

**PERANCANGAN DAN IMPLEMENTASI SISTEM INFORMASI
ADMINISTRASI PADA LABORATORIUM KIMIA FAKULTAS
MIPA UNIVERSITAS NEGERI JAKARTA**

SKRIPSI

Disusun untuk melengkapi syarat-syarat guna memperoleh gelar
Sarjana Ilmu Komputer



**Oleh:
MIKAEL YURUBELI
3145136196**

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS NEGERI JAKARTA**

2017

LEMBAR PERSETUJUAN HASIL SIDANG SKRIPSI

Perancangan dan Implementasi Sistem Informasi Administrasi Pada Laboratorium Kimia Fakultas MIPA Universitas Negeri Jakarta

Nama : Mikael Yurubeli

No. Registrasi : 3145136196

	Nama	Tanda Tangan	Tanggal
Penanggung Jawab			
Dekan	: Prof. Dr. Suyono, M.Si. NIP. 19671218 199303 1 005
Wakil Penanggung Jawab			
Wakil Dekan I	: Dr. Muktiningsih, M.Si. NIP. 19640511 198903 2 001
Ketua	: Ir. Fariani Hermin I, M.T. NIP. 19600211 198703 2 001
Sekretaris	: Ratna Widyati, S.Si, M.Kom. NIP. 19750925 200212 2 002
Penguji	: Drs. Mulyono, M.Kom. NIP. 19660517 199403 1 003
Pembimbing I	: Ria Arafyah, M.Si. NIP. 19751121 200501 2 004
Pembimbing II	: Med Irzal, M.Kom. NIP. 19770615 200312 1 001

Dinyatakan lulus ujian skripsi tanggal: 11 Agustus 2017

HALAMAN PERSEMBAHAN

*Untuk Mama, Papa,
Kakak dan adikku,
serta sahabat-sahabatku.*

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena hanya dengan rahmat-Nya, Tugas Akhir ini dapat terselesaikan tanpa halangan berarti. Keberhasilan dalam menyusun Skripsi ini tidak lepas dari bantuan berbagai pihak yang dengan tulus dan ikhlas memberikan saran dan dukungan dalam menyelesaikan Skripsi ini. Oleh karena itu dalam kesempatan ini, dengan kerendahan hati penulis mengucapkan terima kasih kepada:

1. Bapak Drs. Mulyono, M.Kom, selaku Koordinator Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Jakarta dan dosen pembimbing akademik penulis,
2. Ibu Ria Arafiyah, M.Si. selaku dosen pembimbing pertama yang telah memberikan banyak bantuan, bimbingan, serta arahan dalam menyusun Skripsi ini,
3. Bapak Med Irzal, M.Kom. selaku dosen pembimbing kedua yang juga telah memberikan banyak bantuan, bimbingan, serta arahan dalam menyusun Skripsi,
4. Bapak Muhammad Eka Suryana, M.Kom. dan Bapak Agus Agung Permana, S.Si. selaku dosen dan pembina DEFAULT yang telah memberikan pengajaran dan arahan selama penulis tergabung sebagai anggota,
5. Seluruh Dosen Prodi Ilmu Komputer FMIPA UNJ, yang tidak bisa disebutkan satu-satu, atas ilmu dan bimbingannya selama penulis berkuliah di Program Studi Ilmu Komputer FMIPA UNJ,
6. Papa dan Mama yang selama ini telah sabar membimbing, mengarahkan, dan mendoakan penulis tanpa kenal lelah untuk selama-lamanya,

7. Tissa Marlina Yurubeli dan Andreas Buluama yang senantiasa memberi dukungan kepada penulis,
8. Sahabat-Sahabat yang selalu memberi semangat dan bertukar pikiran dalam menyelesaikan Skripsi,
9. Teman-teman Ilmu Komputer 2013 yang tidak dapat disebutkan satu persatu atas kebersamaan dan semangat selama menjalani masa pembelajaran di Program Studi Ilmu Komputer, dan
10. Seluruh kerabat yang tidak dapat penulis sebutkan satu persatu atas doa dan dukungan yang diberikan kepada penulis.

Penulis menyadari bahwa penyusunan Skripsi ini jauh dari pada sempurna dan masih banyak kekurangan dalam penguasaan materi. Akhir kata penulis mohon maaf yang sebesar-besarnya apabila ada kekeliruan di dalam penulisan Skripsi ini.

Jakarta, 7 Agustus 2017

Penulis

DAFTAR ISI

HALAMAN PERSEMBAHAN	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL	xii
ABSTRAK	xiii
<i>ABSTRACT</i>	xiv
I PENDAHULUAN	1
A. Latar Belakang Masalah	1
B. Rumusan Masalah	2
C. Tujuan Penelitian	3
D. Manfaat Penelitian	3
II KAJIAN PUSTAKA	4
A. Administrasi	4
1. Pengertian Administrasi	4
2. Fungsi Administrasi	4
B. Sistem Informasi Administrasi	5
1. Sistem	5
2. Informasi	5
3. Sistem Informasi	6

4.	Sistem Informasi Administrasi	6
C.	Pengembangan Aplikasi Perangkat Lunak	7
1.	<i>Model Waterfall</i>	7
D.	<i>Unified Modelling Language (UML)</i>	11
1.	<i>Use Case Diagram</i>	12
2.	<i>Activity Diagram</i>	13
3.	<i>Class Diagram</i>	14
E.	<i>Entity Relationship Diagram</i>	14
F.	Arsitektur <i>Model-View-Controller (MVC)</i>	15
G.	<i>Framework Codeigniter</i>	17
III IMPLEMENTASI PROGRAM		19
A.	Tahap Penelitian	19
B.	Analisis Kebutuhan	20
1.	Analisis Kebutuhan <i>Hardware</i> dan <i>Software</i>	20
2.	Analisis Kebutuhan Sistem	21
C.	Desain Sistem (<i>Design</i>)	23
1.	<i>Use Case Diagram</i>	23
2.	<i>Activity Diagram</i>	24
3.	<i>Class Diagram</i>	26
4.	<i>Entity Relationship Diagram</i>	28
5.	Desain Antarmuka <i>User Interface</i>	30
D.	Implementasi Program	33
1.	Membangun Basis Data (<i>Database</i>)	33
2.	Implementasi Desain Antarmuka Sistem	34
3.	Implementasi Sistem (Pemrograman)	39
4.	Integrasi <i>Codeigniter</i> dengan <i>Library FPDF</i>	45

IV UJI COBA DAN HASIL UJI COBA	47
A. Uji Coba	47
B. Hasil Uji Coba	49
1. Administrator	49
2. Mahasiswa	51
3. Kepala Laboratorium	53
C. Pemeliharaan (<i>Maintenance</i>)	54
V KESIMPULAN DAN SARAN	55
A. Kesimpulan	55
B. Saran	55
DAFTAR PUSTAKA	58
LAMPIRAN	59
A Implementasi Desain (Administrator)	59
B Sampel Kode Pemrograman	64
C Analisis Kebutuhan Sistem	70
D <i>User Acceptance Test</i> Administrator	73
E <i>User Acceptance Test</i> Mahasiswa	75
F <i>User Acceptance Test</i> Kepala Laboratorium	77

DAFTAR GAMBAR

Gambar 2.1	Ilustrasi <i>Model Waterfall</i>	11
Gambar 2.2	Contoh <i>Use Case Diagram</i>	12
Gambar 2.3	Contoh <i>Activity Diagram</i>	13
Gambar 2.4	Ilustrasi Arsitektur <i>Model-View-Controller</i>	17
Gambar 3.1	Diagram <i>Use Case</i> Sistem Informasi Administrasi	24
Gambar 3.2	Diagram <i>Activity</i> dari Sistem Informasi Administrasi	25
Gambar 3.3	Diagram <i>Class</i> dari Sistem Informasi Administrasi	27
Gambar 3.4	<i>Entity Relationship Diagram</i> dari sistem informasi administrasi	29
Gambar 3.5	Halaman Login Sistem Informasi Administrasi	30
Gambar 3.6	Halaman Data User Sistem Informasi Administrasi	30
Gambar 3.7	Halaman Beranda Sistem Informasi Administrasi	31
Gambar 3.8	Halaman Praktikum Sistem Informasi Administrasi	31
Gambar 3.9	Halaman Penelitian Sistem Informasi Administrasi	32
Gambar 3.10	Halaman Peminjaman Sistem Informasi Administrasi	32
Gambar 3.11	Sampel Syntax Query untuk Membangun Database Sistem	33
Gambar 3.12	Database Sistem Informasi Administrasi Laboratorium	34
Gambar 3.13	Halaman Login dari Sistem Informasi Administrasi	35
Gambar 3.14	Halaman Data Diri dari Sistem Informasi Administrasi	35
Gambar 3.15	Halaman Awal dari Sistem Informasi Administrasi	36
Gambar 3.16	Halaman Jadwal Praktikum dari Sistem Informasi Administrasi	36
Gambar 3.17	Halaman Pendaftaran Praktikum dari Sistem Informasi Admi- nistrasi	37
Gambar 3.18	Halaman Peminjaman dari Sistem Informasi Administrasi	38
Gambar 3.19	Halaman Penelitian dari Sistem Informasi Administrasi	38

Gambar 3.20	Halaman Pendaftaran Asisten dari Sistem Informasi Administrasi	39
Gambar 3.21	Proses Pembuatan Sebuah Fitur	40
Gambar 3.22	Sampel Kode Model Login dari Sistem Informasi Administrasi	40
Gambar 3.23	Sampel Kode View Login dari Sistem Informasi Administrasi	41
Gambar 3.24	Sampel Kode Controller Login dari Sistem Informasi Administrasi	41
Gambar 3.25	Sampel Kode Controller Praktikum dari Sistem Informasi Administrasi	42
Gambar 3.26	Sampel Kode Model Penelitian dari Sistem Informasi Administrasi	43
Gambar 3.27	Sampel Kode View Peminjaman dari Sistem Informasi Administrasi	44
Gambar 3.28	Penggunaan Library FPDF pada Sistem Informasi Administrasi	45
Gambar 3.29	PDF Hasil dari Penggunaan Library FPDF	46
Gambar 1.1	Halaman Login dari Sistem Informasi Administrasi	59
Gambar 1.2	Halaman Awal dari Sistem Informasi Administrasi	59
Gambar 1.3	Halaman Tambah Praktikum dari Sistem Informasi Administrasi	60
Gambar 1.4	Halaman Jadwal Praktikum dari Sistem Informasi Administrasi	60
Gambar 1.5	Halaman Peserta Praktikum dari Sistem Informasi Administrasi	61
Gambar 1.6	Halaman Peminjaman dari Sistem Informasi Administrasi . .	61
Gambar 1.7	Halaman Tambah Alat dari Sistem Informasi Administrasi . .	62
Gambar 1.8	Halaman Asisten dari Sistem Informasi Administrasi	62
Gambar 1.9	Halaman Asisten dari Sistem Informasi Administrasi	63

Gambar 2.1	Sampel Kode Controller dari Fitur Praktikum	64
Gambar 2.2	Sampel Kode Model dari Fitur Praktikum	64
Gambar 2.3	Sampel Kode View dari Fitur Praktikum	65
Gambar 2.4	Sampel Kode Controller dari Fitur Penelitian	65
Gambar 2.5	Sampel Kode Model dari Fitur Penelitian	66
Gambar 2.6	Sampel Kode View dari Fitur Penelitian	66
Gambar 2.7	Sampel Kode Controller dari Fitur Peminjaman	67
Gambar 2.8	Sampel Kode Model dari Fitur Peminjaman	67
Gambar 2.9	Sampel Kode View dari Fitur Peminjaman	68
Gambar 2.10	Sampel Kode Controller dari Fitur Asisten	68
Gambar 2.11	Sampel Kode Model dari Fitur Asisten	69
Gambar 2.12	Sampel Kode View dari Fitur Asisten	69

DAFTAR TABEL

Tabel 3.1	Tabel analisis <i>user</i> Sistem Informasi Administrasi	21
Tabel 3.2	Tabel analisis menu Sistem Informasi Administrasi	22
Tabel 4.1	Komponen Penilaian Uji Coba Sistem Administrator	50
Tabel 4.2	Komponen Penilaian Uji Coba Sistem Mahasiswa	52
Tabel 4.3	Komponen Penilaian Uji Coba Sistem Kepala Laboratorium .	53

ABSTRAK

Sistem informasi administrasi adalah kumpulan dari komponen-komponen yang saling berhubungan satu dengan yang lain guna mendukung proses pendataan, pengaturan, pengalokasian dan pengawasan suatu kegiatan untuk mencapai tujuan tertentu. Pada kajian skripsi ini, dipaparkan alternatif terhadap permasalahan administrasi yang terjadi yaitu dengan merancang dan mengimplementasikan sistem informasi administrasi laboratorium kimia yang mengakomodasi kebutuhan administrasi dan dikembangkan dengan metode *System Development Life Cycle (SDLC) model waterfall*. Pertama kali mahasiswa *login*, mahasiswa diminta untuk melengkapi data diri yang nantinya akan menjadi sumber data bagi sistem sehingga setiap kali mahasiswa melakukan permohonan, tidak diperlukan lagi permintaan data diri yang berulang-ulang. Selanjutnya sistem dapat menampilkan daftar praktikum yang telah ditempuh oleh mahasiswa sebagai rekap untuk pihak laboratorium. Pada proses inventaris, sistem informasi administrasi dengan sendirinya akan melakukan pengurangan atau penambahan pada stok alat dan bahan kimia jika terjadi peminjaman.

Kata kunci : Administrasi, Sistem informasi administrasi, laboratorium kimia, *model waterfall*, inventaris.

ABSTRACT

Information system administration is a collection of components that are interconnected with each other to support the process of data entry, arrangement, allocation and supervision of an activity to achieve certain goals. In this thesis study, presented an alternative to the administrative problems that occurred in the chemical laboratory that is by designing and implementing information systems administration that accommodates the needs of the existing administration and was developed by System Development Life Cycle (SDLC) model waterfall. The first time students login to system, students are asked to complete their own data which will be a source for the system so that every time students make a request so there is no need for repetitive data requests. Furthermore the system can display a list of practicum course that have been taken by students as a recap for the laboratory. In the inventory process, the information system administration by itself will make reductions or additions to the stock of chemistry tools and chemicals.

Keywords : *administration, Information systems administration, chemistry laboratory, model waterfall, inventory.*

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Administrasi merupakan salah satu kegiatan yang selalu ada dalam setiap aspek kehidupan manusia. Dalam administrasi terdapat sekumpulan proses yang meliputi pengarsipan dokumen, surat-menyurat, pembukuan dan lain sebagainya yang bersifat teknis ketatausahaan baik menggunakan media kertas maupun secara digital.

Dalam dunia pendidikan khususnya perguruan tinggi, proses administrasi yang baik dan teratur dapat menunjang kegiatan belajar mengajar serta meningkatkan efisiensi dan mutu dari layanan administrasi itu sendiri. Seperti halnya dengan proses administrasi yang dilakukan di laboratorium kimia pada Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta yang bertujuan menunjang kegiatan akademik di lingkungan laboratorium kimia.

Laboratorium kimia sebagai salah satu sarana akademik di Universitas Negeri Jakarta, memberikan layanan kepada dosen dan mahasiswa program studi pendidikan kimia dan kimia murni serta mahasiswa lintas program studi (fisika dan biologi) untuk dapat melakukan beberapa kegiatan akademik di lingkungan laboratorium kimia. Kegiatan akademik yang ada dan dilakukan di laboratorium kimia adalah kegiatan praktikum, peminjaman alat dan bahan kimia, peminjaman laboratorium, pendaftaran asisten laboratorium serta perizinan penelitian di lingkungan laboratorium kimia.

Sebelum dapat melakukan kegiatan akademik yang telah disebutkan, baik mahasiswa atau dosen diharuskan untuk mengikuti proses administrasi yang ada. Dalam hal ini, proses administrasi yang harus dilakukan berupa mengisi *form* atau dokumen yang disediakan oleh pengurus laboratorium kimia. *Form* atau dokumen tersebut ber-

guna untuk keperluan pengarsipan dan pendataan kegiatan akademik yang terjadi, yang nantinya akan digunakan sebagai acuan pembuatan laporan untuk Kepala Laboratorium Kimia.

Seperti yang telah disebutkan, proses administrasi pada laboratorium kimia masih menggunakan media konvensional, yaitu dengan menggunakan *form* atau dokumen dalam pendataan dan pengarsipan yang rentan akan risiko rusak atau hilangnya data, serta lamanya waktu dalam pencarian dan memproses data [9].

Guna meningkatkan efisiensi waktu dalam pencarian dan memproses data, meningkatkan keamanan data serta mutu dari pelayanan administrasi, dibutuhkan sebuah teknologi atau sebuah sistem yang mampu mengakomodasi seluruh kebutuhan administrasi yang telah disebutkan. Dengan adanya sistem tersebut, maka akan membantu dalam proses pendataan dan manajemen pada laboratorium, mulai dari pendaftaran praktikum, penelitian, inventarisasi alat dan bahan, pendaftaran asisten laboratorium serta laporan kegiatan [7].

Berdasarkan hal tersebut penulis memberikan alternatif dengan merancang sebuah sistem informasi administrasi yang mampu mengakomodasi seluruh kebutuhan administrasi pada Laboratorium Kimia Fakultas MIPA di Universitas Negeri Jakarta. Sistem informasi administrasi tersebut hanya melayani atau menangani mahasiswa dengan jenjang pendidikan strata-1 dan proses seleksi dilakukan secara manual (sistem hanya menangani *form pendaftaran*).

B. Rumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah dalam penelitian ini adalah bagaimana perancangan dan implementasi dari sistem informasi administrasi pada laboratorium kimia fakultas matematika dan ilmu pengetahuan alam di Universitas Negeri Jakarta?

C. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk merancang dan mengimplementasikan sistem informasi administrasi pada laboratorium kimia fakultas matematika dan ilmu pengetahuan alam di Universitas Negeri Jakarta.

D. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut :

1. Menjadi alternatif dalam menghadapi permasalahan administrasi yang terdapat pada proses administrasi laboratorium kimia.
2. Sebagai gambaran dan bahan referensi untuk penelitian dalam membangun sistem informasi administrasi.

BAB II

KAJIAN PUSTAKA

A. Administrasi

1. Pengertian Administrasi

Menurut Ulbert Silalahi, administrasi didefinisikan dalam dua pengertian, yaitu administrasi dalam arti yang sempit dan luas. Administrasi dalam arti yang sempit dapat diartikan sebagai penyusunan dan pencatatan data secara sistematis yang bertujuan untuk menyediakan keterangan atau informasi serta memudahkan dalam proses pencarian jika diperlukan kembali. Jadi administrasi dalam arti sempit lebih tepat disebut tata usaha (*clerical work, office work*), sedangkan administrasi dalam arti luas adalah segala sesuatu yang berhubungan dengan kegiatan kerjasama yang dilakukan oleh manusia atau sekelompok orang untuk mencapai tujuan tertentu [10].

Administrasi meliputi kegiatan individu-individu dalam suatu organisasi yang bertugas untuk mengatur, memajukan dan menyediakan fasilitas dalam usaha kerjasama kelompok untuk mencapai suatu tujuan yang telah ditentukan.

Dari penjelasan di atas, maka dapat disimpulkan administrasi adalah proses pengarsipan, pengaturan dan penyimpanan informasi pada suatu media secara sistematis dengan tujuan tertentu sehingga informasi tersebut dapat dengan mudah ditemukan apabila diperlukan kembali.

2. Fungsi Administrasi

Untuk mencapai efektifitas dan efisiensi yang baik dalam proses administrasi, setiap kegiatan administrasi harus dilaksanakan dengan pertimbangan dan perhitungan yang tepat. Untuk menciptakan pertimbangan dan perhitungan yang tepat maka

diperlukan langkah-langkah kegiatan yang dapat disebut sebagai fungsi pokok administrasi. Adapun fungsi pokok dari administrasi adalah sebagai berikut :

- a. Perencanaan (*Planning*)
- b. Pengorganisasian (*Organizing*)
- c. Pengarahan (*Direction*)
- d. Koordinasi (*Coordination*)
- e. Pengawasan (*Controlling*)
- f. Pelaporan (*Reporting*)

B. Sistem Informasi Administrasi

1. Sistem

Menurut Jogiyanto, sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu [4]. Menurut Azhar Susanto, sistem adalah kumpulan dari sub sistem atau bagian atau komponen apapun baik fisik ataupun non-fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu [12].

Berdasarkan beberapa pendapat yang dikemukakan di atas maka dapat ditarik kesimpulan bahwa sistem adalah kumpulan bagian-bagian atau subsistem-subsistem yang disatukan dan dirancang untuk mencapai suatu tujuan.

2. Informasi

Informasi merupakan hal yang sangat penting bagi sebuah organisasi dalam mengambil sebuah keputusan. Menurut Agus Mulyanto, informasi adalah data yang

diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya, sedangkan data merupakan sumber informasi yang menggambarkan suatu kejadian yang nyata [6]. Sedangkan menurut Jogiyanto, informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya [4].

3. Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang memperhatikan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan [4].

Sistem informasi dapat diartikan sebagai kumpulan elemen-elemen atau komponen yang saling berhubungan guna mengumpulkan (*input*), memanipulasi (proses) dan menghasilkan (*output*) data dan informasi serta menyediakan mekanisme balasan untuk mencapai tujuan tertentu. Mekanisme balasan membantu suatu organisasi dalam mencapai tujuannya seperti meningkatkan keuntungan atau memperbaiki pelayanan pelanggan [8].

4. Sistem Informasi Administrasi

Berdasarkan teori-teori yang telah diuraikan di atas mengenai sistem informasi dan administrasi serta fungsinya, maka dapat diartikan bahwa sistem informasi administrasi adalah kumpulan atau kesatuan dari komponen-komponen atau elemen-elemen yang saling berhubungan satu dengan yang lain guna mendukung proses pendataan, pengaturan, pengalokasian dan pengawasan suatu kegiatan untuk mencapai tujuan tertentu.

C. Pengembangan Aplikasi Perangkat Lunak

Dalam mengembangkan aplikasi perangkat lunak dibutuhkan alur atau prosedur agar aplikasi dapat diselesaikan dengan baik. Dalam pengembangan aplikasi perangkat lunak terdapat metode pengembangan yang dikenal dengan *System Development Life Cycle*.

System Development Life Cycle merupakan suatu urutan atau prosedur dari beberapa proses yang dilakukan secara bertahap di dalam merancang dan mengembangkan sistem, yang juga dikenal dengan nama *Information System Development* atau *Application Development*. *System Development Life Cycle* (SDLC) atau Siklus Hidup Pengembangan Sistem adalah metode pengembangan sistem tradisional yang digunakan sebagian besar organisasi saat ini [14].

Terdapat beberapa model pengembangan sistem yang ditawarkan oleh *System Development Life Cycle*, diantaranya: *Model Waterfall*, *Spiral*, *Prototyping*, *Rapid Application Development*, *Object Oriented Technology* dan lain sebagainya. Dalam penelitian ini penulis menggunakan model pengembangan sistem *waterfall*.

1. Model Waterfall

Model waterfall pertama kali diperkenalkan oleh Winston W. Royce pada tahun 1970. *Model waterfall* merupakan model klasik yang sederhana dengan aliran sistem yang linier. *Model waterfall* merupakan proses pengembangan perangkat lunak yang berurutan (*sequential*) di mana prosesnya dari atas ke bawah melalui tahapan-tahapan yang harus dijalankan dalam proses membangun perangkat lunak.

Model pengembangan sistem *waterfall* memiliki beberapa kelebihan, yaitu kualitas dari sistem yang dihasilkan akan baik. Hal ini dikarenakan proses pelaksanaannya dilakukan secara bertahap sehingga tidak terfokus pada tahapan tertentu. Kemudian dokumentasi dari pengembangan sistem sangat terorganisir, karena seti-

ap tahap harus terselesaikan dengan lengkap dan baik sebelum melangkah ke tahap berikutnya.

Pada *model* pengembangan sistem *waterfall* terdapat lima langkah atau lima tahapan yang harus dilaksanakan secara terurut atau bertahap untuk dapat membuat sebuah sistem secara utuh. Keluaran dari setiap tahap merupakan masukan bagi tahap selanjutnya. Berikut adalah penjelasan mengenai lima tahap pada *Model* pengembangan sistem *waterfall* :

a. Analisis (*Analysis*)

Tahap analisis merupakan tahap awal yang dilakukan untuk menganalisa segala kebutuhan terhadap sistem yang akan dikembangkan. Pengumpulan data mengenai kebutuhan dari sistem dapat dilakukan dengan cara observasi, *study literature*, atau dengan metode wawancara kepada pengguna yang akan menggunakan sistem nantinya.

Pada tahap ini bertujuan untuk mengidentifikasi masalah yang ada dan mengidentifikasi kebutuhan dari *user* akan sistem yang akan dikembangkan. Informasi yang terperinci mengenai kebutuhan sistem akan memudahkan dalam proses selanjutnya.

1) Analisa Identifikasi Masalah

Identifikasi masalah merupakan langkah awal dari analisis sistem. Pada tahap ini didefinisikan berbagai masalah yang harus dipecahkan berkaitan dengan sistem yang lama yang kemudian akan menjadi dasar dalam pembuatan sistem baru agar mampu menyelesaikan masalah-masalah yang telah diidentifikasi dari sistem yang sebelumnya.

2) Analisa Kebutuhan

Tahap analisa kebutuhan bertujuan untuk menganalisis kebutuhan dari *user*

akan sistem atau perangkat lunak yang nantinya akan dibangun serta mengembangkan dan mengolah kebutuhan *user* yang telah didapat agar mampu mengakomodasi kebutuhan secara nyata.

b. Desain Sistem (*Design*)

Tahap desain merupakan tahap penerjemahan kebutuhan pengguna atau *user* terhadap sistem. Pembuatan desain sistem didasari oleh data yang telah didapatkan dari tahap analisis dan diubah kedalam bentuk visual. Tahap desain ini berfokus pada: pemodelan sistem, struktur data atau *database*, representasi tampilan antarmuka dan detail dari prosedural sistem.

1) Desain *Model*

Desain model adalah model logika yang menjelaskan kepada *user* mengenai fungsi-fungsi dari sistem informasi yang akan dibangun serta menggambarkan bagaimana fungsi-fungsi dalam sistem informasi akan bekerja secara logika. Desain model dapat digambarkan atau direpresentasikan melalui *class diagram*.

2) Desain *Database*

Desain *database* bertujuan untuk menentukan dan menunjukkan hubungan antara entity dan relasinya (*Entity Relationship Diagram*). Desain *database* dilakukan dengan cara mengidentifikasi data-data yang diperlukan oleh sistem berdasarkan desain model yang telah digambarkan. Langkah selanjutnya adalah menentukan parameter dari data *database* seperti: tipe data, organisasi data dan akses data yang kemudian akan dibentuk tabel-tabel serta relasi antar tabel.

3) Desain Antarmuka (*Design Interface*)

Desain antarmuka adalah tahap yang harus dilakukan sebelum mulai mem-

buat sistem. Tujuan dari pembuatan desain antarmuka ini adalah untuk membuat interaksi pengguna dengan sistem menjadi lebih sederhana dan efisien dalam hal mencapai tujuan pengguna tanpa membuat pengguna kebingungan dalam menggunakan sistem sehingga tidak memicu kesalahan dalam menggunakan sistem oleh pengguna.

c. Implementasi (*Implementation*)

Tahap implementasi merupakan tahap di mana keseluruhan desain dari sistem akan direalisasikan ke dalam bentuk program, *database* atau komponen sistem secara menyeluruh. Dengan kata lain tahap ini adalah tahap di mana kode nyata ditulis dan disusun menjadi sebuah sistem operasional. Pada tahap ini akan dihasilkan sebuah sistem yang siap untuk di uji coba.

d. Uji Coba (*Testing*)

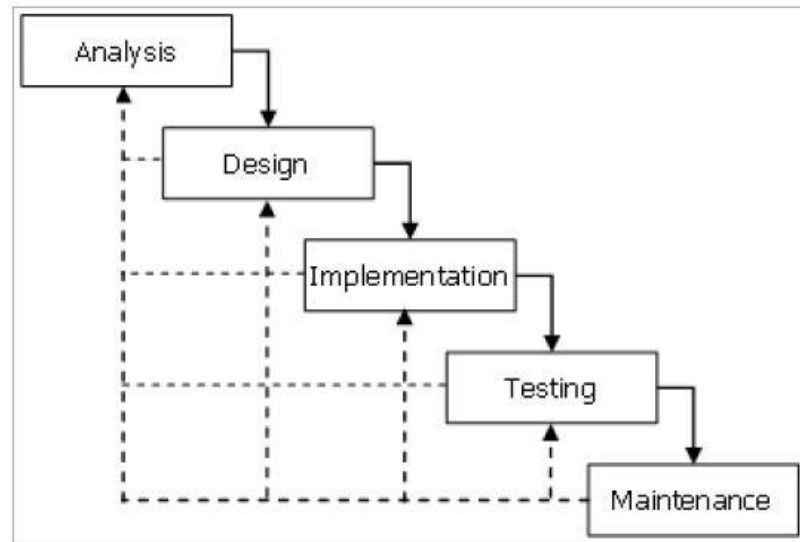
Setelah sebuah sistem diimplementasi, maka akan dilanjutkan dengan tahap uji coba terhadap fungsi kerja dari sistem terhadap *user* dan lingkungan *user*. Setelah melakukan uji coba dan mendapatkan timbal-balik dari hasil pengujian sistem maka akan dilakukan evaluasi dan perbaikan jika terdapat kesalahan pada sistem. Sistem akan kembali di uji sampai seluruh fungsi kerja dari sistem berjalan dengan baik dan telah memenuhi kebutuhan *user*.

e. Pemeliharaan (*Maintenance*)

Setelah melalui tahap uji coba, akan dilanjutkan dengan tahap pemeliharaan. Pada tahap ini akan dilakukan pemeliharaan secara berkala terhadap sistem untuk memastikan sistem tetap berjalan dengan baik serta meningkatkan performa dan kualitas dari sistem dan penambahan fitur jika ada permintaan atau kebutuhan baru dari *user* atau dirasa diperlukan adanya penambahan fitur pada sistem.

Berikut adalah ilustrasi dari tahapan-tahapan yang terdapat dalam SDLC *mo-*

del waterfall :



Gambar 2.1: Ilustrasi *Model Waterfall*

Peneliti memilih menggunakan *model waterfall* sebagai model dari *System Development Life Cycle* dalam proses pengembangan sistem informasi administrasi laboratorium kimia dengan alasan karena kebutuhan dari *user* yang rinci dan jelas, kecilnya persentase penambahan fitur pada sistem informasi administrasi yang akan dikembangkan serta kelebihan-kelebihan dari *model waterfall* yang telah disebutkan sebelumnya.

D. *Unified Modelling Language (UML)*

Dalam mengembangkan sebuah sistem, diperlukan sebuah media atau bahasa pemodelan yang berguna untuk memodelkan atau memvisualisasikan alur kerja atau proses perancangan dari sebuah sistem. Bahasa yang menjadi standar dalam memvisualisasikan proses pengembangan sistem adalah *Unified Modelling Language* atau biasa disebut dengan UML.

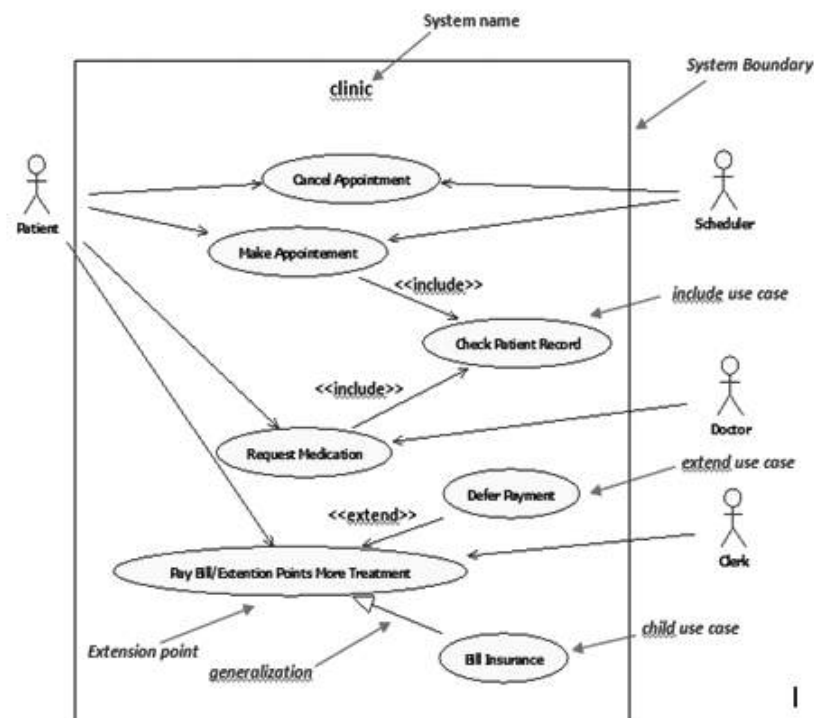
Unified Modelling Language atau UML sendiri adalah sebuah bahasa yang

telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak [11]. Tujuan dari pemodelan ini adalah untuk mempermudah pembuat sistem dalam menjelaskan bagaimana sistem akan bekerja nantinya.

Dalam penelitian ini, peneliti menggunakan tiga jenis diagram yaitu *Use Case Diagram*, *Activity Diagram* dan *Class Diagram*.

1. *Use Case Diagram*

Use case diagram adalah sebuah diagram yang menggambarkan bagaimana setiap aktor berinteraksi dengan sistem tersebut. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. Use case mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan [2].

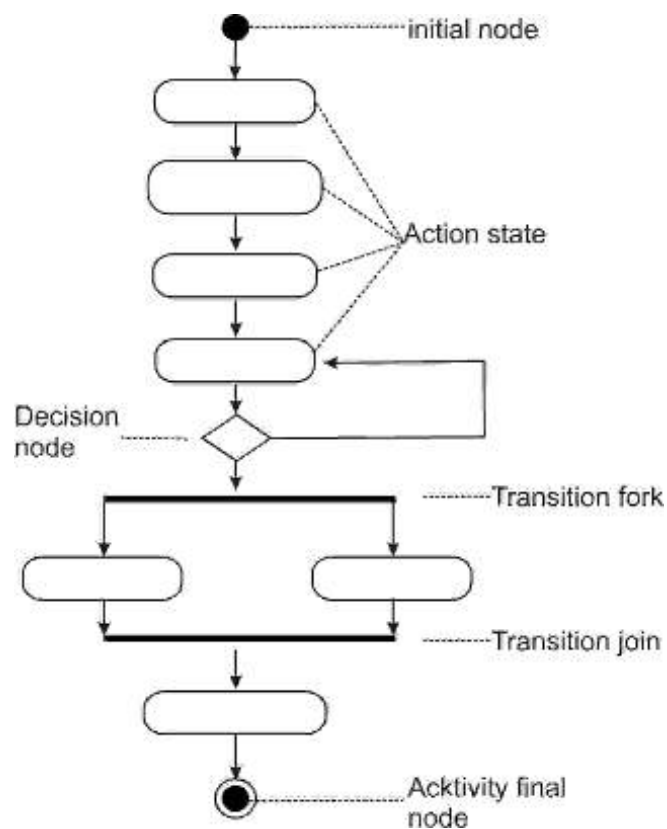


Gambar 2.2: Contoh *Use Case Diagram*

2. Activity Diagram

Activity diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* adalah diagram yang menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

Setiap *activity diagram* selalu mempunyai satu *initial state*. *Initial node* yang digambarkan dengan simbol lingkaran padat, merupakan titik yang mengawali *activity diagram*. *Activity diagram* dapat diakhiri dengan memberikan *activity final diagram* yang digambarkan dengan lingkaran padat dengan mempunyai cincin dibagian luarnya.



Gambar 2.3: Contoh Activity Diagram

3. *Class Diagram*

Class diagram adalah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Sebuah *class* memiliki 3 bagian utama yaitu *attribute*, *operation*, dan *name*. *Class* adalah kumpulan objek-objek yang mempunyai struktur umum, behavior umum, relasi umum, dan semantic/kata yang umum [3]. Semua *class* yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem.

E. *Entity Relationship Diagram*

Menurut Marlinda, *Entity Relationship Diagram* (ERD) adalah diagram dari sistem yang menggambarkan hubungan antar entitas beserta relasinya yang saling terhubung [5]. ERD menyediakan bentuk untuk menunjukkan struktur keseluruhan dari data yang ada. Dalam ERD data-data tersebut digambarkan dengan menggunakan simbol entitas.

Entity Relationship Diagram (ERD) memiliki dua komponen utama yaitu Entitas (*Entity*) dan Relasi (*Relation*). Entitas adalah segala sesuatu dalam dunia nyata yang dapat digambarkan oleh data. Setiap entitas pasti mempunyai sebuah atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Contohnya atribut nim yang berfungsi untuk membedakan mahasiswa yang satu dengan yang lainnya. Pada setiap entitas harus memiliki 1 atribut unik atau yang disebut dengan *primary key*.

Sedangkan relasi adalah hubungan antara beberapa entitas. Di dalam relasi terdapat kardinalitas atau derajat relasi yang menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Berdasarkan hal tersebut, terdapat 3 jenis relasi dalam hubungan antar entitas, yaitu : *one-to-one*, *one-to-many* dan *many-to-many*. Untuk relasi *one-to-one*, setiap entitas hanya boleh

berhubungan dengan satu entitas lainnya, begitu pula sebaliknya. Untuk relasi *one-to-many*, setiap entitas dapat berhubungan dengan lebih dari satu entitas tetapi tidak sebaliknya. Sedangkan untuk relasi *many-to-many*, setiap entitas dapat berhubungan dengan banyak entitas lainnya dan demikian pula sebaliknya.

Didalam suatu hubungan antar entitas dalam ERD, terdapat 2 jenis atribut yaitu *identifier (key)* dan *descriptor (non-key attribute)*. *Identifier* merupakan satu dari banyak anggota suatu entitas yang dapat menjadi *primary key* atau anggota entitas yang unik. Sedangkan *descriptor* digunakan untuk mendeskripsikan karakteristik dari suatu entitas yang tidak unik atau yang tidak terpilih sebagai *primary key*. Didalam satu tabel ERD tidak diperbolehkan memiliki lebih dari satu *primary key*, karena *primary key* merupakan variabel data yang menjadi identitas utama dari tabel tersebut.

Selain *primary key* terdapat istilah lain untuk menggambarkan *key* tamu dalam suatu hubungan entitas yang disebut sebagai *foreign key*. *Foreign key* dapat diartikan sebagai satu atribut yang melengkapi satu relasi yang merujuk ke induk entitas aslinya.

F. Arsitektur Model-View-Controller (MVC)

Model-View-Controller atau MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. Arsitektur MVC memisahkan komponen-komponen pada aplikasi menjadi beberapa bagian, seperti: manipulasi data, *controller*, dan *user interface*. Setiap komponen hanya mengatur bagiannya masing-masing tanpa mencampuri bagian lainnya kecuali pada bagian *controller*.

Model-View-Controller (MVC) adalah sebuah konsep yang diperkenalkan oleh penemu *Smalltalk* (Trygve Reenskaug) untuk meng-ekapsulasi data bersama dengan pemrosesan (*model*), mengisolasi dari proses manipulasi (*controller*) dan tampilan (*view*) untuk direpresentasikan pada sebuah *user interface* [1].

Berikut dijelaskan tiga komponen pembangun arsitektur Model-View-Controller (MVC), yaitu :

1. *Model*

Model mewakili struktur data. Biasanya *model* berisi fungsi-fungsi yang membantu *developer* dalam pengelolaan basis data seperti memasukkan data ke basis data, pembaruan data dan lain-lain. Hanya *model* yang mengandung data dan fungsi yang berhubungan dengan pemrosesan data.

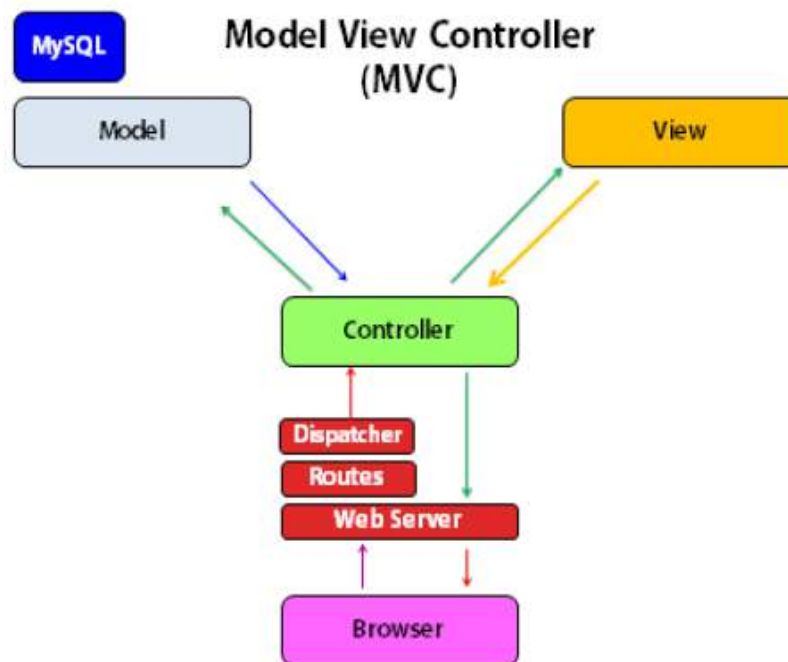
2. *View*

View adalah bagian yang mengatur bagaimana tampilan akan ditampilkan kepada *user*. *View* berkaitan atau berhubungan dengan *model*. Ketika *model* berubah, *view* secara otomatis menggambar ulang bagian layar yang terkena perubahan untuk menunjukkan perubahan tersebut. Satu *model* dapat menjadi rujukan bagi beberapa *view* dan masing-masing *view* tersebut akan menampilkan isi *model* yang dijalankan dengan tampilan yang berbeda pula.

3. *Controller*

Controller merupakan bagian yang menjadi penghubung antara *model* dan *view*. *Controller* dapat dikatakan sebagai jembatan antara *model* dan *View*. *Controller* berfungsi untuk merespon perintah dari *user* yang kemudian akan menentukan bagaimana aplikasi akan dijalankan.

Secara singkat dapat dikatakan bahwa *model* berfungsi untuk mengatur alur *database*, *view* berfungsi untuk mengatur tampilan dari *web*, dan *controller* berfungsi untuk mengatur alur kerja antara *model* dan *view*. Berikut adalah ilustrasi dari konsep *Model-View-Controller*:



Gambar 2.4: Ilustrasi Arsitektur *Model-View-Controller*

Arsitektur *Model-View-Controller* memberikan kemudahan kepada *developer* atau pengembang dengan adanya pemisahan antara *model* dan *view* sehingga memungkinkan beberapa *view* menggunakan *model* yang sama. Sehingga, komponen *model* sebuah aplikasi lebih mudah untuk diterapkan, diuji, dan dipelihara.

G. Framework Codeigniter

Codeigniter adalah sebuah *framework php opensource* yang dibangun dengan konsep MVC (*Model-View-Controller*). *Codeigniter* pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc., yaitu sebuah perusahaan yang memproduksi sebuah CMS (*content management system*) yang disebut dengan *expression engine*.

Codeigniter adalah sebuah *framework PHP* yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis *PHP* tanpa harus menulis semua kode program dari awal. *Codeigniter* menyediakan banyak *library* untuk

mengerjakan tugas-tugas yang umumnya ada pada sebuah aplikasi berbasis *web*.

Codeigniter memiliki seperangkat fungsionalitas yang sangat kaya, yang akan meningkatkan kecepatan kerja dalam mengembangkan sebuah *web*. *Codeigniter* juga memiliki kemampuan untuk mencegah berbagai serangan yang terjadi pada sistem [13]. Selain itu, struktur dan susunan logis dari *codeigniter* membuat penulisan kode program pada sistem yang dibangun menjadi semakin teratur dan rapi sehingga penulisan kode program menjadi lebih minimal.

Berikut adalah kelebihan dari *Framework Codeigniter* [13]:

1. *Codeigniter* berukuran sangat kecil.
2. *Codeigniter* mudah dipelajari, diadopsi dan diterapkan.
3. *Codeigniter* memiliki dokumentasi yang baik.
4. *Codeigniter* mampu berjalan dengan baik pada hampir semua *platform hosting*.
5. Mudah dalam melakukan migrasi dari satu *server hosting* ke *server hosting* lainnya.
6. Tidak ada aturan *coding* yang ketat.
7. Kinerja yang baik.

BAB III

IMPLEMENTASI PROGRAM

A. Tahap Penelitian

Pada penelitian ini, penulis melakukan beberapa tahap dalam melaksanakan penelitian. Terdapat tiga tahapan penelitian yang dilakukan yaitu perencanaan, pelaksanaan dan penulisan. Adapun tahap-tahap yang penulis lakukan adalah sebagai berikut :

1. Tahap Perencanaan

Pada tahap perencanaan, penulis melakukan beberapa kegiatan secara terurut yakni penulis memilih tema atau topik penelitian, melaksanakan studi pendahuluan, merumuskan masalah, membuat instrumen penelitian serta menyusun rencana penelitian.

2. Tahap Pelaksanaan Penelitian

Pada tahap ini, penulis mengimplementasikan metode pengembangan sistem model *waterfall*. Berdasarkan model *waterfall* pengembangan sistem dimulai dari tahap analisis kebutuhan baik kebutuhan *hardware* dan *software* ataupun kebutuhan sistem, kemudian dilanjutkan dengan desain sistem, implementasi atau *coding*, uji coba sampai pada tahap pemeliharaan setelah sistem berjalan sesuai dengan kebutuhan pengguna.

3. Tahap Penulisan

Pada tahap penulisan, penulis membuat sebuah laporan penelitian dengan bentuk dan sistematik skripsi berdasarkan pedoman skripsi jenis rekaya produk atau aplikasi produk.

B. Analisis Kebutuhan

1. Analisis Kebutuhan *Hardware* dan *Software*

Dalam membangun sebuah sistem diperlukan adanya perangkat keras dan perangkat lunak untuk menunjang saat pembuatan maupun saat mengoperasikan sistem yang sedang dikembangkan. Adapun untuk menunjang berjalannya sistem informasi administrasi dengan baik, penulis menyarankan adanya perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut :

a *Hardware*

- 1) *Processor* dengan kecepatan 2 Gigahertz (GHz) atau lebih tinggi
- 2) *Memory* atau RAM sebesar 1 Gigabyte atau lebih
- 3) *Harddisk* sebesar 80 GB atau lebih besar
- 4) *Graphics Memory* 512mb atau lebih tinggi
- 5) Monitor dengan resolusi 1024x768 pixel atau lebih tinggi
- 6) *Mouse*
- 7) *Keyboard*

b Perangkat lunak (*Software*)

1) Filezilla

Filezilla berguna untuk melakukan migrasi *file-file* sistem informasi ke dalam *server hosting* agar dapat diakses secara *online* nantinya.

2) *Browser*

Browser berguna sebagai media untuk mengakses sistem informasi secara *online*. Peneliti menyarankan untuk menggunakan browser mozilla firefox atau google chrome agar sistem dapat berjalan dengan baik.

2. Analisis Kebutuhan Sistem

Pada tahap ini, penulis mengumpulkan data mengenai kebutuhan dari sistem dengan metode tanya jawab dan kuisioner analisis kebutuhan terhadap narasumber. Adapun narasumber yang penulis pilih dalam menentukan kebutuhan sistem adalah administrator laboratorium yang mengetahui dengan baik proses administrasi yang terjadi di dalam laboratorium kimia.

Berdasarkan hasil tanya jawab serta kuisioner analisis kebutuhan sistem, penulis mengklasifikasikan *user* serta tugasnya di dalam sistem dan menu-menu di dalam sistem informasi administrasi laboratorium kimia.

Pada sistem informasi administrasi yang dikembangkan, terdapat tiga *user* yang akan menggunakan sistem ini. Adapun *user* tersebut adalah administrator, mahasiswa dan kepala laboratorium. Pada tabel 3.1 dijelaskan mengenai tanggung jawab dari tiap *user* beserta hak akses dari *user* di dalam sistem.

Tabel 3.1: Tabel analisis *user* Sistem Informasi Administrasi

<i>User</i>	Tanggung Jawab	Hak Akses
Administrastor	Input jadwal praktikum Input data alat dan bahan kimia Manajemen mahasiswa praktikum Manajemen <i>user</i>	Administrator
Mahasiswa	Mendaftar praktikum Membuat izin penelitian Membuat <i>form</i> peminjaman Mendaftar asisten laboratorium	Mahasiswa
Kepala Laboratorium	Monitoring kegiatan laboratorium Menerima laporan	Pejabat

Setelah mengklasifikasikan *user*, penulis mengklasifikasikan menu-menu yang ada di dalam sistem. Pada sistem informasi administrasi yang dikembangkan terdapat lima menu utama yang akan mengakomodasi kebutuhan dari *user*. Adapun menu-menu yang terdapat di dalam sistem adalah menu praktikum, penelitian, peminjaman, asisten dan menu *user*. Kelima menu tersebut memiliki fungsinya masing-masing berdasarkan pada *user* yang menggunakannya. Hal tersebut dapat dilihat pada tabel 3.2 mengenai menu dan fungsinya pada *user* yang menggunakannya.

Tabel 3.2: Tabel analisis menu Sistem Informasi Administrasi

Menu	Mahasiswa	Administrator
Praktikum	Melihat Jadwal Mendaftar praktikum Mencetak bukti daftar Melihat riwayat praktikum	Input Jadwal Manajemen mahasiswa
Penelitian	Membuat izin penelitian Mencetak <i>form</i> perizinan	Validasi perizinan
Peminjaman	Membuat <i>form</i> peminjaman	Input data alat dan bahan Manajemen peminjaman
Asisten	Mendaftar asisten	Manajemen pendaftaran
User	Mengisi data diri Mengubah data diri	Manajemen user

Pada tabel 3.2 penulis tidak memasukkan *user* kepala laboratorium ke dalam tabel, dikarenakan fungsi kepala laboratorium sebagai pengawas dan penerima laporan seperti yang telah dijelaskan pada tabel 3.1. Namun, pada *user* kepala laboratorium tetap memiliki lima menu tersebut untuk melihat setiap kegiatan yang terjadi pada sistem.

C. Desain Sistem (*Design*)

Pada tahap desain sistem, penulis menerjemahkan kebutuhan sistem yang telah didapatkan kedalam bentuk visual berupa *Use Case Diagram*, *Class Diagram*, *Activity Diagram*, *Entity Relationship Diagram* serta tampilan antarmuka sistem.

1. *Use Case Diagram*

Pada tahap desain sistem, penulis menggunakan *use case diagram* untuk menggambarkan seluruh pengguna atau aktor yang terdapat dalam sistem beserta peranan atau tugas dari aktor pada sistem informasi administrasi yang sedang dikembangkan oleh penulis.

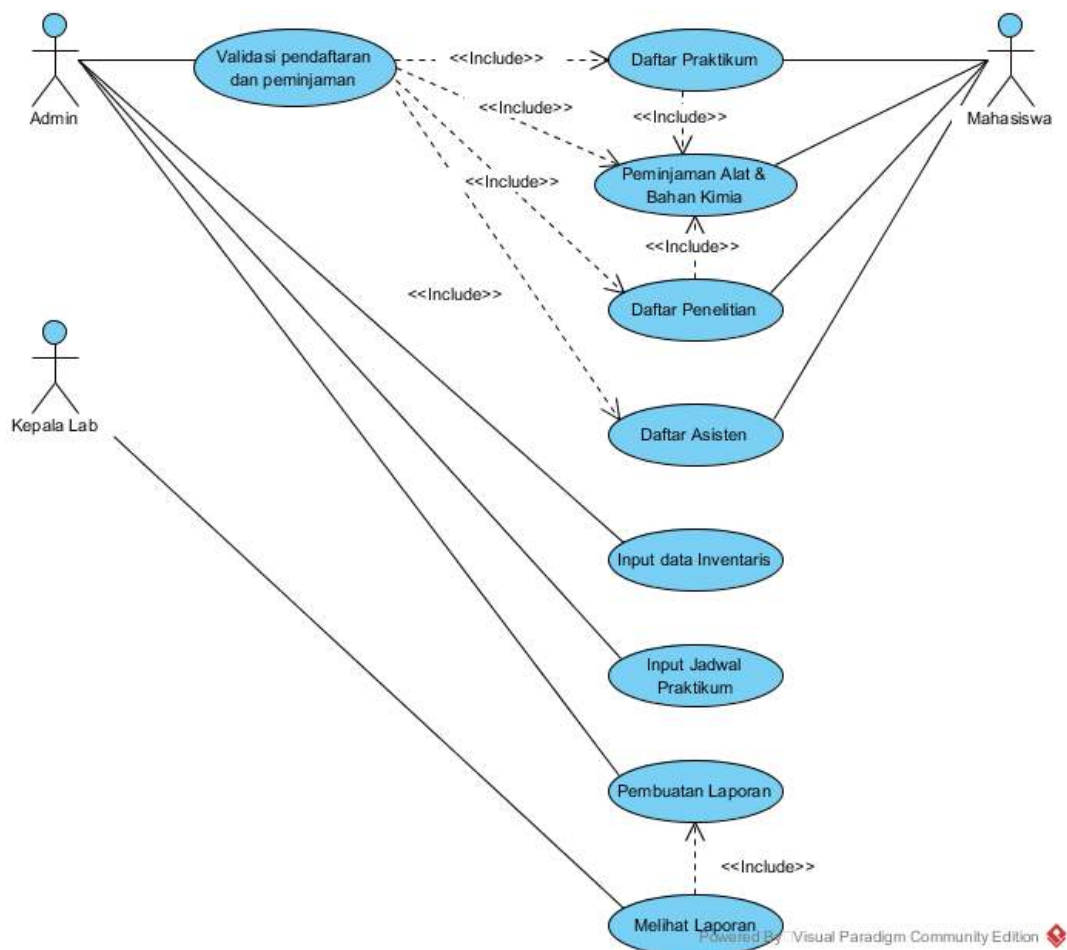
Pada sistem informasi administrasi laboratorium ini, terdapat tiga aktor penting yang berperan dalam mendukung berjalannya sistem. Adapun aktor-aktor pada sistem informasi administrasi ini adalah administrator, kepala laboratorium dan mahasiswa. Dalam sistem, aktor mahasiswa berperan sebagai penerima layanan atau *client*. Adapun tugas dari aktor mahasiswa adalah melakukan pendaftaran praktikum, membuat izin untuk melakukan penelitian, melakukan peminjaman alat atau bahan kimia serta dapat melakukan pendaftaran diri sebagai asisten laboratorium.

Aktor administrator bertugas untuk memproses terhadap setiap pengajuan permohonan dari aktor mahasiswa seperti pendaftaran praktikum, izin untuk melakukan penelitian, peminjaman alat atau bahan kimia serta manajemen terhadap pendaftaran asisten laboratorium. Administrator juga bertugas untuk membuat dan mengatur jadwal praktikum, melakukan *input* data inventaris laboratorium serta pembuatan laporan yang ditujukan kepada Kepala Laboratorium.

Selain aktor mahasiswa dan administrator yang telah dijelaskan tugas-tugasnya, terdapat juga aktor kepala laboratorium. Aktor kepala laboratorium berperan sebagai pengawas dan menerima laporan dari setiap kegiatan yang terjadi pada lingkungan

laboratorium serta memeriksa inventaris laboratorium.

Berikut adalah *Use Case Diagram* dari sistem informasi administrasi :



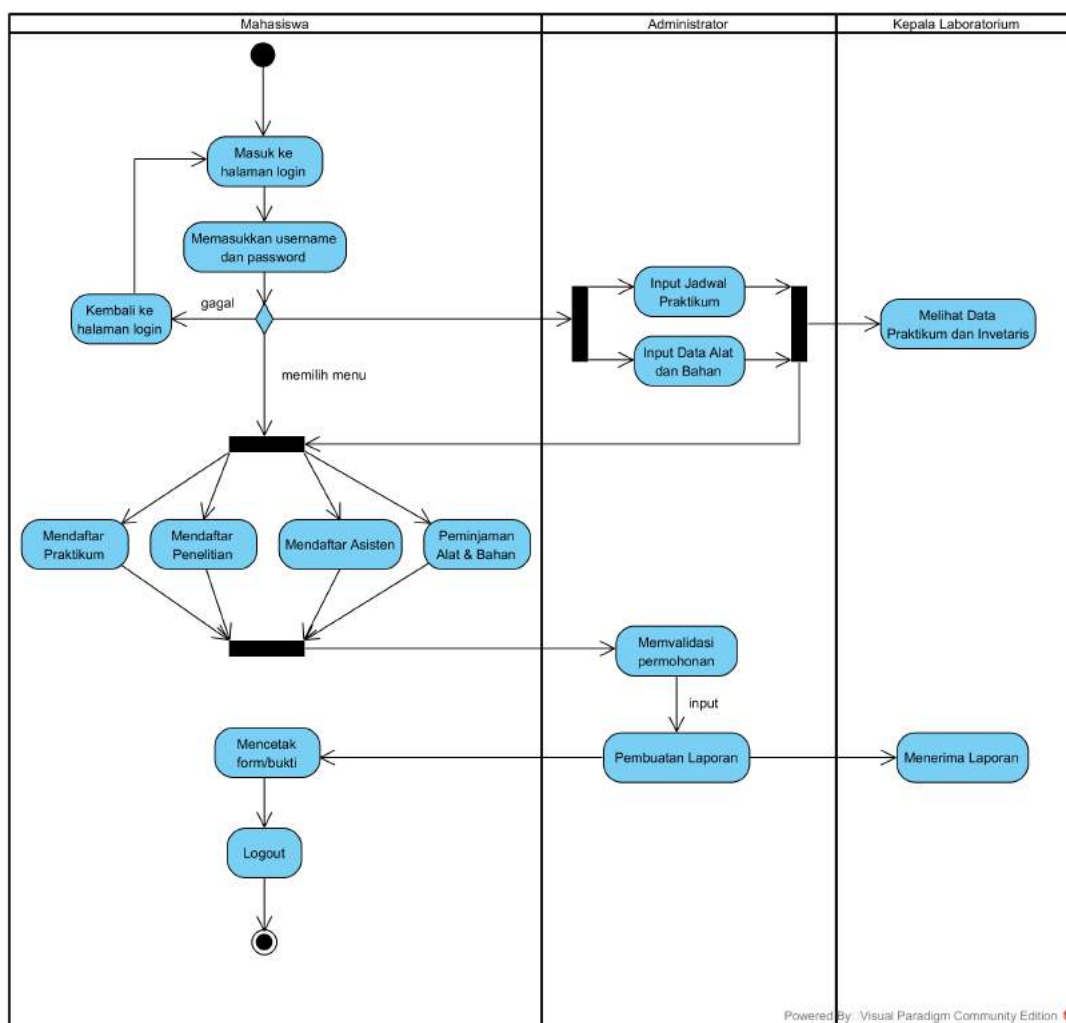
Gambar 3.1: Diagram *Use Case* Sistem Informasi Administrasi

2. Activity Diagram

Pada saat membuka sistem, seluruh *user* diharuskan *login* terlebih dahulu untuk dapat masuk kedalam sistem. *User* dalam hal ini mahasiswa diharuskan untuk mengisi data diri terlebih dahulu sebelum dapat menggunakan sistem secara keseluruhan. Setelah melengkapi data diri, mahasiswa dapat memilih menu yang terdapat dalam sistem baik untuk melakukan pendaftaran praktikum, membuat izin penelitian, peminjaman alat dan bahan kimia ataupun pendaftaran asisten laboratorium.

Setelah mahasiswa melakukan pengajuan atau permintaan di dalam sistem, administrator kemudian memproses pengajuan atau permintaan dari mahasiswa. Setelah permintaan terproses mahasiswa dapat mencetak dokumen sebagai bukti untuk melakukan kegiatan di laboratorium kimia. Sebelum mahasiswa dapat menggunakan sistem informasi administrasi, administrator sudah melakukan *input* terhadap jadwal praktikum serta data alat dan bahan kimia. Selain hal tersebut administrator juga membuat laporan di dalam sistem, berdasarkan pengajuan-pengajuan yang dilakukan oleh mahasiswa yang nantinya akan ditujukan kepada kepala laboratorium.

Berikut adalah *Activity Diagram* dari sistem informasi administrasi :



Gambar 3.2: Diagram *Activity* dari Sistem Informasi Administrasi

3. *Class Diagram*

Pada sistem informasi administrasi laboratorium kimia, penulis membuat sebuah *class diagram* untuk menggambarkan *class-class* yang mempunyai fungsi-fungsi tertentu dalam mendukung kinerja dari sistem.

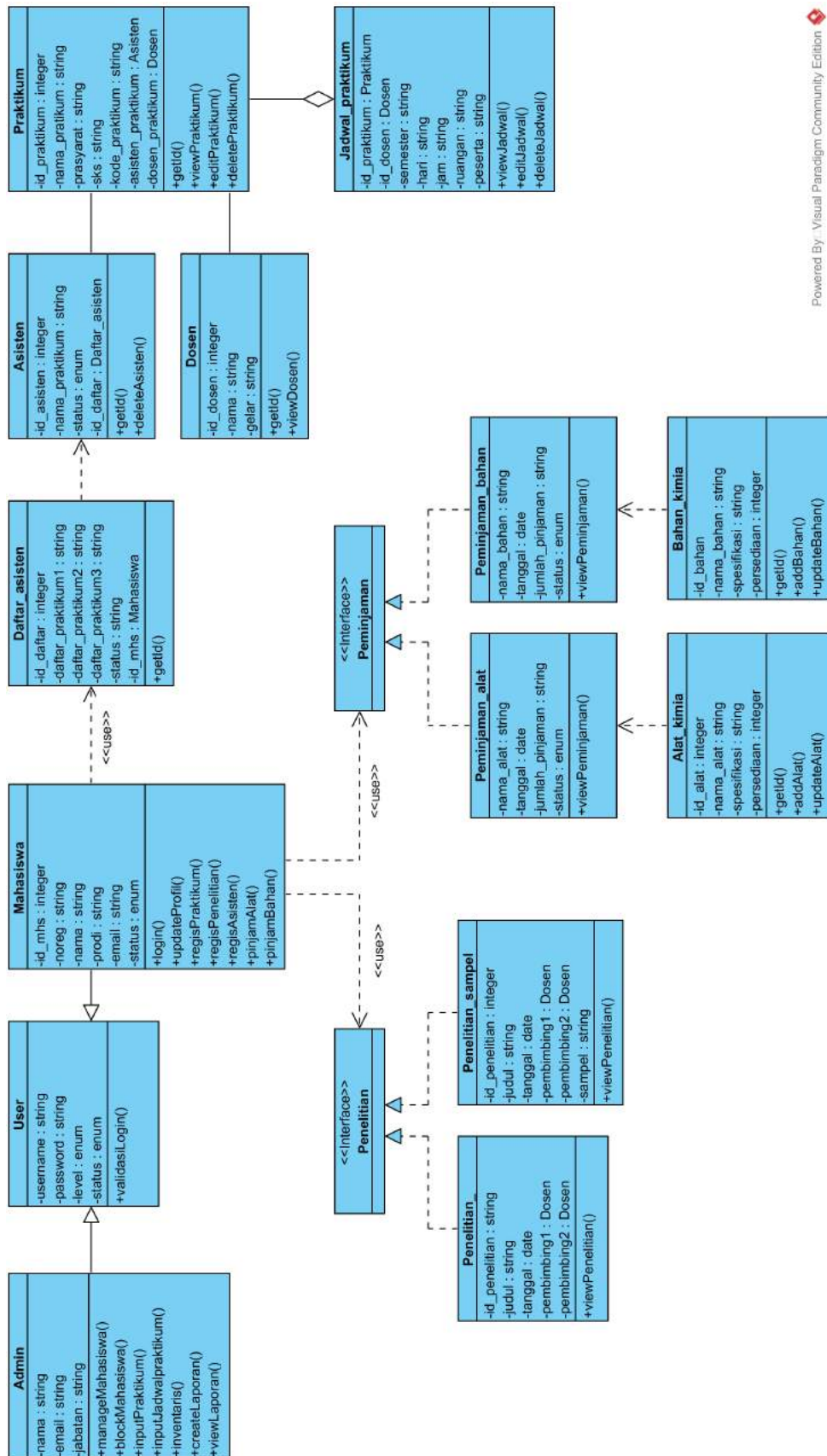
Dalam memanipulasi data pribadi dari *user*, penulis membuat 3 buah *class* yang mengatur penyimpanan, pengambilan dan penghapusan data dari setiap *user* yang ada pada sistem. Adapun ketiga *class* tersebut adalah *class* admin, *class* mahasiswa, dan *class* user yang menjadi *parent* dari kedua *class* yang telah disebutkan.

Pada *Class* penelitian terbagi menjadi dua yaitu *class* penelitian dan *class* penelitian sampel. *Class* penelitian berfungsi untuk mengatur fitur penelitian yang tidak menggunakan sampel khusus dalam penelitian yang dilakukan, sedangkan *class* penelitian sampel mengatur fitur penelitian yang menggunakan sampel khusus dalam penelitian tersebut.

Class peminjaman terbagi ke dalam empat *class* yaitu *class* peminjaman alat, *class* peminjaman bahan, *class* alat dan *class* bahan. *Class-class* tersebut mengatur data-data yang dibutuhkan dalam fitur peminjaman. *Class* peminjaman alat dan peminjaman bahan, mengatur data mengenai keterangan dari peminjaman seperti nama alat atau bahan, jumlah peminjaman dan tanggal peminjaman. *Class* alat dan bahan berguna untuk menyimpan data dari alat dan bahan kimia yang akan dimasukkan (*input*) kedalam sistem.

Pada bagian *class* praktikum, *class* ini membutuhkan *class* lain untuk menjalankan fungsinya. *Class* praktikum membutuhkan *class* asisten untuk mengambil data asisten yang akan menjadi salah satu parameter pada *class* praktikum. Kemudian *class* praktikum sendiri akan menjadi bagian untuk dari *class* jadwal praktikum untuk membuat jadwal praktikum yang akan ditampilkan di dalam sistem.

Berikut adalah gambar dari *class diagram* dari sistem informasi administrasi :



Gambar 3.3: Diagram Class dari Sistem Informasi Administrasi

4. *Entity Relationship Diagram*

Pada penelitian ini, penulis menggunakan *Entity Relationship Diagram* untuk menggambarkan bagaimana alur dari penyimpanan data pada sistem informasi administrasi yang dikembangkan. Dalam desain EDR dari sistem, terdapat 23 tabel yang akan menjadi media penyimpanan data dari sistem.

Tabel user, mahasiswa dan admin, berfungsi untuk menyimpan setiap data dari *user*. Ketiga tabel tersebut berelasi *one-to-one*, dengan tabel user sebagai tabel *parent* dari tabel mahasiswa dan admin. Tabel tahun merupakan tabel yang memberikan data tahun pada tabel mahasiswa.

Tabel praktikum, jadwal praktikum, nama praktikum dan riwayat praktikum adalah tabel-tabel yang berfungsi dalam menyimpan data-data yang berkaitan dengan praktikum. Tabel hari, semester, dan dosen merupakan tabel-tabel yang memberikan data kepada tabel praktikum dan tabel jadwal praktikum.

Tabel peminjaman alat, peminjaman bahan, daftar peminjaman alat, daftar peminjaman bahan, alat dan bahan merupakan tabel-tabel yang berfungsi untuk menyimpan data yang berkaitan dengan fitur peminjaman alat dan bahan serta untuk melakukan pendataan atau inventaris pada sistem.

Tabel penelitian, penelitian aas, penelitian saa dan penelitian lembur merupakan tabel-tabel yang berfungsi untuk menyimpan data-data yang dibutuhkan dalam fitur penelitian pada sistem.

Tabel mahasiswa merupakan tabel *parent* dari beberapa tabel yaitu tabel daftar asisten, peminjaman alat dan bahan, penelitian dan riwayat praktikum. Setiap tabel berelasi *one-to-one* dengan tabel mahasiswa karena mahasiswa hanya boleh melakukan satu kegiatan dalam satu fitur yang ada pada sistem.

Berikut adalah *Entity Relationship Diagram* dari sistem informasi administrasi :

5. Desain Antarmuka *User Interface*

Pada tahap ini, penulis membuat desain awal sistem untuk *user* mahasiswa. Pada awal penggunaan sistem, pada tampilan setiap *user* akan ditampilkan halaman *login* yang tidak berbeda.



Layanan Administrasi Laboratorium Kimia
Fakultas Matematika dan Ilmu Pengetahuan Alam

Username :

Password :

Login



Gambar 3.5: Halaman Login Sistem Informasi Administrasi

Pada *user* mahasiswa, sistem akan mengalihkan tampilan pada halaman data diri jika *user* mahasiswa tersebut belum mengisi data diri.



Layanan Administrasi Laboratorium Kimia
Fakultas Matematika dan Ilmu Pengetahuan Alam

Beranda | Praktikum | Penelitian | Peminjaman | Asisten | User | Logout

Data User

Foto

Unggah

Nama

Noreg

Jenjang

Prodi

Angkatan

No.Hp

Alamat

Simpan

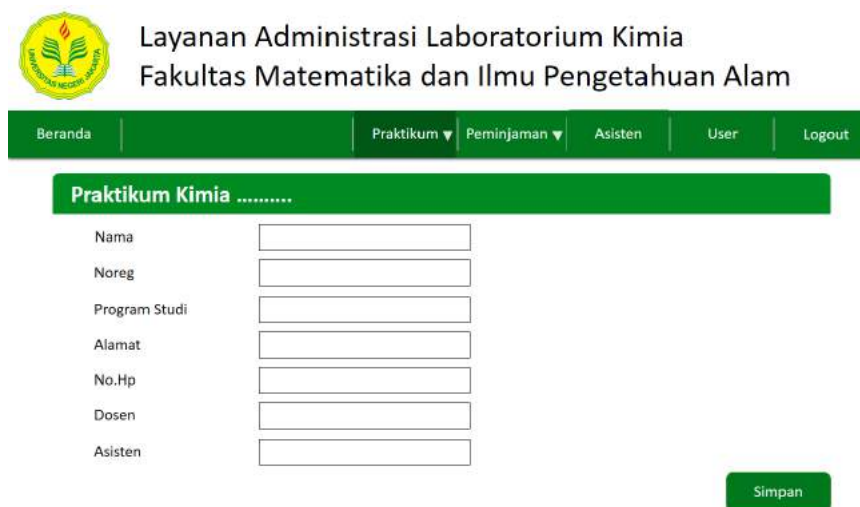
Gambar 3.6: Halaman Data User Sistem Informasi Administrasi

Setelah *user* mengisi data diri, sistem akan mengalihkan *user* pada halaman beranda yang menampilkan *user guide* atau panduan singkat dari sistem informasi administrasi.

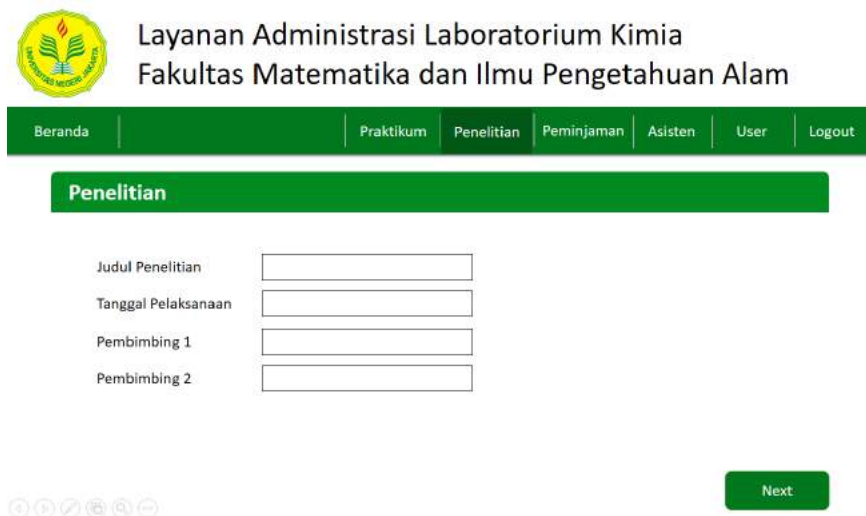


Gambar 3.7: Halaman Beranda Sistem Informasi Administrasi

Setelah melalui hal tersebut, *user* mahasiswa dapat menggunakan fitur-fitur yang tersedia pada sistem informasi administrasi mulai dari fitur praktikum, penelitian, peminjaman, asisten, *user* dan dapat keluar dari sistem pada menu navigasi *logout*.



Gambar 3.8: Halaman Praktikum Sistem Informasi Administrasi



Layanan Administrasi Laboratorium Kimia
Fakultas Matematika dan Ilmu Pengetahuan Alam

Beranda | Praktikum | Penelitian | Peminjaman | Asisten | User | Logout

Penelitian

Judul Penelitian

Tanggal Pelaksanaan

Pembimbing 1

Pembimbing 2

Next

Gambar 3.9: Halaman Penelitian Sistem Informasi Administrasi

Pada fitur peminjaman *user* mahasiswa dapat melakukan peminjaman terhadap alat dan bahan kimia. Adapun batasan jumlah alat ataupun bahan pada satu waktu peminjaman adalah 20 item *alat* kimia atau 20 *item* bahan kimia.



Layanan Administrasi Laboratorium Kimia
Fakultas Matematika dan Ilmu Pengetahuan Alam

Beranda | Praktikum | Penelitian | Peminjaman | Asisten | User | Logout

Peminjaman Bahan

Nama Bahan

+

Jumlah

Simpan

Gambar 3.10: Halaman Peminjaman Sistem Informasi Administrasi

Setelah membuat desain antarmuka dari sistem informasi administrasi, tahap selanjutnya yang penulis lakukan adalah tahap implementasi program.

D. Implementasi Program

Pada tahap implementasi program, penulis mengimplementasikan desain sistem yang telah dibuat ke dalam bentuk kode pemrograman yang dimengerti oleh komputer. Adapun tahapan dalam implementasi program adalah sebagai berikut :

1. Membangun Basis Data (*Database*)

Pada penelitian ini, penulis menggunakan Basis Data MySQL atau sekarang yang dikenal dengan nama MariaDB dalam membangun *database* dari sistem. *Database* dibuat berdasarkan desain ERD yang telah dibuat pada tahap desain. Penulis menggunakan *syntax SQL* yang dieksekusi dalam aplikasi phpMyAdmin untuk membuat tabel atau entitas serta relasi yang terdapat dalam *database* sistem informasi administrasi.

Berikut adalah sebagian dari *syntax SQL* yang membangun *database* dari sistem informasi administrasi :

```

20 create table Mahasiswa(
21     id_mahasiswa int(5) AUTO_INCREMENT,
22     noreg varchar(15) NOT NULL,
23     nama varchar(60) NOT NULL,
24     program_studi varchar(30) NOT NULL,
25     email varchar(50) NOT NULL,
26     alamat varchar(100) NOT NULL,
27     no_hp char(12) NOT NULL,
28     url_foto varchar(50) NOT NULL,
29     status enum('0','1','2') DEFAULT '0',
30     id_tahun int(3),
31     id_user int(5),
32     PRIMARY KEY(id_mahasiswa),
33     FOREIGN KEY(id_user) REFERENCES tb_user(id_user),
34     FOREIGN KEY(id_tahun) REFERENCES Tahun(id_tahun)
35 );
36
37 create table Tahun(
38     id_tahun int(3) AUTO_INCREMENT,
39     tahun_angkatan char(4) NOT NULL,
40     PRIMARY KEY(id_tahun)
41 );

```

Gambar 3.11: Sampel Syntax Query untuk Membangun Database Sistem

Setelah *syntax SQL* dieksekusi pada aplikasi phpMyAdmin, akan terbentuk sebuah *database* yang terdiri dari 23 tabel yang dapat dilihat pada gambar 3.12.

Table	Action	Rows	Type	Collation	Size	Overhead
admin	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 K1B	-
alat	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
asisten	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 K1B	-
bahan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K1B	-
daftar_asisten	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	32 K1B	-
daftar_peminjaman_alat	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	48 K1B	-
daftar_peminjaman_bahan	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	48 K1B	-
dosen	Browse Structure Search Insert Empty Drop	26	InnoDB	latin1_swedish_ci	16 K1B	-
hari	Browse Structure Search Insert Empty Drop	7	InnoDB	latin1_swedish_ci	16 K1B	-
jadwal_praktikum	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	32 K1B	-
mahasiswa	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	48 K1B	-
nama_praktikum	Browse Structure Search Insert Empty Drop	17	InnoDB	latin1_swedish_ci	16 K1B	-
peminjaman_alat	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	32 K1B	-
Console peminjaman_bahan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	32 K1B	-
penelitian	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 K1B	-
penelitian_aas	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 K1B	-
penelitian_Jembur	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 K1B	-
penelitian_saa	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 K1B	-
praktikum	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K1B	-
riwayat_praktikum	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	48 K1B	-
semester	Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 K1B	-
tahun	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	16 K1B	-
tb_user	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	32 K1B	-
23 tables	Sum	111	InnoDB	latin1_swedish_ci	672 K1B	0 B

Gambar 3.12: Database Sistem Informasi Administrasi Laboratorium

2. Implementasi Desain Antarmuka Sistem

Pada tahap ini, desain awal yang telah dibuat oleh penulis akan diimplementasikan dengan menggunakan *framework PHP codeigniter* serta *framework CSS bootstrap*. Setelah implementasi dilakukan, desain dari sistem informasi administrasi dapat dilihat melalui *browser*.

Pada sistem informasi administrasi, pertama kali *user* akan melihat halaman *login* dari sistem informasi administrasi. Setelah melakukan proses *login*, *user* akan masuk ke dalam sistem dan akan dialihkan oleh sistem pada halaman beranda dari sistem atau halaman pengisian data diri tergantung pada status dari *user*.



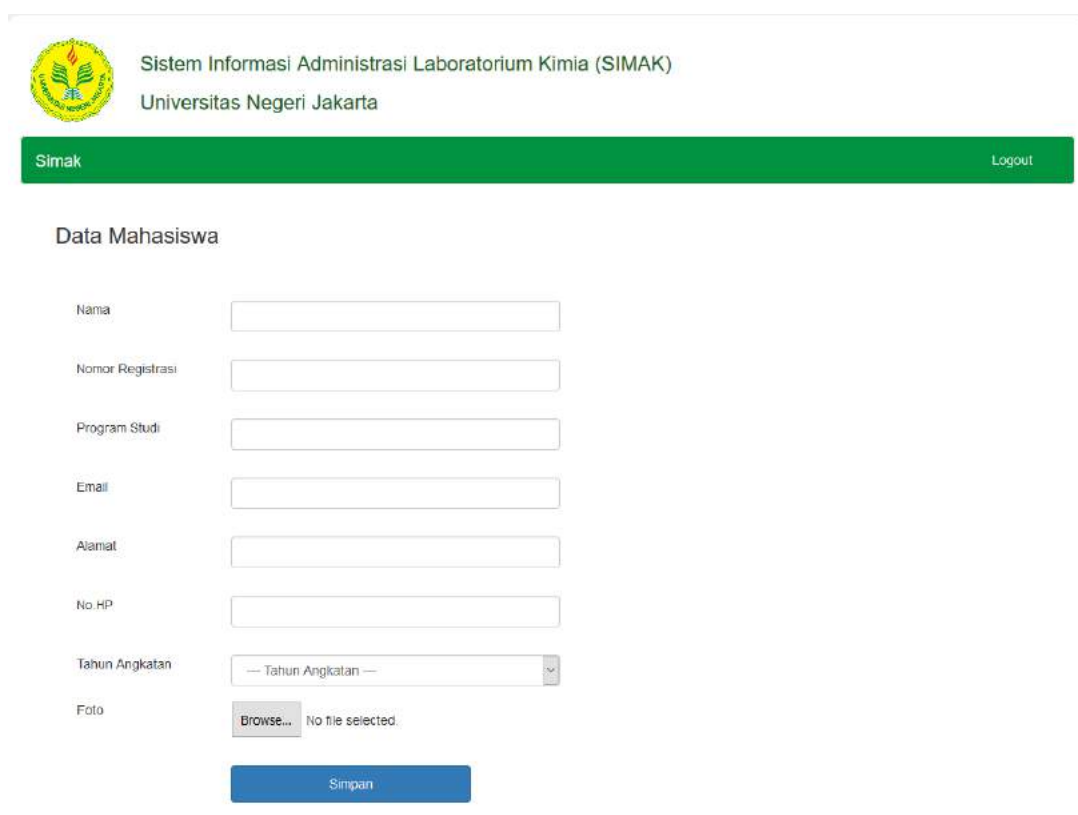

Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta


Username

Password

Gambar 3.13: Halaman Login dari Sistem Informasi Administrasi

Setelah *user* mahasiswa berhasil masuk ke dalam sistem, sistem akan menampilkan halaman data mahasiswa jika mahasiswa belum mengisi data diri.




Sistem Informasi Administrasi Laboratorium Kimia (SIMAK)
Universitas Negeri Jakarta

Simak Logout

Data Mahasiswa

Nama

Nomor Registrasi

Program Studi

Email

Alamat

No.HP

Tahun Angkatan

Foto No file selected.

Gambar 3.14: Halaman Data Diri dari Sistem Informasi Administrasi

Setelah mengisi data diri, sistem akan mengalihkan tampilan pada halaman utama dari sistem yang akan menampilkan panduan singkat dari sistem. Selanjutnya *user* mahasiswa dapat menggunakan fitur-fitur yang tersedia di dalam sistem.



Gambar 3.15: Halaman Awal dari Sistem Informasi Administrasi

Pada fitur praktikum, *user* mahasiswa dapat melihat jadwal praktikum yang ada dan dapat melakukan pendaftaran praktikum pada submenu pendaftaran praktikum di dalam menu praktikum.



Gambar 3.16: Halaman Jadwal Praktikum dari Sistem Informasi Administrasi

Pada gambar 3.17 ditampilkan halaman praktikum yang berfungsi untuk mendaftarkan matakuliah praktikum dengan memasukkan kode praktikum yang terdapat pada jadwal praktikum.

Sistem Informasi Administrasi Laboratorium Kimia (SIMAK)
Universitas Negeri Jakarta

Simak Praktikum - Penelitian - Peminjaman - Asisten User Logout

Masukkan kode praktikum

Simpan

Daftar Praktikum

Cetak

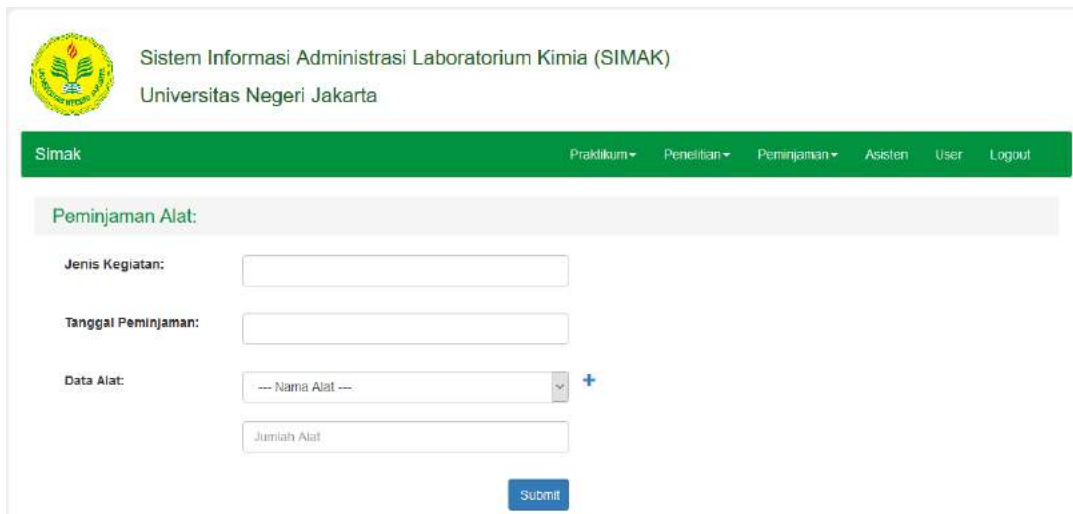
No	Kode	Nama Praktikum	Dosen
1	1001	Kimia Dasar I	Dr. A. Ridwan, M.Si
2	1002	Kimia Dasar II	Dr. Maria Paristiwati, M.Si
3	1003	Kimia Analisa Kualitatif Dan Kuantitatif	Dr. Afrizal, M.Si
4	1004	Kimia Pemisahan	Dr. Agung Purwanto, M.Si

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 3.17: Halaman Pendaftaran Praktikum dari Sistem Informasi Administrasi

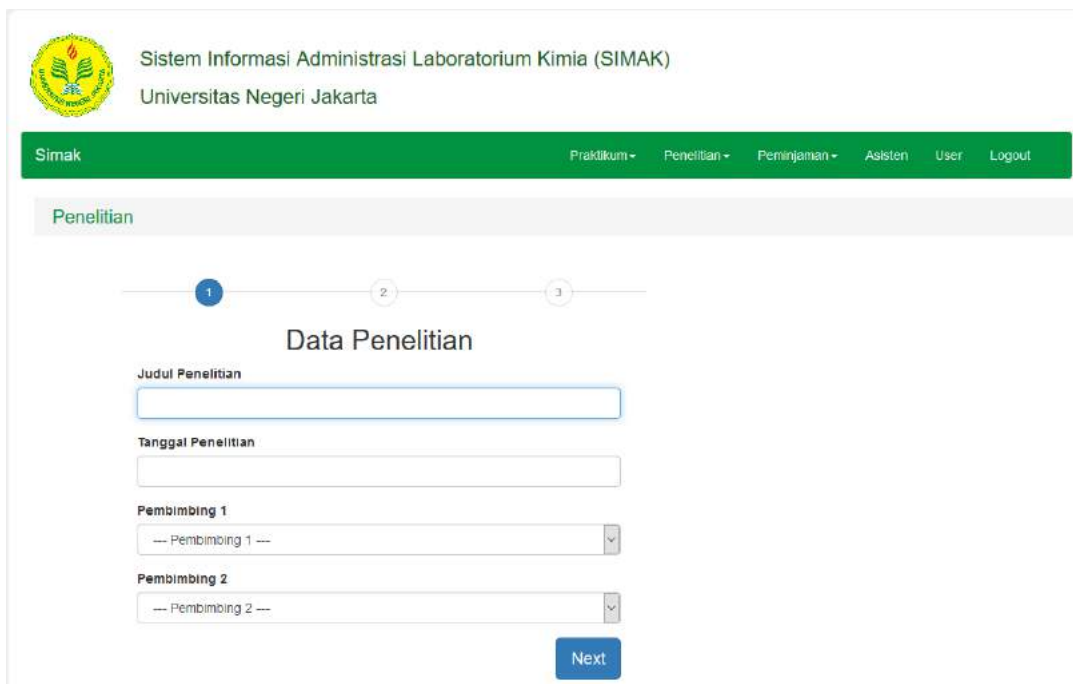
Selain fitur praktikum, *user* dapat melakukan peminjaman alat dan bahan kimia dengan menggunakan fitur peminjaman pada sistem. *User* akan dihadapkan pada *form input* mengenai data peminjaman yaitu jenis kegiatan, tanggal peminjaman, dan data dari alat atau bahan yang akan dipinjam.

Pada fitur peminjaman (gambar 3.18), masing-masing alat dan bahan yang dapat dipinjam oleh *user* adalah sebanyak 20 *item*. Setelah mengisi data yang dibutuhkan, *user* akan mendapatkan *file* yang kemudian akan diserahkan kepada administrator laboratorium.



Gambar 3.18: Halaman Peminjaman dari Sistem Informasi Administrasi

Pada fitur penelitian, *user* akan diminta untuk mengisi *form* berkaitan dengan data penelitian yang akan dilaksanakan, mulai dari judul penelitian, tanggal penelitian serta pembimbing dari penelitian. Setelah mengisi data-data penelitian *user* akan mendapatkan *file* dengan format PDF sebagai surat izin untuk melakukan penelitian.



Gambar 3.19: Halaman Penelitian dari Sistem Informasi Administrasi

Pada fitur asisten (gambar 3.20), *user* akan dihadapkan pada *form input* berkaitan dengan data pendaftaran asisten seperti pilihan praktikum yang diminati, nilai dari praktikum kimia organik dan jam kosong dari *user*.

The screenshot shows the 'Pendaftaran Asisten' (Assistant Registration) page. At the top left is the logo of Universitas Negeri Jakarta. The page title is 'Sistem Informasi Administrasi Laboratorium Kimia (SIMAK) Universitas Negeri Jakarta'. A green navigation bar contains the following links: Simak, Praktikum, Penilaian, Peminjaman, Asisten, User, and Logout. The main content area is titled 'Pendaftaran Asisten:' and contains the following form fields:

- Pilihan Praktikum 1:
- Pilihan Praktikum 2:
- Pilihan Praktikum 3:
- Nilai Praktikum Kimia Organik:
- Semester:
- Jam Kosong:

A blue 'Submit' button is positioned below the 'Jam Kosong' field.

Gambar 3.20: Halaman Pendaftaran Asisten dari Sistem Informasi Administrasi

3. Implementasi Sistem (Pemrograman)

Dalam mengimplementasikan seluruh fungsi dari sistem, penulis menggunakan bahasa pemrograman PHP dengan menggunakan *framework codeigniter* untuk mempercepat dalam proses *coding*. Proses *coding* atau pemrograman dibagi kedalam tiga bagian berdasarkan konsep MVC yang diterapkan dalam *framework codeigniter* yaitu bagian untuk mengatur tampilan (*view*), bagian untuk manipulasi *database* (*model*) serta bagian yang mengatur antara tampilan dan database (*controller*).

Pada sistem informasi administrasi terdapat empat fitur utama yang akan digunakan oleh *user* yaitu fitur pendaftaran praktikum, penelitian, peminjaman alat dan bahan serta pendaftaran asisten laboratorium. Selain fitur-fitur yang telah disebutkan, peneliti juga membuat fitur-fitur pendukung dalam sistem informasi administrasi seperti fitur login, manajemen *user* dan fitur *export pdf*. Dalam membuat sebuah fitur

yang utuh, pertama kali penulis membuat bagian *model* kemudian dilanjutkan dengan membuat bagian *controller* dan yang terakhir adalah membuat bagian *view* (tampilan).



Gambar 3.21: Proses Pembuatan Sebuah Fitur

Sebelum mengimplementasikan fitur-fitur utama dalam sistem, penulis pertama kali membuat fitur *login* pada sistem. Peneliti membuat tiga *user* yang berbeda yang memiliki tugasnya masing-masing. Adapun ketiga *user* tersebut adalah Administrator, Mahasiswa dan Kepala Laboratorium. Sesuai dengan penjelasan dalam membuat sebuah fitur, penulis pertama kali membuat *class model login* (gambar 3.23) yang berfungsi untuk memeriksa data pada *database* berkaitan dengan data *username* dan *password*. Kemudian penulis membuat *class controller* (gambar 3.24) yang akan memanggil *model login* yang telah dibuat untuk proses validasi *login* pada sistem yang kemudian *controller* juga akan memanggil *view* (gambar 3.22) yang dibutuhkan.

```

1  <?php
2  class M_login extends CI_Model {
3      function __construct()
4      {
5          parent::__construct();
6          $this->tbl = 'tb_user';
7      }
8
9      function cek_user($username, $password)
10     {
11         $query = $this->db->get_where($this->tbl, array('username' => $username, 'password' => $password));
12         return $query->result_array();
13     }
14 }
  
```

Gambar 3.22: Sampel Kode Model Login dari Sistem Informasi Administrasi

```

16 <body style="background-color: #fff">
17 <div class="container-fluid logins">
18   <div class="row">
19     <div class="col-md-12 login-header">
20       <div class="col-md-12 col-xs-2 text-center">
21         
22       </div>
23       <div class="col-md-12 col-xs-10 text-center titles">
24         <p>Sistem Informasi Administrasi Laboratorium Kimia (SIMAK)</p>
25         <p>Universitas Negeri Jakarta</p>
26       </div>
27     </div>
28   </div>
29   <div class="col-md-12 col-xs-12 login-body">
30     <form action="c_login/do_login" method="post">
31       <div class="form-group">
32         <div class="row colbox">
33           <div class="col-md-1 col-sm-2 col-md-offset-4">
34             <label for="username" class="control-label">Username</label>
35           </div>
36           <div class="col-md-3 col-sm-4">
37             <input class="form-control" id="username" name="username" placeholder="Noreg/Username" type="text" \
38           </div>
39         </div>
40     </form>

```

Gambar 3.23: Sampel Kode View Login dari Sistem Informasi Administrasi

```

24 function do_login()
25 {
26     $username = $this->input->post('username');
27     $password = $this->input->post('password');
28     $cek = $this->m_login->cek_user($username, md5($password));
29
30     if (count($cek) == 1) {
31         foreach ($cek as $cek) {
32             $username = $cek['username'];
33             $password = $cek['password'];
34             $level = $cek['level'];
35             $status = $cek['status'];
36             $id_user = $cek['id_user'];
37         }
38
39         $this->session->set_userdata(array(
40             'isLogin' => TRUE,
41             'username' => $username,
42             'password' => $password,
43             'level' => $level,
44             'status' => $status,
45             'id_user' => $id_user));
46
47         redirect('c_login/dashboard');
48     } else {
49         echo "<script>alert('Username atau Password salah. Login Gagal!')</script>";
50         redirect('c_login', 'refresh');
51     }
52 }
53 }

```

Gambar 3.24: Sampel Kode Controller Login dari Sistem Informasi Administrasi

Pada ke empat fitur utama dari sistem informasi administrasi baik fitur praktikum, penelitian, peminjaman dan pendaftaran asisten, proses pembuatan dari fitur-fitur tersebut tidak berbeda dengan proses pembuatan fitur *login*.

Adapun proses pembuatan ke empat fitur tersebut adalah sebagai berikut :

a. Fitur Praktikum

Pada fitur praktikum, adapun alur kerja dari sistem adalah pertama kali *user* akan ditampilkan jadwal dari praktikum yang ada dan dapat melakukan pendaftaran dengan memasukkan kode praktikum yang nantinya *user* dapat mencetak sebuah *file* sebagai bukti dari pendaftaran praktikum. Sedangkan pada halaman administrator, admin memiliki tugas untuk menambahkan matakuliah praktikum kemudian membuat jadwal praktikum serta dapat melihat daftar praktikum yang telah diambil oleh mahasiswa (*user*).

Pada proses implementasi program penulis pertama kali membuat bagian *view* yang akan menjadi media untuk proses *input* dari *user* ke dalam *database*. Selanjutnya penulis membuat bagian *model* yang akan mengatur penyimpanan data dari *user* ke dalam *database*. Terakhir penulis membuat *controller* yang akan menjadi jembatan antara *view* dan *model* yang telah dibuat.

```

4 class C_praktikum extends CI_Controller{
5
6     function __construct(){
7         parent::__construct();
8         $this->load->model('m_praktikum');
9         $this->load->model('m_mahasiswa');
10    }
11
12    //View
13    public function index(){
14        $data = array(
15            'title' => 'Praktikum',
16            'isi'    => 'mahasiswa/praktikum/v_praktikum',
17            'jadwal' => $this->m_praktikum->jadwal_praktikum()
18        );
19
20        $this->load->view('mahasiswa/layout/wrapper', $data);
21    }
22
23    public function daftar_praktikum(){

```

Gambar 3.25: Sampel Kode Controller Praktikum dari Sistem Informasi Administrasi

b. Fitur Penelitian

Pada fitur penelitian, sistem akan menampilkan halaman dengan *form* yang akan diisi oleh *user* untuk pendataan permohonan penelitian dan setelah pengisian *form* tersebut *user* akan mendapatkan *file* dengan format pdf sebagai surat pengantar yang ditujukan pada ketua laboratorium.

Pada proses penulisan program, penulis pertama kali membuat bagian *view* untuk menampilkan *form* penelitian, kemudian penulis membuat bagian *model* dan seperti sebelumnya penulis membuat *controller* yang menghubungkan keduanya. Kemudian penulis membuat sebuah *view* yang akan menghasilkan *file* berformat pdf yang akan diterima oleh *user*.

```

3 class M_penelitian extends CI_Model{
4
5 public function add($data, $table){
6     $this->db->insert($table, $data);
7     $query = $this->db->insert_id();
8     return $query;
9 }
10
11 public function penelitian(){
12     $query = $this->db->from('penelitian')
13         ->join('mahasiswa', 'mahasiswa.id_mahasiswa = penelitian.id_mahasiswa')
14         ->get();
15
16     return $query->result();
17 }
18
19 public function data_penelitian($id){
20     $query = $this->db->from('penelitian')
21         ->join('mahasiswa', 'mahasiswa.id_mahasiswa = penelitian.id_mahasiswa')
22         ->where(array('penelitian.id_penelitian' =>$id))
23         ->get();
24
25     return $query->row_array();
26 }

```

Gambar 3.26: Sampel Kode Model Penelitian dari Sistem Informasi Administrasi

c. Fitur Peminjaman

Pada fitur peminjaman, peminjaman dibagi menjadi 2 yaitu peminjaman alat dan peminjaman bahan kimia namun keduanya memiliki model, view dan controller yang menyerupai dan yang hanya membedakannya adalah data yang ditampilkan yaitu peminjaman alat menampilkan data alat dan peminjaman bahan kimia me-

nampilkan bahan kimia. Pada fitur ini, *user* juga akan mendapatkan *file* berformat pdf yang akan menjadi lampiran peminjaman alat dan bahan untuk kegiatan di dalam laboratorium.

Seperti fitur-fitur lainnya, penulisan program dimulai dari pembuatan *view* yang akan menerima *input* dari *user* serta *view* lain yang menghasilkan *file* dengan format pdf. Kemudian dilanjutkan pembuatan *model* yang akan mengatur penyimpanan data dan mengambil data untuk ditampilkan. Selanjutnya dibuat *controller* yang akan menghubungkan keduanya.

```

1 <div class="container-fluid mains">
2   <h3>Peminjaman Alat:</h3>
3
4   <div class="container margs">
5     <form action="<?php echo base_url();?>mahasiswa/C_peminjaman/do_peminjaman_alat" enctype="multipart/form-d
6
7     <div class="form-group">
8       <div class="row colbox">
9         <div class="col-md-2 col-sm-2">
10          <label for="jenis_kegiatan" class="control-label">Jenis Kegiatan:</label>
11        </div>
12        <div class="col-md-4 col-sm-4">
13          <input class="form-control" name="jenis_kegiatan" type="date" required>
14        </div>
15      </div>
16    </div>
17
18    <div class="form-group">
19      <div class="row colbox">
20        <div class="col-md-2 col-sm-2">
21          <label for="tgl_peminjaman" class="control-label">Tanggal Peminjaman:</label>
22        </div>
23        <div class="col-md-4 col-sm-4">
24          <input id="datepicker" class="form-control" name="tgl_peminjaman" type="date" required>
25        </div>
26      </div>
27    </div>
28
29    <div class="field_wrapper">
30      <div class="form-group">
31        <div class="row colbox">
32          <div class="col-md-2 col-sm-2">
33            <label class="control-label">Data Alat:</label>

```

Gambar 3.27: Sampel Kode View Peminjaman dari Sistem Informasi Administrasi

d. Fitur Pendaftaran Asisten

Pada fitur pendaftaran asisten, sistem akan menampilkan *form* pendaftaran yang harus dilengkapi oleh *user*. Setelah mengisi *form* tersebut, data dari *user* pendaftar akan ditampilkan pada halaman asisten dari administrator yang selanjutnya *form* pendaftaran tersebut akan dicetak oleh administrator.

4. Integrasi Codeigniter dengan Library FPDF

Pada sistem informasi administrasi, penulis menggunakan salah satu *library* dari PHP yaitu *library* FPDF untuk menghasilkan *output* dari sistem berupa *file* PDF. Format *file* PDF digunakan untuk membuat *form* penilaian praktikum dan *form* pendaftaran asisten yang dapat dicetak pada fitur praktikum dan fitur asisten untuk *user* administrator. Sedangkan untuk *user* mahasiswa, format *file* PDF digunakan untuk mencetak *form* izin penelitian dan peminjaman alat ataupun bahan kimia.

Untuk dapat menggunakan *library* FPDF, penulis membuat sebuah *file library* yang berisi *class-class* dari FPDF yang dapat diambil dari *website* FPDF (<http://www.fpdf.org/>) yang kemudian penulis menempatkan *file* tersebut ke dalam *folder library* dari *codeigniter*. Setelah menempatkan *file* tersebut, penulis dapat memanggil fungsi-fungsi yang terdapat pada *library* FPDF untuk menghasilkan *file* dengan format PDF. Berikut adalah salah satu *file view* dalam sistem yang akan menghasilkan format *file pdf* :

```

1  <?php
2  date_default_timezone_set('Asia/Jakarta');
3  $this->fpdf->FPDF("P","cm", "A4");
4
5  // kita set marginnya dimulai dari kiri, atas, kanan. jika tidak diset, defaultnya 1 cm
6  $this->fpdf->SetMargins(3.5,2.5,2);
7  /* AliasNbPages() merupakan fungsi untuk menampilkan total halaman
8  di footer, nanti kita akan membuat page number dengan format : number page / total page
9  */
10 $this->fpdf->AliasNbPages();
11
12 // AddPage merupakan fungsi untuk membuat halaman baru
13 $this->fpdf->AddPage();
14
15 // Setting Font : String Family, String Style, Font size
16 $this->fpdf->SetFont('Times','',10);
17
18 /*Header*/
19 $this->fpdf->SetFont('Times', '', 10);
20 $this->fpdf->Cell(16.5, 1, $lampiran, 0, 0, 'L');
21 $this->fpdf->Ln();
22 $this->fpdf->SetFont('Times', 'bu', 12);
23 $this->fpdf->Cell(16.5, 1, $title, 0, 0, 'C');
24 $this->fpdf->Ln(1.5);
25 /*End of Header*/
26
27 /*Main*/
28 $this->fpdf->SetFont('Times', '', 10);
29 $this->fpdf->Cell(1, 0.5, 'No', 0, 0, 'C');
```

Gambar 3.28: Penggunaan Library FPDF pada Sistem Informasi Administrasi

Saat sistem informasi administrasi dijalankan, dan menjalankan salah satu *controller* yang menggunakan *file view* tersebut maka sistem akan bekerja untuk menghasilkan *file* dengan format pdf. Berikut adalah hasil dari penggunaan *library* *fpdf* yang menghasilkan *file* penilaian praktikum pada sistem informasi administrasi:

PRAKTIKUM KIMIA DASAR I											
NAMA	:	Abimanyu Briliansyah	DOSEN	:	Dr. A. Ridwan, M.Si						
NO REGISTRASI	:	3325136395	ASISTEN	:							
PROGRAM STUDI	:	Kimia									
ALAMAT	:	Jakarta									
NO.HP	:	08788885555									

NO	PENILAIAN	PRAKTIKUM									
		1	2	3	4	5	6	7	8	9	10
I	Aktivitas Harian (20%)										
II	Laporan (25%)										
III	Tes Tertulis/Lisan (15%)										
IV	Keterampilan (40%)										
Nilai (100%)											

Keterangan : Aspek yang dinilai : I. Aktivitas Harian a. Kesiapan Kerja b. Disiplin Kerja c. Kerjasama II. Laporan a. Sistematika Laporan b. Isi dan Analisis Hasil c. Penggunaan Bahasa	III. Tes Tertulis/Lisan : Pre Test IV. Keterampilan a. Kecakapan/Skill b. Ketelitian c. Hasil Kerja Nilai Praktikum = Total Nilai / Jumlah Praktikum Nilai Akhir Praktikum = 60% Nilai Praktikum + 40% Nilai Ujian Praktikum
--	--

Gambar 3.29: PDF Hasil dari Penggunaan Library FPDF

BAB IV

UJI COBA DAN HASIL UJI COBA

A. Uji Coba

Pada metode pengembangan sistem model *waterfall*, setelah melakukan tahap implementasi proses pengembangan dilanjutkan dengan tahap uji coba atau *testing*. Pada tahap uji coba penulis melakukan pengujian sistem terhadap 3 responden yang akan menggunakan pengguna sistem nantinya. Pengujian terhadap sistem dimaksudkan untuk mengetahui apakah sistem yang dikembangkan sudah sesuai dengan kebutuhan dari pengguna.

Seperti yang telah disebutkan, uji coba sistem dilakukan terhadap 3 responden yaitu administrator laboratorium kimia, mahasiswa program studi kimia dan Kepala Laboratorium Kimia. Setiap responden akan melakukan pengujian terhadap sistem berdasarkan peran dari setiap responden yaitu *user* administrator, *user* mahasiswa dan *user* kepala laboratorium.

Pada tahap uji coba ini, penulis menggunakan kuisioner *user acceptance test* dalam mengumpulkan informasi dari hasil pengujian komponen-komponen dari sistem informasi administrasi. Adapun komponen-komponen yang akan dilakukan pengujian terhadap sistem informasi administrasi berdasarkan tugas dari setiap *user* adalah sebagai berikut:

1. Administrator
 - a. Manajemen praktikum : tambah dan edit
 - b. Manajemen jadwal praktikum : tambah, edit dan hapus
 - c. Mencetak *form* penilaian praktikum

- d. Melihat rekap pengajuan penelitian
- e. Melihat rekap peminjaman alat dan bahan kimia
- f. Manajemen alat dan bahan kimia : tambah, edit dan hapus
- g. Mencetak *form* pendaftaran asisten
- h. Manajemen *user* : tambah, edit dan hapus

2. Mahasiswa

- a. Melihat jadwal praktikum
- b. Mendaftar praktikum
- c. Mencetak bukti pendaftaran praktikum
- d. Mengajukan permohonan penelitian
- e. Mencetak *form* permohonan penelitian
- f. Melakukan peminjaman alat kimia
- g. Melakukan peminjaman bahan kimia
- h. Mencetak *form* peminjaman alat dan bahan
- i. Melakukan pendaftaran asisten
- j. Merubah data diri

3. Kepala Laboratorium

- a. Melihat data praktikum
- b. Melihat rekap penelitian
- c. Melihat rekap peminjaman alat dan bahan kimia
- d. Melihat rekap alat dan bahan kimia
- e. Melihat data asisten

B. Hasil Uji Coba

1. Administrator

Pada bagian uji coba untuk administrator, penulis melakukan pengujian dengan administrator laboratorium kimia. Berikut adalah komponen-komponen yang di uji penulis bersama dengan administrator laboratorium kimia :

a. Praktikum

Pertama kali administrator melakukan pengujian terhadap fitur praktikum. Uji coba dilakukan pada penambahan dan *edit* praktikum, menambahkan, *edit* dan menghapus jadwal praktikum. Pada bagian ini, penulis juga menguji penambahan praktikum dengan kasus jika dimasukkan *input* dengan kode praktikum yang sudah ada dan sistem menolak dengan memberikan *alert* atau peringatan bahwa praktikum dengan kode tersebut telah dibuat. Hal tersebut bertujuan untuk melihat apakah sistem sudah menolak kasus seperti yang disebutkan.

b. Penelitian

Pada bagian penelitian, administrator menguji apakah sistem sudah menampilkan data rekap dari permohonan penelitian yang diajukan oleh *user*.

c. Peminjaman

Pada bagian peminjaman, administrator menguji sistem berkaitan dengan penambahan, *edit*, menghapus alat dan bahan serta menguji apakah sistem sudah menampilkan rekap dari peminjaman yang telah dilakukan oleh *user*.

d. Asisten

Administrator menguji 2 komponen yang terdapat pada fitur ini yaitu melihat rekap dari pendaftaran asisten dan mencetak *form* pendaftaran yang telah diajukan

e. Manajemen *User*

Pada bagian manajemen *user*, administrator menguji apakah sistem dapat melakukan manajemen user seperti menambahkan, *edit* dan menghapus data dari *user*.

Adapun hasil dari pengujian untuk setiap komponen yang telah penulis buat adalah sebagai berikut:

Tabel 4.1: Komponen Penilaian Uji Coba Sistem Administrator

No.	Fitur	Detail	Keterangan
1	Praktikum	Tambah Praktikum	[x] Sesuai
		Edit Praktikum	[x] Sesuai
		Membuat Jadwal Praktikum	[x] Sesuai
		Edit Jadwal Praktikum	[x] Sesuai
		Hapus Jadwal Praktikum	[x] Sesuai
2	Penelitian	Melihat rekap penelitian	[x] Sesuai
3	Peminjaman	Tambah alat dan bahan	[x] Sesuai
		Edit alat dan bahan	[x] Sesuai
		Hapus alat dan bahan	[x] Sesuai
		Melihat rekap peminjaman	[x] Sesuai
4	Asisten	Melihat rekap pendaftaran asisten	[x] Sesuai
		Mencetak form pendaftaran asisten	[x] Sesuai
5	User	Menambah data user	[x] Sesuai
		Menghapus data user	[x] Sesuai
		Merubah data user	[x] Sesuai

2. Mahasiswa

Pada bagian uji coba untuk mahasiswa, penulis melakukan pengujian dengan salah satu mahasiswa program studi kimia.

a. Praktikum

Pada bagian ini, mahasiswa menguji apakah sistem sudah menampilkan halaman jadwal praktikum. selanjutnya mahasiswa menguji apakah fitur pendaftaran praktikum berjalan baik serta menguji untuk mencetak bukti pendaftaran. Pada bagian pendaftaran praktikum penulis meminta mahasiswa untuk memasukkan kode praktikum yang sama dan kode yang tidak terdaftar pada jadwal praktikum untuk menguji apakah sistem sudah memberikan *alert* atau peringatan untuk kedua kasus tersebut.

b. Penelitian

Pada bagian penelitian, mahasiswa menguji apakah *form* permohonan penelitian sudah ditampilkan dan melakukan uji coba melakukan permohonan penelitian serta mencetak *file* permohonan praktikum.

c. Peminjaman

Pada bagian peminjaman, terdapat dua bagian yaitu peminjaman alat dan peminjaman bahan kimia. Pengujian oleh mahasiswa dilakukan untuk melihat apakah fitur peminjaman berjalan dengan baik. Mahasiswa melakukan peminjaman terhadap alat dan bahan serta mencetak bukti peminjaman yang telah dilakukan apakah sesuai dengan *input* peminjaman.

d. Asisten

Pada bagian ini mahasiswa menguji sistem dengan melakukan pendaftaran asisten sebanyak 2 kali. Pendaftaran pertama diharapkan sistem menampilkan *di-*

alog box yang menampilkan bahwa pendaftaran telah diterima dan memberitahukan untuk menemui administrator laboratorium sedangkan untuk yang kedua diharapkan menampilkan *alert* bahwa mahasiswa tersebut telah melakukan pendaftaran dan diharapkan untuk menghubungi administrator.

e. *User*

Pada bagian *user*, hanya terdapat sebuah fitur untuk merubah data dari mahasiswa. Penulis meminta mahasiswa menguji apakah sistem sudah menampilkan data dari user dengan benar dan dapat merubah data dari mahasiswa bersangkutan.

Adapun hasil dari pengujian mahasiswa untuk setiap komponen yang telah dipaparkan sebelumnya adalah sebagai berikut :

Tabel 4.2: Komponen Penilaian Uji Coba Sistem Mahasiswa

No.	Fitur	Detail	Keterangan
1	Praktikum	Melihat jadwal praktikum	[x] Sesuai
		Mendaftar praktikum	[x] Sesuai
		Mencetak bukti daftar praktikum	[x] Sesuai
2	Penelitian	Mengajukan permohonan penelitian	[x] Sesuai
		Mencetak form permohonan praktikum	[x] Sesuai
3	Peminjaman	Melakukan peminjaman alat kimia	[x] Sesuai
		Melakukan peminjaman bahan kimia	[x] Sesuai
		Mencetak form peminjaman	[x] Sesuai
4	Asisten	Melakukan pendaftaran praktikum	[x] Sesuai
5	User	Melihat data user	[x] Sesuai
		Merubah data user	[x] Sesuai

3. Kepala Laboratorium

Pada bagian uji coba untuk kepala laboratorium, penulis melakukan pengujian dengan kepala laboratorium kimia. Pada bagian ini pengujian dilakukan apakah sistem sudah menampilkan halaman setiap fitur dengan baik. Untuk *user* kepala laboratorium hanya menampilkan hasil rekap dari setiap fitur yang tersedia sehingga kepala laboratorium dapat mengetahui kegiatan yang terjadi pada lingkungan laboratorium.

Adapun hasil dari pengujian untuk setiap komponen yang telah dipaparkan sebelumnya adalah sebagai berikut :

No.	Fitur	Detail	Keterangan
1	Praktikum	Melihat rekap Praktikum	[x] Sesuai
2	Penelitian	Melihat rekap penelitian	[x] Sesuai
3	Peminjaman	Melihat rekap data dan bahan kimia	[x] Sesuai
		Melihat rekap peminjaman	[x] Sesuai
4	Asisten	Melihat data asisten	[x] Sesuai

Tabel 4.3: Komponen Penilaian Uji Coba Sistem Kepala Laboratorium

Pada setiap uji coba yang telah dilakukan dengan administrator laboratorium, mahasiswa dan kepala laboratorium, seluruh fitur yang terdapat dalam sistem informasi administrasi dapat berjalan baik dan tidak ditemukan permasalahan selama uji coba berlangsung.

Berdasarkan tabel 4.1, tabel 4.2 dan tabel 4.3 yang berkaitan dengan hasil percobaan dari setiap *user* yang akan menjadi pengguna dalam sistem, dapat ditarik kesimpulan bahwa sistem informasi administrasi pada Laboratorium Kimia Fakultas MIPA di Universitas Negeri Jakarta telah berjalan dengan baik.

C. Pemeliharaan (*Maintenance*)

Pada metode pengembangan perangkat lunak *System Develop Life Cycle model waterfall*, langkah selanjutnya setelah tahap uji coba adalah tahap pemeliharaan. Setelah sistem berjalan dan beroperasi, diperlukan adanya pemeliharaan agar sistem dapat berjalan dengan optimal. Adapun pada sistem informasi administrasi, diperlukan adanya pengawasan terhadap jalannya sistem serta teknis dari pemeliharaan berkaitan dengan media penyimpanan data atau *database*.

Setelah tahap *deployment* atau pemasangan sistem, diperlukan pengawasan terhadap sistem setidaknya dalam rentang waktu 3 bulan untuk memastikan sistem berjalan dengan baik dan memeriksa apakah terdapat *bug* atau kesalahan pada sistem informasi administrasi.

Pada lingkungan laboratorium kimia fakultas MIPA, kurang lebih melayani 300 mahasiswa dari program studi kimia dan pendidikan kimia. Diasumsikan setiap mahasiswa melakukan minimal melakukan 1 kali untuk layanan praktikum, 1 kali layanan penelitian, 3 sampai 5 kali untuk fitur peminjaman untuk setiap semesternya serta 1 kali layanan pendaftaran asisten untuk syarat kelulusan. Berdasarkan hal tersebut penulis menyarankan untuk melakukan pembersihan atau *backup* data pada *database* sehingga sistem tidak terbebani karena banyak data yang sudah masuk ke dalam sistem minimal 1 tahun sekali.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan dari hasil implementasi dan uji coba sistem informasi administrasi, dapat ditarik kesimpulan sebagai berikut:

1. Sistem informasi administrasi pada laboratorium kimia Fakultas MIPA Universitas Negeri Jakarta dirancang dan diimplementasikan merujuk pada *System Develop Life Cycle (SLDC)* model *waterfall* dengan tahapan analisis kebutuhan, desain sistem, implementasi sistem, uji coba dan pemeliharaan.
2. Berdasarkan hasil uji coba, sistem informasi administrasi dengan fitur pendaftaran praktikum, penelitian, asisten serta inventaris laboratorium berhasil diimplementasikan dan dapat menjadi alternatif dalam permasalahan administrasi yang terjadi pada laboratorium kimia.

B. Saran

Adapun saran dari penulis untuk penggunaan sistem informasi administrasi dan penelitian selanjutnya adalah :

1. Sistem informasi administrasi dapat berjalan dengan baik jika setiap pengguna menjalankan tugasnya dengan baik dan berlanjut.
2. Dibutuhkan pemeriksaan dan pembaharuan data (jika diperlukan) terhadap stok dari alat dan bahan kimia, sedikitnya satu kali dalam satu bulan untuk memastikan kesamaan data sesungguhnya dan data di dalam sistem dalam jangka tiga bulan pertama setelah tahap *deployment*.

3. Dalam waktu satu tahun sekali, dibutuhkan pembersihan data dalam database agar sistem informasi administrasi dapat berjalan secara optimal.
4. Memperbaiki sistem informasi administrasi dari aspek *user interface* dan *user experience* untuk lebih memudahkan *user* dalam menggunakan sistem informasi administrasi.
5. Menambahkan algoritma khusus untuk memproses pendaftaran asisten sehingga sistem dapat menempatkan satu mahasiswa pada satu praktikum berdasarkan tiga pilihan praktikum mahasiswa pada saat mengisi *form* pendaftaran asisten.

DAFTAR PUSTAKA

- [1] Deacon, J. (2017) “Model-View-Controller Architecture.” [Online] <http://jdl.co.uk/briefings/index.html#mvc>
- [2] Fowler, M., *UML distilled 3rd edition*. Yogyakarta: Andi, 2005.
- [3] Henderi,, *Analysis and Design System with Unified Modeling Language (UML)*. Tangerang: STMIK Raharja, 2007.
- [4] Jogiyanto,, *Analisis dan Desain Sistem Informasi: pendekatan terstruktur teori dan praktek aplikasi bisnis*. Yogyakarta: ANDI, 2005.
- [5] Marlinda, L., *Sistem Basis Data*. Yogyakarta: Andi, 2004.
- [6] Mulyanto, A., *Sistem Informasi Konsep dan Aplikasi*. Yogyakarta: Pustaka Pelajar, 2009.
- [7] Nanik Susanti, M. A., “Sistem Informasi Manajemen Laboratorium (SIMLAB) (Studi Kasus Laboratorium Progdi Sistem Informasi UMK),” *Majalah Ilmiah Informatika*, vol. 3, no. 1, 2012.
- [8] Ralph Stair, G. R., *Principles of Information Systems*. USA: Thomson Learning, 2006.
- [9] Ricky Eka Saputra, N. R., “Pengembangan Sistem Informasi Inventaris Laboratorium Jurusan Teknik Informatika FT-UNESA,” *Prosiding SNRT (Seminar Nasional Riset Terapan)*, 2016.
- [10] Silalahi, U., *Studi Tentang Ilmu Administrasi*. Bandung: Sinar Baru Aglesindo, 2003.

- [11] Sulistyorini, P., “Pemodelan Visual dengan Menggunakan UML dan Rational Rose,” *Jurnal Teknologi Informasi Dinamik*, vol. 14, no. 1.
- [12] Susanto, A., *Sistem Informasi Akuntansi*. Bandung: Lingga Jaya, 2013.
- [13] Tawari Trupti, A., “Comparative Study of Different Frameworks of PHP,” *International Journal of Research in Computer and Information Technology (IJR-CIT)*, vol. Special Issue 2, no. 1, 2016.
- [14] Turban, E., *Introduction to Information Technology, 3rd Edition*. USA: John Wiley and Sons, 2006.

LAMPIRAN A

Implementasi Desain (Administrator)

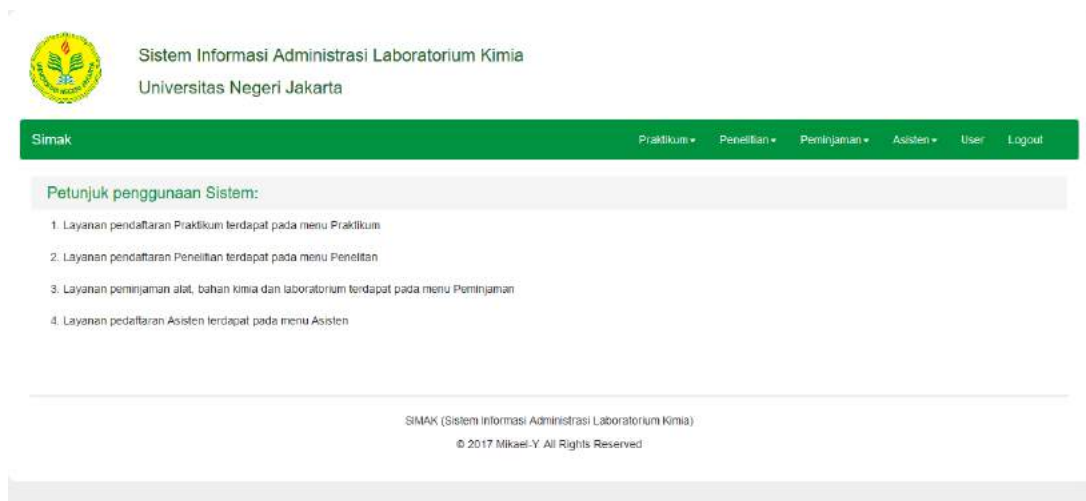


Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

Username

Password

Gambar 1.1: Halaman Login dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

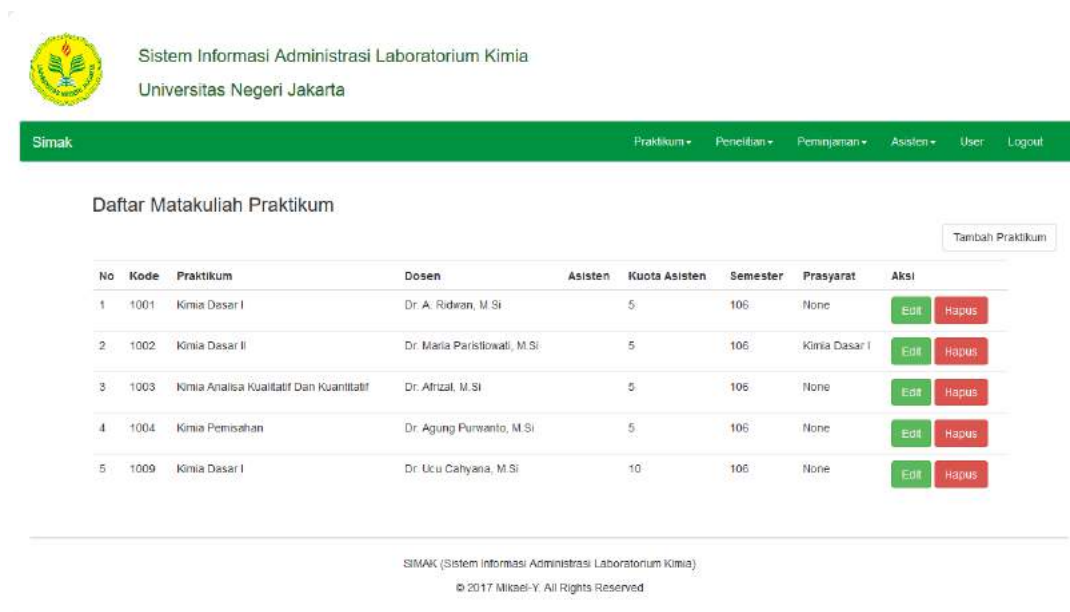
Simak Praktikum Penelitian Peminjaman Asisten User Logout

Petunjuk penggunaan Sistem:

1. Layanan pendaftaran Praktikum terdapat pada menu Praktikum
2. Layanan pendaftaran Penelitian terdapat pada menu Penelitian
3. Layanan peminjaman alat, bahan kimia dan laboratorium terdapat pada menu Peminjaman
4. Layanan pendaftaran Asisten terdapat pada menu Asisten

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 1.2: Halaman Awal dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

Simak Praktikum Penelitian Peminjaman Asisten User Logout

Daftar Matakuliah Praktikum

Tambah Praktikum

No	Kode	Praktikum	Dosen	Asisten	Kuota Asisten	Semester	Prasyarat	Aksi
1	1001	Kimia Dasar I	Dr. A. Ridwan, M.Si	5	105	None		Edit Hapus
2	1002	Kimia Dasar II	Dr. Maria Paristiwati, M.Si	5	105	Kimia Dasar I		Edit Hapus
3	1003	Kimia Analisa Kualitatif Dan Kuantitatif	Dr. Afrizal, M.Si	5	105	None		Edit Hapus
4	1004	Kimia Pemisahan	Dr. Agung Purwanto, M.Si	5	105	None		Edit Hapus
5	1009	Kimia Dasar I	Dr. Ucu Cahyana, M.Si	10	105	None		Edit Hapus

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 1.3: Halaman Tambah Praktikum dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

Simak Praktikum Penelitian Peminjaman Asisten User Logout

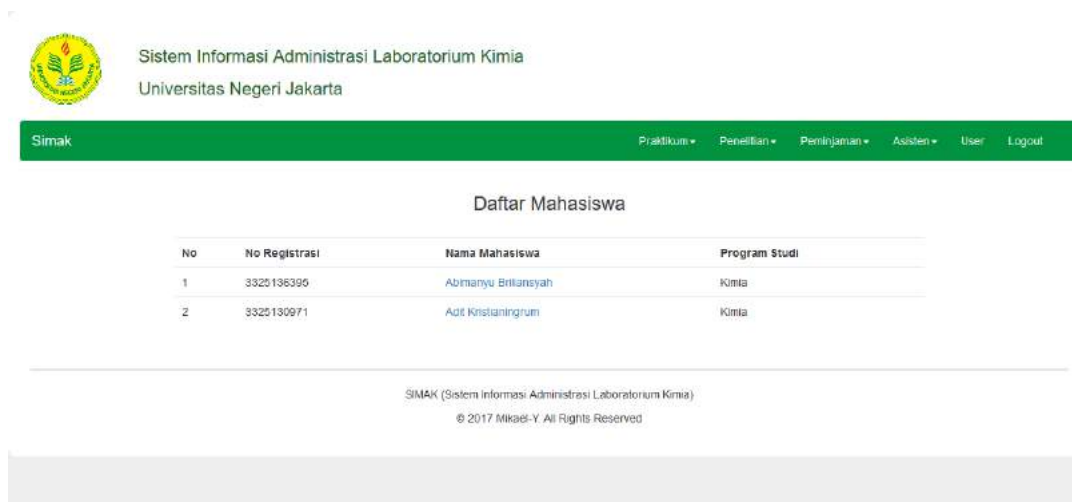
Daftar Jadwal Praktikum

Buat Jadwal

No	Kode	Hari/Jam	Praktikum	Dosen	Semester	Aksi
1	1001	Senin I (08:00-12:00)	Kimia Dasar I	Dr. A. Ridwan, M.Si	105	Edit Hapus
2	1002	Selasa II (13:00-16:00)	Kimia Dasar II	Dr. Maria Paristiwati, M.Si	105	Edit Hapus
3	1003	Kamis I (08:00-12:00)	Kimia Analisa Kualitatif Dan Kuantitatif	Dr. Afrizal, M.Si	105	Edit Hapus

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 1.4: Halaman Jadwal Praktikum dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

Simak Praktikum Peneltian Peminjaman Asisten User Logout

Daftar Mahasiswa

No	No Registrasi	Nama Mahasiswa	Program Studi
1	3325136395	Abimanyu Britiansyah	Kimia
2	3325130971	Adit Kristianingrum	Kimia

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 1.5: Halaman Peserta Praktikum dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta


Simak Praktikum Peneltian Peminjaman Asisten User Logout

Daftar Peminjaman Alat

No	No Registrasi	Mahasiswa	Jenis Kegiatan	Tanggal Peminjaman	Aksi
1	3325136395	Abimanyu Britiansyah	Peneltian	08/06/2017	check
2	3325130971	Adi Kristianingrum	Praktikum	08/05/2017	check
3	3325136395	Abimanyu Britiansyah	Praktikum 23	08/08/2017	check

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 1.6: Halaman Peminjaman dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

Simak Praktikum • Penelitian • Peminjaman • Asisten • User • Logout

Daftar Alat Kimia

[Tambah Alat](#)

No	Nama Alat	Spesifikasi	Jumlah	Aksi
1	Alat Desfilasi Mikro	Glastronic (KIT Praktikum)	2	Edit Hapus
2	Batu Didih	Pyrex Ass. Ø=5mm	048	Edit Hapus
3	Botol Akuatdes 500mL	RRC	28	Edit Hapus
4	Botol Kaca 05L	Lokal, Clear	1	Edit Hapus

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 1.7: Halaman Tambah Alat dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

Simak Praktikum • Penelitian • Peminjaman • Asisten • User • Logout

Pendaftaran Asisten

No	Nama Mahasiswa	Pilihan 1	Pilihan 2	Pilihan 3	Nilai Kimia Organik	Jam Kosong	Aksi
1	Abimanyu Brilliansyah	Kimia Dasar I	Kimia Dasar II	Kimia Dasar II	A+	Senin (06.00-12.00)	Cetak Form
2	Adit Kristianingrum	Kimia Pemisahan	Kimia Dasar II	Kimia Analisa Kualitatif Dan Kuantitatif	A	Jumat (full day)	Cetak Form

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y. All Rights Reserved

Gambar 1.8: Halaman Asisten dari Sistem Informasi Administrasi



Sistem Informasi Administrasi Laboratorium Kimia
Universitas Negeri Jakarta

Simak Praktikum Peneltian Peminjaman Asisten User Logout

Pendaftaran Asisten

No	Nama Mahasiswa	Pilihan 1	Pilihan 2	Pilihan 3	Nilai Kimia Organik	Jam Kosong	Aksi
1	Abimanyu Briliansyah	Kimia Dasar I	Kimia Dasar II	Kimia Dasar II	A-	Senin (08.00-12.00)	Cetak Form
2	Adil Kristeningrum	Kimia Pemisahan	Kimia Dasar II	Kimia Analisa Kualitatif Dan Kuantitatif	A	Jumat (full day)	Cetak Form

SIMAK (Sistem Informasi Administrasi Laboratorium Kimia)
© 2017 Mikael-Y All Rights Reserved

Gambar 1.9: Halaman Asisten dari Sistem Informasi Administrasi

LAMPIRAN B

Sampel Kode Pemrograman

```
4 class C_praktikum extends CI_Controller{
5
6 function __construct(){
7     parent::__construct();
8     $this->load->model('m_praktikum');
9     $this->load->model('m_mahasiswa');
10 }
11
12 //View
13 public function index(){
14     $data = array(
15         'title' => 'Praktikum',
16         'isi' => 'mahasiswa/praktikum/v_praktikum',
17         'jadwal' => $this->m_praktikum->jadwal_praktikum()
18     );
19
20     $this->load->view('mahasiswa/layout/wrapper', $data);
21 }
22
23 public function daftar_praktikum(){
```

Gambar 2.1: Sampel Kode Controller dari Fitur Praktikum

```
42 public function praktikum(){
43     $check = $this->db->get('praktikum');
44
45     //jika data lebih dari 0
46     if($check->num_rows() > 0){
47         foreach($check->result() as $data){
48             $hasil_praktikum[] = $data;
49         }
50         return $hasil_praktikum;
51     }
52 }
53
54 public function jadwal_praktikum(){
55     // $check = $this->db->get('jadwal_praktikum');
56     $hasil_jadwal = $this->db->select('*')
57         ->from('jadwal_praktikum')
58         ->join('praktikum', 'praktikum.id_praktikum = jadwal_praktikum.id_praktikum')
59         ->get();
60
61     if($hasil_jadwal->num_rows() > 0){
62         foreach ($hasil_jadwal->result() as $data) {
63             $hasil[] = $data;
64         }
65
66         return $hasil;
67     }
68 }
```

Gambar 2.2: Sampel Kode Model dari Fitur Praktikum

```

1 <div class="container-fluid" >
2   <div class="row">
3     <div class="col-md-12 col-xs-12" style="border: 1px solid #dfdfdf">
4       <center><h3>Jadwal Praktikum</h3></center><hr><br>
5       <?php
6         if(empty($jadwal)){
7           echo "<center><h4>Belum ada jadwal untuk semester ini!</h4></center>";
8         } else {
9           ?>
10          <div class="col-md-10 col-md-offset-1">
11            <table class="table table-striped">
12              <tr>
13                <th>No</th>
14                <th>Kode</th>
15                <th>Hari / Jam</th>
16                <th>Praktikum</th>
17                <th>Dosen</th>
18              <tr>

```

Gambar 2.3: Sampel Kode View dari Fitur Praktikum

```

4 class C_penelitian extends CI_Controller{
5
6   function __construct(){
7     parent::__construct();
8     $this->load->model('m_praktikum');
9     $this->load->model('m_peminjaman');
10    $this->load->model('m_mahasiswa');
11    $this->load->model('m_penelitian');
12  }
13
14  public function index(){
15    $data = array(
16      'title' => 'Penelitian',
17      'isi'   => 'mahasiswa/penelitian/v_penelitian_baru',
18      'dosen' => $this->m_praktikum->dosen()
19      //'alat' => $this->m_peminjaman->data_alat(),
20      //'bahan' => $this->m_peminjaman->data_bahan()
21    );
22
23    $this->load->view('mahasiswa/layout/wrapper', $data);
24  }
25
26  public function penelitian_aas(){
27    $data = array(
28      'title' => 'Penelitian',
29      'isi'   => 'mahasiswa/penelitian/v_penelitian_aas',
30      'dosen' => $this->m_praktikum->dosen()
31    );
32
33    $this->load->view('mahasiswa/layout/wrapper', $data);
34  }

```

Gambar 2.4: Sampel Kode Controller dari Fitur Penelitian

```

3 class M_penelitian extends CI_Model{
4
5 public function add($data, $table){
6     $this->db->insert($table, $data);
7     $query = $this->db->insert_id();
8     return $query;
9 }
10
11 public function penelitian(){
12     $query = $this->db->from('penelitian')
13         ->join('mahasiswa', 'mahasiswa.id_mahasiswa = penelitian.id_mahasiswa')
14         ->get();
15
16     return $query->result();
17 }
18
19 public function data_penelitian($id){
20     $query = $this->db->from('penelitian')
21         ->join('mahasiswa', 'mahasiswa.id_mahasiswa = penelitian.id_mahasiswa')
22         ->where(array('penelitian.id_penelitian' =>$id))
23         ->get();
24
25     return $query->row_array();
26 }

```

Gambar 2.5: Sampel Kode Model dari Fitur Penelitian

```

19 <div class="row setup-content" id="step-1">
20     <div class="col-xs-6 col-md-offset-1">
21         <div class="col-md-12">
22             <center><h2>Data Penelitian</h2></center>
23
24             <div class="form-group">
25                 <label class="control-label">Judul Penelitian</label>
26                 <input type="text" class="form-control" name="judul_penelitian" required/>
27             </div>
28
29             <div class="form-group">
30                 <label class="control-label">Tanggal Penelitian</label>
31                 <input id="datepicker" type="text" class="form-control" name="tgl_penelitian" required/>
32             </div>
33
34             <div class="form-group">
35                 <label class="control-label">Pembimbing 1</label>
36                 <select class="form-control" id="pembimbing1" name="pembimbing1" required>
37                     <option value="">--- Pembimbing 1 ---</option>
38                     <?php foreach($dosen->result() as $row) { ?>
39                         <?php echo "<option value='".$row->nama.'">".$row->nama;"</option>";
40                     } ?>
41                 </select>
42             </div>

```

Gambar 2.6: Sampel Kode View dari Fitur Penelitian


```

4 class C_peminjaman extends CI_Controller{
5
6 public function __construct(){
7     parent::__construct();
8     $this->load->model('m_peminjaman');
9     $this->load->model('m_mahasiswa');
10 }
11
12 public function peminjaman_alat(){
13     $data = array(
14         'title' => 'Peminjaman Alat',
15         'alat' => $this->m_peminjaman->data_alat(),
16         'isi' => 'mahasiswa/peminjaman/v_peminjaman_alat'
17     );
18
19     $this->load->view('mahasiswa/layout/wrapper', $data);
20 }
21
22 public function peminjaman_bahan(){
23     $data = array(
24         'title' => 'Peminjaman Bahan',
25         'bahan' => $this->m_peminjaman->data_bahan(),
26         'isi' => 'mahasiswa/peminjaman/v_peminjaman_bahan'
27     );
28
29     $this->load->view('mahasiswa/layout/wrapper', $data);
30 }

```

Gambar 2.7: Sampel Kode Controller dari Fitur Peminjaman

```

3 class M_peminjaman extends CI_Model{
4
5 public function mahasiswa($id_user){
6     $query = $this->db->get_where('mahasiswa', array('id_user' => $id_user));
7
8     return $query->row_array();
9 }
10
11 public function data_alat(){
12     $this->db->from('alat');
13     $this->db->order_by('nama_alat asc');
14     $query = $this->db->get();
15     return $query->result();
16 }
17
18 public function data_bahan(){
19     $this->db->from('bahan');
20     $this->db->order_by('nama_bahan asc');
21     $query = $this->db->get();
22     return $query->result();
23 }

```

Gambar 2.8: Sampel Kode Model dari Fitur Peminjaman

```

1 <div class="container-fluid mains">
2   <h3>Peminjaman Alat:</h3>
3
4   <div class="container margns">
5     <form action="<?php echo base_url();?>mahasiswa/C_peminjaman/do_peminjaman_alat" enctype="multipart/form-da
6
7     <div class="form-group">
8       <div class="row colbox">
9         <div class="col-md-2 col-sm-2">
10          <label for="jenis_kegiatan" class="control-label">Jenis Kegiatan:</label>
11        </div>
12        <div class="col-md-4 col-sm-4">
13          <input class="form-control" name="jenis_kegiatan" type="date" required>
14        </div>
15      </div>
16    </div>
17
18    <div class="form-group">
19      <div class="row colbox">
20        <div class="col-md-2 col-sm-2">
21          <label for="tgl_peminjaman" class="control-label">Tanggal Peminjaman:</label>
22        </div>
23        <div class="col-md-4 col-sm-4">
24          <input id="datepicker" class="form-control" name="tgl_peminjaman" type="date" required>
25        </div>
26      </div>
27    </div>
28
29    <div class="field_wrapper">
30      <div class="form-group">
31        <div class="row colbox">
32          <div class="col-md-2 col-sm-2">
33            <label class="control-label">Data Alat:</label>

```

Gambar 2.9: Sampel Kode View dari Fitur Peminjaman

```

4 class C_asisten extends CI_Controller{
5
6   function __construct(){
7     parent::__construct();
8     $this->load->model('m_asisten');
9     $this->load->model('m_praktikum');
10    $this->load->model('m_mahasiswa');
11  }
12
13  public function index(){
14    $data = array(
15      'title' => 'Daftar Asisten',
16      'isi'    => 'mahasiswa/v_asisten',
17      'praktikum' => $this->m_praktikum->praktikum2()
18    );
19
20    $this->load->view('mahasiswa/layout/wrapper', $data);
21  }
22
23  //untuk handle pendaftaran
24  public function do_daftar_asisten(){
25    $this->form_validation->set_rules('pilihan_praktikum1', 'pilihan_praktikum1', 'required');
26    $this->form_validation->set_rules('pilihan_praktikum2', 'pilihan_praktikum2', 'required');
27    $this->form_validation->set_rules('pilihan_praktikum3', 'pilihan_praktikum3', 'required');
28    $this->form_validation->set_rules('nilai_praktikum', 'nilai_praktikum', 'required');
29    $this->form_validation->set_rules('semester', 'semester', 'required');
30    $this->form_validation->set_rules('jam_kosong', 'jam_kosong', 'required');

```

Gambar 2.10: Sampel Kode Controller dari Fitur Asisten

```

3 class M_asisten extends CI_Model{
4
5 public function asisten_mahasiswa($id_mahasiswa){
6     $query = $this->db->join('tahun', 'tahun.id_tahun = mahasiswa.id_tahun')
7         ->where(array('id_mahasiswa' => $id_mahasiswa))
8         ->get_where('mahasiswa');
9
10    return $query->row_array();
11 }
12
13 public function check_pendaftaran($id_mahasiswa){
14     $query = $this->db->get_where('daftar_asisten', array('id_mahasiswa' => $id_mahasiswa));
15     return $query->result_array();
16 }
17
18 public function daftar_pendaftar(){
19     $check = $this->db->select('*')
20         ->from('daftar_asisten')
21         ->join('mahasiswa', 'mahasiswa.id_mahasiswa = daftar_asisten.id_mahasiswa')
22         ->get('');
23
24     //jika data lebih dari 0
25     if($check->num_rows() > 0){
26         foreach($check->result() as $data){
27             $hasil[] = $data;
28         }
29         return $hasil;
30     }
31 }

```

Gambar 2.11: Sampel Kode Model dari Fitur Asisten

```

1 <div class="container-fluid mains">
2     <h3>Pendaftaran Asisten:</h3>
3     <div class="container margns">
4         <div class="col-md-12 col-xs-12">
5             <form method="POST" action="<?php echo base_url();?>mahasiswa/c_asisten/do_daftar_asisten">
6
7             <div class="form-group">
8                 <div class="row colbox">
9                     <div class="col-md-2 col-sm-2">
10                        <label for="pilihan_praktikum1" class="control-label">Pilihan Praktikum 1:</label>
11                    </div>
12                    <div class="col-md-4 col-sm-4">
13                        <select class="form-control" id="pilihan_praktikum1" name="pilihan_praktikum1" required>
14                            <option value="">Pilihan Praktikum 1</option>
15                            <?php foreach($praktikum->result() as $row) { ?>
16                                <?php echo "<option value='".$row->nama_praktikum."'>".$row->nama_praktikum."</option>";
17                            } ?>
18                        </select>
19                    </div>
20                </div>
21            </div>

```

Gambar 2.12: Sampel Kode View dari Fitur Asisten

LAMPIRAN C

Analisis Kebutuhan Sistem



ANALISIS KEBUTUHAN
PENGEMBANGAN SISTEM INFORMASI ADMINISTRASI PADA
LABORATORIUM KIMIA FAKULTAS MATEMATIK DAN IPA
UNIVERSITAS NEGERI JAKARTA
Untuk : Administrator Laboratorium

Dengan hormat,

Saya Mikael Yurubeli (NRM : 3145136196) mahasiswa Ilmu Komputer. Topik skripsi yang saya kerjakan adalah pengembangan sistem informasi administrasi pada laboratorium kimia Fakultas MIPA Universitas Negeri Jakarta. Sehubungan dengan hal tersebut, saya mohon kesediaan Bapak/Ibu untuk membantu mengisi kuisisioner yang telah disiapkan dengan maksud untuk mengumpulkan data dalam rangka memenuhi kebutuhan sistem.

Terimakasih atas partisipasi yang Bapak/Ibu berikan.

1. Apa saja layanan administrasi yang diberikan laboratorium?

- Praktikum
- Penelitian
- Peminjaman alat dan bahan
- Pendaftaran Asisten Laboratorium

Lainnya :

2. Siapa saja yang berhak mendapatkan layanan tersebut?

- Mahasiswa
- Dosen

Lainnya :

3. Apa yang harus dilakukan sebelum diizinkan untuk menggunakan layanan praktikum?

.....
.....
.....

4. Apa yang harus dilakukan sebelum diizinkan untuk melakukan penelitian di laboratorium kimia?

.....
.....
.....
.....

5. Apa yang harus dilakukan sebelum dapat melakukan peminjaman alat dan bahan?

.....
.....
.....

6. Apa syarat bagi mahasiswa untuk dapat mendaftarkan diri sebagai asisten laboratorium ?

.....
.....
.....

7. Bagaimana alur dari pendaftaran asisten laboratorium?

.....
.....
.....
.....

8. Berkaitan dengan layanan di laboratorium kimia, apa peranan atau tugas dari :

- Mahasiswa :

.....
.....
.....
.....

- Dosen :

.....
.....
.....
.....

- Asisten Laboratorium :

.....
.....
.....
.....

- Administrator :

.....
.....
.....
.....

- Kepala Laboratorium :

.....
.....
.....
.....

LAMPIRAN D

User Acceptance Test Administrator



KUISIONER
INSTRUMEN *USER ACCEPTANCE TEST*
PENGEMBANGAN SISTEM INFORMASI ADMINISTRASI PADA
LABORATORIUM KIMIA FAKULTAS MATEMATIK DAN IPA
UNIVERSITAS NEGERI JAKARTA
untuk : Administrator Laboratorium

Dengan hormat,

Saya Mikael Yurubeli (NRM : 3145136196) mahasiswa Ilmu Komputer, bersama ini saya mohon kesediaan bapak/ibu untuk membantu mengisi kuisisioner yang telah disiapkan dengan maksud untuk mengetahui seberapa jauh sistem informasi administrasi dapat memenuhi kebutuhan dari pengguna sistem ini.

Keterangan pilihan jawaban :

TS : Tidak Setuju

S : Setuju

1. Kuisisioner *User Acceptance Test*

No.	Keterangan	TS	S
1	Fitur login dapat berjalan dengan baik		
2	Input praktikum dapat berjalan dengan baik		
3	Mengubah data praktikum		
4	Dapat membuat jadwal praktikum		
5	Dapat mengubah jadwal praktikum		
6	Dapat menghapus jadwal praktikum		
7	Dapat mencetak form penilaian praktikum		
8	Sistem mempermudah dalam melihat rekap penelitian		



9	Sistem mempermudah dalam melihat rekap peminjaman		
10	Input alat dan bahan berjalan dengan baik		
11	Dapat mencetak form pendaftaran asisten		
12	Manajemen user dapat berjalan dengan baik		

2. Saran dan Masukkan

.....

.....

.....

.....

.....

.....

LAMPIRAN E

User Acceptance Test Mahasiswa



KUISIONER
INSTRUMEN *USER ACCEPTANCE TEST*
PENGEMBANGAN SISTEM INFORMASI ADMINISTRASI PADA
LABORATORIUM KIMIA FAKULTAS MATEMATIK DAN IPA
UNIVERSITAS NEGERI JAKARTA
untuk : Mahasiswa

Dengan hormat,

Saya Mikael Yurubeli (NRM : 3145136196) mahasiswa Ilmu Komputer, bersama ini saya mohon kesediaan anda untuk membantu mengisi kuisisioner yang telah disiapkan dengan maksud untuk mengetahui seberapa jauh sistem informasi administrasi dapat memenuhi kebutuhan dari pengguna sistem ini.

Keterangan pilihan jawaban :

TS : Tidak Setuju

S : Setuju

1. *Kuisisioner User Acceptance Test*

No.	Keterangan	TS	S
1	Fitur login dapat berjalan dengan baik		
2	Dapat mengisi form data diri		
3	Sistem dapat menunjukkan jadwal praktikum		
4	Dapat melakukan pendaftaran praktikum		
5	Dapat mencetak form pendaftaran praktikum		
6	Dapat mengajukan form penelitian		
7	Dapat mencetak form permohonan penelitian		
8	Dapat melakukan peminjaman alat kimia		
9	Dapat melakukan peminjaman bahan kimia		



10	Dapat mencetak form peminjaman		
11	Dapat melakukan pendaftaran asisten		
12	Dapat merubah data diri		

2. Saran dan Masukkan

.....

.....

.....

.....

.....

LAMPIRAN F

User Acceptance Test Kepala Laboratorium



KUISIONER
INSTRUMEN *USER ACCEPTANCE TEST*
PENGEMBANGAN SISTEM INFORMASI ADMINISTRASI PADA
LABORATORIUM KIMIA FAKULTAS MATEMATIK DAN IPA
UNIVERSITAS NEGERI JAKARTA
untuk : Kepala Laboratorium

Dengan hormat,

Saya Mikael Yurubeli (NRM : 3145136196) mahasiswa Ilmu Komputer, bersama ini saya mohon kesediaan bapak/ibu untuk membantu mengisi kuisisioner yang telah disiapkan dengan maksud untuk mengetahui seberapa jauh sistem informasi administrasi dapat memenuhi kebutuhan dari pengguna sistem ini.

Keterangan pilihan jawaban :

TS : Tidak Setuju

S : Setuju

1. Kuisisioner *User Acceptance Test*

No.	Keterangan	TS	S
1	Fitur login dapat berjalan dengan baik		
2	Sistem mempermudah dalam melihat rekap penelitian		
3	Sistem mempermudah dalam melihat rekap peminjaman		
4	Sistem mempermudah dalam melihat rekap alat dan bahan		
5	Sistem mempermudah dalam melihat data asisten		



2. Saran dan Masukkan

.....

.....

.....

.....

.....

.....

DAFTAR RIWAYAT HIDUP



MIKAEL YURUBELI. Lahir di Jakarta, 7 Mei 1995. Anak kedua dari pasangan Bapak Marthen Ngongo Bulla dan Ibu Sumarmi. Saat ini beralamatkan di Jln.Hikmah no.64 rt 01/005 cilangkap, cipayung, Jakarta Timur.

No. Ponsel : 087875076738

Email : mikaelyuru@gmail.com

Riwayat Pendidikan : Penulis mengawali pendidikan di TK Pelita pada tahun 2000 - 2001, dan kemudian melanjutkan pendidikan di SDK Nusa Melati pada tahun 2001 - 2007. Setelah itu, penulis melanjutkan ke SMPN 09 Jakarta hingga tahun 2010. Kemudian kembali melanjutkan ke SMAN 105 Jakarta pada tahun 2010 - 2013. Di tahun yang sama penulis melanjutkan ke Universitas Negeri Jakarta (UNJ), Program Studi Ilmu Komputer, melalui jalur UMBPTN. Pada akhir semester genap tahun 2017 (Jumat, 11 Agustus 2017) penulis telah memperoleh gelar Sarjana Komputer (S.Kom), Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Jakarta.

Riwayat Organisasi : Selama di perkuliahan, penulis aktif di organisasi keilmiahan Program Studi Ilmu Komputer sebagai anggota merangkap Pengurus Harian tahun 2014-2015. Penulis juga berpartisipasi dalam kegiatan BINER (Be Innovative and Educated Researcher) yaitu kegiatan workshop dan seminar yang diadakan oleh DE-Fault, dimana penulis tergabung sebagai anggota merangkap humas.

Riwayat Penelitian : Selain mengikuti organisasi, peneliti juga aktif dalam kegiatan Penelitian Dosen FMIPA UNJ bersama Dra. Sri Sudaryati, M.Pd tahun 2015 dan 2016.