpam.ac.id](http://www.my.unpam.ac.id)), namun informasi yang diberikan masih membuat sebagian mahasiswa kurang paham tentang prosedur perkuliahan tersebut. Kurang pahamnya sebagian mahasiswa dalam menelaah setiap informasi Skripsi menjadikan mereka lambat dalam proses studi. Peran aktif mahasiswa dalam mencari informasi seputar Skripsi sangat dibutuhkan, karena prosedur Skripsi dapat berubah sewaktu - waktu sesuai kebijakan jurusan dan universitas.

Jurusan Teknik Informatika UNPAM mempunyai sebuah layanan bagi mahasiswa yang ingin bertanya seputar informasi akademik tanpa harus datang ke jurusan selain situs *web*, yaitu layanan telepon dan *email*. Kontak *email* dan nomor telepon yang disediakan oleh jurusan Teknik Informatika UNPAM ditujukan bagi masyarakat umum maupun mahasiswa, layanan ini sebenarnya dapat dimanfaatkan oleh mahasiswa dalam mencari informasi seputar prosedur Skripsi namun pelayanan telepon yang masih dikenakan tarif panggilan dan *email* yang responnya tidak interaktif membuat mahasiswa enggan untuk menggunakan layanan yang telah disediakan oleh jurusan. Penulis mencari tahu masalah mahasiswa Teknik Informatika UNPAM dalam mencari informasi seputar Skripsi yaitu sulitnya mencari informasi tentang persyaratan dan langkah - langkah

Skripsi yang harus disiapkan kurang lengkap yang kadang masih membuat mahasiswa bingung yang terbatas jarak dan waktu dan harus mencari sendiri atau harus bertanya kepada mahasiswa lain

Berdasarkan permasalahan yang ada maka dilakukan penelitian ini yang bertujuan untuk membuat sebuah aplikasi *web* yang menjadi layanan interaktif dalam melakukan pencarian informasi yang *update* dan *valid* seputar Skripsi di jurusan Teknik Informatika UNPAM. Aplikasi dari penelitian yang dilakukan ini dapat digunakan kapan pun dan dapat diakses dimana pun, agar para mahasiswa yang ingin mengetahui informasi seputar prosedur Skripsi mendapatkan informasi secara cepat dan *valid*. Penelitian ini menggunakan sebuah platform milik *Google Inc* yaitu *Dialogflow*. *Dialogflow* adalah sebuah *platform chatbot* yang berbasis *machine learning* yang ditujukan untuk mengolah masukan kata dan akan menghasilkan sebuah jawaban berupa teks berformat JSON (*Javascript Object Notation*), pada aplikasi *web* yang format JSON yang diterima akan diolah menjadi format teks untuk pengguna. Dalam penerapannya nanti, aplikasi ini dapat menjadi sebuah layanan interaktif yang dapat digunakan untuk mahasiswa dalam mencari informasi seputar Skripsi.

## **1.2 Identifikasi Masalah**

Berdasarkan latar belakang masalah di atas, identifikasi masalah dalam penelitian ini adalah sebagai berikut:

- a. Sulitnya mencari informasi tentang persyaratan dan langkah - langkah Skripsi yang harus disiapkan kurang lengkap yang masih membuat mahasiswa bingung dengan terbatas jarak dan waktu dan harus mencari sendiri atau harus bertanya kepada mahasiswa lain

## **1.3 Rumusan Masalah**

Berdasarkan latar belakang dan identifikasi masalah di atas maka dapat dirumuskan masalah yaitu:

- a. Bagaimana menyediakan informasi Skripsi Universitas Pamulang yang interaktif dan efektif?

#### **1.4 Batasan Penelitian**

Berikut beberapa batasan masalah pada penelitian ini:

- a. Aplikasi ini hanya ditujukan untuk menginformasikan sesuatu yang umum seputar Skripsi pada jurusan Teknik Informatika UNPAM
- b. Aplikasi ini hanya menampilkan informasi kontak Dosen jurusan Teknik Informatika UNPAM
- c. Aplikasi ini hanya menampilkan judul Skripsi yang sudah pernah dibuat di UNPAM
- d. Aplikasi ini hanya dapat diakses jika ada koneksi internet

#### **1.5 Tujuan Penelitian**

Tujuan dari penelitian ini adalah:

- a. Perancangan aplikasi *web* untuk layanan interaktif dalam menjawab pertanyaan seputar Skripsi jurusan Teknik Informatika UNPAM
- b. Melihat pengaruh adanya aplikasi ini dengan cara membandingkan kondisi mahasiswa saat mencari informasi setelah adanya aplikasi

#### **1.6 Manfaat Penelitian**

Manfaat yang didapat dari penelitian ini adalah:

- a. Memudahkan mahasiswa dalam mencari tahu informasi yang berkaitan dengan penyusunan Skripsi jurusan Teknik Informatika UNPAM
- b. Jurusan Teknik Informatika UNPAM mempunyai layanan informasi yang interaktif untuk mahasiswa dalam penyusunan Skripsi.

#### **1.7 Metode Penelitian**

Metode penelitian merupakan tahapan - tahapan yang dilalui oleh peneliti mulai dari perumusan masalah sampai kesimpulan, yang membentuk sebuah alur yang sistematis. Metode penelitian yang dilakukan terdiri dari metode pengumpulan data, metode pengembangan perangkat lunak serta metode pengembangan perangkat lunak serta metode pengujian perangkat lunak

### 1.7.1 Metode Pengumpulan Data

Metodologi yang digunakan dalam penulisan skripsi ini adalah sebagai berikut:

a. Sumber data primer:

1) Observasi

Teknik mengumpulkan data dengan cara melakukan penelitian langsung mengunjungi lokasi penelitian, yaitu Universitas Pamulang. Hal ini dilakukan untuk melihat dari dekat masalah-masalah yang berhubungan dengan pokok bahasan yang diperlukan dalam penelitian ini.

2) Sampel data

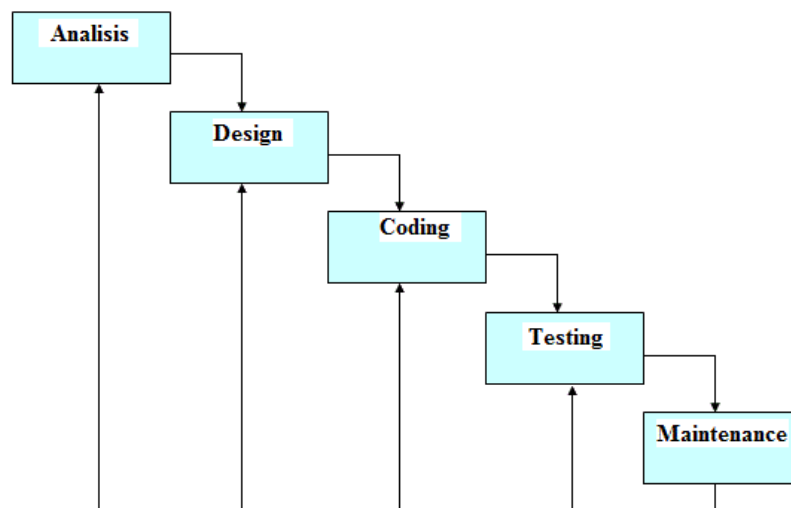
Pengumpulan data dengan cara mengumpulkan data pertanyaan-pertanyaan masalah umum tentang skripsi kepada mahasiswa yang belum, sedang atau sudah skripsi.

b. Sumber data sekunder

Sumber data sekunder diperoleh dengan cara mencari dan mengumpulkan data lengkap dengan mempelajari dan membaca buku-buku yang berhubungan serta menunjang penulisan hasil penelitian.

### 1.7.2 Metode Pengembangan Perangkat Lunak

Dalam membangun Aplikasi Web Sistem Informasi Skripsi Interaktif Pada Jurusan Teknik Informatika Universitas Pamulang, digunakan metode pengembangan perangkat lunak adalah dengan menggunakan metode *waterfall*. Beberapa proses diagram *waterfall* sebagai berikut:



Gambar 1. 1 Skema Model *Waterfall*

a. *Analisa*

Langkah ini merupakan analisa terhadap kebutuhan sistem yang diperlukan dalam pelaksanaan proyek pembuatan aplikasi.

b. *Design*

Tahapan dimana dilakukan penuangan pikiran dan perancangan sistem terhadap solusi dari permasalahan yang ada dengan menggunakan permodelan sistem yang nanti mudah dimengerti oleh pengguna.

c. *Coding*

Merupakan penerjemah *design* dalam bahasa pemrograman yang bisa dikenali oleh komputer.

d. *Testing*

Tahapan akhir dimana sistem dibangun diuji kemampuannya sehingga didapatkan kekurangan dan kelemahan sistem yang kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi tersebut agar menjadi lebih baik dan sempurna.

e. *Maintenance*

Tahapan dimana dilakukan pemeliharaan terhadap aplikasi yang telah dibuat ketika aplikasi sudah diimplementasikan kepada pengguna.

### 1.7.3 Metode Pengujian Perangkat Lunak

Pengujian sistem yang digunakan pada penelitian ini yaitu pengujian *blackbox*. *Blackbox testing* merupakan pengujian yang mengabaikan mekanisme

*internal system* atau komponen dan fokus semata-mata pada *output* yang dihasilkan yang merespon input yang dipilih dan kondisi eksekusi. *Blackbox testing* pengujian yang dilakukan untuk mengevaluasi pemenuhan sistem atau komponen dengan kebutuhan fungsional tertentu. Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut (Crisdiandiandy Herry Nowor, 2013).

## **1.8 Sistematika Penulisan**

Sistematika penulisan yang digunakan dalam penyusunan laporan ini dibedakan dengan pembagian bab sebagai berikut:

### **BAB I PENDAHULUAN**

Pada bab ini memaparkan dan menguraikan tentang latar belakang, identifikasi masalah, rumusan masalah, batasan penelitian, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

### **BAB II LANDASAN TEORI**

Pada bab ini mendeskripsikan tentang teori-teori yang menunjang penelitian/penulisan yang bisa diperkuat dengan menunjukkan hasil penelitian sebelumnya.

### **BAB III ANALISAN DAN PERANCANGAN SISTEM**

Pada bab ini membahas analisa sistem, model sistem informasi serta menguraikan tentang prosedur sistem yang akan dibuat.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN**

Pada bab ini menjelaskan tentang rancangan basis data serta rancangan dialog layar yang akan dilakukan pengujian terhadap aplikasi yang dirancang.

### **BAB V PENUTUP**

Pada bab ini membahas tentang kesimpulan dan saran tentang solusi yang akan diterapkan untuk permasalahan yang dibahas pada penelitian ini



## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Dalam rangka mendapat hasil penelitian yang baik, selain melakukan penelitian langsung, penulis juga melakukan kajian kepustakaan dari hasil penelitian yang telah dilakukan, berikut hasil penelitian yang menjadi acuan:

Albert Yakobus Chandra, Didik Kurniawan, Rahmat Musa (2020) dalam penelitiannya yang berjudul "Perancangan Chatbot Menggunakan Dialogflow Natural Language Processing (Studi Kasus: Sistem Pemesanan pada Coffee Shop)" pada penelitian ini peneliti menemukan beberapa kasus yang sering dialami di lembaga tertentu seperti Perusahaan Mikro seringkali merupakan staf / karyawan dalam memberikan layanan informasi dan transaksi yang dilakukan secara manual kepada pelanggan terkait dengan kegiatan bisnis ini. Siklus ini selalu berulang dari satu pelanggan ke pelanggan lainnya. Dampaknya jika ada kondisi dimana antrian pelanggan yang cukup ramai dari pada beban kerja staf / karyawan akan lebih tinggi dan risiko kesalahan dalam transaksi juga akan tinggi. Perkembangan teknologi informasi dalam kecerdasan buatan pada era industri 4.0 bergerak maju. Salah satunya adalah Machine Learning - Natural Language Processing (NLP) yang merupakan salah satu ilmu yang berfokus pada bagaimana komputer dapat memahami bahasa manusia dan menanggapi. Untuk itu dalam penelitian ini sistem chatbot akan dibangun dalam memberikan informasi dan melakukan transaksi dengan pelanggan. Chatbot ini akan dikembangkan menggunakan alat Dialogflow yang disediakan oleh Google. Chatbot yang dibangun ini diharapkan dapat menjadi alternatif yang dapat diimplementasikan di berbagai bisnis untuk memberikan layanan yang lebih baik bagi pelanggan.

Radhian, Dimar (2019) dalam penelitiannya yang berjudul "Pembangunan Aplikasi Chatbot Sebagai Media Pencarian Informasi Dalam Bidang Peternakan." Pada penelitian ini Dinas Ketahanan Pangan Dan Peternakan Provinsi Jawa Barat merupakan unsur penyelenggara Pemerintah Provinsi Jawa Barat yang melaksanakan urusan pemerintahan bidang pangan dan bidang pertanian. Dinas

Ketahanan Pangan Dan Peternakan Provinsi Jawa Barat dalam misinya berupaya meningkatkan pengetahuan peternak dalam peternakan seperti budidaya ternak. Kurangnya media informasi yang dimiliki, membuat peternak kesulitan mendapatkan informasi dan pengetahuan dalam budidaya ternak. Oleh karena itu dibangunlah aplikasi chatbot. Chatbot merupakan salah satu bentuk aplikasi NLP (Natural Language Processing). Sistem ini digunakan untuk memudahkan pengguna dalam melakukan pencarian informasi yang berisikan informasi pengetahuan dalam budidaya ternak untuk membantu peternak dalam menambah pengetahuan peternak tentang budidaya. Platform yang digunakan untuk mengembangkan chatbot adalah Dialogflow. Platform ini menyediakan layanan NLP (Natural Language Processing) yang membuat chatbot lebih cerdas sehingga dapat memahami maksud dari apa yang ditanyakan oleh pengguna. Hasil penelitian menunjukkan bahwa 72% peternak setuju aplikasi chatbot ini dapat membantu peternak dalam menambah pengetahuan budidaya ternak dan 78% setuju memudahkan peternak dalam mencari informasi budidaya ternak.

Rahmiati, Rahmiati (2019) dalam penelitiannya yang berjudul “Perancangan Sistem Informasi Desa Tegal Sari Kecamatan Satui Menggunakan *Framework Codeigniter*.” penelitian ini dibuat suatu situs web untuk menjawab masalah dalam bentuk media informasi. Desa Tegal Sari adalah salah satu desa di Kabupaten Satui Kalimantan Selatan yang belum memiliki situs web sebagai media informasi. Setiap komunitas yang ingin mengakses informasi di Desa Tegal Sari masih terbatas, untuk mengakses informasi setiap warga negara akan mendapatkan SMS (*Short Message Service*) atau melalui aplikasi Whatsapp oleh masyarakat kabupaten dan prosesnya masih cenderung rumit. Maka dalam penelitian ini dibuat suatu situs web untuk menjawab masalah dalam bentuk media informasi. Aplikasi ini diharapkan mampu menjawab masalah yang ada di Desa Tegal Sari. Aplikasi yang dikembangkan dalam sistem ini menggunakan kerangka kerja Codeigniter (CI). Peneliti memproses informasi dari Desa Tegal Sari menggunakan kerangka Codeigniter (CI) yang mendukung model,

Permana, Eko (2017) dalam penelitiannya yang berjudul “*Perancangan Sistem Informasi Akuntansi Pada Bina Plus Cisarua Dengan Menggunakan PHP Framework Codeigniter Dan MYSQL*” Peneliti dirancang untuk membangun

sistem informasi akuntansi pengelolaan aset tetap pada asrama bina plus cisarua diharapkan untuk membantu mengurangi masalah yang ada sehingga pengelolaan aset tetap bisa lebih efektif dan efisien. Asrama Bina Plus Cisarua beralamatkan di jalan di Jl. Terusan Kolonel Masturi No. 64 Desa Pasir Halang, RT 001 RW 07, Cisarua, Lembang, Kabupaten Bandung, Jawa Barat 40551 adalah sebuah lembaga dibidang pendidikan. Proses pengelolaan aset tetap pada asrama masih manual dengan menggunakan microsoft excel dan microsoft word, sehingga mengakibatkan sulitnya mengetahui kondisi keseluruhan aset tetap yang mengakibatkan data yang dihasilkan kurang akurat. Jenis penelitian yang peneliti gunakan adalah jenis penelitian akademik, jenis desain penelitian adalah deskriptif analitis, metode penelitian adalah metode survei, teknik pengumpulan data yang peneliti gunakan adalah penelitian lapangan yang terdiri dari wawancara, observasi, dan penelitian kepustakaan, struktur pengembangan sistem yang digunakan adalah waterfall, perancangan sistem informasi yang digunakan adalah diagram sequencediagram , BPMN diagram dan use caseDiagram. Peneliti mengusulkan untuk merancang sistem informasi akuntansi pengelolaan aset tetap pada asrama bina plus cisarua diharapkan untuk membantu mengurangi masalah yang ada sehingga pengelolaan aset tetap bisa lebih efektif dan efisien.

Widyantika, Nuke Brilian (2018) dalam penelitiannya yang berjudul “Perancangan Chatbot Menggunakan *Rivescript* Pada *Website E-Commerce* Sebagai *Virtual Customer Service*.” Penelitian ini yang dikarenakan teknologi sekarang berkembang dan kebutuhan pemilik toko online mlkpancing.com untuk menyelesaikan permintaan layanan pelanggan, oleh karena itu desain sistem dibuat untuk memberikan informasi cepat kepada pelanggan. Penelitian ini bertujuan untuk membangun *chatbot* yang dapat menggantikan peran layanan pelanggan manusia dalam hal memberikan informasi kepada pelanggan menggunakan file yang tersimpan yang berisi informasi stok dalam perdagangan di mlkpancing.com di dalam sistem. Desain dan implementasi perangkat lunak ini menghasilkan prototipe chatbot yang telah dibangun dengan memanfaatkan basis pengetahuan *RiveScript*. *RiveScript* ini membawa input yang dapat diintegrasikan dengan baik oleh *chatbot* sebagai input teks. Selain itu, komunikasi sistem pelanggan dikembangkan. Pemanfaatan chatbot buatan intelijen ditingkatkan ini

dapat menyebabkan kemudahan pelanggan untuk mendapatkan informasi dari database dengan cepat. Sistem chatbot ini lulus kompatibilitas pengujian tanggapan. Hasilnya membuktikan bahwa dari 10 topik pertanyaan, ada 9 tanggapan yang kompatibel dan 1 tidak sesuai. Sementara itu, validitas *respons* dari 10 pertanyaan, ada 8 memberikan informasi valid dan 2 tidak valid. *Chatbot* dengan *RiveScript* dapat membantu pelanggan mendapatkan informasi yang mereka inginkan, dengan kesesuaian jawaban hingga 88%. ada 8 memberikan informasi yang *valid* dan 2 tidak *valid*. *Chatbot* dengan *RiveScript* dapat membantu pelanggan mendapatkan informasi yang mereka inginkan, dengan kesesuaian jawaban hingga 88%. ada 8 memberikan informasi yang valid dan 2 tidak *valid*. *Chatbot* dengan *RiveScript* dapat membantu pelanggan mendapatkan informasi yang mereka inginkan, dengan kesesuaian jawaban hingga 88%.

## 2.2 Tinjauan Teori

Teori umum pada bab ini menjelaskan tentang penjabaran teori-teori yang di dapat penulis berlandaskan studi pustaka maupun dari buku, yaitu sebagai berikut:

### 2.2.1 Pengertian Perancangan

Perancangan adalah tahapan perancangan (*design*) memiliki tujuan untuk mendesain sistem baru yang dpat menyelesaikan masalah-masalah yang dihadapi dari pemilihan sistem terbaik. (bin Ladjamudin, 2005).

Perancangan adalah proses pengembangan spesifikasi sitem baru berdasarkan hasil rekomendasi analisis sistem. (Kusrini dkk, 2007).

Perancangan adalah penggambaran, perancangan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi, perencanaan sistem dapat dirancang dalam bentuk bagan alur sistem (*system flowchart*), yang merupakan alat bentuk grafik yang dapat digunakan untuk menunjukan urutan-urutan proses dari sistem. (Syifaun Nafisah, 2003).

### **2.2.2 Pengertian Aplikasi**

Aplikasi adalah suatu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktifitas seperti perniagaan, pelayanan, masyarakat, periklanan atau semua proses yang dilakukan manusia. (Henry, 2004).

Aplikasi adalah program yang memiliki aktifitas pemrosesan perintah yang diperlukan untuk melaksanakan permintaan pengguna dengan tujuan tertentu. (Supriyanto, 2005).

### **2.2.3 Pengertian Perancangan Aplikasi**

Konsep Merancang Multimedia merupakan aplikasi multimedia yang akan dibuat. Untuk dapat merancang konsep dalam membuat aplikasi multimedia dibutuhkan kreatifitas. Kreatifitas adalah kemampuan untuk menyajikan gagasan atau ide baru. Sedangkan inovasi merupakan aplikasi dari gagasan atau ide baru tersebut. Untuk menciptakan ide yang orisinil tidaklah mudah, maka dapat digunakan beberapa teknik untuk menciptakan ide, yaitu penyesuaian (adaptasi), multimedia yang telah ada dianggap belum sesuai dengan lingkungan yang dituju.

Merancang konsep analisis sistem bekerjasama dengan pemakai, mungkin juga bekerjasama dengan professional komunikasi seperti prosedur, sutradara, penulis naskah, editor elektronik terlibat dalam merancang konsep yang menentukan keseluruhan pesan dan membuat aliran (urutan) pada aplikasi multimedia yang akan dibuat. Untuk dapat merancang konsep dalam membuat aplikasi multimedia dibutuhkan kreatifitas. Kreatifitas adalah kemampuan untuk menyajikan gagasan atau ide baru. Sedangkan inovasi merupakan aplikasi dari gagasan atau ide baru tersebut. Untuk menciptakan ide yang orisinil tidaklah mudah, maka dapat digunakan beberapa teknik untuk menciptakan ide, yaitu penyesuaian (adaptasi).

### **2.2.4 Pengertian Sistem**

Sistem berasal dari bahasa Latin (*Systema*) dan bahasa Yunani (*Sustena*) adalah suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan. Pada prinsipnya, setiap sistem selalu terdiri atas empat elemen:

1. Objek yang dapat berupa bagian, elemen, ataupun *variable*. Ia dapat benda fisik, abstrak, ataupun keduanya sekaligus tergantung kepada sifat sistem tersebut.
2. Atribut yang menentukan kualitas atau sifat kepemilikan sistem dan objeknya.
3. Hubungan internal di antara objek-objek di dalamnya.
4. Lingkungan tempat di mana sistem berada.

#### **2.2.4.1 Karakteristik Sstem**

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah, dan sasaran atau tujuan. (Ladjamudin, 2013)

Adapun penjelasan dari masing-masing karakteristik sistem menurut tersebut adalah sebagai berikut:

##### **a. Komponen Sistem**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

##### **b. Batasan Sistem**

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

##### **c. Lingkungan Luar Sistem**

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan.

d. Penghubung Sistem

Penghubung sistem merupakan media yang menghubungkan antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini kemungkinan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya.

e. Masukan Sistem

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan. Sinyal input adalah energi yang diproses untuk mendapatkan keluaran dari sistem.

f. Keluaran Sistem

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem yang lain.

g. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

h. Sasaran Sistem

Suatu sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

### 2.2.5 Pengertian Informasi

Informasi merupakan hasil pengolahan data dari satu atau berbagai sumber yang kemudian diolah, sehingga menghasilkan nilai, arti, dan manfaat. Jadi dapat disimpulkan bahwa informasi adalah data yang telah diolah menjadi sebuah bentuk yang lebih berguna dan bermanfaat bagi yang menerimanya (Pratama E. A., 2014)

Definisi sistem menurut Mulyadi (2016:5), Sistem adalah “suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan”. Berdasar kan pengertian diatas dapat disimpulkan bahwa

sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai tujuan dalam melaksanakan suatu kegiatan pokok perusahaan.

#### **2.2.5.1 Komponen Sistem Informasi**

Komponen sistem informasi terdiri dari:

a. Blok Masukan

Blok masukan (*input*) ini mewakili data yang masuk kedalam sistem informasi. Termasuk dalam blok masukan ini adalah metode-metode dan media untuk menangkap data yang akan dimasukkan.

b. Blok Model

Blok model ini terdiri dari kombinasi antara prosedur, logika dan model matematika yang akan memanipulasi data inputan agar menghasilkan keluaran (*output*) seperti yang diinginkan.

c. Blok Keluaran

Blok keluaran (*output*) ini berupa informasi yang berkualitas serta dokumentasi yang bermanfaat untuk semua pemakai sistem.

d. Blok Teknologi

Blok teknologi adalah kotak alat (*tool-box*) dalam sistem informasi yang digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan output serta membantu pengendalian sistem secara keseluruhan. Teknologi ini terdiri dari 3 bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*).

e. Blok Basis Data

Basis data (*database*) merupakan kumpulan data yang saling terhubung, tersimpan dalam perangkat keras komputer. Yang mana diperlukan perangkat lunak untuk memanipulasinya agar menjadi informasi yang berkualitas.



f. Blok Kendali

Blok kendali adalah suatu perancangan yang diterapkan untuk mencegah terjadinya kesalahan-kesalahan, kegagalan sistem atau yang lainnya agar dapat segera teratasi.

### 2.2.6 Pengertian Web

Web adalah (Arief 2011a:7) salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, animasi, video) didalamnya yang menggunakan protocol HTTP (*Hypertext Transfer Protoco*) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut broser, semua dokumen web ditampilkan oleh browser dengan cara diterjemahkan. Beberapa jenis browser yang populer saat ini diantaranya adalah Internet Explorer yang diproduksi oleh Microsoft, Mozilla Firefox, Opera, dan Safari.

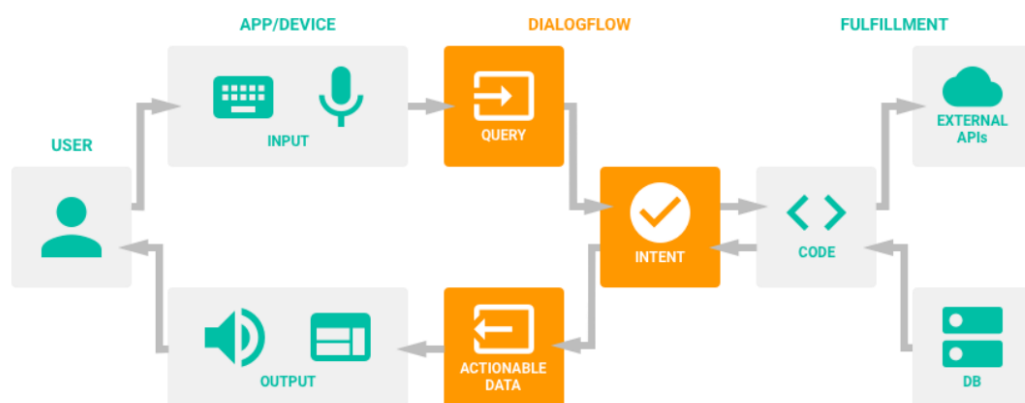
### 2.2.7 Google Dialogflow

Google Dialogflow adalah sebuah platform yang berbasis pengolahan bahasa alami untuk para pengembang perangkat lunak membuat sebuah sistem chatbot. Google Dialogflow sebelum diakuisi oleh perusahaan Google pada September 2016 bernama Api.ai. Api.ai awalnya dimiliki oleh perusahaan Speaktoit, sebuah perusahaan yang dikenal dengan aplikasi virtual buddy berbasis sistem operasi Android, iOS, dan windows phone yaitu Assistant. Perusahaan Google membeli Api.ai milik Speaktoit, karena platform tersebut menyediakan alat - alat untuk mengembangkan aplikasi - aplikasi pada Google *Virtual Assistant*. Pada 10 Oktober 2017 Api.ai resmi berganti nama menjadi Dialogflow. Berikut adalah fitur – fitur yang ditawarkan pada Google Dialogflow:

1. Dialogflow Agents

*Agents* mempunyai definisi yang sama seperti *Natural Language Understanding* (NLU). *Agents* dapat digunakan oleh sistem yang pengembang punya untuk dapat mentransformasi permintantaan bahasa alami dari pengguna sistem menjadi sebuah *actionable* data. Transformasi terjadi ketika masukan dari pengguna cocok dengan salah satu intent yang terdapat pada *agent* itu sendiri. *Intent* adalah komponen di *agent* yang

belum ditetapkan atau sudah ditetapkan oleh pengembang yang memproses permintaan – permintaan pengguna. *Agent* dapat didesain untuk mengatur sebuah alur percakapan dengan bantuan *context*, *intent priorities*, *slot filling*, *responsibilities*, dan *fulfillment* dengan perantara *webhook*. Fitur untuk mengatur pembelajaran mesin yang digunakan untuk sistem pengembang. Dalam menu ini terdapat fitur *match mode*, yaitu sebuah pengaturan yang menentukan algoritma yang harusnya digunakan untuk pembelajaran mesinnya. Algoritma yang telah ditentukan akan digunakan pada semua *intent*. Terdapat 2 algoritma yang disediakan oleh Dialogflow agent untuk pembelajaran mesinnya, yaitu: *Rule-based Grammar Matching (Hybrid)* dan *Machine Learning Matching*. Pada Gambar di bawah ini adalah alur kerja Dialogflow Agent yang merupakan alur kerja *agent* pada Google Dialogflow



Gambar 2. 1 Alur Kerja Dialogflow Agent

Sumber: <https://dialogflow.com/docs/images/overview/agents/overview.png>

## 2. Dialogflow Intent

*Intent* adalah sebuah representasi sebuah dialog antara apa yang pengguna katakan atau masukan dan apa tindakan atau balasan yang diberikan oleh sistem. *Intent* memiliki beberapa 9 bagian di dalamnya, yaitu: *Context*, *Events*, *Training Phrase*, *Action and Parameters* dan *Responses*.

### 3. Dialogflow *Entities*

*Entity* pada Dialogflow adalah sebuah alat yang berfungsi untuk mengekstrak parameter values dari masukan pengguna. *Entity* yang digunakan pada agent tertentu akan bergantung pada parameter *values*, karena *entity* akan memberikan output sesuai dengan fungsi parameter yang ada pada *intent*. Terdapat 3 tipe *entity* pada Dialogflow, yaitu: *System Entities*, *Developer Entities* dan *User Entities*.

### 4. Dialogflow *Tools*

Dialogflow *tools* merupakan sekumpulan alat – alat untuk membantu pengembangan sebuah *chatbot* pada Dialogflow. Dialogflow *tools* mempunyai 4 alat - alat, yaitu: *Fulfillment*, *Intergrations*, *Training*, *History*, *Analytics* dan *Prebuilt Agents*.

## 2.3 Aplikasi Pendukung

Berikut ini adalah beberapa aplikasi pendukung yang penulis gunakan dalam membangun sebuah Aplikasi dalam tugas akhir ini adalah sebagai berikut:

### 2.3.1 XAMPP

XAMPP adalah (Choliviana, Triyono, & Sukadi, 2012) paket PHP dan MySQL berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP.

Manfaat Xampp sebagai perangkat lunak bebas yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsi XAMPP sendiri adalah sebagai *server* yang berdiri sendiri (*localhost*). XAMPP merupakan perangkat yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server* Apache, PHP dan MySQL secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis.

### 2.3.2 PHP

PHP adalah (Arief 2011c:43) Bahasa *server-side –scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. Karena PHP merupakan *server-side-scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya akan dikirimkan ke *browser* dengan format HTML

PHP singkatan dari PHP *Hypertext Preprocessor* yang merupakan sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang *web* menulis halaman *web* dinamik dengan cepat. Hubungan PHP dengan HTML, halaman *web* biasanya disusun dari kode-kode HTML yang disimpan dalam sebuah file berekstensi .html. File HTML ini dikirimkan oleh *server* ke *browser*, kemudian *browser* menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah, Lain halnya dengan program PHP, Program ini harus diterjemahkan oleh *web-server* sehingga menghasilkan kode html yang dikirim ke *browser* agar dapat ditampilkan.

### 2.3.3 MySQL

MySQL adalah (Arief 2011:152) salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengolahan datanya”.

MySQL dikembangkan oleh perusahaan swedia bernama MySQL AB yang pada saat ini bernama Tcx DataKonsult AB sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak tahun 1979. Awalnya Tcx merupakan perusahaan pengembang *software* dan konsultan *database*, dan saat ini MySQL sudah diambil alih oleh Oracle Corp.

Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga bersifat *open source* (tidak berbayar). MySQL merupakan *database* yang pertama

kali didukung oleh bahasa pemrograman *script* untuk internet (PHP dan Perl). MySQL dan PHP dianggap sebagai pasangan *software* pembangun aplikasi *web* yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis web, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script* PHP.

#### 2.3.4 Codeigniter

CodeIgniter adalah Kerangka Pengembangan Aplikasi - *toolkit* - untuk orang yang membangun situs *web* menggunakan PHP. Tujuannya adalah untuk memungkinkan Anda mengembangkan proyek lebih cepat daripada yang Anda bisa jika Anda menulis kode dari awal, dengan menyediakan serangkaian perpustakaan yang kaya untuk tugas-tugas yang biasanya dibutuhkan, serta antarmuka yang sederhana dan struktur logis untuk mengakses perpustakaan ini. CodeIgniter memungkinkan Anda secara kreatif fokus pada proyek Anda dengan meminimalkan jumlah kode yang diperlukan untuk tugas yang diberikan.

Menurut Hakim (2010:3) CodeIgniter adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis PHP dibandingkan jika menulis semua kode program dari awal.

Jadi CodeIgniter adalah sebuah *framework* buatan Rick Ellis yang digunakan untuk mempermudah pada *developer* dalam mengembangkan suatu aplikasi *web*.

#### 2.3.5 Bootstrap

Bootstrap sebuah *library framework* CSS yang dibuat khusus untuk bagian pengembang *front-end* website. Menurut Alatas (2013:2) “bootstrap merupakan *framework* ataupun *tools* untuk membuat aplikasi *web* ataupun situs *web responsive* secara tepat, mudah dan gratis”. Lain pula menurut Riyanto (2014:18)

#### 2.3.6 Sublime Text

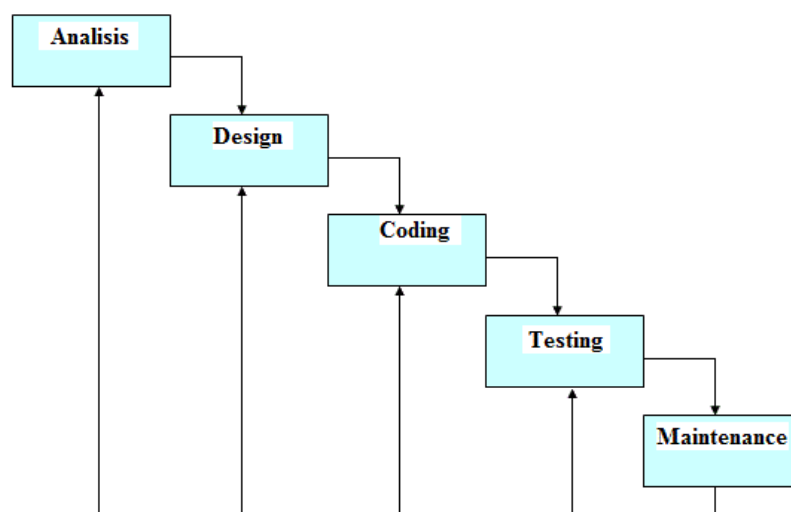
Sublime text salah satu kode editor yang biasa digunakan oleh para programmer untuk membuat suatu program.

Menurut Supono dan Putratama (2016:14) “*Sublime text* merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-edit suatu aplikasi. *Sublime text* mempunyai fitur plugin tambahan yang memudahkan programmer”.

Selain itu, menurut Faridi (2015:3) menjelaskan bahwa “*Sublime Text 3* adalah editor berbasis python, sebuah teks editor yang elegan, kaya akan fitur, *cross platform*, mudah dan simple yang cukup terkenal di kalangan developer (pengembang), penulis dan desainer”.

## 2.4 Pengertian Metode Waterfall

Metode *Waterfall* adalah suatu metodologi pengembangan perangkat lunak yang mengusulkan pendekatan kepada perangkat lunak sistematis dan sekuensial yang mulai pada tingkat kemajuan sistem pada seluruh analisis, design, kode, pengujian dan pemeliharaan. Model pengembangan sistem yang digunakan adalah *Waterfall Strategy*. Strategi ini mengisyaratkan ‘penyelesaian’ tiap proses satu per satu (Whitten et al, 2004), Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, desain, coding, *testing* atau *verification*, dan *maintenance*. Sebagai contoh tahap desain harus menunggu selesainya tahap sebelumnya yaitu tahap requirement. Berikut gambar model waterfall.



Gambar 2. 2 Metode *Waterfall*


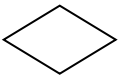
## 2.5 Teori Perancangan Basis Data

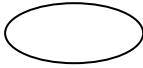


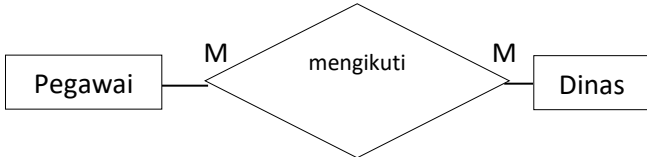
Menurut (Kristanto, 2012), Basis data adalah Suatu sistem penyusunan dan pengelolaan *record-record* dengan menggunakan komputer dengan tujuan untuk menyimpan serta memelihara data operasional lengkap suatu perusahaan atau organisasi, sehingga mampu menyediakan informasi yang optimal yang diperlukan maka untuk proses pengambilan keputusan.

Untuk mengelola basis data diperlukan perangkat lunak yang disebut *Database Management System* (DBMS). DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol dan mengakses basis data dengan cara yang praktis dan efisien. DBMS juga dapat digunakan untuk mengakomodasikan berbagai macam pemakai yang memiliki kebutuhan akses yang berbeda-beda.

### 2.5.1 ERD (*Entity Relationship Diagram*)

ERD atau *Entity Relationship Diagram* merupakan suatu model jaringan yang menggunakan data yang disimpan pada sistem secara abstrak. ERD juga menggambarkan hubungan antara entitas yang memiliki jumlah atribut dengan entitas yang lain dalam suatu sistem yang terintegrasi. ERD digunakan oleh perancang sistem untuk memodelkan data yang nantinya akan dikembangkan menjadi basis data (database). Model data ini juga akan membantu pada saat melakukan analisis dan perancangan basis data, karena model data ini akan menunjukkan bermacam-macam data yang dibutuhkan dan hubungan antara data. ERD juga merupakan model konseptual yang dapat mendeskripsikan hubungan antara file yang akan digunakan untuk memodelkan struktur data serta hubungan antara data (Membara, 2014).

Notasi	Keterangan
Entitas 	<b>Entitas</b> adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
Relasi 	<b>Relasi</b> menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.

Atribut		<b>Atribut</b> yaitu berfungsi sebagai mendeskripsikan karakter entitas (atribut yang berfungsi sebagai key diberi garis bawah).
Garis		<b>Garis</b> penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.
		<b>One to Many</b> adalah perbandingan antara entity pertama dengan entity kedua berbanding satu berbanding banyak.
		<b>Many to Many</b> yaitu perbandingan antara entity pertama dengan entity kedua berbanding banyak berbanding banyak.

Tabel 2. 1 ERD ( *Entity Relationship Diagram* )

### 2.5.2 LRS (*Logical Record Structure*)

LRS (*Logical Record Structure*) adalah representasi dari struktur *record* pada table-table yang terbentuk dari hasil antar himpunan entitas. LRS dibentuk dengan nomor dari tipe *record*. LRS terdiri dari link-link diantara tipe *record*. *Link* ini menunjukkan arah dari satu tipe *record* lainnya. (Riyanto, 2005).

## 2.6 *Unified Modelling Language (UML)*

Menurut (Pandawa, 2016) UML (*Unified Modelling Language*) adalah bahasa garis untuk mendokumentasikan menspesifikasikan, dan membangun system perangkat lunak UML berorientasi objek menerapkan banyak level abstraksi, tidak tergantung pada Bahasa dan teknologi, pemanduan beberapa notasi diberagam metodologi, usaha bersama dari banyak pihak. Standar UML dikelola oleh OMG (*Object Management Group*).

Sedangkan menurut (Akil, 2018, p. 3), *Unified Model Language (UML)* adalah bahasa pemodelan visual yang digunakan untuk mendefinisikan, memvisualisasikan, membangun dan mendokumentasikan rancangan dari suatu sistem perangkat lunak.


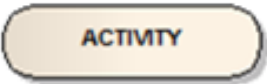

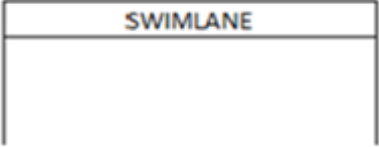





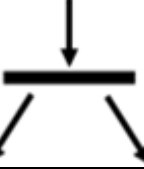


Pada UML terdiri dari beberapa macam diagram berikut penjelasan masing – masing diagram yang penulis gunakan dalam mendesain sistem:

### 2.6.1 Activity Diagram

*Activity Diagram* ialah sebuah diagram yang digunakan untuk menggambarkan jalur aktivitas pada sistem yang sedang dirancang. *Activity diagram* juga digunakan untuk menggambarkan *decision* yang mungkin terjadi pada saat sistem dirancang serta digunakan untuk menggambarkan sebuah proses parallel yang mungkin terjadi pada saat eksekusi.

Menurut (Anatasia, 2018, p. 49), *Activity Diagram* yaitu aktivitas yang menggambarkan urutan kegiatan atau urutan aktivitas dari sebuah sistem. Tujuan dibuatnya *activity diagram* adalah untuk memudahkan dalam memahami proses bisnis sistem. Adapun simbol – simbol *activity diagram* dapat dilihat pada tabel berikut:


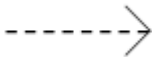
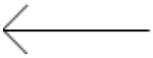
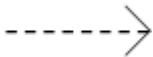
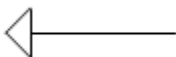
No.	Simbol	Keterangan
1		<i>Start</i> merupakan status awal sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2		<i>Activity</i> merupakan aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3		<i>Decision</i> merupakan asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari 1.
4		<i>Swimlane</i> digunakan untuk memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
5		<i>Finish</i> merupakan status akhir sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6		<i>Asosiasi</i> penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.






7		<i>Join</i> digunakan untuk menunjukkan kegiatan yang digabungkan.
8		<i>Fork</i> digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel.
9		<i>Miracle Activities</i> digunakan untuk masukan dan keluaran, dipakai pada waktu <i>start point</i> .
10		<i>Black Hole Activities</i> digunakan untuk masukan dan keluaran.

Tabel 2. 2 Simbol Activity Diagram

### 2.6.2 Use Case Diagram

Menurut (Anatasia, 2018, p. 45), *Use Case Diagram* adalah diagram yang menggambarkan interaksi antara pengguna (*actor*) dengan sistem informasi yang akan dibuat. *Use case* menggambarkan siapa saja actor yang terlibat dan fungsi apa saja yang dapat digunakan *actor* pada sistem informasi tersebut. *Use Case Diagram* terdiri dari *Use Case*, *Actor*, *Relationship*, *System Boundary* atau batas sistem (opsional). Adapun simbol – simbol *Use Case Diagram* dapat dilihat pada tabel berikut:


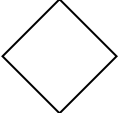
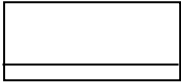

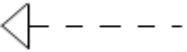
No	Gambar	Nama	Keterangan
1		Actor	Menspesifikasi antara himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case.
2		Depedency	Hubungan perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
3		<i>Generalization</i>	Hubungan dimana objek anak( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
4		Include	Menspesifikasikan atau bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan atau bahwa <i>use case</i> target memperluas perilaku dari

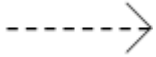

			<i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa sajakah yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasi paket yang menampilkan sistem secara terbatas.
8		<i>Use case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerjasama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemen.
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 3 Simbol *Use Case* Diagram

### 2.6.3 Class Diagram

Menurut (Anatasia, 2018, p. 56), *Class diagram* menggambarkan struktur dan deskripsi *class*, *package*, dan objek serta hubungan satu sama lain seperti *inheritance*, *association*, dan lain – lain. Dibawah ini adalah gambaran dari simbol - simbol *class* diagram:


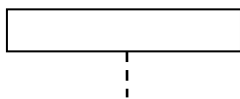
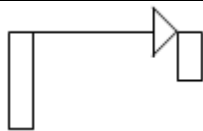
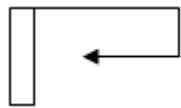
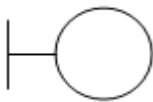

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagai perilaku dan struktur data dari objek yang ada diatasnya objek induk ( <i>ancestor</i> ).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.

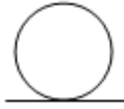
6		<i>Dependency</i>	Hubungan yang dimana ada perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek yang lainnya.

Tabel 2. 4 Simbol *Class Diagram*

#### 2.6.4 *Sequence Diagram*

Menurut (Anatasia, 2018, p. 51), *Sequence Diagram* menggambarkan interaksi antar objek dalam sistem. *Sequence Diagram* digunakan untuk menggambarkan scenario pada use case. Jumlah *Sequence Diagram* harus sama dengan jumlah use case. Adapun simbol – simbol *Sequence Diagram* dapat dilihat pada gambar dibawah ini:

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
4		<i>Message to Self</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
5		<i>Boundary</i>	Untuk menunjukkan adanya <i>interfaces requirement</i> , bukan untuk menunjukkan bagaimana <i>interface</i> yang akan di implementasikan.
6		<i>Controller</i>	Sebuah control adalah meniru suatu kelas yang menggambarkan suatu

			pengendalian manajer atau kesatuan. Suatu control mengorganisir dan menjadwalkan aktivitas.
7		Entity	Suatu entity adalah tempat atau ketekunan mekanisme yang menangkap pengetahuan dan informasi di dalam suatu sistem.

Tabel 2. 5 Simbol *Sequence Diagram*

## 2.7 Notasi UML

Notasi UML dturunkan dari tiga notasi yang sudah ada sebelumnya yaitu Grady Booch OOD (*Object Oriented Design*), Jim Rumbough OMT (*Object Modelling Technique*), dan Ivar Jacobson OOSE (*Object Oriented Software Engineering*). Notasi UML yang digunakan sekarang merupakan penggabungan dari tiga notasi tersebut, yang terdiri dari: (Nurhasanah, 2014)

### 1. Aktor (*Actor*)

Dalam pemodelan sistem dengan UML, actor adalah seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang kita kembangkan. *Actor* berada diluar lingkup sistem atau perangkat lunak yang sedang kita kembangkan, bersifat eksternal.

Secara prinsip dapat kita kenali 3 jenis actor untuk hampir semua sistem atau perangkat lunak yang kita kembangkan: para pengguna sistem, perangkat lunak yang berinteraksi dengan sistem, dan perangkat lunak yang kita kembangkan serta waktu. Jadi aktor ini bisa berupa orang, perangkat keras atau mungkin juga objek lain dalam sistem yang sama. Biasanya yang dilakukan oleh aktor adalah memberikan informasi pada sistem dan atau memerintahkan sistem untuk melakukan sesuatu.

Kelas seperti juga objek, adalah sesuatu yang membungkus (encapsulate) informasi dan perilaku dalam dirinya. Dalam pengembangan system tradisional, kita mengadakan pendekatan dengan cara memisahkan informasi-informasi pada sisi basis data dan perilaku yang mengaksesnya di sisi aplikasi pemasup atau pengakses. Pendekatan berorientasi objek berbeda, yaitu menggabungkan potongan-potongan informasi dengan perilaku yang akan mengaksesnya dalam apa yang dinamakan kelas.

## 2. *Usecase*

*Usecase* adalah peringkat tertinggi dari fungsional yang dimiliki system. Dengan kata lain, *usecase* menggambarkan bagaimana seseorang akan menggunakan sistem. *Usecase* menjelaskan suatu urutan kegiatan yang dilakukan oleh aktor dan sistem untuk mencapai tujuan tertentu walaupun menjelaskan kegiatan namun usecase hanya menjelaskan apa yang dilakukan oleh aktor dan sistem, bukan bagaimana aktor dan sistem melakukan kegiatan tersebut.

Keunggulan dari cara memandang sistem sebagai kumpulan usecase adalah kemampuannya untuk memisahkan implementasi sistem dari alasan mengapa sistem harus ada. Ia akan membantu kita untuk berfokus pada apa yang paling penting, yaitu menentukan apa yang dibutuhkan serta apa harapan pengguna terhadap sistem atau perangkat lunak yang sedang dikembangkan.

## 3. *Interaction*

*Interaction* digunakan untuk menunjukan baik aliran pesan atau informasi antara objek. Biasanya interection ini dilengkapi juga dengan teks bernama operationsignature yang tersusun dari nama operasi, parameter yang dikirim dan tipe parameter yang dikembalikan.

## 4. *Interface*

*Interface* merupakan kumpulan operasi tanpa implementasi dari suatu class. Implementasi operasi dalam *interface* dijabarkan dalam operasi dalam *class*. Oleh karena itu keberadaan *interface* selalu disertai oleh *class* yang mengimplementasikan operasinya. *Interface* ini merupakan salah satu cara mewujudkan prinsip enkapsulasi dalam objek.

## 5. *Package*

*Package* adalah *container* atau wadah konseptual yang digunakan untuk mengelompokkan elemen-elemen dari suatu sistem yang sedang dibangun, sehingga bisa dibuat model yang lebih sederhana. Tujuannya

adalah untuk mempermudah penglihatan (*visibility*) dari suatu model yang sedang dibangun.

#### 6. *Note*

*Note* dibangun untuk memberikan keterangan dan komentar tambahan dari suatu elemen sehingga bisa langsung terlampir dalam model. *Note* ini bisa ditempelkan ke semua elemen notasi yang lain.

#### 7. *Dependency*

Merupakan relasi yang menunjukkan bahwa perubahan pada suatu elemen memberikan pengaruh pada elemen yang lain. Elemen yang ada di bagian tanda panah adalah elemen yang tergantung pada elemen yang ada di bagian tanpa ada tanda panah. Terdapat dua *stereotype* dari *dependency*, yaitu *include* dan *extend*. *Include* menunjukkan bahwa suatu bagian dari elemen (yang ada di garis tanpa panah) memicu eksekusi bagian dari elemen yang lain (yang ada di garis dengan panah), misalnya untuk notasi A B operasi yang ada di class A memicu dieksekusinya operasi yang berada di class B.

*Extend* menunjukkan bahwa suatu bagian dari elemen di garis tanpa panah bisa disiapkan kedalam elemen yang ada di garis dengan panah, misalnya untuk notasi A B suatu fungsi dari *usecase* A bisa disisipkan ke dalam *usecase* B atau dengan kata lain A optimal untuk B. Kedua *stereotype* ini di representasikan dengan menambahkan *text include* atau *extend* di notasi *dependency*.

#### 8. *Association*

*Association* menggambarkan navigasi antara *class* (*navigation*), berapa banyak objek lain yang bisa berhubungan dengan satu objek (*multiplicity* antar *class*) dan apakah suatu *class* menjadi bagian dari *class* lainnya (*aggregation*).

*Navigation* di lambangkan dengan penambahan tanda panah di akhir garis. *Bidirectional navigation* menunjukkan bahwa dengan mengetahui salah satu class bisa didapatkan dari informasi lainnya.

Sementara dengan *unidirectional navigation* hanya dengan mengetahui *class* di ujung garis *association* tanpa panah kita bisa mendapatkan informasi dari *class* di ujung dengan panah, tetapi tidak sebaliknya.

Pada penelitian kali ini penulis menggunakan visual paradigm untuk merancang aplikasi. Visual paradigm adalah perangkat lunak yang digunakan untuk perancangan aplikasi dengan tools UML.

## **2.8 Pengujian Perangkat Lunak**

Pengujian adalah aktivitas yang digunakan untuk dapat melakukan evaluasi suatu atribut atau kemampuan dari program atau sistem dan menentukan apakah telah memenuhi kebutuhan atau hasil yang diharapkan. (Nidhra and Dondetti, 2012)

Menurut Munawar (2018, p. 38), Pengetesan *software* adalah kegiatan yang sangat penting sebelum diimplementasikan. Beberapa organisasi besar, biasanya memiliki bagian khusus untuk pengetesan *software* sebelum diimplementasikan. Namun tidak semua organisasi memiliki bagian khusus untuk pengetesan *software*. Oleh karena itu, beberapa organisasi melibatkan analisis dalam pengetesan, karena analisis lebih memahami kebutuhan organisasi sehingga bisa mengatur kinerja *software* dari sisi kebutuhan fungsional dan non-fungsional.

Beberapa metode yang biasa digunakan untuk melakukan pengujian antara lain *Black Box* dan *White Box Testing*.

### **2.8.1 Black Box Testing**

*Black Box Testing* merupakan sebuah metode yang digunakan untuk menemukan kesalahan dan mendemonstrasikan fungsional aplikasi saat dioperasikan, apakah input diterima dengan benar dan output yang dihasilkan telah sesuai dengan yang diharapkan. (Nidhra and Dondetti, 2012).

Pengujian perangkat lunak mempunyai beberapa level, untuk pengujian menggunakan metode *Black Box*, terdapat enam level yaitu *Integration*, *Functional*, *System*, *Acceptance*, *Beta*, dan *Regression*.



Salah satu dari pengujian *Black Box* yang dapat dilakukan oleh seorang penguji independen adalah *Functional Testing*. Basis uji dari *Functional Testing* ini adalah pada spesifikasi dari komponen perangkat lunak yang diuji.

Fokus dari pengujian menggunakan metode *Black Box* adalah pada pengujian fungsionalitas dan output dihasilkan aplikasi. Pengujian *Black Box* didesain untuk mengungkap kesalahan pada persyaratan fungsional dengan mengabaikan mekanisme internal atau komponen dari suatu program. *Functional Testing* memastikan bahwa semua kebutuhan – kebutuhan telah dipenuhi dalam sistem aplikasi. Dengan demikian fungsinya adalah tugas – tugas yang didesain untuk dilaksanakan sistem. *Functional Testing* berkonsentrasi pada hasil dari proses bukan bagaimana proses terjadi

*Black Box Testing* cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.

Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut:

1. Bagaimana fungsi-fungsi diuji agar dapat dinyatakan *valid*?
2. Input seperti apa yang dapat menjadi bahan kasus uji yang baik?
3. Apakah sistem sensitif pada *input-input* tertentu?
4. Bagaimana sekumpulan data dapat diisolasi?
5. Berapa banyak rata-rata dan jumlah data yang dapat ditangani sistem?
6. Efek apa yang dapat membuat kombinasi data ditangani spesifik pada operasi sistem?

Saat ini terdapat banyak metode atau teknik untuk melaksanakan *Black Box Testing*, antara lain:

1. *Equivalence Partitioning*
2. *Boundary Value Analysis/Limit Testing*
3. *Comparison Testing*
4. *Sample Testing*
5. *Robustness Testing*

6. *Behavior Testing*
7. *Requirement Testing*
8. *Performance Testing*
9. Uji Ketahanan (*Endurance Testing*)
10. Uji Sebab-Akibat (*Cause-Effect Relationship Testing*)

## **BAB III**

### **ANALISA DAN PERANCANGAN**

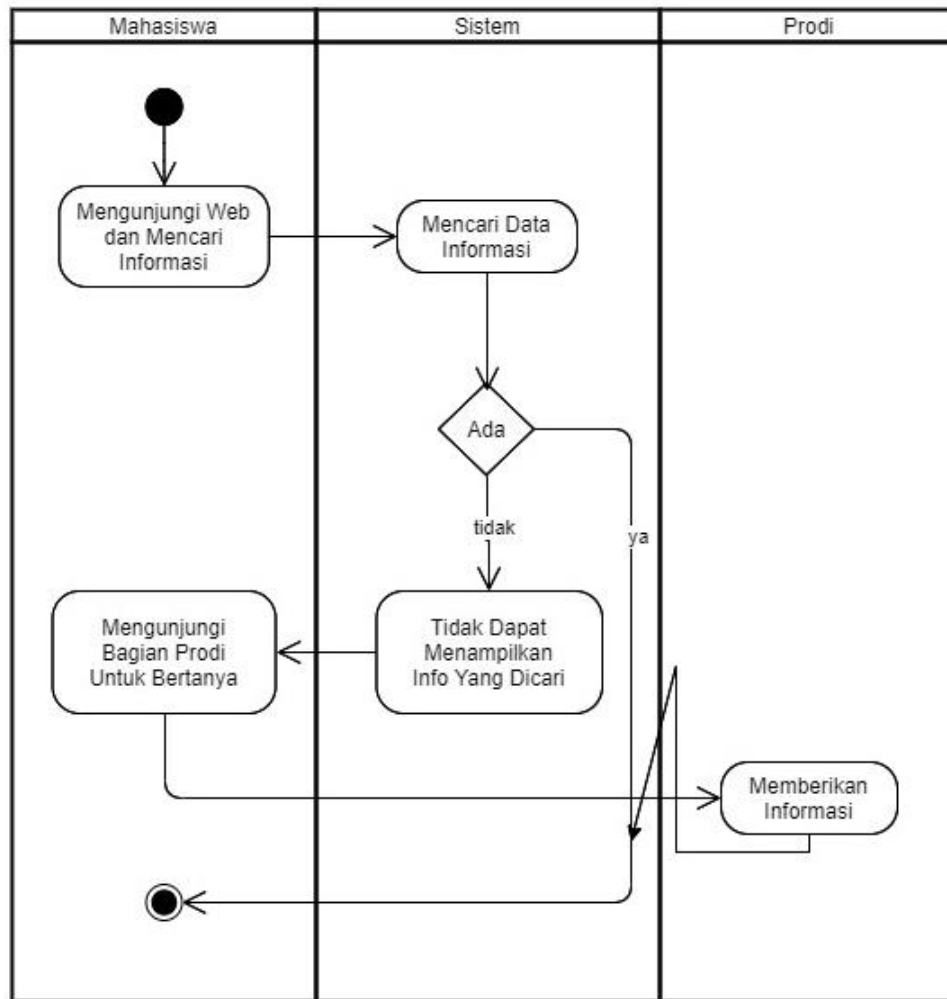
#### **3.1 Analisa Sistem**

Tahap analisis dilakukan setelah tahap perencanaan system dan sebelum tahap desain system. Tahap ini merupakan tahap yang kritis karena kesalahan dalam tahap ini menyebabkan kesalahan pada tahap selanjutnya.

Analisa system informasi akan membantu dalam mengetahui informasi – informasi tentang system yang sedang berjalan. Sehingga dengan analisa system, diharapkan dapat diketahui sejauh mana kebutuhan yang telah ditangani oleh system yang berjalan dan bagaimana agar kebutuhan – kebutuhan yang belum dapat terpenuhi dapat diberikan solusinya dan dditerapkan dalam tahapan perancangan system.

##### **3.1.1 Analisa Sistem Berjalan**

Analisa sistem berjalan saat ini pada proses pencatatan kesehatan ibu dan anak dapat digambarkan dengan activity diagram sebagai berikut:



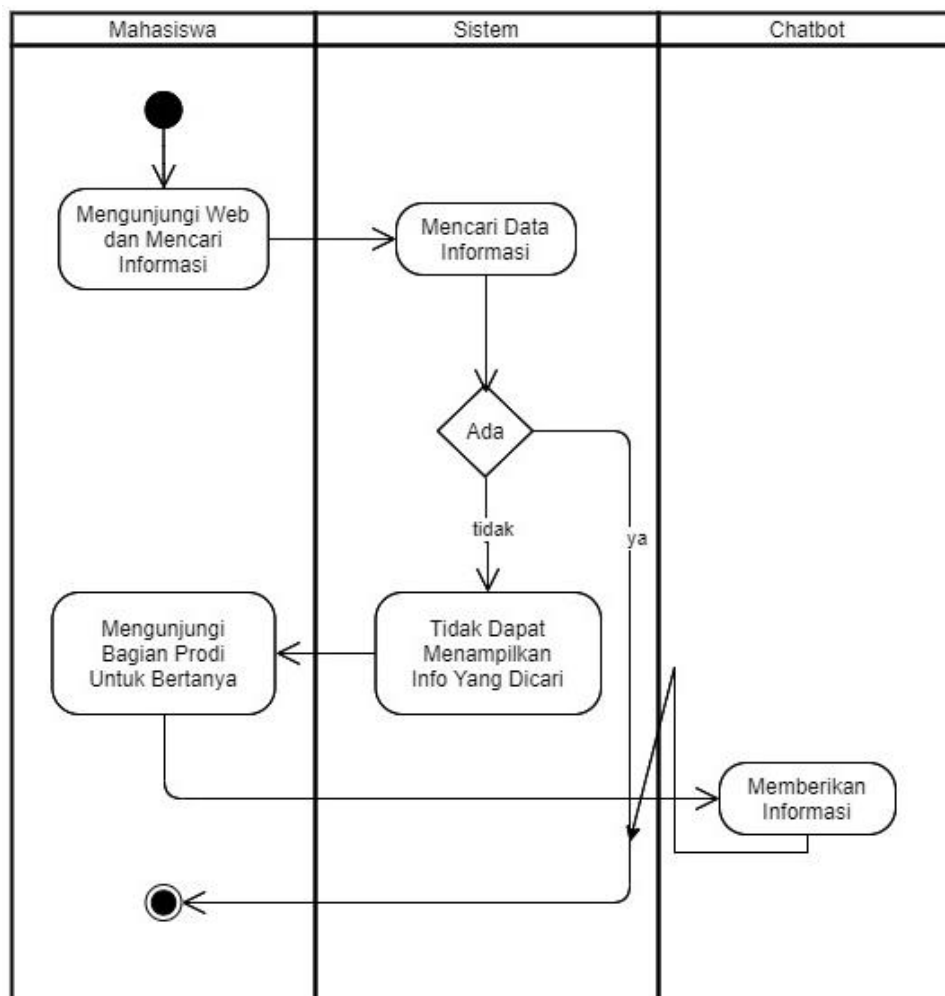
Gambar 3. 1 Activity Diagram Sistem Yang Berjalan

Gambar diagram diatas menjelaskan sistem yang berjalan di Universitas Pamulang, dimulai mahasiswa yang mencari informasi yang mereka butuhkan dari mulai dengan mengunjungi *web* sistem informasi Universitas Pamulang, dan jika mahasiswa tersebut tidak menemukan informasi yang mereka cari, mereka akan bertanya langsung ke bagian prodi Teknik Informatika di Universitas Pamulang, yang dimana tidak efektif untuk jarak dan waktu.

### 3.1.2 Analisa Sistem Usulan

Analisa adalah penguraian suatu pokok sistem yang utuh atas berbagai bagiannya untuk memecahkan suatu masalah dengan mengidentifikasi dan mengevaluasi permasalahan. Dalam tahap analisa sistem diantaranya identifikasi masalah, memahami kerja sistem yang ada, menganalisis sistem, dan membuat laporan hasil analisis.

Dibawah ini adalah gambaran sistem usulan yang akan peneliti implementasikan pada proses pencarian informasi pada Sistem Informasi Skripsi Teknik Informatika di Universitas Pamulang:



Gambar 3. 2 Activity Diagram Sistem Yang Diusulkan

Pada gambar diagram system yang diusulkan oleh peneliti, hanya pada bagian prodi yang digantikan oleh *Chatbot* akan lebih efektif dalam jarak dan

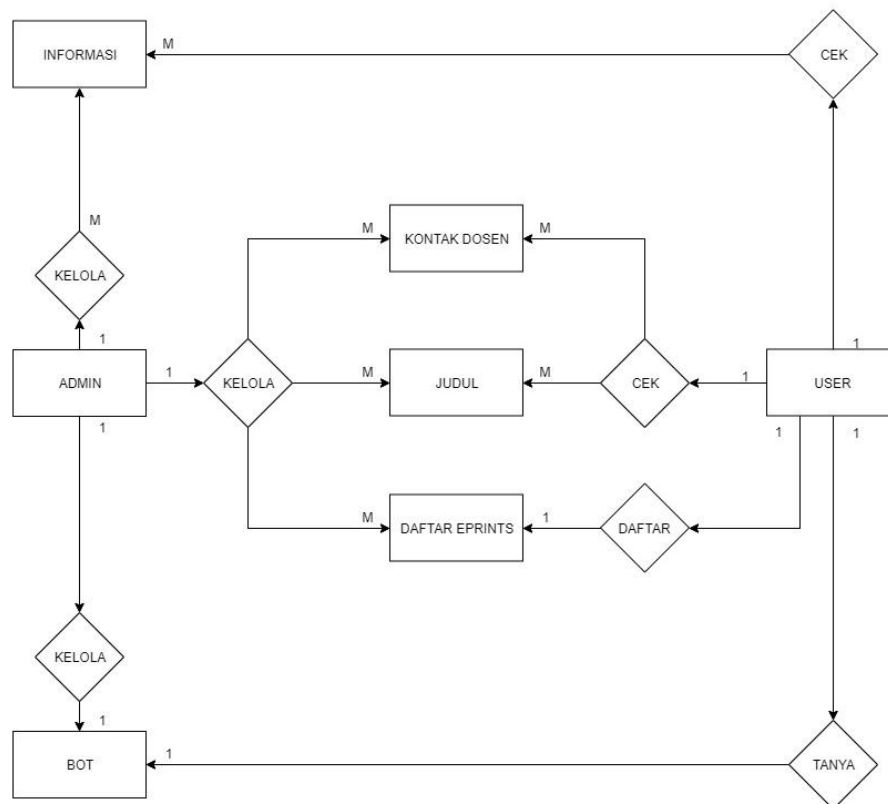
waktu dan interaktif tanpa mengharuskan mahasiswa datang ke bagian prodi Teknik Informatika di Universitas Pamulang,

### 3.2 Perancangan Basis Data

Dalam membuat system ini, ada database sebagai media penyimpanan data. Berikut adalah beberapa rancangan dalam membuat database dari system yang diusulkan.

#### 3.2.1 Entity Relationship Diagram (ERD)

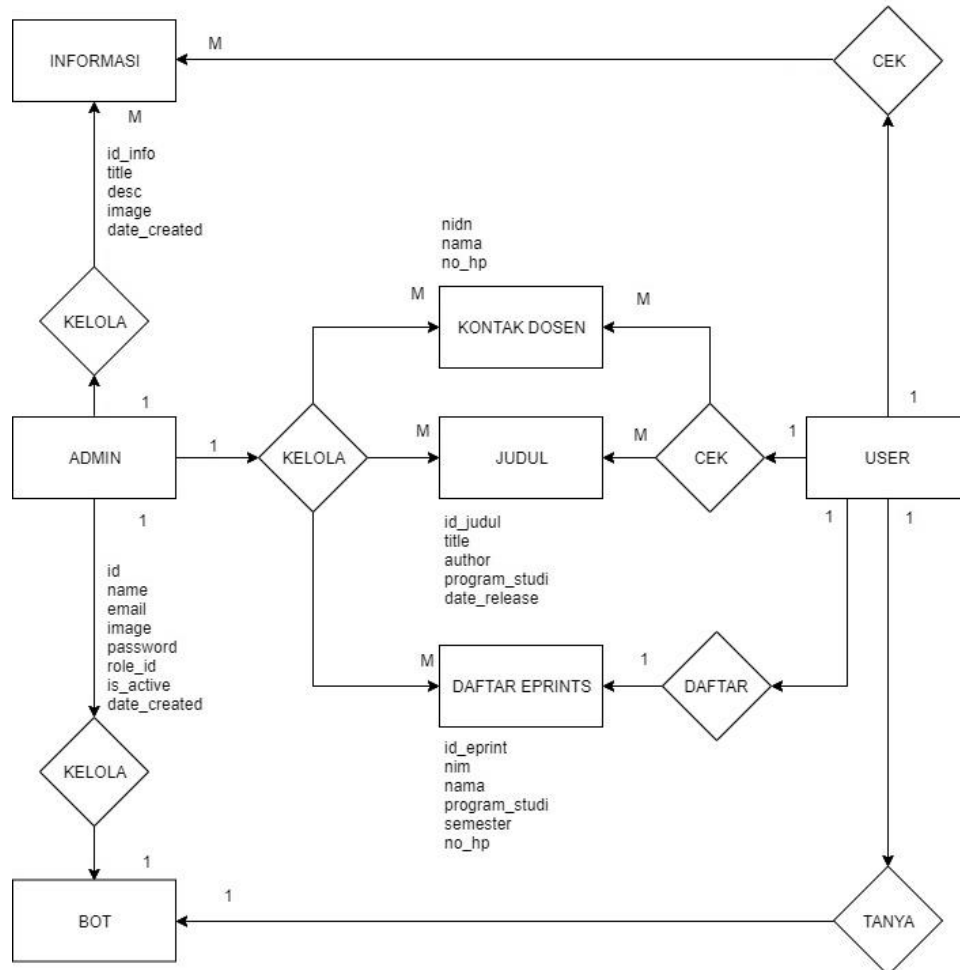
Diagram erd dibawah ini menggambarkan struktur database yang akan dibangun dalam aplikasi ini.



Gambar 3. 3 Entity Relationship Diagram

### 3.2.2 Transformasi ERD Ke *Logical Record Structure (LRS)*

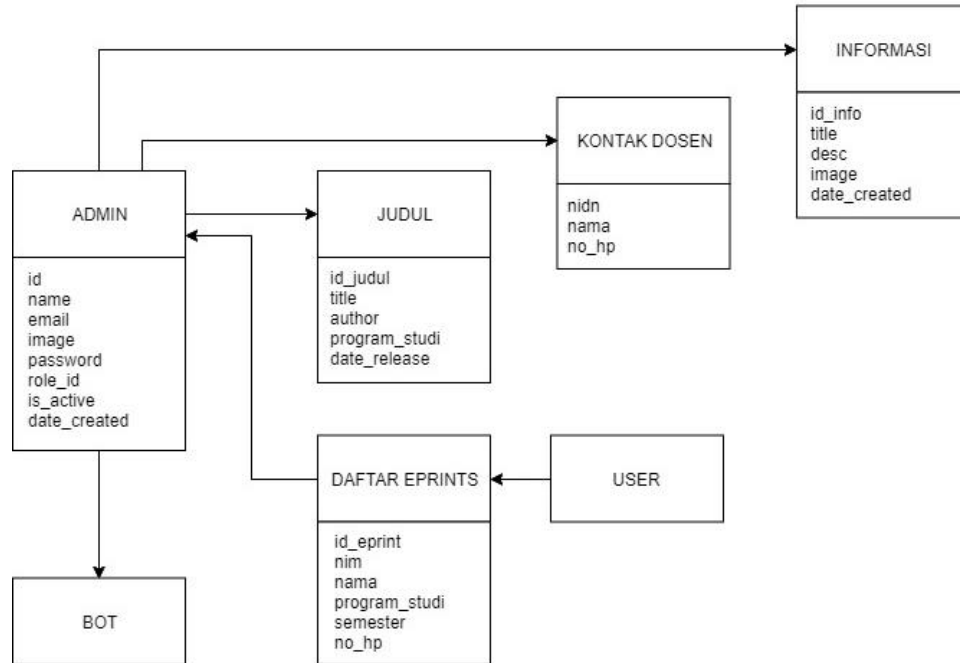
Berikut adalah transformasi *entitas relationship diagram* (LRS) ke *logical record structure* (LRS) digambarkan sebagai berikut:



Gambar 3. 4 ERD ke *Logical Record Structure*

### 3.2.3 Logical Record Structure (LRS)

Berikut adalah *logical record structure* (LRS) digambarkan sebagai berikut:



Gambar 3. 5 Logical Record Structure

### 3.2.4 Spesifikasi Basis Data

Berikut spesifikasi basis data yang menjelaskan masing-masing relasi berisi nama file, media penyimpanan yang dipakai, isi atau keterangan dari file, primary key, panjang record, jumlah record, dan struktur file.

a. Nama Tabel: user

No	Nama Field	Type	Length	Keterangan
1	Id	int	11	Primary Key
2	Name	varchar	128	Nama Admin
3	Email	varchar	128	Email Admin
4	Password	varchar	256	Password Admin
5	role_id	Int	11	Role Access
6	is_active	Int	1	Kondisi Akun



7	<i>date_created</i>	Int	11	Tanggal Daftar
---	---------------------	-----	----	----------------

Tabel 3. 1 Tabel user

b. Nama Tabel: *post\_dosen*

No	Nama Field	Type	Length	Keterangan
1	Nidn	Int	20	<i>Primary Key</i>
2	Nama	Varchar	100	Nama Dosen
3	Hp	Varchar	50	Kontak Dosen

Tabel 3. 2 Tabel *post\_dosen*

c. Nama Tabel: *post\_article*

No	Nama Field	Type	Length	Keterangan
1	id_info	Int	11	<i>Primary Key</i>
2	<i>Title</i>	Varchar	128	Judul Info
3	<i>Desc</i>	Text		Deskripsi Info
4	<i>Image</i>	Varchar	256	Gambar Info
5	<i>date_created</i>	Int	11	Tanggal Info Dibuat

Tabel 3. 3 Tabel *post\_article*

d. Nama Tabel: *post\_eprints*

No	Nama Field	Type	Length	Keterangan
1	id_eprint	Int	11	<i>Primary Key</i>
2	Nim	Varchar	128	<i>NIM User</i>
3	Nama	Varchar	256	<i>Nama User</i>
4	program_studi	Varchar	256	<i>Program Studi User</i>
5	Semester	Varchar	128	<i>Semester User</i>
6	no_hp	Varchar	256	<i>No Hp User</i>

Tabel 3. 4 Tabel *post\_eprints*

e. Nama Tabel: *post\_judul*

No	Nama Field	Type	Length	Keterangan
1	id_judul	Int	11	<i>Primary Key</i>
2	<i>Title</i>	Varchar	512	Judul Skripsi
3	<i>Author</i>	Varchar	256	Nama Penulis
4	program_studi	Varchar	256	Program Studi
5	<i>Date_release</i>	Int	11	Tanggal Rilis

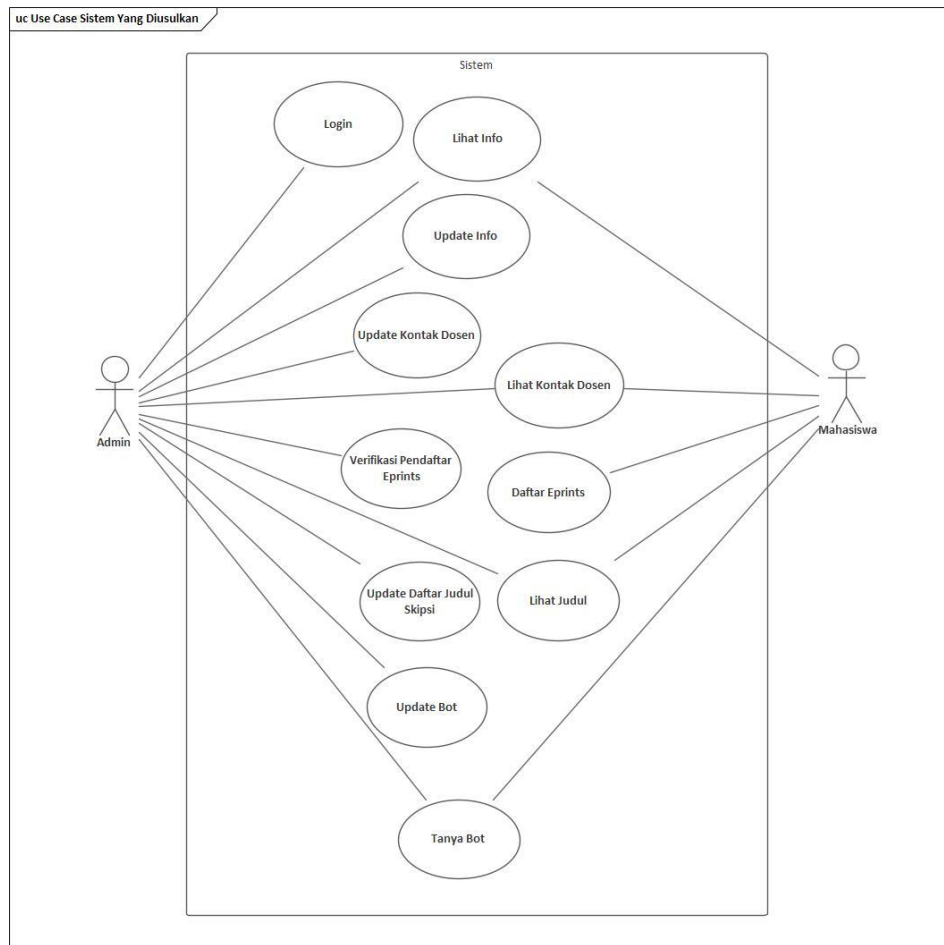
Tabel 3. 5 Tabel *post\_judul*

### 3.3 Pengajuan *Unified Modelling Language* (UML)

Perancangan pengembangan perangkat dengan menggunakan metode grafis serta merupakan bahasa untuk visualisasi, spesifikasi, konstruksi serta dokumentasi. *unified modelling language* (UML) meliputi perancangan *usecase* diagram, *activity* diagram, *class* diagram, *sequence* diagram yang akan digunakan.

#### 3.3.1 *Use Case* Diagram

*Use Case* merupakan gambaran skenario dari interaksi antara *user* dengan sistem. Sebuah *use case diagram* menggambarkan hubungan antara aktor (pengguna) dan kegiatan yang dapat dilakukannya terhadap aplikasi. Berikut ini adalah *Use case diagram* yang memperlihatkan peranan *actor* dalam interaksinya dengan sistem.

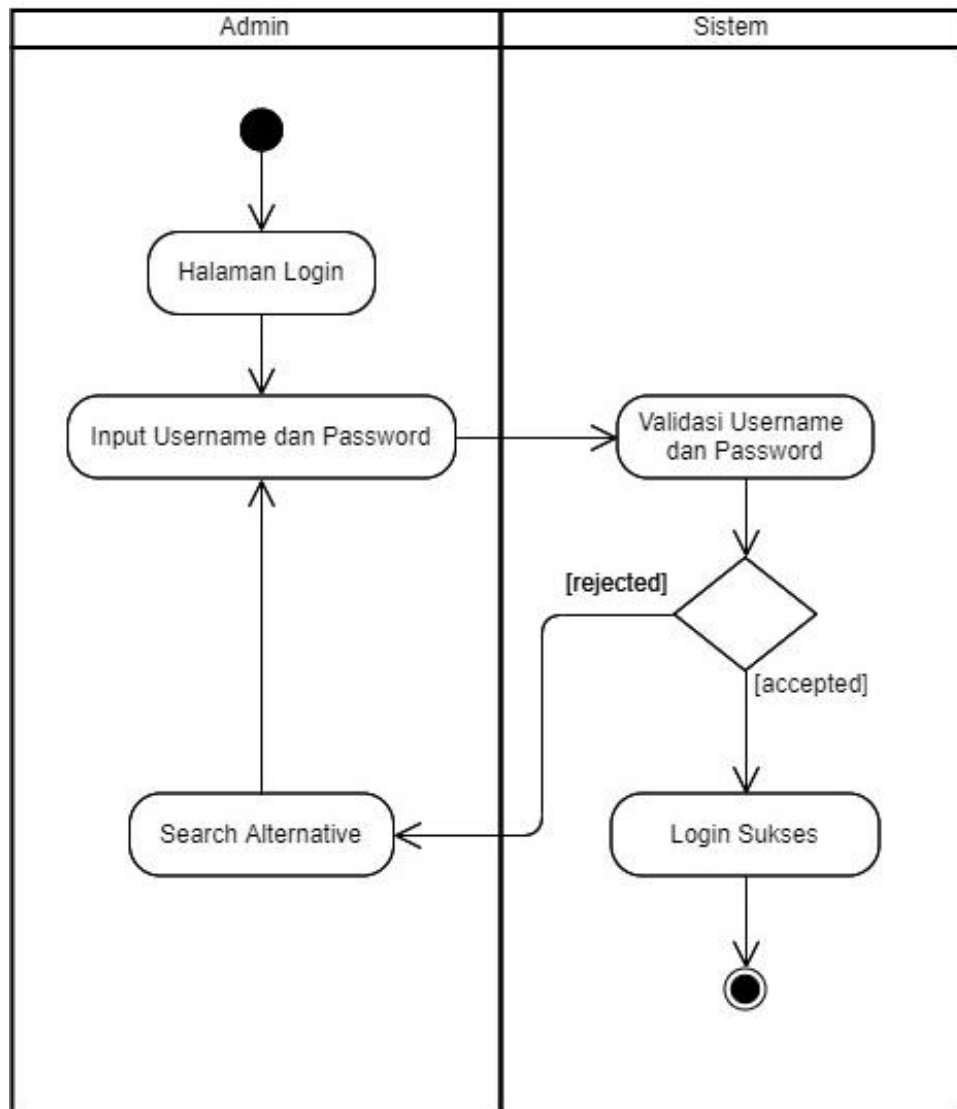


Gambar 3. 6 Use Case Diagram

### 3.3.2 Activity Diagram

*Activity* diagram bertujuan sebagai gambaran sebuah alur kerja pada aplikasi berdasarkan aktifitas yang dilakukan oleh pengguna terhadap sistem. Pada penelitian ini *activity* diagram yang akan dibuat mempunyai alur yang sama dengan setiap kemungkinan percakapan yang akan terjadi. Pada Gambar dibawah *Activity* Diagram merupakan *activity* diagram yang terdapat pada aplikasi:

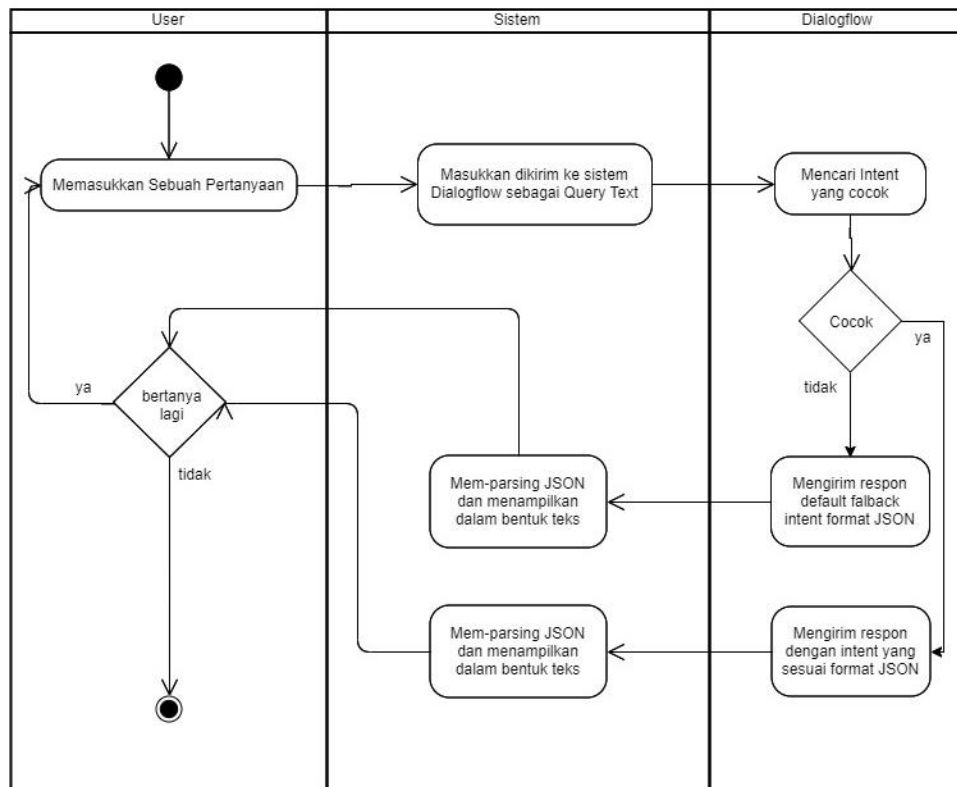
## a. Activity Diagram Login



Gambar 3. 7 Activity Diagram Login

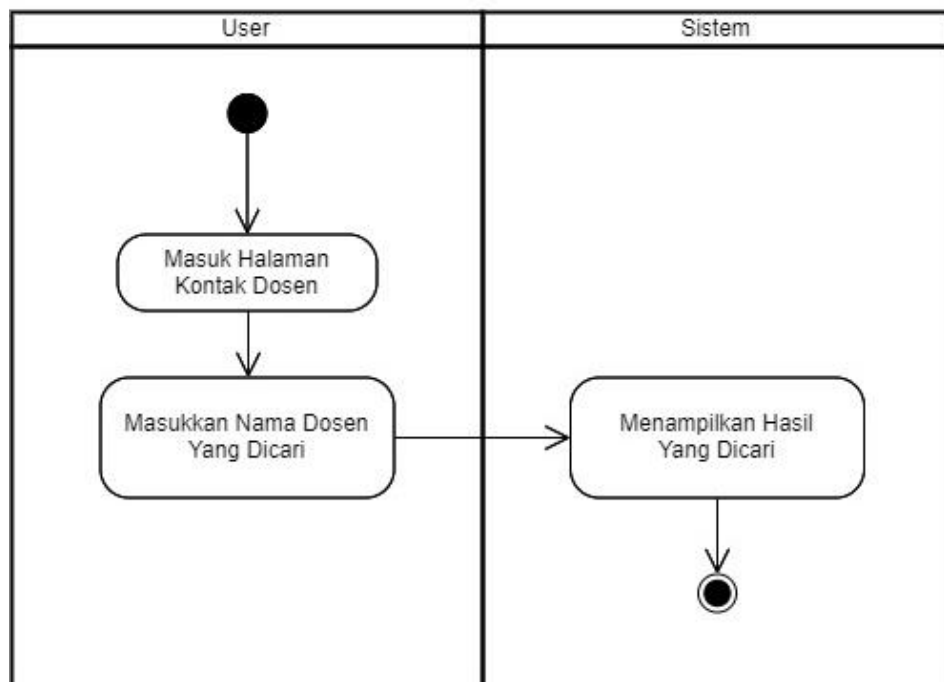
Activity diagram *login* ini menunjukkan proses admin *login* ke *dashboard* admin dengan memasukkan *username* dan *password*, jika sukses maka akan ke halaman *dashboard* dan jika tidak maka akan diminta memasukkan *username* dan *password* kembali

*b. Activity Bot*

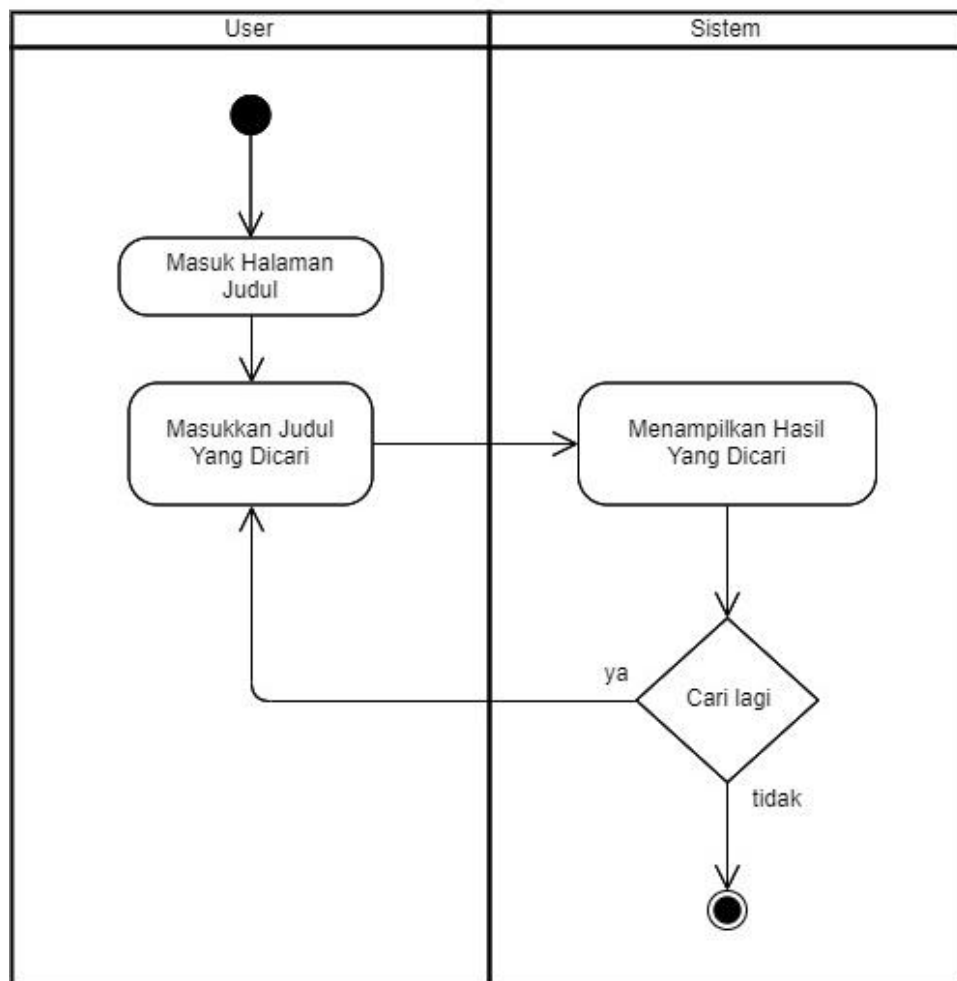


Gambar 3. 8 Activity Diagram Bot

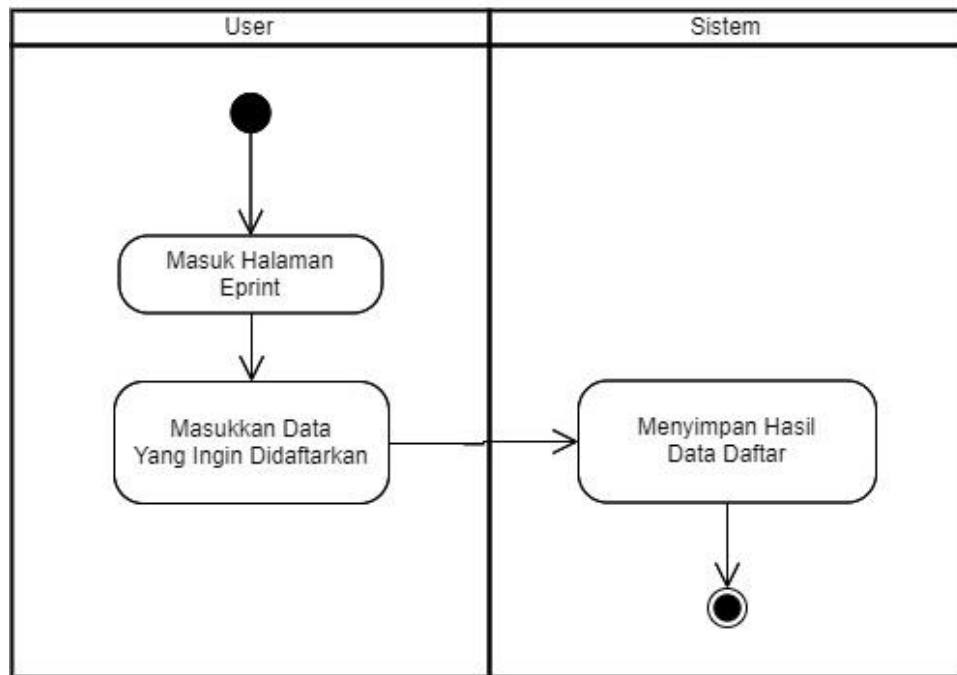
Activity diagram *bot* ini menunjukkan proses *user* bertanya kepada *bot* dengan memasukkan kata yang ingin dicari dan *bot* menampilkan dengan respon sesuai format JSON jika ada maka akan ditampilkan ke *user* dan jika tidak maka *bot* menampilkan *fallback default*.

c. *Activity* Kontak DosenGambar 3. 9 *Activity* Diagram Kontak Dosen

*Activity* diagram kontak dosen diatas menunjukkan proses untuk *user* mencari kontak dosen, *user* hanya perlu memasukkan nama dosen yang ingin dicari.

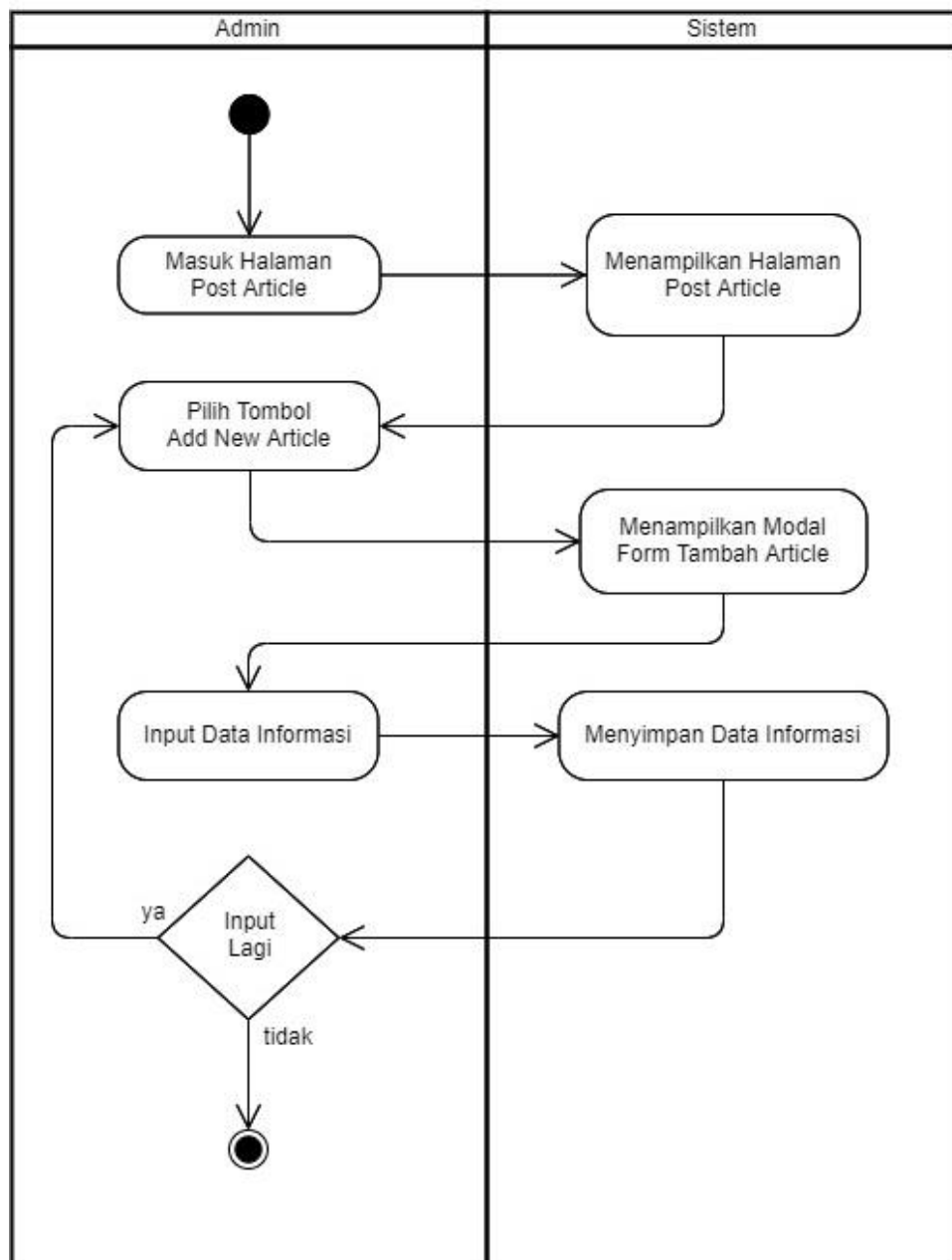
d. *Activity* JudulGambar 3. 10 *Activity* Diagram Judul

*Activity* diagram judul ini menunjukkan proses sebagai *user* ingin mencari judul skripsi *user* hanya perlu memasukkan judul yang ingin dicari dan jika tidak ada maka *user* diminta memasukkan kembali judul yang lain.

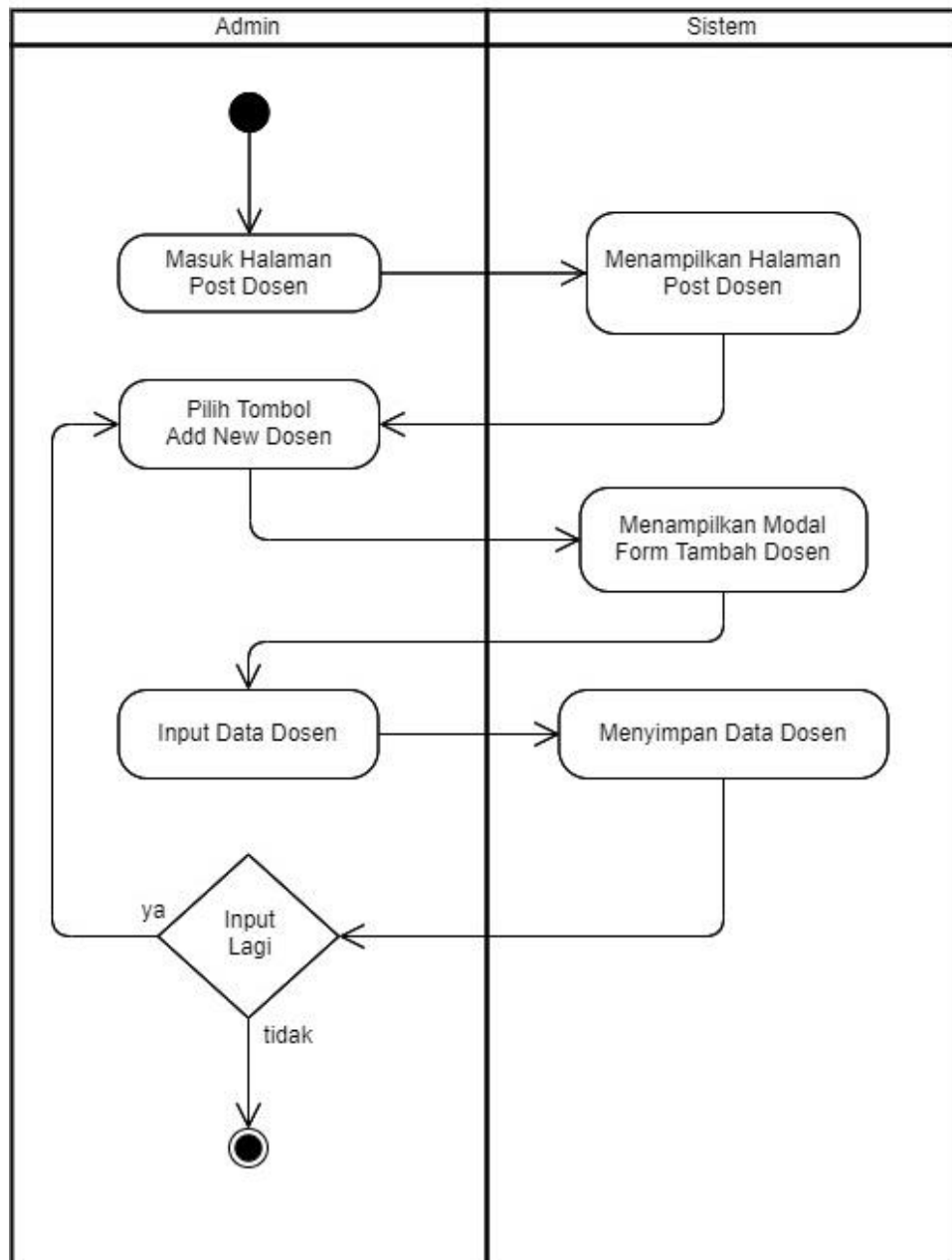
e. *Activity Eprints*Gambar 3. 11 *Activity Diagram Eprints*

*Activity* diagram eprints ini menunjukkan proses sebagai *user* yang ingin mendaftarkan untuk mendapatkan akun untuk ke eprints diharuskan memasukkan data diri yang ingin didaftarkan.



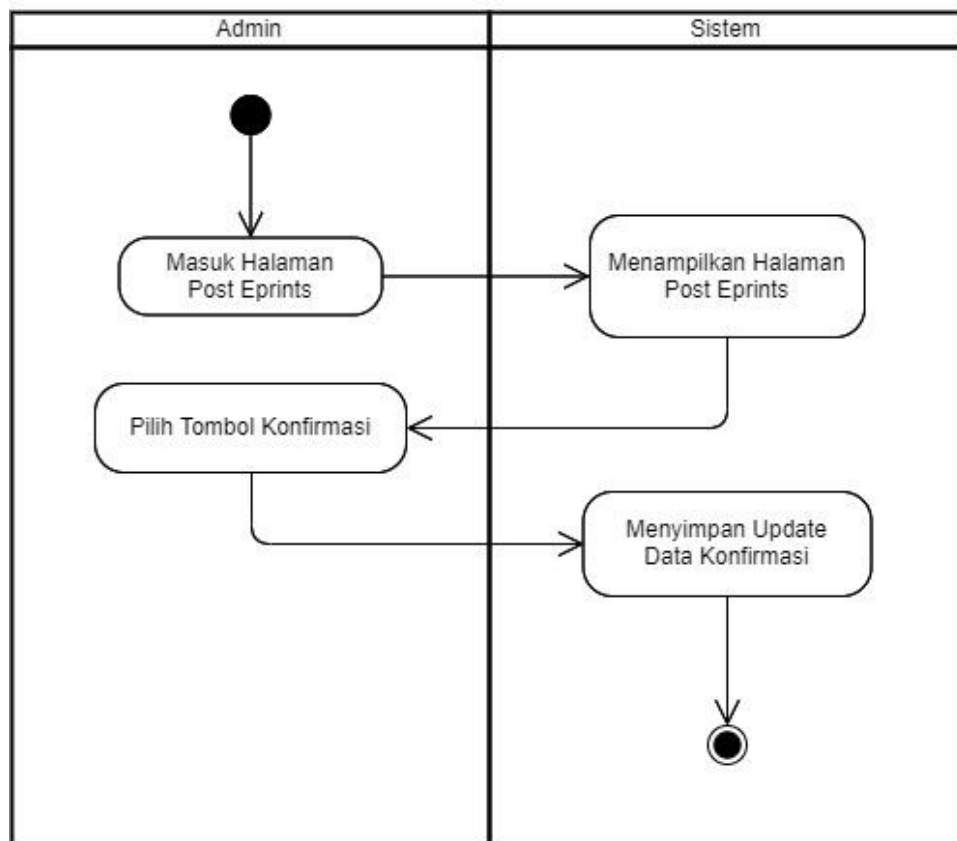
f. *Activity Post Informasi*Gambar 3. 12 *Activity Diagram Post Informasi*

*Activity diagram post informasi* ini menunjukkan proses untuk admin menambahkan informasi dengan menuju ke halaman *post informasi* di *web admin* lalu dengan menekan tombol tambah *article* maka akan memunculkan modal *form* untuk memasukkan informasi dan menyimpannya.

g. *Activity Post Dosen*Gambar 3. 13 *Activity Diagram Post Dosen*

*Activity* diagram post dosen untuk menunjukkan proses admin menambahkan dosen dengan menuju ke halaman post dosen di *web* admin lalu dengan menekan tombol tambah dosen maka akan memunculkan modal *form* untuk memasukkan info dosen dan menyimpannya.

#### h. Activity Update Data Eprints

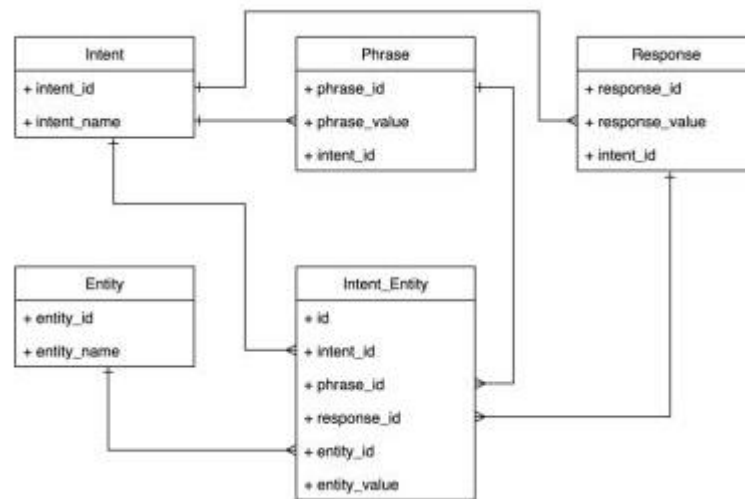


Gambar 3. 14 Activity Diagram Update Data Eprints

Activity diagram *update data eprints* untuk menunjukkan proses admin mengharuskan mengkonfirmasi data mahasiswa yang mendaftarkan eprints, dengan menuju halaman *post eprints* dan pilih mahasiswa yang ingin dikonfirmasi datanya dengan klik tombol *confirm*.

### 3.3.3 Class Diagram

Penulis menggambarkan *class diagram* bertujuan untuk menampilkan kelas-kelas dan paket didalam *system* aplikasi ini. Class diagram memberikan gambaran *system* aplikasi dan relasi antar mereka. Biasanya, dibuat beberapa *class diagram* untuk *system* tunggal. Beberapa diagram akan menampilkan subset dari kelas-kelas dan relasinya.

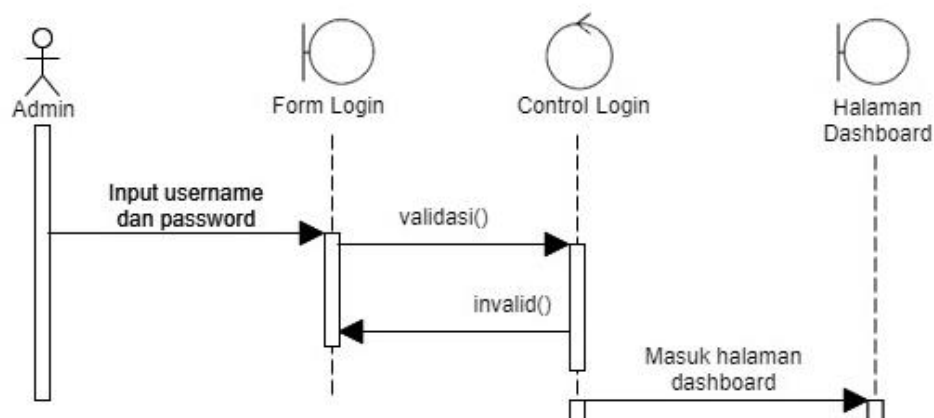


Gambar 3. 15 Class Diagram Bot

### 3.3.4 Sequence Diagram

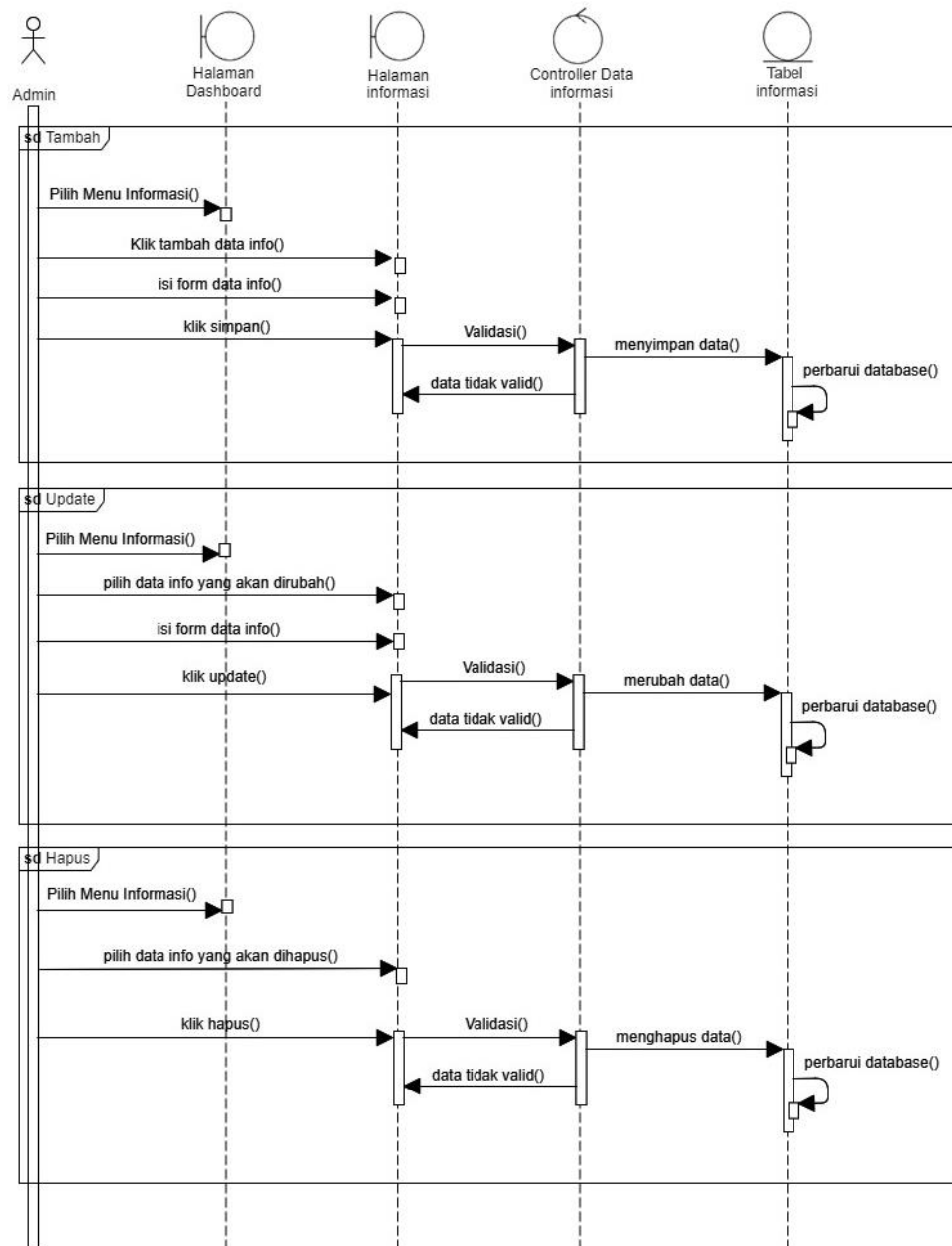
Sequence diagram menjelaskan interaksi objek yang disusun dalam suatu urutan waktu. Diagram ini secara khusus berasosiasi dengan *usecase*. *Sequence* diagram juga menggambarkan kelakuan objek pada *usecase* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan yang diterima oleh objek oleh karena itu untuk menggambarkan diagram *sequence* harus diketahui objek-objek yang terlibat dalam sebuah *usecase* beserta metode-metode yang dimiliki *class* yang diinstansiasi menjadi objek itu.

#### a. Sequence Diagram Login Admin



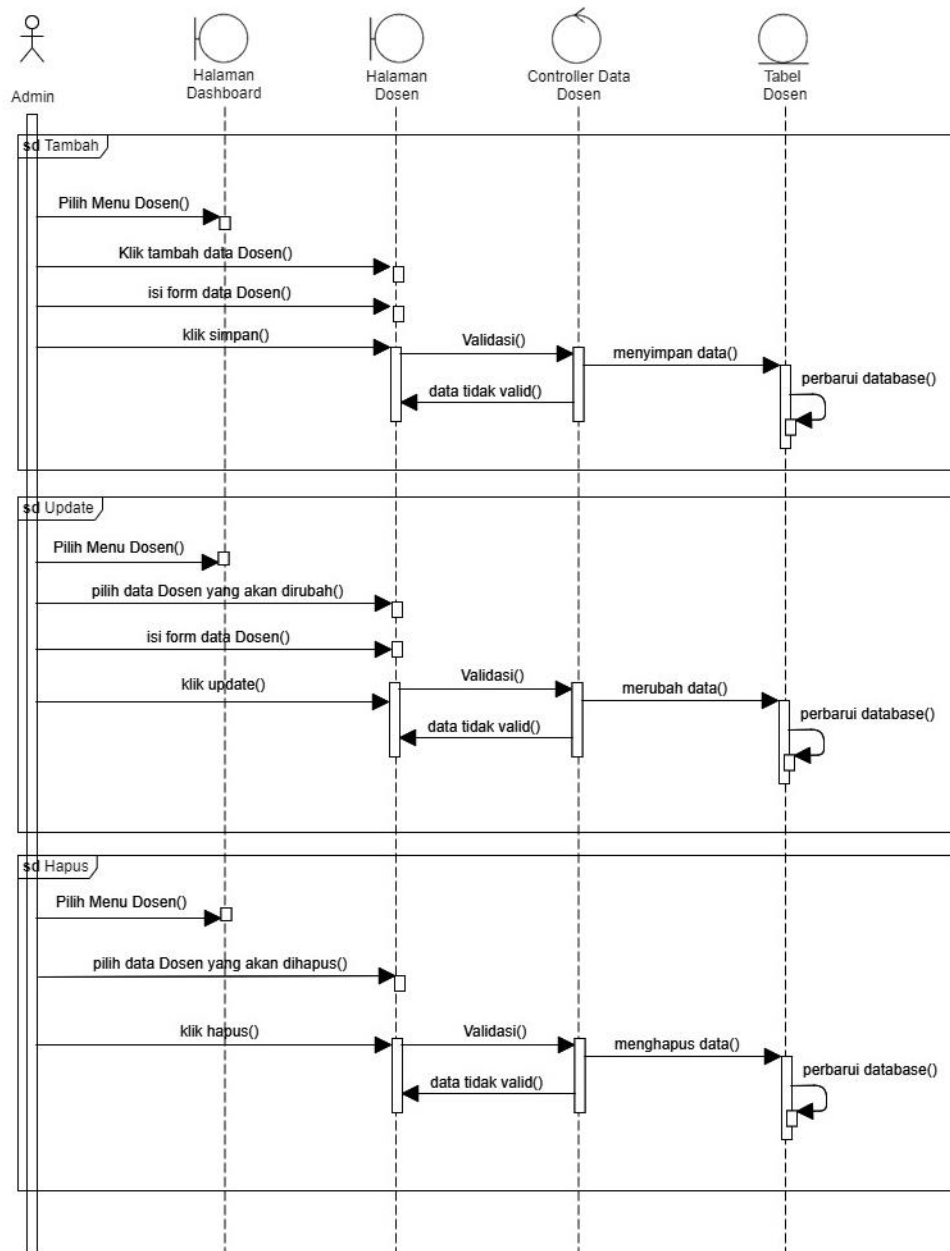
Gambar 3. 16 Sequence Diagram Login Admin

b. *Sequence Diagram Post Informasi*



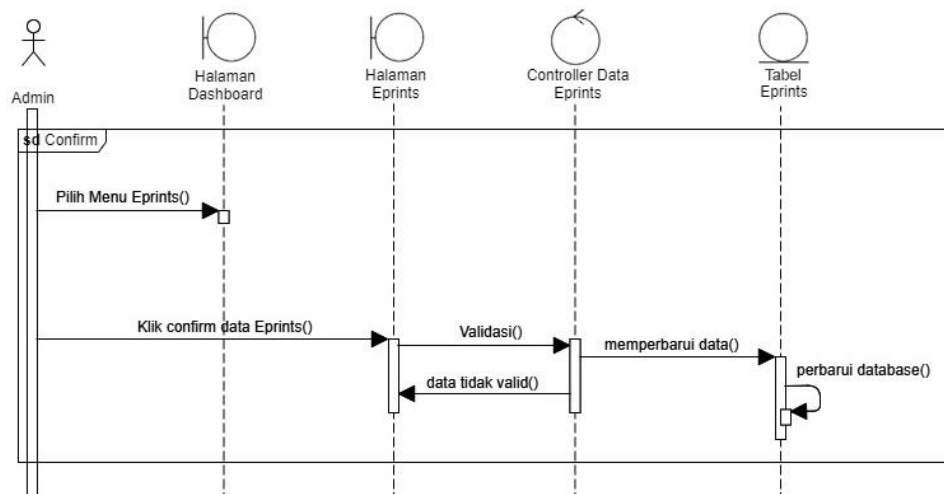
Gambar 3. 17 *Sequence Diagram Post Informasi*

c. *Sequence Diagram Post Dosen*



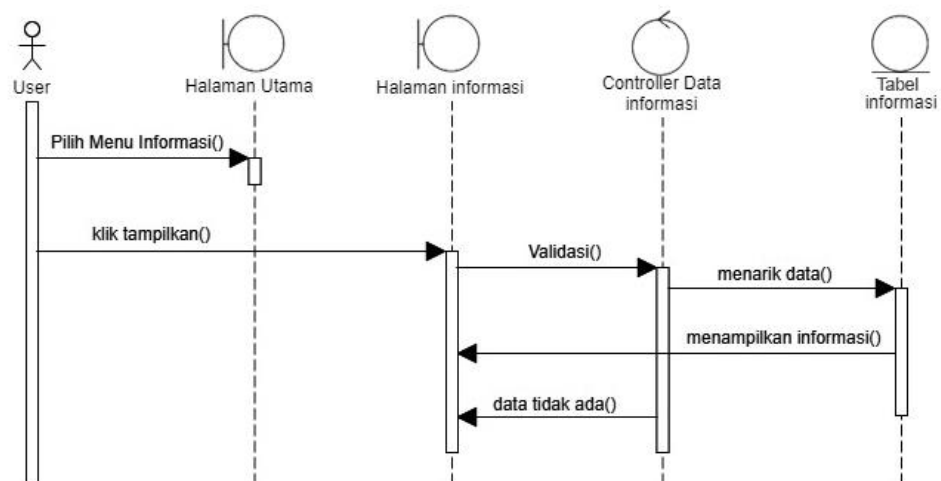
Gambar 3. 18 *Sequence Diagram Post Dosen*

d. *Sequence Diagram Post Eprints*



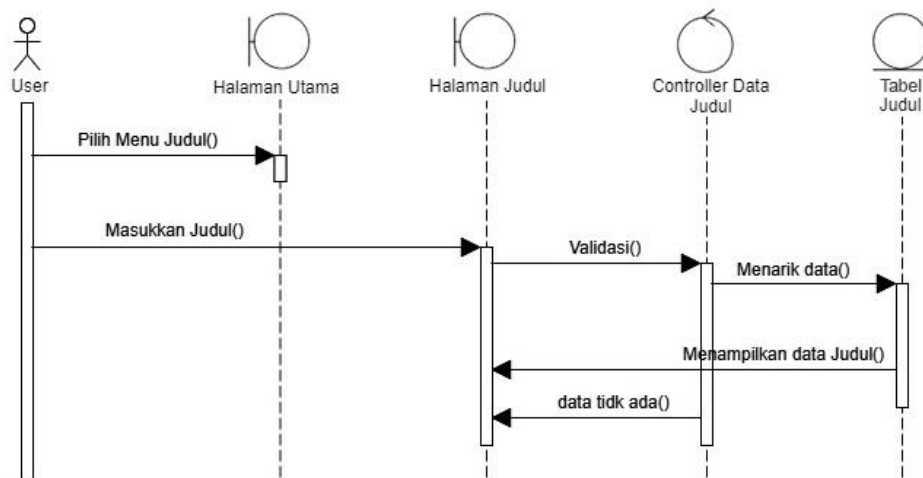
Gambar 3. 19 *Sequence Diagram Post Eprints*

e. *Sequence Diagram Informasi*



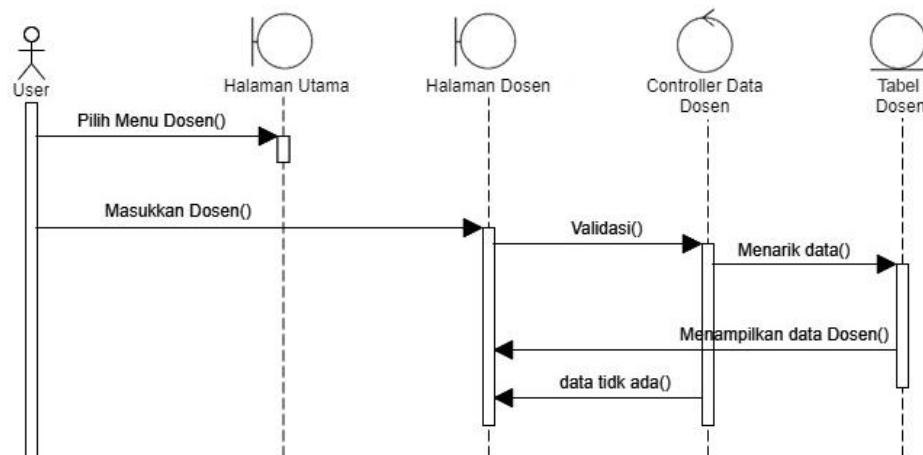
Gambar 3. 20 *Sequence Diagram Informasi*

f. *Sequence Diagram Judul*



Gambar 3. 21 *Sequence Diagram Judul*

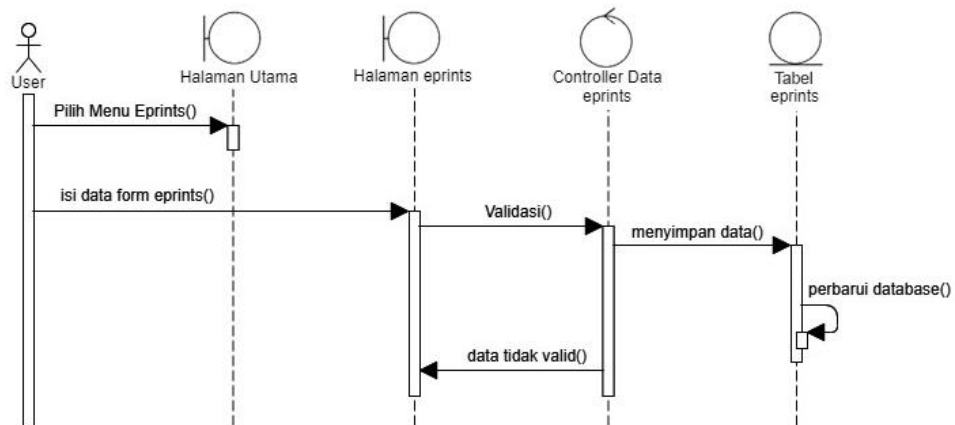
g. *Sequence Diagram Dosen*



Gambar 3. 22 *Sequence Diagram Dosen*

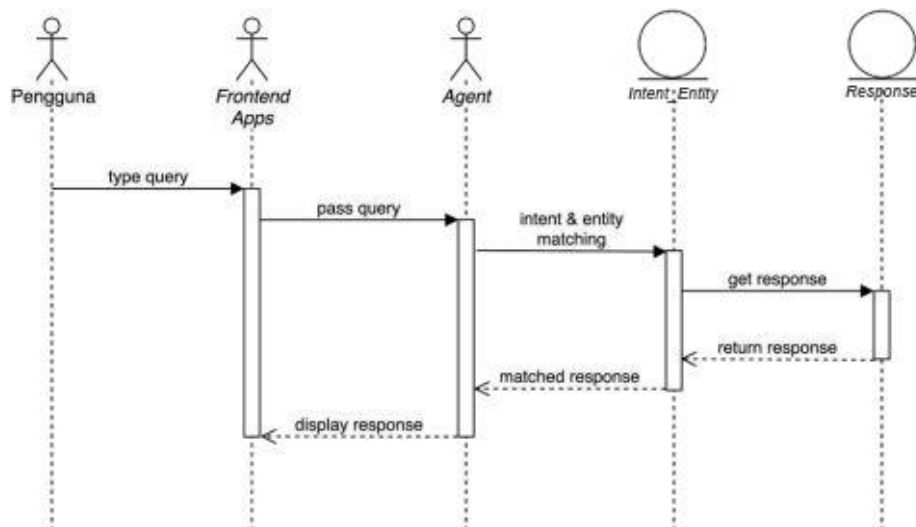


### h. *Sequence Diagram Eprints*



Gambar 3. 23 *Sequence Diagram Eprints*

### i. *Sequence Diagram Chatbot*

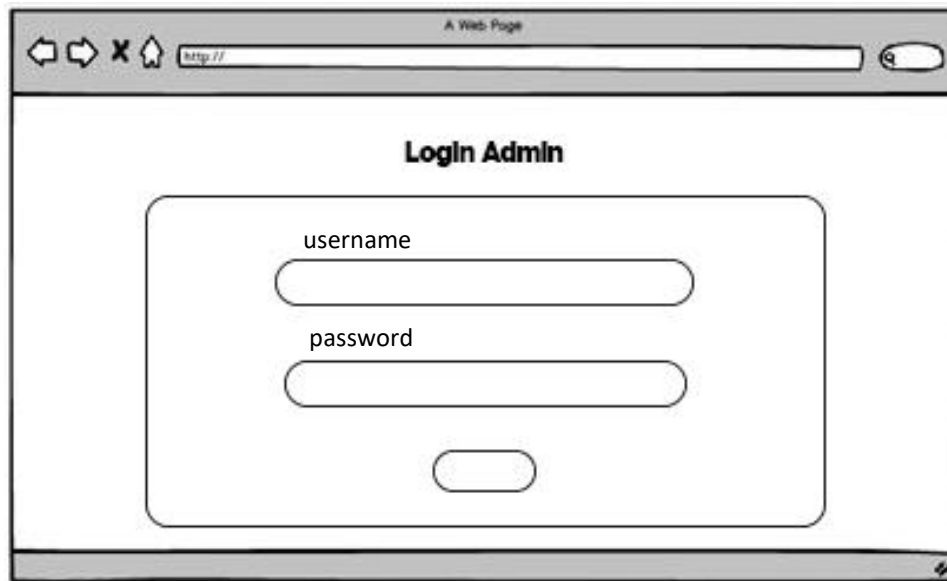


Gambar 3. 24 *Sequence Diagram Chatbot*

## 3.4 Perancangan Antar Muka (*User Interface*)

perancangan antarmuka merupakan suatu deskripsi layout antar muka system yang akan dibuat. Dengan adanya perancangan antar muka diharapkan baik pengembang aplikasi maupun pengguna dapat menyatukan pandangan mengenai layout tampilan aplikasi yang dibuat.

### 3.4.1 Halaman *Login*



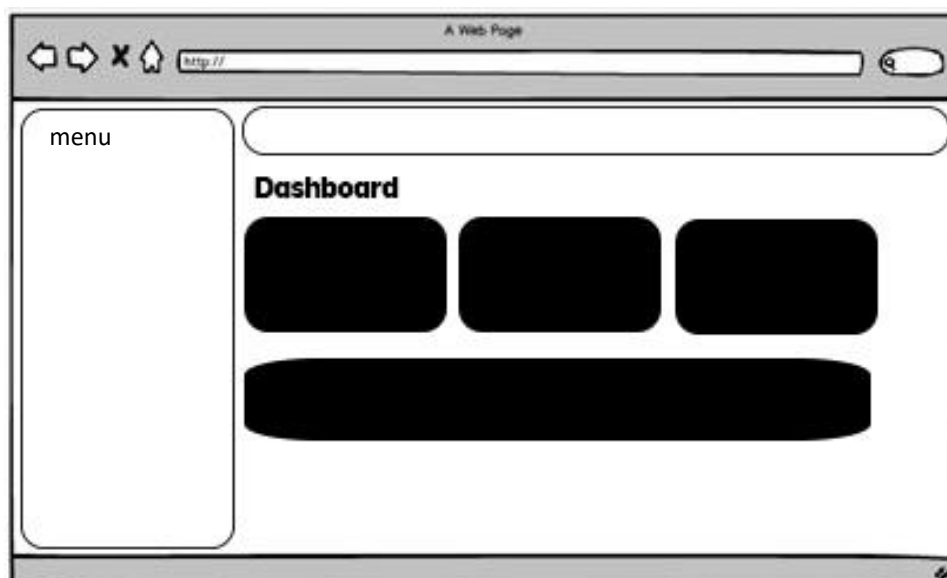
The image shows a wireframe of a web browser window titled "A Web Page". The address bar contains "http://". The main content area is titled "Login Admin". Inside this area, there is a rounded rectangle containing three input fields: a text input labeled "username", a text input labeled "password", and a button below them.

Gambar 3. 25 Rancangan Tampilan Halaman *Login*

Gambar 3.25 menunjukkan rancangan tampilan halaman *login* untuk admin yang mempunyai komponen:

- a. *input field* yang terdiri dari:
  - *text input username* untuk memasukkan *username* admin
  - *text input password* untuk memasukkan *password* admin

### 3.4.2 Halaman *Dashboard*



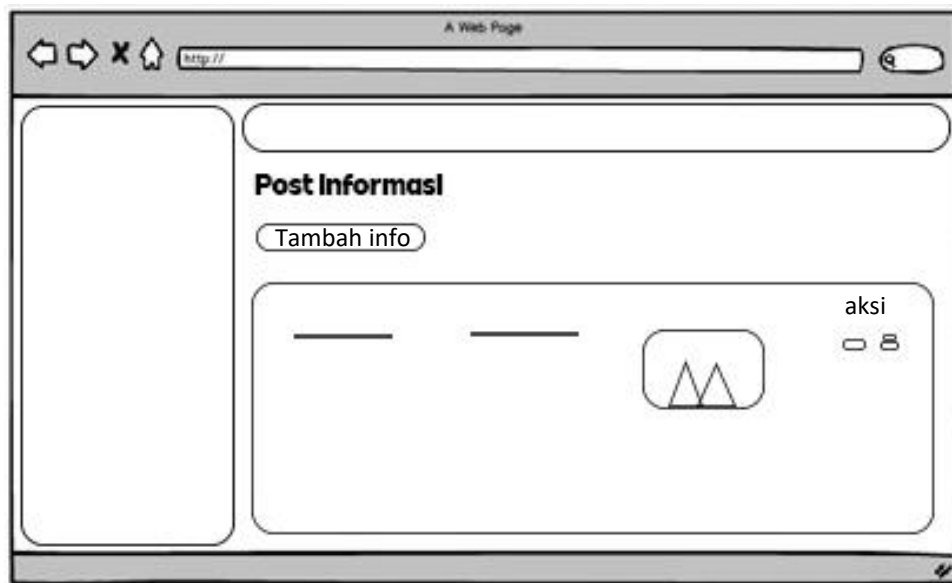
The image shows a wireframe of a web browser window titled "A Web Page". The address bar contains "http://". The main content area is titled "Dashboard". On the left side, there is a sidebar with a "menu" label. The main content area contains a search bar at the top, followed by three black rectangular blocks arranged horizontally, and a single long black rectangular block below them.

Gambar 3. 26 Rancangan Tampilan Halaman *Dashboard*

Gambar 3.26 menunjukkan rancangan tampilan halaman *dashboard* admin yang mempunyai komponen:

- Sidebar* disebelah kiri adalah menu untuk mengelola aktifitas admin
- Navbar* dibagian atas hanya menampilkan nama *username* admin
- Bagian *body dashboard* terdapat beberapa kotak menampilkan berapa banyak data informasi, judul, dosen, dan eprints.

### 3.4.3 Halaman *Post Info*



Gambar 3. 27 Rancangan Tampilan Halaman *Post Info*

Gambar 3.27 menunjukkan rancangan tampilan halaman post info admin yang mempunyai komponen:

- Tombol tambah info untuk memunculkan modal *form* tambah informasi
- Table data informasi terdapat judul informasi, deskripsi dan gambar
- Terdapat tombol aksi *edit* dan *delete* untuk setiap data

### 3.4.4 Halaman *Post Dosen*

The wireframe shows a web browser window titled 'A Web Page'. The address bar contains 'http://'. The page layout includes a sidebar on the left and a main content area. The main content area has a header section titled 'Post Dosen' which contains a search bar and a 'Tambah dosen' button. Below this is a table with three columns: 'Nama', 'Kontak', and 'Aksi'.

Nama	Kontak	Aksi

Gambar 3. 28 Rancangan Tampilan Halaman *Post Dosen*

Gambar 3.28 menunjukkan rancangan tampilan halaman *post* dosen yang mempunyai komponen:

- Tombol tambah dosen untuk memunculkan modal form tambah dosen
- Table data dosen terdapat nama, nidn dosen dan kontak dosen
- Terdapat tombol aksi *edit* dan *delete* untuk setiap data

### 3.4.5 Halaman *Post Eprints*

The wireframe shows a web browser window titled 'A Web Page'. The address bar contains 'http://'. The page layout includes a sidebar on the left and a main content area. The main content area has a header section titled 'Post Eprints' which contains a search bar and a table with three columns: 'Nama', 'Data', and 'Konfirmasi'.

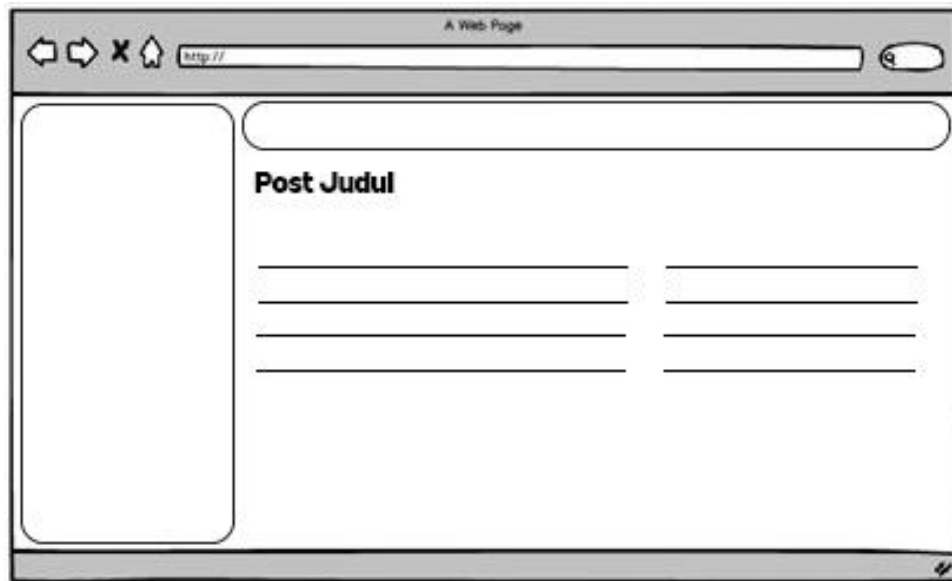
Nama	Data	Konfirmasi

Gambar 3. 29 Rancangan Tampilan Halaman *Post Eprints*

Gambar 3.29 menunjukkan rancangan tampilan halaman *post* eprints yang mempunyai komponen:

- a. Tabel data eprints yang terdapat nim, nama, jurusan, *email*, dan nomor hp
- b. Terdapat tombol konfirmasi untuk mengkonfirmasi data pendaftar eprints

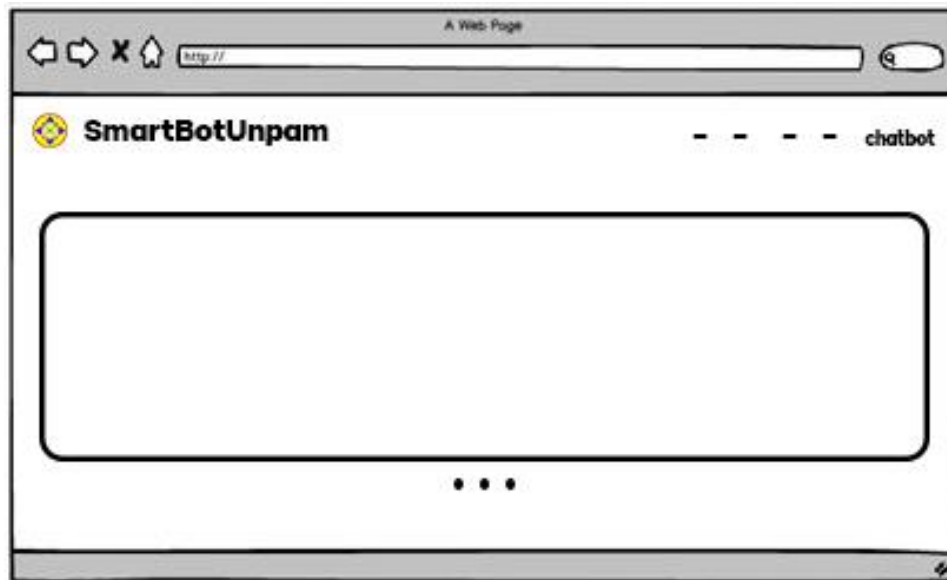
#### 3.4.6 Halaman *Post* Judul



Gambar 3. 30 Rancangan Tampilan Halaman *Post* Judul

Gambar 3.30 menunjukkan rancangan tampilan halaman *post* dosen yang hanya menampilkan judul judul skripsi dan terdapat penulis dan tahun terbit.

### 3.4.7 Halaman Utama

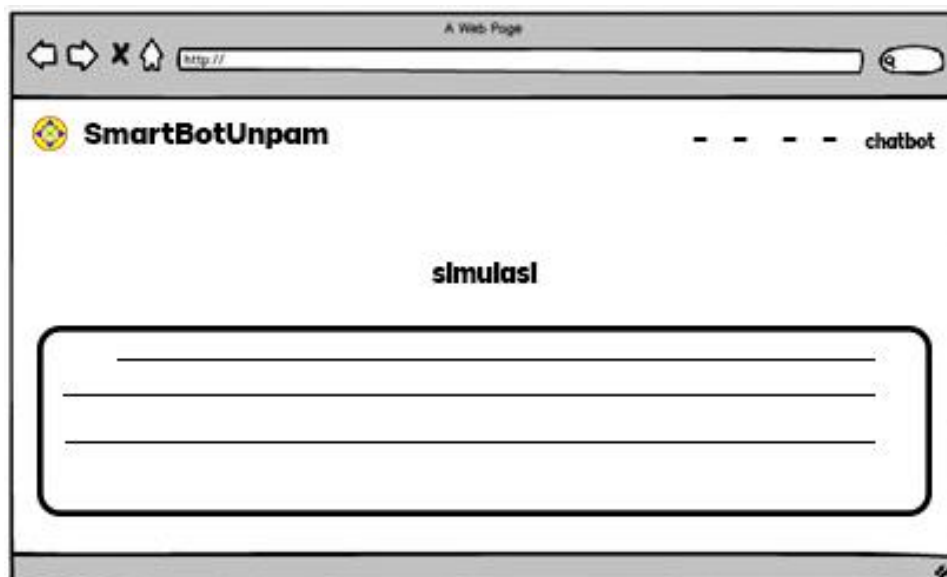


Gambar 3. 31 Rancangan Tampilan Halaman Utama

Gambar 3.31 menunjukkan rancangan tampilan halaman utama yang mempunyai komponen:

- a. Terdapat logo UNPAM dan nama *web* yaitu *SmartBotUnpam*
- b. *Navbar* untuk navigasi menu
- c. *Carousel* untuk menampilkan banner info-info skripsi di UNPAM

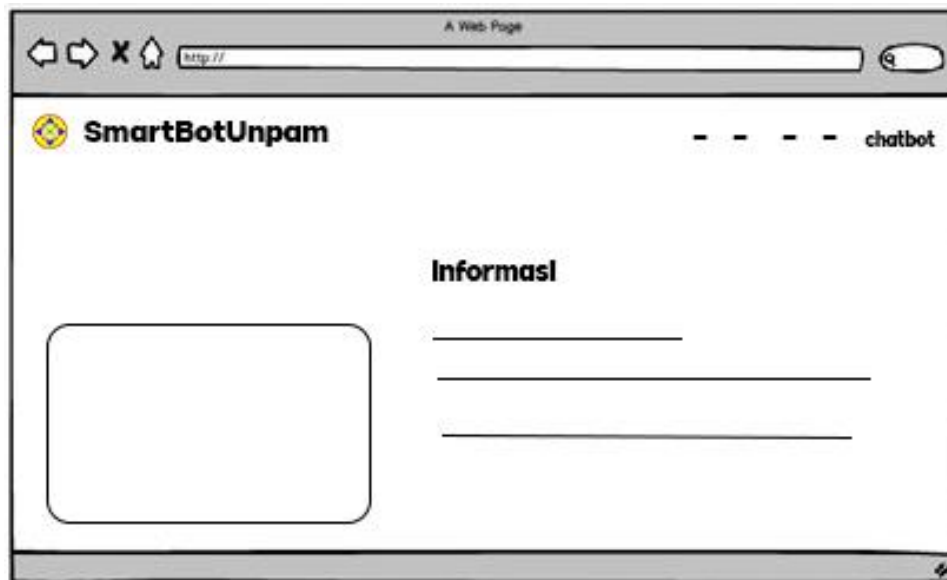
### 3.4.8 Halaman Simulasi



Gambar 3. 32 Rancangan Tampilan Halaman Simulasi

Gambar 3.32 menunjukkan rancangan tampilan halaman simulasi yang menampilkan proses atau tahapan tahapan prosedur menyusun skripsi di UNPAM

#### 3.4.9 Halaman Informasi

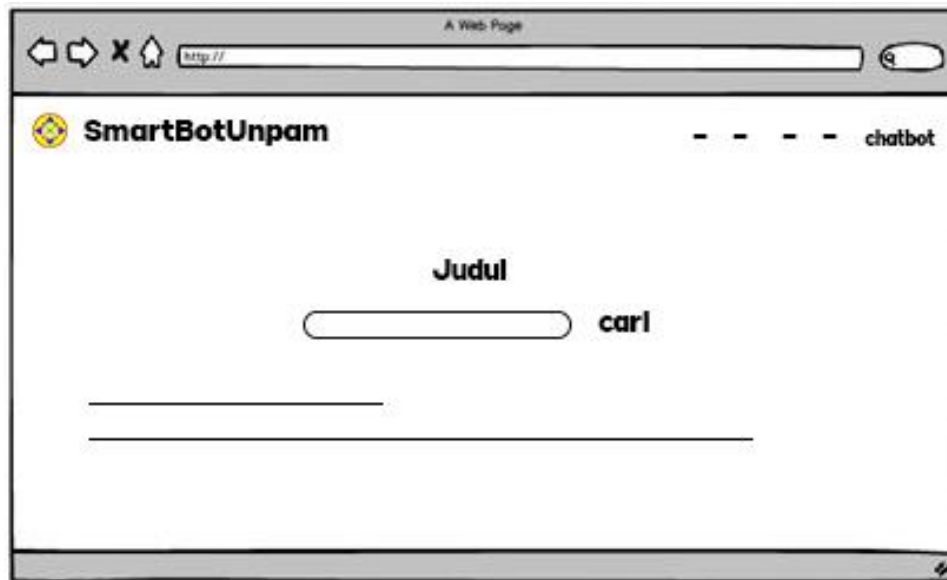


Gambar 3. 33 Rancangan Tampilan Halaman Informasi

Gambar 3.33 menunjukkan rancangan tampilan halaman informasi yang mempunyai komponen:

- Kotak disebelah kiri menunjukkan gambar dari informasi
- Tulisan disebelah kanan menunjukkan deskripsi dari informasi tersebut

### 3.4.10 Halaman Judul

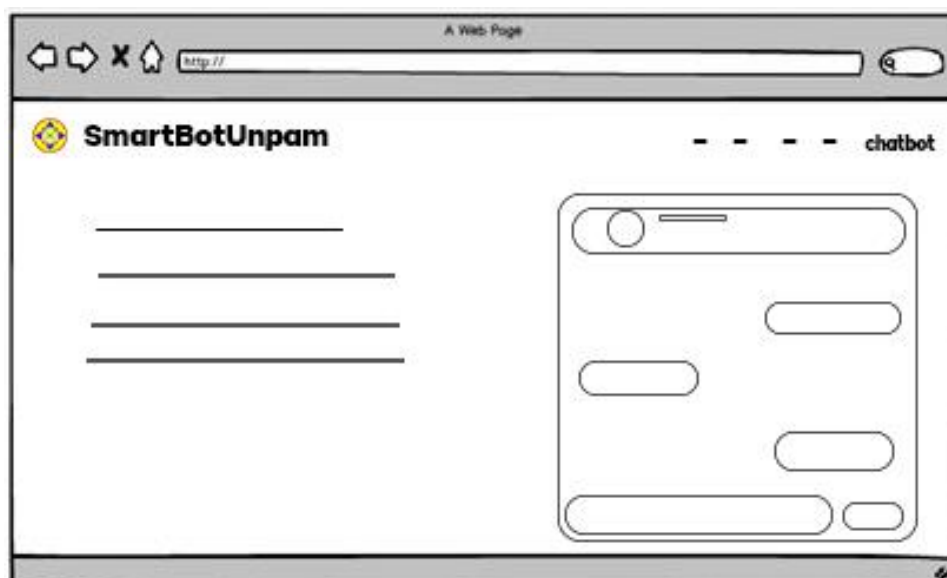


Gambar 3. 34 Rancangan Tampilan Halaman Judul

Gambar 3.34 menunjukkan rancangan tampilan halaman judul yang mempunyai komponen:

- Form* pencarian untuk memasukkan *keyword* judul yang ingin dicari oleh *user*
- Bagian bawah terdapat judul – judul yang muncul

### 3.4.11 Halaman *Chatbot*



Gambar 3. 35 Rancangan Tampilan Halaman *Chatbot*



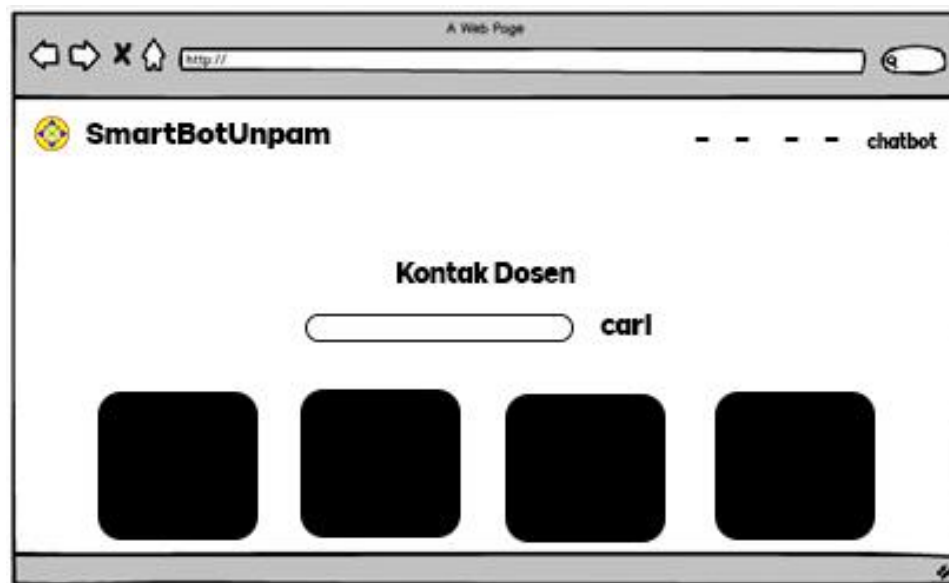
Gambar 3.35 menunjukkan rancangan tampilan halaman *post* dosen yang hanya terdapat *frame box chatbot* dimana tempat *user* bertanya informasi skripsi.

#### 3.4.12 Halaman *Form Eprints*

Gambar 3. 36 Rancangan Tampilan Halaman *Form Eprints*

Gambar 3.28 menunjukkan rancangan tampilan halaman *post* dosen yang mempunyai *form* untuk mengisi data diri mahasiswa mendaftarkan eprints.

### 3.4.13 Halaman Kontak Dosen



Gambar 3. 37 Rancangan Tampilan Halaman Kontak Dosen

Gambar 3.37 menunjukkan rancangan tampilan halaman kontak dosen yang mempunyai komponen:

- Form* pencarian untuk memasukkan nama dosen yang ingin dicari
- Kotak data kontak dosen yang terdapat nama, nidn dan kontak dosen.

## **BAB IV**

### **IMPLEMENTASI DAN PENGUJIAN**

#### **4.1 Spesifikasi Sistem**

Implementasi system merupakan dokumen yang berfungsi menggambarkan fungsi dan kinerja system berbasis computer yang akan dikembangkan, membatasi elemen-elemen yang telah dialokasikan, serta memberikan indikasi mengenai perangkat lunak dan konteks system keseluruhan dan informasi data dan control yang dimasukkan dan dikeluarkan oleh system yang telah digambarkan dalam diagram aliran arsitektur. Berikut salah satu format dokumen spesifikasi sistem yang biasa digunakan.

##### **4.1.1 Spesifikasi Perangkat Keras**

*Hardware* atau disebut dengan perangkat keras yaitu salah satu komponen dari sebuah computer yang sifatnya bisa dilihat dan diraba secara langsung atau yang berbentuk nyata, yang berfungsi untuk mendukung proses komputerisasi. *Hardware* dapat bekerja berdasarkan perintah yang telah ditentukan ada padanya, atau yang juga disebut dengan istilah *instruction set*. Dengan adanya perintah yang dapat dimengerti oleh *hardware* tersebut, maka *hardware* tersebut dapat melakukan berbagai kegiatan yang telah ditentukan oleh pemberi perintah.

Adapun perangkat keras yang digunakan untuk mendukung pembuatan program ini adalah sebagai berikut:

- a. Processor: Intel Core2duo
- b. Monitor: 14-inch HD
- c. Memori: 2GB
- d. Hardisk: 500 GB

##### **4.1.2 Implementasi Perangkat Lunak**

Sekumpulan data elektronik yang disimpan oleh *computer* itu dapat berupa program atau instruksi yang akan menjalankan suatu perintah. Perangkat lunak

disebut juga sebagai penerjemah perintah-perintah yang dijalankan pengguna *computer* untuk diteruskan atau diproses oleh perangkat keras. Melalui *software* atau perangkat lunak inilah suatu computer dapat menjalankan perintah.

Adapun perangkat lunak (*software*) yang dibutuhkan untuk menjalankan aplikasi system informasi skripsi adalah sebagai berikut:

- a. Sistem Operasi Windows 10
- b. Sublime Text 3
- c. XAMPP Control Panel versi 3.2.2

## 4.2 Implementasi Antarmuka (*Interface*)

Implementasi antar muka dilakukan dengan setiap lempiran program yang dibangun dan pengkodeannya dalam bentuk *file* program, berikut beberapa tampilan program yang terdapat didalam aplikasi media pemberlajaran pengenalan *hardware computer*. Berikut ini adalah implementasi antarmuka dari Aplikasi Sistem Informasi Skripsi di Unversitas Pamulang:

### 4.2.1 Halaman Utama

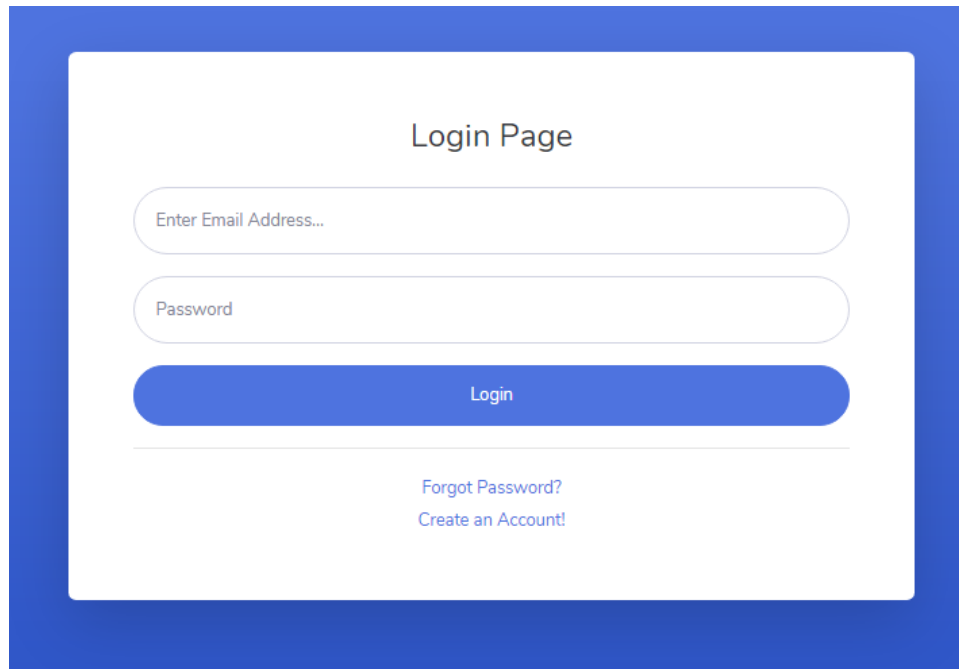
Tampilan halaman utama ini berisi tentang informasi singkat tentang *web* Sistem Informasi Skripsi Universitas Pamulang (*smartbot unpam*).



Gambar 4. 1 Tampilan Halaman Utama

### 4.2.2 Halaman *Login*

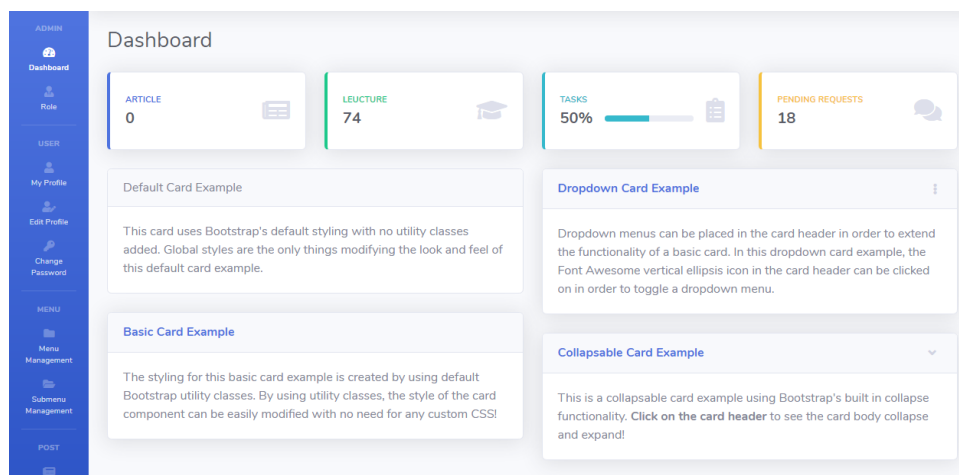
Tampilan halaman *login* diperuntukan bagi admin *website* untuk mengatur data didalam *web* sistem informasi skripsi ini.



Gambar 4. 2 Tampilan Halaman *Login* Admin

### 4.2.3 Halaman *Dashboard* Admin

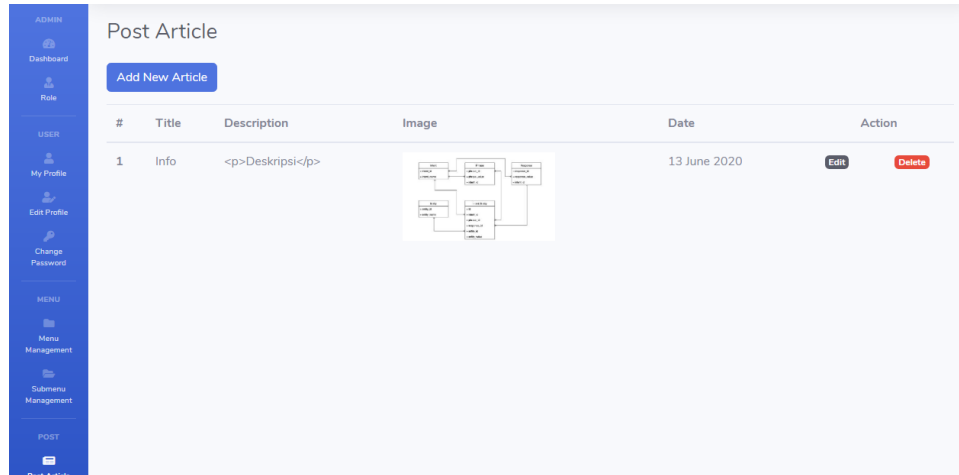
Sebuah halaman untuk menampilkan jumlah data yang terdapat di dalam *web* sistem informasi skripsi ini.



Gambar 4. 3 Tampilan Halaman *Dashboard*

#### 4.2.4 Halaman *Post Info*

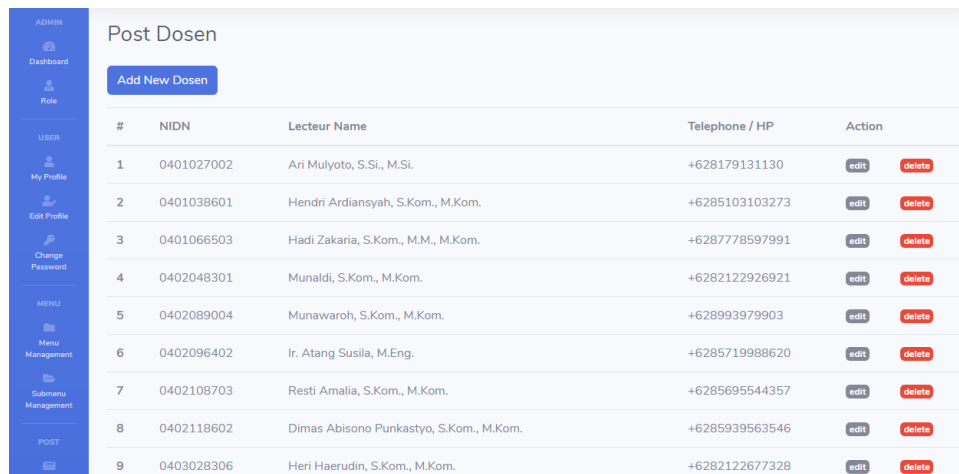
Halaman *post info* atau artikel adalah sebuah halaman admin untuk mengelola sebuah postingan informasi untuk pengguna *web*.



Gambar 4. 4 Tampilan Halaman *Post Info*

#### 4.2.5 Halaman *Post Dosen*

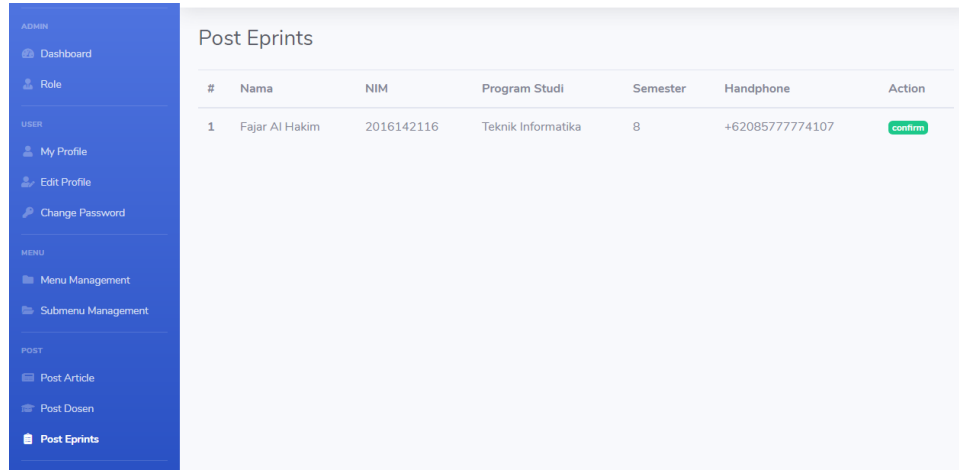
Halaman *post dosen* adalah halaman untuk mengelola data-data dosen teknik informatika di universitas pamulang.



Gambar 4. 5 Tampilan Halaman *Post Dosen*

#### 4.2.6 Halaman *Post Eprints*

Halaman *post eprints* yaitu halaman untuk mengelola data pendaftar eprints untuk dikonfirmasi oleh admin untuk didaftarkan ke bagian eprints universitas pamulang.

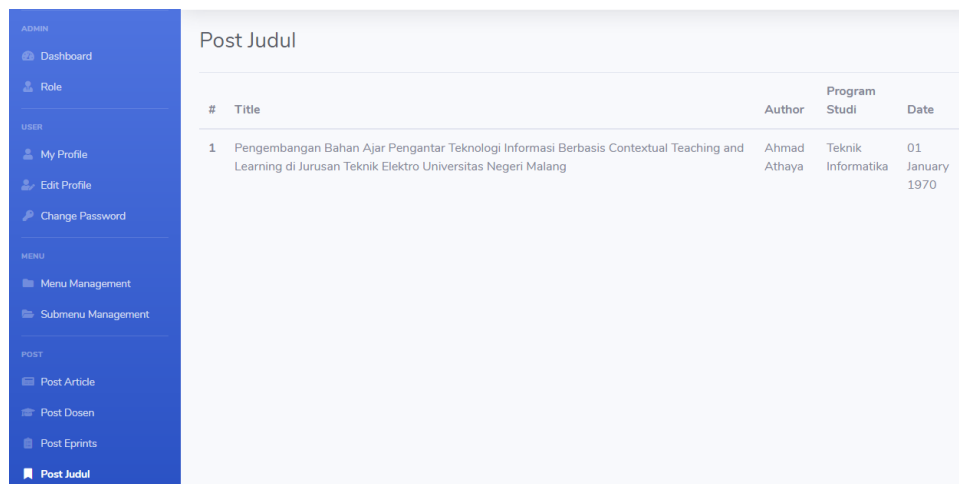


#	Nama	NIM	Program Studi	Semester	Handphone	Action
1	Fajar Al Hakim	2016142116	Teknik Informatika	8	+6208577774107	<a href="#">confirm</a>

Gambar 4. 6 Tampilan Halaman *Post Eprints*

#### 4.2.7 Halaman *Post Judul*

Halaman *post judul* yaitu halaman untuk mengelola data judul – judul skripsi yang ada di Universitas Pamulang Jurusan Teknik Informatika



#	Title	Author	Program Studi	Date
1	Pengembangan Bahan Ajar Pengantar Teknologi Informasi Berbasis Contextual Teaching and Learning di Jurusan Teknik Elektro Universitas Negeri Malang	Ahmad Athaya	Teknik Informatika	01 January 1970

Gambar 4. 7 Tampilan Halaman *Post Judul*

### 4.2.8 Halaman Skripsi

Halaman ini adalah halaman dimana terdapat simulasi atau tahapan pedoman penyusunan skripsi.



Gambar 4. 8 Tampilan Halaman Skripsi

### 4.2.9 Halaman Informasi

Halaman informasi halaman dimana terdapat kumpulan informasi yang dibutuhkan pengguna.



Gambar 4. 9 Tampilan Halaman Informasi



#### 4.2.10 Halaman Judul

Halaman judul yaitu halaman untuk pencarian judul-judul skripsi yang sudah pernah ada di Universitas Pamulang

SmartBot UNPAM

🏠 Beranda Simulasi Informasi Judul Dosen Eprints ChatBot

### Judul Skripsi

Info Judul Skripsi jurusan Teknik Informatika Universitas Pamulang.

Cari Judul Skripsi... Submit

#	Judul	Penulis	Program Studi	Tanggal Terbit
1	Pengembangan Bahan Ajar Pengantar Teknologi Informasi Berbasis Contextual Teaching and Learning di Jurusan Teknik Elektro Universitas Negeri Malang	Ahmad Athaya	Teknik Informatika	01 January 1970

UNPAM My UNPAM E-Prints Tentang Bantuan FAQ

Facebook Instagram

Gambar 4. 10 Tampilan Halaman Judul

#### 4.2.11 Halaman Dosen

Halaman dosen yaitu halaman yang terdapat nama-nama dosen beserta kontak jurusan teknik informatika yang ada di universitas pamulang.

SmartBot UNPAM

🏠 Beranda Simulasi Informasi Judul Dosen Eprints ChatBot

### Info Dosen

Info Kontak Dosen jurusan Teknik Informatika Universitas Pamulang.

Cari Nama Dosen... Submit

Q Hasil : 74 Dosen

0401027002  
Ari Mulyoto, S.Si., M.Si.  
📞 +628179131130

0401038601  
Hendri Ardiansyah, S.Kom., M.Kom.  
📞 +6285103103273

0401044503  
Hadi Zakaria, S.Kom., M.Mi., M.Kom.  
📞 +6287778597991

0402048301  
Munaldi, S.Kom., M.Kom.  
📞 +6282122926921

Gambar 4. 11 Tampilan Halaman Dosen

#### 4.2.12 Halaman Eprints

Halaman eprints ini adalah halaman *form* pendaftaran eprints untuk mendapatkan akses *login* ke *website* eprints universitas pamulang.

SmartBot UNPAM

🏠 Beranda Simulasi Informasi Judul Dosen Eprints ChatBot

### E-Prints

Form Pendaftaran Perpustakaan Online Universitas Pamulang.

Hey kamu yang sedang mau mencari referensi jurnal untuk penelitian skripsi kamu, Universitas Pamulang menyediakan perpustakaan online loh namanya Eprints. Bagi kamu yang mau mengakses harus daftar terlebih dahulu, gampang kok kamu cukup mengisi form dibawah dengan lengkap dan benar via email kamu, jika sudah tunggu proses pembuatan akun selama kurang lebih 7x24jam.

NIM

Masukkan NIM Aktif

Nama

Masukkan Nama Mahasiswa

Program Studi

Masukkan Program Studi

Contoh : Teknik Informatika

Semester

Masukkan Semester

No Handphone

Masukkan Nomor Handphone Aktif

Gambar 4. 12 Tampilan Halaman Eprints

#### 4.2.13 Halaman Chatbot

Halaman ini hanya terdapat *frame chatbot* dimana pengguna dapat mengajukan pertanyaan ke *smartbot* tersebut.

SmartBot UNPAM

🏠 Beranda Simulasi Informasi Judul Dosen Eprints ChatBot

## Sistem Informasi Skripsi Jurusan Teknik Informatika Universitas Pamulang

Juni  
Juni Assistance Bot

POWERED BY Dialogflow

hi

Hai, Selamat datang, aku juni asisten bot yang akan membantu kamu menjawab pertanyaan-pertanyaan umum tentang skripsi jurusan Teknik Informatika Universitas Pamulang, semoga juni bisa membantu yaa :)

Ask something...

Gambar 4. 13 Tampilan Halaman Chatbot

### 4.3 Pengujian

Pengujian merupakan bagian yang terpenting dalam siklus pembangunan perangkat lunak. Pengujian dilakukan untuk menjamin kualitas dan mengetahui kelemahan dari perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang handal, yaitu mampu merepresentasikan kajian pokok dari spesifikasi, analisa, perancangan dan pengkodean dari perangkat lunak itu sendiri.

#### 4.3.1 Pengujian *Black Box*

Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Pengujian sistem informasi skripsi ini menggunakan data uji berupa data input dari administrator pada sistem yang telah dibuat.

##### a. *Black Box* Tampilan *Login Admin*

No.	Halaman	Test Case	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman <i>Login</i>	Isi <i>Username Admin</i> dan <i>Password</i> yang sudah didaftarkan sebelumnya	Akan Berhasil dan akan muncul Halaman <i>Dashboard Admin</i>	Sukses	<i>Valid</i>
2	Halaman <i>Login</i>	Isi <i>Username Admin</i> dan <i>Password</i> yang belum didaftarkan sebelumnya	Jika Gagal tidak akan masuk ke Halaman <i>Dashboard</i> dan tetap berada di Halaman <i>Login</i>	Sukses	<i>Valid</i>

Tabel 4. 1 *Blackbox Login Admin*

b. *Black Box* Halaman *Post Info*

No.	Halaman	Test Case	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman <i>Post Info</i>	Klik Tombol <i>Add New Article</i>	Akan muncul Modal Form untuk menambahkan Arikel / Informasi	Sukses	<i>Valid</i>
2	Halaman <i>Post Info</i>	Isi <i>Form</i> Artikel / Informasi untuk menambahkan Informasi	Jika Sukses maka Data Informasi akan bertambah	Sukses	<i>Valid</i>
3	Halaman <i>Post Info</i>	Klik Tombol Edit pada salah satu Data	Jika Sukses maka akan ke Halaman <i>Form</i> Edit Info	Sukses	<i>Valid</i>
4	Halaman <i>Post Info</i>	Ubah Data Info	Jika Sukses maka akan kembali ke Halaman <i>Post Info</i> dan Data akan <i>terupdate</i>	Sukses	<i>Valid</i>
5	Halaman <i>Post Info</i>	Klik Tombol <i>Delete</i> pada salah satu data	Jika sukses maka akan menghapus data yang dipilih	Sukses	<i>Valid</i>

Tabel 4. 2 *Blackbox Post Info*

c. *Black Box* Halaman *Post Dosen*

No.	Halaman	Test Case	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman <i>Post Dosen</i>	Klik Tombol <i>Add New Dosen</i>	Akan muncul modal <i>Form</i> untuk menambahkan Dosen	Sukses	<i>Valid</i>
2	Halaman <i>Post Dosen</i>	Isi Form Dosen untuk menambahkan Dosen	Jika sukses maka Data Dosen akan bertambah	Sukses	<i>Valid</i>
3	Halaman <i>Post Dosen</i>	Klik Tombol <i>Edit</i> pada salah satu data	Jika sukses maka akan ke Halaman <i>Form Edit</i> Dosen	Sukses	<i>Valid</i>
4	Halaman <i>Post Dosen</i>	Ubah Data info	Jika sukses maka akan kembali ke Halaman <i>Post Dosen</i> dan data akan ter <i>update</i>	Sukses	<i>Valid</i>
5	Halaman <i>Post Dosen</i>	Klik Tombol <i>Delete</i> pada salah satu Data	Jika sukses maka akan menghapus data yang dipilih	Sukses	<i>Valid</i>

Tabel 4. 3 *Blackbox Post Dosen*d. *Black Box* Halaman *Post Eprints*

No.	Halaman	Test Case	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman <i>Post Eprints</i>	Klik Tombol <i>Confirm</i> pada Data yang dipilih	Jika Sukses Data yang dipilih akan terkonfirmasi dan terhapus	Sukses	<i>Valid</i>

Tabel 4. 4 *Blackbox Post Eprints*e. *Black Box* Halaman Judul

No.	Halaman	Test Case	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman <i>Post Judul</i>	Input Judul Skripsi yang ingin dicari	Jika Sukses Data yang dicari akan	Sukses	<i>Valid</i>

		pada form pencarian	ditampilkan		
2	Halaman <i>Post Judul</i>	Input Judul Skripsi yang ingin dicari pada form pencarian	Jika Gagal akan menampilkan pemberitahuan tidak adanya data yang dicari	Sukses	<i>Valid</i>

Tabel 4. 5 Blackbox Judul

f. *Black Box* Halaman Utama

No.	Halaman	<i>Test Case</i>	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman Utama	Klik Tombol <i>Chatbot</i>	Akan masuk ke Halaman <i>Chatbot</i>	Sukses	<i>Valid</i>

Tabel 4. 6 Blackbox Utama

g. *Black Box* Halaman Info Dosen

No.	Halaman	<i>Test Case</i>	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman Info Dosen	Isi nama Dosen pada <i>Form</i> pencarian	Jika Berhasil maka akan menampilkan data dosen	Sukses	<i>Valid</i>
2	Halaman Info Dosen	Isi nama Dosen yang tidak terdaftar pada <i>Form</i> pencarian	Jika Gagal maka tidak ada data Dosen yang muncul	Sukses	<i>Valid</i>

Tabel 4. 7 Blackbox Info Dosen

h. *Black Box* Halaman Pendaftaran Eprints

No.	Halaman	<i>Test Case</i>	Hasil Yang Diharapkan	Hasil	Keterangan
1	Halaman <i>Eprints</i>	Isi data diri lengkap dan benar pada <i>Form</i> pendaftaran	Jika Berhasill maka akan muncul <i>Alert</i> Sukses	Sukses	<i>Valid</i>
2	Halaman <i>Eprints</i>	Isi data diri belum lengkap dan benar pada	Jika Gagal maka akan muncul <i>Alert</i> Gagal	Sukses	<i>Valid</i>

		<i>Form</i> pendaftaran			
--	--	----------------------------	--	--	--

Tabel 4. 8 *Blackbox* Form Eprints

## **BAB V**

### **PENUTUPAN**

Setelah mengadakan penelitian, merancang sistem informasi, mengimplementasi dan merancang, maka dalam bab ini penulis akan memberikan kesimpulan dan saran yang diharapkan dapat membantu dalam kemajuan Sistem Informasi Skripsi Jurusan Teknik Informatika di Universitas Pamulang

#### **5.1 Kesimpulan**

Berdasarkan hasil pembahasan yang telah diuraikan pada bab – bab sebelumnya, maka penulis dapat menarik kesimpulan sebagai berikut:

- a. Dengan adanya sistem informasi skripsi jurusan teknik informatika di universitas pamulang ini dapat membantu mahasiswa dalam proses mencari informasi skripsi yang sedang mengerjakan skripsi dengan efektif dan interaktif.

#### **5.2 Saran**

Adapun saran yang dapat penulis berikan dalam Skripsi ini adalah perlu dikembangkanya program ini dengan mencakup beberapa hal sebagai berikut:

- a. UI aplikasi masih terlalu sederhana, sehingga masih bisa diperbaiki sehingga lebih menarik.
- b. Diharapkan pada penelitian atau pengembangan aplikasi selanjutnya agar membangun sebuah sistem yang dapat meliputi seluruh kegiatan atau penambahan fitur yang belum terdapat di aplikasi ini.



## DAFTAR PUSTAKA

**Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015).** *Pengujian aplikasi menggunakan black box testing boundary value analysis (studi kasus: Aplikasi prediksi kelulusan smnptn).* *Jurnal Ilmiah Teknologi Infomasi Terapan*, 1(3)

**Chandra, A. Y., Kurniawan, D., & Musa, R. (2020).** Perancangan Chatbot Menggunakan Dialogflow Natural Language Processing (Studi Kasus: Sistem Pemesanan pada Coffee Shop). *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 4(1), 208-215.

**Radhian, D. (2019).** *Pembangunan Aplikasi Chatbot Sebagai Media Pencarian Informasi Dalam Bidang Peternakan* (Doctoral dissertation, Universitas Komputer Indonesia).

**Rahmiati, R. (2019).** *Perancangan Sistem Informasi Desa Tegal Sari Kecamatan Satui Menggunakan Framework Codeigniter* (Doctoral dissertation, University of Muhammadiyah Malang).

**Permana, E. (2017).** *Perancangan Sistem Informasi Akuntansi Pada Bina Plus Cisarua Dengan Menggunakan PHP Framework Codeigniter Dan MYSQL* (Doctoral dissertation, Universitas Komputer Indonesia).

**Widyantika, N. B. (2018).** *Perancangan Chatbot Menggunakan Rivescript Pada Website E-Commerce Sebagai Virtual Customer Service* (Doctoral dissertation, University of Muhammadiyah Malang).

**Feladi, V. (2016).** Perancangan Sistem Informasi Penilaian Data Siswa Sma Pancasila Kabupaten Kubu Raya. *Jurnal Pendidikan Informatika dan Sains*, 5(1), 142-156.

**Bianto, M. A., Rahayu, S., Huda, M., & Kusrini, K. (2018).** Perancangan Sistem Pendeteksi Plagiarisme Terhadap Topik Penelitian Menggunakan Metode K-Means Clustering dan Model Bayesian. *SEMNASTEKNOMEDIA ONLINE*, 6(1), 1-2.

**Rosmalina, R., & Nurdesni, A. (2019).** Analisis Dan Perancangan Sistem Informasi Keuangan Untuk Pembayaran Dan Tabungan Siswa Pada Bank Mini Di Smk Bppi Baleendah. *Akurat/ Jurnal Ilmiah Akuntansi Fe Unibba*, 10(2), 70-79.

**Khaidir.2004.** *Microsoft Visual Basic 6.0*. Jakarta : Elex Media Komputindo.

**Supriyanto.2005.** *Perancangan Aplikasi*. Surabaya : Widyastana

**Susanti, F. (2014).** *Aplikasi Pengolahan Data Distribusi Farmasi Berbasis Web Pada PT Mega Mulia Pharma Palembang* (Doctoral dissertation, Politeknik Negeri Sriwijaya).

**Risdiansyah, D. (2017).** Perancangan Sistem Informasi Bimbingan Konseling Berbasis Desktop pada SMA Kemala Bhayangkari 1 Kubu Raya. *Jurnal Khatulistiwa Informatika*, 5(2).

**Dianty, A. (2016).** *Perancangan Sistem Informasi Akuntansi Penggajian Berbasis Komputer Pada Pt Ladang Makmur* (Doctoral dissertation, Politeknik Negeri Sriwijaya).

**Suwarno, F., & Yasmindra, H. (2019).** Sistem Informasi Pengalokasian Kavling Siap Bangun (Ksb) Di Direktorat Pemanfaatan Aset Bp Batam Berbasis Web. *Zona Komputer: Program Studi Sistem Informasi Universitas Batam*, 7(3).

**Sabharwal, N., & Agrawal, A. (2020).** Introduction to Google Dialogflow. In *Cognitive Virtual Assistants Using Google Dialogflow* (pp. 13-54). Apress, Berkeley, CA.

**Google. (n.d.). Docs. Retrieved July 23, 2018, from Dialogflow:** <https://dialogflow.com/docs>

**Sari, T. R. (2017).** *Aplikasi Pendataan Barang Keperluan Praktikum Berbasis Android Di Jurusan Teknik Komputer Politeknik Negeri Sriwijaya* (Doctoral Dissertation, Politeknik Negeri Sriwijaya).

**Wibowo, Y., & Hermawan, A. (2019).** *Rancang Bangun Alat Pengaman Pintu Kamar Hotel Berbasis Password Terpusat* (Doctoral dissertation, University of Technology Yogyakarta).

**Dzulhaq, M. I., Tullah, R., & Nugraha, P. S. (2017).** Sistem Informasi Akademik Sekolah Berbasis Kurikulum 2013. *Jurnal Sisfotek Global*, 7(1).

**Fadillah, N. N. (2016).** *Aplikasi Web Pemesanan Gedung Aula Pada Smk Negeri 3 Palembang* (Doctoral Dissertation, Politeknik Negeri Sriwijaya).

**Kusniawan, A., & Sardiarto, S. (2016).** Perancangan Website Jasa Desain Interior Sebagai Media Pemasaran Studi Kasus: CV. Focalpoint Interior. *EVOLUSI: Jurnal Sains dan Manajemen*, 4(2).

**Monica, T. (2018).** *Web Online Layanan Informasi Siswa Mipa Dan Bahasa Inggris Pada Ekselen Palembang* (Doctoral Dissertation, Politeknik Negeri Sriwijaya).

**Kiki Rizki, K. R. (2019).** *Rancang Bangun Aplikasi Jango (Kunjungan) Desa Berbasis Web (Studi Kasus Desa Mama Kabupaten Sumbawa)* (Doctoral Dissertation, Universitas Teknologi Sumbawa).