

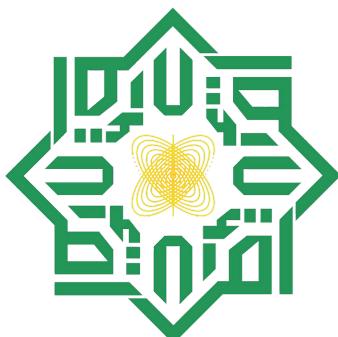
**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN  
LABORATORIUM MENGGUNAKAN METODE *AGILE*  
*DEVELOPMENT***

**TUGAS AKHIR**

Diajukan Sebagai Salah Satu Syarat  
untuk Memperoleh Gelar Sarjana Komputer pada  
Program Studi Sistem Informasi

Oleh:

**HAFIZ ARYAN SIREGAR**  
**12150310904**



**UIN SUSKA RIAU**

**FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU  
PEKANBARU  
2025**

**LEMBAR PERSETUJUAN**

**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN**

**LABORATORIUM MENGGUNAKAN METODE *AGILE***

***DEVELOPMENT***

**TUGAS AKHIR**

Oleh:

**HAFIZ ARYAN SIREGAR**  
**12150310904**

Telah diperiksa dan disetujui sebagai Laporan Tugas Akhir  
di Pekanbaru, pada tanggal 02 Mei 2025

**Ketua Program Studi**

**Pembimbing**

Eki Saputra, S.Kom., M.Kom.  
NIP. 198307162011011008

T. Khairil Ahsyar, S.Kom., M.Kom.  
NIP. 198505202023211020

**LEMBAR PENGESAHAN**  
**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN**  
**LABORATORIUM MENGGUNAKAN METODE AGILE**  
***DEVELOPMENT***

**TUGAS AKHIR**

Oleh:

**HAFIZ ARYAN SIREGAR**  
**12150310904**

Telah dipertahankan di depan sidang dewan penguji  
sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer  
Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau  
di Pekanbaru, pada tanggal 18 Maret 2025

Pekanbaru, 18 Maret 2025  
Mengesahkan,

Dekan

**Ketua Program Studi**

**Dr. Hartono, M.Pd.**

NIP. 196403011992031003

**Eki Saputra, S.Kom., M.Kom.**

NIP. 198307162011011008

**DEWAN PENGUJI:**

Ketua : Arif Marsal, Lc., MA. \_\_\_\_\_

Sekretaris : T. Khairil Ahsyar, S.Kom., M.Kom. \_\_\_\_\_

Anggota 1 : Eki Saputra, S.Kom., M.Kom. \_\_\_\_\_

Anggota 2 : Muhammad Jazman, S.Kom., M.Infosys. \_\_\_\_\_

## **LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL**

Tugas Akhir yang tidak diterbitkan ini terdaftar dan tersedia di Perpustakaan Universitas Islam Negeri Sultan Syarif Kasim Riau adalah terbuka untuk umum, dengan ketentuan bahwa hak cipta ada pada peneliti. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau ringkasan hanya dapat dilakukan atas izin peneliti dan harus dilakukan mengikuti kaedah dan kebiasaan ilmiah serta menyebutkan sumbernya.

Penggandaan atau penerbitan sebagian atau seluruh Tugas Akhir ini harus memperoleh izin tertulis dari Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau. Perpustakaan dapat meminjamkan Tugas Akhir ini untuk anggotanya dengan mengisi nama, tanda peminjaman, dan tanggal pinjam pada *form* peminjaman.

## **LEMBAR PERNYATAAN**

Dengan ini saya menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Per-guruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan di dalam daftar pustaka.

Pekanbaru, 18 Maret 2025  
Yang membuat pernyataan,

**HAFIZ ARYAN SIREGAR**  
**NIM. 12150310904**

## LEMBAR PERSEMPAHAN



*Assalamu'alaikum Warahmatullahi Wabarakatuh,*

Dengan penuh rasa syukur, peneliti ingin mengucapkan terima kasih kepada Allah *Subhanahu Wa Ta'ala* atas segala kemudahan yang telah diberikan selama proses penelitian ini. *Shalawat* serta salam senantiasa kita panjatkan kepada Nabi Muhammad *Shalallahu'Alaihi Wa Sallam* dengan melafalkan *Allahumma Sholli 'Ala Muhammad Wa 'Ala Ali Muhammad*.

Tugas Akhir ini peneliti dedikasikan kepada orang tua tercinta, Ayahanda Almarhum Ahmad Sofyan Siregar dan Ibunda Arni Shopiyah Nasution yang telah berjuang tanpa lelah demi mengantarkan peneliti mencapai pendidikan tinggi. Hanif Luthfi Siregar dan Nia Fanesa Siregar selaku adik kandung peneliti. Terima kasih yang tak terhingga atas do'a, dukungan, serta semangat yang selalu diberikan sehingga peneliti dapat menyelesaikan Tugas Akhir ini sesuai waktu yang direncanakan. Pencapaian ini tak terlepas dari didikan dan bimbingan yang mereka berikan sepanjang perjalanan hidup peneliti. Semoga Allah *Subhanahu Wa Ta'ala* membalsas segala kebaikan mereka dengan keberkahan yang melimpah.

Peneliti juga ingin menyampaikan banyak terima kasih kepada Seluruh Dosen Program Studi Sistem Informasi yang telah memberikan ilmu dan pengalaman berharga selama masa studi. Untuk sahabat terdekat saya terima kasih sudah menjadi teman baik saya, berkat kalian masa perkuliahan menjadi lebih bermakna. Semoga Allah *Subhanahu Wa Ta'ala* membalsas segala kebaikan mereka dengan keberkahan yang berlipat ganda.

*Wassalamu'alaikum Warahmatullahi Wabarakatuh.*

## KATA PENGANTAR

*Assalamu'alaikum Warahmatullahi Wabarakatuh.*

*Alhamdulillahi Rabbil'Alamin*, bersyukur kehadirat Allah *Subhanahu Wa Ta'ala* atas segala rahmat dan karunia-Nya sehingga peneliti dapat menyelesaikan Tugas Akhir ini. *Shalawat* beserta salam selalu tercurahkan untuk Nabi Muhammad *Shallallahu 'Alaihi Wa Sallam* dengan mengucapkan *Allahumma Sholli'ala Sayyidina Muhammad Wa'ala Ali Sayyidina Muhammad*. Tugas Akhir ini dibuat sebagai salah satu syarat untuk mendapatkan gelar Sarjana Komputer di Program Studi Sistem Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau. Pada penulisan Tugas Akhir ini, banyak pihak yang telah berperan dalam mendukung dan membantu peneliti. Maka dari itu, ungkapan terima kasih peneliti ucapan kepada:

1. Bapak Prof. Dr. Hairunas, M.Ag sebagai Rektor Universitas Islam Negeri Sultan Syarif Kasim Riau.
2. Bapak Dr. Hartono, M.Pd sebagai Dekan Fakultas Sains dan Teknologi.
3. Bapak Eki Saputra, S.Kom., M.Kom sebagai Ketua Program Studi Sistem Informasi dan Pengaji I peneliti.
4. Ibu Siti Monalisa, ST., M.Kom sebagai Sekretaris Program Studi Sistem Informasi.
5. Bapak T. Khairil Ahsyar, S.Kom., M.Kom sebagai Dosen Pembimbing Tugas Akhir ini yang telah banyak memberikan arahan, masukan, nasihat, serta motivasi dalam penyelesaian Tugas Akhir ini.
6. Bapak Arif Marsal, Lc., MA sebagai Ketua Sidang peneliti yang telah memberikan arahan dan masukan pada penelitian ini.
7. Bapak Muhammad Jazman, S.Kom., M.Infosys sebagai Pengaji II peneliti yang telah memberikan arahan dan masukan pada penelitian ini.
8. Ibu Mona Fronita, S.Kom., M.Kom sebagai Dosen Pembimbing Akademik yang telah memberikan dukungan, arahan, dan masukan kepada peneliti dari awal hingga akhir perkuliahan.
9. Seluruh Dosen Program Studi Sistem Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau yang telah memberikan ilmu dan bimbingan selama masa perkuliahan.
10. Kedua orang tua peneliti yaitu Ayah Almarhum Ahmad Sofyan Siregar dan Ibu Arni Shopiyah Nasution yang selalu memberikan *do'a*, dukungan, dan semangat kepada peneliti. Terima kasih atas segala keringat, jerih payah

pengorbanan, dan kerja keras yang telah diberikan kepada peneliti. Semoga Allah *Subhanahu Wa Ta'ala* membalas semua kebaikan Ayah dan Ibu dengan balasan yang lebih baik.

11. Adik-adik peneliti yaitu Hanif Luthfi Siregar dan Nia Fanesa Siregar yang selalu memberikan *do'a*, dukungan, dan semangat kepada peneliti.
12. Saudari Rahma Yulia Fani yang telah banyak memberikan motivasi, dukungan, dan semangat kepada peneliti dalam menyelesaikan Tugas Akhir ini.
13. Keluarga besar *Study Club Information System Networking Club Research* (ISNC Research). Terima kasih sudah memberikan dukungan untuk menyelesaikan Tugas Akhir ini.
14. Teman-teman Sistem Informasi Angkatan 2021 terkhusus teman seperjuangan di Kelas D yang selalu mendukung, berbagi informasi, dan membantu peneliti dalam menjalankan masa perkuliahan.
15. Teman-teman di Bangkit *Academy* dan *Coding Camp* yaitu Dani Harmade, Rahmat Afriyanto, Fajri Nurhadi, Hapid Ramdani, Erlienda Syahputra, Ahmeid Aqeil, dan Novrian Pratama yang telah menjadi bagian dari cerita perkuliahan ini.
16. Seluruh pihak yang tidak dapat disebutkan satu-persatu yang telah memberikan kontribusi secara langsung maupun tidak langsung dalam menyelesaikan Tugas Akhir ini.

Semoga segala *do'a* dan dorongan yang telah diberikan selama ini menjadi amal kebajikan dan mendapat balasan setimpal dari Allah *Subhanahu Wa Ta'ala*. Peneliti menyadari bahwa penulisan Tugas Akhir ini masih terdapat kekurangan dan jauh dari kata sempurna. Peneliti berharap untuk kritik dan saran yang membangun yang dapat disampaikan melalui *email* 12150310904@students.uin-suska.ac.id atau hafizaryansiregar@gmail.com untuk Tugas Akhir ini dan semoga Laporan Tugas Akhir ini bermanfaat bagi kita semua. Akhir kata peneliti ucapan terima kasih.

*Wassalamu'alaikum Warahmatullahi Wabarakatuh.*

Pekanbaru, 02 Mei 2025  
Peneliti,

**HAFIZ ARYAN SIREGAR**  
**NIM. 12150310904**

**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN  
LABORATORIUM MENGGUNAKAN METODE AGILE  
DEVELOPMENT**

**HAFIZ ARYAN SIREGAR  
NIM: 12150310904**

Tanggal Sidang: 18 Maret 2025  
Periode Wisuda:

Program Studi Sistem Informasi  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Sultan Syarif Kasim Riau  
Jl. Soebrantas, No. 155, Pekanbaru

**ABSTRAK**

Laboratorium memainkan peran krusial dalam institusi pendidikan dengan mendukung kegiatan praktikum dan penelitian. Pengelolaan yang efisien sangat penting untuk memastikan optimalisasi penggunaan sumber daya dan fasilitas. Penelitian ini bertujuan mengembangkan Sistem Informasi Inventaris Laboratorium (SITARIS) menjadi Integrated Laboratory Management Information System (ILMIS) pada Program Studi Sistem Informasi, UIN Sultan Syarif Kasim Riau. Penelitian ini menekankan penggunaan Metode Agile Development sebagai pendekatan utama dalam pengembangan sistem. Berbagai keterbatasan pada sistem SITARIS SI sebelumnya berhasil diidentifikasi dan diselesaikan, seperti kesalahan pengkodean, disfungsi fitur peminjaman, dan ketidaksesuaian format laporan. Sistem hasil pengembangan tidak hanya lebih stabil, tetapi juga dilengkapi dengan fitur tambahan seperti manajemen jadwal laboratorium. Penggabungan dengan sistem pendukung lainnya, seperti Laboratory Visitor Information System (LABVIS) dan Laboratory Assistant Registration Information System (LARIS) yang sebelumnya berjalan secara mandiri, kini memungkinkan akses sistem dilakukan secara terintegrasi. Hasil pengembangan menjadikan sebuah sistem yg terintegrasi, dan menambahkan fitur penjadwalan laboratorium. Penelitian ini menunjukkan bahwa ILMIS dapat meningkatkan efisiensi operasional, menyederhanakan pengelolaan inventaris, serta memperbaiki tata kelola laboratorium secara keseluruhan. Dengan demikian, penelitian ini memberikan kontribusi yang signifikan dalam mengoptimalkan tata kelola laboratorium di lingkungan Program Studi Sistem Informasi.

**Kata Kunci:** Agile Development, ILMIS, Integrasi, Manajemen Inventaris, Tata Kelola laboratorium

**DEVELOPMENT OF LABORATORY MANAGEMENT  
INFORMATION SYSTEM USING AGILE DEVELOPMENT  
METHOD**

**HAFIZ ARYAN SIREGAR  
NIM: 12150310904**

*Date of Final Exam: March 18<sup>th</sup> 2025  
Graduation Period:*

*Department of Information System  
Faculty of Science and Technology  
State Islamic University of Sultan Syarif Kasim Riau  
Soebrantas Street, No. 155, Pekanbaru*

**ABSTRACT**

*Laboratories play a crucial role in educational institutions by supporting practicum and research activities. Efficient management is essential to ensure the optimal use of resources and facilities. This study aims to develop the Laboratory Inventory Information System (SITARIS) into an Integrated Laboratory Management System in the Information Systems Study Program, UIN Sultan Syarif Kasim Riau. This study emphasizes the use of the Agile Development Method as the main approach in system development. Various obstacles in the previous SITARIS SI system have been resolved and resolved, such as coding errors, dysfunctional borrowing features, and inconsistencies in report formats. The developed system is not only more stable, but also equipped with additional features such as laboratory schedule management. Integration with other supporting systems, such as the Laboratory Visitor Information System (LABVIS) and the Laboratory Assistant Registration Information System (LARIS) which previously ran separately, makes access to the system individual. The development results create an integrated system, and add laboratory scheduling features. This study shows that an integrated laboratory management system can improve operational efficiency, organize inventory management, and improve overall laboratory governance. Thus, this study makes a significant contribution to optimizing laboratory governance in the Information Systems Study Program environment.*

**Keywords:** Agile Development, Integrated Laboratory Management System, Integration, Inventory Management, Laboratory Governance

## DAFTAR ISI

<b>LEMBAR PERSETUJUAN</b>	<b>ii</b>
<b>LEMBAR PENGESAHAN</b>	<b>iii</b>
<b>LEMBAR HAK ATAS KEKAYAAN INTELEKTUAL</b>	<b>iv</b>
<b>LEMBAR PERNYATAAN</b>	<b>v</b>
<b>LEMBAR PERSEMBAHAN</b>	<b>vi</b>
<b>KATA PENGANTAR</b>	<b>vii</b>
<b>ABSTRAK</b>	<b>ix</b>
<b>ABSTRACT</b>	<b>x</b>
<b>DAFTAR ISI</b>	<b>xi</b>
<b>DAFTAR GAMBAR</b>	<b>xv</b>
<b>DAFTAR TABEL</b>	<b>xviii</b>
<b>DAFTAR SINGKATAN</b>	<b>xix</b>
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	4
1.3 Batasan Masalah . . . . .	4
1.4 Tujuan . . . . .	5
1.5 Manfaat . . . . .	5
1.6 Sistematika Penulisan . . . . .	5
<b>2 LANDASAN TEORI</b>	<b>7</b>
2.1 Profil Instansi . . . . .	7
2.1.1 Sejarah Laboratorium Program Studi Sistem Informasi . . . . .	7
2.1.2 Visi . . . . .	8
2.1.3 Misi . . . . .	8
2.1.4 Struktur Organisasi . . . . .	8

2.2	Pengembangan Sistem Informasi . . . . .	8
2.3	Manajemen Laboratorium . . . . .	9
2.3.1	Laboratorium Rekayasa Sistem Informasi (RSI) . . . . .	10
2.3.2	Laboratorium Internet (INT) . . . . .	11
2.3.3	Laboratorium <i>Software Engineering</i> (SE) . . . . .	11
2.4	Lab SI Website . . . . .	11
2.5	<i>Laboratory Visitor System</i> (LABVIS) . . . . .	12
2.6	<i>Laboratory Assistant Registration Information System</i> (LARIS) . .	12
2.7	Sistem Informasi Inventaris (SITARIS) . . . . .	13
2.8	<i>Agile Development</i> . . . . .	15
2.9	<i>Unified Modelling Language</i> . . . . .	15
2.10	Observasi . . . . .	18
2.11	Wawancara . . . . .	18
2.12	PHP . . . . .	19
2.13	<i>Framework</i> . . . . .	19
2.14	CodeIgniter . . . . .	19
2.15	<i>Black Box Testing</i> . . . . .	20
2.16	Visual Studio Code . . . . .	20
2.17	Astah . . . . .	20
2.18	Balsamiq . . . . .	20
2.19	<i>Database</i> . . . . .	21
2.20	MariaDB . . . . .	21
2.21	XAMPP . . . . .	21
<b>3</b>	<b>METODOLOGI PENELITIAN</b> . . . . .	<b>22</b>
3.1	Tahap Perencanaan . . . . .	22
3.1.1	Identifikasi Masalah . . . . .	22
3.1.2	Studi Literatur . . . . .	23
3.1.3	Menentukan Tujuan dan Manfaat . . . . .	23
3.1.4	Menentukan Batasan Masalah . . . . .	23
3.2	Tahap Analisis dan Perancangan . . . . .	23
3.2.1	Menganalisis SITARIS . . . . .	24
3.2.2	<i>Agile Development</i> . . . . .	24
3.3	Tahap Implementasi dan Pengujian . . . . .	24
3.4	Tahap Dokumentasi . . . . .	25

<b>4 ANALISIS DAN PERANCANGAN</b>	<b>26</b>
4.1 Analisis . . . . .	26
4.1.1 Analisis Sistem Berjalan . . . . .	26
4.1.2 Analisis Sistem Usulan . . . . .	27
4.1.3 Analisis Kebutuhan Fungsional Sistem . . . . .	28
4.1.4 Analisis Kebutuhan Non-Fungsional Sistem . . . . .	28
4.2 Perancangan . . . . .	30
4.2.1 <i>Use Case Diagram</i> . . . . .	30
4.2.2 <i>Activity Diagram</i> . . . . .	32
4.2.3 <i>Class Diagram</i> . . . . .	35
4.2.4 Perancangan <i>Database</i> . . . . .	35
4.2.5 Perancangan Struktur Menu . . . . .	40
4.2.6 Perancangan <i>Interface</i> . . . . .	41
<b>5 IMPLEMENTASI DAN PENGUJIAN</b>	<b>46</b>
5.1 Implementasi Sistem . . . . .	46
5.1.1 Batasan Implementasi . . . . .	46
5.1.2 Implementasi Perangkat Keras ( <i>Hardware</i> ) . . . . .	46
5.1.3 Implementasi Perangkat Lunak ( <i>Software</i> ) . . . . .	47
5.1.4 Implementasi <i>Database</i> . . . . .	47
5.1.5 Implementasi <i>Routes</i> . . . . .	50
5.1.6 Implementasi <i>Model</i> . . . . .	53
5.1.7 Implementasi <i>View</i> . . . . .	53
5.1.8 Implementasi <i>Controller</i> . . . . .	53
5.1.9 Hasil Implementasi . . . . .	53
5.2 Pengujian Sistem . . . . .	57
<b>6 PENUTUP</b>	<b>60</b>
6.1 Kesimpulan . . . . .	60
6.2 Saran . . . . .	60

## DAFTAR PUSTAKA

<b>LAMPIRAN A HASIL WAWANCARA</b>	<b>A - 2</b>
<b>LAMPIRAN B HASIL OBSERVASI</b>	<b>B - 1</b>
<b>LAMPIRAN C DOKUMENTASI WAWANCARA</b>	<b>C - 1</b>



## **DAFTAR GAMBAR**

2.1	Struktur Organisasi Laboratorium . . . . .	9
2.2	<i>Laboratory Visitor Information System (LABVIS)</i> . . . . .	12
2.3	<i>Laboratory Assistant Registration Information System LARIS</i> . . . . .	13
2.4	Halaman <i>Login</i> . . . . .	14
2.5	Halaman Beranda . . . . .	14
2.6	Halaman Barang <i>Index</i> . . . . .	15
2.7	<i>Agile Development</i> . . . . .	16
3.1	Metodologi Penelitian . . . . .	22
4.1	<i>Use Case Diagram Admin</i> . . . . .	31
4.2	<i>Activity Diagram Login</i> . . . . .	33
5.1	Tampilan <i>Database</i> Tabel Dosen . . . . .	48
5.2	Tampilan <i>Database</i> Tabel Matkul . . . . .	49
5.3	Tampilan <i>Database</i> Tabel Jadwal . . . . .	49
5.4	Tampilan <i>Database</i> Tabel Ruangan . . . . .	50
5.5	Tampilan <i>Database</i> Tabel User . . . . .	50
5.6	Tampilan <i>Routes</i> Dosen . . . . .	51
5.7	Tampilan <i>Routes</i> Matkul . . . . .	51
5.8	Tampilan <i>Routes</i> Jadwal . . . . .	52
5.9	Tampilan <i>Routes</i> Ruangan . . . . .	52
5.10	Tampilan <i>Routes</i> User . . . . .	53
5.11	Tampilan Kode Barang ILMIS . . . . .	54
A.1	Transkip Wawancara Kaprodi . . . . .	A - 2
A.2	Transkip Wawancara Kalab . . . . .	A - 7
B.1	Laboratorium Rekayasa Sistem Informasi . . . . .	B - 1
B.2	Laboratorium Internet . . . . .	B - 1
B.3	Laboratorium Internet . . . . .	B - 2
B.4	Kegiatan Laboratorium . . . . .	B - 2
B.5	Website Laboratorium Program Studi Sistem Informasi . . . . .	B - 3
B.6	<i>Laboratory Visitor System (LABVIS)</i> . . . . .	B - 3
B.7	<i>Laboratory Assistant Registration Information System (LARIS)</i> . . . . .	B - 3
B.8	Disfungsi Fitur Peminjaman Barang . . . . .	B - 4

B.9	Disfungsi Fitur Peminjaman Ruangan . . . . .	B - 4
B.10	Disfungsi Pembuatan Kode Barang . . . . .	B - 4
C.1	Bukti Wawancara . . . . .	C - 1
D.1	Laboratorium Rekayasa Sistem Informasi . . . . .	D - 1
D.2	Laboratorium Internet . . . . .	D - 1
D.3	Laboratorium <i>Software Engineering</i> . . . . .	D - 2
D.4	Website Laboratorium Program Studi Sistem Informasi . . . . .	D - 2
D.5	<i>Laboratory Visitor System (LABVIS)</i> . . . . .	D - 2
D.6	<i>Laboratory Assistant Registration Information System (LARIS)</i> . . D - 3	
D.7	<i>Usecase Diagram</i> Kalab . . . . .	D - 3
D.8	<i>Usecase Diagram</i> Kaprodi . . . . .	D - 4
D.9	<i>Usecase Diagram</i> Sekprodi . . . . .	D - 4
D.10	<i>Usecase Diagram</i> Aslab . . . . .	D - 5
D.11	<i>Activity Diagram</i> Tambah Dosen . . . . .	D - 5
D.12	<i>Activity Diagram</i> Edit Dosen . . . . .	D - 6
D.13	<i>Activity Diagram</i> Hapus Dosen . . . . .	D - 6
D.14	<i>Activity Diagram</i> Tambah Mata Kuliah . . . . .	D - 7
D.15	<i>Activity Diagram</i> Edit Mata Kuliah . . . . .	D - 8
D.16	<i>Activity Diagram</i> Hapus Mata Kuliah . . . . .	D - 8
D.17	<i>Activity Diagram</i> Tambah Jadwal . . . . .	D - 9
D.18	<i>Activity Diagram</i> Edit Jadwal . . . . .	D - 10
D.19	<i>Activity Diagram</i> Hapus Jadwal . . . . .	D - 10
D.20	<i>Class Diagram</i> ILMIS . . . . .	D - 11
D.21	Struktur Menu ILMIS . . . . .	D - 12
D.22	Tampilan <i>Landing Page</i> ILMIS . . . . .	D - 13
D.23	Tampilan Pilihan <i>Login</i> untuk ILMIS . . . . .	D - 14
D.24	Tampilan Kelola Jadwal Laboratorium . . . . .	D - 14
D.25	Tampilan Tambah Jadwal Laboratorium . . . . .	D - 15
D.26	Tampilan Edit Jadwal Laboratorium . . . . .	D - 15
D.27	<i>Database</i> . . . . .	D - 16
D.28	Implementasi Model . . . . .	D - 17
D.29	Implementasi View . . . . .	D - 18
D.30	Implementasi Controller . . . . .	D - 19
D.31	Tampilan Pilihan Peminjaman . . . . .	D - 20
D.32	Tampilan Pengelolaan Peminjaman Barang . . . . .	D - 20

D.33 Tampilan Pengelolaan Peminjaman Ruangan . . . . .	D - 20
D.34 Tampilan Pilih Peminjam . . . . .	D - 21
D.35 Tampilan <i>Landing Page</i> ILMIS . . . . .	D - 22
D.36 Tampilan Pilihan Aplikasi untuk <i>Login</i> . . . . .	D - 23
D.37 Tampilan Kalender ILMIS . . . . .	D - 23
D.38 Tampilan Pengelolaan Jadwal Laboratorium . . . . .	D - 23
D.39 Tampilan Pengelolaan Jadwal Laboratorium RSI . . . . .	D - 24
D.40 Tampilan Pengelolaan Jadwal Laboratorium SE . . . . .	D - 24
D.41 Tampilan Pengelolaan Jadwal Laboratorium INT . . . . .	D - 24
D.42 Tampilan Tambah Jadwal Laboratorium . . . . .	D - 25
D.43 Tampilan Edit Jadwal Laboratorium . . . . .	D - 25
D.44 Tampilan Cetak Jadwal Laboratorium dalam Format PDF . . . . .	D - 25
D.45 Tampilan Kelola Mata Kuliah Laboratorium . . . . .	D - 26
D.46 Tampilan Tambah Mata Kuliah Laboratorium . . . . .	D - 26
D.47 Tampilan Edit Mata Kuliah Laboratorium . . . . .	D - 26
D.48 Tampilan Kelola Dosen Laboratorium . . . . .	D - 27
D.49 Tampilan Tambah Dosen Laboratorium . . . . .	D - 27
D.50 Tampilan Edit Dosen Laboratorium . . . . .	D - 27
D.51 <i>Source Code</i> . . . . .	D - 28

## DAFTAR TABEL

2.1	Deskripsi <i>Use Case Diagram</i> . . . . .	16
2.2	Deskripsi <i>Activity Diagram</i> . . . . .	17
2.3	Deskripsi <i>Class Diagram</i> . . . . .	18
4.1	Analisis Kebutuhan Perangkat Lunak Pengembang . . . . .	29
4.2	Analisis Kebutuhan Perangkat Keras Pengembang . . . . .	29
4.3	Deskripsi Aktor . . . . .	30
4.4	Perancangan Tabel Dosen . . . . .	36
4.5	Perancangan Tabel Matkul . . . . .	37
4.6	Perancangan Tabel Ruangan . . . . .	38
4.7	Perancangan Tabel Jadwal . . . . .	39
4.8	Perancangan Tabel <i>User</i> . . . . .	40
4.9	Keterangan Tampilan Halaman Utama . . . . .	42
4.10	Keterangan Tampilan Halaman Selamat Datang . . . . .	43
4.11	Keterangan Tampilan Kelola Jadwal . . . . .	43
4.12	Keterangan Tampilan Tambah Jadwal . . . . .	44
4.13	Tabel Keterangan Tampilan Edit Jadwal . . . . .	45
5.1	Spesifikasi Perangkat Keras ( <i>Hardware</i> ) . . . . .	47
5.2	Spesifikasi Perangkat Lunak ( <i>Software</i> ) . . . . .	47
5.3	Pengujian <i>Black Box</i> . . . . .	58

## **DAFTAR SINGKATAN**

DB	: <i>Database</i>
DevOps	: <i>Development and Operations</i>
FST	: Fakultas Sains dan Teknologi
ILMIS	: <i>Integrated Laboratory Management Information System</i>
INT	: Internet
Kalab	: Kepala Laboratorium
Kaprodi	: Ketua Program Studi
Lab	: Laboratorium
LABVIS	: <i>Laboratory Visitor Information System</i>
LARIS	: <i>Laboratory Assistant Registration Information System</i>
NIDN	: Nomor Induk Dosen Nasional
NIP	: Nomor Induk Pegawai
PHP	: <i>Hypertext Preprocessor</i>
PRODI	: Program Studi
QR	: <i>Quick Response</i>
RSI	: Rekayasa Sistem Informasi
SE	: <i>Software Engineering</i>
Sekprodi	: Sekretaris Program Studi
SI	: Sistem Informasi
SIKAPE	: Sistem Kerja Praktek
SITARIS	: Sistem Informasi Inventaris Laboratorium
SIREPO	: Sistem Repotori Laboratorium
SOP	: Standar Operasional Prosedur
SITASI	: Sistem Informasi Tugas Akhir
SKS	: Satuan Kredit Semester
TA	: Tugas Akhir
UIN Suska Riau	: Universitas Islam Negeri Sultan Syarif Kasim Riau
UML	: <i>Unified Modelling Language</i>

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Laboratorium merupakan salah satu fasilitas vital dalam institusi pendidikan yang berperan penting dalam mendukung kegiatan praktikum dan penelitian (La Braca dan Kalman, 2021). Tata kelola laboratorium yang baik menjadi kunci dalam memastikan penggunaan peralatan dan fasilitas secara optimal dan efisien (Abrantes, 2020). Pengelolaan inventaris yang akurat dan efisien adalah bagian integral dari tata kelola ini yang bertujuan untuk membangun budaya kualitas dalam pendidikan tinggi (Abrantes, 2020). Dengan tata kelola yang efektif, institusi dapat memastikan bahwa semua sumber daya laboratorium digunakan secara maksimal untuk mendukung pengembangan ilmu pengetahuan dan teknologi.

Universitas Islam Negeri Sultan Syarif Kasim Riau (UIN Suska Riau) sebagai salah satu perguruan tinggi negeri di Indonesia memiliki Program Studi Sistem Informasi di bawah naungan Fakultas Sains dan Teknologi (UIN Suska Riau, 2023). Program Studi ini dilengkapi dengan fasilitas laboratorium yang menyeluruh untuk mendukung pelaksanaan Tri Dharma Perguruan Tinggi yaitu, Pengajaran, Penelitian, dan Pengabdian kepada masyarakat. Ketiga pilar ini berperan sinergis dalam menjadikan UIN Suska Riau sebagai kontributor signifikan dalam pengembangan ilmu pengetahuan, teknologi, dan kemajuan masyarakat.

Sejak tahun 2002, Program Studi Sistem Informasi telah mengelola tiga laboratorium terpadu di bawah Fakultas Sains dan Teknologi, yaitu Laboratorium Rekayasa Sistem Informasi (RSI), Laboratorium Internet (INT), dan Laboratorium Software Engineering (SE) (Laboratorium Sistem Informasi, 2023). Laboratorium-laboratorium ini berfungsi tidak hanya sebagai sarana praktikum bagi mahasiswa sesuai kurikulum, tetapi juga sebagai pusat kegiatan riset dan inovasi yang memberikan manfaat substansial bagi civitas akademika, termasuk mahasiswa dan dosen.

Laboratorium-laboratorium Program Studi Sistem Informasi dilengkapi dengan fasilitas yang memadai untuk mendukung pembelajaran mahasiswa dan berbagai kegiatan akademik lainnya (Lampiran B). Untuk memastikan efektivitas dan efisiensi tata kelola laboratorium, dilakukan evaluasi berkala terhadap seluruh aspek fasilitas yang ada (Laboratorium Sistem Informasi, 2023). Tata kelola laboratorium yang baik sangat penting untuk memantau dan mengelola penggunaan peralatan serta fasilitas laboratorium secara optimal sehingga dapat mendukung kegiatan

akademik dan penelitian dengan lebih baik (Dongapure, Choudhari, Yawale, dan Kawalkar, 2024).

Program Studi Sistem Informasi UIN Suska Riau dalam tata kelola laboratoriumnya sudah dilakukan dengan beberapa cara yaitu mulai dari pengelolaan inventaris laboratorium sebelumnya dilakukan secara manual yang mengakibatkan berbagai kendala seperti kesulitan dalam pemantauan dan pengelolaan data inventaris serta ketidakefisienan dalam pengolahan data. Wild (2021) menunjukkan bahwa pengelolaan inventaris yang tidak efisien dapat menghambat operasional laboratorium. Dalam mengatasi permasalahan tersebut, Laboratorium Program Studi Sistem Informasi telah mengimplementasikan sistem informasi inventaris bernama SITARIS (Lampiran B). Tata kelola laboratorium dalam hal kunjungan juga sudah diterapkan sistem informasi kunjungan bernama *Laboratory Visitor Information System* yang disingkat (LABVIS) pada tahun 2023 (Lampiran B). Hal tersebut bertujuan untuk mempermudah pemantauan dan pengelolaan kunjungan laboratorium secara efisien. Serta sedang dalam proses pengembangan sebuah sistem informasi pendaftaran asisten laboratorium yang bernama *Laboratory Assistant Registration Information System* yang disingkat (LARIS) (Lampiran B).

Setelah pengembangan SITARIS, manajemen tata kelola laboratorium mengalami peningkatan signifikan dibandingkan dengan proses manual sebelumnya. Sistem yang baru memungkinkan otomatisasi berbagai proses yang sebelumnya memerlukan penanganan manual yang memakan waktu. Pencatatan inventaris yang dahulu dilakukan dengan pembukuan manual, kini dapat dilakukan secara digital dengan sistem pengkodean otomatis yang akurat. Proses pencatatan pendanaan, pengelolaan barang, efisiensi informasi barang, pemeliharaan, dokumentasi, pemusnahan barang dan peminjaman barang, serta ruangan yang sebelumnya membutuhkan pencatatan berulang dan validasi manual, kini dapat dikelola melalui sistem dengan *workflow* yang jelas.

Namun demikian, meskipun SITARIS telah berperan penting dalam tata kelola laboratorium, perkembangan teknologi yang pesat dan kebutuhan laboratorium yang semakin kompleks telah memunculkan berbagai tantangan baru. Beberapa permasalahan yang teridentifikasi meliputi kesalahan dalam pembuatan kode barang (Lampiran B), disfungsi fitur peminjaman barang dan ruangan (Lampiran B), serta ketidaksesuaian format laporan akhir dengan kebutuhan kepala laboratorium (Lampiran B). Keterbatasan ini berdampak signifikan pada efektivitas manajemen laboratorium secara keseluruhan. Kesalahan dalam pembuatan kode barang menyebabkan kesulitan dalam pelacakan dan inventarisasi aset, sementara disfungsi

fitur peminjaman mengakibatkan terhambatnya proses administrasi yang mempengaruhi kelancaran kegiatan praktikum dan penelitian. Selain itu, tidak adanya fitur pengelolaan jadwal laboratorium membuat informasi ketersediaan laboratorium untuk tempat praktikum menjadi tidak *real-time*. Permasalahan-permasalahan ini secara kolektif menjadikan sistem tersebut tidak mampu secara optimal dalam mendukung kebutuhan tata kelola laboratorium.

Dalam pengembangan sistem informasi, berbagai metodologi pengembangan telah diperkenalkan untuk mengatasi tantangan yang timbul, masing-masing dengan kelebihan dan kekurangannya. Salah satu pendekatan yang semakin populer adalah *Agile Development* yang telah terbukti efektif dalam menghadapi dinamika dan perubahan kebutuhan pengguna yang cepat (Al-Saqqa, Sawalha, dan Abdel-Nabi, 2020). Berbeda dengan metodologi tradisional seperti *Waterfall* yang bersifat linier dan kurang fleksibel terhadap perubahan (Firdaus dan Irfan, 2020). Metode *Agile Development* memungkinkan pengembangan sistem secara iteratif dengan penyesuaian terus-menerus berdasarkan umpan balik yang diperoleh dari pengguna. Penelitian-penelitian sebelumnya menunjukkan bahwa pendekatan ini sangat cocok untuk pengembangan sistem informasi yang memerlukan fleksibilitas. Misalnya, Tuunanen dkk. (2023) berhasil mengembangkan metode prioritas risiko persyaratan dalam proyek pengembangan sistem informasi dengan pendekatan *Agile*. Sementara Wisnumurti dkk. (2022) membuktikan efektivitas metode ini dalam pengembangan sistem penjualan untuk toko lokal. Selain itu, Galimberti (2021) dan Zhang dkk. (2024) juga menunjukkan keberhasilan implementasi *Agile Development* dalam lingkungan universitas dan pengembangan arsitektur layanan mikro yang menyoroti fleksibilitas metode ini dalam berbagai konteks. Meskipun demikian, setiap metodologi pengembangan memiliki kelebihan dan keterbatasannya sendiri. Misalnya, *Waterfall* lebih cocok untuk proyek dengan persyaratan yang sudah jelas dan tidak berubah (Al-Saqqa dkk., 2020). Sementara V-Model menekankan pentingnya pengujian pada setiap tahap pengembangan. Namun, kedua metodologi tersebut kurang fleksibel dalam mengakomodasi perubahan yang terjadi di tengah proses. Sementara itu, meskipun DevOps memberikan keunggulan dalam integrasi pengembangan dan operasi secara berkelanjutan, fokusnya lebih pada otomatisasi dan pengelolaan infrastruktur Teknologi Informasi yang berbeda dengan prinsip dasar *Agile* yang lebih mengutamakan fleksibilitas. Oleh karena itu, *Agile Development* menjadi pilihan yang tepat dalam pengembangan sistem informasi yang membutuhkan adaptasi cepat, dan kemampuan untuk merespons perubahan secara efektif yang pada akhirnya dapat meningkatkan kualitas dan produktivitas.

tas proyek sistem informasi.

Berdasarkan temuan-temuan tersebut, pengembangan sistem diperlukan untuk meningkatkan kualitas dan fungsionalitas SITARIS dalam mendukung tata kelola laboratorium. Pengembangan berfokus pada peningkatan efisiensi pengelolaan dengan memperhatikan umpan balik pengguna secara berkelanjutan, mencakup perbaikan fitur yang ada dan penambahan fitur baru yang relevan. Sistem juga diintegrasikan dengan teknologi terbaru untuk memastikan kesesuaian dengan perkembangan zaman dan standar yang berlaku. Proses pengembangan dilakukan secara bertahap, mulai dari analisis kebutuhan hingga evaluasi dan pemeliharaan, dengan melibatkan partisipasi aktif pengguna untuk memastikan hasil yang sesuai dengan kebutuhan dan harapan. Selain itu, pengembangan ini juga bertujuan untuk menciptakan sistem yang lebih responsif dan adaptif terhadap perubahan kebutuhan laboratorium di masa mendatang.

Oleh karena itu, penelitian ini bertujuan untuk mengembangkan SITARIS Program Studi Sistem Informasi UIN Suska Riau menjadi Sistem Manajemen Laboratorium Terintegrasi.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, diperoleh rumusan masalah untuk penelitian ini adalah bagaimana pengembangan SITARIS dan integrasi dengan Sistem Laboratorium lainnya yaitu LABVIS dan LARIS menjadi Sistem Manajemen Laboratorium Terintegrasi menggunakan Metode *Agile Development*.

## 1.3 Batasan Masalah

Dalam melakukan penelitian diperlukan batasan agar tidak menyimpang dari apa yang direncanakan. Adapun batasan masalah dalam penelitian Tugas Akhir ini adalah sebagai berikut:

1. Penelitian ini hanya fokus pada pengembangan Sistem Informasi Inventaris Laboratorium (SITARIS) menjadi Sistem Manajemen Laboratorium Terintegrasi (ILMIS).
2. Pengembangan sistem menggunakan Metode *Agile Development*.
3. Penelitian ini hanya menambahkan fitur penjadwalan laboratorium dan integrasi dengan LABVIS dan LARIS.
4. Evaluasi sistem dilakukan berdasarkan wawancara dari pengguna di Program Studi Sistem Informasi UIN Sultan Syarif Kasim Riau.

## **1.4 Tujuan**

Tujuan Tugas Akhir ini adalah:

1. Menganalisis SITARIS sebagai Sistem Informasi Inventaris Laboratorium untuk mengetahui kekurangan yang ada.
2. Mengembangkan SITARIS menjadi Sistem Manajemen Laboratorium Ter-integrasi (ILMIS).

## **1.5 Manfaat**

Hasil penelitian diharapkan dapat menjadi sebuah Sistem Informasi Manajemen Laboratorium yang terintegrasi yang memberikan kemudahan dalam tata kelola laboratorium. Sistem ini diharapkan mampu mengelola tata kelola laboratorium dengan lebih efisien, memantau penggunaan peralatan dan fasilitas secara *real-time*, serta menyediakan laporan yang akurat dan sesuai dengan kebutuhan kepala laboratorium. Selain itu, sistem ini juga diharapkan dapat memfasilitasi peminjaman barang dan ruangan dengan lebih mudah, serta mengelola kunjungan laboratorium secara efektif. Dengan demikian, sistem ini mendukung kegiatan praktikum dan penelitian dengan lebih baik, serta meningkatkan kualitas pengelolaan laboratorium secara keseluruhan.

## **1.6 Sistematika Penulisan**

Sistematika penulisan laporan adalah sebagai berikut:

### **BAB 1. PENDAHULUAN**

BAB 1 pada Tugas Akhir ini berisi tentang: (1) Latar Belakang masalah; (2) Rumusan Masalah; (3) Batasan Masalah; (4) Tujuan; (5) Manfaat; dan (6) Sistematika Penulisan.

### **BAB 2. LANDASAN TEORI**

BAB 2 pada Tugas Akhir ini berisi tentang: (1) Profil Instansi; (2) Pengembangan Sistem Informasi; (3) Manajemen Laboratorium; (4) Lab SI Website; (5) *Laboratory Visitor Information System* (LARIS); (6) *Laboratory Assistant Registration Information System* (LARIS); (7) Sistem Informasi Inventaris (SITARIS); (8) *Agile Development*; (9) *Unified Modelling Language*; (10) Observasi; (11) Wawancara; (12) PHP; (13) Framework; (14) CodeIgniter; (15) *Black Box Testing*; (16) Visual Studio Code; (17) Astah; (18) Balsamiq; (19) Database; (20) MariaDB; (21) XAMPP.

### **BAB 3. METODOLOGI PENELITIAN**

BAB 3 pada Tugas Akhir ini berisi tentang: (1) Tahap Perencanaan; (2) Tahap Analisis dan Perancangan; (3) Tahap Implementasi dan Pengujian; (4) Tahap

Dokumentasi.

**BAB 4. ANALISIS DAN PERANCANGAN**

BAB 4 pada Tugas Akhir ini berisi tentang: (1) Analisis; (2) Perancangan.

**BAB 5. IMPLEMENTASI DAN PENGUJIAN**

BAB 5 pada Tugas Akhir ini berisi tentang: (1) Implementasi Sistem; (2) Pengujian Sistem.

**BAB 6. PENUTUP**

BAB 6 pada Tugas Akhir ini berisi tentang: (1) Kesimpulan; (2) Saran.

## **BAB 2**

### **LANDASAN TEORI**

#### **2.1 Profil Instansi**

Program Studi Sistem Informasi merupakan salah satu program studi di Fakultas Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim (UIN Suska) Riau. Program studi ini menawarkan jenjang pendidikan Strata 1 (S1) dan telah berdiri sejak Juli 2002 (Prodi Sistem Informasi UIN Suska Riau, 2023). Program Studi Sistem Informasi secara resmi didirikan berdasarkan SK Pendirian No. DJ.II/26/2006 tertanggal 20 Februari 2006 yang ditandatangani oleh Direktur Jenderal Perguruan Tinggi. Penyelenggaraan program studi ini diperkuat dengan SK Penyelenggaraan No. 3480/D/T/K-AI/2009 dan SK Izin Operasional No. Dj.I/123/2012 yang diterbitkan pada 25 Januari 2012 (Prodi Sistem Informasi UIN Suska Riau, 2023).

Saat ini, Program Studi Sistem Informasi telah terakreditasi Baik Sekali oleh LAM INFOKOM dengan nomor SK 018/SK/LAM-INFOKOM/Ak/S/III/2024 yang berlaku dari 19 Maret 2024 hingga 19 Maret 2029 (Prodi Sistem Informasi UIN Suska Riau, 2023). Program studi ini dapat dihubungi melalui email [faste.sif@uin-suska.ac.id](mailto:faste.sif@uin-suska.ac.id) dan informasi lebih lanjut dapat diakses melalui website resmi <https://sif.uin-suska.ac.id/>. Kampus Program Studi Sistem Informasi berlokasi di Jl. HR. Soebrantas No. 155 KM 15, Pekanbaru 28293.

#### **2.1.1 Sejarah Laboratorium Program Studi Sistem Informasi**

UIN Suska Riau memiliki fasilitas infrastruktur pendukung Tridharma Perguruan Tinggi yang baik, salah satunya adalah laboratorium terpadu di bawah Fakultas Sains dan Teknologi yang dikelola oleh Program Studi Sistem Informasi sejak tahun 2002 (UIN Suska Riau, 2023). Terdapat tiga laboratorium yang dikelola oleh Program Studi Sistem Informasi, yaitu Laboratorium Rekayasa Sistem Informasi (RSI), Laboratorium Internet (INT), dan Laboratorium *Software Engineering* (SE). Ketiga laboratorium tersebut merupakan aset penting yang dapat dimanfaatkan dengan baik untuk mencapai target-target universitas dan menghasilkan lulusan Program Studi Sistem Informasi yang kompeten dalam pendidikan, penelitian, serta pengabdian masyarakat dengan mengintegrasikan nilai-nilai keislaman. Laboratorium-laboratorium tersebut tidak hanya digunakan untuk praktikum mahasiswa sesuai dengan kurikulum, tetapi juga mampu mendukung berbagai kegiatan mahasiswa dan dosen dalam meningkatkan pengetahuan di bidang Sistem Informasi.

### **2.1.2 Visi**

Menjadi Laboratorium Program Studi Sistem Informasi yang memiliki keunggulan dalam bidang pendidikan, penelitian, dan pengabdian kepada masyarakat dengan menghasilkan lulusan yang proaktif, inovatif, dan profesional dalam bidang Sistem Informasi di tingkat lokal, regional, dan nasional yang berbasis nilai-nilai islami pada tahun 2030.

### **2.1.3 Misi**

Untuk mencapai Visi Laboratorium Program Studi Sistem Informasi, berikut Misi-misi yang harus dicapai, diantaranya:

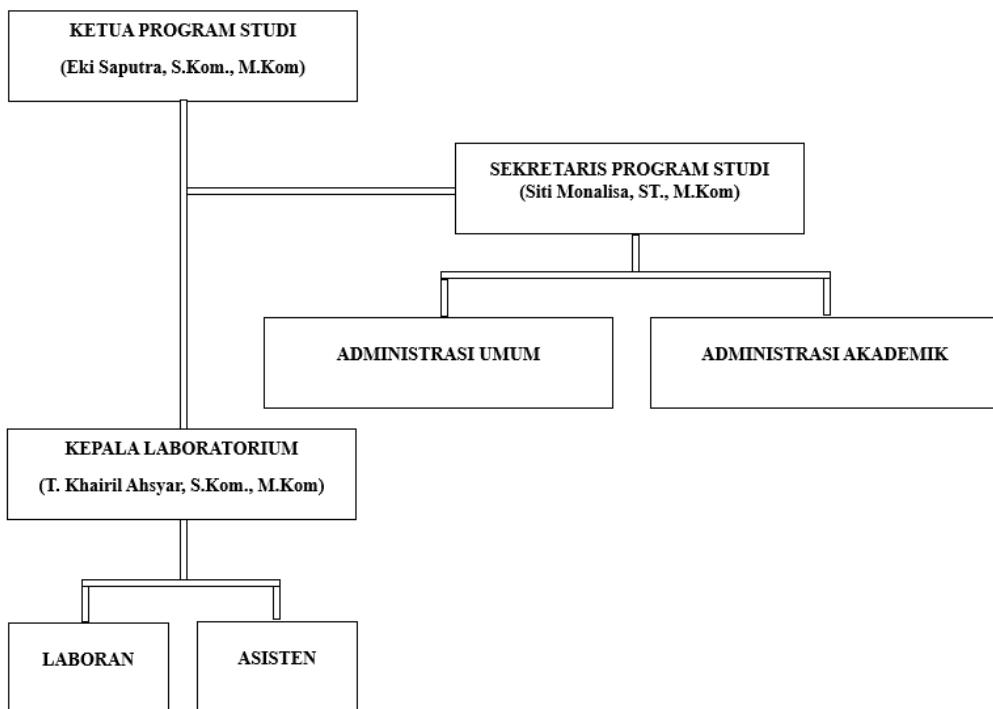
1. Mendukung penyelenggaraan kegiatan pendidikan akademik dan praktikum berbasis teknologi kepada mahasiswa, dosen, dan *stakeholder*.
2. Mendukung pelaksanaan kegiatan penelitian yang berbasis teknologi kepada mahasiswa, dosen, dan *stakeholder*.
3. Mendukung kegiatan pengabdian kepada masyarakat yang berbasis teknologi.
4. Menyiapkan sumber daya manusia yang mampu menerapkan teknologi informasi khususnya dibidang Sistem Informasi.
5. Membangun kemitraan dan jejaring dengan industri, pemerintah, dan organisasi nasional.

### **2.1.4 Struktur Organisasi**

Untuk menjalankan Tridharma Perguruan Tinggi dengan baik, pengelola laboratorium harus memiliki kemampuan manajerial yang baik dan dibantu dengan keahlian IT. Untuk mencapai hal ini, diperlukan sekelompok pengelola laboratorium yang percaya diri dan memiliki kemampuan. Gambar 2.1 menunjukkan struktur organisasi pengelola Laboratorium Program Studi Sistem Informasi dari 2021 hingga 2024.

## **2.2 Pengembangan Sistem Informasi**

Pengembangan sistem mengacu pada proses terstruktur pembuatan dan pemeliharaan sistem informasi yang mencakup perangkat keras, perangkat lunak, data, prosedur, dan personel. Proses ini sangat penting untuk mengatasi tantangan organisasi dan memanfaatkan peluang melalui implementasi sistem berbasis komputer (Efendi, Ramadhani, Zihad, dkk., 2023). Pengembangan sistem informasi merupakan serangkaian aktivitas yang melibatkan analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan sistem. Dalam prosesnya, pengem-



**Gambar 2.1.** Struktur Organisasi Laboratorium

bangun sistem memerlukan pemahaman mendalam tentang kebutuhan pengguna, proses bisnis organisasi, serta teknologi terkini. Selain itu, pengembangan sistem juga harus mempertimbangkan aspek keamanan, skalabilitas, dan kemudahan penggunaan untuk menghasilkan sistem yang efektif dan efisien. Keberhasilan pengembangan sistem sangat bergantung pada komunikasi yang baik antara pengembang dan pemangku kepentingan, serta penerapan metodologi pengembangan yang tepat sesuai dengan karakteristik proyek (Efendi dkk., 2023).

### 2.3 Manajemen Laboratorium

Manajemen didefinisikan sebagai proses perencanaan, pengorganisasian, memimpin, dan mengendalikan sumber daya untuk mencapai tujuan tertentu secara efisien dan efektif (Kaehler, Grundei, Kaehler, dan Grundei, 2019). Definisi dasar ini menggariskan bahwa manajemen dalam berbagai konteks, termasuk olahraga dan bisnis, di mana ia memastikan realisasi tujuan operasional dan strategis (Kaehler dkk., 2019). Fungsi utama manajemen adalah perencanaan, pengorganisasian, kepemimpinan, dan pengendalian sangat penting untuk kelancaran operasi organisasi manapun (Feng, Li, dan McVay, 2009). Selain itu, berbagai teori

manajemen, seperti teori klasik dan perilaku, menyediakan kerangka kerja yang membantu manajer mengembangkan strategi efektif yang disesuaikan dengan lingkungan unik mereka (Hussain, Haque, dan Baloch, 2019). Pada akhirnya, manajemen strategis mencakup perumusan dan implementasi tujuan utama dengan mempertimbangkan faktor internal dan eksternal yang sangat penting untuk kesuksesan jangka panjang dan keunggulan kompetitif (Schühly dan Schühly, 2022).

Manajemen laboratorium mencakup pendekatan sistematis untuk mengawasi operasi laboratorium yang mencakup pengumpulan data, manajemen inventaris, dan memastikan kontrol kualitas. Ini melibatkan integrasi berbagai komponen seperti tenaga kerja, peralatan, dan sumber daya keuangan untuk meningkatkan efisiensi operasional dan mendukung inovasi ilmiah (Marwah, Puspitorini, dkk., 2024). Sistem manajemen laboratorium modern telah berkembang untuk memasukkan solusi digital yang mengotomatiskan proses, meningkatkan aksesibilitas data, dan memfasilitasi berbagi sumber daya, sehingga mengatasi keterbatasan metode tradisional (Rihm dkk., 2024). Aspek kunci dari manajemen laboratorium yang efektif juga melibatkan praktik penjaminan kualitas yang memastikan kepatuhan terhadap praktik laboratorium yang baik dan keandalan hasilnya (Kawai dkk., 2021). Secara keseluruhan, manajemen laboratorium yang efektif sangat penting untuk mengoptimalkan fungsi laboratorium, meningkatkan hasil pendidikan, dan mendorong kemajuan ilmiah (Marwah dkk., 2024).

### **2.3.1 Laboratorium Rekayasa Sistem Informasi (RSI)**

Laboratorium merupakan sarana dalam melaksanakan sebuah riset dalam bidang ilmiah, eksperimen, pengukuran maupun pelatihan ilmiah. Meski laboratorium telah memiliki alat-alat yang lengkap, pengelolaan laboratorium juga harus diperhatikan. Adanya alat-alat yang sudah lengkap dan penggunaan yang sudah baik tentunya perlu untuk dilakukan manajemen yang baik pada laboratorium tersebut, karena terdapat beberapa hal yang harus diperhatikan kembali seperti pengelolaan masing-masing laboratorium dan pengolahan data (Sweden dkk., 2022).

Laboratorium Rekayasa Sistem Informasi atau yang disingkat dengan nama Laboratorium RSI merupakan laboratorium pertama yang dimiliki oleh Program Studi Sistem Informasi sejak pindahnya aktivitas perkuliahan kampus dari kampus Sukajadi ke kampus utama Panam Pekanbaru Riau pada tahun 2007. Fungsi utama dari laboratorium ini adalah sebagai fasilitas infrastruktur pendukung untuk pelaksanaan kegiatan perkuliahan praktikum bagi mahasiswa Program Studi Sistem Informasi terkait bidang Rekayasa Sistem Informasi. Bidang Rekayasa Sistem In-

formasi merupakan bidang yang paling dominan yang ada di Program Studi Sistem Informasi (Laboratorium Sistem Informasi, 2023). Gambar Laboratorium Rekayasa Sistem Informasi dapat dilihat pada Lampiran D.1.

### **2.3.2 Laboratorium Internet (INT)**

Laboratorium Internet atau yang disingkat dengan nama Laboratorium INT merupakan laboratorium milik Program Studi Sistem Informasi di bawah Fakultas Sains dan Teknologi kedua yang aktivitas perkuliahanya berada di kampus utama Panam Pekanbaru Riau. Secara spesifik, laboratorium ini lebih dioperasikan untuk kebutuhan perkuliahan terkait matakuliah praktikum dasar, seperti matakuliah Jaringan Komputer dan Pemrograman Dasar (Laboratorium Sistem Informasi, 2023). Gambar Laboratorium Internet dapat dilihat pada Lampiran D.2.

### **2.3.3 Laboratorium Software Engineering (SE)**

Laboratorium ketiga yang dimiliki oleh Program Studi Sistem Informasi adalah Laboratorium *Software Engineering* atau yang disingkat dengan nama Laboratorium SE. Laboratorium ini merupakan laboratorium terbaru milik yang dikelola oleh Program Studi dari usulan pengadaan barang tahun anggaran 2021 di bawah naungan Fakultas Sains dan Teknologi UIN Suska Riau. Adapun laboratorium SE sebagai pendukung dalam pelaksanaan kegiatan perkuliahan praktikum bagi mahasiswa Program Studi Sistem Informasi yang terkait dengan bidang keilmuan seperti Praktikum Basis Data, Pemrograman Berorientasi Objek, dan matakuliah wajib praktikum lainnya (Laboratorium Sistem Informasi, 2023). Gambar Laboratorium *Software Engineering* (SE) dapat dilihat pada Lampiran D.3.

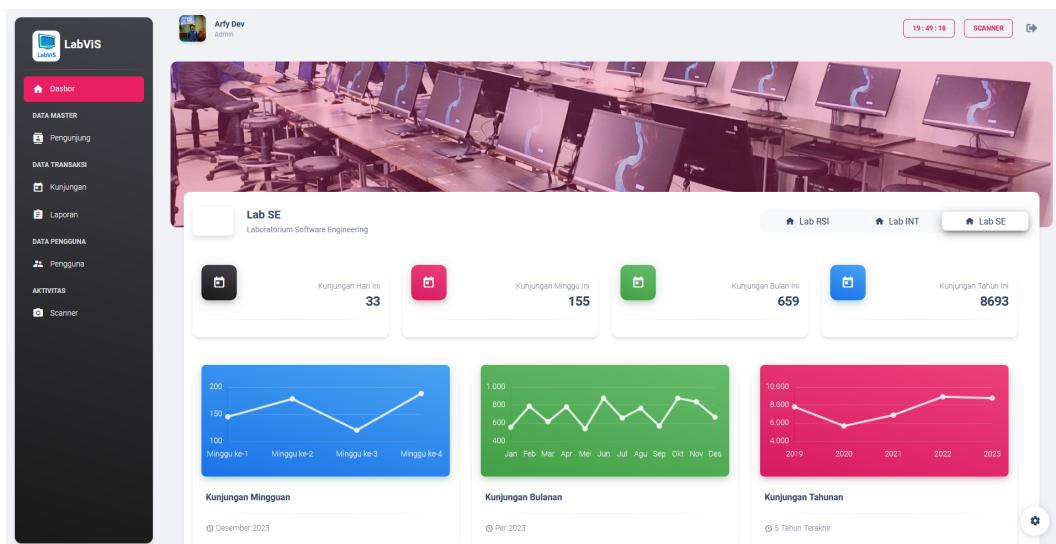
## **2.4 Lab SI Website**

Seiring dengan bertambah banyaknya informasi yang perlu disampaikan ke publik mengenai keberadaan (eksistensi) Laboratorium Program Studi Sistem Informasi, maka diperlukan media publik yang dapat menyampaikan informasi sekali-gus promosi fasilitas dan layanan kepada khalayak umum mengenai keberadaan Laboratorium yang dimiliki oleh Program Studi Sistem Informasi Fakultas Sains dan Teknologi UIN Sultan Syarif Kasim Riau. *Website* ini dibangun untuk dapat meng-cover hal-hal yang bersifat informasi agar pihak-pihak luar dapat mengetahui lebih mendalam tentang Profil Laboratorium Program Studi (Kusuma dan Ahsyar, 2024). Hal ini juga memberikan peluang kepada pihak luar yang ingin bekerja sama dengan memanfaatkan fasilitas laboratorium program studi untuk kegiatan-kegiatan akademik dan non akademik. Link Website Laboratorium : <https://lab-si.uin-suska.ac.id>

si.uin-suska.ac.id (Prodi Sistem Informasi UIN Suska Riau, 2023). Gambar Lab SI Website dapat dilihat pada Lampiran D.4.

## 2.5 *Laboratory Visitor System (LABVIS)*

LABVIS adalah sistem yang dirancang untuk mengelola dan memantau aktivitas kunjungan di laboratorium komputer Prodi Sistem Informasi. Dengan memanfaatkan scan *QR Code* pada kartu kunjungan, pengunjung dapat secara praktis mencatat riwayat kehadirannya. Grafik kunjungan interaktif memberikan visualisasi yang jelas tentang tren kunjungan mingguan, bulanan, dan tahunan. Terlebih lagi, LABVIS mampu menyimpan riwayat kunjungan untuk setiap pengunjung, memungkinkan mereka untuk terlacak di setiap kali aktivitasnya di laboratorium. Dengan begitu, LABVIS secara efektif berperan dalam melindungi lingkungan laboratorium agar dapat terjaga dengan baik (Prodi Sistem Informasi UIN Suska Riau, 2023). Gambar LABVIS dapat dilihat pada Gambar 2.2.



**Gambar 2.2.** *Laboratory Visitor Information System (LABVIS)*

## 2.6 *Laboratory Assistant Registration Information System (LARIS)*

LARIS adalah sistem pendaftaran dan pengelolaan asisten laboratorium. Sistem ini digunakan untuk mengelola asisten laboratorium program studi sistem informasi mulai dari tahap pendaftaran, rekrutmen, hingga menjadi asisten laboratorium (Prodi Sistem Informasi UIN Suska Riau, 2023). Gambar LARIS dapat dilihat pada Gambar 2.3.



**Gambar 2.3.** *Laboratory Assistant Registration Information System LARIS*

## 2.7 Sistem Informasi Inventaris (SITARIS)

SITARIS adalah sebuah *platform* yang dibuat dengan menggunakan *Framework* CodeIgniter4 dan bahasa pemrograman PHP yang dimaksudkan untuk membantu mengelola dan memantau inventaris barang dan peralatan laboratorium. Sistem ini memiliki potensi untuk meningkatkan efisiensi dan produktivitas dalam operasional laboratorium berkat berbagai fitur utama yang ditawarkannya. Selain itu, sistem ini juga dirancang untuk memudahkan pelacakan dan pemeliharaan aset laboratorium secara real-time.

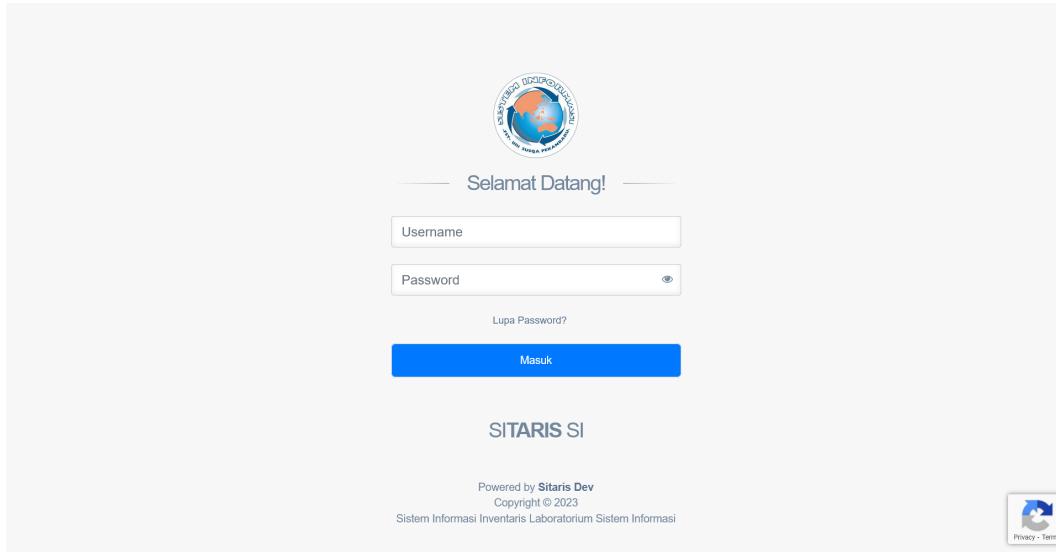
Dengan adanya sistem ini, diharapkan pengelolaan inventaris laboratorium menjadi lebih mudah dan efisien, sehingga staf laboratorium dapat fokus pada tugas yang lebih penting. Selain itu, laporan yang dihasilkan oleh sistem dapat membantu dalam pengambilan keputusan yang lebih baik tentang persediaan barang dan peralatan laboratorium (SITARIS SI, 2024). SITARIS dapat diakses di alamat <https://sitaris.lab-si.uin-suska.ac.id>. Terdapat beberapa menu yang ada pada SITARIS, yaitu:

1. Halaman *login*

Halaman *login* merupakan tampilan awal sistem ketika diakses. Terdapat formulir *username* dan *password* dan dilindungi oleh anti spam dari google reCAPTCHA yang digunakan untuk masuk ke dalam sistem informasi inventaris seperti pada Gambar 2.4.

2. Halaman Beranda

Halaman beranda merupakan tampilan awal yang ditampilkan kepada *user*



**Gambar 2.4.** Halaman *Login*

jika *user* berhasil *login* seperti pada Gambar 2.5.

**Gambar 2.5.** Halaman Beranda

### 3. Halaman Barang

Halaman barang merupakan tampilan untuk melihat dan mengelola data barang, tombol tambah data merupakan tombol yang dapat digunakan untuk beralih ke halaman tambah data barang, dan tombol pensil digunakan untuk mengedit data barang dan tombol *trash* untuk menghapus data barang, lalu terdapat juga tombol berwarna biru toska yang dibedakan menjadi beberapa tombol yang bertujuan untuk mencetak dokumen laporan berdasarkan pen-

danaan, ruangan, kategori, tahun, dan QR seperti pada Gambar 2.6.

NO	QR	Nama Barang	Subkategori	Spesifikasi Barang	Aksi
1		Keyboard Lenovo 2015-4 KEY AW9	Keyboard	-	
2		Keyboard Lenovo 2015-4 KEY IS3	Keyboard	-	
3		Mouse Logitech 2015-4 MOU ROS	Mouse	-	
4		Mouse Acer 2015-4 MOU AN5	Mouse	-	
5		Mouse Acer 2015-4 MOU ZW1	Mouse	-	
6		Mouse Wernes 2015-4 MOU IX2	Mouse	-	
7		Mouse Wernes 2015-4 MOU LZ9	Mouse	-	

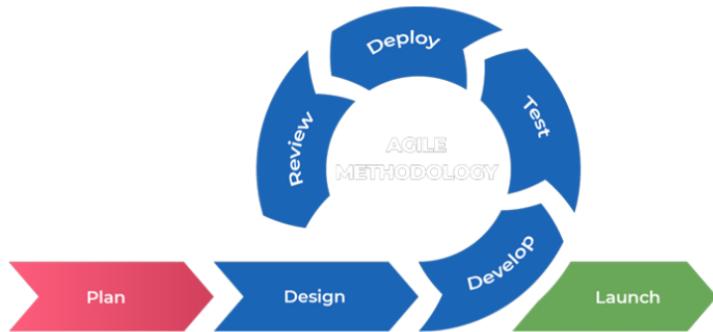
Gambar 2.6. Halaman Barang Index

## 2.8 Agile Development

Dalam penelitian ini, digunakan model pengembangan sistem *Agile* sebagai metodologi pengembangan perangkat lunak. Ada berbagai metode dan pendekatan pengembangan perangkat lunak tradisional seperti pendekatan *waterfall*. Pendekatan-pendekatan ini sering disebut sebagai pendekatan pengembangan perangkat lunak terencana atau pendekatan kelas berat. Pendekatan-pendekatan ini sangat berguna dalam mengembangkan perangkat lunak yang kompleks, membantu menghindari pengembangan perangkat lunak gaya lama yang informal dan memberikan perangkat lunak berkualitas tinggi secara sistematis, sehingga memenuhi persyaratan pengguna dalam batas waktu yang telah ditentukan (Al-Saqqa dkk., 2020).

## 2.9 Unified Modelling Language

*Unified Modelling Language* (UML) merupakan sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software*. UML sendiri juga memberikan standar penelitian sebuah sistem *blueprint* yang meliputi konsep bisnis proses, penelitian kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software* (Mubarak, 2019). Ada beberapa jenis diagram UML untuk membantu perancangan



**Gambar 2.7. Agile Development**

sistem antara lain:

1. *Use Case Diagram*

Menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, dan merepresentasikan sebuah interaksi antara aktor dan sistem. Didalam *use case* terdapat aktor sebagai gambaran entitas dari manusia atau sebuah sistem yang melakukan pekerjaan di sistem. Keterangan Simbol *Use Case Diagram* dapat dilihat pada Tabel 2.1.

**Tabel 2.1.** Deskripsi *Use Case Diagram*

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

**Tabel 2.1** Deskripsi *Use Case Diagram* (Tabel lanjutan...)

No	Simbol	Nama	Keterangan
6	—	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasiakan paket yang menampilkan sistem secara teraktas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).

## 2. *Activity Diagram*

Diagram aktivitas atau *Activity Diagram* menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, Diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business work flow*). Dapat juga digunakan untuk menggambarkan aliran kejadian (*flow of event*) dalam *use case*. Aktivitas menggambarkan proses yang berjalan, sembari *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Tabel Keterangan Simbol *Activity Diagram* dapat dilihat pada Tabel 2.2.

**Tabel 2.2.** Deskripsi *Activity Diagram*

No	Simbol	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antau antar muka saling berinteraksi satu sama lain.
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity final</i>	Bagaimana objek dibentuk dan dihancurkan
5	—	<i>Fork node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran.

## 3. *Class Diagram*

*Class Diagram* menggambarkan struktur statis dari kelas dalam sistem dan menggambarkan atribut, operasi dan hubungan antara kelas. *Class Diagram* juga membantu dalam memvisualisasikan struktur kelas-kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak dipakai. Keterangan Simbol *Class Diagram* dapat dilihat pada Tabel 2.3.

**Tabel 2.3.** Deskripsi *Class Diagram*

No	Simbol	Nama	Keterangan
1		Package	Package merupakan sebuah bungkusan dari satu atau lebih kelas.
2		Operasi	Kelas pada struktur sistem.
3		Asosiasi berarah/ <i>Directed association</i>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4		Generalisasi	Relasi antar kelas dengan makna generalisasi spesialisasi (umum khusus).
5		Kebergantungan/ <i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas.

## 2.10 Observasi

Observasi adalah tindakan mengamati secara langsung perilaku individu, objek, atau aktivitas dengan cara yang teratur tanpa melakukan interaksi langsung dengan subjek yang diamati. Observasi merupakan metode pengumpulan data di mana pengamat mengamati suatu sistem atau entitas saat sedang beroperasi untuk mendapatkan wawasan dan pemahaman yang lebih mendalam tentang bagaimana sistem tersebut bekerja (Tilley dan Rosenblatt, 2017).

## 2.11 Wawancara

Wawancara adalah metode pengumpulan data yang dilakukan melalui tanya jawab langsung antara peneliti dengan narasumber atau responden untuk mendapatkan informasi yang dibutuhkan (Monday, 2020). Teknik ini memungkinkan peneliti untuk mengumpulkan data kualitatif yang mendalam dan memahami perspektif, pengalaman, serta pengetahuan dari narasumber secara langsung. Wawancara dapat dilakukan secara terstruktur dengan menggunakan daftar pertanyaan yang telah disiapkan sebelumnya, semi-terstruktur yang memungkinkan fleksibilitas dalam mengajukan pertanyaan, atau tidak terstruktur yang bersifat lebih informal dan mengalir (Balza, Cusatis, McDonnell, Basir, dan Flynn, 2022). Dalam konteks

pengembangan sistem, wawancara sering digunakan untuk mengidentifikasi kebutuhan pengguna, mengumpulkan persyaratan sistem, dan memahami proses bisnis yang ada (Rueda, Panach, dan Distante, 2020).

## 2.12 PHP

*Hypertext Preprocessor* (PHP) adalah bahasa pemrograman *server-side* yang digunakan untuk membuat situs *web* dinamis dan interaktif. PHP merupakan bahasa pemrograman yang populer dan mudah dipelajari, serta memiliki banyak fungsi yang dapat digunakan untuk membuat situs web yang interaktif dan dinamis. PHP dapat digunakan untuk membuat situs web yang interaktif, seperti form pendaftaran, login, dan lainnya. PHP juga dapat digunakan untuk membuat situs web yang dinamis, seperti situs web yang dapat menampilkan data dinamis dari *database* (Cowls, Tsamados, Taddeo, dan Floridi, 2021). Dalam penelitian ini, versi PHP yang digunakan adalah versi 8.3.17.

## 2.13 Framework

*Framework* dalam pengembangan sistem adalah kerangka kerja atau struktur yang digunakan untuk memudahkan pengembangan aplikasi atau sistem (Sallaby dan Kanedi, 2020). *Framework* menyediakan berbagai fitur dan fungsi yang dapat digunakan oleh pengembang untuk mempercepat proses pengembangan dan memastikan konsistensi dalam pengembangan aplikasi atau sistem (Simanullang, Silalahi, dan Manalu, 2021). *Framework* juga membantu pengembang dalam mengelola kode program dan memperbaiki *bug*. Beberapa contoh *framework* yang sering digunakan dalam pengembangan sistem adalah Laravel, CodeIgniter, dan beberapa *framework* lainnya (Fadllullah, Mulyadi, Rochaniati, dan Nabil, 2022).

## 2.14 CodeIgniter

Codeigniter merupakan *framework* untuk membangun aplikasi *web* berbasis PHP. Codeigniter menyediakan banyak *library* untuk fungsi-fungsi umum, antar muka yang sederhana, dan struktur yang logis. CodeIgniter menjadi sebuah *framework* PHP dengan *Model*, *View*, *Controller* (MVC) untuk membangun *website* dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi *web*. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kode-nya. *Programmer* dapat membuat aplikasi dengan lebih cepat karena tidak perlu menulis kode dari awal, selain itu Codeigniter juga menyediakan banyak fungsi yang siap

digunakan. Seorang *programmer* bisa lebih fokus dengan aplikasi yang sedang dibangun dan meminimalkan penulisan kode (Tyowati dan Irawan, 2017). Dalam penelitian ini, versi CodeIgniter yang digunakan adalah versi CodeIgniter 4.

## 2.15 Black Box Testing

*Black Box Testing* adalah metode pengujian perangkat lunak yang berfokus pada fungsionalitas eksternal sistem tanpa memperhatikan struktur internal kode. Dalam pengujian ini, pengujian dilakukan berdasarkan spesifikasi fungsional sistem dan tidak memerlukan pengetahuan tentang implementasi internal sistem (Henard, Papadakis, Harman, Jia, dan Le Traon, 2016).

Metode *Black Box Testing* dilakukan dengan cara menguji sistem dari luar, seperti pengguna akhir akan melakukannya. Pengujian ini bertujuan untuk memastikan bahwa sistem berfungsi sesuai dengan kebutuhan pengguna dan spesifikasi fungsional yang telah ditentukan.

## 2.16 Visual Studio Code

Visual Studio Code (VS Code) adalah editor kode serbaguna yang telah berkembang secara signifikan untuk mendukung berbagai lingkungan pemrograman di Windows, macOS, dan Linux. Ini mengintegrasikan fitur untuk menulis dan men-debug kode, termasuk dukungan untuk .NET 7 dan konsumsi layanan AI, meningkatkan produktivitas dan efisiensi pengembang (Bree dan Gallagher, 2016). Dalam penelitian ini, versi Visual Studio Code yang digunakan adalah versi 1.99.3.

## 2.17 Astah

Astah adalah perangkat lunak pemodelan UML yang digunakan untuk merancang dan mendokumentasikan sistem perangkat lunak. Astah menyediakan berbagai fitur untuk membuat berbagai jenis diagram UML, seperti *use case diagram*, *class diagram*, dan *activity diagram* (Hayati, Rahayu, dan Saputra, 2021). Dalam pengembangan ini, versi Astah yang digunakan adalah versi 10.1.0.

## 2.18 Balsamiq

Balsamiq adalah perangkat lunak prototyping *wireframing* yang digunakan untuk membuat desain awal aplikasi *website* dan *mobile*. Balsamiq menyediakan berbagai komponen *User Interface* yang siap pakai, seperti tombol, formulir, dan tabel yang memungkinkan pengembang untuk membuat prototipe dengan cepat dan mudah. Balsamiq juga mendukung kolaborasi tim dan integrasi dengan berbagai alat pengembangan (Balsamiq, 2024). Pada perancangan interface ini, versi balsamiq yang digunakan adalah versi 4.7.5.

## **2.19 Database**

*Database* adalah suatu kumpulan data yang telah diatur secara terstruktur, memungkinkan akses dan pengelolaan melalui sistem komputer. Jenis data yang dapat disimpan di dalamnya mencakup teks, gambar, suara, dan video, dengan berbagai tujuan seperti penyimpanan informasi, analisis data, dan pengambilan keputusan. Untuk membuat dan mengelola *database*, diperlukan perangkat lunak khusus seperti *MariaDB*, *Oracle*, atau *Microsoft SQL Server* (Cowls dkk., 2021).

## **2.20 MariaDB**

*MariaDB* adalah sistem manajemen basis data relasional (RDBMS) yang dapat dijalankan di server web. *MariaDB* adalah versi terbaru dari *MySQL* yang merupakan sistem manajemen basis data relasional (RDBMS) yang populer untuk aplikasi web. *MariaDB* memiliki fitur yang mirip dengan *MySQL*, tetapi memiliki beberapa perbedaan dalam implementasi dan performa. *MariaDB* juga memiliki dokumentasi yang lengkap dan dukungan komunitas yang kuat, sehingga memudahkan pengembang untuk membuat dan mengembangkan aplikasi web yang berjalan di server *MariaDB* (*MariaDB*, 2024). Dalam pengembangan ini, versi *MariaDB* yang digunakan adalah versi 8.0.30.

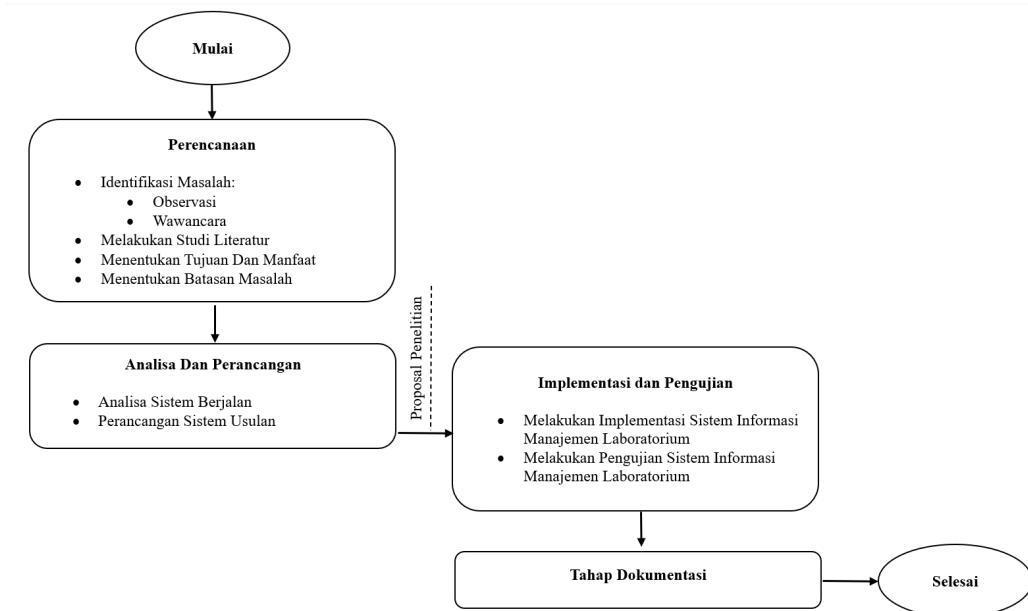
## **2.21 XAMPP**

*XAMPP* adalah sebuah paket lengkap untuk *web server* yang dapat dengan mudah *diinstal* di berbagai sistem operasi. Dalam paket ini sudah termasuk beberapa komponen penting seperti *Apache* (*web server*), *MariaDB* (*database*), PHP (*server side scripting*), dan berbagai pustaka pendukung lainnya. *XAMPP* dapat digunakan pada berbagai sistem operasi, termasuk Linux, Windows, dan Mac OS sehingga memudahkan pembuatan *server web multi-platform* (Pakpahan, Faâ, dkk., 2020). Dalam pengembangan ini, versi *XAMPP* yang digunakan adalah versi 8.0.30.

## BAB 3

# METODOLOGI PENELITIAN

Kerangka penelitian ini adalah langkah demi langkah dalam penyusunan Tugas Akhir mulai dari Tahap Perencanaan penelitian hingga Tahap Hasil dan Dokumentasi. Berikut ini adalah gambar Metodologi Penelitian dapat dilihat pada Gambar 3.1.



**Gambar 3.1.** Metodologi Penelitian

### 3.1 Tahap Perencanaan

Langkah pertama dalam penelitian ini adalah mengidentifikasi masalah, studi literatur, menentukan tujuan dan manfaat, menentukan batasan masalah, menentukan data-data serta informasi yang dibutuhkan saat penelitian.

#### 3.1.1 Identifikasi Masalah

Tahap ini bertujuan mengembangkan SITARIS menjadi sistem manajemen laboratorium. Rumusan masalahnya adalah "Bagaimana menerapkan Metode *Agile Development* dalam pengembangan SITARIS menjadi sistem manajemen laboratorium untuk meningkatkan kualitas dan fungsionalitas".

##### 1. Observasi

Pada tahap awal, peneliti melakukan pengamatan langsung pada studi kasus yang telah dipilih untuk mengidentifikasi kegiatan atau masalah yang terjadi pada studi kasus tersebut dan mengumpulkan data terkait.

## 2. Wawancara

Wawancara merupakan teknik pengumpulan data yang dilakukan melalui tatap muka dan tanya jawab langsung antara peneliti dan narasumber. Wawancara dilakukan untuk mendapatkan informasi secara tepat dan akurat dari narasumber yang terpercaya. Narasumber yang terkait pada penelitian ini yaitu Bapak Tengku Khairil Ahsyar S.Kom., M.Kom selaku Kepala Laboratorium Sistem Informasi.

### 3.1.2 Studi Literatur

Pada tahap ini, hal pertama yang dilakukan adalah melakukan penelitian literatur untuk mendapatkan informasi yang diperlukan untuk menulis tentang topik yang diangkat. Selain itu, kegiatan penelitian ini juga membantu mengetahui teori-teori, serta metode dan teknik yang berkaitan dengan topik atau masalah yang akan digunakan untuk mencapai tujuan yang diinginkan. Teori yang digunakan di sini berasal dari artikel jurnal.

### 3.1.3 Menentukan Tujuan dan Manfaat

Pada tahap ini dibahas tentang rumusan kalimat yang menunjukkan adanya hasil, tujuan penelitian, dan apa yang diperoleh setelah penelitian selesai. Tujuan penelitian dirumuskan untuk memberikan arahan yang jelas dalam pengembangan sistem. Manfaat penelitian dijelaskan untuk menunjukkan kontribusi yang diberikan terhadap pengembangan ilmu pengetahuan dan praktik. Hasil yang diharapkan dari penelitian ini adalah sistem manajemen laboratorium yang lebih efektif dan efisien.

### 3.1.4 Menentukan Batasan Masalah

Pada tahap ini yang dilakukan adalah membatasi subjek penelitian. Untuk mengumpulkan masalah, penelitian ini menggunakan observasi dan wawancara serta penelitian ini menggunakan Metode *Agile Development*.

## 3.2 Tahap Analisis dan Perancangan

Langkah ketiga dalam penelitian ini adalah menganalisis sistem yang sedang berjalan yaitu SITARIS. Analisis dilakukan untuk mengidentifikasi kekurangan dan kelebihan sistem yang ada. Hasil analisis ini akan menjadi dasar untuk pengembangan sistem yang lebih baik.

### **3.2.1 Menganalisis SITARIS**

Analisis SITARIS merupakan tahap krusial dalam pengembangan sistem. Proses ini melibatkan pemeriksaan menyeluruh terhadap berbagai aspek sistem untuk memastikan kualitas dan efektivitasnya. Langkah pertama dalam pengembangan ini adalah evaluasi fungsionalitas sistem. Peneliti akan memeriksa setiap fitur SITARIS untuk memastikan bahwa semua berfungsi sesuai dengan spesifikasi yang telah ditentukan. Ini mencakup pengujian setiap modul, menu, dan fungsi dalam sistem untuk memverifikasi bahwa mereka beroperasi dengan benar dan memberikan output yang diharapkan.

### **3.2.2 Agile Development**

Berikut adalah penjelasan mengenai tahapan Metode *Agile Development*:

1. Tahap Perencanaan dimana pihak pengembang sistem dan klien, Laboratorium Prodi Sistem Informasi, dapat melakukan perencanaan kebutuhan yang akan dikerjakan.
2. Tahap Rancangan dimana pihak pengembang sistem dapat merancang alur dan sistem manajemen yang akan dibuat.
3. Tahap Pengujian perangkat lunak dimana pihak pengembang sistem telah membuat sistem dan melakukan pengecekan apakah ada kesalahan dari sistem yang telah dibuat, dan jika ada kesalahan maka harus diperbaiki.
4. Tahap Dokumentasi dimana memberikan kemudahan bagi pengguna untuk memelihara sistem kedepannya.
5. Tahap Implementasi dimana pengembang sistem dapat menjamin kualitas sistem yang telah dibuat dengan menguji kualitas, keamanan, dan kecepatan dari sistem yang telah dibuat.

## **3.3 Tahap Implementasi dan Pengujian**

Selanjutnya melakukan tahap implementasi dan pengujian pada sistem. Langkah-langkahnya adalah sebagai berikut:

1. Implementasi dalam penelitian ini bahasa pemrograman yang dipilih untuk membangun sistem adalah PHP dengan *Framework CodeIgniter 4* dan VS Code sebagai editor codingnya.
2. Pengujian Sistem Setelah sistem selesai dibangun dengan menggunakan bahasa pemrograman yang dipilih, langkah selanjutnya adalah menguji sistem tersebut agar mengetahui suatu kesalahan yang terjadi. Pengujian dilakukan dengan menggunakan *Black Box Testing* dan *Manual Testing*.

### **3.4 Tahap Dokumentasi**

Langkah terakhir ialah melakukan dokumentasi semua kegiatan yang telah dilakukan mulai dari awal hingga akhir dengan membuat Laporan Tugas Akhir. Dokumentasi ini mencakup seluruh proses pengembangan sistem mulai dari perencanaan, analisis, desain, implementasi, hingga pengujian. Laporan Tugas Akhir ini juga berfungsi sebagai referensi dan pedoman untuk pengembangan sistem di masa depan.

## **BAB 4**

### **ANALISIS DAN PERANCANGAN**

#### **4.1 Analisis**

Analisis merupakan tahapan penting dalam pengembangan sistem yang melibatkan penguraian suatu pokok permasalahan menjadi bagian-bagian lebih kecil untuk dipelajari secara mendalam. Dalam konteks pengembangan sistem informasi, analisis bertujuan memahami kebutuhan pengguna, mengidentifikasi masalah yang ada pada sistem berjalan, serta merumuskan solusi tepat untuk mengatasi permasalahan tersebut. Proses analisis meliputi beberapa kegiatan utama, antara lain:

1. Pengumpulan data dan informasi melalui observasi, wawancara, dan studi dokumentasi
2. Identifikasi masalah pada sistem yang sedang berjalan
3. Analisis kebutuhan fungsional dan non-fungsional sistem
4. Evaluasi terhadap sistem yang ada untuk menemukan kekurangan dan peluang pengembangan
5. Perumusan solusi dan rekomendasi untuk pengembangan sistem baru

Pada bab ini, analisis dilakukan terhadap sistem yang berjalan di Laboratorium Program Studi Sistem Informasi, khususnya SITARIS, untuk mengidentifikasi kekurangan dan kebutuhan pengembangan lebih lanjut menjadi sistem manajemen laboratorium yang terintegrasi. Hasil analisis ini akan menjadi dasar dalam perancangan dan pengembangan sistem usulan.

##### **4.1.1 Analisis Sistem Berjalan**

Dalam proses tata kelola yang berlangsung di laboratorium Program Studi Sistem Informasi, hingga saat ini laboratorium telah menerapkan beberapa sistem informasi untuk mengelola berbagai aspek operasionalnya. Sistem-sistem tersebut meliputi:

1. *Laboratory Visitor Information System* disingkat LABVIS adalah sistem informasi yang digunakan untuk mengelola data kunjungan masuk dan keluar laboratorium, memungkinkan pemantauan dan pencatatan aktivitas pengunjung secara efisien seperti pada Lampiran D.5.
2. *Laboratory Assistant Registration Information System* disingkat LARIS adalah sistem informasi untuk mengelola data pendaftar dan proses rekrutmen asisten laboratorium seperti pada Lampiran D.6.
3. Sistem Informasi Inventaris disingkat SITARIS adalah sistem informasi

inventarisasi yang memfasilitasi pengelolaan dan pemantauan alat serta barang di laboratorium, meningkatkan efisiensi dalam manajemen inventaris seperti pada Gambar 2.4.

Implementasi sistem-sistem ini telah secara signifikan meningkatkan efektivitas dan efisiensi tata kelola laboratorium Program Studi Sistem Informasi, memungkinkan pengelolaan yang lebih terstruktur dan terintegrasi dalam berbagai aspek operasional laboratorium. Berdasarkan hasil observasi, ditemukan sejumlah permasalahan yang cukup krusial dalam sistem, antara lain kesalahan dalam pembuatan kode barang (Lampiran B), serta disfungsi pada fitur peminjaman barang dan ruangan (Lampiran B). Fitur peminjaman tersebut belum berjalan secara optimal dan belum sepenuhnya memenuhi Standar Operasional Prosedur (SOP) yang telah ditetapkan. Beberapa kelemahan yang teridentifikasi meliputi kurangnya akurasi dalam pencatatan data peminjaman, tidak adanya sistem verifikasi yang memadai untuk memastikan status barang atau ruangan yang dipinjam, serta minimnya notifikasi atau pelacakan terhadap barang yang masih berada dalam status dipinjam. Selain itu, sistem juga belum mampu secara otomatis mendeteksi dan memperbarui status barang yang telah dikembalikan, sehingga berpotensi menimbulkan kesalahan dalam inventarisasi. Kekurangan-kekurangan ini tidak hanya menghambat efisiensi operasional, tetapi juga berisiko menimbulkan kehilangan atau ketidaksesuaian data dalam manajemen aset secara keseluruhan, serta ketidaksesuaian format laporan akhir dengan kebutuhan kepala laboratorium (Lampiran B). Sistem tersebut masih memiliki kekurangan dalam menunjang tata kelola laboratorium, terutama dalam hal penjadwalan. Saat ini, tidak ada sistem informasi yang secara khusus mengelola penjadwalan laboratorium Program Studi Sistem Informasi. Pengelolaan penjadwalan masih dilakukan secara manual dengan melakukan validasi dan pengecekan pada jadwal yang diperoleh dari Ketua Program Studi. Keterbatasan ini berdampak signifikan pada efektivitas manajemen laboratorium secara keseluruhan dan mengakibatkan ketidaksesuaian dan kurangnya informasi mengenai jadwal praktikum di laboratorium. Oleh karena itu, perlu dilakukan penyempurnaan pada sistem informasi yang ada, khususnya SITARIS, agar dapat memenuhi kebutuhan tata kelola laboratorium dalam hal penjadwalan ruangan.

#### **4.1.2 Analisis Sistem Usulan**

Pengembangan sistem ini menyajikan fitur penjadwalan laboratorium yang dapat digunakan oleh Admin, Kaprodi, Sekprodi, dan Aslab. Fitur ini dirancang untuk mempermudah pengelolaan jadwal kegiatan di laboratorium, sehingga setiap

pengguna dapat dengan mudah mengakses dan mengelola informasi terkait jadwal. Selain itu, sistem ini juga akan mengintegrasikan sistem informasi yang sudah ada seperti LABVIS, dan LARIS. Penyempurnaan ini bertujuan untuk mencapai tujuan laboratorium dalam menerapkan *Integrated Laboratory Management Information System* (ILMIS) yang akan mengintegrasikan seluruh aspek manajemen laboratorium, termasuk penjadwalan, ke dalam satu sistem yang efisien. Dengan adanya sistem ini, diharapkan akan tercipta efisiensi dalam pengelolaan waktu dan sumber daya di laboratorium, serta mengurangi kemungkinan terjadinya bentrok jadwal antara berbagai kegiatan yang berlangsung dan juga mengintegrasikan sistem yang mulanya berdiri sendiri. Hasil akhir dari sistem ini adalah sebuah sistem terintegrasi yang memungkinkan semua pengguna untuk mengelola dan memantau jadwal laboratorium secara *real-time*, meningkatkan koordinasi antar pengguna, dan memastikan bahwa semua kegiatan laboratorium dapat berjalan dengan lancar tanpa adanya konflik jadwal.

#### **4.1.3 Analisis Kebutuhan Fungsional Sistem**

Sistem ini dirancang untuk memenuhi berbagai kebutuhan fungsional yang esensial dalam pengelolaan penjadwalan laboratorium. Sistem ini juga mendukung proses validasi yang terstruktur untuk memantau dan memberitahukan status penggunaan laboratorium kepada pengguna. Pengelolaan akses pengguna yang aman dan integrasi yang lancar dengan sistem internal laboratorium lainnya juga menjadi bagian integral dari fungsi sistem ini, memastikan efisiensi dan transparansi dalam seluruh proses penjadwalan laboratorium.

#### **4.1.4 Analisis Kebutuhan Non-Fungsional Sistem**

Kebutuhan non-fungsional sistem terbagi dalam dua kategori utama yaitu kebutuhan perangkat lunak dan kebutuhan perangkat keras. Analisis terhadap kebutuhan perangkat keras dilakukan untuk mengoptimalkan dan mempermudah proses perancangan serta implementasi sistem yang akan dibangun.

##### **1. Analisis Kebutuhan Perangkat Lunak**

Pada tahap analisis ini, peneliti mengidentifikasi dan mendefinisikan segala kebutuhan yang harus dipenuhi oleh sistem yang akan dikembangkan. Fokus utama dari analisis ini adalah memahami secara mendalam tujuan dan kebutuhan pengguna akhir, baik itu Admin, Kalab, Kaprodi, Sekprodi dan Aslab. Analisis kebutuhan perangkat lunak dapat dilihat pada Tabel 4.1.

**Tabel 4.1.** Analisis Kebutuhan Perangkat Lunak Pengembang

No	Perangkat Lunak	Versi Minimal	Versi Tersedia
1	Windows	W8	W11
2	Balsamiq Mockup	4.0.0	4.7.5
3	Google Chrome	-	127.0.6533.100
4	MySQL	8.0.0	8.0.30
5	VS Code	1.71.1	1.92.1
6	Hypertext Preprocessor (PHP)	8.0.0	8.2.16
7	CodeIgniter	4	4

2. Analisis Kebutuhan Perangkat Keras

Pada tahap analisis ini, peneliti mengidentifikasi dan mendefinisikan kebutuhan perangkat keras yang diperlukan dalam pengembangan sistem. Analisis ini bertujuan untuk memahami kebutuhan pengguna dan merancang solusi yang tepat dalam mengelola tata kelola laboratorium. Rincian analisis kebutuhan dapat dilihat pada Tabel 4.2.

**Tabel 4.2.** Analisis Kebutuhan Perangkat Keras Pengembang

No	Perangkat Keras	Versi Minimal	Versi Tersedia
1	<i>Processor</i>	<i>Intel Core i3 atau AMD Ryzen 3</i>	<i>AMD Ryzen 5 5600U, 6 Cores, 12 Threads</i>
2	<i>Memory</i>	<i>4 GB DDR4</i>	<i>16 GB DDR4-3200 MHz</i>
3	<i>Storage</i>	<i>256 GB SSD atau 500 GB HDD</i>	<i>512 GB M.2 NVMe</i>
4	<i>Keyboard</i>	<i>Standard QWERTY keyboard</i>	<i>6-row, multimedia Fn keys</i>
5	<i>Connection</i>	<i>Wi-Fi 802.11n atau Ethernet</i>	<i>Wi-Fi® 6</i>
6	<i>Monitor</i>	<i>14 inch, resolusi 1366x768</i>	<i>13 inc</i>

Dalam spesifikasi perangkat keras yang disarankan pada Sistem *Integrated Laboratory Management Information System* sesuai yang tertera pada Tabel 4.2 sebaiknya memenuhi syarat spesifikasi minimum agar sistem dapat berjalan dengan sempurna. Penggunaan perangkat keras dengan spesifikasi yang lebih tinggi dari minimum dapat meningkatkan performa sistem secara signifikan. Selain itu, pemilihan perangkat keras yang tepat juga akan mempengaruhi stabilitas dan keandalan sistem dalam jangka panjang.

## 4.2 Perancangan

Perancangan sistem perlu dilakukan sebelum dilakukan pembuatan sistem. tujuan dari perancangan sistem adalah untuk menentukan, mengorganisir, dan membentuk komponen dari solusi sistem akhir sehingga memiliki *blueprint* untuk membangun sistem.

### 4.2.1 Use Case Diagram

*Use Case Diagram* terdiri dari *actor*, *use case* serta hubungannya. *Use Case Diagram* adalah sesuatu yang penting untuk memvisualisasikan, menspesifikasi dan mendokumentasikan kebutuhan perilaku sistem. *Use Case Diagram* digunakan untuk menjelaskan kegiatan apa saja yang dapat dilakukan oleh *user* pengguna sistem yang sedang berjalan (Carstoiu dan Grigorescu, 1995). Dalam pengembangan sistem manajemen laboratorium ini, *Use Case Diagram* membantu mengidentifikasi interaksi antara pengguna dengan sistem secara jelas. Setiap aktor dalam diagram memiliki peran dan hak akses yang berbeda sesuai dengan kebutuhan fungsional sistem. Dengan adanya *Use Case Diagram*, tim pengembang dapat memahami alur kerja sistem dari perspektif pengguna. Selain itu, diagram ini juga berfungsi sebagai dasar komunikasi antara pengembang dan pemangku kepentingan untuk memastikan bahwa sistem yang dibangun sesuai dengan kebutuhan.

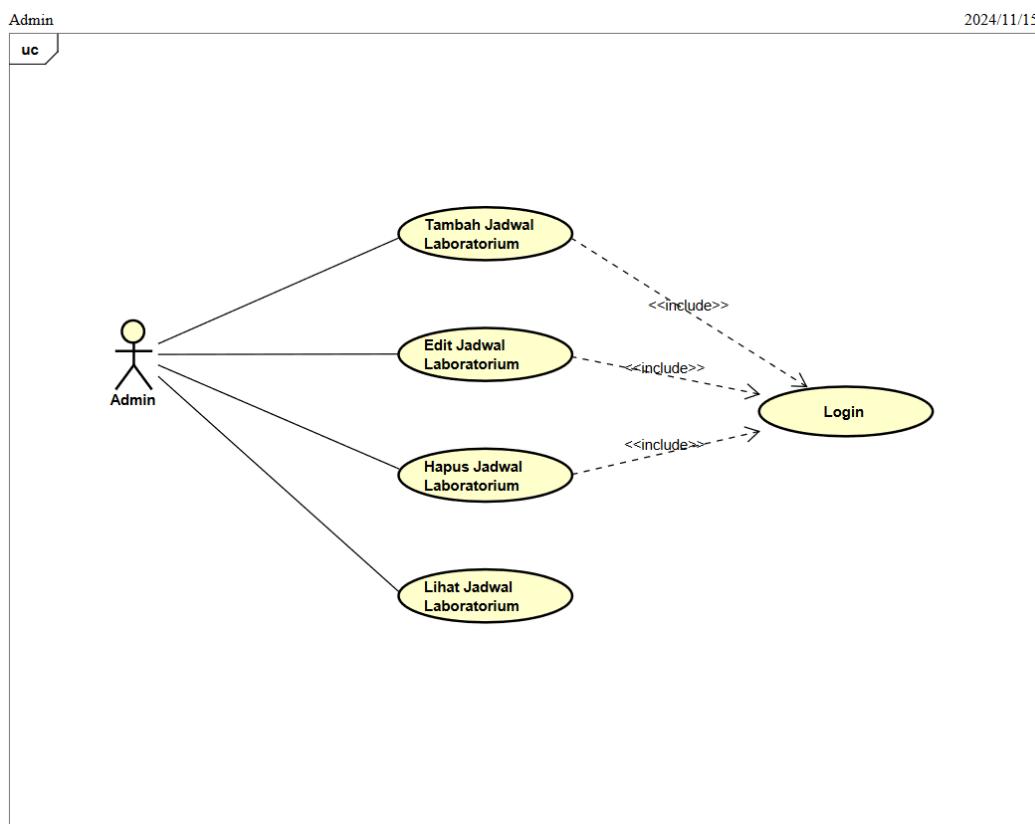
**Tabel 4.3.** Deskripsi Aktor

No	Aktor	Deskripsi
1	Admin	Mengelola penjadwalan laboratorium termasuk lihat, tambah, edit, dan hapus
2	Kalab	Mengelola penjadwalan laboratorium termasuk lihat, tambah, edit, dan hapus
3	Kaprodi	Melihat penjadwalan laboratorium
4	Sekprodi	Melihat penjadwalan laboratorium
5	Aslab	Mengelola penjadwalan laboratorium termasuk lihat, tambah, edit, dan hapus

Sistem manajemen laboratorium adalah sistem yang dikelola secara terpusat oleh seorang Admin. Sistem ini digambarkan menggunakan diagram *use case* yang menunjukkan interaksi antara admin dengan berbagai komponen sistem. Dalam sistem ini terdapat 4 fungsi utama yang dapat digunakan oleh admin dalam mengelola sistem. Fungsi-fungsi tersebut meliputi kemampuan Admin untuk menambahkan jadwal baru melalui menu "Tambah Jadwal Laboratorium", mengubah jadwal yang ada melalui "Edit Jadwal Laboratorium", menghapus jadwal yang tidak diperlukan

melalui "Hapus Jadwal Laboratorium", dan melihat daftar jadwal melalui "Lihat Jadwal Laboratorium".

Keamanan sistem dijamin melalui mekanisme *login* yang ditunjukkan dengan relasi "*include*" pada diagram. Tiga fungsi utama yaitu penambahan, pengeditan, dan penghapusan jadwal termasuk ke dalam lingkup wewenang Admin, oleh karena itu fungsi-fungsi tersebut mengharuskan Admin untuk *login* terlebih dahulu sebelum dapat mengaksesnya. Sementara itu, fungsi melihat jadwal laboratorium dapat diakses langsung tanpa memerlukan *login* seperti yang ditunjukkan dalam Gambar 4.1.



**Gambar 4.1. Use Case Diagram Admin**

Dalam sistem ini, Kalab memiliki empat fungsi utama yang dapat diakses. Fungsi-fungsi tersebut meliputi kemampuan untuk menambahkan jadwal baru melalui fitur "Tambah Jadwal Laboratorium", melakukan perubahan pada jadwal yang sudah ada menggunakan fitur "Edit Jadwal Laboratorium", menghapus jadwal yang tidak diperlukan dengan fitur "Hapus Jadwal Laboratorium", dan melihat seluruh daftar jadwal yang tersedia melalui fitur "Lihat Jadwal Laboratorium". Keamanan sistem dijamin melalui mekanisme *login* yang ditunjukkan dengan re-

lasi "include" pada diagram. Tiga fungsi utama, yaitu menambah, mengedit, dan menghapus jadwal, mengharuskan Kalab untuk melakukan *login* terlebih dahulu sebelum dapat mengakses fungsi-fungsi tersebut. Sementara itu, fungsi untuk melihat jadwal laboratorium dapat diakses secara langsung tanpa perlu melalui proses *login* terlebih dahulu, seperti yang ditunjukkan pada Lampiran D.7.

Dalam sistem ini, Kaprodi memiliki satu fungsi utama yang dapat diakses, yaitu melihat seluruh daftar jadwal yang tersedia melalui fitur "Lihat Jadwal Laboratorium". Fungsi ini dapat diakses secara langsung tanpa perlu melalui proses *login* terlebih dahulu, sehingga Kaprodi dapat dengan mudah memantau jadwal laboratorium yang ada. Hal ini ditunjukkan pada Lampiran D.8.

Dalam sistem ini, Sekprodi memiliki satu fungsi utama yang dapat diakses, yaitu melihat seluruh daftar jadwal yang tersedia melalui fitur "Lihat Jadwal Laboratorium". Fungsi ini dapat diakses secara langsung tanpa perlu melalui proses *login* terlebih dahulu, sehingga Sekprodi dapat dengan mudah memantau jadwal laboratorium yang ada. Hal ini ditunjukkan pada Lampiran D.9.

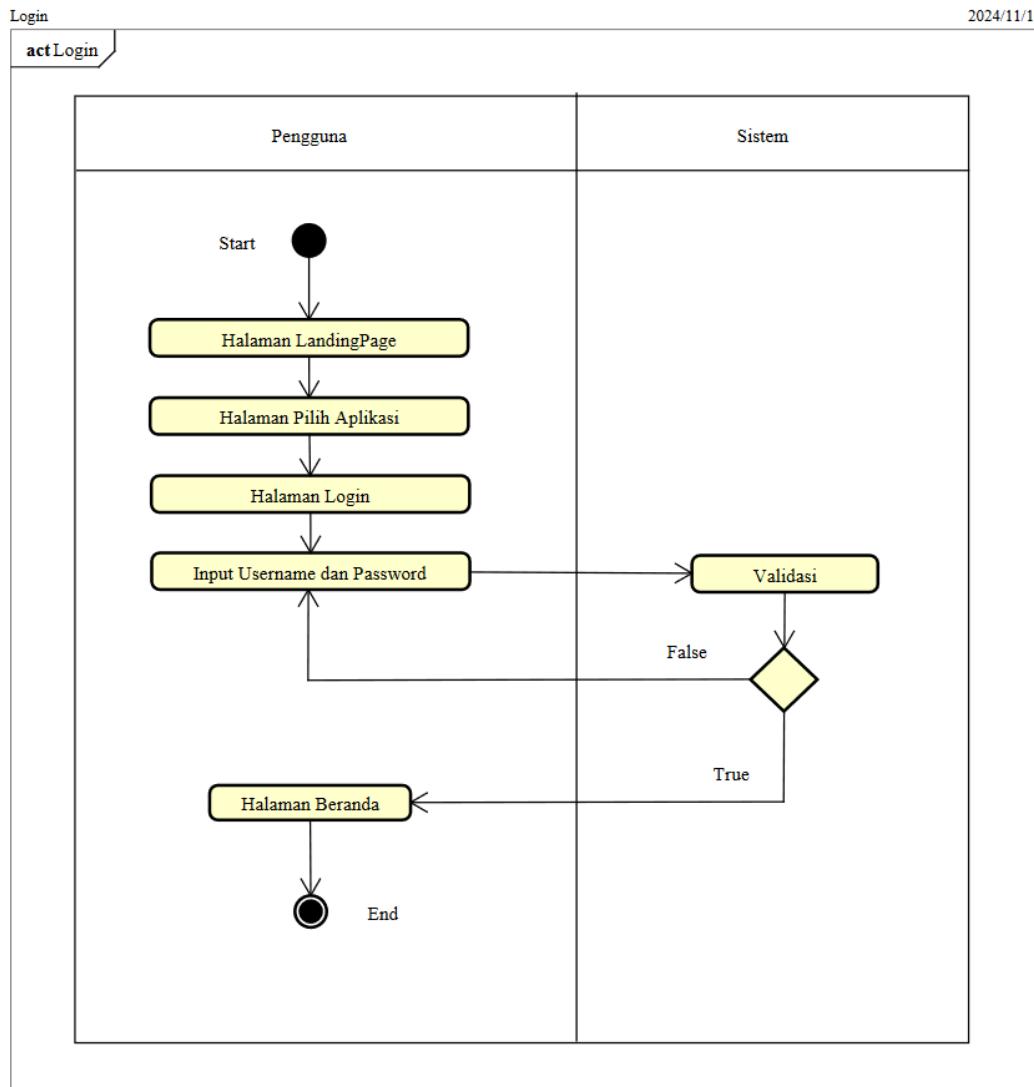
Dalam sistem ini, Aslab memiliki empat fungsi utama yang dapat diakses. Fungsi-fungsi tersebut meliputi kemampuan untuk menambahkan jadwal baru melalui fitur "Tambah Jadwal Laboratorium", melakukan perubahan pada jadwal yang sudah ada menggunakan fitur "Edit Jadwal Laboratorium", menghapus jadwal yang tidak diperlukan dengan fitur "Hapus Jadwal Laboratorium", dan melihat seluruh daftar jadwal yang tersedia melalui fitur "Lihat Jadwal Laboratorium". Keamanan sistem dijamin melalui mekanisme *login* yang ditunjukkan dengan relasi "include" pada diagram. Tiga fungsi utama, yaitu menambah, mengedit, dan menghapus jadwal, mengharuskan Aslab untuk melakukan *login* terlebih dahulu sebelum dapat mengakses fungsi-fungsi tersebut. Sementara itu, fungsi untuk melihat jadwal laboratorium dapat diakses secara langsung tanpa perlu melalui proses *login* terlebih dahulu, seperti yang ditunjukkan pada Lampiran D.10.

#### **4.2.2 Activity Diagram**

*Activity Diagram* adalah salah satu alat dalam *Unified Modeling Language* (UML) yang digunakan untuk memvisualisasikan alur kerja atau aktivitas dalam suatu sistem (Linzhang dkk., 2004). Pada sistem ini, *Activity Diagram* memberikan gambaran rinci tentang proses yang dilalui oleh berbagai aktor seperti Admin, Kalab, Kaprodi, Sekprodi, Aslab.

*Activity Diagram login* memberikan gambaran rinci tentang proses *login* yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat

dalam proses *login*, mulai dari input data oleh pengguna, validasi oleh sistem hingga dapat masuk ke dalam sistem menggunakan akun yang sudah ada. *Activity Diagram* ini dapat dilihat pada Gambar 4.2.



**Gambar 4.2.** *Activity Diagram Login*

*Activity Diagram tambah dosen* memberikan gambaran rinci tentang proses penambahan dosen yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses penambahan dosen, mulai dari input data oleh pengguna, validasi oleh sistem hingga penyimpanan data dosen baru ke dalam sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.11.

*Activity Diagram edit dosen* memberikan gambaran rinci tentang proses pengeditan data dosen yang dilalui oleh pengguna. Diagram ini memandu langkah-

langkah yang terlibat dalam proses pengeditan dosen, mulai dari pemilihan data dosen yang akan diedit, input data baru oleh pengguna, validasi oleh sistem hingga penyimpanan data dosen yang telah diperbarui ke dalam sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.12.

*Activity Diagram* hapus dosen memberikan gambaran rinci tentang proses penghapusan dosen yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses penghapusan dosen, mulai dari pemilihan data dosen yang akan dihapus, konfirmasi oleh pengguna, hingga penghapusan data dosen dari sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.13.

*Activity Diagram* tambah mata kuliah memberikan gambaran rinci tentang proses penambahan mata kuliah yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses penambahan mata kuliah, mulai dari input data oleh pengguna, validasi oleh sistem hingga penyimpanan data mata kuliah baru ke dalam sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.14.

*Activity Diagram* edit mata kuliah memberikan gambaran rinci tentang proses pengeditan data mata kuliah yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses pengeditan mata kuliah, mulai dari pemilihan data mata kuliah yang akan diedit, input data baru oleh pengguna, validasi oleh sistem hingga penyimpanan data mata kuliah yang telah diperbarui ke dalam sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.15.

*Activity Diagram* hapus mata kuliah memberikan gambaran rinci tentang proses penghapusan mata kuliah yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses penghapusan mata kuliah, mulai dari pemilihan data mata kuliah yang akan dihapus, konfirmasi oleh pengguna, hingga penghapusan data mata kuliah dari sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.16.

*Activity Diagram* tambah jadwal memberikan gambaran rinci tentang proses penambahan jadwal yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses penambahan jadwal, mulai dari input data oleh pengguna, validasi oleh sistem hingga penyimpanan data jadwal baru ke dalam sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.17.

*Activity Diagram* edit jadwal memberikan gambaran rinci tentang proses pengeditan jadwal yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses pengeditan jadwal, mulai dari pemilihan data jadwal yang akan diedit, input data baru oleh pengguna, validasi oleh sistem hingga penyimpanan data jadwal yang telah diperbarui ke dalam sistem. *Activity Diagram*

ini dapat dilihat pada Lampiran D.18.

*Activity Diagram* hapus jadwal memberikan gambaran rinci tentang proses penghapusan jadwal yang dilalui oleh pengguna. Diagram ini memandu langkah-langkah yang terlibat dalam proses penghapusan jadwal, mulai dari pemilihan data jadwal yang akan dihapus, konfirmasi oleh pengguna, hingga penghapusan data jadwal dari sistem. *Activity Diagram* ini dapat dilihat pada Lampiran D.19.

#### 4.2.3 *Class Diagram*

*Class Diagram* adalah representasi visual dari struktur kelas dalam sistem manajemen laboratorium, yang menunjukkan hubungan logis antar kelas. *Class Diagram* ini menyediakan deskripsi terperinci dari setiap kelas yang terlibat dalam sistem, termasuk atribut dan operasi yang diperlukan untuk mendukung fungsi manajemen laboratorium secara efektif. *Class Diagram* Sistem Informasi Manajemen Laboratorium dapat dilihat pada Lampiran D.20.

Relasi antara jadwal dengan ruangan, dosen, dan mata kuliah diwakili oleh id\_ruangan, id\_dosen, dan id\_matkul pada jadwal. Id\_ruangan menunjukkan keterkaitan jadwal dengan ruangan tertentu, id\_dosen menunjukkan keterkaitan jadwal dengan dosen tertentu, dan id\_matkul menunjukkan keterkaitan jadwal dengan mata kuliah tertentu. Hubungan ini digambarkan dengan garis yang menghubungkan entitas jadwal ke masing-masing entitas ruangan, dosen, dan mata kuliah dalam diagram.

#### 4.2.4 Perancangan *Database*

Perancangan *database* adalah perancangan basis data yang akan digunakan pada sebuah sistem, didasari oleh data perusahaan. Perancangan ini bertujuan agar tiap *field* data yang memiliki relasi dapat terhubung pada tabel di *database*, sehingga proses pengaksesan data akan dapat telaksana dengan lebih baik. Berikut adalah detail perancangan serta relasi yang ada pada *database* sistem informasi inventaris laboratorium pada Laboratorium Sistem Informasi. Berikut tabel perancangan *database*:

##### 1. Perancangan *Database* Tabel Dosen

Tabel Dosen dirancang untuk menyimpan informasi komprehensif tentang staf pengajar. Struktur tabel ini mencakup berbagai atribut yang diperlukan untuk mengidentifikasi dan mengelola data dosen secara efisien. Berikut adalah penjelasan ilmiah mengenai struktur dan fungsi Tabel Dosen:

- a. Tabel ini menetapkan `id_dosen` sebagai kunci utama dengan tipe data `tinyint (4)` dan fitur `auto increment` yang memastikan bahwa se-

- tiap dosen diberikan identifikasi yang unik dalam sistem.
- Atribut 'nama\_dosen' dan 'nip\_dosen' berfungsi untuk menyimpan informasi dasar tentang dosen yang memudahkan pengenalan personal dalam lingkungan akademis.
  - Atribut 'jenis\_kelamin' memakai tipe data enum untuk menjaga konsistensi data dan membantu dalam analisis demografis.
  - 'email\_dosen' dan 'no\_hp' merupakan saluran komunikasi yang vital, memungkinkan komunikasi yang efektif antara dosen dan sistem.
  - Atribut 'nidn' mencatat identifikasi nasional yang unik untuk dosen yang mendukung integrasi dengan sistem pendidikan tinggi yang lebih luas.

Struktur tabel ini dirancang dengan mempertimbangkan kebutuhan manajemen data dosen yang komprehensif, efisiensi penyimpanan, dan kemudahan dalam pemrosesan dan analisis data.

**Tabel 4.4.** Perancangan Tabel Dosen

<b>Field</b>	<b>Type</b>	<b>Length</b>	<b>Key</b>
id_dosen	tinyint	4	<i>Primary key (AI)</i>
nama_dosen	varchar	100	
nip_dosen	varchar	50	
jenis_kelamin	enum		('Laki-laki', 'Perempuan')
email_dosen	varchar	100	
nidn	varchar	100	
no_hp	varchar	100	

- Perancangan *Database* Tabel Matkul Tabel Matkul dirancang untuk menyimpan informasi tentang mata kuliah yang ditawarkan dalam program akademik. Struktur tabel ini mencakup berbagai atribut yang diperlukan untuk mengidentifikasi dan mengelola data mata kuliah secara efisien. Berikut adalah penjelasan ilmiah mengenai struktur dan fungsi Tabel Matkul:
  - Tabel ini memanfaatkan id\_matkul sebagai kunci utama dengan tipe data tinyint(4) dan fitur auto increment, yang menjamin bahwa setiap mata kuliah diberi identifikasi yang unik di dalam sistem.
  - Atribut 'kode\_matkul' dan 'nama\_matkul' berfungsi untuk menyimpan informasi dasar yang membantu dalam mengenali mata kuliah dengan cepat di lingkungan akademik.

- c. Atribut 'sks' dan 'semester' berisi informasi krusial mengenai bobot akademik dan penjadwalan mata kuliah dalam kurikulum, yang penting untuk perencanaan pendidikan.
- d. Atribut 'jenis\_matkul' memakai tipe data enum untuk membedakan mata kuliah menjadi kategori wajib atau pilihan, mendukung kelancaran dalam pengelolaan kurikulum.

Struktur tabel ini dirancang dengan mempertimbangkan kebutuhan manajemen data mata kuliah yang komprehensif, efisiensi penyimpanan, dan kemudahan dalam pemrosesan dan analisis data kurikulum.

**Tabel 4.5.** Perancangan Tabel Matkul

Field	Type	Length	Key
id_matkul	tinyint	4	<i>Primary key (AI)</i>
kode_matkul	varchar	50	
nama_matkul	varchar	100	
sks	tinyint	4	
semester	tinyint	4	
jenis_matkul	enum		('Wajib', 'Pilihan')

### 3. Perancangan *Database* Tabel Ruangan

Tabel Ruangan dirancang untuk menyimpan informasi tentang ruangan-ruangan yang tersedia untuk kegiatan akademik. Struktur tabel ini mencakup berbagai atribut yang diperlukan untuk mengidentifikasi dan mengejola data ruangan secara efisien. Berikut adalah penjelasan ilmiah mengenai struktur dan fungsi Tabel Ruangan:

- a. Tabel ini memanfaatkan id\_ruangan sebagai primary key. Dengan menggunakan tipe data tinyint (4) dan fitur auto increment, tabel ini memastikan bahwa setiap ruangan tercatat dengan identitas uniknya sendiri dalam sistem.
- b. Atribut 'id\_gedung' bertindak sebagai foreign key yang mengaitkan setiap ruangan dengan gedungnya, membantu dalam mengatur lokasi dengan lebih terstruktur.
- c. Atribut 'nama\_ruangan' berisi nama yang memudahkan pengguna dalam mengenali setiap ruangan.
- d. Atribut 'deskripsi\_ruangan' memberikan ruang untuk menambahkan keterangan lebih lanjut mengenai fasilitas atau ciri khas dari

ruangan tersebut.

- e. Atribut 'gambar\_ruangan' berisi jalur ke file gambar yang berkaitan dengan ruangan, memudahkan dalam visualisasi dan lebih memahami penampilan ruangan tersebut.

Struktur tabel ini dirancang dengan mempertimbangkan kebutuhan manajemen data ruangan yang komprehensif, efisiensi penyimpanan, dan kemudahan dalam pemrosesan dan analisis data fasilitas.

**Tabel 4.6.** Perancangan Tabel Ruangan

<b>Field</b>	<b>Type</b>	<b>Length</b>	<b>Key</b>
id_ruangan	tinyint	4	<i>Primary key (A.I)</i>
id_gedung	tinyint	4	<i>Foreign key</i>
nama_ruangan	varchar	100	
deskripsi_ruangan	text		
gambar_ruangan	varchar	255	

#### 4. Perancangan *Database* Tabel Jadwal

Tabel Jadwal dirancang untuk menyimpan informasi tentang penjadwalan kegiatan akademik. Struktur tabel ini mencakup berbagai atribut yang diperlukan untuk mengidentifikasi dan mengelola data jadwal secara efisien. Berikut adalah penjelasan ilmiah mengenai struktur dan fungsi Tabel Jadwal:

- a. Tabel ini dilengkapi dengan id\_jadwal sebagai primary key. Dengan tipe data tinyint(4) dan fitur auto increment, setiap jadwal dijamin memiliki identifikasi yang unik dalam sistem.
- b. Atribut seperti 'id\_ruangan', 'id\_matkul', dan 'id\_dosen' bertindak sebagai foreign key yang mengaitkan jadwal dengan informasi tentang ruangan, mata kuliah, dan dosen yang relevan, sehingga memudahkan pengelolaan jadwal secara terpadu.
- c. Atribut seperti 'tanggal', 'hari', 'jam\_masuk', dan 'jam\_keluar' berperan penting dalam mencatat waktu spesifik untuk setiap kegiatan dalam jadwal.
- d. Atribut 'deskripsi' memberikan ruang untuk menambahkan informasi detail tentang kegiatan atau jadwal yang direncanakan.
- e. Field-field seperti 'kode\_matkul', 'nama\_matkul', 'nama\_dosen', dan 'nama\_ruangan' walaupun redundan, tetapi sangat membantu dalam mempercepat akses informasi tanpa perlu

menggabungkan tabel berulang kali.

Struktur tabel ini dirancang dengan mempertimbangkan kebutuhan manajemen data jadwal yang komprehensif, efisiensi dalam pengambilan data, dan fleksibilitas dalam pengelolaan jadwal akademik.

**Tabel 4.7.** Perancangan Tabel Jadwal

<b>Field</b>	<b>Type</b>	<b>Length</b>	<b>Key</b>
id_jadwal	tinyint	4	Primary key (A_I)
id_ruangan	tinyint	4	Foreign key
id_matkul	varchar	4	Foreign key
id_dosen	varchar	4	Foreign key
tanggal	date		
hari	varchar	50	
jam_masuk	time		
jam_keluar	time		
deskripsi	text		
kode_matkul	tinyint	4	
nama_matkul	varchar	100	
nama_dosen	varchar	100	
nama_ruangan	varchar	100	

## 5. Perancangan Database Tabel User

Tabel *User* dirancang untuk menyimpan informasi pengguna dalam sistem manajemen laboratorium. Struktur tabel ini mencakup berbagai atribut yang diperlukan untuk mengidentifikasi dan mengautentikasi pengguna, serta mengelola hak akses mereka. Berikut adalah penjelasan ilmiah mengenai struktur dan fungsi Tabel *User*:

- a. Tabel ini memberikan setiap pengguna identifikasi unik melalui *id\_user* yang bertindak sebagai primary key. Tipe data yang digunakan adalah smallint (4) dengan fitur auto increment.
- b. Atribut '*nama*' dan '*no\_identitas*' berfungsi untuk menyimpan informasi dasar tentang pengguna, sehingga memudahkan pengenalan personal dalam lingkup organisasi.
- c. Atribut '*foto*' berisi lokasi penyimpanan file gambar profil pengguna yang membantu dalam personalisasi tampilan antarmuka pengguna.
- d. Username dan password digunakan sebagai kredensial untuk masuk ke sistem dengan password yang telah dienkripsi guna menjaga keamanan data.

- e. Atribut 'role\_user' dengan tipe data enum digunakan untuk menentukan peran pengguna dalam sistem yang mendukung pengelolaan hak akses secara efektif.

Struktur tabel ini dirancang dengan mempertimbangkan aspek keamanan, efisiensi penyimpanan data, dan fleksibilitas dalam pengelolaan pengguna sistem. Desain tabel mengoptimalkan penggunaan ruang penyimpanan dengan memilih tipe data yang sesuai untuk setiap field. Normalisasi database diterapkan untuk menghindari duplikasi data dan memastikan integritas referensial. Selain itu, struktur ini memungkinkan skalabilitas sistem dengan memfasilitasi penambahan pengguna baru tanpa mengganggu operasional yang ada.

**Tabel 4.8.** Perancangan Tabel *User*

Field	Type	Length	Key
id_user	smallint	4	Primary key (A_I)
nama	varchar	50	
foto	varchar	50	
no_identitas	varchar	50	
username	varchar	100	
password	varchar	100	
role_user	enum	('Admin', 'Kalab', 'Kaprodi', 'Sekprodi', 'Aslab')	

#### 4.2.5 Perancangan Struktur Menu

Perancangan menu sistem informasi manajemen inventaris laboratorium dibagi menjadi 5 tingkatan hak akses sesuai kewenangan pengguna. Setiap pengguna dapat mengakses fitur sesuai peran dan tanggung jawab mereka.

Menu Admin, sebagai tingkat akses tertinggi, memiliki wewenang penuh untuk mengelola semua fitur, termasuk pendanaan, manajemen barang, penjadwalan, dll. Kepala Laboratorium (Kalab) memiliki akses hampir sama dengan Admin, tetapi dengan beberapa pembatasan. Kalab dapat mengelola pendanaan, barang, jadwal, dan data dosen serta mata kuliah. Ketua Program Studi (Kaprodi) dan Sekretaris Program Studi (Sekprodi) fokus pada pengawasan dan monitoring, dengan akses ke fitur pendanaan, manajemen barang, dan dokumentasi. Asisten

Laboratorium (Aslab) memiliki akses untuk mendukung operasional harian termasuk pengelolaan barang dan pemeliharaan peralatan laboratorium.

Struktur menu yang terorganisir ini dirancang secara cermat untuk memudahkan pengelolaan inventaris laboratorium secara sistematis, terkontrol, dan terintegrasi. Setiap tingkatan pengguna memiliki batasan akses yang jelas dan terdefinisi dengan baik yang membantu dalam menjaga keamanan dan integritas data sistem secara menyeluruh. Perancangan menu ini menjadi landasan penting dan fundamental dalam pengembangan antarmuka pengguna yang intuitif serta implementasi berbagai fungsi sistem yang akan dibahas lebih lanjut pada bagian berikutnya. Dengan struktur menu yang terorganisir dan hierarkis ini, diharapkan pengelolaan inventaris laboratorium dapat berjalan lebih efisien, terstruktur, dan terkoordinasi dengan baik antar berbagai tingkatan pengguna. Struktur menu yang dirancang juga mempertimbangkan aspek skalabilitas sehingga memungkinkan penambahan fitur baru di masa mendatang tanpa mengganggu fungsionalitas yang sudah ada. Selain itu, pembagian akses yang jelas membantu mencegah konflik wewenang dan memperjelas alur kerja dalam pengelolaan laboratorium. Implementasi struktur menu yang komprehensif ini diharapkan dapat meningkatkan produktivitas dan efektivitas pengelolaan laboratorium secara keseluruhan. Gambar struktur menu ini dapat dilihat pada Lampiran D.21.

Pengembangan sistem ini menambahkan fitur baru yang ditandai dengan warna hijau pada struktur menu, termasuk fitur Jadwal untuk semua level pengguna (Admin, Kalab, Kaprodi, Sekprodi, dan Aslab) serta fitur Dosen dan Mata Kuliah yang kini dapat diakses oleh Kalab. Penambahan ini bertujuan untuk meningkatkan pengelolaan laboratorium dan efisiensi koordinasi antara pengelola dan dosen.

#### **4.2.6 Perancangan *Interface***

Perancangan *interface* berfungsi untuk menjelaskan tentang desain program sistem informasi manajemen laboratorium yang akan dibangun. Hal ini dilakukan untuk mempermudah pengguna dalam mengetahui proses yang terdapat pada sistem informasi manajemen laboratorium tersebut.

1. Rancangan tampilan halaman utama yang berfungsi sebagai halaman pertama untuk sistem informasi manajemen laboratorium dapat dilihat pada Lampiran D.22.

**Tabel 4.9.** Keterangan Tampilan Halaman Utama

Nomor <i>Callouts</i>	Keterangan
1	Browser toolbar dengan width: 100%, height: 40px, background-color: #FFFFFF
2	Logo berukuran width: 40px, height: 40px, dengan margin-left: 24px pada pojok kiri atas
3	Navigation menu menggunakan font-family: Poppins, font-size: 14px, dengan gap: 32px antar menu
4	Button Login dengan padding: 8px 16px, border-radius: 6px, background-color: #2563EB
5	Container header dengan width: 100%, height: 64px, box-shadow: 0 1px 3px rgba(0,0,0,0.1)
6	Text "Integrated Laboratory" menggunakan font-family: Poppins, font-size: 48px, font-weight: 600, margin-bottom: 16px
7	Text "Management Information System" berwarna #2563EB dengan font-weight: 600
8	Button "Lihat Lebih Lanjut" dengan padding: 12px 24px, background-color: #2563EB
9	Container image dengan width: 560px, height: 400px, border-radius: 8px
10	Container content dengan max-width: 560px dan margin-right: 64px
11	Heading "Sistem Terintegrasi" dengan text-align: center, font-size: 36px, margin: 64px 0 32px
12	Container cards dengan display: grid, grid-template-columns: repeat(2, 1fr), gap: 24px
13	Card pertama dengan padding: 24px, background: white, border-radius: 8px, box-shadow: 0 1px 3px rgba(0,0,0,0.1)
14	Text content pada card dengan font-size: 14px, line-height: 1.6, color: #374151
15	Card kedua dengan styling sama seperti card pertama
16	Card ketiga dengan styling sama seperti card pertama
17	Section FAQ dengan max-width: 800px, margin: 64px auto
18	Container question dengan padding: 24px, border-bottom: 1px solid #E5E7EB

**Tabel 4.9** Keterangan Tampilan Halaman Utama (Tabel lanjutan...)

<b>Nomor Callouts</b>	<b>Keterangan</b>
19	Footer section dengan background: #1E293B, padding: 64px 24px
20	Footer links dengan display: flex, gap: 32px, margin-top: 32px
21	Copyright text dengan font-size: 14px, color: #9CA3AF

2. Rancangan tampilan pilih aplikasi untuk login dapat dilihat pada Lampiran D.23.

**Tabel 4.10.** Keterangan Tampilan Halaman Selamat Datang

<b>Nomor Callouts</b>	<b>Keterangan</b>
1	Text header "Selamat Datang" dengan fonts Poppins, warna font #73879C
2	Text sub-header "Silahkan pilih aplikasi yang akan anda gunakan" dengan fonts Helvetica Neue
3	Container card dengan background color #FFFFFF dan border-radius 8px
4	Link ILMIS dengan icon users dan text "Integrated Laboratory Management Information System"
5	Link LABVIS dengan icon users dan text "Laboratory Visitor Information System"
6	Link LARIS dengan icon users dan text "Laboratory Assistant Registration Information System"

3. Rancangan tampilan kelola jadwal yang berfungsi untuk mengelola jadwal laboratorium dapat dilihat pada Lampiran D.24.

**Tabel 4.11.** Keterangan Tampilan Kelola Jadwal

<b>Nomor Callouts</b>	<b>Keterangan</b>
1	Button "Admin" dengan icon arrow-right di pojok kanan atas
2	Text "Logout Aktif: Beranda / Jadwal" dengan alignment right
3	Profile section dengan nama user "NIM: 1234567" dan foto profil
4	Text header "Jadwal" dengan fonts Helvetica Neue
5	Button "Tambah Data" dengan warna primary (#2563EB)

**Tabel 4.11** Keterangan Tampilan Kelola Jadwal (Tabel lanjutan...)

Nomor Callouts	Keterangan
6	Search input field dengan icon search di sebelah kanan
7	Action buttons (edit, delete, info) di kolom Aksi
8	Data table dengan kolom: No, Hari/Tanggal, Jam, Ruangan, Aksi
9	Sidebar menu dengan icons dan text untuk navigasi sistem

4. Rancangan tampilan tambah jadwal yang berfungsi untuk menambah jadwal laboratorium dapat dilihat pada Lampiran D.25.

**Tabel 4.12.** Keterangan Tampilan Tambah Jadwal

Nomor Callouts	Keterangan
1	Berisi menu navigasi dengan ikon dan teks, menggunakan font-family: Poppins, font-size: 14px. Memiliki warna latar gelap dan ikon menu terorganisir vertikal.
2	Field input untuk memasukkan tanggal dengan placeholder dd/mm/yyyy. Terdapat ikon kalender di sebelah kanan untuk membantu memilih tanggal.
3	Berisi beberapa form input untuk memasukkan data berikut: - Hari: Input teks untuk memilih atau memasukkan hari. - Jam Masuk dan Jam Keluar: Input waktu dengan ikon jam di sebelah kanan. - Nama Ruangan, Nama Mata Kuliah, dan Nama Dosen: Dropdown untuk memilih opsi. - Deskripsi: Area teks kosong untuk menambahkan keterangan lebih lanjut.
4	Dua tombol aksi: - Tombol Kembali: Berwarna merah (#FF5A5F) dengan padding: 12px 24px, border-radius: 6px. - Tombol Simpan: Berwarna biru (#2563EB) dengan padding: 12px 24px, border-radius: 6px.
5	Teks di bagian bawah layar yang menampilkan informasi seperti hak cipta atau detail versi aplikasi. Font-size kecil dengan warna abu-abu muda.

5. Rancangan tampilan edit jadwal yang berfungsi untuk mengedit jadwal laboratorium dapat dilihat pada Lampiran D.26.

**Tabel 4.13.** Tabel Keterangan Tampilan Edit Jadwal

<b>Nomor Callouts</b>	<b>Keterangan</b>
1	Berisi menu navigasi dengan ikon dan teks, menggunakan font-family: Poppins, font-size: 14px. Memiliki warna latar gelap dan ikon menu terorganisir vertikal.
2	Field input untuk memasukkan tanggal dengan placeholder dd/mm/yyyy. Terdapat ikon kalender di sebelah kanan untuk membantu memilih tanggal.
3	Berisi beberapa form input untuk memasukkan data berikut: <ul style="list-style-type: none"><li>- Hari: Input teks untuk memilih atau memasukkan hari.</li><li>- Jam Masuk dan Jam Keluar: Input waktu dengan ikon jam di sebelah kanan.</li><li>- Nama Ruangan, Nama Mata Kuliah, dan Nama Dosen: Dropdown untuk memilih opsi.</li><li>- Deskripsi: Area teks kosong untuk menambahkan keterangan lebih lanjut.</li></ul>
4	Dua tombol aksi: <ul style="list-style-type: none"><li>- Tombol Kembali: Berwarna merah (#FF5A5F) dengan padding: 12px 24px, border-radius: 6px.</li><li>- Tombol Simpan: Berwarna biru (#2563EB) dengan padding: 12px 24px, border-radius: 6px.</li></ul>
5	Teks di bagian bawah layar yang menampilkan informasi seperti hak cipta atau detail versi aplikasi. Font-size kecil dengan warna abu-abu muda.

## **BAB 5**

### **IMPLEMENTASI DAN PENGUJIAN**

#### **5.1 Implementasi Sistem**

Implementasi sistem merupakan tahap dimana sistem yang telah dirancang akan dijalankan. Implementasi sistem ini meliputi implementasi *database*, implementasi *routes*, implementasi *model*, implementasi *view*, dan implementasi *controller*. Implementasi sistem ini dilakukan dengan menggabungkan 2 *framework* yaitu Laravel dan Codeigniter.

##### **5.1.1 Batasan Implementasi**

Batasan implementasi Sistem Informasi Inventaris Laboratorium (SITARIS) dalam penelitian untuk Kerja Praktek ini adalah:

1. Sistem yang dibangun memiliki *platform* berbasis *Web*.
2. Sistem yang dibangun memiliki hak akses seperti Admin, Kalab, Kaprodi, Sekprodi, dan Aslab.
3. Menggunakan bahasa pemrograman PHP dengan *Framework* CodeIgniter dan *Database* MariaDB/PHPMyadmin.
4. Sistem dapat menampilkan data barang, pendanaan, dokumentasi, pem-injaman barang, peminjaman ruangan, *maintenance*, pemusnahan barang, fakultas/lembaga, program studi/unit, gedung, ruangan, jadwal, dosen, mata kuliah dan pengguna.

##### **5.1.2 Implementasi Perangkat Keras (*Hardware*)**

Implementasi pada lingkungan *hardware* adalah implementasi pada perangkat keras yang digunakan untuk menjalankan sistem informasi manajemen laboratorium. Implementasi *hardware* yang digunakan dapat dilihat pada Tabel 5.1.

Minimum kebutuhan pada implementasi *hardware* untuk menjalankan sistem informasi manajemen laboratorium adalah spesifikasi perangkat keras yang harus terpenuhi agar sistem dapat beroperasi secara optimal. Tabel 5.1. menyajikan daftar rinci dari komponen perangkat keras yang diperlukan dan spesifikasinya yang mencakup prosesor, RAM, Hardisk, Monitor, dan perangkat masukan yang harus memenuhi standar minimum agar sistem berfungsi dengan baik.

Pemilihan spesifikasi hardware yang tepat sangat penting untuk memastikan performa sistem yang optimal. Prosesor dengan kecepatan yang memadai diperlukan untuk menangani beban komputasi sistem. Kapasitas RAM yang cukup akan mempengaruhi kecepatan akses data dan kelancaran operasi sistem.

Penyimpanan hardisk yang memadai diperlukan untuk menyimpan semua data dan aplikasi sistem. Monitor dengan resolusi yang baik akan meningkatkan pengalaman pengguna dalam berinteraksi dengan antarmuka sistem. Perangkat masukan seperti keyboard dan mouse yang ergonomis akan mendukung efisiensi kerja pengguna.

**Tabel 5.1.** Spesifikasi Perangkat Keras (*Hardware*)

No	Komponen <i>Hardware</i>	Spesifikasi
1	Processor	Intel ® CoreTM i3-4160, 3.60GHz
2	Memory (RAM)	2 GB
3	Hardisk (HDD)	1 TB
4	LCD	Lenovo 17"

### 5.1.3 Implementasi Perangkat Lunak (*Software*)

Implementasi pada lingkungan *software* adalah implementasi pada perangkat lunak yang digunakan untuk menjalankan sistem informasi inventaris laboratorium. Implementasi *software* yang digunakan dapat dilihat pada Tabel 5.2.

**Tabel 5.2.** Spesifikasi Perangkat Lunak (*Software*)

No	Komponen <i>Software</i>	Spesifikasi
1	Sistem Operasi	Windows 7, 8, 10, dan 11
2	Browser	Google Chrome dan Mozilla Firefox
3	Bahasa Pemrograman	PHP dan Javascript
4	Web Database	MariaDB
5	Framework	CodeIgniter 4

### 5.1.4 Implementasi *Database*

Pada implementasi *database* ini nama yang digunakan adalah "man\_lab". Implementasi *database* ini terdapat beberapa tabel yang akan digunakan dalam sistem. Pembuatan *database* dilakukan dengan menggunakan *database* MariaDB. Tabel-tabel ini akan digunakan untuk menyimpan data-data yang diperlukan dalam sistem. Berikut adalah struktur tabel yang telah diintegrasikan dan digunakan dalam sistem ini dapat dilihat pada Lampiran D.27. Pada pengembangan kali ini dilakukan penambahan tabel baru yaitu tabel "jadwal", "dosen", "mata kuliah", dan tabel "biodatas", "users", "kelola pendaftarans", "kelola ujians", "postingans", "soals", "jawabans", dan "wawancaras" sebagai tabel tambahan untuk integrasi sistem.

1. Tabel Dosen dirancang untuk menyimpan informasi penting terkait dosen dalam sebuah sistem informasi. Kolom `id_dosen` berfungsi sebagai kunci utama (*primary key*) yang memastikan setiap data dosen bersifat unik dan tidak terjadi duplikasi. Kolom `nama_dosen` digunakan untuk mencatat nama lengkap dosen, sedangkan `nip_dosen` menyimpan Nomor Induk Pegawai (NIP) sebagai identitas resmi dosen, jika tersedia. Untuk mencatat jenis kelamin, digunakan kolom `jenis_kelamin` dengan opsi seperti "Laki-laki" atau "Perempuan". Informasi kontak dosen dicatat melalui kolom `email_dosen` dan `no_hp` yang masing-masing menyimpan alamat email serta nomor handphone. Selain itu, kolom `nidn` digunakan untuk menyimpan Nomor Induk Dosen Nasional (NIDN) sebagai identitas unik dosen di tingkat nasional. Semua kolom ini dirancang untuk memastikan data dosen tersimpan secara lengkap, terstruktur, dan mudah diakses. Tampilan dapat dilihat pada Gambar 5.1.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<code>id_dosen</code>	tinyint(4)			No	None		AUTO_INCREMENT	Change  Drop  More
2	<code>nama_dosen</code>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
3	<code>nip_dosen</code>	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
4	<code>jenis_kelamin</code>	enum('Laki-laki', 'Perempuan')	utf8mb4_general_ci		No	None			Change  Drop  More
5	<code>email_dosen</code>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
6	<code>nidn</code>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
7	<code>no_hp</code>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
8	<code>deleted_at</code>	datetime			Yes	NULL			Change  Drop  More
9	<code>updated_at</code>	datetime			Yes	NULL			Change  Drop  More
10	<code>created_at</code>	datetime			Yes	NULL			Change  Drop  More

**Gambar 5.1.** Tampilan *Database* Tabel Dosen

2. Tabel Matkul digunakan untuk menyimpan informasi mengenai mata kuliah dalam sistem informasi akademik. Kolom `id_matkul` berfungsi sebagai kunci utama (*primary key*) untuk mengidentifikasi setiap mata kuliah secara unik dan mencegah terjadinya duplikasi data. Kolom `kode_matkul` menyimpan kode unik mata kuliah sebagai referensi singkat. Nama mata kuliah dicatat dalam kolom `nama_matkul` untuk memberikan informasi deskriptif terkait mata kuliah tersebut. Kolom `sks` digunakan untuk mencatat jumlah Sistem Kredit Semester (SKS) dari mata kuliah, sedangkan `semester` mencatat pada semester berapa mata kuliah tersebut ditawarkan. Selain itu, kolom `jenis_matkul` digunakan untuk mengklasifikasikan jenis mata kuliah, misalnya wajib dan pilihan. Struktur tabel ini dirancang untuk memastikan data terkait mata kuliah tersimpan secara terorganisasi dan mudah diakses sesuai kebutuhan. Tampilan dapat dilihat pada Gambar 5.2.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id_matkul</b> 📚	tinyint(4)			No	None		AUTO_INCREMENT	Change  Drop  More
2	<b>kode_matkul</b>	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
3	<b>nama_matkul</b>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
4	<b>sks</b>	tinyint(4)			No	None			Change  Drop  More
5	<b>semester</b>	tinyint(4)			Yes	NULL			Change  Drop  More
6	<b>jenis_matkul</b>	enum('Wajib', 'Pilihan')	utf8mb4_general_ci		No	None			Change  Drop  More
7	<b>created_at</b>	datetime			Yes	NULL			Change  Drop  More
8	<b>updated_at</b>	datetime			Yes	NULL			Change  Drop  More
9	<b>deleted_at</b>	datetime			Yes	NULL			Change  Drop  More

**Gambar 5.2.** Tampilan Database Tabel Matkul

3. Tabel Jadwal terdiri dari kolom `id_jadwal` yang menjadi kunci utama dari tabel tersebut yang digunakan sebagai penanda agar tidak terjadi duplikasi data, `id_ruangan` digunakan untuk menunjukkan ruangan yang digunakan, `id_matkul` digunakan untuk menunjukkan mata kuliah yang digunakan, `id_dosen` digunakan untuk menunjukkan dosen yang mengajar, `tanggal` digunakan untuk menunjukkan tanggal pelaksanaan, `hari` digunakan untuk menunjukkan hari laboratorium digunakan, `jam_masuk` dan `jam_keluar` digunakan untuk menunjukkan jam digunakannya laboratorium, dan `deskripsi` digunakan untuk menunjukkan deskripsi dari jadwal tersebut. Tampilan dapat dilihat pada Gambar 5.3.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id_jadwal</b> 📚	tinyint(4)			No	None		AUTO_INCREMENT	Change  Drop  More
2	<b>id_ruangan</b>	tinyint(4)			No	None			Change  Drop  More
3	<b>id_matkul</b>	varchar(4)	utf8mb4_general_ci		No	None			Change  Drop  More
4	<b>id_dosen</b>	varchar(4)	utf8mb4_general_ci		No	None			Change  Drop  More
5	<b>tanggal</b>	date			No	None			Change  Drop  More
6	<b>hari</b>	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
7	<b>jam_masuk</b>	time			No	None			Change  Drop  More
8	<b>jam_keluar</b>	time			No	None			Change  Drop  More
9	<b>deskripsi</b>	text	utf8mb4_general_ci		No	None			Change  Drop  More
10	<b>created_at</b>	datetime			Yes	NULL			Change  Drop  More
11	<b>updated_at</b>	datetime			Yes	NULL			Change  Drop  More
12	<b>deleted_at</b>	datetime			Yes	NULL			Change  Drop  More

**Gambar 5.3.** Tampilan Database Tabel Jadwal

4. Tabel Ruangan terdiri dari kolom `id_ruangan` yang menjadi kunci utama dari tabel tersebut yang digunakan sebagai penanda agar tidak terjadi duplikasi data, `id_gedung` menjadi kunci asing dalam tabel ruangan karena nama gedung diperlukan dalam pencatatan data ruangan, `nama_ruangan` adalah kolom yang menyimpan nama ruangan yang di-

catat, deskripsi\_ruangan menjelaskan detail tentang ruangan yang dicatat, gambar\_ruangan merupakan kolom untuk menyimpan data gambar dari ruangan yang dicatat. Tampilan dapat dilihat pada Gambar 5.4.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□ 1	<b>id_ruangan</b> 📄	tinyint(4)			No	None		AUTO_INCREMENT	Change  Drop  More
□ 2	<b>id_gedung</b> 🏠	tinyint(4)			No	None			Change  Drop  More
□ 3	<b>nama_ruangan</b>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
□ 4	<b>deskripsi_ruangan</b>	text	utf8mb4_general_ci		No	None			Change  Drop  More
□ 5	<b>gambar_ruangan</b>	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More

**Gambar 5.4.** Tampilan Database Tabel Ruangan

5. Tabel User terdiri dari kolom `id_user` yang menjadi kunci utama dari tabel tersebut yang digunakan sebagai penanda agar tidak terjadi duplikasi data, `nama` merupakan kolom yang menyimpan nama pengguna, `foto` merupakan kolom untuk menyimpan foto profil pengguna, `no_identitas` merupakan kolom yang digunakan untuk menyimpan data NIM, NIP, atau NIK dari pengguna, `username` merupakan kolom yang digunakan untuk menyimpan username pengguna, `password` merupakan kolom yang digunakan untuk menyimpan password pengguna, `email` merupakan kolom yang digunakan untuk menyimpan email pengguna, `role_user` merupakan kolom yang digunakan untuk menyimpan level akses pengguna. Tampilan dapat dilihat pada Gambar 5.5.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□ 1	<b>id</b> 📄	smallint			No	None		AUTO_INCREMENT	Change  Drop  More
□ 2	<b>nama</b>	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
□ 3	<b>foto</b>	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
□ 4	<b>no_identitas</b> 📄	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
□ 5	<b>password</b>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
□ 6	<b>email</b> 📩	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
□ 7	<b>role_user</b>	enum('Kalab','Admin','Kaprodi','Sekprodi','pen...')	utf8mb4_general_ci		No	pendaftar			Change  Drop  More
□ 8	<b>created_at</b>	timestamp			Yes	NULL			Change  Drop  More
□ 9	<b>updated_at</b>	timestamp			Yes	NULL			Change  Drop  More
□ 10	<b>deleted_at</b>	timestamp			Yes	NULL			Change  Drop  More

**Gambar 5.5.** Tampilan Database Tabel User

### 5.1.5 Implementasi *Routes*

*Routes* dalam konsep *Model View Controller* (MVC) adalah mekanisme yang digunakan untuk mengatur bagaimana permintaan (*requests*) dari pengguna atau *client* akan ditangani oleh aplikasi web. *Routes* menentukan hubungan antara URL yang diminta oleh pengguna dengan controller yang akan menangani

permintaan tersebut (Kelvin dan Amalia, 2022). Dalam pengembangan ini ditambahkan beberapa *routes* untuk menangani *requests* pada sistem informasi.

1. *Routes* dalam implementasi ILMIS pada data dosen dapat dilihat pada Gambar 5.6.



```
1 // Dosen
2 $routes->get('/dosen', 'DosenController::index');
3 $routes->get('/dosen/tambah', 'DosenController::tambah');
4 $routes->post('/dosen/tambah', 'DosenController::insertData');
5 $routes->get('/dosen/edit/(:any)', 'DosenController::edit');
6 $routes->post('/dosen/edit', 'DosenController::updateData');
7 $routes->get('/dosen/hapus/(:any)', 'DosenController::delete');
```

**Gambar 5.6.** Tampilan *Routes* Dosen

2. *Routes* dalam implementasi ILMIS pada data matkul dapat dilihat pada Gambar 5.7.



```
1 // Matkul
2 $routes->get('/matkul', 'MatkulController::index');
3 $routes->get('/matkul/tambah', 'MatkulController::tambah');
4 $routes->post('/matkul/tambah', 'MatkulController::insertData');
5 $routes->get('/matkul/edit/(:any)', 'MatkulController::edit');
6 $routes->post('/matkul/edit', 'MatkulController::updateData');
7 $routes->get('/matkul/hapus/(:any)', 'MatkulController::delete');
```

**Gambar 5.7.** Tampilan *Routes* Matkul

3. *Routes* dalam implementasi ILMIS pada data jadwal dapat dilihat pada Gambar 5.8.

```
1 // Jadwal
2 $routes->get('/jadwal-rsi', 'JadwalController::jadwalRsi');
3 $routes->get('/jadwal-se', 'JadwalController::jadwalSe');
4 $routes->get('/jadwal-int', 'JadwalController::jadwalInt');
5 $routes->get('/jadwal_laboratorium', 'JadwalController::index');
6 $routes->get('/jadwal/tambah', 'JadwalController::tambah');
7 $routes->post('/jadwal/tambah', 'JadwalController::insertData');
8 $routes->get('/jadwal/edit/:any', 'JadwalController::edit');
9 $routes->post('/jadwal/edit', 'JadwalController::updateData');
10 $routes->get('/jadwal/hapus/:any', 'JadwalController::delete');
```

**Gambar 5.8.** Tampilan *Routes* Jadwal

4. *Routes* dalam implementasi ILMIS pada data ruangan dapat dilihat pada Gambar 5.9.

```
1 // Ruangan
2 $routes->get('/ruangan', 'RuanganController::index');
3 $routes->get('/ruangan/tambah', 'RuanganController::tambah');
4 $routes->post('/ruangan/tambah', 'RuanganController::insertData');
5 $routes->get('/ruangan/edit/:any', 'RuanganController::edit');
6 $routes->post('/ruangan/edit', 'RuanganController::updateData');
7 $routes->get('/ruangan/hapus/:any', 'RuanganController::delete');
```

**Gambar 5.9.** Tampilan *Routes* Ruangan

5. *Routes* dalam implementasi ILMIS pada data user dapat dilihat pada Gambar 5.10.



```
1 // User
2 $routes->get('/user', 'UserController::index');
3 $routes->get('/user/profil', 'UserController::profil');
4 $routes->get('/user/tambah', 'UserController::tambah');
5 $routes->post('/user/tambah', 'UserController::insertData');
6 $routes->get('/user/edit/(:num)', 'UserController::edit');
7 $routes->get('/user/ubahdata/(:num)', 'UserController::ubahData');
8 $routes->post('/user/edit', 'UserController::updateData');
9 $routes->post('/user/ubahdata', 'UserController::gantiData');
10 $routes->get('/user/hapus/(:any)', 'UserController::delete');
```

**Gambar 5.10.** Tampilan *Routes User*

### 5.1.6 Implementasi *Model*

Model adalah komponen yang bertanggung jawab untuk mengatur data, aturan bisnis, dan logika aplikasi. Ini merupakan representasi dari data dalam aplikasi. Model mengelola semua operasi data, seperti pengambilan, pembaruan, dan penyimpanan data (Firdaus dan Irfan, 2020). Model-model ini akan diatur pada folder app/model pada *Framework CodeIgniter* dapat dilihat pada Lampiran D.28.

### 5.1.7 Implementasi *View*

*View* adalah komponen yang menampilkan antarmuka pengguna dan menampilkan data dari Model. *View* mengamati perubahan pada Model dan *Controller*, dan diperbarui sesuai keadaan terkini. Penggunaan *View* memisahkan tugas penyajian dan manajemen data dalam aplikasi yang memberikan fleksibilitas dan pemeliharaan yang lebih baik (Firdaus dan Irfan, 2020). *View* ini akan diatur pada folder app/view pada *framework CodeIgniter* dapat dilihat pada Lampiran D.29.

### 5.1.8 Implementasi *Controller*

*Controller* adalah komponen yang bertanggung jawab untuk mengatur logika pengendalian atau interaksi antara Model (data), *View* (tampilan), dan pengguna (Rahman dan Ratna, 2018). *Controller* ini akan diatur pada folder app/controller pada *framework CodeIgniter* dapat dilihat pada Lampiran D.30.

### 5.1.9 Hasil Implementasi

Beberapa permasalahan yang teridentifikasi meliputi kesalahan dalam pembuatan kode barang (Lampiran B), disfungsi fitur peminjaman barang dan ruangan (Lampiran B), serta ketidaksesuaian format laporan akhir dengan kebutuhan kepala laboratorium (Lampiran B). Kesalahan dalam pembuatan kode barang menyeb-

babkan ketidakakuratan dalam pencatatan dan pelacakan inventaris yang dapat mengakibatkan kesulitan dalam manajemen barang. Disfungsi fitur peminjaman barang dan ruangan menghambat proses peminjaman yang seharusnya berjalan lancar, sehingga pengguna mengalami kesulitan dalam meminjam barang dan ruangan yang dibutuhkan. Ketidaksesuaian format laporan akhir dengan kebutuhan kepala laboratorium menyebabkan informasi yang disampaikan tidak sesuai dengan yang diharapkan, sehingga memerlukan penyesuaian agar laporan lebih relevan dan mudah dipahami. Saya sudah memperbaiki kesalahan ini dengan melakukan beberapa perubahan dan penyesuaian pada sistem. Hasil dari perbaikan tersebut akan saya tampilkan untuk menunjukkan peningkatan yang telah dicapai.

1. Tampilan kode barang yang berfungsi untuk memberikan informasi mengenai kode barang dalam sistem manajemen laboratorium. Halaman ini memberikan gambaran umum tentang pengelolaan kode barang. Dapat dilihat pada Gambar 5.11.

**Gambar 5.11.** Tampilan Kode Barang ILMIS

2. Tampilan pilihan peminjaman yang memungkinkan pengguna memilih jenis peminjaman yang ingin dilakukan. Tampilan ini dirancang untuk memudahkan pengguna dalam menentukan pilihan peminjaman yang sesuai dengan kebutuhan mereka, dengan menyediakan opsi yang jelas dan terstruktur. Dapat dilihat pada Lampiran D.31.
3. Tampilan untuk pengelolaan peminjaman barang yang memudahkan pengguna dalam mengatur dan melihat status peminjaman barang. Tampilan ini menyediakan fitur untuk menambah data peminjaman, serta menampilkan

informasi peminjaman dalam format yang mudah dibaca. Dapat dilihat pada Lampiran D.32.

4. Tampilan untuk pengelolaan peminjaman ruangan yang memungkinkan pengguna menambah data peminjaman ruangan ke dalam sistem. Tampilan ini dilengkapi dengan form input yang intuitif, sehingga pengguna dapat dengan mudah memasukkan detail peminjaman seperti waktu, tanggal, dan deskripsi kegiatan. Dapat dilihat pada Lampiran D.33.
5. Tampilan untuk memilih peminjam yang memungkinkan pengguna menentukan peminjam yang akan melakukan peminjaman. Tampilan ini dilengkapi dengan daftar peminjam antara internal maupun eksternal, sehingga pengguna dapat dengan mudah memilih peminjam yang sesuai. Dapat dilihat pada Lampiran D.34.

Hasil implementasi *interface* berfungsi untuk menunjukkan desain program sistem informasi manajemen laboratorium yang telah dilakukan penambahan. Hal ini dilakukan untuk mempermudah pengguna dalam memahami proses yang terdapat pada sistem informasi manajemen laboratorium tersebut.

1. Tampilan *landing page* yang berfungsi sebagai halaman pertama dari sistem informasi manajemen laboratorium. Halaman ini memberikan gambaran umum dan akses awal ke fitur-fitur sistem, serta menyambut pengguna dengan antarmuka yang ramah dan informatif. Dapat dilihat pada Lampiran D.35.
2. Tampilan pilihan aplikasi untuk *login* yang memungkinkan pengguna memilih aplikasi yang ingin diakses. Tampilan ini dirancang untuk memudahkan pengguna dalam menentukan aplikasi mana yang sesuai dengan kebutuhan mereka, dengan menyediakan opsi yang jelas dan terstruktur. Dapat dilihat pada Lampiran D.36.
3. Tampilan kalender laboratorium yang menyajikan jadwal kegiatan laboratorium secara visual dan interaktif. Tampilan ini memungkinkan pengguna untuk melihat jadwal berdasarkan periode waktu tertentu, dengan fitur navigasi bulan dan tahun. Pengguna dapat dengan mudah menambahkan, mengedit, atau menghapus jadwal langsung melalui antarmuka kalender yang responsif. Dapat dilihat pada Lampiran D.37.
4. Tampilan untuk mengelola jadwal laboratorium yang memudahkan pengguna dalam mengatur dan melihat jadwal kegiatan laboratorium. Tampilan ini menyediakan fitur untuk menambah, mengedit, dan menghapus jadwal serta menampilkan jadwal dalam format yang mudah dibaca. Dapat dilihat

pada Lampiran D.38 sampai Lampiran D.41.

5. Tampilan untuk menambah jadwal laboratorium yang memungkinkan pengguna menambahkan jadwal baru ke dalam sistem. Tampilan ini dilengkapi dengan form input yang intuitif, sehingga pengguna dapat dengan mudah memasukkan detail jadwal seperti waktu, tanggal, dan deskripsi kegiatan. Dapat dilihat pada Lampiran D.42.
6. Tampilan untuk mengedit jadwal laboratorium yang memudahkan pengguna dalam melakukan perubahan pada jadwal yang sudah ada. Tampilan ini menyediakan fitur untuk memperbarui informasi jadwal dengan cepat dan efisien, serta memastikan bahwa semua perubahan tersimpan dengan baik. Dapat dilihat pada Lampiran D.43.
7. Tampilan untuk mencetak jadwal laboratorium dalam format PDF yang memudahkan pengguna untuk menyimpan dan membagikan jadwal dalam bentuk dokumen digital. Tampilan ini menyediakan fitur untuk mengonversi jadwal yang ada menjadi file PDF dengan cepat dan efisien. Dapat dilihat pada Lampiran D.44.
8. Tampilan untuk mengelola mata kuliah laboratorium yang memungkinkan pengguna mengatur mata kuliah yang terkait dengan laboratorium. Tampilan ini menampilkan daftar mata kuliah yang dapat diakses dan dikelola, serta menyediakan opsi untuk menambah atau menghapus mata kuliah sesuai kebutuhan. Dapat dilihat pada Lampiran D.45.
9. Tampilan untuk menambah mata kuliah yang memudahkan pengguna dalam menambahkan mata kuliah baru ke dalam sistem. Tampilan ini menyediakan form input yang lengkap untuk memasukkan informasi mata kuliah seperti nama, kode, dan deskripsi. Dapat dilihat pada Lampiran D.46.
10. Tampilan untuk mengedit mata kuliah laboratorium yang memungkinkan pengguna melakukan perubahan pada mata kuliah yang sudah ada. Tampilan ini dirancang untuk memudahkan pengguna dalam memperbarui informasi mata kuliah dengan cepat dan akurat. Dapat dilihat pada Lampiran D.47.
11. Tampilan untuk mengelola dosen laboratorium yang memudahkan pengguna dalam mengatur data dosen yang terkait dengan laboratorium. Tampilan ini menampilkan daftar dosen yang terdaftar dan menyediakan fitur untuk menambah, mengedit, atau menghapus data dosen. Dapat dilihat pada Lampiran D.48.
12. Tampilan untuk menambah dosen laboratorium yang memungkinkan peng-

guna menambahkan data dosen baru ke dalam sistem. Tampilan ini dilengkapi dengan form input yang memudahkan pengguna dalam memasukkan informasi dosen seperti nama, NIDN, dan bidang keahlian. Dapat dilihat pada Lampiran D.49.

13. Tampilan untuk mengedit dosen laboratorium yang memudahkan pengguna dalam melakukan perubahan pada data dosen yang sudah ada. Tampilan ini menyediakan fitur untuk memperbarui informasi dosen dengan mudah dan memastikan bahwa data yang tersimpan selalu akurat dan terkini. Dapat dilihat pada Lampiran D.50.

## 5.2 Pengujian Sistem

Metode *Black Box* digunakan untuk menguji fitur-fitur yang ada pada sistem. Pengujian ini dilakukan untuk memastikan bahwa setiap fungsi dalam sistem berjalan sesuai dengan spesifikasi yang telah ditentukan. Metode ini hanya berfokus pada pengujian keluaran sistem berdasarkan masukan tertentu, tanpa memperhatikan proses internal atau struktur kode.

Pengujian *Black Box* sangat penting untuk memverifikasi fungsionalitas sistem dari perspektif pengguna akhir. Metode ini menguji apakah sistem dapat menghasilkan output yang benar ketika diberikan input tertentu. Pengujian ini juga membantu mengidentifikasi kesalahan atau bug yang mungkin terjadi pada antarmuka pengguna. Selain itu, pengujian ini memastikan bahwa sistem dapat menangani berbagai jenis input dengan benar dan memberikan respon yang sesuai.

Pada pengujian Metode *Black Box*, peneliti melakukan pengujian terhadap fitur-fitur yang ada pada sistem. Pengujian ini dilakukan oleh peneliti dan beberapa pengguna seperti Aslab dan Admin untuk memastikan bahwa fitur-fitur yang ada pada sistem berjalan dengan baik. Setiap skenario pengujian dirancang untuk mencakup berbagai kasus penggunaan yang mungkin terjadi dalam sistem. Pengujian ini juga mencakup kasus-kasus batas untuk memastikan sistem dapat menangani input yang tidak biasa atau ekstrim.

Hasil pengujian kemudian dianalisis untuk menentukan apakah sistem memenuhi persyaratan fungsional yang telah ditetapkan. Jika ditemukan ketidaksesuaian, tim pengembang akan melakukan perbaikan sebelum sistem diluncurkan. Proses perbaikan ini dilakukan secara iteratif hingga semua masalah teratasi dan sistem memenuhi kriteria yang ditetapkan. Pengujian ini dapat dilihat pada Tabel 5.3 yang merangkum semua skenario pengujian yang telah dilakukan beserta hasilnya.

Keuntungan utama dari metode *Black Box* adalah kemampuannya untuk

menguji sistem dari sudut pandang pengguna tanpa perlu pengetahuan teknis yang mendalam tentang implementasi sistem. Hal ini memungkinkan pengujian yang lebih objektif dan fokus pada pengalaman pengguna. Selain itu, metode ini juga efektif untuk menguji integrasi antar komponen sistem dan memastikan alur kerja yang lancar.

**Tabel 5.3.** Pengujian *Black Box*

No Fitur	Hasil yang diharapkan	Hasil Pengujian	Status
1 Registrasi	Berhasil melakukan registrasi akun	Admin dapat melakukan registrasi akun untuk Kalab, Kaprodi, Sekprodi, dan Aslab	Berhasil
2 <i>Login</i> Umum	Berhasil melakukan <i>login</i> dan masuk ke dalam sistem	Pengguna dapat melakukan <i>login</i> ke dalam sistem	Berhasil
3 <i>Login</i> ke SITARIS	Berhasil masuk ke dalam sistem SITARIS	Pengguna dapat melakukan <i>login</i> ke dalam sistem SITARIS	Berhasil
4 <i>Login</i> ke LAB-VIS	Berhasil masuk ke dalam sistem LABVIS	Pengguna dapat melakukan <i>login</i> ke dalam sistem LAB-VIS	Berhasil
5 <i>Login</i> ke LARIS	Berhasil masuk ke dalam sistem LARIS	Pengguna dapat melakukan <i>login</i> ke dalam sistem LARIS	Berhasil
6 Profil	Berhasil melakukan perubahan informasi pribadi	Pengguna dapat melakukan perubahan informasi pribadi	Berhasil
7 Tambah Jadwal	Berhasil menambah Jadwal	Pengguna berhasil menambah Jadwal	Berhasil
8 Detail Jadwal	Berhasil melihat detail informasi jadwal	Pengguna dapat melihat seluruh informasi jadwal	Berhasil
9 Edit Jadwal	Berhasil melakukan perubahan status seleksi administrasi jadwal	Pengguna dapat melakukan perubahan status seleksi administrasi jadwal	Berhasil
10 Hapus Jadwal	Berhasil menghapus jadwal	Pengguna dapat menghapus jadwal yang sudah terdaftar di sistem	Berhasil
11 Cetak Jadwal	Berhasil mencetak jadwal dalam bentuk PDF	Pengguna berhasil mencetak dan <i>download</i> jadwal dalam bentuk PDF	Berhasil
12 Tambah Dosen	Berhasil menambah Dosen	Pengguna berhasil menambah Dosen	Berhasil

**Tabel 5.3.** Pengujian *Black Box* (Tabel lanjutan...)

No Fitur	Hasil yang diharapkan	Hasil Pengujian	Status
13 Detail Dosen	Berhasil melihat detail informasi Dosen	Pengguna dapat melihat seluruh informasi Dosen	Berhasil
14 <i>Edit</i> Dosen	Berhasil melakukan perubahan status seleksi administrasi Dosen	Pengguna dapat melakukan perubahan status seleksi administrasi Dosen	Berhasil
15 Hapus Dosen	Berhasil menghapus Dosen	Pengguna dapat menghapus Dosen yang sudah terdaftar di sistem	Berhasil
16 Tambah Mata Kuliah	Berhasil menambahkan mata kuliah	Dapat menambahkan mata kuliah	Berhasil
17 <i>Edit</i> Mata Kuliah	Berhasil melakukan perubahan mata kuliah	Dapat melakukan perubahan terhadap mata kuliah	Berhasil
18 Hapus Mata Kuliah	Berhasil menghapus mata kuliah	Dapat menghapus mata kuliah	Berhasil
19 Ubah <i>password</i>	Berhasil melakukan perubahan <i>password</i>	Pengguna dapat melakukan perubahan <i>password</i>	Berhasil
20 <i>Logout</i>	Berhasil melakukan <i>logout</i>	Pengguna dapat keluar dari sistem	Berhasil

## **BAB 6**

### **PENUTUP**

#### **6.1 Kesimpulan**

Berdasarkan hasil penelitian yang dilakukan pada Laboratorium Program Studi Sistem Informasi UIN Suska Riau, maka dapat ditarik kesimpulan yaitu:

1. Hasil analisis penelitian ini berhasil mengidentifikasi kekurangan yang ada pada sistem SITARIS.
2. Hasil pengembangann SITARIS berhasil mengintegrasikan sistem informasi yang berjalan masing-masing menjadi Sistem Manajemen Laboratorium Terintegrasi yang dapat memberikan manfaat yang signifikan bagi pengelolaan laboratorium.

#### **6.2 Saran**

Peneliti menyadari dalam pelaksanaan penelitian dan pembuatan laporan maupun sistem masih terdapat celah dan kekurangan. Berdasarkan hal tersebut peneliti membuka diri untuk menerima saran maupun kritik yang membangun bagi peneliti kedepannya. Adapun saran yang ingin peneliti sampaikan diantaranya:

1. Meningkatkan fitur dan kapabilitas Sistem Manajemen Laboratorium Terintegrasi agar lebih sesuai dengan kebutuhan pengguna dan perkembangan teknologi.
2. Menambahkan fitur *backup* dan *recovery system* guna memastikan keamanan data serta mempermudah pemulihan sistem apabila terjadi gangguan atau kehilangan data.
3. Melakukan penelitian lanjutan tentang efektivitas Metode *Agile Development* dibandingkan dengan metode pengembangan lain dalam konteks sistem manajemen laboratorium.
4. Membuat dokumentasi proses *sprint* dan *iterative development* sebagai panduan praktis bagi pengembang sistem serupa yang menggunakan pendekatan *Agile*.

## DAFTAR PUSTAKA

- Abrantes, B. F. (2020). Governance of academic laboratories and the capabilisation of higher education students. *International Journal of Management in Education*, 14(2), 135–158.
- Al-Saqqa, S., Sawalha, S., dan Abdel-Nabi, H. (2020). Agile software development: Methodologies and trends. *International Journal of Interactive Mobile Technologies*, 14(11).
- Balsamiq. (2024). *Balsamiq*. Retrieved from <https://balsamiq.com/> (Akses Tanggal: 25 Juli 2024)
- Balza, J. S., Cusatis, R., McDonnell, S. M., Basir, M. A., dan Flynn, K. E. (2022). Effective questionnaire design: how to use cognitive interviews to refine questionnaire items. *Journal of neonatal-perinatal medicine*, 15(2), 345–349.
- Bree, R. T., dan Gallagher, G. (2016). Using microsoft excel to code and thematically analyse qualitative data: a simple, cost-effective approach. *All Ireland Journal of Higher Education*, 8(2).
- Carstoiu, D., dan Grigorescu, C. (1995). OOAD Methods for O-O and KBS Development. *IFAC Proceedings Volumes*, 28(24), 263–268. doi: 10.1016/s1474-6670(17)46560-8
- Cowls, J., Tsamados, A., Taddeo, M., dan Floridi, L. (2021). A definition, benchmark and database of ai for social good initiatives. *Nature Machine Intelligence*, 3, 111 - 115. Retrieved from <https://api.semanticscholar.org/> CorpusID:233940459
- Dongapure, A., Choudhari, P., Yawale, P., dan Kawalkar, A. (2024). Good laboratory practices (glp) in an academic institute for science. *Journal of Advanced Zoology*, 45(1).
- Efendi, E., Ramadhani, R. N., Zihad, A., dkk. (2023). Perkembangan sistem informasi pada organisasi dakwah. *Da'watuna: Journal of Communication and Islamic Broadcasting*, 3(2), 938–949.
- Fadllullah, A., Mulyadi, M., Rochaniati, R., dan Nabil, F. M. (2022). Pengembangan sistem informasi manajemen kearsipan surat menyurat berbasis framework codeigniter untuk kph-ktt. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*. Retrieved from <https://api.semanticscholar.org/> CorpusID:249829228
- Feng, M., Li, C., dan McVay, S. (2009). Internal control and management guidance.

- Journal of accounting and economics*, 48(2-3), 190–209.
- Firdaus, N., dan Irfan, D. (2020). Rancang bangun sistem informasi arsip berbasis web menggunakan framework codeigniter. *Voteteknika (Vocational Teknik Elektronika Dan Informatika)*, 8(1), 44–52.
- Hayati, N., Rahayu, S., dan Saputra, T. I. (2021). Sistem informasi pemilihan asisten laboratorium dengan metode weighted product dan weighted sum model. *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, 6(1), 1–8.
- Henard, C., Papadakis, M., Harman, M., Jia, Y., dan Le Traon, Y. (2016). Comparing white-box and black-box test prioritization. Dalam *Proceedings of the 38th international conference on software engineering* (hal. 523–534).
- Hussain, N., Haque, A. U., dan Baloch, A. (2019). Management theories: The contribution of contemporary management theorists in tackling contemporary management challenges. *Yaşar Üniversitesi E-Dergisi*, 14, 156–169.
- Kaehler, B., Grundei, J., Kaehler, B., dan Grundei, J. (2019). The concept of management: In search of a new definition. *HR governance: A theoretical introduction*, 3–26.
- Kawai, H., Ando, K., Maruyama, D., Yamamoto, K., Kiyohara, E., Terui, Y., ... others (2021). Phase ii study of e7777 in japanese patients with relapsed/refractory peripheral and cutaneous t-cell lymphoma. *Cancer science*, 112(6), 2426–2435.
- Kelvin, K., dan Amalia, R. (2022). Sistem informasi inventaris berbasis qr-code dengan metode rapid application development. *Journal of Information System Research (JOSH)*, 4(1), 111–116.
- Kusuma, M. W. A., dan Ahsyar, T. K. (2024). Penerapan website laboratorium sebagai media informasi dengan metode object oriented analysis and design (ooad). *JoMMiT: Jurnal Multi Media dan IT*, 8(1), 7–15.
- Laboratorium Sistem Informasi. (2023). *Website laboratorium sistem informasi uin suska riau*. Retrieved from <https://lab-si.uin-suska.ac.id> (Akses Tanggal: 30 Oktober 2024)
- La Braca, F., dan Kalman, C. S. (2021). Comparison of labatorials and traditional labs: The impacts of instructional scaffolding on the student experience and conceptual understanding. *Physical Review Physics Education Research*, 17(1), 010131.
- Linzhang, W., Jiesong, Y., Xiaofeng, Y., Jun, H., Xuandong, L., dan Guoliang, Z. (2004). Generating test cases from uml activity diagram based on gray-box method. Dalam *11th asia-pacific software engineering conference* (hal. 284–

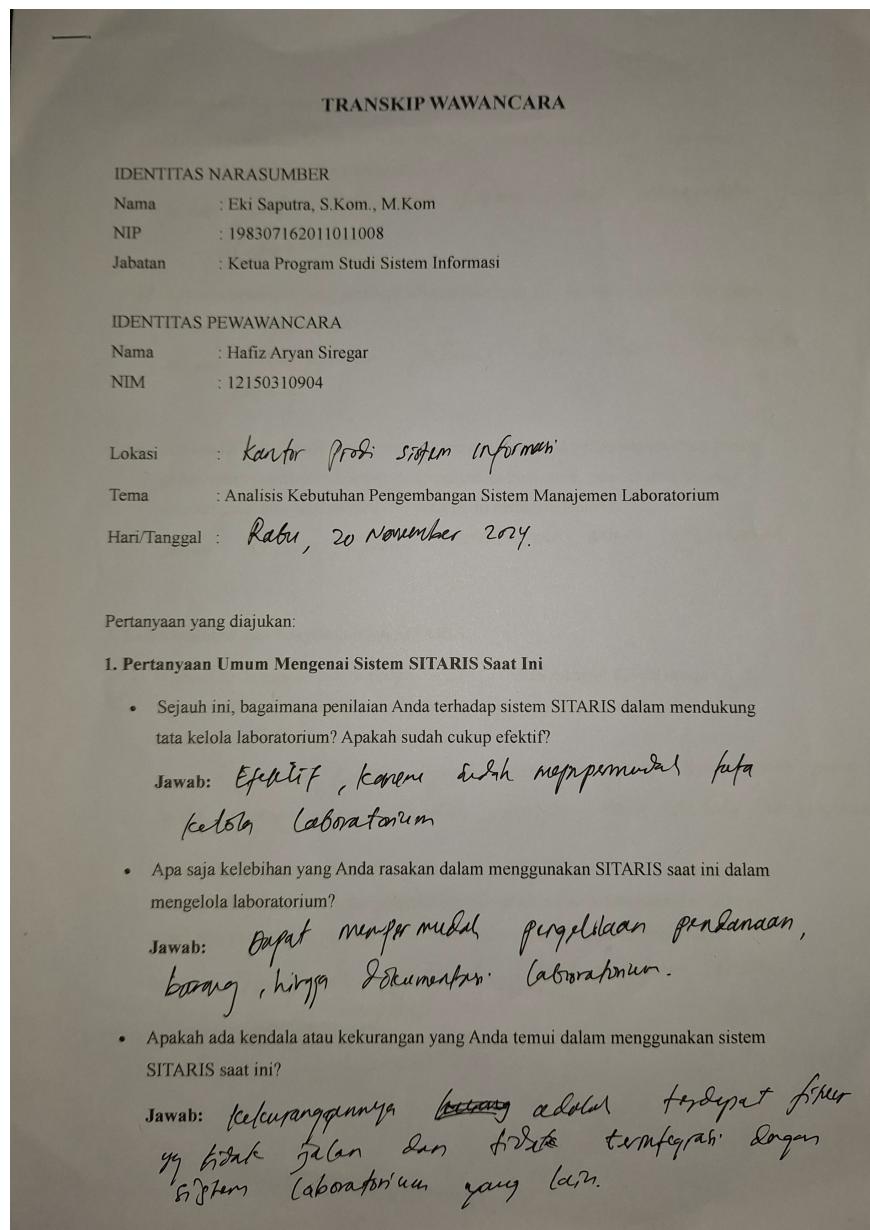
291).

- MariaDB, P. (2024). Mariadb: Sistem manajemen basis data relasional untuk aplikasi web. *Jurnal Teknologi Informasi*.
- Marwah, S., Puspitorini, S., dkk. (2024). Sistem informasi manajemen laboratorium pada jurusan farmasi politeknik kesehatan kemenkes jambi. *FORTECH (Journal of Information Technology)*, 8(1), 13–19.
- Monday, T. U. (2020). Impacts of interview as research instrument of data collection in social sciences. *Journal of Digital Art & Humanities*, 1(1), 15–24.
- Mubarak, A. (2019). Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek. *JIKO (Jurnal Informatika dan Komputer)*, 2(1), 19–25. doi: 10.33387/jiko.v2i1.1052
- Pakpahan, S., Faâ, A., dkk. (2020). Sistem informasi pengelolaan dana desa pada desa hilizoliga berbasis web. *Jurnal Teknik Informatika UNIKA Santo Thomas*, 109–117.
- Prodi Sistem Informasi UIN Suska Riau. (2023). *Website prodi sistem informasi uin suska riau*. Retrieved from <https://sif.uin-suska.ac.id> (Akses Tanggal: 30 Oktober 2024)
- Rahman, F., dan Ratna, S. (2018). Perancangan e-learning berbasis web menggunakan framework codeigniter. *Technologia: Jurnal Ilmiah*, 9(2), 95–100.
- Rihm, S. D., Tan, Y. R., Ang, W., Hofmeister, M., Deng, X., Laksana, M. T., ... others (2024). The digital lab manager: Automating research support. *SLAS technology*, 29(3), 100135.
- Rueda, S., Panach, J. I., dan Distante, D. (2020). Requirements elicitation methods based on interviews in comparison: A family of experiments. *Information and Software Technology*, 126, 106361.
- Sallaby, A. F., dan Kanedi, I. (2020). Perancangan sistem informasi jadwal dokter menggunakan framework codeigniter. *Jurnal Media Infotama*, 16(1).
- Schühly, A. M., dan Schühly, A. M. (2022). What is strategic management and why do we need it: theoretical foundations of strategic management. *Cultural Influences on the Process of Strategic Management: Using Scenario Planning for Decision Making in Multinational Corporations*, 33–135.
- Simanullang, H. G., Silalahi, A. P., dan Manalu, D. R. (2021). Sistem informasi pendaftaran mahasiswa baru menggunakan framework codeigniter dan application programming interface. *Ultima InfoSys: Jurnal Ilmu Sistem Informasi*, 12(1), 67–73.

- SITARIS SI. (2024). *Sistem informasi inventaris prodi sistem informasi uin suska riau*. Retrieved from <https://sitaris.lab-si.uin-suska.ac.id> (Akses Tanggal: 30 Oktober 2024)
- Smith, R., Janicke, H., He, Y., Ferra, F., dan Albakri, A. (2021). The agile incident response for industrial control systems (air4ics) framework. *Computers & Security, 109*, 102398.
- Sweden, I. N., Pemayun, A. A. G. M., Wibawa, K. S., Prayoga, I. K. D. Y., Putra, I. D. M. L., dan Frangginie, N. L. G. M. (2022). Rancang bangun sistem informasi manajemen layanan laboratorium berdasarkan standar iso 9126. *TEMATIK, 9*(2), 108–118.
- Tilley, S., dan Rosenblatt, H. J. (2017). *Systems analysis and design*. Cengage Learning.
- Trelles, S. C. (2021). Agile adoption in information technology departments at research universities.
- Tuunanen, T., Vartiainen, T., Kainulainen, S., dan Ebrahim, M. (2023). Development of an agile requirements risk prioritization method: A design science research study. *Communications of the Association for Information Systems, 52*.
- Tywati, S., dan Irawan, R. (2017). Implementasi framework codeigniter untuk pengembangan website pada dinas perkebunan provinsi kalimantan tengah. *J. SAINTEKOM, 7*(1), 67.
- UIN Suska Riau. (2023). *Uin suska riau*. Retrieved from <https://uin-suska.ac.id> (Akses Tanggal: 30 Oktober 2024)
- Wisnumurti, W., Trimarsiah, Y., dan Faulina, S. T. (2022). Penerapan agile development methodology pada sistem informasi penjualan ecer dan grosir toko kiananti martapura. *JUTIM (Jurnal Teknik Informatika Musirawas), 7*(2), 109–120.
- Zhang, X., Xu, T., Wei, X., Tang, J., dan Ordonez de Pablos, P. (2024). The establishment of transactive memory system in distributed agile teams engaged in ai-related knowledge work. *Journal of Knowledge Management, 28*(2), 381–408.

## LAMPIRAN A

### HASIL WAWANCARA



**Gambar A.1.** Transkip Wawancara Kaprodi

## 2. Kebutuhan dan Tantangan Tata Kelola Laboratorium

- Dalam konteks tata kelola laboratorium, aspek apa saja yang menurut Anda perlu diperbaiki atau lebih diperhatikan?

Jawab: SOP, Pelaporan barang masuk, manajemen jadwal laboratorium

- Apa saja tantangan yang dihadapi laboratorium saat ini yang tidak dapat diselesaikan dengan SITARIS yang ada?

Jawab: Manajemen jadwal dan laporan kerusakan barang

- Apakah SITARIS sudah cukup mengakomodasi semua aktivitas laboratorium seperti pengelolaan pendanaan, barang, peminjaman, dokumentasi, pemusnahan barang, maintenance, atau adakah area lain yang perlu perhatian lebih?

Jawab: Belum, karena masih ada sistem yg terpisah.

## 3. Keberlanjutan dan Pengembangan SITARIS

- Apakah Anda merasa bahwa pengembangan SITARIS menjadi ILMIS dengan menambahkan fitur penjadwalan dapat meningkatkan efisiensi tata kelola laboratorium? Mengapa?

Jawab: Tidak, karena perlu mengintegrasikan sistem lain yang ada agar memperoleh efisiensi tata kelola laboratorium

- Apakah Anda merasa bahwa adanya fitur penjadwalan laboratorium akan membantu meningkatkan koordinasi dan pengelolaan jadwal aktivitas di laboratorium? Jika ya, aspek apa yang paling penting untuk diatur dalam penjadwalan laboratorium?

Jawab: Transaksi, jadwal, keterkaitan labor, ruangan, dll.

- Apakah Anda percaya bahwa pengembangan SITARIS ini akan mempermudah pengelolaan sumber daya laboratorium (seperti alat, ruangan, atau tenaga kerja) dengan cara yang lebih efektif dan terintegrasi?

Jawab: Ya.

#### 4. Penjadwalan Laboratorium

- Bagaimana menurut Anda, apakah sistem penjadwalan kegiatan laboratorium harus terintegrasi dengan sistem yang ada di SITARIS atau perlu dibangun secara terpisah?

Jawab: Harus terintegrasi agar mudah dalam pengelolaannya

- Apakah Anda merasa bahwa jadwal penggunaan laboratorium sering bentrok atau tidak efisien? Jika iya, bagaimana seharusnya sistem dapat menangani hal ini?

Jawab: Jadwal penggunaan laboratorium tidak efisien karena bisa mengakibatkan jadwal secara real time

- Seberapa penting bagi Anda agar sistem penjadwalan laboratorium bisa memperhitungkan ketersediaan ruang dan alat secara otomatis?

Jawab: Cukup penting.

- Dalam hal pengelolaan waktu dan kegiatan laboratorium, fitur penjadwalan apa yang menurut Anda sangat dibutuhkan oleh pengguna laboratorium?

Jawab: Kelola jadwal, matkul, dan dozen.

#### 5. Integrasi dan Kolaborasi Sistem

- Dalam pengembangan ILMIS, bagaimana sebaiknya sistem penjadwalan laboratorium terhubung dengan sistem pengelolaan lainnya, seperti pengelolaan inventaris, peminjaman alat, dan pemeliharaan?

Jawab: Terhubung dengan fitur manajemen agar sistem yg dimuat dapat secara otomatis terintegrasi.

- Apakah pengintegrasian fitur penjadwalan dengan fitur-fitur lain seperti pengelolaan pendanaan dan inventaris akan mempercepat proses pengambilan keputusan dan meningkatkan tata kelola secara keseluruhan?

Jawab: Ya.

#### 6. Pengalaman Pengguna dan Interface Sistem

- Seberapa mudah Anda merasa menggunakan sistem SITARIS saat ini? Apakah ada bagian dari antarmuka (*user interface*) yang menurut Anda perlu perbaikan agar lebih *user-friendly*?

Jawab: Mudah, sistem termasuk user friendly dan mudah dipahami.

- Sejauh mana kemudahan penggunaan dan kesederhanaan menjadi faktor penting dalam penerimaan dan penggunaan sistem ILMIS yang baru di laboratorium?

Jawab: Sangat penting, untuk mempermudah penggunaan

#### 7. Pengelolaan Sumber Daya Laboratorium

- Apa saja aspek pengelolaan sumber daya laboratorium (seperti alat, bahan, ruang) yang menurut Anda harus diprioritaskan dalam pengembangan ILMIS?

Jawab: Keamanan dan integrasi data.

- Apakah pengelolaan alat dan peminjaman alat laboratorium sudah cukup efektif menggunakan SITARIS saat ini? Adakah masalah atau hambatan dalam hal tersebut yang menurut Anda perlu diperbaiki?

Jawab: Peminjaman masih sedikit susah. Karena dan aturan yang ada.

#### 8. Evaluasi dan Keberlanjutan Sistem

- Setelah sistem ILMIS diimplementasikan, bagaimana Anda menilai keberhasilan sistem tersebut dalam menunjang tata kelola laboratorium? Apa indikator atau kriteria yang harus dipenuhi?

Jawab: kemudahan, integrasi, data, pengambilan keputusan.

- Apakah Anda siap melakukan adaptasi terhadap perubahan atau fitur baru yang akan diterapkan dalam sistem ILMIS, terutama dalam hal penjadwalan laboratorium?

Jawab: Ya.

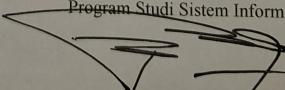
- Apa saja langkah-langkah yang perlu diambil untuk memastikan transisi dari SITARIS ke ILMIS berjalan lancar dan efektif bagi seluruh pengguna laboratorium?

Jawab: Mengoptimalkan akses internet data.

#### 9. Penutup

- Apa harapan Anda terhadap pengembangan sistem ILMIS dalam konteks tata kelola laboratorium? Apakah ada saran lain untuk pengembangan lebih lanjut?

Jawab: Semoga sistem tersebut dapat terintegrasi dengan SITARIS dalam bentuk data office dan jadi satu, mudah dimanajemen, terintegrasi dan tambahan pengalaman barang-barang seperti

Ketua  
Program Studi Sistem Informasi,  
  
Eki Saputra, S.Kom., M.Kom  
NIP. 198307162011011008

## TRANSKIP WAWANCARA

### IDENTITAS NARASUMBER

Nama : Tengku Khairil Ahsyar, S.Kom., M.Kom.  
NIP : 198505202023211020  
Jabatan : Kepala Laboratorium Program Studi Sistem Informasi

### IDENTITAS PEWAWANCARA

Nama : Hafiz Aryan Siregar  
NIM : 12150310904

Lokasi : Ruang Dosen GB Lantai 1

Tema : Analisis Kebutuhan Pengembangan Sistem Manajemen Laboratorium

Hari/Tanggal : Selasa, 12 November 2024

Pertanyaan yang diajukan:

#### 1. Pertanyaan Umum Mengenai Sistem SITARIS Saat Ini

- Sejauh ini, bagaimana penilaian Anda terhadap sistem SITARIS dalam mendukung tata kelola laboratorium? Apakah sudah cukup efektif?

**Jawab:** Efektif, sitaris sangat membantu dan menunjang tata kelola laboratorium menjadi lebih cepat dan tepat.

- Apa saja kelebihan yang Anda rasakan dalam menggunakan SITARIS saat ini dalam mengelola laboratorium?

**Jawab:** Dapat mempercepat pengelolaan keluar masuk barang laboratorium, sehingga proses dokumentasi barang menjadi jelas.

- Apakah ada kendala atau kekurangan yang Anda temui dalam menggunakan sistem SITARIS saat ini?

**Jawab:** Ada, seperti disfungsi fitur saat peminjaman barang laboratorium

## Gambar A.2. Transkip Wawancara Kalab

## 2. Kebutuhan dan Tantangan Tata Kelola Laboratorium

- Dalam konteks tata kelola laboratorium, aspek apa saja yang menurut Anda perlu diperbaiki atau lebih diperhatikan?

**Jawab:** SOP, jadwal pemeliharaan sistem, jadwal penggunaan laboratorium, laporan terkait barang yang ada

- Apa saja tantangan yang dihadapi laboratorium saat ini yang tidak dapat diselesaikan dengan SITARIS yang ada?

**Jawab:** Manajemen jadwal yang belum tersedia di laboratorium dan juga integrasi dengan sistem yang ada agar tidak ada duplikasi data.

- Apakah SITARIS sudah cukup mengakomodasi semua aktivitas laboratorium seperti pengelolaan pendanaan, barang, peminjaman, dokumentasi, pemusnahan barang, maintenance, atau adakah area lain yang perlu perhatian lebih?

**Jawab:** Belum, karena masih ada sistem yang belum terintegrasi

## 3. Keberlanjutan dan Pengembangan SITARIS

- Apakah Anda merasa bahwa pengembangan SITARIS menjadi ILMIS dengan menambahkan fitur penjadwalan dapat meningkatkan efisiensi tata kelola laboratorium? Mengapa?

**Jawab:** Ya, namun perlu ditambahkan intergrasi dengan sistem yang lainnya agar hak tersebut dapat terwujud.

- Apakah Anda merasa bahwa adanya fitur penjadwalan laboratorium akan membantu meningkatkan koordinasi dan pengelolaan jadwal aktivitas di laboratorium? Jika ya, aspek apa yang paling penting untuk diatur dalam penjadwalan laboratorium?

**Jawab:** Keterbukaan jadwal, ketersediaan ruangan labor, asisten, dll.

- Apakah Anda percaya bahwa pengembangan SITARIS ini akan mempermudah pengelolaan sumber daya laboratorium (seperti alat, ruangan, atau tenaga kerja) dengan cara yang lebih efektif dan terintegrasi?

**Jawab:** Ya, karena dengan sistem yang ada proses menjadi lebih cepat karena semua terkomputerisasi

#### **4. Penjadwalan Laboratorium**

- Bagaimana menurut Anda, apakah sistem penjadwalan kegiatan laboratorium harus terintegrasi dengan sistem yang ada di SITARIS atau perlu dibangun secara terpisah?

**Jawab:** Sistem yang ada harus saling terhubung dan terintegrasi secara real-time, agar memudahkan saat proses pengelolaan sistem dan tidak ada duplikasi data.

- Apakah Anda merasa bahwa jadwal penggunaan laboratorium sering bentrok atau tidak efisien? Jika iya, bagaimana seharusnya sistem dapat menangani hal ini?

**Jawab:** Jadwal yang ada saat ini tidak bisa dikatakan efisien, karena jadwal disusun saat diawal sehingga kita tidak mengetahui kondisi saat ini/ tidak real-time

- Seberapa penting bagi Anda agar sistem penjadwalan laboratorium bisa memperhitungkan ketersediaan ruang dan alat secara otomatis?

**Jawab:** Penting, agar tidak terjadi bentrok jadwal/kekosongan barang

- Dalam hal pengelolaan waktu dan kegiatan laboratorium, fitur penjadwalan apa yang menurut Anda sangat dibutuhkan oleh pengguna laboratorium?

**Jawab:** Kelola Jadwal ruangan, dosen, mata kuliah, dan kelas.

#### **5. Integrasi dan Kolaborasi Sistem**

- Dalam pengembangan ILMIS, bagaimana sebaiknya sistem penjadwalan laboratorium terhubung dengan sistem pengelolaan lainnya, seperti pengelolaan inventaris, peminjaman alat, dan pemeliharaan?

**Jawab:** terintegrasi satu sama lain agar fitur ruangan dapat berjalan secara otomatis

- Apakah pengintegrasian fitur penjadwalan dengan fitur-fitur lain seperti pengelolaan pendanaan dan inventaris akan mempercepat proses pengambilan keputusan dan meningkatkan tata kelola secara keseluruhan?

**Jawab:** Ya, karena dengan begitu pengambilan keputusan sudah dapat diperoleh dari data yang ada dan tidak perlu lagi untuk cek secara berulang

#### **6. Pengalaman Pengguna dan Interface Sistem**

- Seberapa mudah Anda merasa menggunakan sistem SITARIS saat ini? Apakah ada bagian dari antarmuka (*user interface*) yang menurut Anda perlu perbaikan agar lebih *user-friendly*?

**Jawab:** Tidak, karena sejauh ini sitaris sudah cukup user-friendly

- Sejauh mana kemudahan penggunaan dan kesederhanaan menjadi faktor penting dalam penerimaan dan penggunaan sistem ILMIS yang baru di laboratorium?

**Jawab:** Sangat penting agar mempercepat proses pengambilan keputusan

#### 7. Pengelolaan Sumber Daya Laboratorium

- Apa saja aspek pengelolaan sumber daya laboratorium (seperti alat, bahan, ruang) yang menurut Anda harus diprioritaskan dalam pengembangan ILMIS?

**Jawab:** Keamanan

- Apakah pengelolaan alat dan peminjaman alat laboratorium sudah cukup efektif menggunakan SITARIS saat ini? Adakah masalah atau hambatan dalam hal tersebut yang menurut Anda perlu diperbaiki?

**Jawab:** Pada fitur peminjaman masih terdapat disfungsi fitur

#### 8. Evaluasi dan Keberlanjutan Sistem

- Setelah sistem ILMIS diimplementasikan, bagaimana Anda menilai keberhasilan sistem tersebut dalam menunjang tata kelola laboratorium? Apa indikator atau kriteria yang harus dipenuhi?

**Jawab:** Kemudahan dan keamanan data

- Apakah Anda siap melakukan adaptasi terhadap perubahan atau fitur baru yang akan diterapkan dalam sistem ILMIS, terutama dalam hal penjadwalan laboratorium?

**Jawab:** Tentu

- Apa saja langkah-langkah yang perlu diambil untuk memastikan transisi dari SITARIS ke ILMIS berjalan lancar dan efektif bagi seluruh pengguna laboratorium?

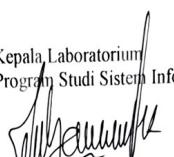
**Jawab:** Memperhatikan aspek database

## 9. Penutup

- Apa harapan Anda terhadap pengembangan sistem ILMIS dalam konteks tata kelola laboratorium? Apakah ada saran lain untuk pengembangan lebih lanjut?

Jawab:

Kepala, Laboratorium  
Program Studi Sistem Informasi,

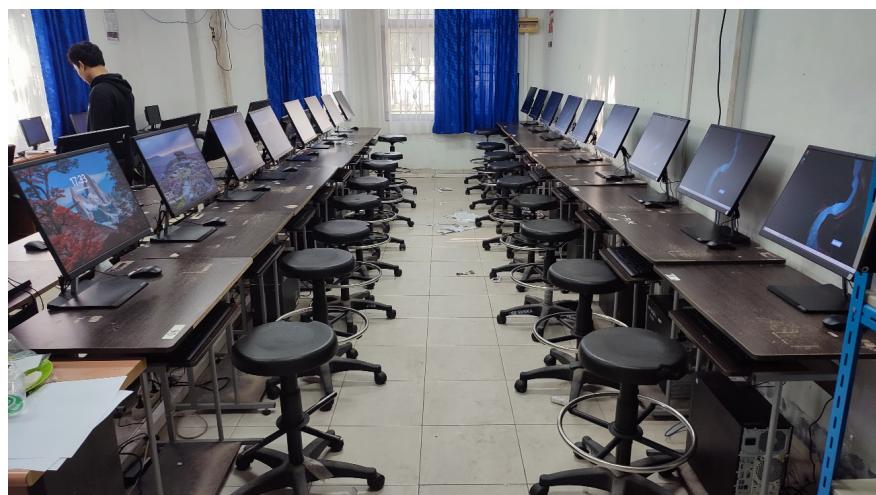
  
Tengku Khairil Ahsyar, S.Kom., M.Kom.  
NIP 198505202023211020

## **LAMPIRAN B**

### **HASIL OBSERVASI**



**Gambar B.1.** Laboratorium Rekayasa Sistem Informasi



**Gambar B.2.** Laboratorium Internet



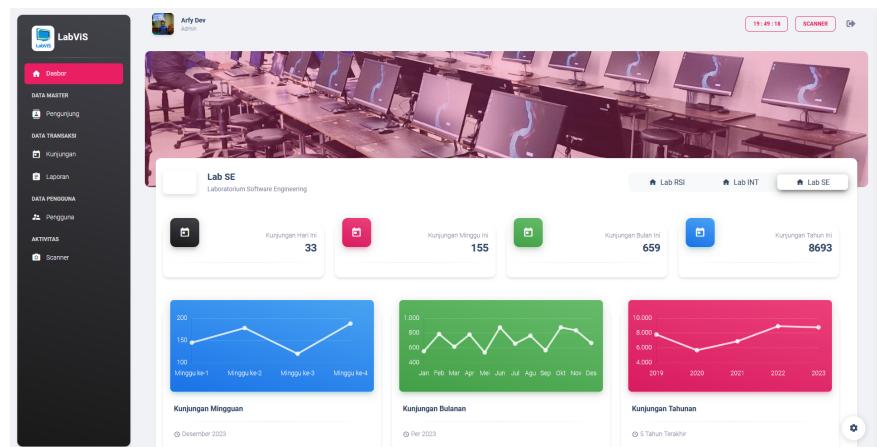
**Gambar B.3.** Laboratorium Internet



**Gambar B.4.** Kegiatan Laboratorium



**Gambar B.5.** Website Laboratorium Program Studi Sistem Informasi



**Gambar B.6.** Laboratory Visitor System (LABVIS)



**Gambar B.7.** Laboratory Assistant Registration Information System (LARIS)

**Ajukan Peminjaman Barang**

Data Peminjaman Internal

Program Studi / Unit \*

Nama Organisasi \*

Nama Peminjam \*

Email Peminjam \*

Nomor Telepon \*

**Selanjutnya** Privacy - Term

**Gambar B.8. Disfungsi Fitur Peminjaman Barang**



**Gambar B.9. Disfungsi Fitur Peminjaman Ruangan**

**Tambah Barang**

Jenis Pendaftaran \*

Ruang \*

Tahun Barang \*

Kategori Barang \*

Subkategori \*

Merk Barang \*

Kondisi \*

Spesifikasi Barang

hafizaryan

Copyright © 2024 Sistem Informasi Inventaris Laboratorium Sistem Informasi

**Gambar B.10. Disfungsi Pembuatan Kode Barang**

**LAMPIRAN C**  
**DOKUMENTASI WAWANCARA**

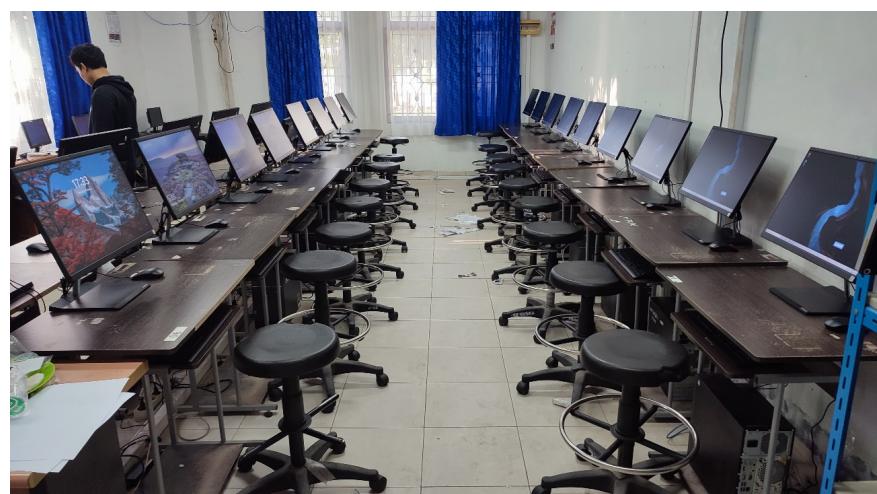


**Gambar C.1. Bukti Wawancara**

**LAMPIRAN D**  
**DOKUMENTASI PERANCANGAN**



**Gambar D.1.** Laboratorium Rekayasa Sistem Informasi



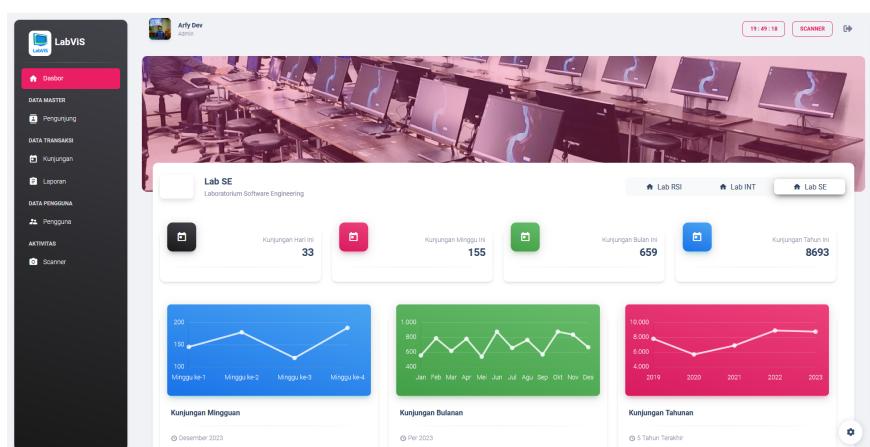
**Gambar D.2.** Laboratorium Internet



**Gambar D.3.** Laboratorium *Software Engineering*

The image consists of two side-by-side screenshots. On the left is the homepage of the 'Laboratorium Program Studi Sistem Informasi' website. It has a dark blue header with a globe icon, followed by the title 'Laboratorium Program Studi Sistem Informasi'. Below the title is a subtext: 'Selamat datang di Laboratorium Program Studi Sistem Informasi Fakultas Sains dan Teknologi UIN Suska Riau'. A blue button labeled 'Lihat Lebih Lanjut' is at the bottom. On the right is a screenshot of a computer monitor displaying the LABVIS system. The screen shows a live video feed of the lab, navigation menu items like 'Beranda', 'Profil', 'Laboratorium', 'Rilis Media', 'Pengumuman', and 'Kontak', and four cards with visitor statistics: 'Kunjungan Hari Ini' (33), 'Kunjungan Minggu Ini' (155), 'Kunjungan Bulan Ini' (659), and 'Kunjungan Tahun Ini' (8693). Below these are three line charts: 'Kunjungan Mingguan' (weekly visitors), 'Kunjungan Bulanan' (monthly visitors), and 'Kunjungan Tahunan' (annual visitors).

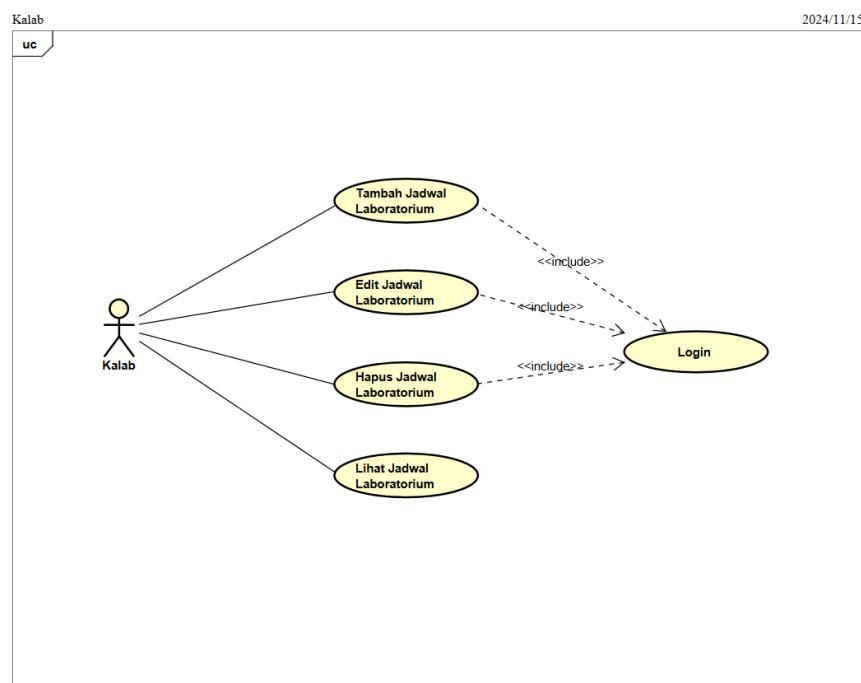
**Gambar D.4.** Website Laboratorium Program Studi Sistem Informasi



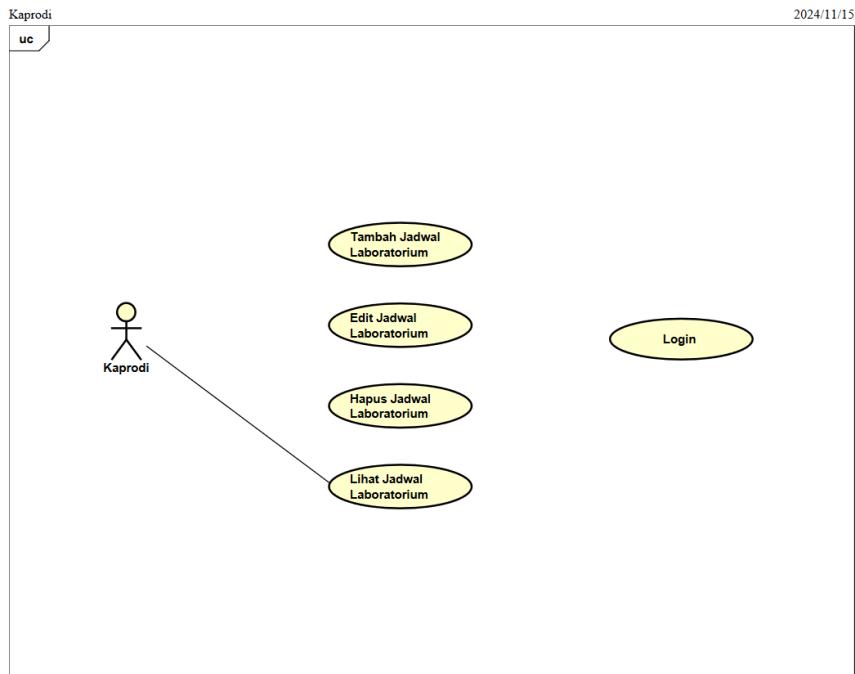
**Gambar D.5.** *Laboratory Visitor System (LABVIS)*



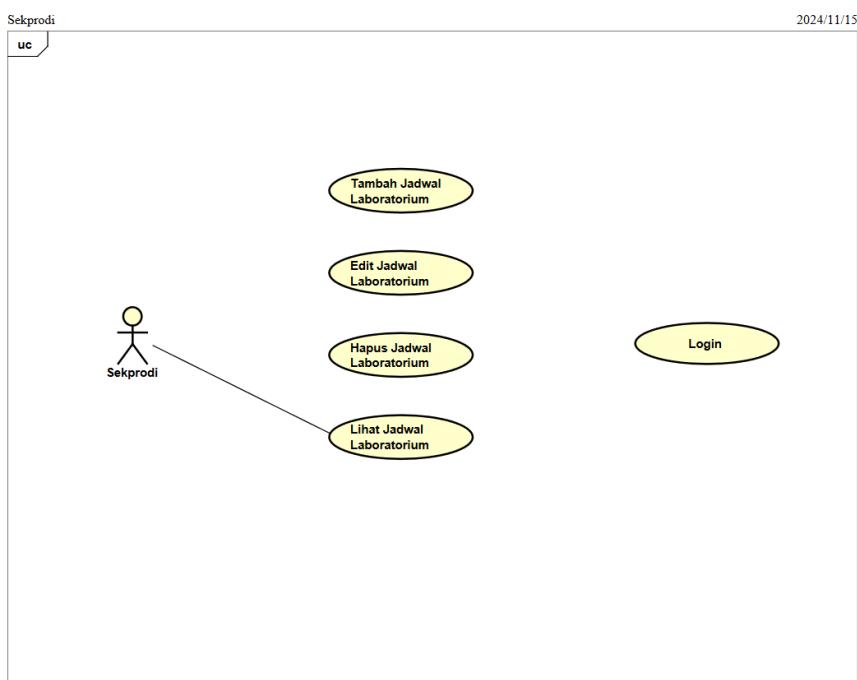
**Gambar D.6.** *Laboratory Assistant Registration Information System (LARIS)*



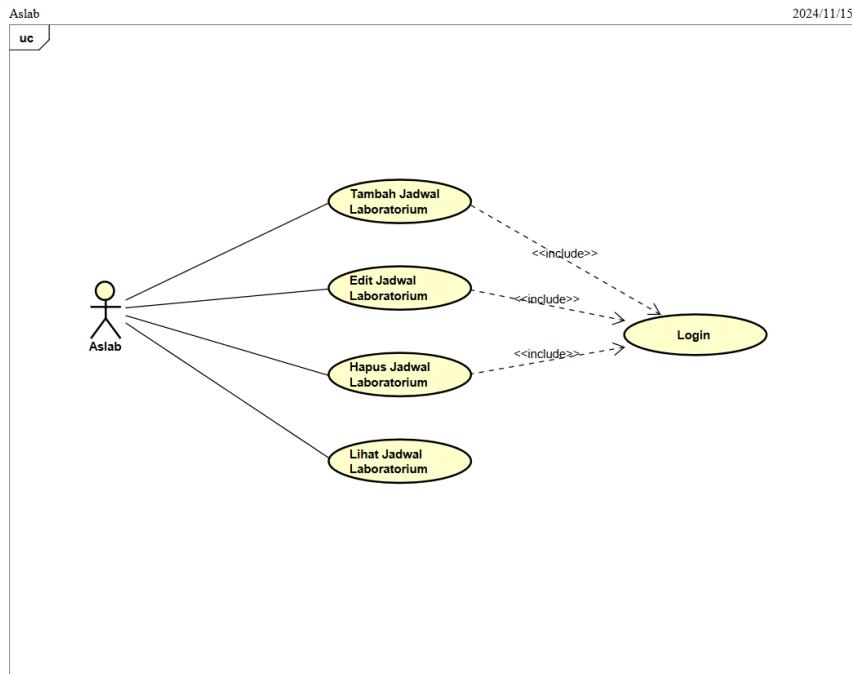
**Gambar D.7.** *Usecase Diagram Kalab*



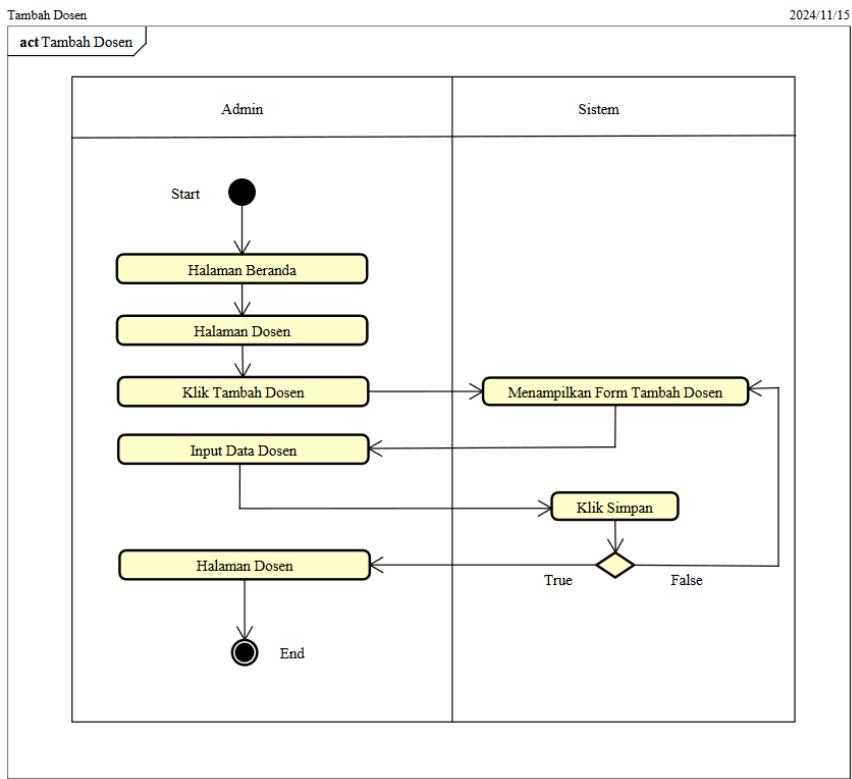
**Gambar D.8.** Usecase Diagram Kaprodi



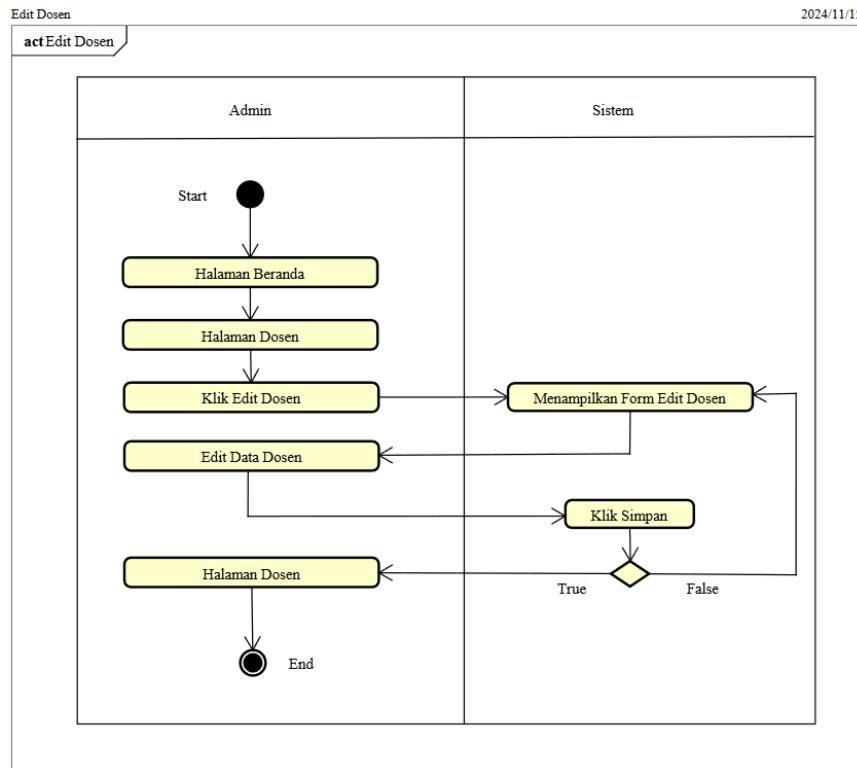
**Gambar D.9.** Usecase Diagram Sekprodi



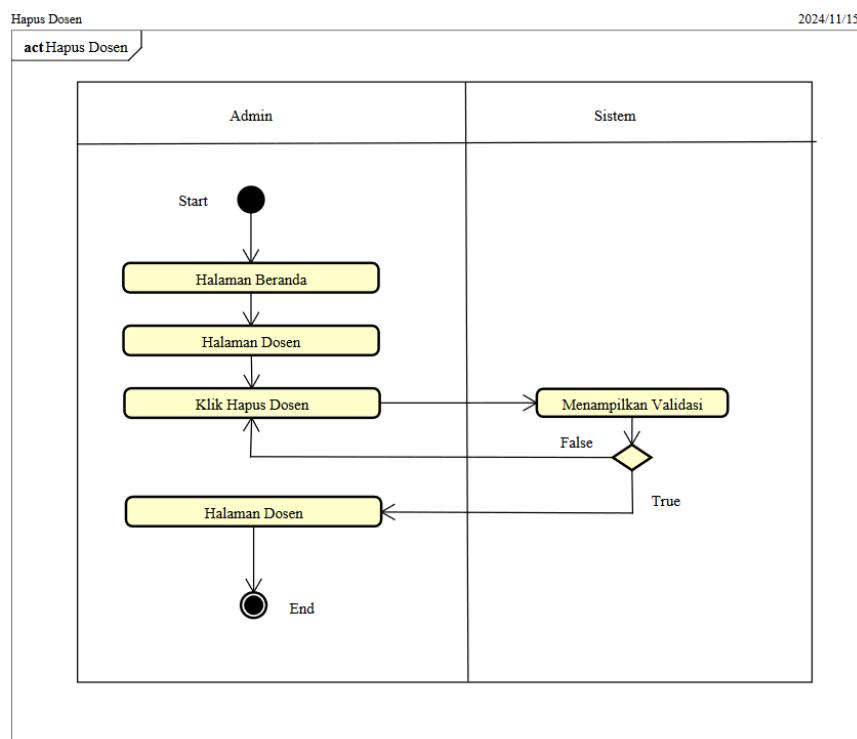
**Gambar D.10.** Usecase Diagram Aslab



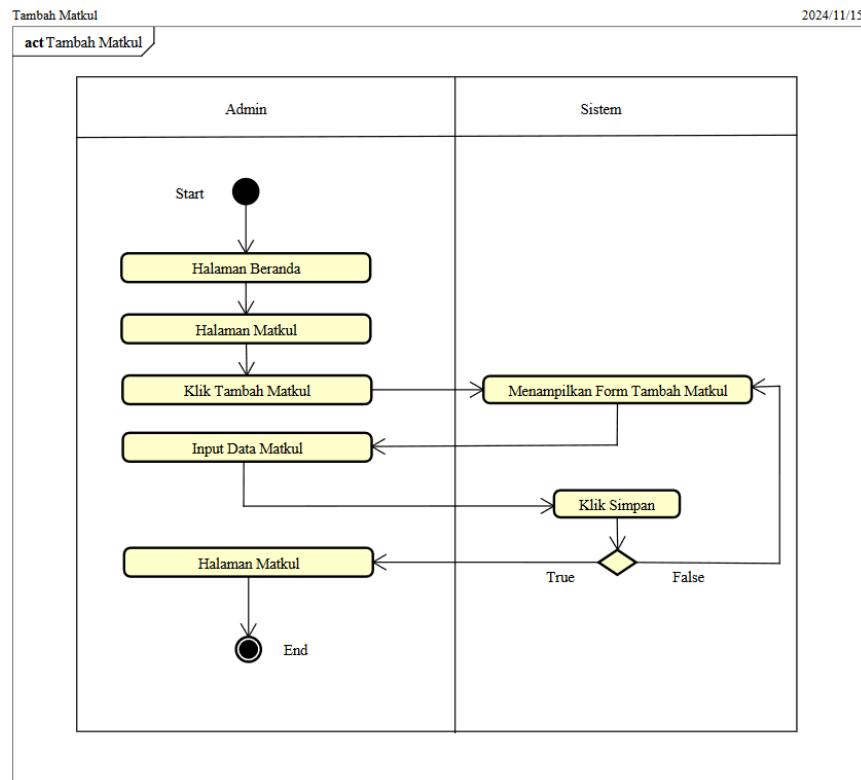
**Gambar D.11.** Activity Diagram Tambah Dosen



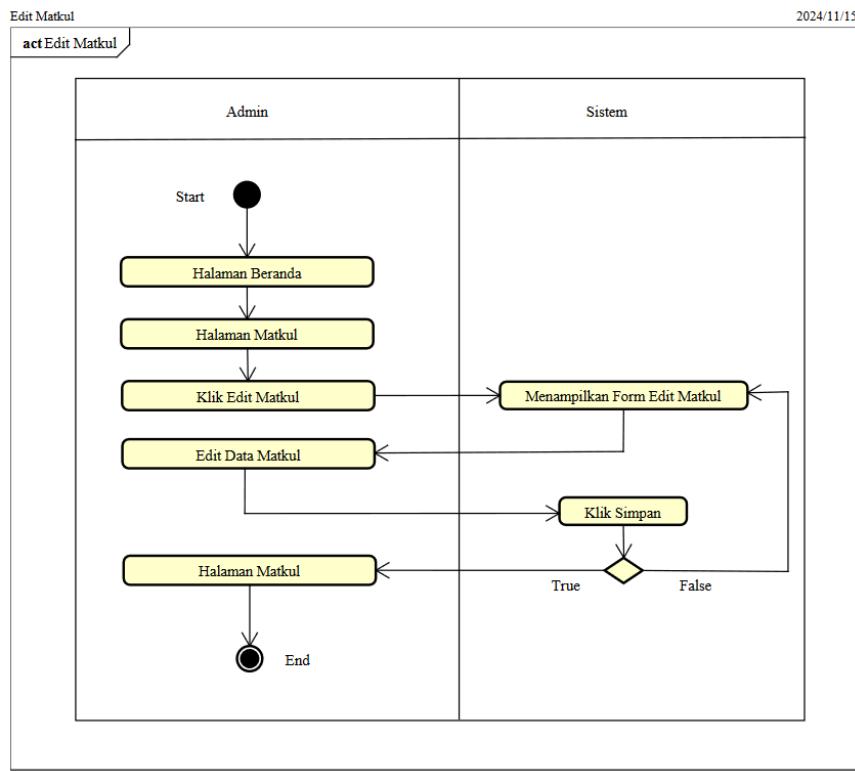
**Gambar D.12.** *Activity Diagram Edit Dosen*



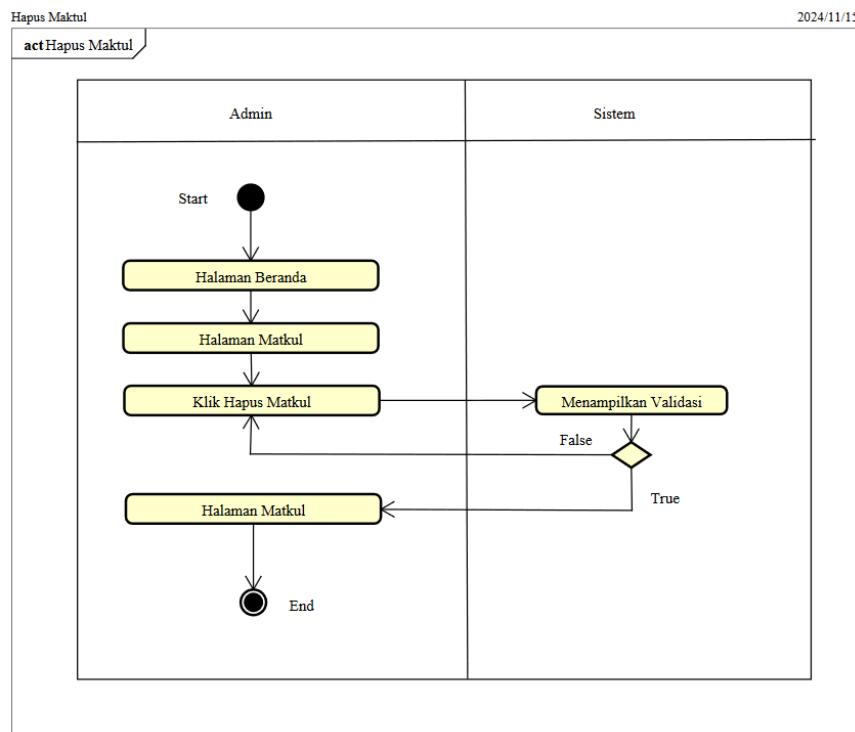
**Gambar D.13.** *Activity Diagram Hapus Dosen*



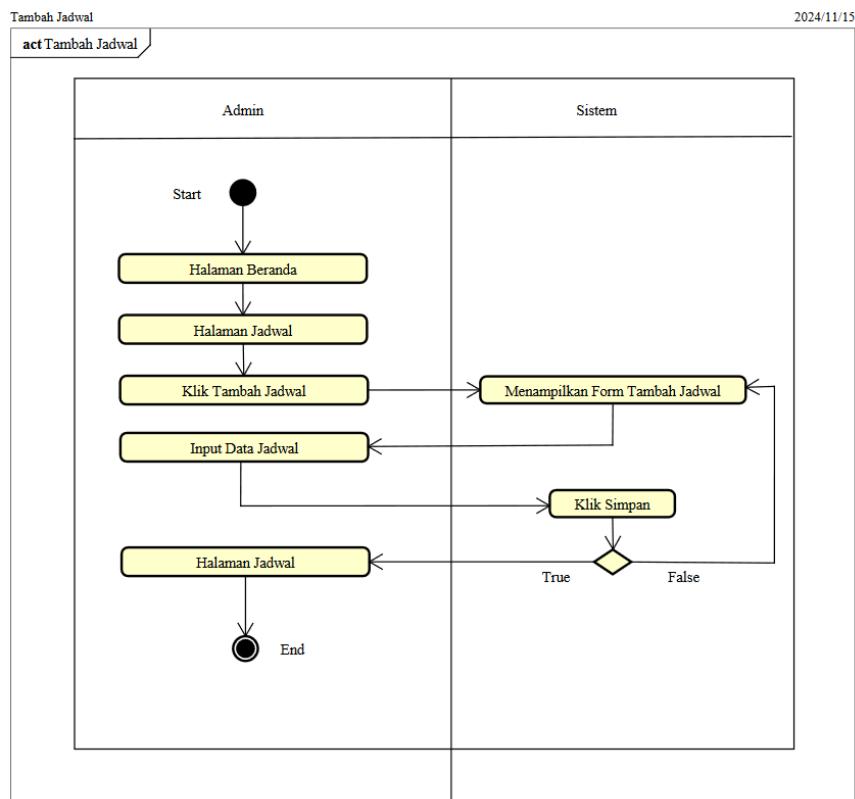
**Gambar D.14.** *Activity Diagram* Tambah Mata Kuliah



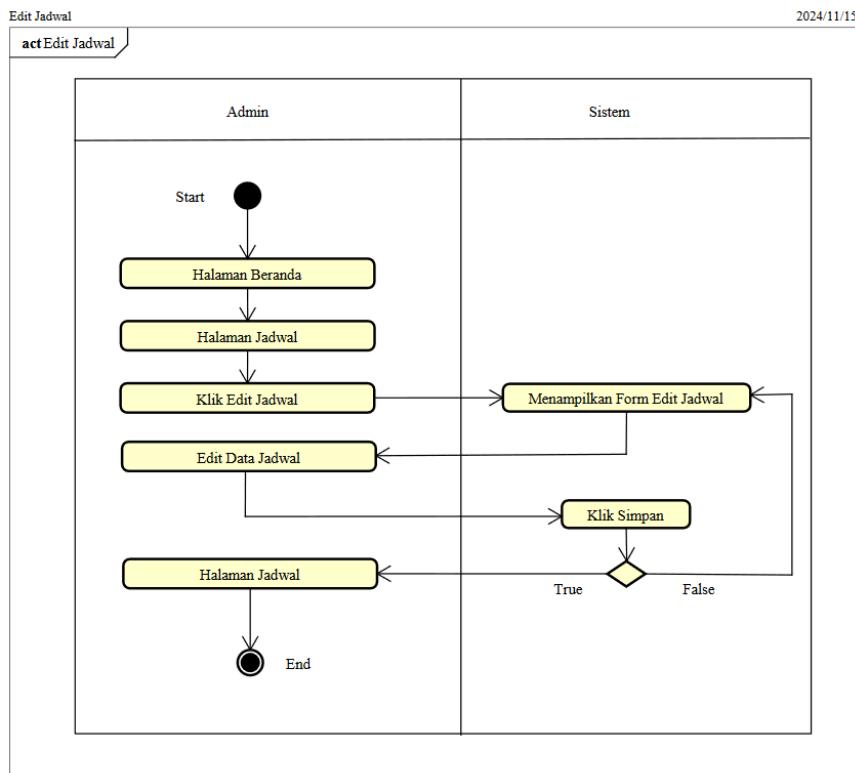
**Gambar D.15.** *Activity Diagram Edit Mata Kuliah*



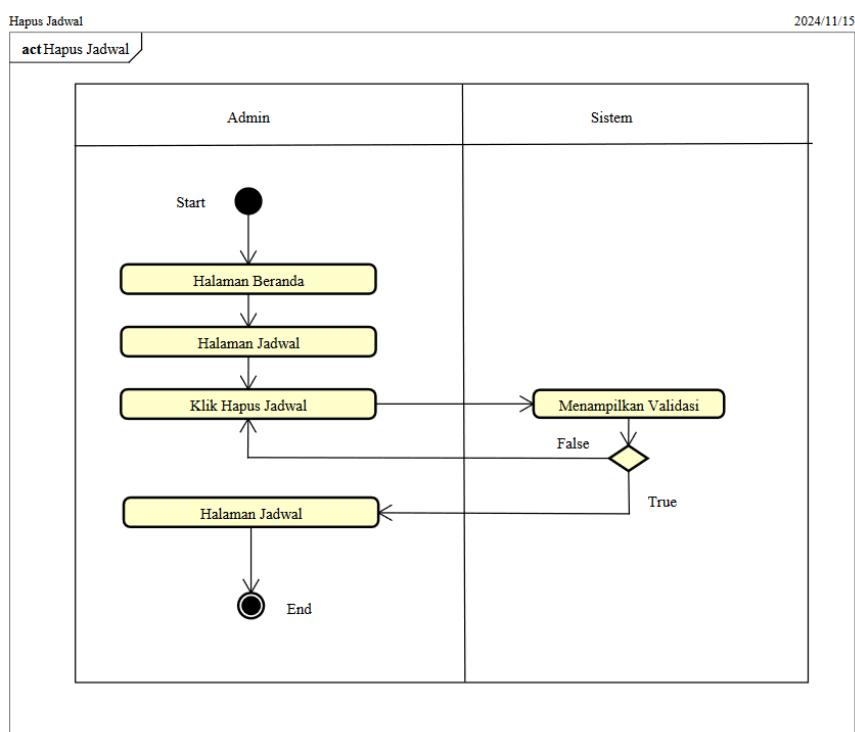
**Gambar D.16.** *Activity Diagram Hapus Mata Kuliah*



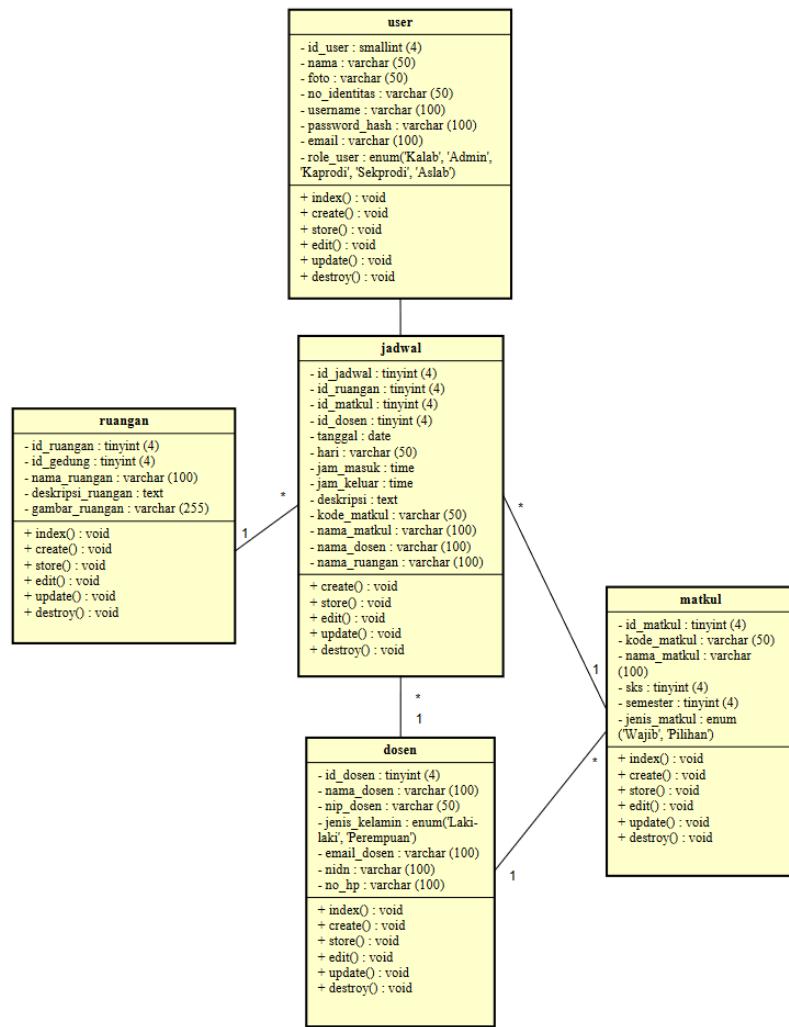
**Gambar D.17.** *Activity Diagram* Tambah Jadwal



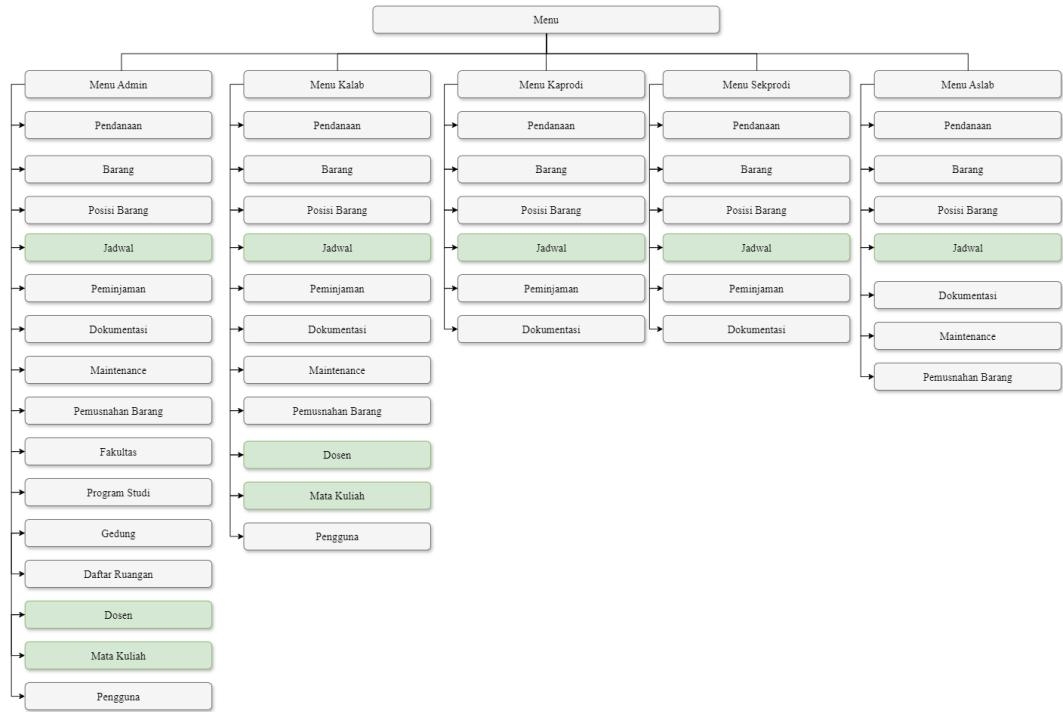
**Gambar D.18.** Activity Diagram Edit Jadwal



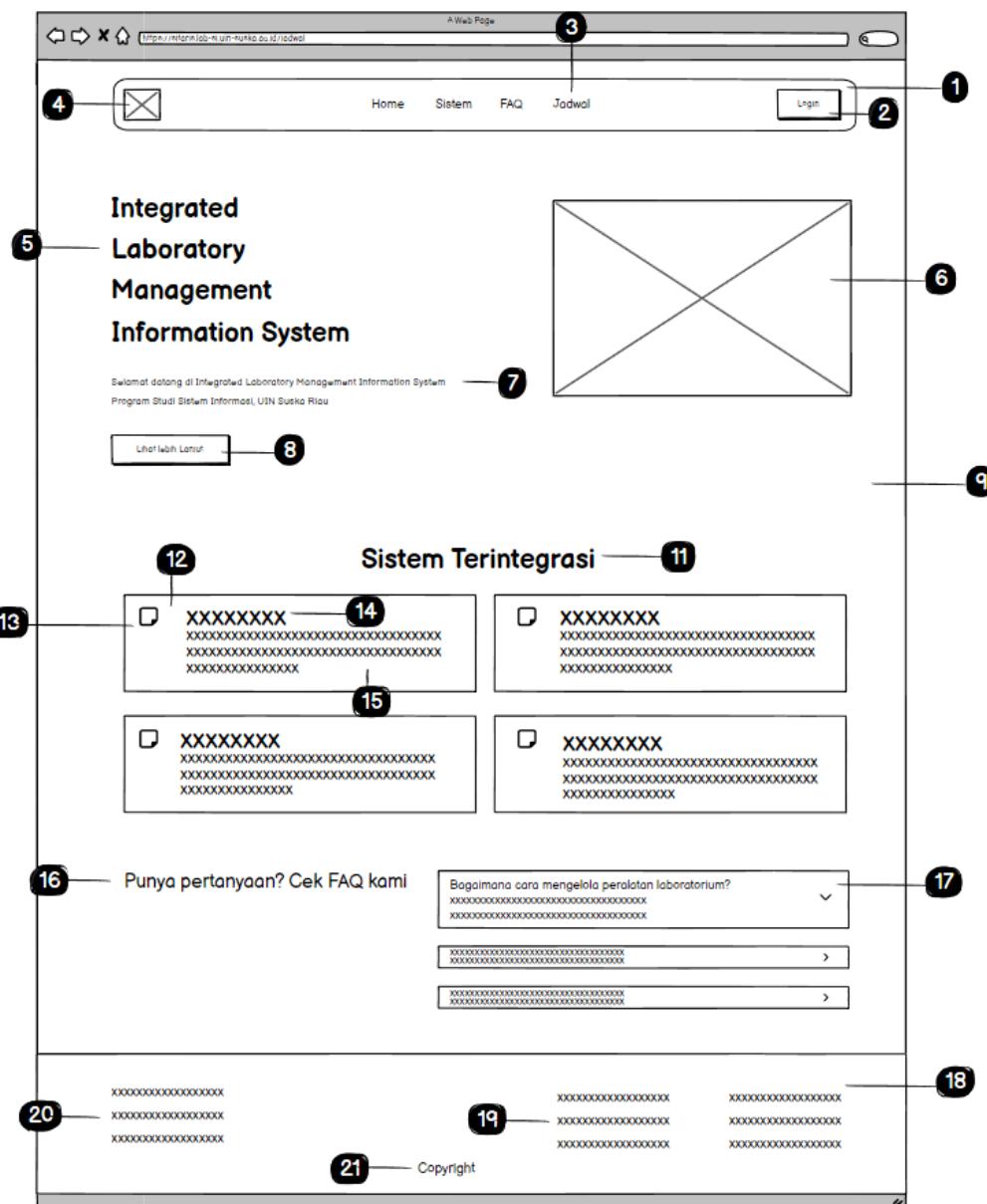
**Gambar D.19.** Activity Diagram Hapus Jadwal



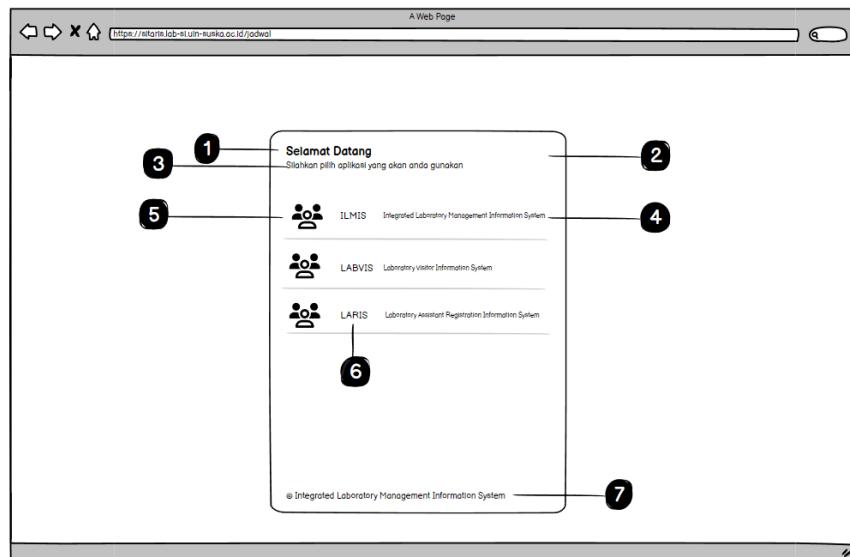
**Gambar D.20. Class Diagram ILMIS**



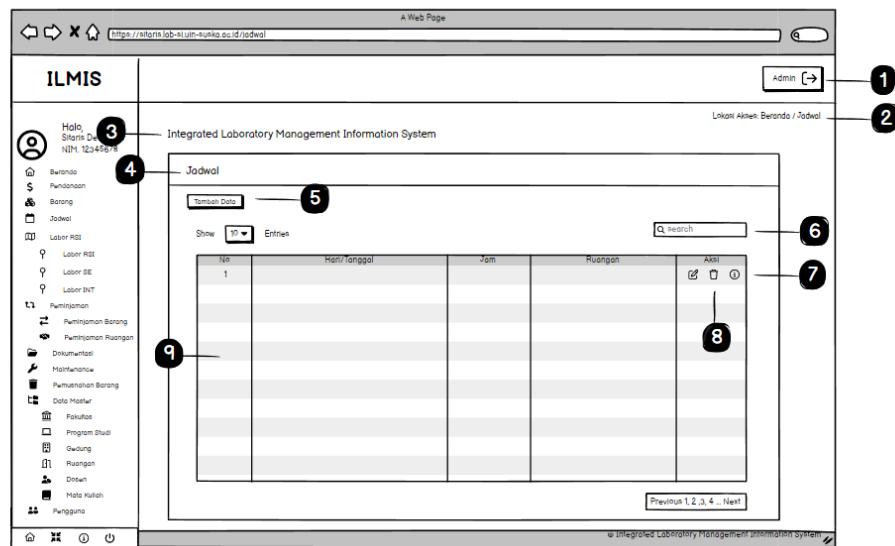
**Gambar D.21.** Struktur Menu ILMIS



**Gambar D.22.** Tampilan *Landing Page* ILMIS



**Gambar D.23.** Tampilan Pilihan Login untuk ILMIS



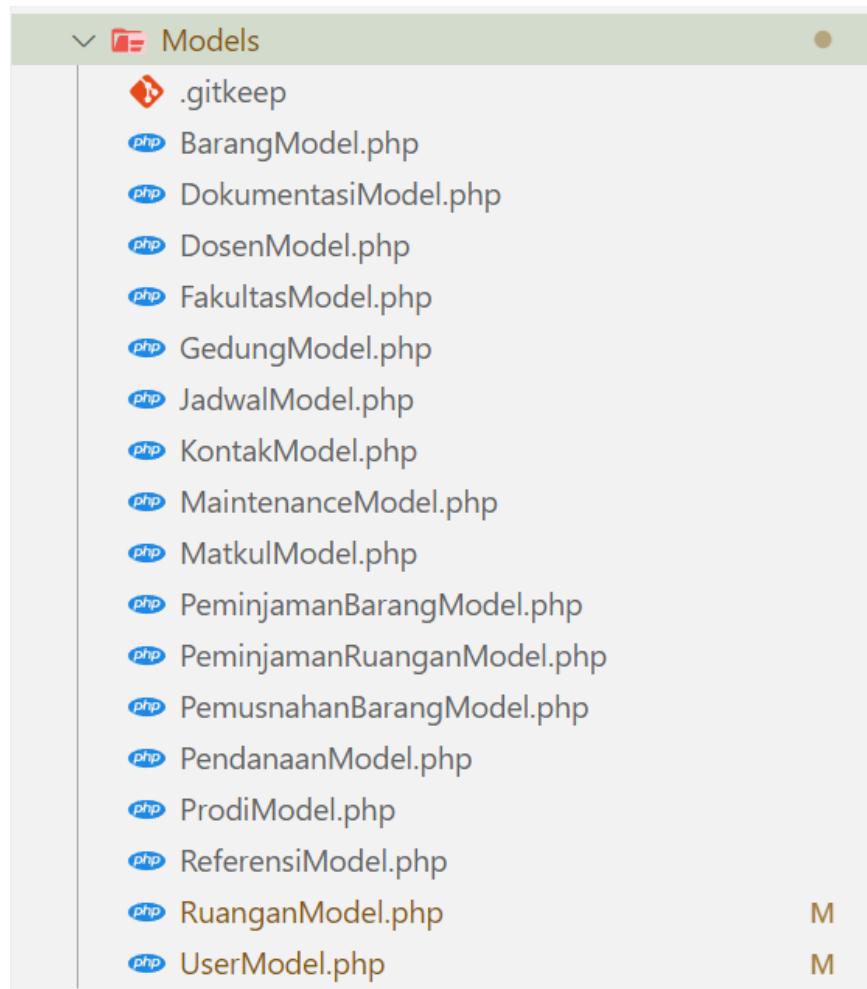
**Gambar D.24.** Tampilan Kelola Jadwal Laboratorium

**Gambar D.25.** Tampilan Tambah Jadwal Laboratorium

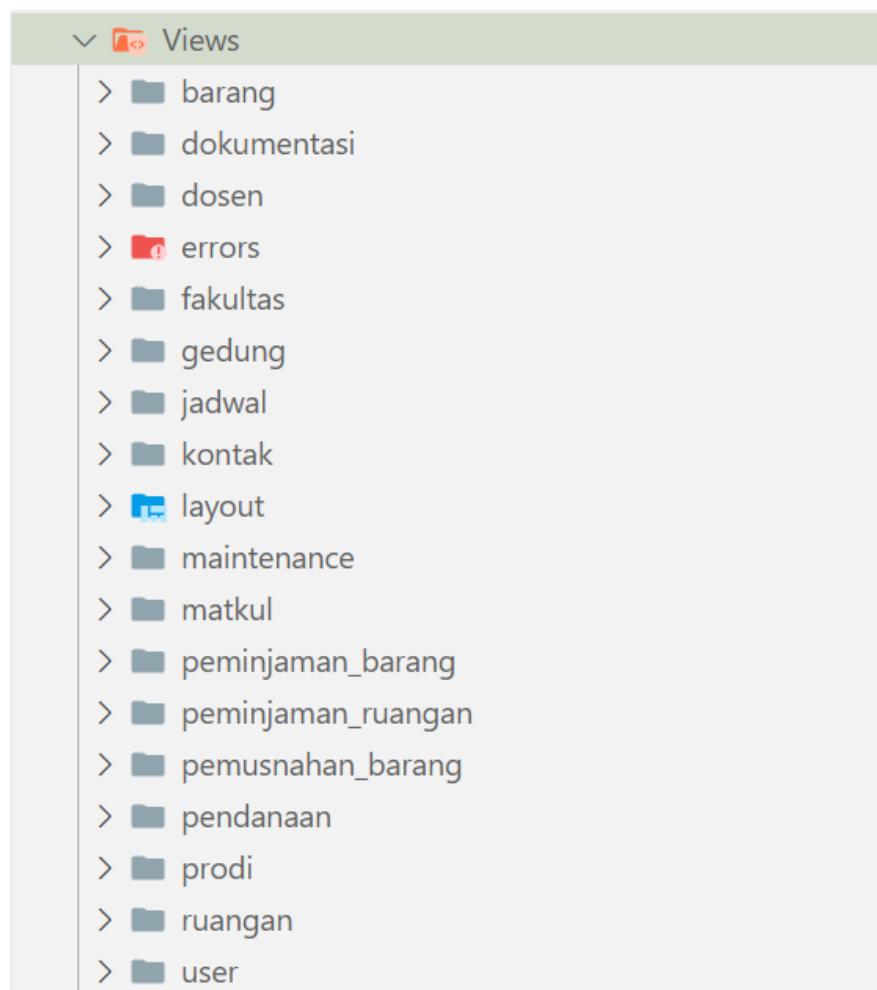
**Gambar D.26.** Tampilan Edit Jadwal Laboratorium

Table	Action	Rows	Type	Collation	Size	Overhead
barang	★ Browse Structure Search   Drop	503	InnoDB	utf8mb4_general_ci	144.0 Kib	-
biodatas	★ Browse Structure Search   Drop	2	InnoDB	utf8mb4_unicode_ci	32.0 Kib	-
dokumentasi	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
dosen	★ Browse Structure Search   Drop	23	InnoDB	utf8mb4_general_ci	16.0 Kib	-
fakultas	★ Browse Structure Search   Drop	9	InnoDB	utf8mb4_general_ci	32.0 Kib	-
gedung	★ Browse Structure Search   Drop	4	InnoDB	utf8mb4_general_ci	32.0 Kib	-
jadwal	★ Browse Structure Search   Drop	4	InnoDB	utf8mb4_unicode_ci	16.0 Kib	-
jawabans	★ Browse Structure Search   Drop	2	InnoDB	utf8mb4_unicode_ci	48.0 Kib	-
kelola_pendaftarans	★ Browse Structure Search   Drop	1	InnoDB	utf8mb4_unicode_ci	16.0 Kib	-
kelola_ujians	★ Browse Structure Search   Drop	1	InnoDB	utf8mb4_unicode_ci	32.0 Kib	-
kontak	★ Browse Structure Search   Drop	12	InnoDB	utf8mb4_general_ci	16.0 Kib	-
kunjungan	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
maintenance	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_general_ci	48.0 Kib	-
matkul	★ Browse Structure Search   Drop	68	InnoDB	utf8mb4_general_ci	16.0 Kib	-
migrations	★ Browse Structure Search   Drop	9	InnoDB	utf8mb4_unicode_ci	16.0 Kib	-
peminjaman_barang	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_general_ci	48.0 Kib	-
peminjaman_ruangan	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_general_ci	64.0 Kib	-
pemusnahan_barang	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_general_ci	48.0 Kib	-
pendanaan	★ Browse Structure Search   Drop	4	InnoDB	utf8mb4_general_ci	16.0 Kib	-
pengunjung	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
personal_access_tokens	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 Kib	-
postingans	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 Kib	-
prodi	★ Browse Structure Search   Drop	51	InnoDB	utf8mb4_general_ci	48.0 Kib	-
referensi	★ Browse Structure Search   Drop	1	InnoDB	utf8mb4_general_ci	16.0 Kib	-
ruangan	★ Browse Structure Search   Drop	4	InnoDB	utf8mb4_general_ci	48.0 Kib	-
soals	★ Browse Structure Search   Drop	2	InnoDB	utf8mb4_unicode_ci	32.0 Kib	-
user	★ Browse Structure Search   Drop	7	InnoDB	utf8mb4_general_ci	48.0 Kib	-
wawancaras	★ Browse Structure Search   Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 Kib	-
28 tables	Sum	707	InnoDB	utf8mb4_general_ci	1.0 Mib	0 B

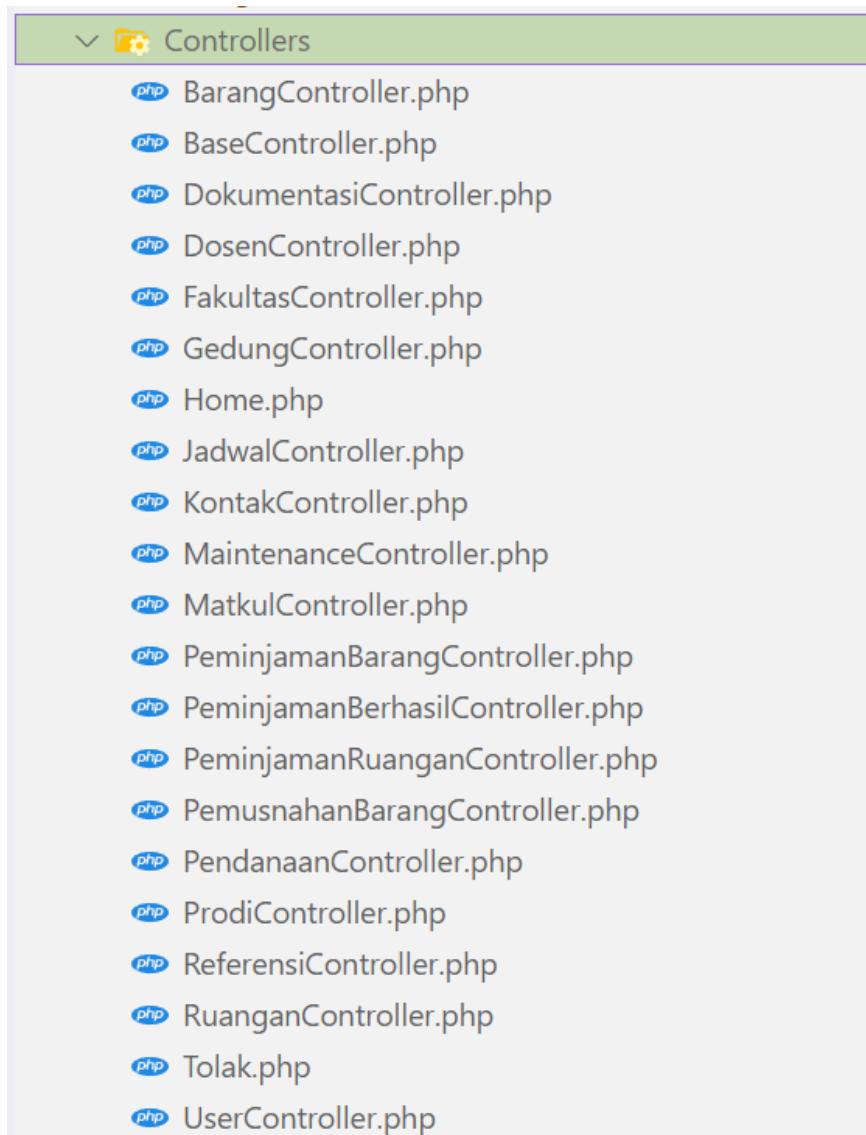
**Gambar D.27. Database**



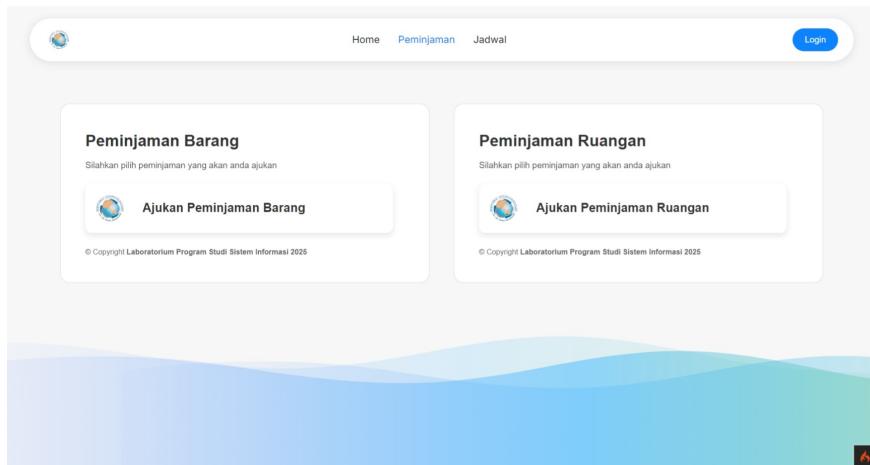
**Gambar D.28.** Implementasi Model



**Gambar D.29.** Implementasi View



**Gambar D.30.** Implementasi Controller



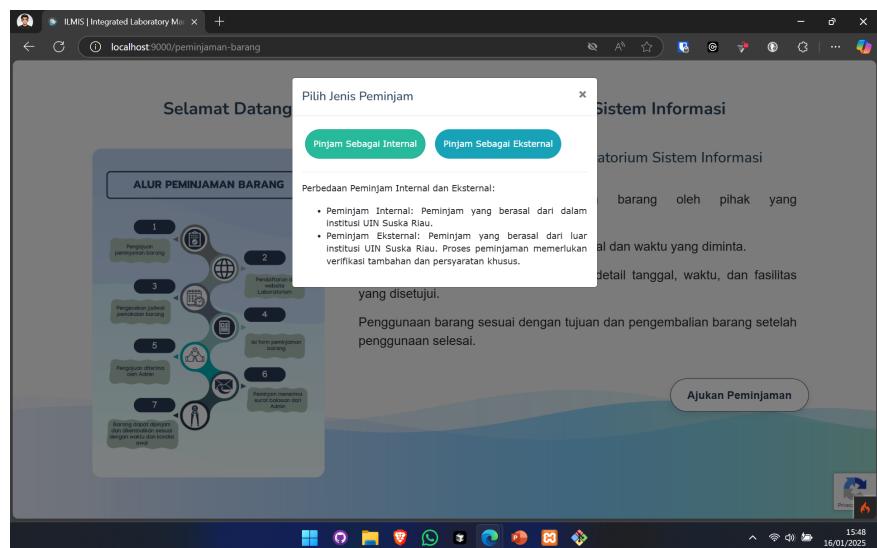
**Gambar D.31.** Tampilan Pilihan Peminjaman



**Gambar D.32.** Tampilan Pengelolaan Peminjaman Barang

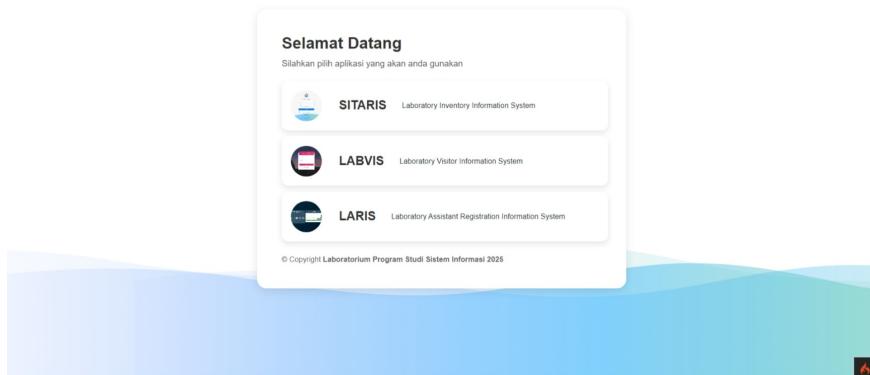


**Gambar D.33.** Tampilan Pengelolaan Peminjaman Ruangan

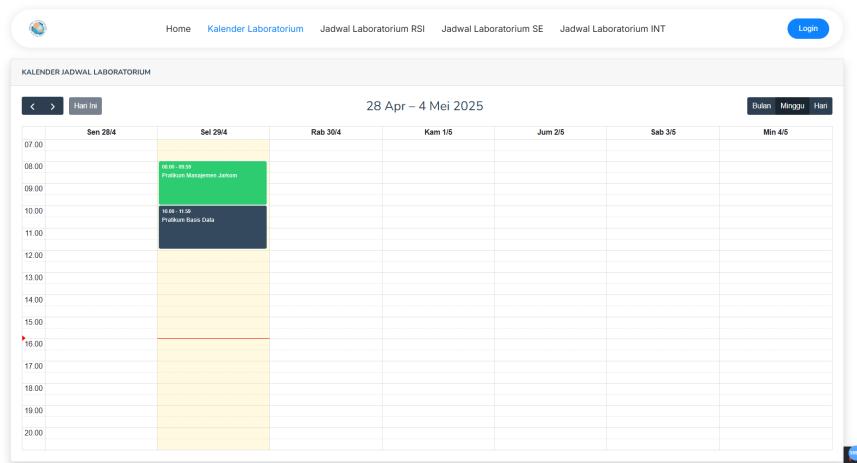


**Gambar D.34.** Tampilan Pilih Peminjam

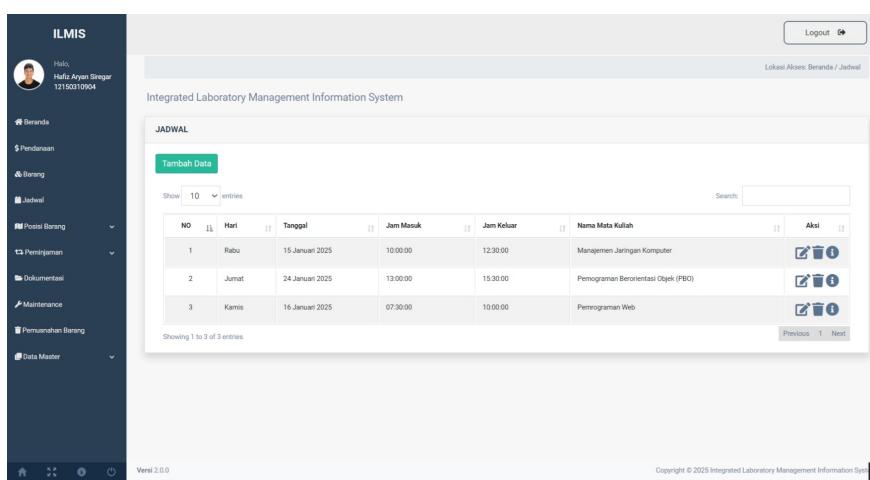
**Gambar D.35.** Tampilan *Landing Page* ILMIS



**Gambar D.36.** Tampilan Pilihan Aplikasi untuk *Login*



**Gambar D.37.** Tampilan Kalender ILMIS



**Gambar D.38.** Tampilan Pengelolaan Jadwal Laboratorium

JADWAL LABORATORIUM REKAYASA SISTEM INFORMASI

NO	Hari	Tanggal	Nama Mata Kuliah	Ruangan	Aksi
1	Rabu	15 Januari 2025	Manajemen Jaringan Komputer	Laboratorium Rekayasa Sistem Informasi	<a href="#">Detail</a>

Showing 1 to 1 of 1 entries

**Gambar D.39.** Tampilan Pengelolaan Jadwal Laboratorium RSI

JADWAL LABORATORIUM SOFTWARE ENGINEERING

NO	Hari	Tanggal	Nama Mata Kuliah	Ruangan	Aksi
1	Kamis	16 Januari 2025	Pemrograman Web	Laboratorium Software Engineering	<a href="#">Detail</a>

Showing 1 to 1 of 1 entries

**Gambar D.40.** Tampilan Pengelolaan Jadwal Laboratorium SE

JADWAL LABORATORIUM INTERNET

NO	Hari	Tanggal	Nama Mata Kuliah	Ruangan	Aksi
1	Jumat	24 Januari 2025	Pemrograman Berorientasi Objek (PBO)	Laboratorium Internet	<a href="#">Detail</a>

Showing 1 to 1 of 1 entries

**Gambar D.41.** Tampilan Pengelolaan Jadwal Laboratorium INT

Tambah Jadwal

Lokasi Akses: Beranda / Jadwal / Tambah

Versi 2.0.0 Copyright © 2025 Integrated Laboratory Management Information System

**Gambar D.42.** Tampilan Tambah Jadwal Laboratorium

Edit Jadwal

Lokasi Akses: Beranda / Jadwal / Edit

Versi 2.0.0 Copyright © 2025 Integrated Laboratory Management Information System

**Gambar D.43.** Tampilan Edit Jadwal Laboratorium

Cetak Jadwal Laboratorium

KEMENTERIAN AGAMA  
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU  
LABORATORIUM PROGRAM STUDI SISTEM INFORMASI  
معلم قسم نظم المعلومات  
LABORATORY OF INFORMATION SYSTEMS

Jl. HR. Soebrantas Km. 18, No. 155 Tushmidiani Tampan - Pekanbaru 28519, Po. Box. 1004, Telp. (0761) 589026-589027, Fax. (0761) 589025, E-Mail: sif.labor@uin-suska.ac.id

DAFTAR JADWAL LABORATORIUM PROGRAM STUDI SISTEM INFORMASI FAKULTAS SAINS DAN TEKNOLOGI UIN SUSKA RIAU

NO	Jam Masuk	Jam Keluar	Nama Mata Kuliah	Ruangan	Nama Dosen
1	11:30:00	12:30:00	Aqidah Akhlak	Laboratorium Internet	Siti Monalisa, ST., M.Kom
2	07:30:00	10:00:00	Aqidah Akhlak	Laboratorium Rekayasa Sistem Informasi	Eki Saputra, S.Kom., M.Kom

Pekanbaru, 16 Januari 2025  
Kepala Laboratorium  
Program Studi Sistem Informasi,  
Tengku Khairi Ahsyar, S.Kom., M.Kom.  
198505202023211020

**Gambar D.44.** Tampilan Cetak Jadwal Laboratorium dalam Format PDF

NO	Kode Mata Kuliah	Nama Mata Kuliah	Aksi
1	UN2001	Aqidah Akhlak	
2	UN2006	Pancasila dan Kewarganegaraan	
3	UN2007	Bahasa Indonesia	
4	UN2008	Bahasa Arab	
5	PSI1101	Pengantar Teknologi Informasi	
6	PSI1102	Organisasi dan Manajemen	
7	PSI1103	Matematika Sistem Informasi	
8	PSI1104	Pengantar Bisnis	

**Gambar D.45.** Tampilan Kelola Mata Kuliah Laboratorium

**Gambar D.46.** Tampilan Tambah Mata Kuliah Laboratorium

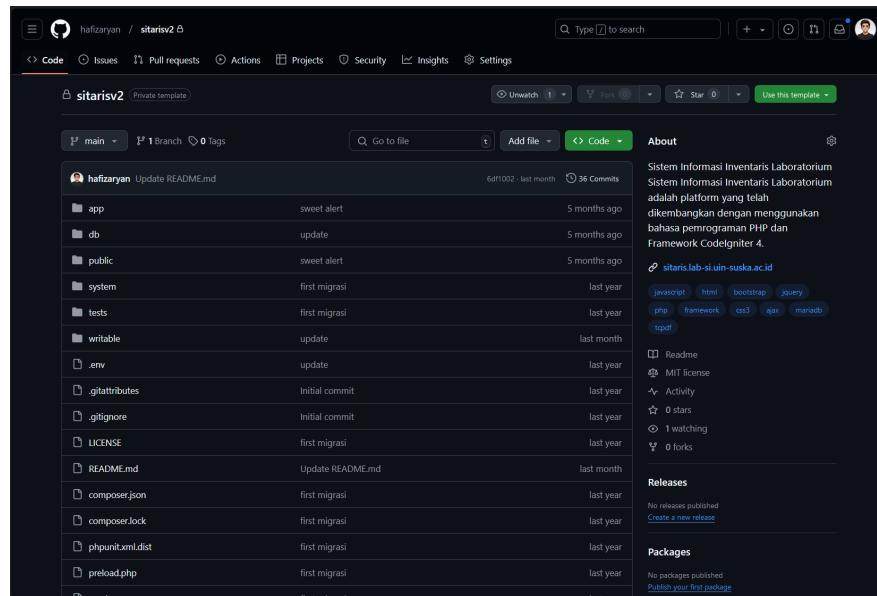
**Gambar D.47.** Tampilan Edit Mata Kuliah Laboratorium

NO	Nama Dosen	NIP	Jenis Kelamin	Aksi
1	Eki Saputra, S.Kom., M.Kom	198307162011011008	Laki-laki	
2	Siti Monalisa, ST, M.Kom	198502142015032004	Perempuan	
3	Tengku Khairil Ahayat, S.Kom., M.Kom	198505202022211020	Laki-laki	
4	Salide, S.Kom., M.Kom., MM, PhD	199008272023211019	Laki-laki	
5	Anggraini, S.Kom., M.Eng., PhD	198408212009012008	Perempuan	
6	Dr. Rice Novita, S.Kom., M.Kom	198511272023212032	Perempuan	
7	Syafullah, SE., M.Sc	198012152009121002	Laki-laki	
8	Irida Maita, S.Kom., M.Sc	197903132007102005	Perempuan	

**Gambar D.48.** Tampilan Kelola Dosen Laboratorium

**Gambar D.49.** Tampilan Tambah Dosen Laboratorium

**Gambar D.50.** Tampilan Edit Dosen Laboratorium



**Gambar D.51. Source Code**

## DAFTAR RIWAYAT HIDUP



Hafiz Aryan Siregar lahir di Sibuhuan, pada tanggal 06 Desember 2002. Peneliti merupakan anak dari Bapak Almarhum Ahmad Sofyan Siregar dan Ibu Arni Shopiyah Nasution. Peneliti adalah anak pertama dari tiga bersaudara yang disebut anak panggoaran dalam istilah Mandailing. Pada tahun 2009 peneliti memulai pendidikan Sekolah Dasar di SD Negeri 0102 Sibuhuan. Setelah menyelesaikan pendidikan Sekolah Dasar peneliti melanjutkan pendidikan tingkat SLTP di Madrasah Tsanawiyah Negeri Sibuhuan dan selesai pada tahun 2018. Peneliti melanjutkan pendidikan di SMA Negeri 1 Barumun. Setelah menyelesaikan pendidikan di SMA Negeri 1 Barumun pada tahun 2021, peneliti pun melanjutkan pendidikan dengan menjadi mahasiswa Program Studi Sistem Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau. Pada masa perkuliahan, peneliti aktif berorganisasi dan menjadi bagian dari beberapa kepanitiaan. Peneliti bergabung di organisasi *Information System Networking Club Research* (ISNC Research), Himpunan Mahasiswa Sistem Informasi (HIMASI) dan *Google Developer Student Clubs* (GDSC). Peneliti aktif menjadi bagian dari Laboratorium Program Studi Sistem Informasi sebagai Asisten Laboratorium dan Tim Pengembang Website. Tidak hanya belajar di dalam kelas, peneliti juga mengikuti program Merdeka Belajar Kampus Merdeka (MBKM) yaitu studi independen bersertifikat di Bangkit Academy 2024 Batch 2 di alur belajar *Cloud Computing*. Peneliti juga mengikuti *Coding Camp powered by DBS Foundation* 2025 di alur belajar *Front-End & Back-End Developer* di akhir masa perkuliahan peneliti. Peneliti menyelesaikan pendidikan Strata-1 dalam waktu 8 semester setelah berhasil menyelesaikan Tugas Akhir (TA) dengan judul penelitian "Pengembangan Sistem Informasi Manajemen Laboratorium Menggunakan Metode *Agile Development*".