

**SISTEM INFORMASI MANAJEMEN PEMINJAMAN
PERALATAN LABORATORIUM DENGAN METODE RAPID
APPLICATION DEVELOPMENT (RAD)**

PROYEK AKHIR

Laporan akhir ini dibuat dan diajukan untuk memenuhi salah satu syarat kelulusan
Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung



Disusun Oleh

Putra Baruna Wijaya *NIM:* 1602151

POLITEKNIK MANUFAKTUR NEGERI

BANGKA BELITUNG

TAHUN 2025

LEMBAR PENGESAHAN

**SISTEM INFORMASI MANAJEMEN PEMINJAMAN PERALATAN
LABORATORIUM DENGAN METODE RAPID APPLICATION
DEVELOPMENT (RAD)**

Oleh:

Putra Baruna Wijaya / 1062151

Laporan akhir ini telah disetujui dan disahkan sebagai salah satu syarat kelulusan
Program Sarjana Terapan Politeknik Manufaktur Negeri Bangka Belitung

Menyetujui,

Pembimbing 1



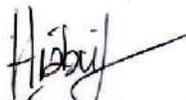
Riki Afriansyah, S.T., M.T
199004042019031013

Pembimbing 2



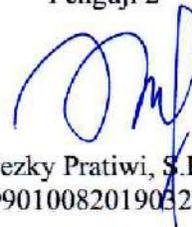
Eko Sulisty, S.T., M.T
197110202021211002

Penguji 1



M. Hizbul Wathan, M.Kom
198904182024061001

Penguji 2



Indah Riezky Pratiwi, S.Pd., M.Pd.
199010082019032018

PERNYATAAN BUKAN PLAGIAT

Yang Bertanda tangan di bawah ini :

Nama Mahasiswa : Putra Baruna Wijaya

NIM : 1062151

Dengan Judul : SISTEM INFORMASI MANAJEMEN PEMINJAMAN
PERALATAN LABORATORIUM DENGAN METODE
RAPID APPLICATION DEVELOPMENT (RAD)

Menyatakan bahwa laporan akhir ini adalah hasil kerja kami sendiri dan bukan merupakan plagiat. Pernyataan ini kami buat dengan sebenarnya dan bila ternyata dikemudian hari ternyata melanggar pernyataan ini, kami bersedia menerima sanksi yang berlaku.

Sungailiat, 18 Juli 2025

Nama Mahasiswa

Tanda Tangan

I. Putra Baruna Wijaya

()

ABSTRAK

Laboratorium pendidikan merupakan salah satu komponen penting dalam mendukung kegiatan praktikum dan penelitian di lingkungan akademik. Namun, proses peminjaman peralatan laboratorium yang masih dilakukan secara manual menimbulkan berbagai kendala, seperti pencatatan yang tidak teratur, keterlambatan pengembalian, serta rendahnya transparansi dan akurasi informasi. Untuk mengatasi permasalahan tersebut, dikembangkan sistem informasi manajemen peminjaman peralatan laboratorium berbasis *web* menggunakan metode *Rapid Application Development* (RAD). Tujuan dari pengembangan sistem ini adalah untuk meningkatkan efisiensi dan kualitas layanan peminjaman alat. Metode RAD dipilih karena mampu mempercepat proses pengembangan melalui iterasi prototipe dan umpan balik langsung dari pengguna. Sistem dibangun menggunakan bahasa pemrograman PHP dengan framework Laravel, didukung oleh MySQL sebagai basis data, Bootstrap sebagai antarmuka pengguna, serta dilengkapi fitur pemindaian *QR code* dan notifikasi otomatis melalui *email* sebagai pengingat pengembalian alat. Pengembangan sistem dilakukan melalui tahapan analisis kebutuhan, perancangan antarmuka, implementasi, dan pengujian menggunakan metode *Black Box Testing* dan *User Acceptance Testing* (UAT). Hasil pengujian menunjukkan bahwa sistem berfungsi sesuai harapan dan diterima dengan baik oleh pengguna. Tingkat kepuasan pengguna berdasarkan hasil UAT berkisar antara 88,5% hingga 90,8%, yang termasuk dalam kategori “sangat baik”. Dengan adanya sistem ini, pengelolaan peralatan laboratorium menjadi lebih efisien, terstruktur, dan mendukung kegiatan akademik secara optimal.

Kata kunci: Bootstrap, Laravel, PHP, QRCode, Sistem Informasi

ABSTRACT

Educational laboratories are essential components in supporting practicum and research activities within academic environments. However, the equipment loan process is still commonly conducted manually, leading to issues such as disorganized data recording, delayed returns, and limited transparency and accuracy of information. To address these problems, a web-based Laboratory Equipment Loan Management Information System was developed using the Rapid Application Development (RAD) method. This system aims to improve efficiency and service quality in the equipment loan process. The RAD method was chosen for its ability to accelerate development through iterative prototyping and direct user feedback. The system was built using the PHP programming language with the Laravel framework, supported by MySQL as the database, Bootstrap as the user interface, and equipped with QR code scanning and automatic email notifications as return reminders. The development process included needs analysis, interface design, system implementation, and testing using Black Box Testing and User Acceptance Testing (UAT). The test results show that the system functions as expected and is well-received by users. Based on UAT results, user satisfaction ranges from 88.5% to 90.8%, which falls under the “very good” category. With this system, laboratory equipment management becomes more efficient, structured, and supportive of academic activities.

Keywords: *Bootstrap, Laravel, PHP, QRCode, Information System*

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji dan syukur penulis panjatkan ke hadirat Allah Swt. atas rahmat, taufik, dan karunia-Nya sehingga penulis dapat menyelesaikan laporan Proyek Akhir yang berjudul "Sistem Informasi Manajemen Peminjaman Peralatan Laboratorium dengan Metode *Rapid Application Development (RAD)*". Laporan ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Terapan pada Program Studi Teknologi Rekayasa Perangkat Lunak, Politeknik Manufaktur Negeri Bangka Belitung.

Penyusunan laporan ini tentunya tidak terlepas dari bantuan, dukungan, dan bimbingan dari berbagai pihak. Oleh karena itu, dengan segala ketulusan dan rasa hormat, penulis menyampaikan ucapan terima kasih kepada:

1. Allah Swt. atas segala rahmat, hidayah, dan kemudahan dalam menyelesaikan Proyek Akhir ini.
2. Kedua orang tua tercinta atas doa, dukungan, dan semangat yang tiada henti.
3. Dosen pembimbing, Bapak Riki Afriansyah, S.T., M.T. dan Bapak Eko Sulistyono, S.T., M.T., atas bimbingan dan arahnya selama proses penyusunan proyek akhir ini.
4. Seluruh pimpinan, dosen, dan staf Politeknik Manufaktur Negeri Bangka Belitung atas ilmu dan dukungan selama masa studi.
5. Pihak laboratorium serta Pengurus Laboratorium (PLP) atas bantuan dan kerja sama yang diberikan.
6. Teman-teman seperjuangan atas kebersamaan, semangat, dan dukungan yang telah terjalin selama masa perkuliahan.

Penulis menyadari sepenuhnya bahwa laporan ini masih jauh dari kata sempurna, baik dari segi isi maupun penulisan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat penulis harapkan untuk perbaikan di masa mendatang.

Akhir kata, penulis berharap laporan ini dapat memberikan manfaat bagi pengembangan sistem informasi di lingkungan akademik maupun pihak lain yang berkepentingan.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Sungailiat, 18 Juli 2025

Putra Baruna Wijaya

DAFTAR ISI

LEMBAR PENGESAHAN	I
PERNYATAAN BUKAN PLAGIAT	II
ABSTRAK	III
ABSTRACT	IV
KATA PENGANTAR	V
DAFTAR ISI	VI
DAFTAR TABEL	VIII
DAFTAR GAMBAR	IX
DAFTAR LAMPIRAN	XI
BAB I PENDAHULUAN	1
1.1. LATAR BELAKANG MASALAH	1
1.2. PERUMUSAN MASALAH	3
1.3. TUJUAN PROYEK AKHIR	3
1.4. MANFAAT.....	3
BAB II DASAR TEORI	4
2.1. PENELITIAN TERDAHULU	4
2.2. SISTEM INFORMASI MANAJEMEN	6
2.3. MANAJEMEN PERALATAN LABORATORIUM.....	7
2.4. METODOLOGI <i>RAPID APPLICATION DEVELOPMENT</i> (RAD)	7
2.5. TEKNOLOGI PENDUKUNG SISTEM.....	8
2.5.1. <i>PHP</i>	8
2.5.2. <i>Laravel Framework</i>	8
2.5.3. <i>Database MySQL</i>	9
2.5.4. <i>PhpMyAdmin</i>	9
2.5.5. <i>Laragon</i>	9
2.5.6. <i>Visual Studio Code</i>	10
2.5.7. <i>Bootstrap</i>	11
2.5.8. <i>Web Hosting</i>	11
2.5.9. <i>QRCode</i>	11
2.5.10. <i>Notifikasi Otomatis</i>	12
2.6. UML	13
2.6.1. <i>Use Case Diagram</i>	15

2.6.2.	<i>Activity Diagram</i>	15
2.6.3.	<i>Class Diagram</i>	16
2.7.	PENGUJIAN.....	17
2.7.1.	<i>Black Box Testing</i>	17
2.7.2.	<i>User Acceptance Testing</i>	18
BAB III	METODE PELAKSANAAN	19
3.1.	JENIS PENELITIAN.....	19
3.2.	METODOLOGI PENGEMBANGAN SISTEM	19
3.3.	DIAGRAM ALIR PROSES PENGEMBANGAN SISTEM	21
3.4.	TEKNIK PENGUMPULAN DATA.....	23
3.5.	PERANCANGAN SISTEM	24
3.5.1.	<i>Flowchart Sistem</i>	24
3.5.2.	<i>Diagram Use Case</i>	30
3.5.3.	<i>Activity Diagram</i>	33
3.5.4.	<i>Class Diagram</i>	48
3.5.5.	<i>Wireframe Antarmuka Pengguna</i>	50
3.6.	RANCANGAN DAN IMPLEMENTASI SISTEM	58
3.7.	PENGUJIAN SISTEM.....	61
3.7.1.	<i>Black Box Testing</i>	61
3.7.2.	<i>User Acceptance Testing (UAT)</i>	63
BAB IV	PEMBAHASAN	67
4.1.	HASIL ANALISIS KEBUTUHAN	67
4.1.1.	<i>Kebutuhan Fungsional</i>	67
4.1.2.	<i>Kebutuhan Nonfungsional</i>	68
4.2.	HASIL PERANCANGAN SISTEM	69
4.3.	HASIL IMPLEMENTASI SISTEM	71
4.3.1.	<i>Arsitektur Sistem</i>	71
4.3.2.	<i>User Interface</i>	75
4.4.	HASIL PENGUJIAN	102
4.4.1.	<i>Umpan Balik Pengguna</i>	102
4.4.2.	<i>Pengujian Black Box</i>	103
4.4.3.	<i>Pengujian User Acceptance Testing (UAT)</i>	105
BAB V	KESIMPULAN DAN SARAN	109
5.1.	KESIMPULAN.....	109
5.2.	SARAN	110
DAFTAR PUSTAKA	111

Daftar Tabel

Table 3.5-1 Tabel Simbol Flowchart.....	25
Table 3.5-2 Tabel Simbol Use Case.....	30
Table 3.5-3 Tabel Simbol Activity Diagram.....	34
Table 3.5-4 Tabel Simbol Class Diagram.....	49
Table 3.7-1 Tabel Parameter Black Box Testing	61
Table 3.7-2 Tabel Parameter Perceived Usefulness (PU)	64
Table 3.7-3 Tabel Parameter Perceived Ease of Use (PEOU).....	65
Table 3.7-4 Tabel Parameter General Acceptance (GA).....	65
Table 4.4-1 Umpan Balik Pengguna	103
Table 4.4-2 Hasil Pengujian Black Box	104
Table 4.4-3 Hasil Penelitian Perceived Usefulness (PU).....	106
Table 4.4-4 Hasil Penelitian Perceived Ease of Use (PEOU)	107
Table 4.4-5 Hasil Penelitian General Acceptance (GA)	107
Table 4.4-6 Hasil Pengujian UAT	108

Daftar Gambar

Gambar 3.2-1 Metode RAD	20
Gambar 3.3-1 flowchart Proses Pengembangan Sistem	22
Gambar 3.5-1 Flowchart Sistem Admin	27
Gambar 3.5-2 Flowchart Sistem Pengurus Laboratorium.....	28
Gambar 3.5-3 Flowchart Sistem Peminjam	29
Gambar 3.5-4 Diagram Use Case	33
Gambar 3.5-5 Activity Diagram Admin (Jurusan)	35
Gambar 3.5-6 Activity Diagram Admin (Jenjang)	36
Gambar 3.5-7 Activity Diagram Admin (Program Studi).....	37
Gambar 3.5-8 Activity Diagram Admin (Kelas)	38
Gambar 3.5-9 Activity Diagram Admin (Peminjam).....	39
Gambar 3.5-10 Activity Diagram Admin (Laboratorium)	40
Gambar 3.5-11 Activity Diagram Admin (Pengurus Laboratorium)	41
Gambar 3.5-12 Activity Diagram Pengurus Laboratorium (Tambah Alat).....	42
Gambar 3.5-13 Activity Diagram Pengurus Laboratorium (Generate QRCode) ..	43
Gambar 3.5-14 Activity Diagram Pengurus Laboratorium (Riwayat)	44
Gambar 3.5-15 Activity Diagram Pengurus Laboratorium (Laporan)	45
Gambar 3.5-16 Activity Diagram Peminjam (Peminjaman)	46
Gambar 3.5-17 Activity Diagram Peminjam.....	47
Gambar 3.5-18 Class Diagram	50
Gambar 3.5-19 Login UI.....	51
Gambar 3.5-20 Tampilan Layout Utama dengan Area Konten Dinamis	52
Gambar 3.5-21 Tampilan Dashboard Admin	53
Gambar 3.5-22 Tampilan Dashboard Pengurus Laboratorium	53
Gambar 3.5-23 Tampilan Dashboard Peminjam	54
Gambar 3.5-24 Tampilan Konten Tabel Data Dinamis.....	55
Gambar 3.5-25 Tampilan Generate QRCode	56
Gambar 3.5-26 Tampilan History	57
Gambar 3.5-27 Tampilan Form Input Data	58
Gambar 4.3-1 Contoh Model.....	72
Gambar 4.3-2 Contoh View.....	73
Gambar 4.3-3 Contoh Controller.....	74
Gambar 4.3-4 Contoh Route	75
Gambar 4.3-5 Tampilan Halaman Login.....	77
Gambar 4.3-6 Tampilan Halaman Dashboard Admin.....	78
Gambar 4.3-7 Tampilan Halaman Jurusan	79

Gambar 4.3-8 Tampilan Halaman Form Jurusan.....	79
Gambar 4.3-9 Tampilan Halaman Jenjang	80
Gambar 4.3-10 Tampilan Halaman Form Jenjang.....	80
Gambar 4.3-11 Tampilan Halaman Program Studi.....	81
Gambar 4.3-12 Tampilan Halaman Form Program Studi	81
Gambar 4.3-13 Tampilan Halaman Kelas	82
Gambar 4.3-14 Tampilan Halaman Form Kelas.....	82
Gambar 4.3-15 Tampilan Halaman Tambah Mahasiswa	83
Gambar 4.3-16 Tampilan Halaman Tambah Dosen.....	83
Gambar 4.3-17 Tampilan Halaman Mahasiswa	84
Gambar 4.3-18 Tampilan Halaman Dosen	84
Gambar 4.3-19 Tampilan Halaman Laboratorium.....	85
Gambar 4.3-20 Tampilan Halaman Form Laboratorium	85
Gambar 4.3-21 Tampilan Halaman Pengurus Laboratorium	86
Gambar 4.3-22 Tampilan Halaman Form Pengurus Laboratorium	86
Gambar 4.3-23 Tampilan Halaman Dashboard Pengurus Laboratorium.....	87
Gambar 4.3-24 Tampilan Halaman Manajemen Alat	88
Gambar 4.3-25 Tampilan Halaman Form Manajemen Alat	88
Gambar 4.3-26 Tampilan Halaman Generate QRCode.....	89
Gambar 4.3-27 Tampilan Halaman QRCode	89
Gambar 4.3-28 Tampilan Halaman Peralatan Tersedia.....	90
Gambar 4.3-29 Tampilan Halaman Peralatan Rusak	91
Gambar 4.3-30 Tampilan Halaman Validasi Peminjaman	92
Gambar 4.3-31 Tampilan Halaman Validasi Pengembalian.....	93
Gambar 4.3-32 Tampilan Halaman History Pengurus Laboratorium	94
Gambar 4.3-33 Tampilan Halaman Laporan	95
Gambar 4.3-34 Tampilan Halaman Dashboard Peminjam.....	96
Gambar 4.3-35 Tampilan Halaman Pinjam Alat.....	97
Gambar 4.3-36 Tampilan Halaman Scan QRCode.....	97
Gambar 4.3-37 Tampilan Halaman Status Peminjaman (pending)	98
Gambar 4.3-38 Tampilan Halaman Status Peminjaman (reject)	98
Gambar 4.3-39 Tampilan Halaman Status Peminjaman (Approve)	99
Gambar 4.3-40 Tampilan Halaman Status Peminjaman (sedang dikembalikan) ..	99
Gambar 4.3-41 Tampilan Halaman Pengembalian Alat.....	100
Gambar 4.3-42 Tampilan Halaman History Peminjaman	101
Gambar 4.3-43 Tampilan Notifikasi Email Keterlambatan.....	102
Gambar 4.4-1 Hasil Pengujian UAT	108

Daftar Lampiran

Lampiran 1: Daftar Riwayat Hidup

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Laboratorium pendidikan merupakan salah satu komponen vital dalam menunjang kegiatan praktikum dan penelitian di lingkungan akademik. Sebagai sarana pembelajaran berbasis praktik, laboratorium menyediakan berbagai alat dan fasilitas yang digunakan oleh mahasiswa maupun dosen guna mendukung proses pendidikan serta pengembangan ilmu pengetahuan. Oleh karena itu, pengelolaan peralatan laboratorium secara efektif dan efisien menjadi hal yang krusial untuk menjamin ketersediaan, kelayakan fungsi, serta ketertiban penggunaannya.

Pada kenyataannya, sistem peminjaman peralatan laboratorium di berbagai institusi pendidikan, khususnya pada jenjang pendidikan vokasi, masih banyak dilakukan secara manual. Prosedur konvensional ini umumnya melibatkan pengisian formulir fisik oleh peminjam, yang kemudian diserahkan kepada petugas laboratorium untuk diproses secara administratif. Metode manual seperti ini dinilai kurang efisien karena membutuhkan waktu yang relatif lama, rentan terhadap kesalahan pencatatan (*misentry*), dan menyulitkan proses pelacakan status alat (Sumartono, Susanto, Huda, & Agustin, 2024).

Permasalahan tersebut menunjukkan bahwa diperlukan suatu sistem digital yang mampu mengotomatisasi proses peminjaman dan pengembalian peralatan laboratorium. Menurut (Susanti, Asmoro, & Maemunah, 2023) dalam pengembangan SiDal (Sistem BigData Laboratorium) di Universitas Pendidikan Indonesia, digitalisasi sistem peminjaman alat terbukti meningkatkan efisiensi pelayanan serta akurasi pendataan alat laboratorium.

Penelitian oleh (Kuswandari, Susanti, & Grey, 2023) juga menegaskan pentingnya penggunaan sistem berbasis QRCode dalam proses inventarisasi laboratorium, yang tidak hanya mempercepat identifikasi alat, tetapi juga mendukung penambahan fitur lanjutan seperti kalibrasi daring. Sementara itu, (Nikmah,

Hamidah, & Sukanti, 2023) menunjukkan bahwa sistem peminjaman alat berbasis daring terbukti lebih efektif dibanding metode manual, terutama dalam situasi yang menuntut efisiensi tinggi seperti masa pandemi.

Lebih lanjut, studi oleh (Andri & Oktaviati, 2025) mengembangkan sistem pelaporan kerusakan berbasis QRCode yang terintegrasi dengan metode Agile, yang memperlihatkan efisiensi pelaporan langsung dari pengguna kepada pengelola laboratorium. Selain itu, (Purnomoa & Alijoyob, 2024) merancang sistem peminjaman barang berbasis QRCode dengan aplikasi Android yang mampu meningkatkan akurasi pendataan dan keterlacakan alat secara real-time.

Dalam konteks penelitian ini, metode *Rapid Application Development* (RAD) dipilih sebagai pendekatan karena memiliki keunggulan dalam mempercepat siklus pengembangan perangkat lunak melalui tahapan iteratif, pembuatan prototipe cepat, serta umpan balik langsung dari pengguna. Pendekatan ini memungkinkan pengembangan sistem yang fleksibel dan adaptif terhadap kebutuhan laboratorium yang dinamis (Anggar, Anshor, & Hadikristanto, 2024).

Sistem informasi ini dirancang menggunakan *framework* Laravel dan bahasa pemrograman PHP. Laravel mengusung arsitektur *Model-View-Controller* (MVC) yang memungkinkan pemisahan antara logika program, tampilan, dan manajemen data. Kombinasi tersebut mempermudah proses pengembangan, pemeliharaan, dan pengembangan lanjutan (Madurapperuma, Shafana, & Sabani, 2022).

Fitur-fitur utama yang diimplementasikan dalam sistem ini meliputi integrasi QRCode untuk identifikasi alat, pengisian *logbook* otomatis, pencatatan alat terintegrasi dengan inventaris, pelaporan kerusakan, serta sistem notifikasi pengingat keterlambatan pengembalian. Seluruh alur kerja sistem disesuaikan dengan Prosedur Operasional Standar (POS) dan kebutuhan aktual Pengurus Laboratorium Pendidikan (PLP).

Dengan dikembangkannya sistem informasi manajemen peminjaman peralatan laboratorium ini, diharapkan proses pencatatan dan pelacakan alat menjadi lebih sistematis, akurat, serta minim kesalahan. Selain itu, sistem ini diharapkan mampu meningkatkan efisiensi dan akuntabilitas pengelolaan laboratorium, serta mendukung keberlangsungan kegiatan praktikum dan penelitian secara optimal.

1.2. Perumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, terdapat pertanyaan yang menjadi fokus dalam proyek akhir ini yaitu, Bagaimana cara merancang dan sistem informasi manajemen peminjaman alat laboratorium yang berbasis digital dengan menggunakan metode *Rapid Application Development* (RAD) dalam meningkatkan efisiensi dalam proses peminjaman alat laboratorium serta meminimalisir resiko penginputan data yang tidak tersistematis?

1.3. Tujuan Proyek Akhir

Proyek akhir ini bertujuan untuk merancang sistem informasi manajemen peminjaman peralatan laboratorium yang berbasis digital dengan menggunakan metode *Rapid Application Development* (RAD) agar proses peminjaman peralatan laboratorium menjadi lebih efisien serta meminimalisir resiko penginputan data yang tidak tersistematis.

1.4. Manfaat

Manfaat dari proyek akhir ini diharapkan dapat dirasakan oleh mahasiswa, dosen, dan pengelola laboratorium yang terlibat dalam proses peminjaman alat laboratorium serta diharapkan dapat meningkatkan kualitas pelayanan peminjaman alat di laboratorium.

BAB II

DASAR TEORI

Dasar Teori ini menyajikan landasan teoritis dan empiris yang relevan dengan penelitian pengembangan Sistem Informasi Manajemen Peminjaman Peralatan Laboratorium. Pembahasan dalam bab ini mencakup konsep-konsep dasar sistem informasi, manajemen peralatan laboratorium, metode pengembangan perangkat lunak *Rapid Application Development (RAD)*, serta teknologi pendukung yang digunakan, yaitu PHP, *framework* Laravel, MySQL, phpMyAdmin, Laragon, Visual Studio Code, Bootstrap, *web hosting*, dan QRCode. Bagian ini juga menyajikan tinjauan terhadap penelitian-penelitian terdahulu yang relevan untuk menunjukkan posisi dan kontribusi penelitian ini dalam konteks ilmiah.

2.1. Penelitian Terdahulu

Sejumlah penelitian telah dilakukan dalam upaya mengembangkan sistem informasi peminjaman atau inventarisasi alat laboratorium dengan pemanfaatan teknologi digital. Mayoritas penelitian tersebut memberikan dampak positif pada efisiensi pencatatan dan pengelolaan alat laboratorium, serta menawarkan langkah awal digitalisasi proses manual yang selama ini menjadi kendala di lingkungan pendidikan vokasi.

Penelitian yang dilakukan oleh (Ashari & Burhan, 2023) mengenai aplikasi peminjaman alat berbasis Android dengan QRCode berhasil meningkatkan efisiensi peminjaman, mempercepat proses pencatatan, dan mengurangi kesalahan manual. Keunggulan utama penelitian ini adalah memanfaatkan teknologi QRCode untuk mempercepat identifikasi alat. Namun demikian, keterbatasan sistem yang hanya berbasis perangkat Android menyebabkan solusi ini kurang fleksibel untuk mengakomodasi akses lintas platform, terlebih apabila sistem perlu diakses oleh banyak pengguna secara bersamaan melalui *web*. Selain itu, pengembangan sistem belum menggunakan suatu metode pengembangan perangkat lunak yang iteratif

dan adaptif, sehingga pembaruan serta penambahan fitur bergantung pada proses pengembangan yang relatif konvensional.

Selanjutnya, penelitian yang dikaji oleh (Jaya, Susilowati, & Akhdiyati, 2023) mengangkat sistem informasi QRCode untuk inventarisasi dan peminjaman alat laboratorium yang berbasis *web*. Sistem ini membuktikan kemudahan dalam penyimpanan data serta pemberian tutorial penggunaan alat bagi pengguna baru, yang membedakannya dari sistem manual. Kelebihannya terletak pada integrasi QRCode dan akses berbasis cloud yang lebih terjangkau. Namun, penelitian ini masih memiliki keterbatasan dalam aspek pelaporan, monitoring status peminjaman secara real-time, dan belum mengintegrasikan fitur notifikasi otomatis sehingga potensi keterlambatan pengembalian alat relatif tetap tinggi.

Adapun penelitian (Suhanda, Rahmi, Efrinalia, & Ariska, 2025) tentang sistem inventarisasi laboratorium berbasis QRCode terintegrasi dengan cloud memastikan pelacakan yang cepat dan akurat serta pelaporan secara langsung. Walaupun sistem ini menawarkan efisiensi pelacakan, kekurangan penelitian ini tampak pada aspek aksesibilitas dan fleksibilitas pengembangan, karena tidak seluruh proses pengelolaan laboratorium seperti prosedur peminjaman, pengembalian, dan pelaporan kerusakan diintegrasikan dalam sistem secara komprehensif serta belum terdapat metode pengembangan berbasis iteratif yang memadai.

Kekurangan-kekurangan pada penelitian terdahulu tersebut memberikan peluang dan ruang inovasi bagi peneliti untuk menghasilkan penyempurnaan sistem informasi manajemen peminjaman alat laboratorium. Adapun kesenjangan yang diisi dalam penelitian ini, di antaranya adalah:

Pertama, sistem yang dikembangkan mengintegrasikan metode *Rapid Application Development* (RAD) yang terbukti lebih adaptif, iteratif, serta melibatkan pengguna secara aktif dalam setiap siklus pengembangan. Hal ini memastikan bahwa setiap kebutuhan dan perubahan dapat segera ditindaklanjuti sehingga sistem cepat beradaptasi dengan situasi laboratorium yang dinamis.

Kedua, penggunaan *framework* Laravel pada penelitian ini menjadikan sistem memiliki arsitektur yang lebih terstruktur, aman, mudah dikembangkan, dan mendukung operasional lintas platform berbasis *web*. *Framework* ini memudahkan pengelolaan data, pengembangan fitur lebih lanjut, serta integrasi dengan sistem informasi kampus yang lain.

Ketiga, penelitian ini tidak hanya mengadopsi QRCode untuk identifikasi digital alat, namun juga menambah fitur notifikasi otomatis yang terintegrasi dengan prosedur peminjaman, pengembalian, serta pelaporan kerusakan alat. Fitur notifikasi ini memberikan efisiensi dan kepastian informasi kepada pengguna terkait batas waktu pengembalian alat maupun informasi status peralatan, yang selama ini luput dalam sistem pada penelitian terdahulu.

Dengan demikian, letak kebaruan pada penelitian ini yaitu mempertemukan keunggulan berbagai penelitian terdahulu serta mendayagunakan metode pengembangan, platform teknologi, dan fitur manajemen laboratorium secara lebih komprehensif, adaptif, serta terintegrasi dengan kebutuhan aktual operasional laboratorium pendidikan vokasi. Pendekatan ini diharapkan mampu memberikan kontribusi nyata dalam peningkatan efisiensi, akurasi, dan akuntabilitas pengelolaan alat laboratorium pendidikan secara digital dan terstruktur.

2.2. Sistem Informasi Manajemen

Sistem informasi manajemen (SIM) didefinisikan sebagai sekumpulan komponen yang saling terkait, bekerja sama untuk mengumpulkan, menyimpan, memproses, dan mendistribusikan informasi guna mendukung pengambilan keputusan dan pengendalian dalam suatu organisasi. (Kenneth C. Laudon, 2021) menjelaskan bahwa SIM mencakup perangkat lunak, perangkat keras, prosedur, serta sumber daya manusia yang berperan dalam pengelolaan data secara efektif. Implementasi SIM dalam konteks laboratorium sangat berperan dalam meningkatkan efisiensi dan akurasi pencatatan inventaris. Dalam konteks pendidikan, SIM juga mendukung pengelolaan aset, termasuk proses peminjaman dan pengembalian peralatan, serta pelacakan status barang yang sebelumnya dilakukan secara manual dan rentan terhadap kesalahan pencatatan (Eriya, Agus, Maulana, & Risna, 2020).

2.3. Manajemen Peralatan Laboratorium

Manajemen laboratorium meliputi perencanaan, pengadaan, pemeliharaan, serta pengawasan peralatan untuk menjamin kelayakan fungsi dan optimalisasi penggunaannya. Buku “Manajemen Sarana dan Prasarana Pendidikan” menyatakan bahwa laboratorium merupakan bagian dari sarana pembelajaran yang harus dikelola secara profesional agar mampu mendukung efektivitas proses belajar mengajar (Hayati, et al., 2025). Menurut (Eriya, Agus, Maulana, & Risna, 2020), penerapan sistem manajemen inventaris berbasis QRCode terbukti efektif dalam mengelola peralatan laboratorium karena mampu meminimalkan kesalahan *input* dan mempercepat proses pencatatan serta pencarian informasi. Sistem ini juga mendukung efisiensi waktu serta akuntabilitas data yang terekam.

2.4. Metodologi *Rapid Application Development* (RAD)

Rapid Application Development (RAD) merupakan salah satu model dalam *Software Development Life Cycle* (SDLC) yang mengutamakan kecepatan pengembangan dengan menggunakan pendekatan iteratif dan prototipe berkelanjutan. Berdasarkan studi literatur (Parlika, Afifudin, Pradana, Wiratama, & Holis., 2023), RAD dianggap sangat efisien untuk pengembangan sistem yang dinamis karena mampu menghasilkan perangkat lunak dalam waktu singkat serta sangat adaptif terhadap perubahan kebutuhan pengguna.

Tahapan umum dalam metode RAD meliputi:

1. **Perencanaan Kebutuhan (*Requirements Planning*):** Tahap awal untuk mendefinisikan ruang lingkup dan persyaratan proyek.
2. **Desain Sistem (*User Design*):** Melibatkan pengembangan prototipe secara interaktif dengan masukan aktif dari pengguna.
3. **Pengembangan (*Construction*):** Tahap pembangunan sistem inti berdasarkan prototipe yang telah disetujui.
4. **Implementasi (*Cutover*):** Tahap penempatan sistem ke lingkungan produksi dan pelatihan pengguna.

Keunggulan metode ini adalah keterlibatan aktif pengguna selama proses pengembangan serta kemampuan iteratif yang tinggi dalam menyesuaikan kebutuhan aktual di lapangan.

2.5. Teknologi Pendukung Sistem

2.5.1. PHP

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang berjalan pada sisi peladen (*server-side*) dan secara luas digunakan dalam pengembangan aplikasi *web* dinamis. PHP bersifat *open-source*, mudah dipelajari, dan memiliki sintaks yang sederhana, sehingga cocok digunakan untuk membangun berbagai jenis sistem informasi, termasuk sistem manajemen berbasis *web*.

Dalam penelitian yang dilakukan oleh (Madurapperuma, Shafana, & Sabani, 2022), PHP terbukti efektif dalam pengembangan sistem informasi karena kemampuannya dalam memproses permintaan pengguna dengan cepat, serta dukungan komunitas yang luas yang memudahkan pemeliharaan dan pengembangan berkelanjutan.

Penggunaan PHP dalam proyek ini dipadukan dengan *framework* Laravel untuk mempermudah strukturisasi kode dan meningkatkan keamanan serta efisiensi pengembangan.

2.5.2. Laravel Framework

Laravel adalah *framework* berbasis PHP yang menggunakan arsitektur *Model-View-Controller* (MVC). Laravel menyediakan struktur kode yang bersih, modular, dan mudah untuk dipelihara. Laravel memiliki berbagai fitur unggulan seperti sistem *routing*, *middleware*, *ORM Eloquent*, serta autentikasi pengguna yang telah tersedia secara bawaan.

Menurut penelitian (Madurapperuma, Shafana, & Sabani, 2022), penggunaan Laravel dapat meningkatkan efisiensi pengembangan sistem, mempercepat proses integrasi data, serta mempermudah pemeliharaan aplikasi *web* dalam jangka panjang. Dalam pengembangan sistem ini, Laravel digunakan sebagai kerangka kerja utama untuk membangun logika sistem dan mengatur alur data antara antarmuka pengguna dan basis data.

2.5.3. Database MySQL

MySQL merupakan sistem manajemen basis data relasional (*Relational Database Management System*) yang bersifat *open-source*. MySQL banyak digunakan dalam pengembangan aplikasi berbasis *web* karena memiliki performa tinggi, stabil, dan mendukung pengelolaan data berskala besar.

Dalam sistem ini, MySQL digunakan untuk menyimpan data pengguna, data alat, data transaksi peminjaman dan pengembalian, serta riwayat log aktivitas. Struktur data yang digunakan dirancang agar mendukung hubungan antar tabel secara efisien dan aman.

2.5.4. PhpMyAdmin

PhpMyAdmin merupakan sebuah aplikasi berbasis *web* yang dirancang untuk mempermudah pengelolaan basis data MySQL atau MariaDB melalui antarmuka grafis. Aplikasi ini memungkinkan pengguna untuk melakukan berbagai operasi pengelolaan data, seperti membuat dan memodifikasi tabel, menjalankan perintah *Structured Query Language* (SQL), mengelola relasi antar-tabel, serta mencadangkan dan memulihkan data, tanpa perlu menuliskan perintah SQL secara manual melalui *command line interface* (CLI).

Menurut (Kenneth C. Laudon, 2021), sistem pengelolaan basis data berbasis antarmuka grafis seperti phpMyAdmin sangat mendukung efisiensi dalam administrasi data karena mampu menjembatani kebutuhan teknis pengguna non-programmer sekalipun. Selain itu, phpMyAdmin juga dilengkapi fitur keamanan seperti otentikasi pengguna, pengelolaan hak akses, dan dukungan terhadap protokol enkripsi untuk menjaga integritas data.

Kemampuan phpMyAdmin yang ringan, fleksibel, serta kompatibel lintas sistem operasi menjadikannya alat yang banyak digunakan dalam pengembangan sistem informasi, terutama pada tahap implementasi dan pemeliharaan basis data.

2.5.5. Laragon

Laragon merupakan perangkat lunak penyedia lingkungan pengembangan lokal (*local development environment*) yang bersifat portabel, ringan, cepat, dan mudah dikonfigurasi. Laragon mendukung berbagai komponen *server* seperti Apache,

Nginx, MySQL, MariaDB, PHP, dan Node.js, serta menyediakan integrasi otomatis dengan *Composer*, *Terminal Console*, dan *auto virtual host* yang sangat memudahkan pengembangan aplikasi berbasis *web*, khususnya yang menggunakan *framework* Laravel (Laragon Documentation, 2024).

Laragon dirancang untuk memberikan pengalaman pengembangan yang efisien dan nyaman, dengan proses instalasi dan konfigurasi yang minim. Keunggulannya terletak pada kemampuannya untuk menangani berbagai versi PHP secara paralel, serta mendukung *isolasi proyek* agar pengembang dapat bekerja dengan lebih rapi dan terstruktur.

Dalam proyek ini, Laragon digunakan sebagai *server* lokal (*local server environment*) untuk memfasilitasi proses perancangan, pengembangan, dan pengujian sistem secara offline sebelum dipublikasikan ke *web hosting*. Penggunaan Laragon memungkinkan pengujian sistem berjalan stabil dan konsisten di lingkungan yang menyerupai *server* produksi.

2.5.6. Visual Studio Code

Visual Studio Code (VS Code) adalah penyunting kode sumber (*source code editor*) yang bersifat *open-source* dan dikembangkan oleh Microsoft. Aplikasi ini dirancang untuk mendukung pengembangan perangkat lunak secara efisien dan fleksibel di berbagai sistem operasi.

VS Code memiliki berbagai fitur unggulan seperti *syntax highlighting*, *auto completion* berbasis *IntelliSense*, *integrated terminal*, serta manajemen *Git* yang terintegrasi. Selain itu, dukungan terhadap ekstensi seperti PHP Intelephense, Laravel Blade Snippets, dan Live Server memungkinkan pengembang untuk bekerja secara produktif dalam membangun aplikasi berbasis PHP, Laravel, dan JavaScript.

Dalam penelitian ini, VS Code digunakan sebagai lingkungan utama dalam proses penulisan, debugging, dan pengelolaan kode sumber sistem informasi yang dikembangkan.

2.5.7. Bootstrap

Bootstrap merupakan *framework* antarmuka (*frontend*) berbasis HTML, CSS, dan JavaScript yang dirancang untuk membangun tampilan situs *web* yang responsif dan menarik. Bootstrap menyediakan berbagai komponen antarmuka siap pakai, seperti tombol, formulir, tabel, dan sistem tata letak (*grid system*), yang dapat digunakan untuk mempercepat proses perancangan tampilan pengguna (*user interface*).

Dalam sistem ini, Bootstrap digunakan untuk memastikan antarmuka sistem dapat ditampilkan secara optimal pada berbagai perangkat, baik komputer, tablet, maupun ponsel pintar. Penggunaan Bootstrap juga mendukung prinsip antarmuka pengguna yang konsisten dan mudah digunakan (Madurapperuma, Shafana, & Sabani, 2022).

2.5.8. Web Hosting

Web hosting adalah layanan penyimpanan data dan file situs *web* di dalam *server* agar dapat diakses secara daring melalui jaringan internet. Layanan *hosting* yang andal memungkinkan sistem dapat diakses secara cepat, aman, dan tersedia selama 24 jam.

Dalam sistem ini, *web hosting* digunakan untuk menyimpan aplikasi berbasis *web* yang telah dikembangkan, sehingga dapat digunakan oleh admin, pengurus laboratorium, dan peminjam di mana saja dan kapan saja. *Hosting* yang dipilih harus mendukung teknologi PHP, MySQL, dan memberikan akses kontrol melalui panel administrasi seperti cPanel. Menurut (Madurapperuma, Shafana, & Sabani, 2022), pemilihan *web hosting* yang tepat dapat meningkatkan ketersediaan layanan serta keamanan data pengguna.

2.5.9. QRCode

Quick Response Code (QRCode) merupakan representasi data dalam bentuk pola visual dua dimensi yang dapat dibaca dengan perangkat pemindai atau kamera digital. Teknologi ini dirancang untuk mempercepat proses identifikasi dan pelacakan objek secara otomatis, serta mengurangi kesalahan input data yang umum terjadi pada proses manual.

Dalam konteks sistem informasi manajemen, QRCode telah banyak digunakan pada sektor inventarisasi, logistik, dan manajemen aset karena kemampuannya dalam menyimpan data yang dapat diakses dengan cepat dan akurat. Integrasi QRCode dalam sistem informasi inventaris terbukti meningkatkan efisiensi pencatatan dan mempercepat proses validasi data.

Menurut (Eriya, Agus, Maulana, & Risna, 2020), integrasi QRCode dalam sistem manajemen inventaris mampu meningkatkan efisiensi pencatatan dan akurasi data. Implementasi QRCode juga telah diterapkan pada conveyor dan logistik ekspedisi, menunjukkan fleksibilitas penggunaannya.

2.5.10. Notifikasi Otomatis

Notifikasi otomatis merupakan fitur tambahan yang digunakan untuk mengingatkan pengguna terhadap suatu kejadian atau batas waktu tertentu secara otomatis, tanpa intervensi langsung dari administrator sistem. Dalam konteks sistem informasi manajemen peminjaman peralatan laboratorium, notifikasi otomatis digunakan sebagai pengingat keterlambatan pengembalian alat oleh peminjam.

Sistem ini menggunakan layanan pengiriman surat elektronik (*email notification*) untuk mengirim pesan peringatan kepada peminjam yang belum mengembalikan alat sesuai batas waktu yang telah ditentukan. Implementasi fitur ini memanfaatkan beberapa komponen teknologi sebagai berikut:

- 1) Layanan SMTP (*Simple Mail Transfer Protocol*): Digunakan sebagai jalur pengiriman *email* dari sistem ke alamat surat elektronik pengguna. Sistem dikonfigurasi agar terhubung dengan layanan SMTP eksternal seperti Gmail atau Mailtrap.
- 2) Fitur Laravel Mail: Laravel menyediakan pustaka bawaan untuk mengelola pengiriman *email*, termasuk format isi pesan, alamat tujuan, serta penggunaan *template* yang dapat disesuaikan.
- 3) Laravel Scheduler dan Cron Job: Digunakan untuk menjalankan pemeriksaan rutin terhadap data transaksi peminjaman. Jika ditemukan keterlambatan, maka sistem akan memicu pengiriman notifikasi secara otomatis pada waktu yang telah dijadwalkan.

Dengan adanya sistem notifikasi otomatis ini, pengguna akan mendapatkan pengingat secara langsung melalui surat elektronik mengenai keterlambatan pengembalian, sehingga dapat meningkatkan kesadaran dan kedisiplinan dalam menjaga ketertiban proses peminjaman. Selain itu, fitur ini juga membantu pengurus laboratorium dalam mengurangi pemantauan manual dan mempercepat penanganan terhadap transaksi yang terlambat.

2.6. UML

Unified Modeling Language (UML) merupakan standar pemodelan visual yang digunakan secara luas dalam rekayasa perangkat lunak untuk memvisualisasikan, menspesifikasikan, membangun, dan mendokumentasikan sistem secara terstruktur. UML dirancang sebagai sarana komunikasi yang efektif antara pengembang dan pemangku kepentingan (*stakeholder*) melalui serangkaian diagram yang menggambarkan struktur maupun perilaku sistem. Beberapa jenis diagram yang umum digunakan antara lain *use case diagram*, *activity diagram*, dan *class diagram*.

Dalam penelitian ini, *UML* digunakan sebagai alat bantu utama dalam merancang *Sistem Informasi Manajemen Peminjaman Peralatan Laboratorium*. Tiga jenis diagram yang digunakan meliputi *use case diagram*, *activity diagram*, dan *class diagram*, yang masing-masing memiliki peran penting dalam proses perancangan sistem.

1) *Use Case Diagram*

Diagram ini menggambarkan hubungan antara aktor (pengguna) dan sistem dalam bentuk skenario interaksi. Fungsinya adalah untuk mengidentifikasi dan memetakan fitur utama yang dibutuhkan oleh masing-masing peran pengguna, yaitu admin, pengurus laboratorium, dan peminjam.

2) *Activity Diagram*

Digunakan untuk memvisualisasikan urutan aktivitas atau alur kerja sistem, baik yang dilakukan oleh pengguna maupun sistem itu sendiri. Diagram ini menunjukkan langkah-langkah logis yang terjadi dalam proses, mulai dari awal hingga akhir, secara rinci.

3) *Class Diagram*

Diagram ini digunakan untuk memodelkan struktur statis dari sistem dalam bentuk kelas-kelas beserta atribut, metode, dan relasi antar kelas. *Class diagram* sangat penting dalam menggambarkan keterkaitan antar entitas dalam sistem serta sebagai dasar implementasi kode.

Efektivitas penggunaan UML telah dibuktikan dalam berbagai studi. Dalam penelitian oleh (Dewi & Voutama, 2024) memanfaatkan *use case* dan *activity diagram* untuk menggambarkan alur interaksi pengguna pada sistem informasi keuangan masjid berbasis *web*, yang terbukti membantu dalam memperjelas kebutuhan sistem dan memudahkan proses pengembangan. Sementara itu, (Hamdani & Saptanji, 2020) menggunakan UML sebagai bagian dari pendekatan *Rational Unified Process* (RUP) dalam pengembangan sistem informasi geografis, yang menghasilkan desain sistem yang terstruktur dan efisien. Selain itu, (Aryani, Aqil, & Paramita, 2024) dalam penelitiannya mengenai digitalisasi sistem informasi perpustakaan menyatakan bahwa UML, khususnya *use case diagram*, berperan penting dalam perancangan arsitektur aplikasi yang dapat diakses lintas perangkat secara konsisten.

Berdasarkan literatur tersebut, penerapan UML dalam penelitian ini mendukung aspek-aspek berikut:

1. Analisis kebutuhan, memetakan aktor dan fungsionalitas sistem.
2. Perancangan system, memvisualisasikan alur proses dan struktur data.
3. Komunikasi tim, memfasilitasi diskusi antara pengembang, pengguna, dan stakeholder.
4. Implementasi system, menjadi acuan yang jelas selama pembangunan dan pengujian.

Dengan demikian, penggunaan UML pada sistem informasi ini mendukung pengembangan yang lebih sistematis, terdokumentasi dengan baik, serta dapat ditindaklanjuti untuk pengembangan berkelanjutan di masa mendatang.

2.6.1. Use Case Diagram

Use case diagram merupakan salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan hubungan dan interaksi antara aktor (pengguna) dengan sistem dalam suatu skenario tertentu. Diagram ini memvisualisasikan fungsionalitas utama yang dapat diakses oleh masing-masing peran pengguna, serta menggambarkan batasan sistem secara keseluruhan (*system boundary*).

Menurut (Dewi & Voutama, 2024) dalam penelitian mereka mengenai pengembangan sistem informasi keuangan berbasis *web*, *use case diagram* efektif digunakan untuk menjabarkan peran dan tanggung jawab setiap aktor dalam sistem, sehingga mempermudah pemetaan kebutuhan fungsional dan penyusunan prototipe berbasis peran.

Dalam penelitian ini, *use case diagram* digunakan untuk mengidentifikasi dan memetakan fitur-fitur utama berdasarkan peran pengguna, yakni admin, pengurus laboratorium (PLP), dan peminjam (mahasiswa atau dosen). Diagram ini menjadi acuan awal dalam proses analisis kebutuhan, karena membantu pengembang memahami skenario interaksi antar pengguna dan sistem secara menyeluruh.

2.6.2. Activity Diagram

Activity diagram merupakan salah satu jenis diagram UML yang digunakan untuk memodelkan alur aktivitas dalam sistem secara logis dan berurutan. Diagram ini menggambarkan proses bisnis atau aliran kerja dalam sistem informasi, baik yang dilakukan oleh pengguna maupun sistem itu sendiri.

Menurut (Hamdani & Saptanji, 2020), *activity diagram* sangat berguna untuk menggambarkan urutan aktivitas dalam sistem informasi geografis, karena mampu menunjukkan proses pengambilan keputusan, percabangan, maupun aliran data dengan jelas. Sementara itu, (Aryani, Aqil, & Paramita, 2024) juga menggunakan *activity diagram* dalam pengembangan sistem informasi perpustakaan berbasis *web* untuk menggambarkan alur peminjaman dan pengembalian buku secara sistematis. Pada penelitian ini, *activity diagram* digunakan untuk menggambarkan proses-proses inti dalam sistem, seperti:

1. Proses peminjaman alat,
2. Proses pengembalian alat,
3. Manajemen data alat oleh pengurus,
4. Pengelolaan pengguna oleh admin,
5. Validasi permintaan oleh PLP.

Dengan memanfaatkan *activity diagram*, pengembangan sistem menjadi lebih sistematis, karena setiap langkah kerja (workflow) dapat divisualisasikan dan diuji kembali sebelum diimplementasikan ke dalam kode program. Diagram ini juga berfungsi sebagai dokumentasi visual yang dapat digunakan untuk validasi kebutuhan dengan pengguna atau tim pengembang lainnya.

2.6.3. Class Diagram

Class diagram merupakan salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk memodelkan struktur statis dari sistem, berupa kelas-kelas beserta atribut, metode, dan hubungan antar kelas. Diagram ini berfungsi untuk menggambarkan bagaimana entitas dalam sistem saling berinteraksi dan bagaimana data dikelola secara terstruktur.

Menurut (Dewi & Voutama, 2024), *class diagram* menjadi fondasi dalam perancangan sistem berbasis objek, karena memberikan representasi visual terhadap entitas dan keterhubungannya, yang akan memudahkan dalam implementasi kode serta pemeliharaan sistem.

Dalam proyek ini, *class diagram* disusun untuk menggambarkan struktur entitas utama seperti pengguna (*user*), alat, peminjaman, pengembalian, dan log aktivitas. Setiap kelas memiliki atribut (seperti nama, ID, status, tanggal) dan metode (seperti *login*, validasi, tambah data, dan hapus data) yang mendukung proses bisnis sistem informasi manajemen peminjaman alat laboratorium.

Diagram ini menjadi acuan penting dalam tahap implementasi, karena memastikan setiap entitas telah terdefinisi dengan baik dan hubungan antar kelas telah ditetapkan sesuai kebutuhan sistem.

2.7. Pengujian

Pengujian perangkat lunak merupakan tahap krusial dalam rekayasa perangkat lunak yang bertujuan memverifikasi dan memvalidasi sistem agar sesuai kebutuhan pengguna dan bebas kesalahan sebelum dioperasikan. Secara umum, dua metode utama pengujian yang digunakan adalah *Black Box Testing* dan *User Acceptance Testing*.

2.7.1. *Black Box Testing*

Black Box Testing adalah metode pengujian perangkat lunak yang berfokus pada pengamatan terhadap keluaran (*output*) berdasarkan masukan (*input*) tertentu, tanpa memerhatikan struktur internal atau logika program. Pengujian ini difokuskan pada aspek fungsionalitas sistem, untuk memastikan bahwa fitur-fitur yang diimplementasikan telah berjalan sesuai dengan spesifikasi.

Dalam penelitian oleh (Jailani & Yaqin, 2024), metode *Black Box* dengan teknik *Boundary Value Analysis* (BVA) diterapkan untuk menguji *input* nilai dalam sistem informasi akademik. Hasilnya menunjukkan bahwa teknik ini efektif dalam mendeteksi kesalahan pada nilai ekstrem dan membantu meningkatkan keandalan sistem terhadap berbagai skenario *input*.

Sementara itu, (Aji, Havis, Ridwan, Mulia, & Atmaja, 2025) dalam pengujian sistem pendaftaran online RSUD Kota Bandung juga menerapkan teknik BVA dan menemukan bahwa pendekatan ini berhasil mengungkap kesalahan validasi data penting seperti NIK, nomor BPJS, serta tanggal lahir yang tidak terdeteksi dengan metode pengujian umum.

Beberapa teknik populer dalam *Black Box Testing* antara lain:

- *Equivalence Partitioning*: Membagi *input* menjadi kelompok yang diharapkan menghasilkan perilaku sistem serupa.
- *Boundary Value Analysis* (BVA): Fokus pada nilai di batas bawah dan atas dari rentang *input*.
- *Cause-Effect Graphing*: Mengidentifikasi hubungan sebab-akibat antara *input* dan *output*.

- Error Guessing: Mengandalkan pengalaman untuk menebak skenario kesalahan potensial.

Penggunaan kombinasi teknik-teknik ini memungkinkan pengujian sistem yang menyeluruh dan sistematis, memastikan bahwa seluruh fungsi utama bekerja sesuai dengan ekspektasi pengguna.

2.7.2. User Acceptance Testing

User Acceptance Testing (UAT) merupakan pengujian tingkat akhir yang dilakukan oleh pengguna akhir (*end-user*) guna menilai apakah sistem yang dikembangkan telah memenuhi kebutuhan dan layak digunakan dalam lingkungan operasional. UAT bertujuan mengevaluasi aspek fungsional, kenyamanan, efisiensi, serta pengalaman penggunaan sistem secara langsung.

Menurut (Aliyah, Hartono, & Muin, 2025) menjelaskan bahwa UAT dilakukan untuk memastikan kelayakan sistem berdasarkan pengalaman pengguna. Pengujian dilakukan menggunakan kuesioner berbasis skala Likert dengan rentang 1–5, yang mewakili tingkat persetujuan terhadap pernyataan tertentu.

Menurut hasil penelitian tersebut, sistem dikategorikan “sangat layak” apabila memperoleh persentase kelayakan di atas 80%, “layak” pada rentang 61–80%, dan “perlu ditingkatkan” apabila berada di bawah 60%.

Dengan metode ini, pengukuran persepsi pengguna dapat dijadikan dasar kuat dalam pengambilan keputusan terkait kesiapan implementasi sistem secara menyeluruh. Selain itu, UAT juga dapat mendeteksi fitur yang kurang intuitif, responsivitas sistem, serta kekurangan yang tidak terdeteksi melalui pengujian teknis semata.

BAB III

METODE PELAKSANAAN

3.1. Jenis Penelitian

Proyek akhir ini merupakan jenis penelitian terapan yang berfokus pada pengembangan *sistem informasi* berbasis *web*. Penelitian ini bersifat aplikatif karena menghasilkan produk nyata berupa perangkat lunak yang dapat digunakan langsung oleh pengguna untuk menyelesaikan permasalahan tertentu di lingkungan kerja.

Sistem yang dikembangkan berjudul “Sistem Informasi Manajemen Peminjaman Peralatan Laboratorium dengan Metode *Rapid Application Development (RAD)*”, yang bertujuan untuk membantu proses pencatatan, pelacakan, dan pengelolaan peminjaman peralatan laboratorium agar lebih efisien, akurat, dan terorganisasi.

Pengembangan sistem ini dilatarbelakangi oleh adanya permasalahan nyata yang dihadapi oleh pengelola laboratorium, khususnya terkait proses pendataan alat, pengajuan dan validasi peminjaman, serta pemantauan kondisi alat secara manual yang dinilai kurang efektif. Oleh karena itu, sistem ini dirancang untuk merespons kebutuhan tersebut melalui pendekatan berbasis teknologi informasi yang terintegrasi.

3.2. Metodologi Pengembangan Sistem

Metode yang digunakan dalam pengembangan sistem ini adalah *Rapid Application Development (RAD)*. Metode RAD dipilih karena memiliki karakteristik pengembangan yang cepat, fleksibel, serta melibatkan pengguna secara langsung melalui proses *prototyping* dan umpan balik (*feedback*) berkelanjutan.

Metode ini menekankan pendekatan iteratif dalam proses pengembangan, sehingga sistem yang dihasilkan dapat disesuaikan secara dinamis dengan kebutuhan pengguna di lapangan. Adapun tahapan utama dalam metode RAD yang digunakan dalam proyek ini meliputi:



Gambar 3.2-1 Metode RAD

1. Perencanaan Kebutuhan (*Requirement Planning*):
Tahap ini dilakukan untuk mengidentifikasi kebutuhan sistem melalui observasi langsung dan wawancara dengan pihak pengelola laboratorium. Informasi yang dikumpulkan mencakup kebutuhan fungsional sistem, jenis peran pengguna (admin, peminjam, dan pengurus laboratorium /PLP), serta alur kerja peminjaman alat yang saat ini diterapkan secara manual.
2. Desain Pengguna (*User Design*):
Tahapan ini mencakup perancangan antarmuka pengguna (*user interface*), alur proses peminjaman dan pengembalian alat, serta struktur basis data sistem. Rancangan dibuat dalam bentuk *prototipe awal* yang disusun secara iteratif dan divalidasi oleh pengguna untuk memastikan kesesuaian dengan kebutuhan aktual di lapangan.
3. Pengembangan (*Construction*):
Pada tahap ini, sistem mulai dikembangkan menggunakan Laravel sebagai *framework* utama. Kode program ditulis berdasarkan hasil desain yang telah disetujui, mencakup pembuatan fungsionalitas utama seperti *login* pengguna, manajemen data alat, peminjaman dan pengembalian alat, serta integrasi QRCode dan notifikasi otomatis.

4. Implementasi (*Cutover*):

Tahap implementasi merupakan tahap akhir dari metode RAD, di mana sistem diuji secara langsung oleh pengguna akhir untuk memastikan bahwa seluruh fungsi berjalan sesuai kebutuhan. Pengujian dilakukan menggunakan metode *Black Box Testing* untuk menguji setiap fitur, serta *User Acceptance Testing* (UAT) untuk mengevaluasi tingkat penerimaan dan kenyamanan pengguna terhadap sistem. Hasil pengujian didokumentasikan sebagai bagian dari evaluasi akhir pengembangan sistem.

3.3. Diagram Alir Proses Pengembangan Sistem

Diagram alir proses pengembangan sistem digunakan untuk menggambarkan tahapan kerja yang dilakukan selama proyek akhir, mulai dari tahap awal identifikasi masalah hingga penyusunan laporan akhir. Setiap tahap dilakukan secara terstruktur untuk memastikan bahwa sistem yang dikembangkan benar-benar sesuai dengan kebutuhan pengguna dan dapat diimplementasikan secara optimal.

Langkah-langkah utama yang dilakukan dalam proyek ini adalah sebagai berikut:

1. Identifikasi dan Analisis Masalah

Pada tahap ini dilakukan observasi dan diskusi dengan pihak laboratorium untuk mengidentifikasi masalah yang muncul dalam proses peminjaman peralatan secara manual, seperti pencatatan yang tidak terorganisir, tidak adanya notifikasi pengembalian, serta keterbatasan akses data oleh pengguna.

2. Pengumpulan Data

Data dikumpulkan melalui wawancara dengan Pengurus Laboratorium (PLP), observasi proses peminjaman, serta studi dokumen seperti SOP laboratorium dan logbook peminjaman sebelumnya. Informasi ini menjadi dasar untuk menentukan kebutuhan fungsional dan non-fungsional sistem.

3. Perancangan Sistem

Tahap ini mencakup pembuatan *flowchart*, *use case diagram*, *activity diagram*, *class diagram*, serta desain antarmuka pengguna. Tujuan dari

tahapan ini adalah untuk memberikan gambaran visual dan teknis sebelum proses implementasi sistem dilakukan.

4. Implementasi Sistem

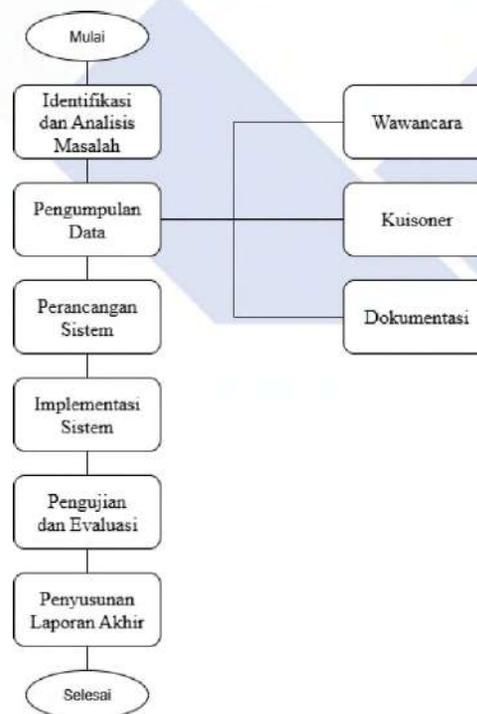
Berdasarkan hasil perancangan, sistem mulai dikembangkan menggunakan Laravel sebagai *framework backend* dan MySQL sebagai *database*. Proses ini juga mencakup pengintegrasian fitur QRCode, autentikasi pengguna, dan notifikasi pengembalian.

5. Pengujian dan Evaluasi

Pengujian dilakukan dengan metode *Black Box* untuk memastikan setiap fitur berjalan sesuai fungsinya. UAT (*User Acceptance Testing*) juga dilakukan oleh pengguna langsung untuk mengevaluasi kenyamanan dan kegunaan sistem.

6. Penyusunan Laporan Akhir

Setelah sistem diuji dan berjalan dengan baik, hasil pengembangan didokumentasikan dalam bentuk laporan proyek akhir sebagai bagian dari syarat kelulusan program studi.



Gambar 3.3-1 flowchart Proses Pengembangan Sistem

3.4. Teknik Pengumpulan Data

Pengumpulan data merupakan proses sistematis dalam memperoleh informasi yang dibutuhkan untuk merancang sistem secara tepat sasaran dan sesuai dengan kebutuhan pengguna. Menurut Sugiyono (2017), pengumpulan data adalah langkah penting dalam penelitian yang berfungsi untuk memperoleh data yang valid dan reliabel, sehingga hasil yang diperoleh dapat dipertanggungjawabkan secara ilmiah. Dalam proyek ini, pengumpulan data dilakukan melalui tiga teknik utama, yaitu wawancara, kuesioner, dan dokumentasi. Ketiga teknik ini digunakan untuk memahami kebutuhan sistem dari para pengguna secara langsung serta untuk memastikan bahwa sistem yang dirancang benar-benar sesuai dengan kondisi operasional laboratorium.

1. Wawancara

Wawancara dilakukan secara langsung dengan Pengurus Laboratorium Pendidikan (PLP) sebelum tahap pengembangan sistem dimulai. Tujuan wawancara ini adalah untuk menggali informasi mendalam mengenai kebutuhan aktual pengguna terhadap sistem informasi yang akan dibangun, serta memperoleh pemahaman mengenai alur kerja peminjaman peralatan yang berlaku di laboratorium.

2. Kuisoner

Kuesioner digunakan sebagai instrumen untuk mengumpulkan umpan balik dari pengguna sistem terkait pengalaman mereka dalam menggunakan aplikasi yang telah dikembangkan. Kuesioner ini disusun berdasarkan aspek-aspek fungsionalitas, kemudahan penggunaan, tampilan antarmuka, kecepatan akses, serta kepuasan secara umum terhadap sistem. Data yang diperoleh melalui kuesioner digunakan untuk mendukung proses evaluasi sistem dalam bentuk pengujian *User Acceptance Testing (UAT)*, yang bertujuan untuk mengetahui tingkat penerimaan pengguna akhir terhadap sistem informasi yang telah dibangun. Selain itu, tanggapan pengguna terhadap fungsionalitas masing-masing fitur juga diperkuat

melalui pengujian *Black Box*, untuk memastikan bahwa setiap bagian dari sistem bekerja sesuai dengan harapan dan kebutuhan pengguna.

3. Dokumentasi

Dokumentasi dilakukan dengan mengumpulkan dokumen-dokumen pendukung yang digunakan dalam proses peminjaman alat di laboratorium.

Dokumen tersebut meliputi:

- a. Daftar inventaris peralatan laboratorium
- b. Standar Operasional Prosedur (SOP) peminjaman alat

Dokumen ini digunakan sebagai referensi dalam mendesain struktur database, form peminjaman, serta alur kerja sistem agar tetap mengikuti prosedur resmi yang berlaku di *lingkungan* laboratorium.

3.5. Perancangan Sistem

Perancangan sistem merupakan tahapan penting dalam pengembangan perangkat lunak yang bertujuan untuk menggambarkan secara detail struktur dan komponen sistem yang akan dibangun. Pada tahap ini, rancangan disusun berdasarkan hasil analisis kebutuhan agar sistem yang dikembangkan mampu menjalankan fungsinya dengan efektif dan efisien.

Perancangan sistem dalam penelitian ini mencakup beberapa aspek utama, yaitu: alur proses sistem, interaksi antara pengguna dan sistem, aktivitas yang terjadi dalam sistem, struktur basis data, serta rancangan antarmuka pengguna. Setiap bagian perancangan disajikan dalam bentuk visual diagram untuk mempermudah pemahaman serta meminimalkan kesalahan implementasi. Berikut adalah komponen-komponen utama dalam perancangan sistem ini:

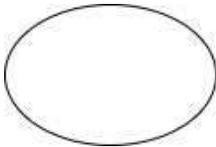
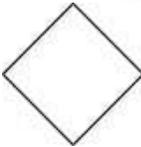
3.5.1. *Flowchart* Sistem

Flowchart merupakan representasi visual yang menggambarkan urutan alur kerja sistem informasi secara logis dan sistematis. Diagram ini digunakan untuk memetakan proses yang terjadi dalam sistem, mulai dari pengguna masuk (*login*) hingga menyelesaikan aktivitas seperti pengajuan peminjaman, verifikasi, pengembalian alat, dan pengelolaan data.

Penyusunan *flowchart* membantu pengembang dan pemangku kepentingan dalam memahami struktur dan logika sistem yang akan dibangun. Selain itu, *flowchart* juga berguna sebagai acuan dalam proses implementasi sistem agar tetap sesuai dengan perancangan fungsional.

Fungsi Simbol dalam Flowchart:

Table 3.5-1 Tabel Simbol Flowchart

Simbol	Nama Simbol	Fungsi
	Terminator	Menunjukkan awal (<i>start</i>) dan akhir (<i>end</i>) dari proses dalam sistem.
	Proses	Menyatakan suatu tindakan atau aktivitas yang dilakukan, seperti “ <i>Login</i> ” atau “Validasi Data”.
	Keputusan	Menunjukkan pengambilan keputusan atau kondisi bercabang (<i>decision</i>), seperti “Apakah data valid?”.
	<i>Input/Output</i>	Menyatakan proses masukan atau keluaran, misalnya “Masukkan data alat” atau “Tampilkan notifikasi”.
	Arah Alur	Menunjukkan urutan atau aliran proses dari satu simbol ke simbol berikutnya.

Pada sistem informasi manajemen peminjaman peralatan laboratorium ini, *flowchart* disusun berdasarkan tiga peran utama pengguna, yaitu:

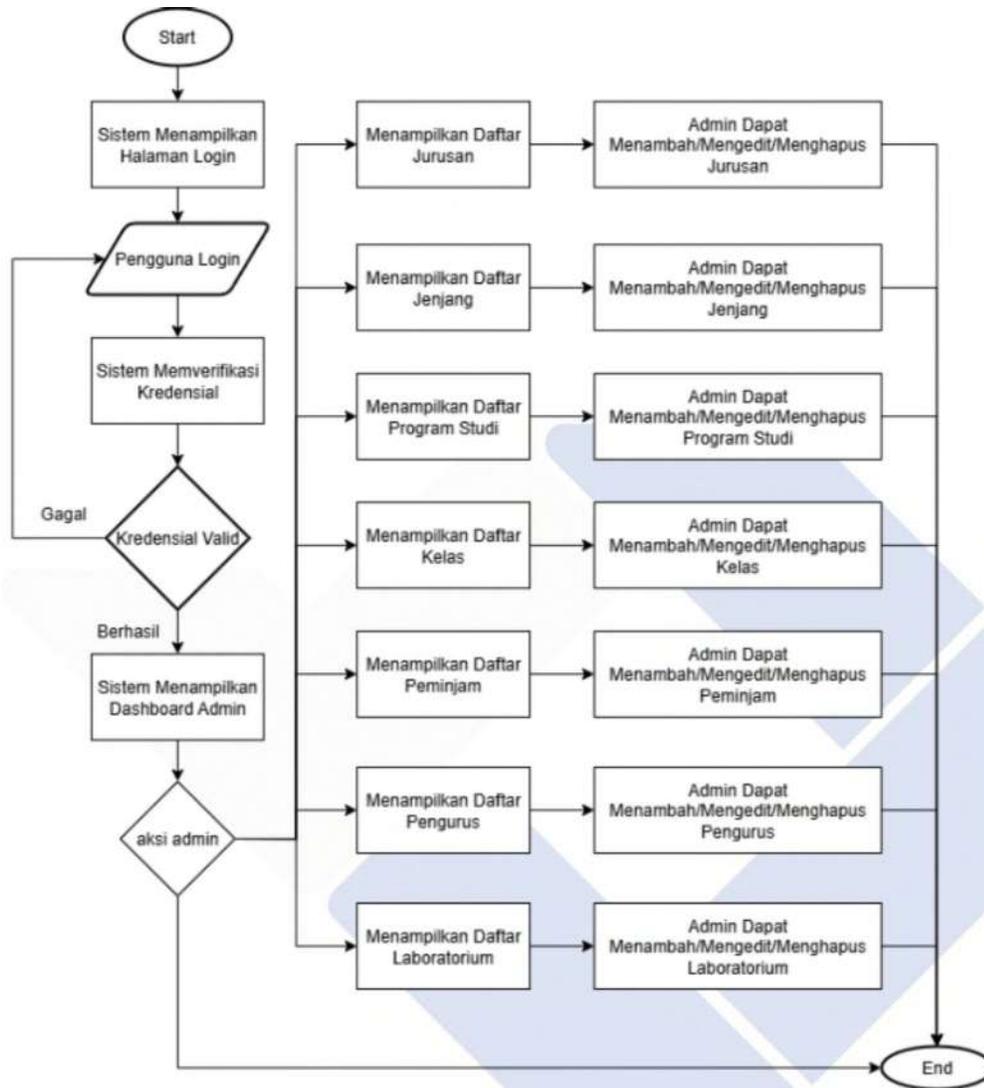
- a. Admin, yang bertugas mengelola data pengguna, jurusan, dan laboratorium.

- b. Pengurus laboratorium (PLP), yang bertanggung jawab terhadap pengelolaan alat, validasi peminjaman dan pengembalian, serta pembuatan laporan.
- c. Peminjam (dosen atau mahasiswa), yang mengajukan permintaan peminjaman dan melakukan pengembalian alat.

Secara umum, alur sistem mencakup langkah-langkah sebagai berikut:

1. Pengguna masuk ke dalam sistem (*login*).
2. Sistem memverifikasi kredensial dan menentukan peran pengguna.
3. Pengguna diarahkan ke halaman dasbor sesuai peran.
4. Peminjam dapat mengajukan peminjaman, yang akan diverifikasi oleh PLP.
5. Setelah disetujui, peminjaman dapat dilakukan.
6. Pengembalian alat juga melalui proses verifikasi.
7. Jika terjadi keterlambatan, sistem akan mengirimkan *notifikasi* melalui *email*.
8. Semua aktivitas tercatat secara otomatis dalam *logbook* sistem.

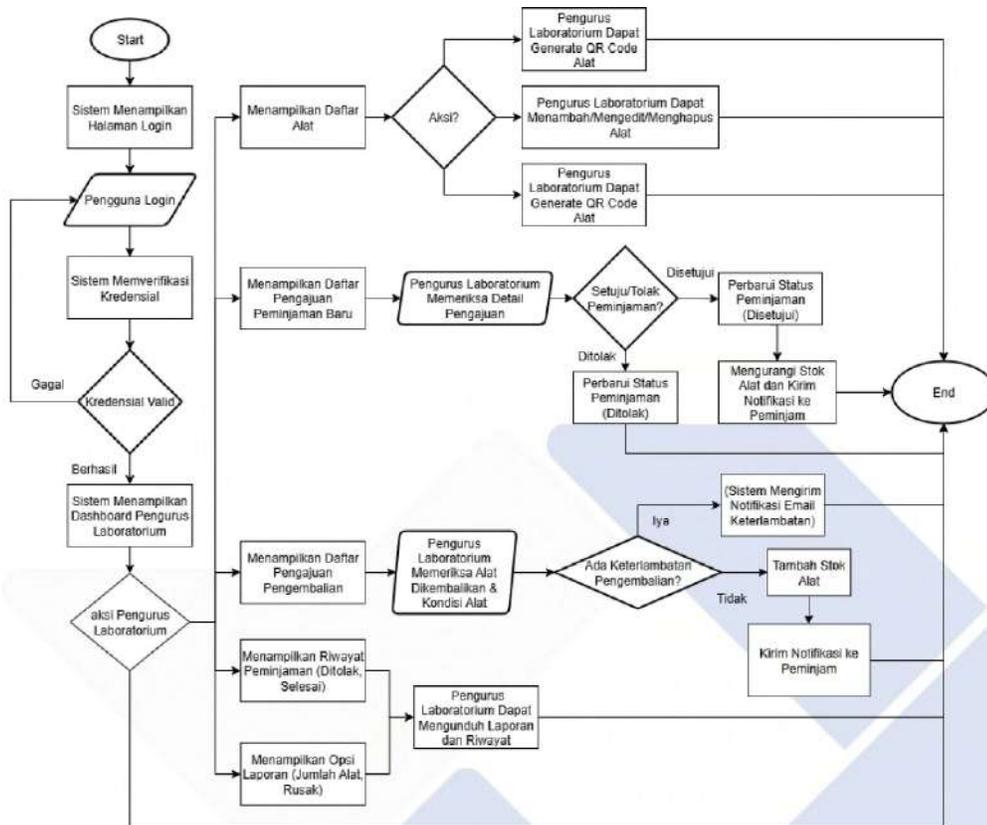
Sistem Admin:



Gambar 3.5-1 Flowchart Sistem Admin

Flowchart ini menggambarkan alur aktivitas admin setelah berhasil login ke dalam sistem. Admin memiliki akses penuh terhadap manajemen data master, seperti data jurusan, jenjang, program studi, kelas, peminjam, pengurus laboratorium, dan data laboratorium itu sendiri. Setiap proses mencakup tindakan melihat daftar data dan melakukan operasi *CRUD* (*Create, Read, Update, Delete*) untuk memastikan data selalu terbaru dan valid.

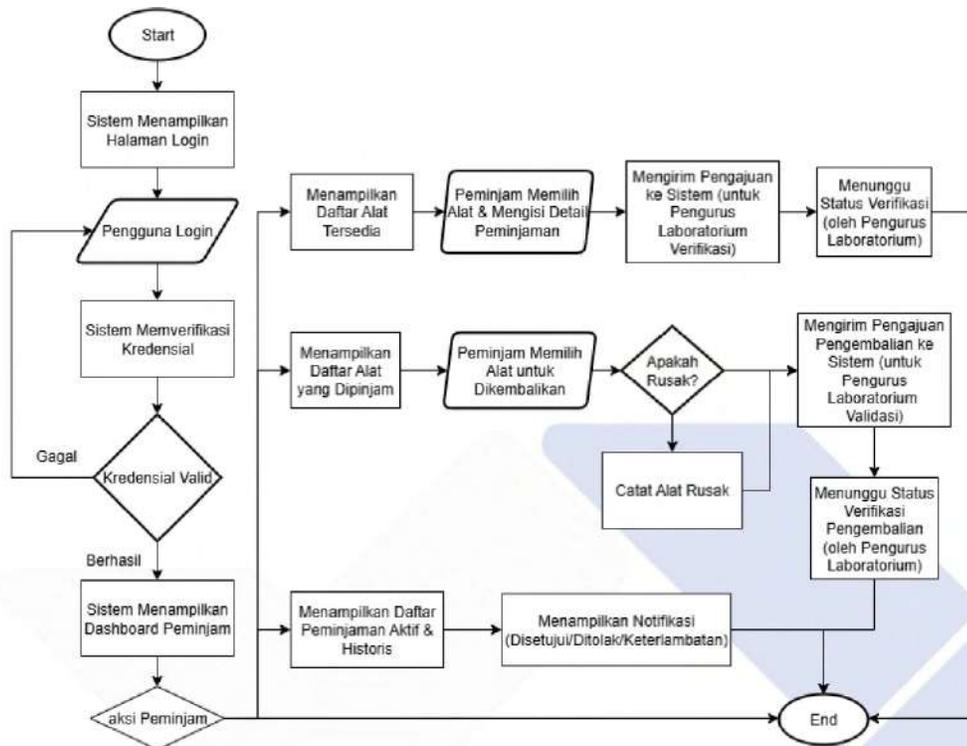
Sistem Pengurus Laboratorium:



Gambar 3.5-2 Flowchart Sistem Pengurus Laboratorium

Flowchart ini menunjukkan proses kerja pengurus laboratorium mulai dari login, melihat daftar pengajuan peminjaman dan pengembalian, serta melakukan validasi terhadap setiap pengajuan tersebut. Pengurus juga dapat memperbarui status alat, mengelola stok, menangani keterlambatan pengembalian, hingga mengunduh laporan riwayat peminjaman. Selain itu, pengurus juga berwenang menambah, mengedit, menghapus, serta menghasilkan QR Code alat.

Sistem Peminjam



Gambar 3.5-3 Flowchart Sistem Peminjam

Flowchart ini mendeskripsikan alur penggunaan sistem oleh peminjam (mahasiswa atau dosen). Setelah login, peminjam dapat melihat daftar alat yang tersedia, mengisi detail peminjaman, serta mengajukan pengembalian alat. Sistem kemudian akan menunggu proses validasi dari pengurus laboratorium. Jika alat yang dikembalikan rusak, peminjam juga dapat mencatat kondisi tersebut sebelum pengajuan diverifikasi. Notifikasi status peminjaman dan pengembalian akan ditampilkan sesuai hasil validasi.

3.5.2. Diagram Use Case

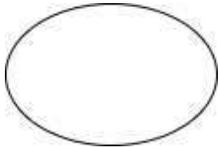
Diagram *use case* menggambarkan hubungan antara aktor (pengguna sistem) dan fungsi-fungsi utama sistem. Diagram ini menunjukkan interaksi yang terjadi antara pengguna seperti admin, pengurus laboratorium (*PLP*), dan peminjam dengan sistem informasi peminjaman peralatan laboratorium.

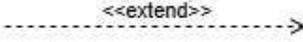
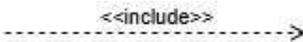
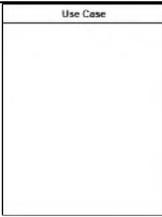
Sistem ini memiliki tiga aktor utama, yaitu:

- a. Admin yang bertanggung jawab terhadap pengelolaan data dasar institusi seperti jurusan, jenjang, program studi, kelas, pengguna, serta laboratorium dan pengurusnya.
- b. Pengurus Laboratorium (*PLP*), yang bertugas dalam pengelolaan alat, validasi proses peminjaman dan pengembalian, serta pengelolaan laporan dan peralatan rusak.
- c. Peminjam (mahasiswa atau dosen), yang menggunakan sistem untuk mengajukan peminjaman, melakukan pengembalian, dan memantau riwayat transaksi.

Penjelasan Simbol dalam *Use Case Diagram*:

Table 3.5-2 Tabel Simbol Use Case

Simbol	Nama Simbol	Fungsi
 Actor	Aktor	Mewakili pengguna sistem (manusia atau sistem eksternal) yang berinteraksi dengan sistem.
	Use Case	Mewakili fungsionalitas atau layanan utama yang disediakan oleh sistem.

	Asosiasi	Menghubungkan aktor dengan <i>use case</i> yang mereka jalankan; menunjukkan alur interaksi.
	Extend	Menunjukkan fungsi opsional yang diperluas dari use case utama.
	Include	Menunjukkan fungsi yang selalu disertakan dalam use case lainnya.
	Sistem Boundary	Menandai batas sistem yang dibangun; semua <i>use case</i> berada dalam batas ini.

Setiap aktor berinteraksi dengan *use case* yang sesuai dengan perannya.

Deskripsi Fungsi Berdasarkan Aktor:

1. Aktor Admin

Admin merupakan pengguna sistem yang memiliki hak akses penuh terhadap pengelolaan data institusi. Fungsi yang dapat dilakukan oleh aktor ini antara lain:

- a) Melakukan proses *login* ke dalam sistem.
- b) Mengelola data jurusan, jenjang pendidikan, program studi, dan kelas.
- c) Mengelola data pengguna yang berperan sebagai peminjam.
- d) Mengelola data laboratorium secara menyeluruh.
- e) Mengelola data pengurus laboratorium.

2. Aktor Peminjam

Peminjam adalah pengguna sistem yang terdiri atas mahasiswa atau dosen, yang dapat melakukan transaksi peminjaman alat. Fungsi yang tersedia bagi aktor ini meliputi:

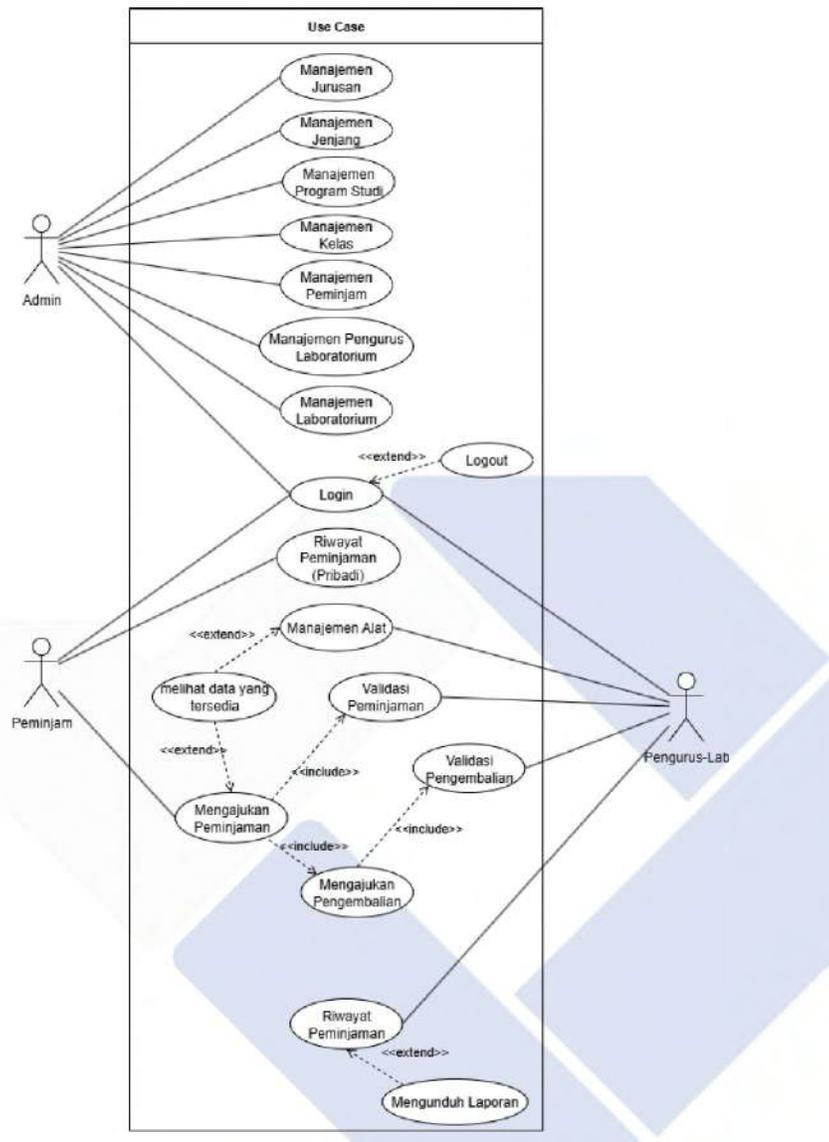
- a) Melakukan *login* ke dalam sistem.
- b) Mengajukan permohonan peminjaman peralatan laboratorium.
- c) Mengajukan permohonan pengembalian alat setelah penggunaan.

d) Melihat dan memantau riwayat peminjaman yang telah dilakukan.

3. Aktor Pengurus Laboratorium (PLP):

Pengurus laboratorium adalah pihak yang bertanggung jawab terhadap operasional harian pengelolaan alat di laboratorium. Tugas dan fungsi yang dijalankan dalam sistem, antara lain:

- a) Melakukan *login* ke dalam sistem.
- b) Mengelola data alat laboratorium, termasuk penambahan, pengubahan, dan penghapusan.
- c) Melakukan validasi terhadap pengajuan peminjaman yang masuk.
- d) Melakukan validasi atas pengembalian alat oleh peminjam.
- e) Mengunduh riwayat seluruh aktivitas peminjaman dan pengembalian alat



Gambar 3.5-4 Diagram Use Case

Diagram *use case* ini memberikan gambaran umum mengenai cakupan interaksi antar pengguna dan sistem, serta menunjukkan bagaimana setiap peran memiliki akses eksklusif terhadap fungsi-fungsi tertentu dalam sistem.

3.5.3. Activity Diagram

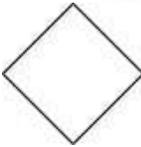
Activity diagram atau diagram aktivitas digunakan untuk menggambarkan alur aktivitas dalam sistem secara lebih rinci. Diagram ini menyajikan urutan langkah-langkah logis yang dilakukan baik oleh pengguna (*user*) maupun sistem, dari awal

hingga akhir dalam suatu proses tertentu. Tujuan dari diagram ini adalah untuk memberikan visualisasi yang jelas terhadap logika pemrosesan dan alur interaksi antar komponen sistem dalam skenario tertentu.

Pada sistem informasi manajemen peminjaman peralatan laboratorium ini, *activity diagram* dibagi berdasarkan jenis pengguna (peran) dan fungsionalitas utama yang dijalankan.

Penjelasan pada simbol dalam *Activity Diagram*:

Table 3.5-3 Tabel Simbol Activity Diagram

Simbol	Nama Simbol	Fungsi
	Initial Node	Menunjukkan titik awal dari aktivitas yang dimodelkan.
	Activity/Action	Menyatakan aktivitas atau tindakan yang dilakukan dalam alur proses.
	Decision Node	Menunjukkan titik pengambilan keputusan; alur dapat bercabang berdasarkan kondisi.
	Control Flow	Menunjukkan urutan alur antaraktivitas dan keputusan dalam diagram.
	Final Node	Menunjukkan akhir dari seluruh proses dalam aktivitas tersebut.

1. *Activity Diagram* Admin:

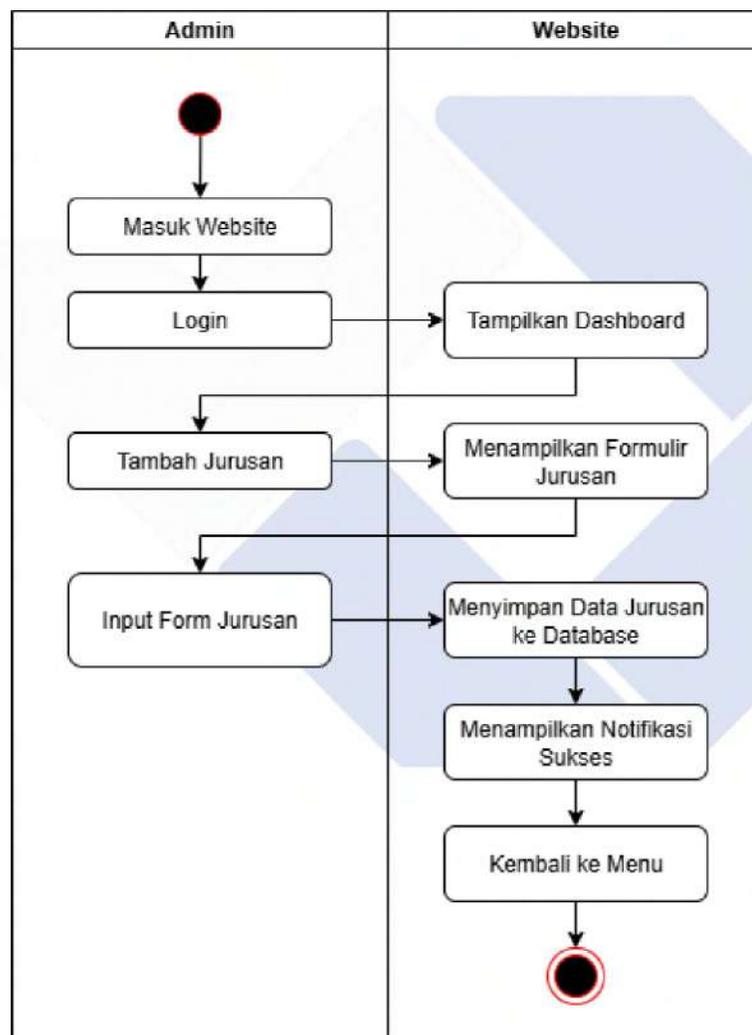
Admin memiliki peran untuk mengelola data master sistem, seperti jurusan, jenjang, program studi, kelas, peminjam, laboratorium, dan pengurus laboratorium.

a) Manajemen Jurusan:

Diagram aktivitas ini menggambarkan alur proses yang dilakukan oleh Admin dalam menambahkan data jurusan ke dalam sistem. Proses dimulai ketika Admin

mengakses sistem melalui halaman *login*. Setelah berhasil masuk, sistem akan menampilkan halaman dasbor.

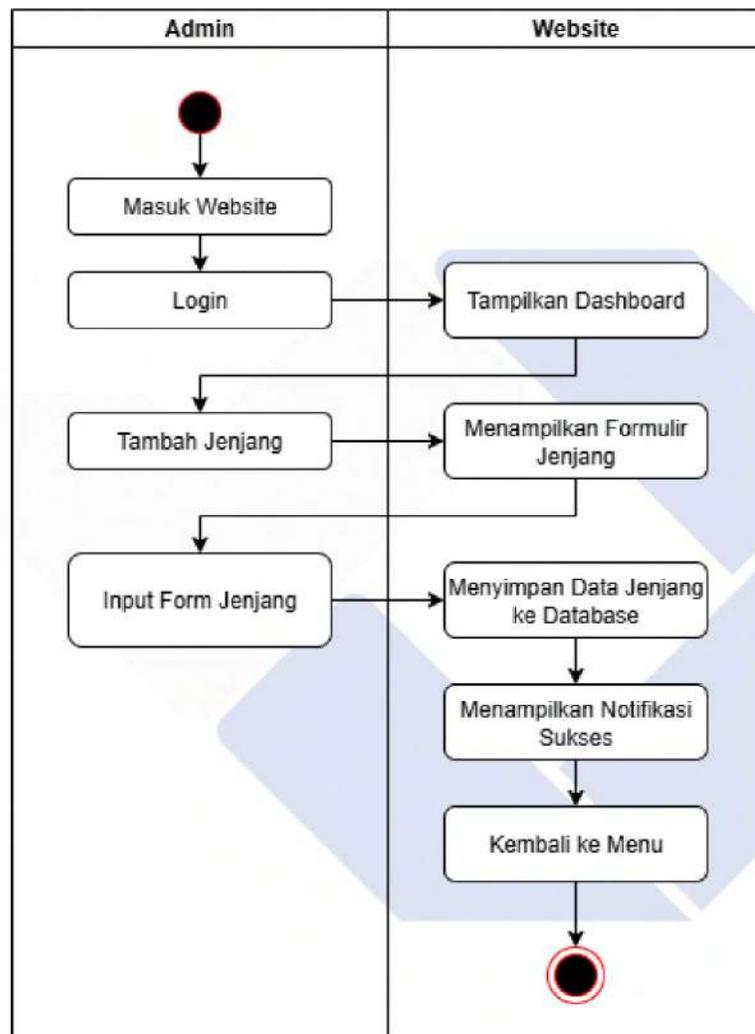
Admin kemudian memilih menu Tambah Jurusan, yang akan memicu sistem untuk menampilkan formulir *input* jurusan. Setelah Admin mengisi formulir tersebut, data dikirim dan disimpan oleh sistem ke dalam basis data. Jika proses penyimpanan berhasil, sistem akan menampilkan notifikasi sukses, lalu mengarahkan kembali ke menu utama.



Gambar 3.5-5 Activity Diagram Admin (Jurusan)

Alur ini memastikan bahwa proses manajemen jurusan dilakukan secara sistematis dan terotomatisasi, memudahkan Admin dalam memperbarui data akademik secara efisien.

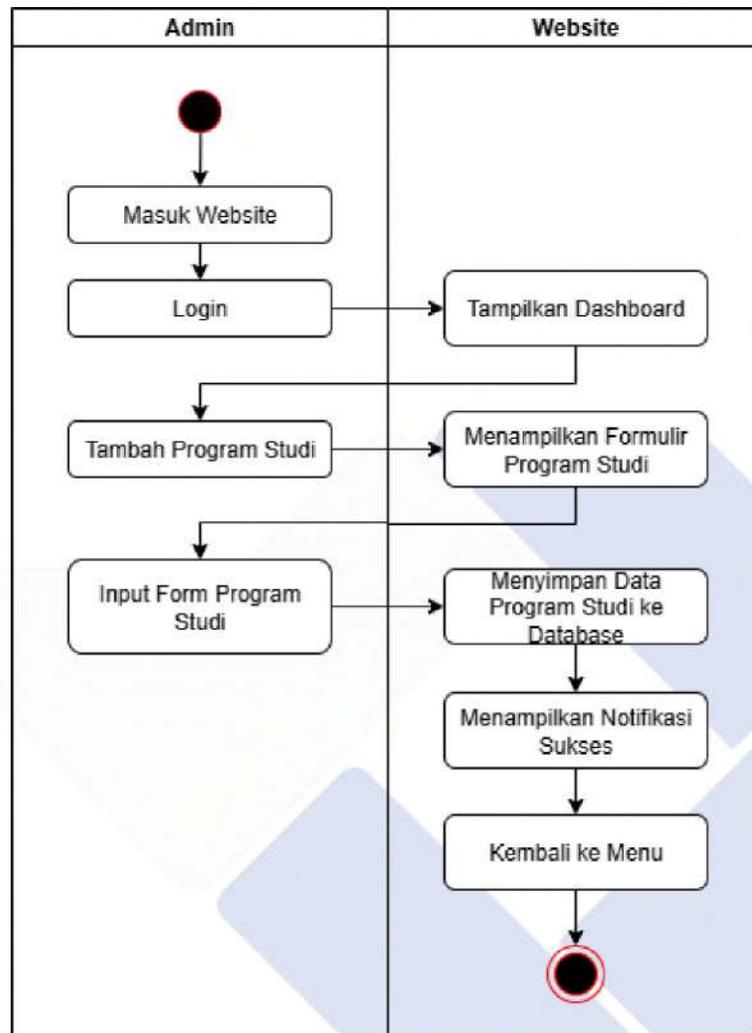
b) Manajemen Jenjang:



Gambar 3.5-6 Activity Diagram Admin (Jenjang)

Diagram ini menggambarkan alur aktivitas Admin dalam menambahkan data jenjang pendidikan ke dalam sistem, dimulai dari *login* hingga penyimpanan data jenjang yang berhasil dilakukan oleh sistem.

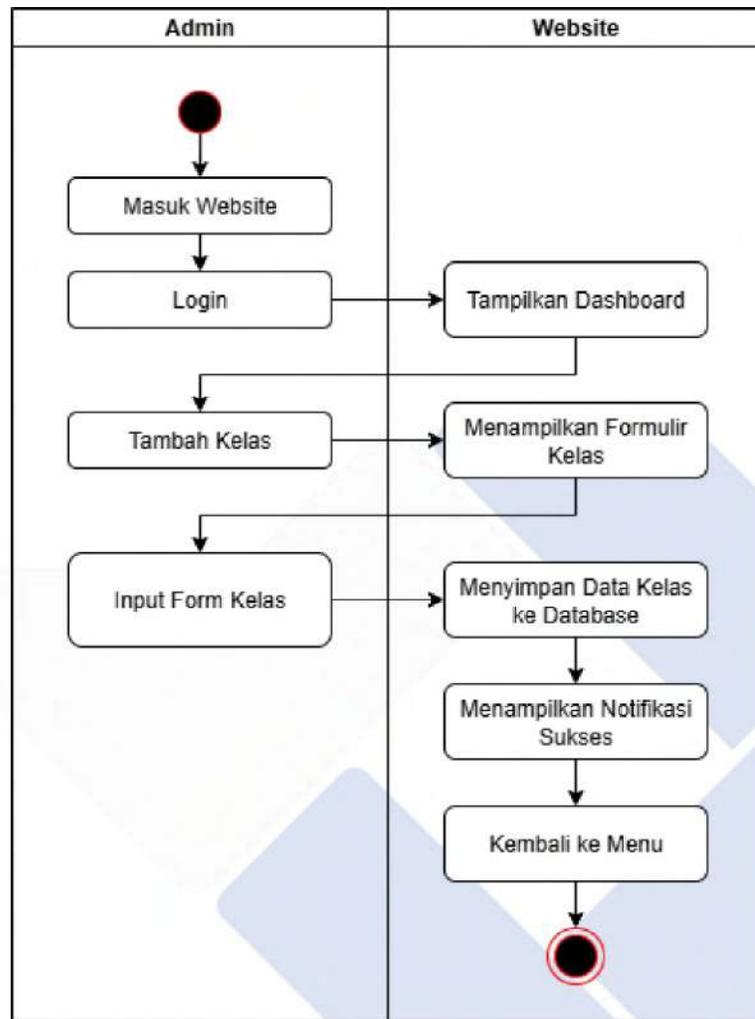
c) Manajemen Program Studi:



Gambar 3.5-7 Activity Diagram Admin (Program Studi)

Diagram ini menjelaskan alur aktivitas Admin dalam mengelola data program studi. Aktivitas dimulai dari *login*, pemilihan menu, pengisian data, hingga penyimpanan data program studi oleh sistem.

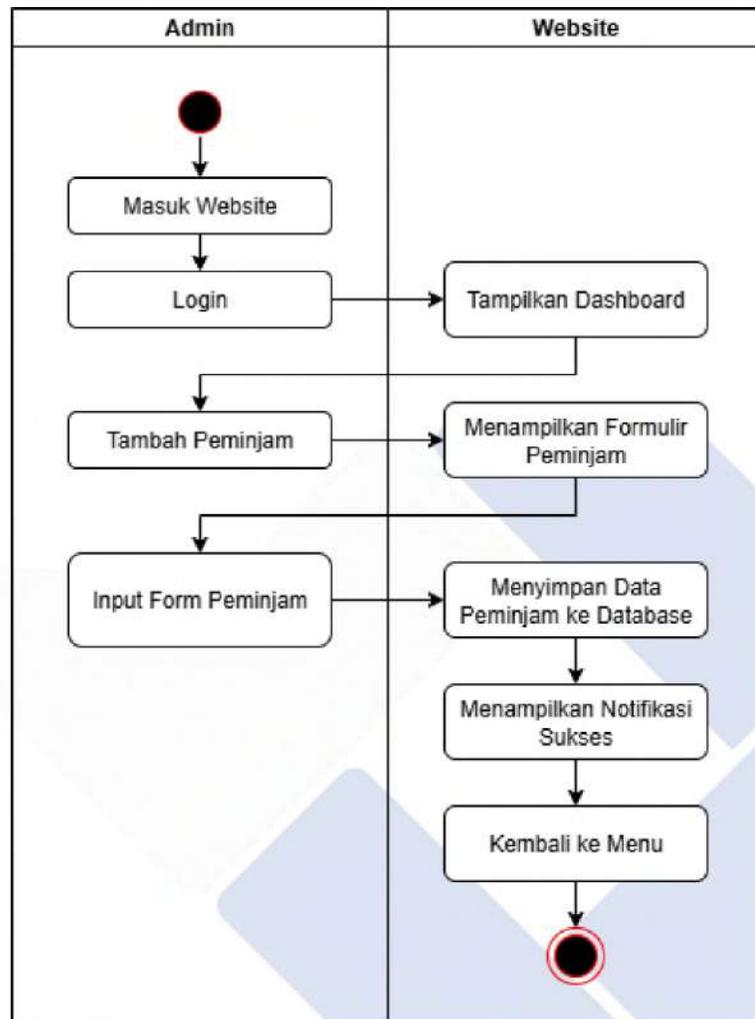
d) Manajemen Kelas:



Gambar 3.5-8 Activity Diagram Admin (Kelas)

Diagram ini menunjukkan tahapan Admin dalam menambahkan data kelas ke dalam sistem, mulai dari akses *login* hingga proses penyimpanan data kelas.

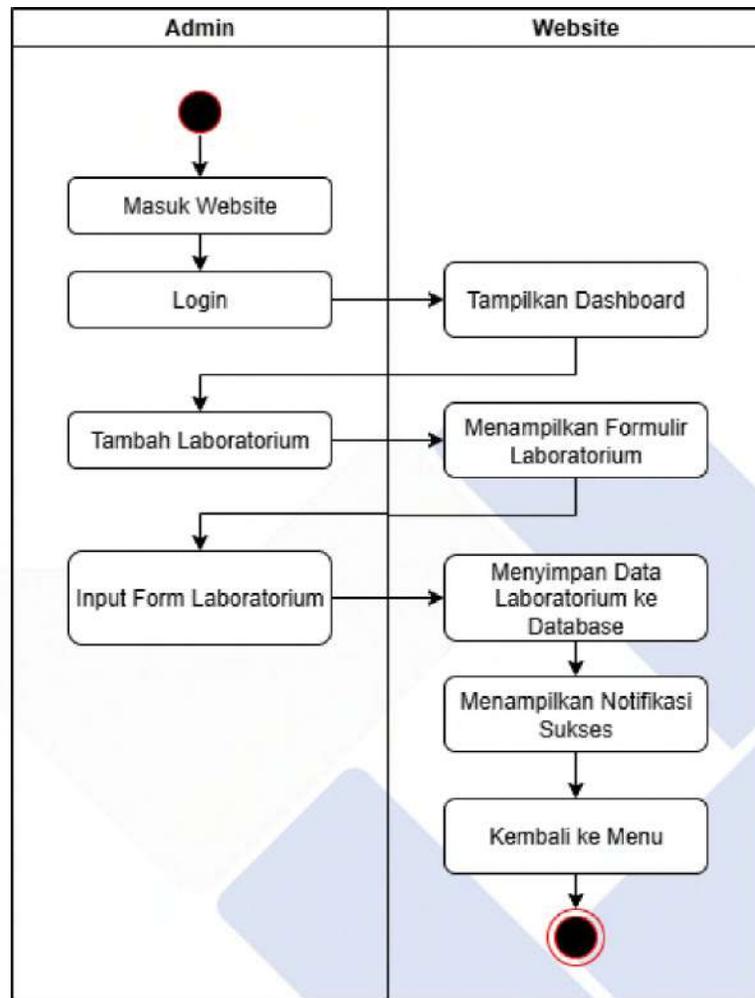
e) Manajemen Peminjam:



Gambar 3.5-9 Activity Diagram Admin (Peminjam)

Diagram aktivitas ini menggambarkan proses yang dilakukan Admin dalam menambahkan atau mengelola data peminjam, seperti mahasiswa atau dosen, ke dalam sistem informasi.

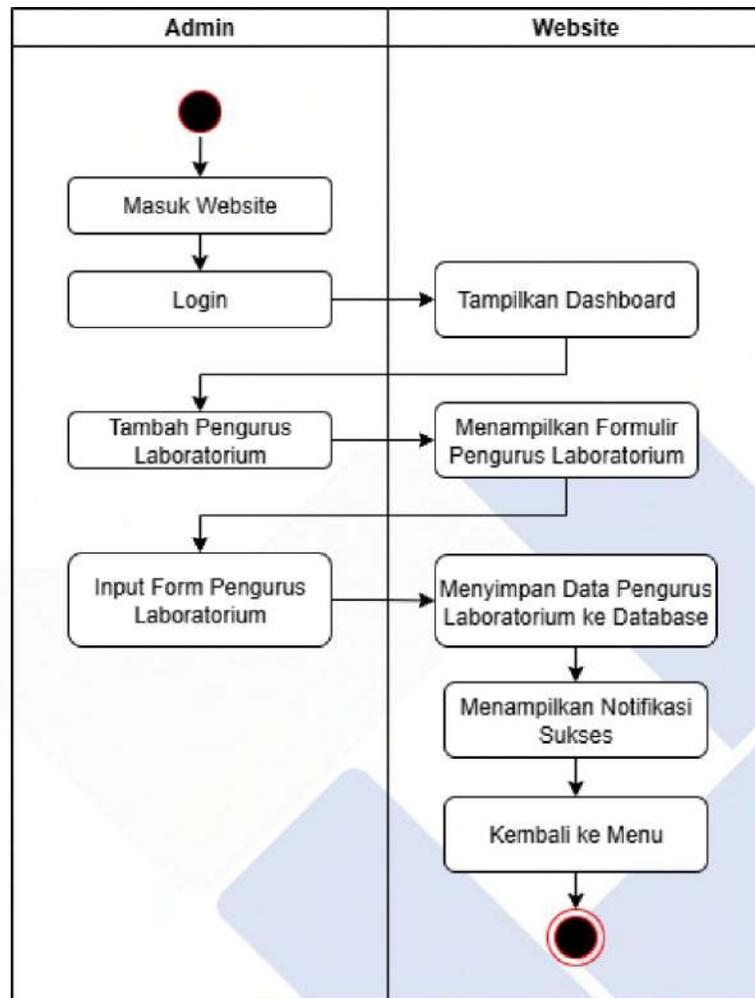
f) Manajemen Laboratorium:



Gambar 3.5-10 Activity Diagram Admin (Laboratorium)

Diagram ini menjelaskan alur aktivitas dalam menambahkan atau memperbarui data laboratorium oleh Admin, yang mencakup akses sistem, *input* data, dan penyimpanan informasi laboratorium.

g) Manajemen Pengurus Laboratorium:



Gambar 3.5-11 Activity Diagram Admin (Pengurus Laboratorium)

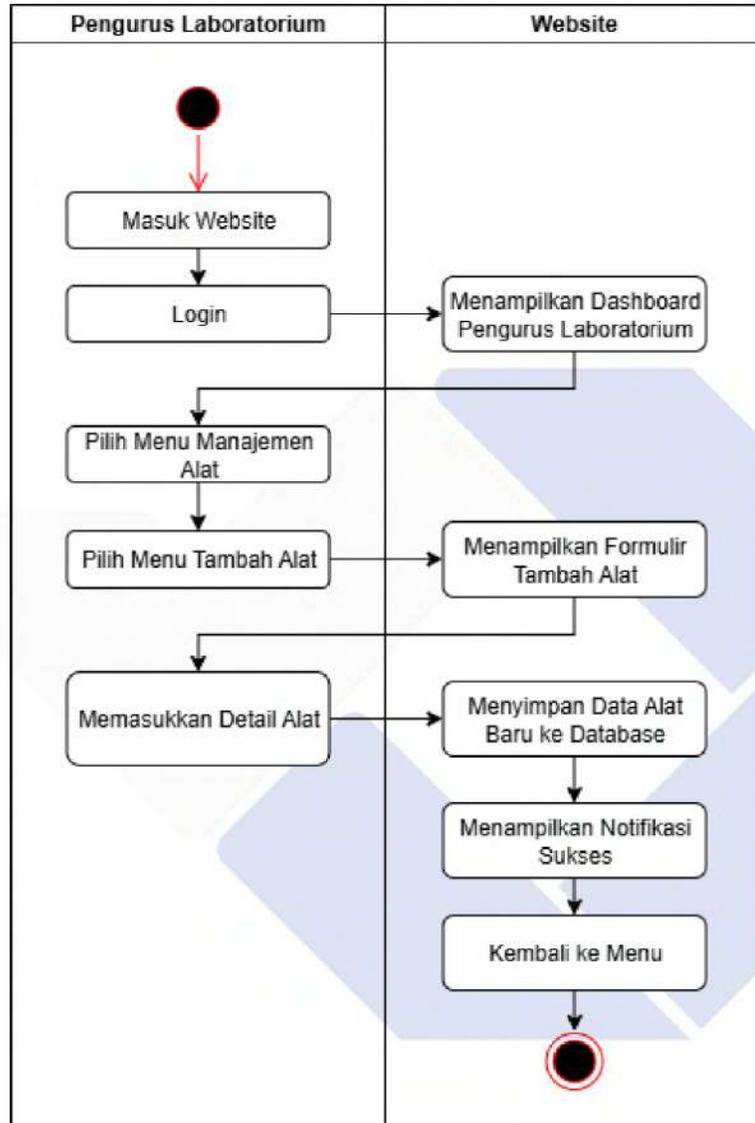
Diagram aktivitas ini menggambarkan langkah-langkah yang dilakukan oleh Admin dalam menambahkan data pengurus laboratorium, mulai dari *login* hingga data berhasil disimpan oleh sistem.

2. Activity Diagram Pengurus Laboratorium:

Pengurus laboratorium bertanggung jawab terhadap operasional alat, mulai dari penambahan alat hingga laporan.

Setiap fungsi digambarkan dalam diagram aktivitas berikut:

a) Tambah Alat:

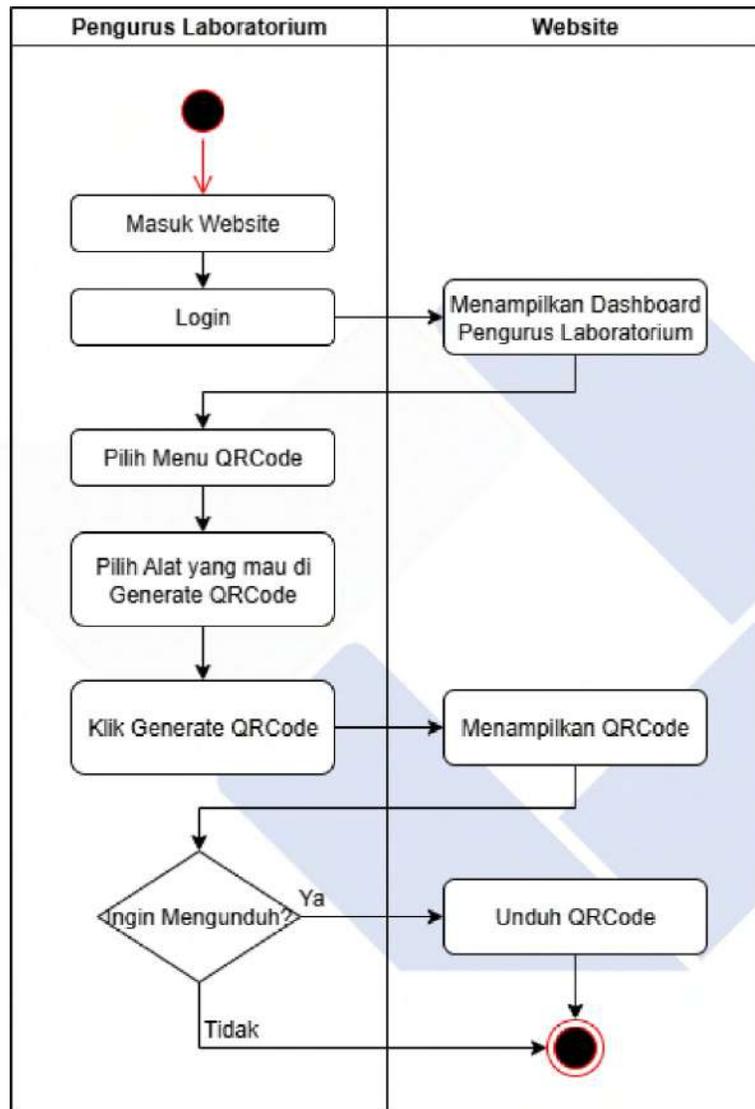


Gambar 3.5-12 Activity Diagram Pengurus Laboratorium (Tambah Alat)

Diagram ini menggambarkan proses yang dilakukan oleh Pengurus Laboratorium dalam menambahkan data alat baru ke dalam sistem. Proses dimulai dari *login* ke sistem, dilanjutkan dengan memilih menu Manajemen Alat, kemudian menu Tambah Alat. Sistem akan menampilkan formulir *input* alat, yang kemudian diisi

oleh pengguna. Data alat disimpan ke dalam basis data, dan jika berhasil, sistem menampilkan notifikasi sukses serta mengarahkan kembali ke menu utama.

b) Generate QRCode:

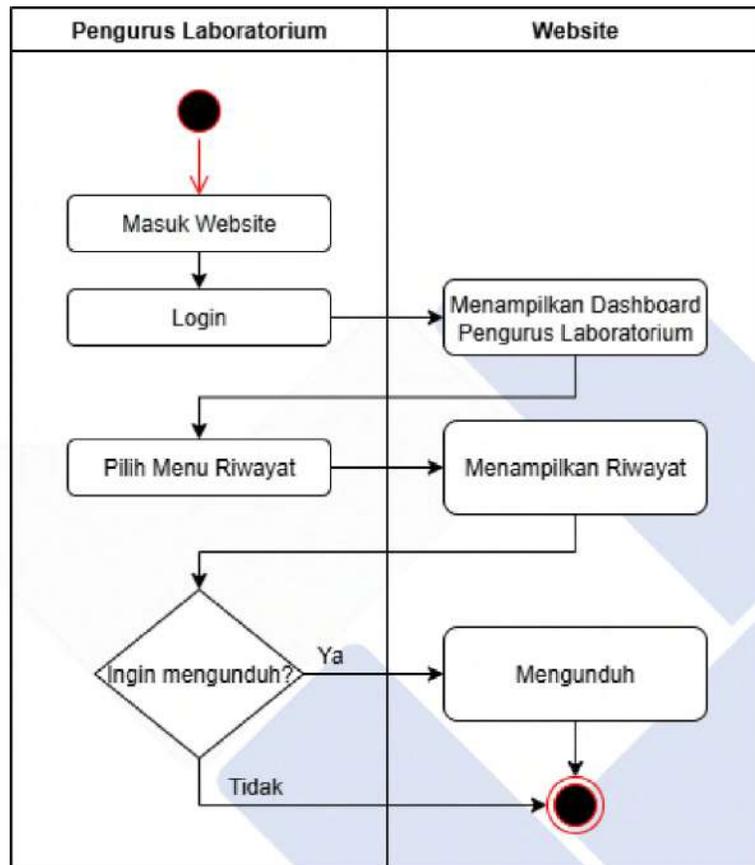


Gambar 3.5-13 Activity Diagram Pengurus Laboratorium (Generate QRCode)

Diagram aktivitas ini menjelaskan alur ketika Pengurus Laboratorium ingin menghasilkan QRCode untuk suatu alat. Setelah berhasil *login*, pengguna memilih menu QRCode dan memilih alat yang ingin dibuatkan QRCode. Sistem akan

menampilkan QRCode yang dihasilkan, lalu pengguna diberi opsi untuk mengunduh QRCode tersebut atau tidak.

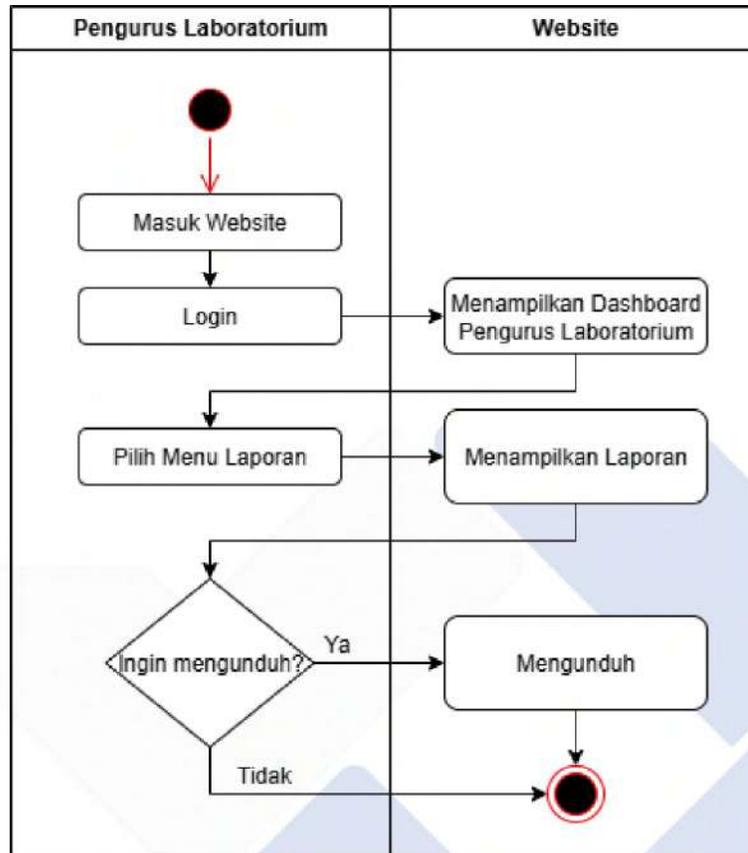
c) Riwayat:



Gambar 3.5-14 Activity Diagram Pengurus Laboratorium (Riwayat)

Diagram ini menggambarkan proses akses riwayat aktivitas peminjaman dan pengembalian alat oleh Pengurus Laboratorium. Setelah login, pengguna memilih menu *Riwayat*, lalu sistem menampilkan data riwayat. Pengguna dapat memilih untuk mengunduh riwayat tersebut atau tidak.

d) Laporan:



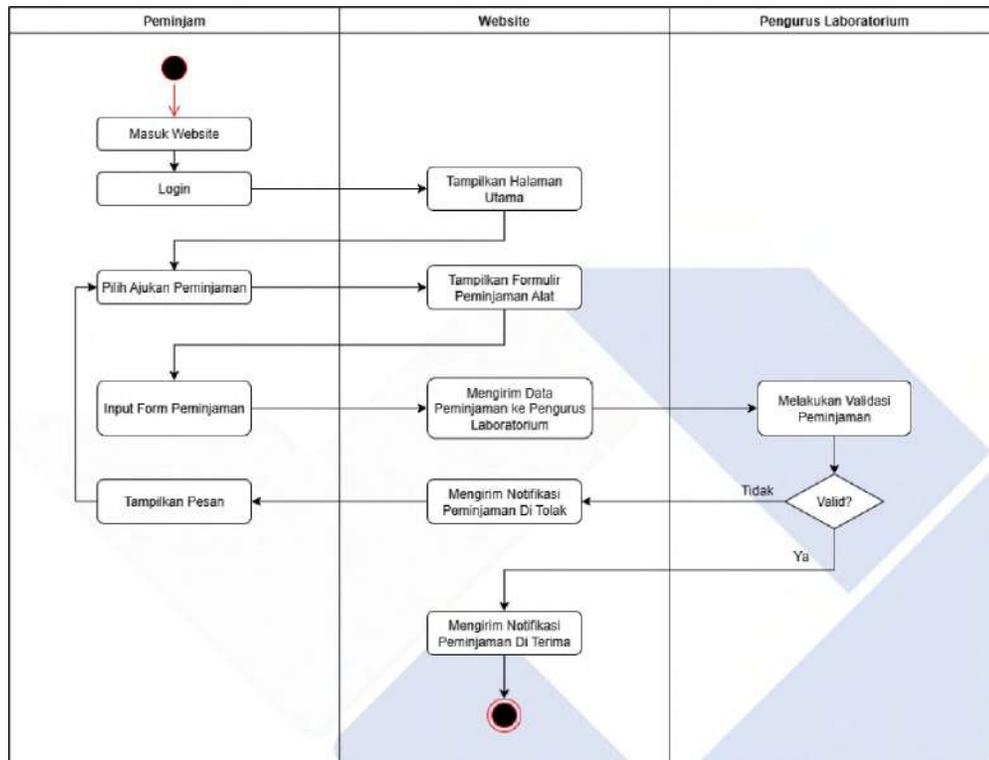
Gambar 3.5-15 Activity Diagram Pengurus Laboratorium (Laporan)

Diagram ini menunjukkan proses akses dan pengelolaan laporan oleh Pengurus Laboratorium. Setelah login, pengguna mengakses menu Laporan, kemudian sistem menampilkan data laporan yang tersedia. Pengguna dapat memilih untuk mengunduh laporan tersebut atau tidak.

3. Activity Diagram Peminjam:

Peminjam (mahasiswa maupun dosen) memiliki akses terhadap sistem untuk melakukan proses peminjaman dan pengembalian alat laboratorium.

a) Peminjaman

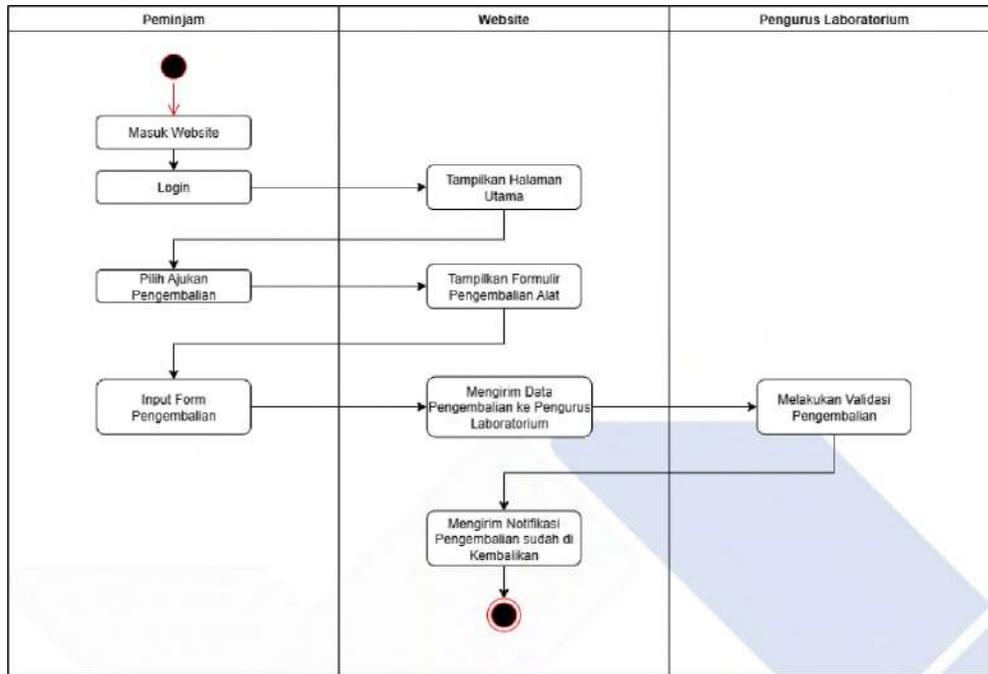


Gambar 3.5-16 Activity Diagram Peminjam (Peminjaman)

Diagram aktivitas ini menggambarkan proses peminjaman alat oleh pengguna yang berperan sebagai peminjam. Proses dimulai dari pengguna masuk ke dalam sistem (*login*), kemudian memilih menu *Ajukan Peminjaman*. Sistem akan menampilkan formulir peminjaman yang diisi oleh peminjam, lalu data dikirim ke pengurus laboratorium untuk divalidasi.

Pengurus laboratorium melakukan proses validasi. Jika pengajuan tidak valid, sistem mengirimkan notifikasi penolakan kepada peminjam. Sebaliknya, jika pengajuan valid, sistem akan mengirimkan notifikasi bahwa peminjaman diterima.

b) Pengembalian



Gambar 3.5-17 Activity Diagram Peminjam

Diagram aktivitas ini menggambarkan proses pengembalian alat laboratorium oleh pengguna yang berperan sebagai peminjam (mahasiswa maupun dosen). Proses dimulai ketika peminjam masuk ke dalam sistem (*login*), kemudian memilih menu Ajukan Pengembalian. Setelah itu, sistem menampilkan formulir pengembalian alat yang harus diisi oleh peminjam.

Setelah formulir dikirim, sistem akan meneruskan data pengembalian kepada Pengurus Laboratorium untuk dilakukan proses validasi pengembalian. Jika alat dinyatakan telah dikembalikan dengan benar, sistem akan mengirimkan notifikasi bahwa pengembalian telah berhasil kepada peminjam. Proses ini memastikan bahwa alur pengembalian tercatat secara terstruktur dan diverifikasi langsung oleh pihak yang berwenang.

3.5.4. Class Diagram

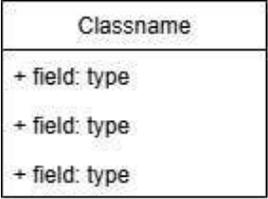
Class diagram merupakan salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk memodelkan struktur statis sistem berbasis objek. Diagram ini menampilkan kelas-kelas utama dalam sistem, atribut, metode, serta relasi antar kelas. Pada sistem informasi manajemen peminjaman peralatan laboratorium ini, *class diagram* menggambarkan hubungan antara pengguna, data alat, proses peminjaman, serta entitas lainnya yang terlibat dalam sistem.

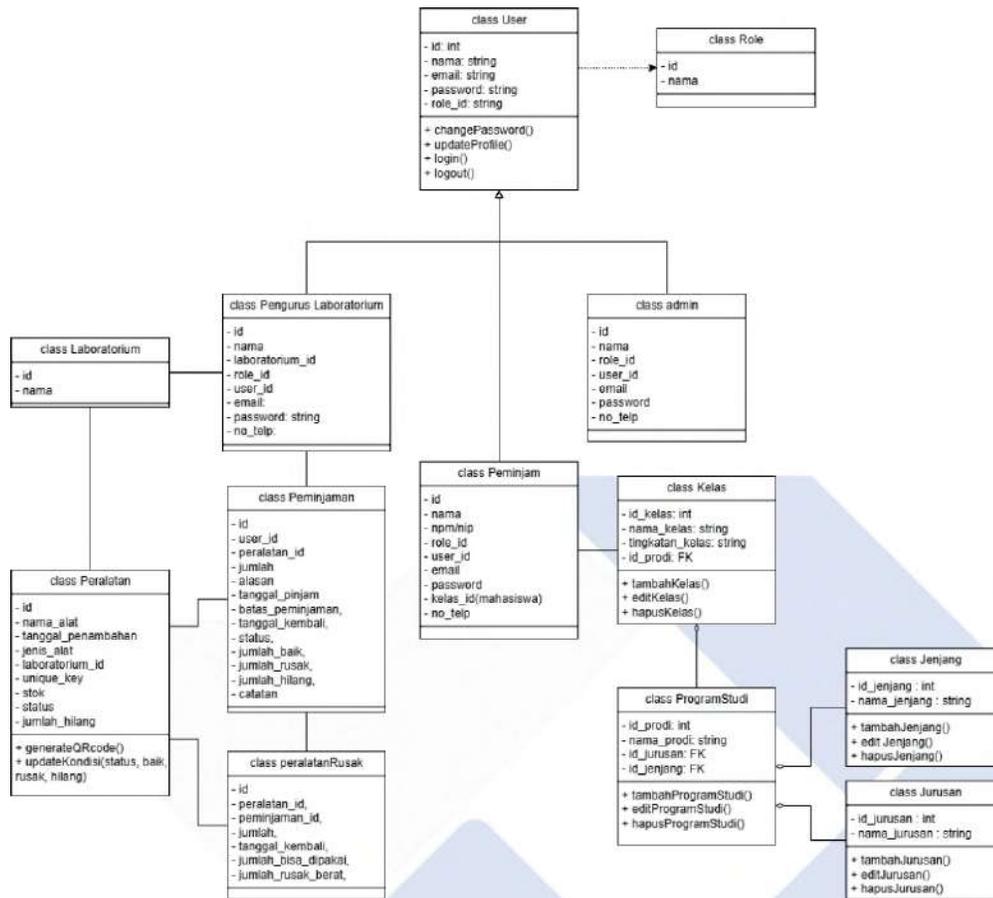
Penjelasan Komponen Utama:

- a. Kelas User: Menyimpan data umum pengguna, baik admin, peminjam, maupun pengurus laboratorium. Memiliki fungsi seperti *login()*, *logout()*, *editProfil()*, dan *changePassword()*.
- b. Kelas Role: Menentukan peran pengguna dalam sistem, yaitu admin, peminjam, dan pengurus laboratorium.
- c. Kelas Admin: Memiliki relasi dengan User dan bertanggung jawab atas pengelolaan data institusi seperti jurusan, jenjang, kelas, serta pengelolaan akun pengguna lainnya.
- d. Kelas Pengurus Laboratorium: Berfungsi dalam pengelolaan alat, validasi peminjaman dan pengembalian, serta pelaporan kerusakan alat.
- e. Kelas Peminjam: Mewakili mahasiswa atau dosen yang dapat mengajukan peminjaman serta melihat riwayat peminjaman.
- f. Kelas Peminjaman: Menyimpan informasi transaksi peminjaman alat, termasuk jumlah alat, tanggal pinjam dan kembali, serta status alat (baik, rusak, atau hilang).
- g. Kelas Peralatan: Menyimpan data inventaris alat, termasuk nama, jenis, lokasi laboratorium, dan status alat. Dilengkapi metode *generateQRCode()* dan *updateKondisi()*.
- h. Kelas PeralatanRusak: Menyimpan catatan alat yang rusak atau tidak bisa dipakai setelah proses peminjaman.
- i. Kelas Laboratorium: Menyimpan informasi nama dan identitas laboratorium. Kelas Program Studi, Jurusan, Jenjang, dan Kelas: Digunakan untuk mengelompokkan peminjam berdasarkan struktur akademik.

Penjelasan pada simbol dalam Class Diagram:

Table 3.5-4 Tabel Simbol Class Diagram

Simbol	Jenis	Penjelasan
	<i>Association</i>	Hubungan struktural antara dua kelas; bisa <i>one-to-one</i> , <i>one-to-many</i> , dll.
	<i>Dependency</i>	Menunjukkan bahwa satu kelas bergantung secara temporer pada kelas lain.
	<i>Aggregation</i>	Menunjukkan "memiliki", tetapi bagian dapat berdiri sendiri (whole-part).
	<i>Inheritance</i>	Menunjukkan pewarisan dari superclass ke subclass.
	<i>Class</i>	Representasi dari entitas dengan atribut dan metode di dalam sistem.



Gambar 3.5-18 Class Diagram

3.5.5. Wireframe Antarmuka Pengguna

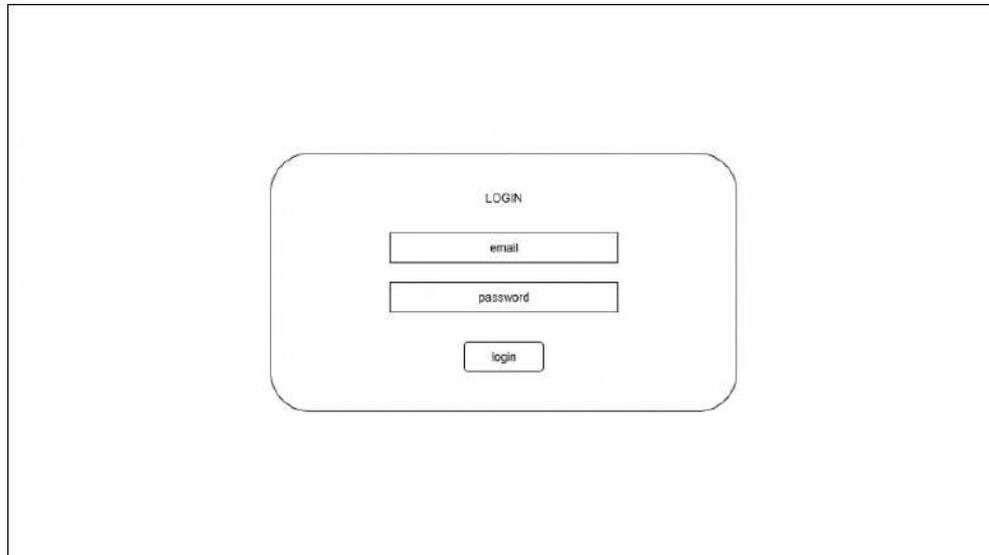
Low-fidelity wireframe antarmuka pengguna merupakan rancangan visual awal dari sistem yang merepresentasikan tata letak elemen-elemen antarmuka secara sederhana, tanpa detail desain grafis, dan bersifat statis. Tujuan utama dari *wireframe* ini adalah untuk menggambarkan struktur navigasi, alur interaksi pengguna, serta mendefinisikan komponen-komponen utama dalam antarmuka sistem.

Perancangan ini berfokus pada penyusunan kerangka logis dari tampilan sistem, sehingga dapat digunakan sebagai dasar komunikasi antara pengembang dan pengguna sebelum memasuki tahap desain visual akhir. Selain itu, *wireframe* ini

juga berfungsi untuk memastikan bahwa rancangan memenuhi prinsip *usability*, keterbacaan, dan keselarasan fungsi yang baik.

Berikut adalah beberapa tampilan antarmuka pengguna dalam bentuk *low fidelity wireframe*:

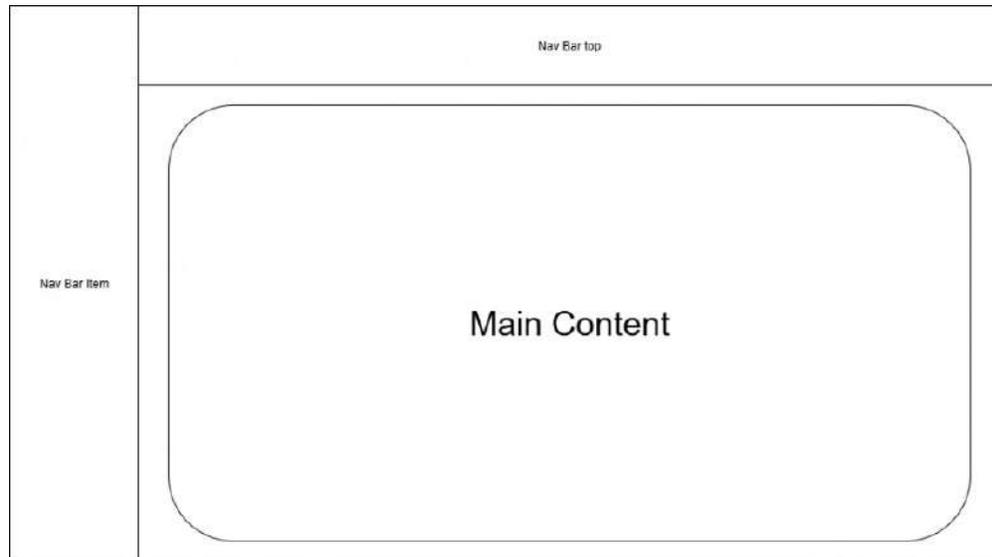
1. Halaman *Login*



Gambar 3.5-19 Login UI

Desain ini menampilkan form *login* sederhana dengan dua *input* field yaitu *email* dan *password*, serta satu tombol aksi bertuliskan “*Login*”. Letaknya berada di tengah layar, dikelilingi oleh kontainer dengan sudut membulat untuk memberikan kesan modern dan ramah pengguna.

2. Halaman Dashboard Utama



Gambar 3.5-20 Tampilan Layout Utama dengan Area Konten Dinamis

Tampilan ini terdiri dari:

- Nav Bar Top: Bagian navigasi horizontal di bagian atas.
- Nav Bar Side: Navigasi vertikal di sebelah kiri layar berisi menu.
- Main Content: Area pusat yang digunakan sebagai wadah dinamis untuk menampilkan berbagai konten tergantung navigasi yang dipilih pengguna.

Dengan struktur ini, pengguna akan tetap berada di dalam satu kerangka layout saat berpindah-pindah konten, meningkatkan konsistensi dan pengalaman pengguna secara keseluruhan.

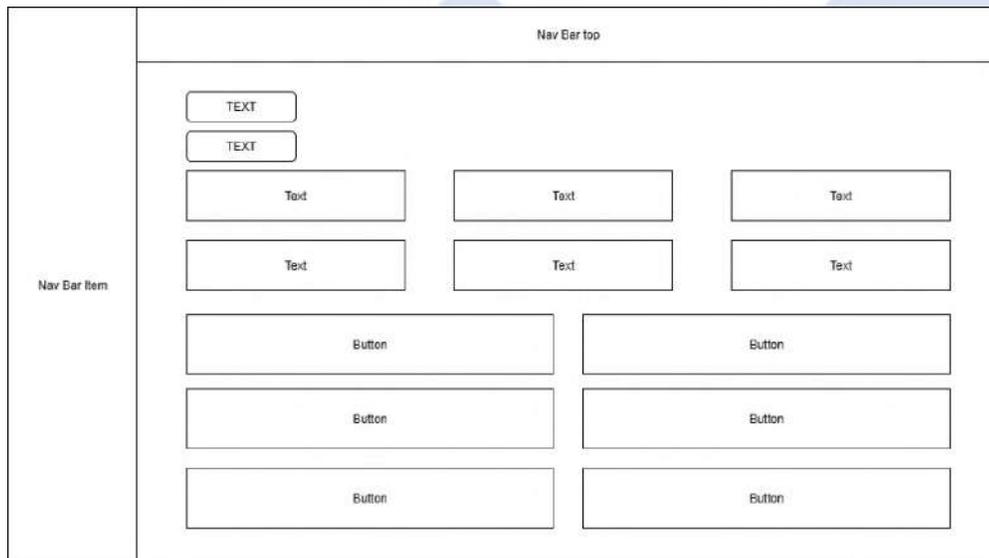
3. Halaman Dashboard Admin



Gambar 3.5-21 Tampilan Dashboard Admin

Tampilan ini merupakan halaman utama yang pertama kali diakses setelah pengguna dengan peran *admin* berhasil masuk ke dalam sistem.

4. Halaman Dashboard Pengurus Laboratorium

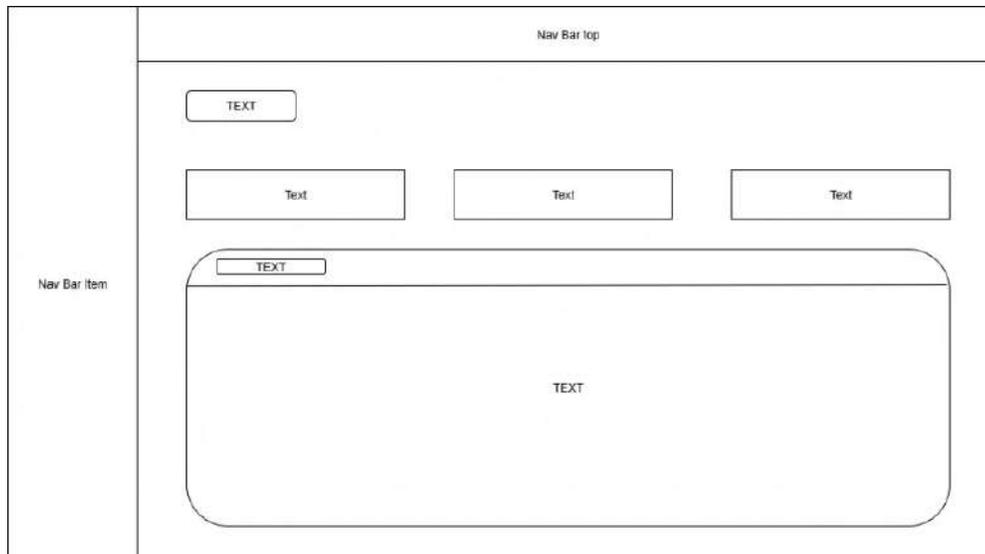


Gambar 3.5-22 Tampilan Dashboard Pengurus Laboratorium

Halaman ini ditampilkan setelah pengguna dengan peran Pengurus Laboratorium (*PLP*) berhasil masuk ke sistem. Dasbor ini menyajikan akses cepat terhadap fitur

utama yang berkaitan dengan pengelolaan alat laboratorium, seperti menambahkan data alat, membuat QRCode, melakukan validasi peminjaman dan pengembalian, serta mengakses laporan dan histori penggunaan. Desain disusun agar memudahkan PLP dalam mengakses seluruh fungsi operasional harian secara efisien.

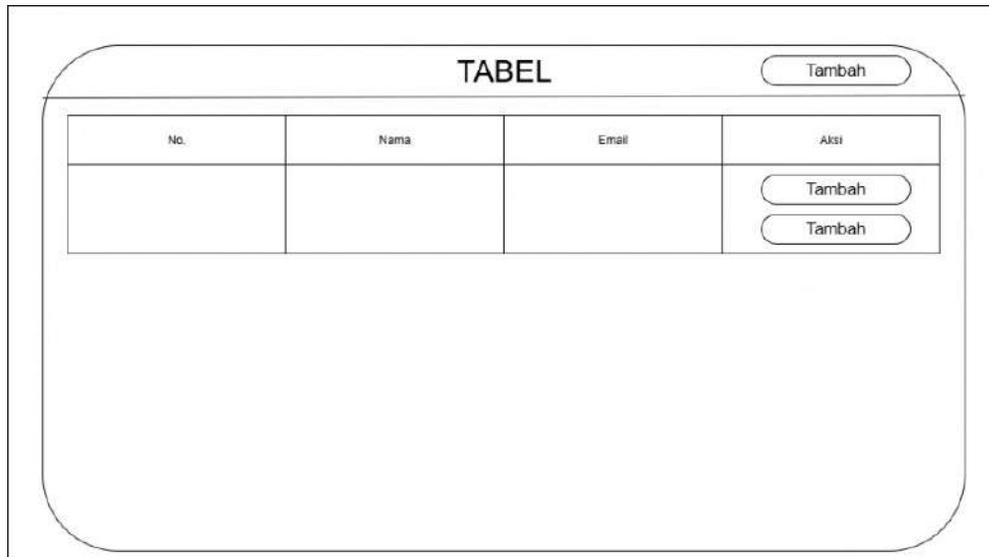
5. Halaman Dashboard Peminjam



Gambar 3.5-23 Tampilan Dashboard Peminjam

Dasbor ini merupakan tampilan awal yang diakses oleh pengguna dengan peran sebagai peminjam (mahasiswa atau dosen). Desain antarmuka fokus pada kemudahan navigasi dan penyajian informasi yang relevan bagi peminjam.

6. Halaman Tabel Data



Gambar 3.5-24 Tampilan Konten Tabel Data Dinamis

Halaman ini digunakan untuk menampilkan data dalam bentuk tabel dinamis yang disesuaikan berdasarkan jenis entitas yang sedang dikelola dalam system. Setiap baris memiliki kolom Aksi, berisi tombol-tombol interaktif seperti Detail, Edit, atau Hapus, sesuai kebutuhan pengelolaan data. Di bagian kanan atas halaman, terdapat tombol Tambah global yang akan membuka form *input* sesuai dengan jenis data yang sedang aktif ditampilkan.

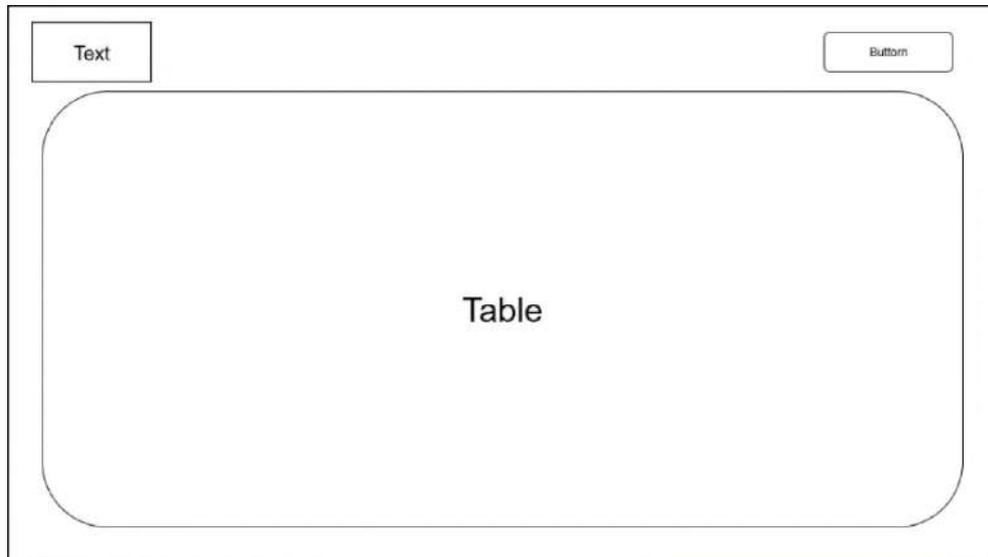
7. Halaman Generate QRCode

The screenshot shows a web interface titled "Generate QRCode". At the top, there is a text input field labeled "Text" containing the placeholder "Selected Data". Below this field are two buttons labeled "Button". In the center of the page, there is a square placeholder labeled "GAMBAR" (Image) with the word "Text" above it. Below the image placeholder is another button labeled "Button".

Gambar 3.5-25 Tampilan Generate QRCode

Fitur ini memungkinkan PLP menghasilkan QRCode unik untuk setiap alat laboratorium. Setelah data alat dimasukkan, sistem akan membuat QRCode yang dapat diunduh dan ditempel pada alat fisik, untuk kemudian dipindai saat proses peminjaman dan pengembalian.

8. Halaman Histroy



Gambar 3.5-26 Tampilan History

Halaman ini menampilkan riwayat aktivitas peminjaman dan pengembalian alat oleh pengguna. Informasi yang ditampilkan meliputi nama alat, tanggal peminjaman, tanggal pengembalian, status, dan validasi dari pengurus laboratorium. Fitur ini memudahkan pengguna memantau transaksi sebelumnya dan membantu pengurus dalam pelacakan data secara cepat.

9. Halaman Form *Input*

The image shows a web form titled "FORM" with a white background and a thin black border. At the top left, the word "FORM" is written in a bold, black, sans-serif font. Below the title, there are five text input fields arranged vertically. Each field is preceded by the word "Text" in a small, black, sans-serif font. The first three fields contain the text "Input Data", and the last two fields contain the text "Selected Data". At the bottom of the form, there are two rounded rectangular buttons. The left button is labeled "Button" and the right button is also labeled "Button". The entire form is centered on the page.

Gambar 3.5-27 Tampilan Form Input Data

Formulir ini terdiri dari beberapa label “Data” dan *input* field sejajar vertikal. Di bagian bawah terdapat dua tombol aksi (kiri dan kanan) yang dapat dikonfigurasi sebagai “Simpan” dan “Kembali” atau aksi lainnya. Desainnya bersih dan simetris untuk memudahkan proses pengisian data oleh pengguna.

3.6. Rancangan dan Implementasi Sistem

Rancangan dan pengembangan sistem merupakan tahapan di mana hasil perancangan diimplementasikan ke dalam bentuk perangkat lunak (*software*) yang dapat dijalankan dan diuji oleh pengguna. Sistem informasi manajemen peminjaman peralatan laboratorium yang dikembangkan pada proyek ini bersifat *web-based* dan dirancang untuk mendukung pengelolaan data secara daring dengan antarmuka yang responsif dan fungsionalitas yang sesuai dengan kebutuhan operasional laboratorium pendidikan.

Pengembangan sistem ini menggunakan kombinasi teknologi modern yang saling terintegrasi untuk memastikan sistem dapat berjalan secara efektif, efisien, dan mudah digunakan.

Adapun teknologi yang digunakan dalam pengembangan sistem ini adalah sebagai berikut:

1. **Bahasa Pemrograman: PHP**

Digunakan sebagai bahasa inti dalam pembangunan logika sistem sisi peladen (*server-side*). Bahasa ini dipilih karena kompatibilitasnya yang tinggi dengan sistem basis data serta dukungan komunitas yang luas.

2. **Kerangka Kerja (Framework): Laravel**

Laravel digunakan untuk mempercepat dan mempermudah pengembangan sistem. Kerangka kerja ini menyediakan struktur kode yang bersih dan terorganisasi dengan pendekatan *Model-View-Controller (MVC)*.

3. **Basis Data: MySQL**

Digunakan untuk menyimpan seluruh data sistem seperti data pengguna, data alat, transaksi peminjaman dan pengembalian, serta riwayat log aktivitas.

4. **Pengelola Basis Data: phpMyAdmin**

Merupakan antarmuka berbasis *web* yang memudahkan pengelolaan basis data MySQL tanpa harus menggunakan baris perintah.

5. **Lingkungan Pengembangan Lokal: Laragon**

Digunakan untuk membangun lingkungan *local server* yang mendukung *PHP*, *MySQL*, dan *Composer* secara cepat dan stabil selama proses pengembangan.

6. **Penyunting Kode Sumber: Visual Studio Code**

Digunakan untuk menulis, mengedit, dan mengelola seluruh kode program dalam proyek ini. Editor ini dipilih karena mendukung berbagai ekstensi dan fitur pengembangan yang canggih.

7. **Antarmuka Pengguna: Bootstrap**

Bootstrap digunakan untuk merancang antarmuka pengguna (*user interface*) yang responsif, konsisten, dan ramah bagi pengguna dari berbagai perangkat.

Fitur utama yang diimplementasikan dalam system ini meliputi:

1. *Login* multi-pengguna:

Sistem menyediakan fitur autentikasi yang mendukung tiga jenis peran, yaitu admin, peminjam (mahasiswa dan dosen), dan pengurus laboratorium (PLP), dengan hak akses yang berbeda-beda sesuai fungsinya.

2. Manajemen data peralatan laboratorium:

Pengurus laboratorium dapat menambahkan, mengubah, dan menghapus data alat, termasuk mengelompokkan alat berdasarkan kategori dan kondisi.

3. Peminjaman dan pengembalian alat:

Peminjam dapat mengajukan permintaan peminjaman alat melalui sistem. Pengurus laboratorium akan menerima permintaan tersebut dan memverifikasinya sesuai ketersediaan alat. Proses pengembalian juga dilakukan melalui sistem dan memerlukan persetujuan pengurus untuk diselesaikan.

4. Notifikasi keterlambatan otomatis:

Sistem dilengkapi dengan fitur pengingat otomatis yang dikirimkan melalui surat elektronik (*email*) kepada peminjam apabila terjadi keterlambatan pengembalian alat. Fitur ini menggunakan layanan *SMTP* dan dijalankan secara terjadwal dengan memanfaatkan *Laravel Scheduler* serta *cron job*.

5. Pemindaian QRCode alat:

Setiap alat laboratorium memiliki QRCode unik untuk mempercepat proses peminjaman dan pengembalian secara otomatis melalui pemindaian.

6. Logbook digital dan Riwayat Transaksi:

Seluruh aktivitas peminjaman dan pengembalian dicatat secara otomatis ke dalam sistem dalam bentuk *logbook* digital yang dapat diakses dan dievaluasi oleh pengurus laboratorium. Sistem juga mendukung pencetakan laporan untuk kebutuhan dokumentasi dan audit.

3.7. Pengujian Sistem

Pengujian sistem merupakan tahap penting dalam proses pengembangan perangkat lunak, yang bertujuan untuk memastikan bahwa setiap fitur telah berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan. Selain itu, pengujian juga dilakukan untuk mengidentifikasi kesalahan (*bug*) serta mengevaluasi kualitas sistem sebelum digunakan oleh pengguna secara nyata.

Dalam penelitian ini, pengujian sistem dilakukan menggunakan dua metode, yaitu *Black Box Testing* dan *User Acceptance Testing (UAT)*.

3.7.1. Black Box Testing

Black Box Testing merupakan metode pengujian yang berfokus pada pengujian fungsionalitas sistem tanpa melihat struktur internal atau kode program. Pengujian dilakukan dengan memberikan sejumlah masukan (*input*) pada fitur tertentu, dan kemudian mengevaluasi keluaran (*output*) yang dihasilkan apakah sesuai dengan yang diharapkan.

Setiap modul atau fitur diuji berdasarkan skenario penggunaan yang telah dirancang sebelumnya. Tujuan dari pengujian ini adalah untuk memastikan bahwa fungsi-fungsi utama seperti *login*, peminjaman, pengembalian, notifikasi, dan manajemen data dapat berjalan dengan benar.

Table 3.7-1 Tabel Parameter Black Box Testing

No.	Fitur	<i>Input</i>	<i>Output</i> Diharapkan
1.	<i>Login</i> sesuai hak akses	<i>Email</i> , kata sandi	<i>Login</i> berhasil sesuai role
2.	Validasi <i>login</i>	Username/kata sandi salah	Muncul pesan error dan tidak bisa <i>login</i>

3.	Tambah Peminjam dan Pengurus Laboratorium	Form isian lengkap	Tersimpan dan tampil
4.	Logout	Klik tombol logout	Kembali ke halaman <i>login</i> dan sesi berakhir
5.	Validasi kesalahan	<i>Input</i> kosong / format salah	Muncul pesan kesalahan yang jelas
6.	Persetujuan peminjaman	Klik tombol “Setujui” / “Tolak”	Status berubah menjadi Akses halaman alat
7.	Cek ketersediaan alat (stok)	Akses halaman alat	Data stok muncul sesuai jumlah aktual
8.	CRUD data alat	Nama alat, jumlah, deskripsi, dll.	Data alat bertambah / diubah / dihapus dan tampil pada daftar alat
9.	Generate QRCode	Klik “Generate QR” pada data alat	QRCode muncul dan bisa dipindai
10.	Unduh laporan atau riwayat	Klik tombol “Unduh Laporan”	File Excel atau PDF berhasil diunduh dan format rapi

3.7.2. *User Acceptance Testing (UAT)*

User Acceptance Testing (UAT) adalah pengujian yang dilakukan oleh pengguna akhir sistem untuk mengevaluasi apakah sistem yang dibangun telah sesuai dengan kebutuhan mereka secara fungsional maupun dari segi kemudahan penggunaan (*usability*).

Pengujian ini dilakukan oleh perwakilan dari masing-masing peran pengguna, yaitu admin, peminjam (mahasiswa/dosen), dan pengurus laboratorium (PLP). Setiap pengguna diberikan tugas untuk mencoba menjalankan fungsi-fungsi yang relevan dengan peran mereka, dan kemudian memberikan umpan balik terhadap kemudahan, kelengkapan, serta keakuratan sistem.

Instrumen yang digunakan pada kuesioner UAT dengan skala Likert 1–5, dengan makna sebagai berikut:

Skor	Deskripsi
1	Sangat Tidak Layak
2	Tidak Layak
3	Cukup Layak
4	Layak
5	Sangat Layak

Data dari kuesioner dihitung untuk memperoleh nilai rata-rata dan persentase kelayakan menggunakan rumus sebagai berikut:

1. Total Skor = $\sum(\text{Jumlah Responden pada tiap skor} \times \text{Bobot Skor})$
2. Skor Maksimum = Jumlah Responden \times Jumlah Pertanyaan \times 5
3. Skor rata-rata =
$$\frac{\text{Total Skor}}{\text{Jumlah Responden} \times \text{Jumlah Pertanyaan}}$$
4. Persentase kelayakan =
$$\frac{\text{Total Skor}}{\text{Skor Maksimum}} \times 100\%$$

Hasil dari UAT ini digunakan sebagai dasar evaluasi akhir sistem sebelum dipublikasikan atau diimplementasikan secara penuh di lingkungan laboratorium.

Untuk menyatakan bahwa sistem diterima oleh pengguna, maka:

- a) Rata-rata skor ≥ 3.5 per dimensi TAM dianggap positif (sukses)
- b) Jika < 3.5 , maka dianggap perlu perbaikan pada aspek tersebut

Klasifikasi berdasarkan persentase kelayakan:

Rentang Kriteria	Kriteria
0% - 20%	Sangat Tidak Layak
21% - 40 %	Tidak Layak
41% - 60%	Kurang Layak
61% - 80%	Layak
81% - 100%	Sangat Layak

Pengujian dilakukan dengan pendekatan *User Acceptance Testing* (UAT) berdasarkan *Technology Acceptance Model* (TAM), yang mencakup tiga dimensi:

A. *Perceived Usefulness* (Kegunaan yang Dirasakan)

Persepsi pengguna terhadap manfaat sistem dalam menunjang tugas-tugas mereka.

Table 3.7-2 Tabel Parameter Perceived Usefulness (PU)

No	Parameter Uji	Indikator Keberhasilan
1.	Sistem membantu menyelesaikan tugas peminjaman dengan efektif	Pengguna menyatakan tugas lebih cepat & efisien selesai
2.	Sistem meningkatkan produktivitas	Pengguna merasa jumlah pekerjaan meningkat atau proses lebih cepat
3.	Sistem relevan untuk tugas harian	Sistem digunakan untuk tugas yang memang dikerjakan pengguna
4.	Sistem meningkatkan kualitas kerja (akurasi, kecepatan)	Hasil kerja lebih rapi, data lebih akurat, lebih cepat
5.	Sistem bermanfaat secara keseluruhan dalam peran pengguna	Pengguna merasa sistem sangat mendukung aktivitas hariannya

B. *Perceived Ease of Use* (Kemudahan Penggunaan yang Dirasakan)

Persepsi pengguna terhadap kemudahan penggunaan sistem.

Table 3.7-3 Tabel Parameter *Perceived Ease of Use* (PEOU)

No	Parameter Uji	Indikator Keberhasilan
6.	Tampilan mudah digunakan dan tidak membingungkan	Navigasi intuitif, tidak membutuhkan pelatihan panjang
7.	Sistem mudah digunakan tanpa usaha besar	Akses cepat, tidak memerlukan waktu belajar lama
8.	Interaksi dengan sistem terasa alami	Button, form, dan alur kerja sesuai ekspektasi
9.	Pengguna cepat merasa mahir	Waktu adaptasi pendek, pengguna langsung paham cara pakai
10.	Akses informasi atau tindakan mudah dalam sistem	Pengguna tidak perlu banyak klik atau cari-cari menu

C. *General Acceptance* (Penerimaan Umum)

Persepsi umum terhadap kelayakan dan kenyamanan sistem secara keseluruhan.

Table 3.7-4 Tabel Parameter *General Acceptance* (GA)

No	Parameter Uji	Indikator Keberhasilan
11.	<i>Login</i> sesuai role berjalan dengan baik	Tidak salah akses, <i>login</i> valid sesuai peran
12.	Dashboard sesuai role	Menu dan data tampil sesuai tanggung jawab pengguna
13.	Fitur utama berfungsi sesuai tugas pengguna	Tidak ada error, fungsi utama dijalankan tanpa kendala
14.	Sistem memberi notifikasi atau umpan balik	Ada info atau alert saat data berubah
15.	Validasi kesalahan tampil jelas	Ada pesan saat <i>input</i> salah, pengguna tahu harus apa

16.	Navigasi dan pencarian mendukung pekerjaan	Menu mudah diakses, fitur pencarian berjalan baik
17.	Sistem mendukung pengelolaan data sesuai kebutuhan pengguna	CRUD alat/peminjaman berjalan sesuai role
18.	Performa sistem baik	Loading cepat, tidak berat
19.	Sistem layak digunakan secara keseluruhan	Pengguna puas dan tidak mengalami frustrasi saat menggunakan



BAB IV

PEMBAHASAN

4.1. Hasil Analisis Kebutuhan

Bagian ini memaparkan hasil analisis kebutuhan sistem berdasarkan wawancara langsung dengan Pengurus Laboratorium Pendidikan (PLP) dan studi dokumen seperti format logbook, daftar inventaris, serta prosedur operasional standar (POS) peminjaman alat. Proses analisis dilakukan untuk mengidentifikasi kebutuhan pengguna dari berbagai peran, yaitu admin, pengurus laboratorium, dan peminjam (mahasiswa atau dosen). Analisis ini menjadi landasan dalam merancang fungsionalitas sistem yang relevan dan tepat guna. Analisis kebutuhan ini disusun berdasarkan data yang diperoleh melalui wawancara kepada pengurus laboratorium, kuesioner dari pengguna, serta dokumentasi berupa format logbook dan inventaris laboratorium sebagaimana dijelaskan pada Subbab 3.4 Teknik Pengumpulan Data. Hasil analisis ini dirinci dalam dua kategori utama, yaitu kebutuhan fungsional dan nonfungsional.

4.1.1. Kebutuhan Fungsional

Kebutuhan fungsional mencakup fitur-fitur inti yang harus tersedia dalam sistem agar dapat mendukung proses kerja yang selama ini dilakukan secara manual, menjadi terdigitalisasi dan lebih efisien. Berdasarkan hasil analisis, kebutuhan fungsional sistem mencakup:

1. Autentikasi berbasis peran
Sistem harus mampu melakukan proses *login* sesuai dengan peran pengguna (admin, pengurus laboratorium, peminjam). Setiap peran memiliki hak akses dan fungsionalitas yang berbeda.
2. Manajemen data pengguna dan laboratorium
Admin dapat menambahkan data jurusan, laboratorium, serta akun pengguna, yang akan digunakan dalam proses peminjaman alat.

3. Pengelolaan inventaris peralatan laboratorium
Pengurus laboratorium dapat menambahkan, memperbarui, dan menandai status alat (tersedia, rusak, sedang dipinjam) secara terstruktur dan terintegrasi dalam sistem.
4. Proses peminjaman dan pengembalian alat
Peminjam dapat mengajukan peminjaman melalui formulir daring, yang kemudian divalidasi oleh pengurus laboratorium. Proses pengembalian juga tercatat secara otomatis setelah diverifikasi.
5. Pemindaian QRCode untuk validasi alat
Setiap alat memiliki QRCode yang dapat dipindai untuk mempercepat proses pencatatan peminjaman dan pengembalian.
6. Pencatatan otomatis dalam logbook digital
Setiap transaksi peminjaman dan pengembalian dicatat secara otomatis dalam *logbook* yang dapat diakses oleh pengurus laboratorium dan admin.
7. Fitur pengunduhan laporan
Pengurus laboratorium dapat mencetak atau mengunduh laporan aktivitas peminjaman dan status alat.
8. Pencatatan alat rusak dan penggantian
Sistem mencatat laporan kerusakan dan mendukung mekanisme kompensasi atau penggantian alat oleh peminjam.

4.1.2. Kebutuhan Nonfungsional

Kebutuhan nonfungsional adalah aspek yang tidak langsung berhubungan dengan fitur utama, tetapi penting untuk menjamin kinerja, keandalan, dan kenyamanan penggunaan sistem. Beberapa kebutuhan nonfungsional yang diidentifikasi, antara lain:

1. Kemudahan penggunaan (*usability*)
Antarmuka sistem harus mudah dipahami oleh semua jenis pengguna, termasuk pengguna yang belum terbiasa menggunakan sistem informasi.

2. Aksesibilitas melalui jaringan internet
Sistem harus dapat diakses secara daring oleh pengguna tanpa bergantung pada perangkat tertentu, melalui browser yang umum digunakan.
3. Keamanan data pengguna dan transaksi
Data pengguna dan riwayat transaksi harus terlindungi dengan sistem autentikasi dan pembatasan hak akses.
4. Notifikasi otomatis melalui surat elektronik (*email*)
Sistem harus mengirimkan pengingat otomatis kepada peminjam terkait batas waktu pengembalian alat atau status keterlambatan.
5. Ketersediaan sistem secara stabil
Sistem diharapkan berjalan tanpa gangguan teknis, baik saat digunakan dalam jaringan lokal maupun saat *dihosting* secara daring.

4.2. Hasil Perancangan Sistem

Tahapan perancangan sistem telah menghasilkan sejumlah komponen utama yang menjadi dasar implementasi dari sistem informasi manajemen peminjaman peralatan laboratorium. Perancangan dilakukan secara iteratif menggunakan pendekatan *Rapid Application Development (RAD)*, serta disesuaikan berdasarkan masukan langsung dari Pengurus Laboratorium Pendidikan (PLP).

Komponen perancangan sistem mencakup:

1. Arsitektur sistem, yang menggambarkan alur data dan hubungan antara pengguna (admin, pengurus laboratorium, dan peminjam) dengan proses utama sistem.
2. Diagram *use case*, yang menggambarkan interaksi antara tiap peran pengguna dan fungsionalitas sistem. Diagram ini telah ditampilkan sebelumnya pada Gambar 3.5-4 Diagram *Use Case*
3. Diagram aktivitas (*activity diagram*), yang mengilustrasikan urutan logis aktivitas pengguna dalam sistem, seperti proses peminjaman, pengembalian, serta pengelolaan inventaris. Detail aktivitas ditunjukkan melalui Gambar 3.5-5 *Activity Diagram Admin (Jurusan)* hingga Gambar 3.5-17 *Activity Diagram Peminjam*

4. Diagram Kelas (*class diagram*), yang menunjukkan struktur statis dari sistem, mencakup entitas utama (kelas), atribut, metode, serta hubungan antar kelas. Diagram ini memberikan representasi mengenai bagaimana data dan logika sistem diorganisasi secara objek. Diagram lengkap disajikan pada Gambar 3.5-18 *Class Diagram*
5. Rancangan antarmuka pengguna (*user interface*), yang disusun dalam bentuk *wireframe* mencakup tampilan halaman masuk (*login*), *dashboard*, formulir *input* data, serta tabel dinamis. Rancangan ini ditampilkan dalam Gambar 3.5-19 Login UI hingga Gambar 3.5-27 Tampilan Form Input Data.

Sistem ini dirancang untuk mendukung beberapa fungsi utama, yaitu:

1. Sistem secara otomatis mencatat setiap transaksi peminjaman dan pengembalian alat pada saat pengguna melakukan tindakan tersebut. Dengan demikian, data yang tersimpan dalam basis data akan diperbarui secara *real-time* tanpa memerlukan entri manual tambahan, sehingga meningkatkan akurasi dan efisiensi pencatatan.
2. Sistem menyediakan logbook digital yang dapat diakses oleh pengurus laboratorium sebagai sarana pencatatan riwayat peminjaman alat, yang berguna untuk proses pelacakan aset dan audit internal.
3. Sistem dilengkapi fitur pengingat otomatis berupa surat elektronik (email) yang dikirim melalui Simple Mail Transfer Protocol (SMTP) kepada pengguna yang terlambat melakukan pengembalian atau yang melaporkan kerusakan peralatan.
4. Sistem diintegrasikan dengan teknologi Quick Response (QR) code untuk mempermudah proses identifikasi alat secara digital, sekaligus mengurangi potensi kesalahan dalam proses entri data.
5. Alur kerja sistem disesuaikan dengan Prosedur Operasional Standar (POS) yang berlaku di lingkungan laboratorium, guna memastikan kesesuaian sistem dengan ketentuan dan tata kelola yang telah ditetapkan.

Seluruh hasil perancangan telah ditampilkan pada BAB III sebagai representasi visual dan teknis yang menjadi pedoman utama dalam proses implementasi serta pengujian sistem di tahap selanjutnya. Perancangan sistem ini merupakan implementasi dari tahapan *user design* pada metodologi RAD yang telah diuraikan pada Subbab 3.2 dan divisualisasikan dalam diagram dan *mockup* pada Subbab 3.5. Perancangan sistem ini menjadi landasan utama dalam tahap implementasi yang akan dijelaskan pada Subbab 4.3

4.3. Hasil Implementasi Sistem

4.3.1. Arsitektur Sistem

Sistem informasi manajemen peminjaman peralatan laboratorium ini dibangun menggunakan arsitektur *Model-View-Controller* (MVC) yang diimplementasikan melalui *framework* Laravel. Arsitektur ini digunakan untuk memisahkan logika bisnis, antarmuka pengguna, dan pengelolaan data dalam struktur yang lebih modular dan mudah dipelihara.

Laravel mendukung struktur MVC secara langsung dan menyediakan *templating engine* bernama Blade, yang digunakan dalam bagian *View* untuk merancang antarmuka sistem.

Berikut adalah implementasi masing-masing komponen berdasarkan kode sistem yang dikembangkan:

1. Model

Model digunakan untuk mewakili struktur data dan berkomunikasi dengan basis data. Dalam sistem ini, salah satu model yang dibuat adalah Peminjaman.php, yang merepresentasikan data peminjaman alat. Model ini terhubung dengan tabel peminjaman dalam basis data dan digunakan untuk menyimpan serta memproses data transaksi peminjaman.

```
1 <?php
2 namespace App\Models;
3 use Illuminate\Database\Eloquent\Model;
4
5 class mahasiswa extends Model
6 {
7     //
8     protected $table = 'mahasiswas';
9     protected $fillable = ['nama', 'npm', 'role_id', 'user_id', 'kelas_id', 'no_telp'];
10
11     public function role()
12     {
13         return $this->belongsTo(\App\Models\roles::class, 'role_id');
14     }
15
16     public function user()
17     {
18         return $this->belongsTo(\App\Models\User::class, 'user_id');
19     }
20
21     public function kelas()
22     {
23         return $this->belongsTo(\App\Models\kelas::class, 'kelas_id');
24     }
25
26     public function prodi()
27     {
28         return $this->kelas ? $this->kelas->prodi() : null;
29     }
30
31     public function jurusan()
32     {
33         return $this->kelas && $this->kelas->prodi ? $this->kelas->prodi->jurusan() : null;
34     }
35 }
```

Gambar 4.3-1 Contoh Model

2. View

View merupakan bagian dari sistem yang menangani tampilan antarmuka pengguna. View ditulis dalam berkas Blade (.blade.php) dan memuat struktur halaman seperti formulir, tabel, dan elemen visual lainnya yang ditampilkan kepada pengguna.

```
1 @extends('admin.layouts.admin')
2 @section('title', 'Edit Data Pengurus')
3
4 @section('content')
5 <div class="container-fluid">
6   <h1 class="h3 mb-4 text-primary fw-bold text-center"><i class="fas fa-edit me-2"></i>Update
   Pengurus Laboratorium</h1>
7   <div class="row justify-content-center">
8     <div class="col-12 col-lg-8 col-xl-6">
9       <div class="card shadow-lg border-0 rounded-4">
10        <div class="card-header bg-gradient bg-primary text-white rounded-top-4">
11          <strong><i class="fas fa-user-cog me-2"></i>Edit Pengurus Laboratorium</strong>
12        </div>
13        <div class="card-body bg-light rounded-bottom-4">
14          <form action="{{ route('proedit-pengurus', $pengurus->id) }}" method="POST"
15            autocomplete="off">
16            @csrf
17            <div class="mb-3">
18              <label class="form-label fw-sembold">Role</label>
19              <input type="text" class="form-control rounded-pill px-3 py-2" value="{{
20                $pengurus->role ? $pengurus->role->nama : 'pengurus' }}" readonly>
21            </div>
22            <div class="mb-3">
23              <label for="nama" class="form-label fw-sembold">Nama Pengurus</label>
24              <input type="text" name="nama" id="nama" value="{{ $pengurus->nama }}"
25                class="form-control rounded-pill px-3 py-2" placeholder="Masukkan nama pengurus" required autofocus>
26            </div>
27            <div class="mb-3">
28              <label for="laboratorium_id" class="form-label fw-
29                semibold">Laboratorium</label>
30              <select name="laboratorium_id" id="laboratorium_id" class="form-control
31                rounded-pill px-3 py-2" required>
32                <option value="">--- Pilih Laboratorium ---</option>
33                @foreach($labs as $lab)
34                  <option value="{{ $lab->id }}" {{ $pengurus->laboratorium_id ==
35                    $lab->id ? 'selected' : '' }}>{{ $lab->nama }}</option>
36                @endforeach
37              </select>
38            </div>
39          </div>
40        </div>
41      </div>
42    </div>
43  </div>
44 </div>
```

Gambar 4.3-2 Contoh View

3. Controller

Controller bertugas sebagai penghubung antara Model dan View. Komponen ini menerima masukan dari pengguna melalui request, memproses data melalui Model, dan mengembalikan hasil ke View.

```
1 <?php
2 namespace App\Http\Controllers;
3 use Illuminate\Http\Request;
4 use Illuminate\Support\Facades\Hash;
5 use App\Models\jurusan;
6
7 class AdminController extends Controller
8 {
9     //Jurusan
10    public function mn_jur() {
11        $jurs = jurusan::orderBy('nama')->paginate(5);
12        return view('admin.Manajemen.manajemen_jurusan',compact('jurs'));
13    }
14    public function jurregister(){
15        return view('admin.form.form_jurusan');
16    }
17    function nambahjur(Request $request){
18        $request->validate([
19            'nama'=> 'required|min:5|max:255',
20        ]);
21        $datajur = [
22            'nama' =>$request->nama
23        ];
24        jurusan::insert($datajur);
25        return redirect()->route('jurusan_manajemen')->with('success','Menambahkan Berhasil');
26    }
27    public function editjur($id){
28        $jurs = jurusan::where('id',$id)->first();
29        $datajur = [
30            'jurusan' => $jurs
31        ];
32        return view('admin.edit_form.edit_jurusan',$datajur);
33    }
34    public function updatejur(Request $request,$id){
35        $nama = $request->input('nama');
36        $jurupdate = [
37            'nama'=>$nama
38        ];
39        jurusan::where('id',$id)->update($jurupdate);
40        return redirect()->route('jurusan_manajemen')->with('success','Berhasil Di Ubah');
41    }
42    public function deljur($id){
43        $user = jurusan::findOrFail($id);
44        $user->delete();
45        return redirect()->route('jurusan_manajemen')->with('success','Data Berhasil Dihapus');
46    }
47 }
```

Gambar 4.3-3 Contoh Controller

4. Route

Laravel mengatur rute (routing) dalam file web.php. Rute digunakan untuk menentukan URL yang diarahkan ke fungsi tertentu di dalam controller. Misalnya, rute untuk menyimpan peminjaman alat:

```
1 <?php
2 use App\Http\Controllers\AdminController;
3 use App\Mail\ReminderPengembalianMail;
4 use Illuminate\Support\Facades\Mail;
5 use Illuminate\Support\Facades\Route;
6
7 Route::get('/', function () {
8     return redirect()->route('login');
9 });
10
11 // Login routes
12 Route::get('/login', [AuthController::class, 'showLoginForm'])->name('login');
13 Route::post('/login', [AuthController::class, 'login'])->name('login');
14 Route::post('/logout', [AuthController::class, 'logout'])->name('logout');
15
16 // Route profil dinamis untuk semua role
17 Route::get('/profil', [AuthController::class, 'profil'])->name('profil');
18
19 //Admin//
20 Route::middleware(["auth", "verified", "role:admin"])->group(function () {
21     Route::get('/dashboard_admin', [AdminController::class, 'dshb_admin'])->name('admin_dashboard');
22     Route::get('/profil_admin', [AdminController::class, 'profile'])->name('profil_admin');
23     Route::get('/manajemen_jurusan', [AdminController::class, 'mnj_jur'])->name('jurusan_manajemen');
24     Route::get('/manajemen_jenjang', [AdminController::class, 'mnj_jenjang'])->
25     >name('jenjang_manajemen');
26     Route::get('/manajemen_prodi', [AdminController::class, 'mnj_prodi'])->name('prodi_manajemen');
27     Route::get('/manajemen_kelas', [AdminController::class, 'mnj_kelas'])->name('kelas_manajemen');
28     Route::get('/manajemen_mahasiswa', [AdminController::class, 'mnj_mhs'])->
29     >name('mahasiswa_manajemen');
30     Route::get('/manajemen_dosen', [AdminController::class, 'mnj_dsn'])->name('dosen_manajemen');
31     Route::get('/manajemen_pengurus-lab', [AdminController::class, 'pngr_lab'])->
32     >name('pengurus_manajemen');
33     Route::get('/manajemen_lab', [AdminController::class, 'mnj_lab'])->name('lab_manajemen');
34     Route::get('/manajemen_admin', [AdminController::class, 'mnj_admin'])->name('admin_manajemen');
35     Route::get('/manajemen_role', [AdminController::class, 'mnj_roles'])->name('role_manajemen');
36 });
```

Gambar 4.3-4 Contoh Route

4.3.2. User Interface

Antarmuka pengguna (*user interface*) merupakan bagian penting dari sistem yang memfasilitasi interaksi antara pengguna dan fungsi-fungsi sistem. Dalam proyek ini, antarmuka sistem informasi manajemen peminjaman peralatan laboratorium dirancang menggunakan *framework* Bootstrap, yang dipilih karena kemampuannya dalam menghasilkan tampilan *web* yang responsif, konsisten, serta mudah dikustomisasi.

Bootstrap memungkinkan sistem untuk tampil optimal di berbagai ukuran layar dan perangkat, baik desktop maupun perangkat bergerak (*mobile-friendly*), tanpa perlu

merancang antarmuka secara terpisah untuk masing-masing jenis perangkat. Penggunaan komponen antarmuka siap pakai yang disediakan oleh Bootstrap juga mempercepat proses implementasi dan menjaga keseragaman desain antarhalaman. Sistem ini dibangun untuk mendukung tiga jenis peran pengguna, masing-masing dengan hak akses dan tanggung jawab yang berbeda, yaitu:

1. Admin

Memiliki hak akses untuk mengelola data jurusan, jurusan, jenjang, program studi, kelas, laboratorium, serta menambahkan peminjam dan pengurus laboratorium.

2. Pengurus Laboratorium (PLP)

bertanggung jawab untuk menambahkan data alat, membuat dan mencetak QRCode, melakukan verifikasi terhadap pengajuan peminjaman dan pengembalian alat, mengelola data alat rusak, serta mengunduh laporan dan riwayat aktivitas.

3. Peminjam (Dosen dan Mahasiswa)

Dapat mengajukan peminjaman dan melakukan pengembalian alat sesuai prosedur.

Beberapa fitur utama sistem yang telah diimplementasikan melalui antarmuka pengguna antara lain:

- 1) Autentikasi pengguna dengan *login* berbasis peran
- 2) Formulir peminjaman alat yang terintegrasi dengan database alat
- 3) Pemindaian QRCode untuk validasi alat saat pengembalian
- 4) Pengiriman *notifikasi* otomatis melalui *email* untuk keterlambatan pengembalian
- 5) Log aktivitas peminjaman dan pengembalian secara digital

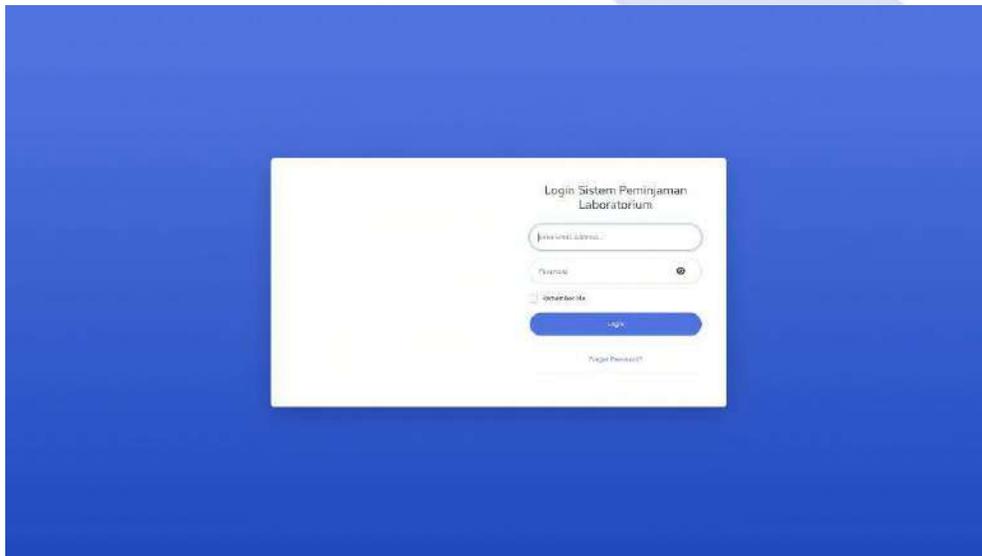
Berdasarkan hasil implementasi, seluruh antarmuka berfungsi dengan baik dan konsisten sesuai dengan rancangan. Navigasi antarmuka berjalan lancar, informasi ditampilkan dengan jelas, dan pengalaman pengguna selama uji coba menunjukkan

bahwa desain yang digunakan telah mendukung efisiensi, kemudahan, serta ketepatan dalam penggunaan sistem.

1. Halaman *Login*

Halaman *login* digunakan untuk mengautentikasi pengguna berdasarkan informasi kredensial berupa email dan kata sandi. Setelah proses *login* berhasil, sistem akan secara otomatis mengarahkan pengguna ke halaman dashboard sesuai dengan perannya.

Halaman ini dirancang secara sederhana dan profesional agar memudahkan pengguna dalam melakukan proses masuk tanpa kebingungan. Validasi akan dilakukan oleh sistem jika terjadi kesalahan *input*, seperti email tidak terdaftar atau kata sandi tidak sesuai.

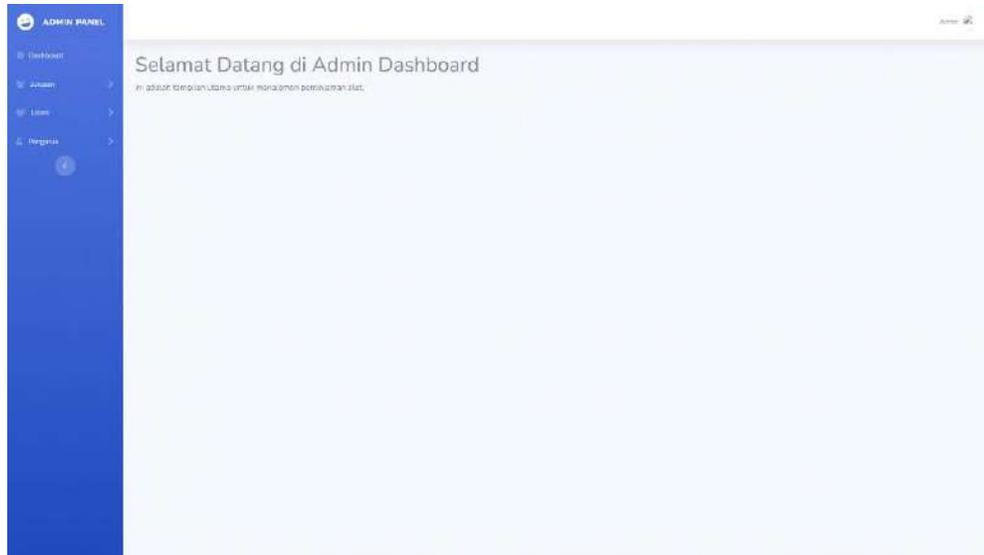


Gambar 4.3-5 Tampilan Halaman Login

2. Admin

a) Dashboard Admin

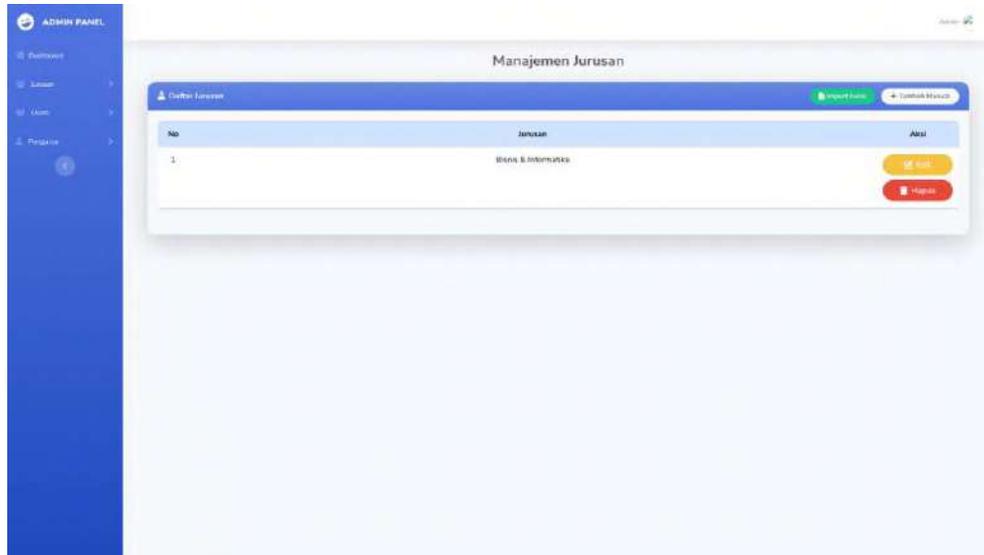
Halaman Dashboard Admin merupakan tampilan awal yang akan dituju setelah seorang admin berhasil masuk ke dalam sistem. Halaman ini menyajikan navigasi utama berupa menu yang terletak di sisi kiri



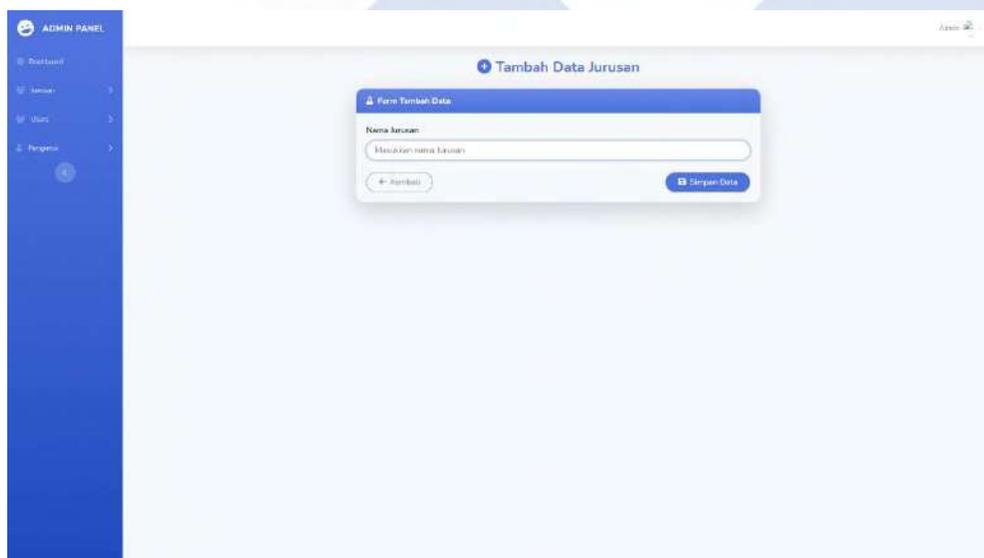
Gambar 4.3-6 Tampilan Halaman Dashboard Admin

b) Jurusan

Halaman Jurusan digunakan oleh admin untuk mengelola data jurusan yang tersedia di institusi. Data jurusan ini menjadi informasi pendukung yang akan terhubung langsung dengan data peminjam dan laboratorium, sehingga keterhubungan antarentitas dalam sistem dapat diatur secara sistematis.



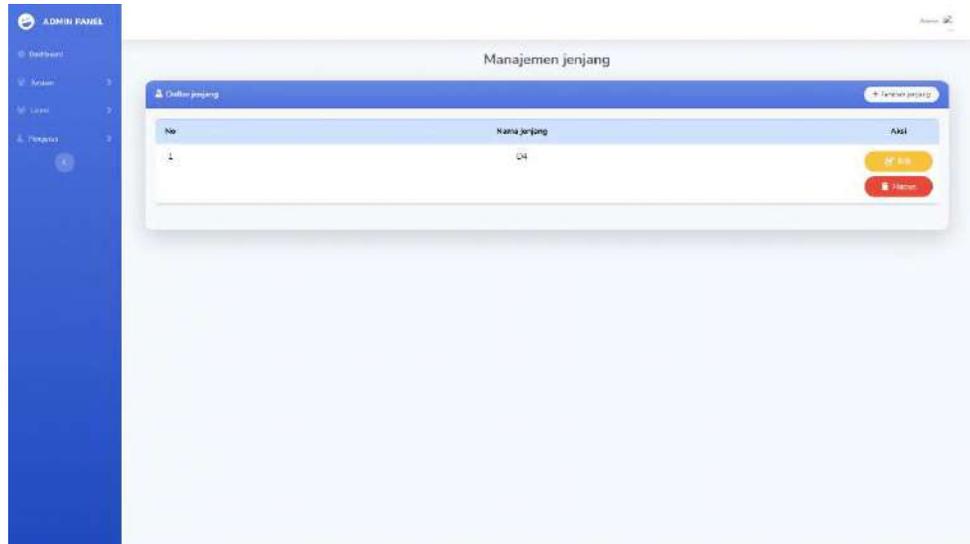
Gambar 4.3-7 Tampilan Halaman Jurusan



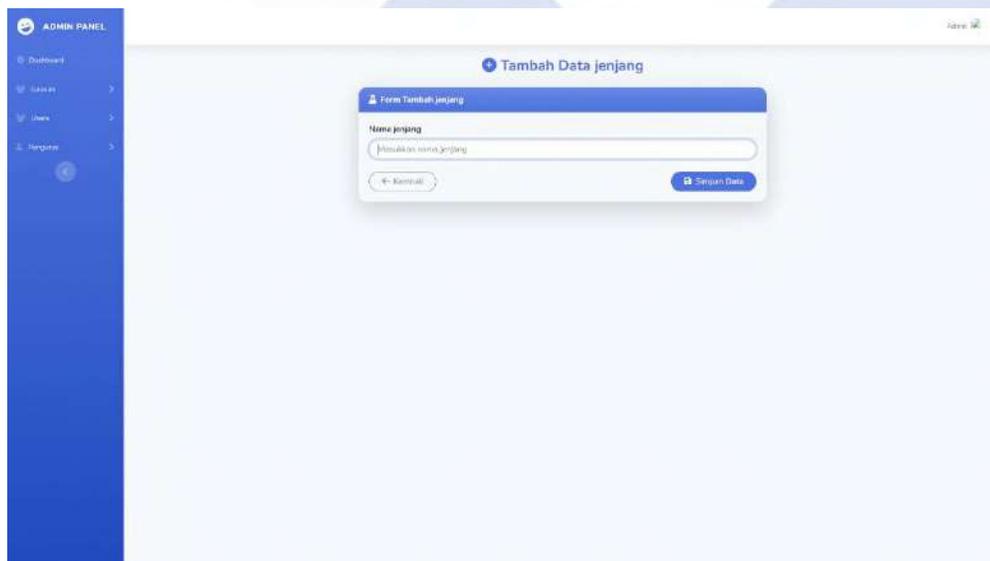
Gambar 4.3-8 Tampilan Halaman Form Jurusan

c) Jenjang

Halaman Jenjang berfungsi sebagai tempat untuk mengelola data jenjang pendidikan di lingkungan institusi, seperti *D3*, *D4*, atau *S1*. Data jenjang ini digunakan sebagai informasi pendukung yang akan terhubung dengan entitas jurusan dan peminjam, khususnya mahasiswa.



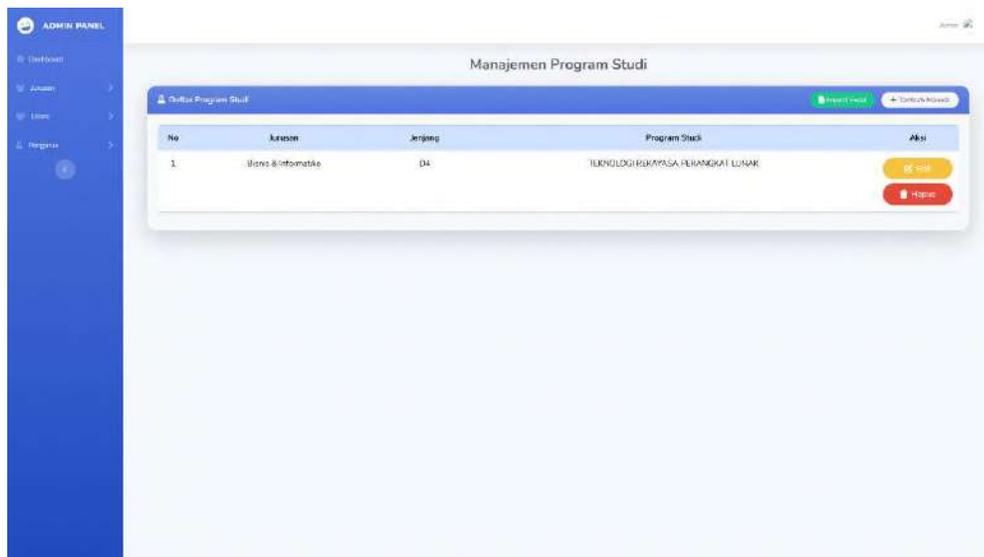
Gambar 4.3-9 Tampilan Halaman Jenjang



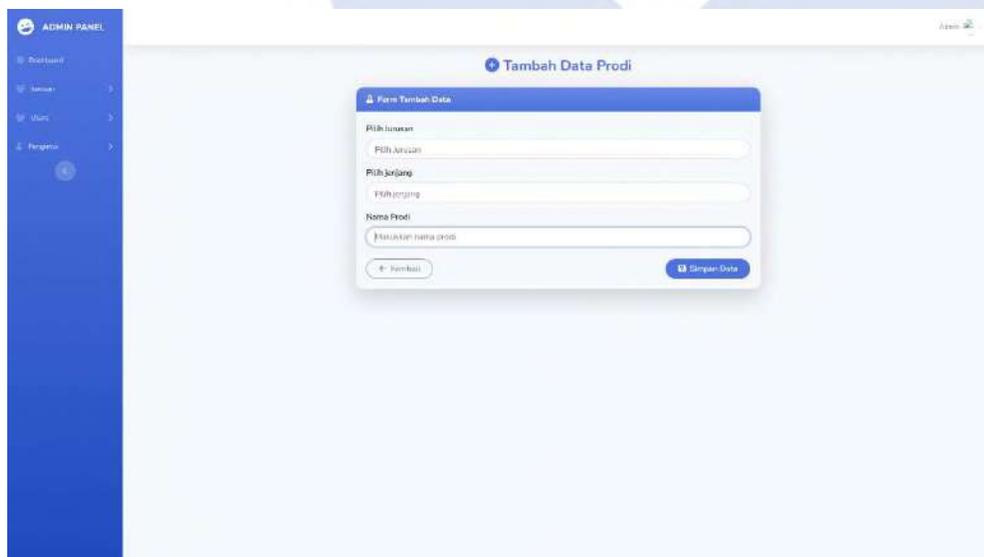
Gambar 4.3-10 Tampilan Halaman Form Jenjang

d) Program Studi

Halaman Program Studi digunakan untuk mengelola data prodi yang terdapat dalam institusi. Admin dapat menambahkan, mengubah, atau menghapus informasi program studi. Data ini secara otomatis akan dikaitkan dengan data jurusan dan jenjang, sehingga struktur data pengguna (peminjam) dapat terbentuk dengan relasi yang jelas.



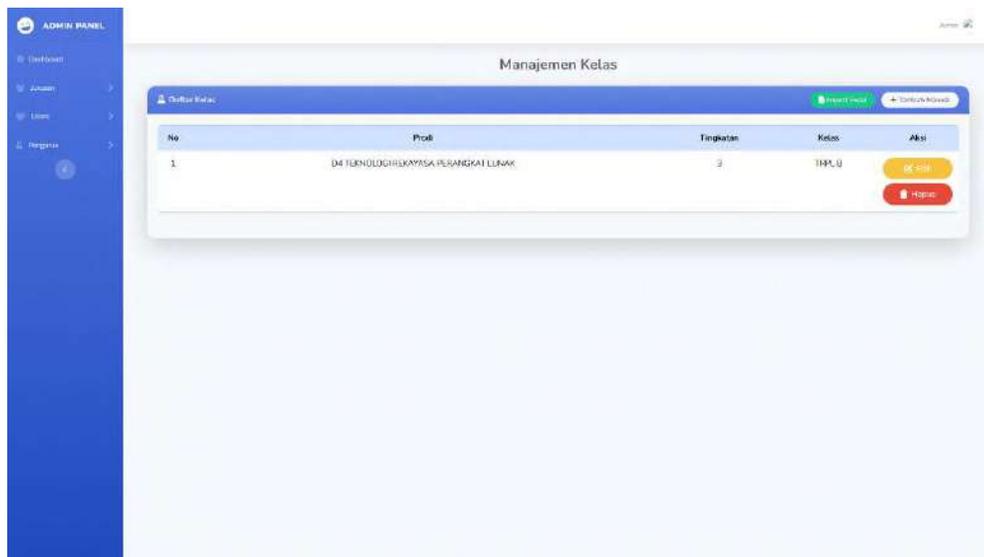
Gambar 4.3-11 Tampilan Halaman Program Studi



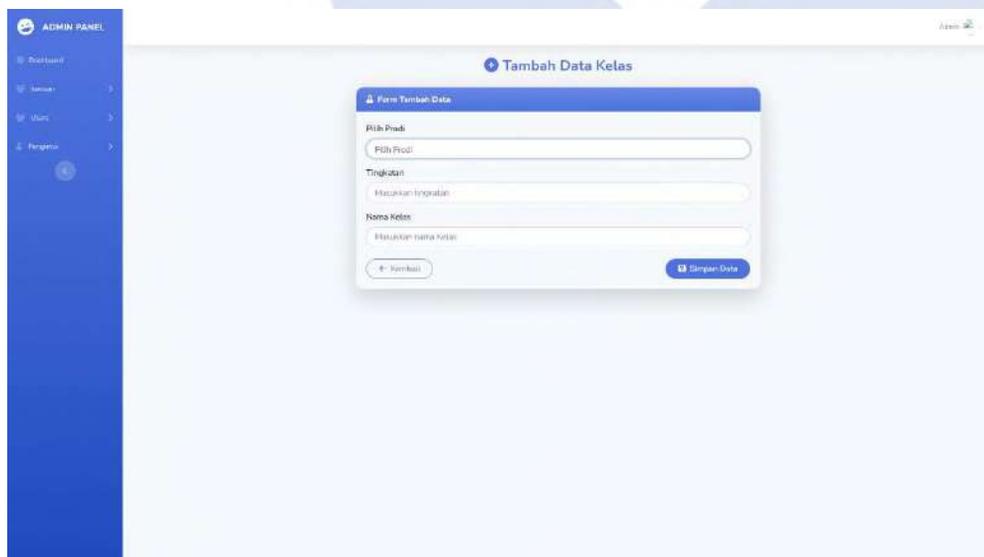
Gambar 4.3-12 Tampilan Halaman Form Program Studi

e) Kelas

Halaman Kelas memungkinkan admin menambahkan informasi kelas untuk peminjam yang berasal dari kalangan mahasiswa. Data kelas penting sebagai identifikasi tambahan dalam profil peminjam. Informasi kelas akan mempermudah pelacakan jika diperlukan pengelompokan data berdasarkan angkatan atau kelas tertentu.



Gambar 4.3-13 Tampilan Halaman Kelas

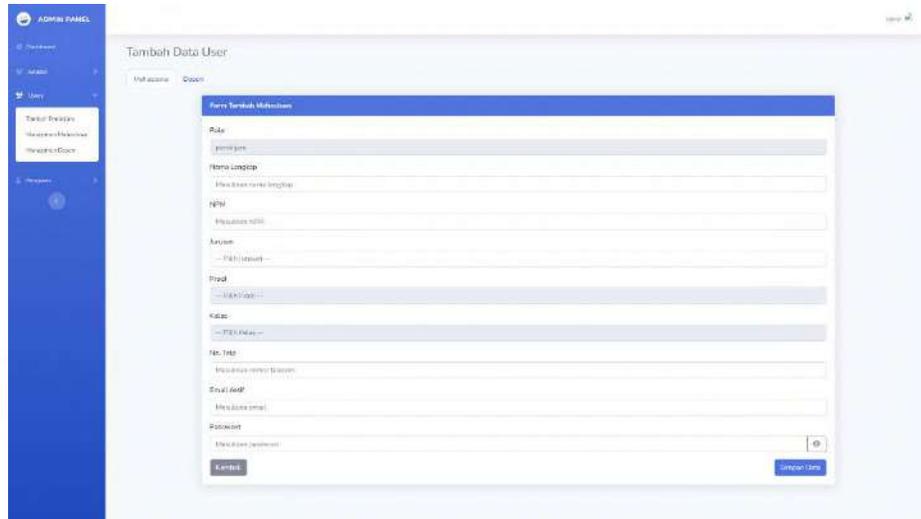


Gambar 4.3-14 Tampilan Halaman Form Kelas

f) Peminjam

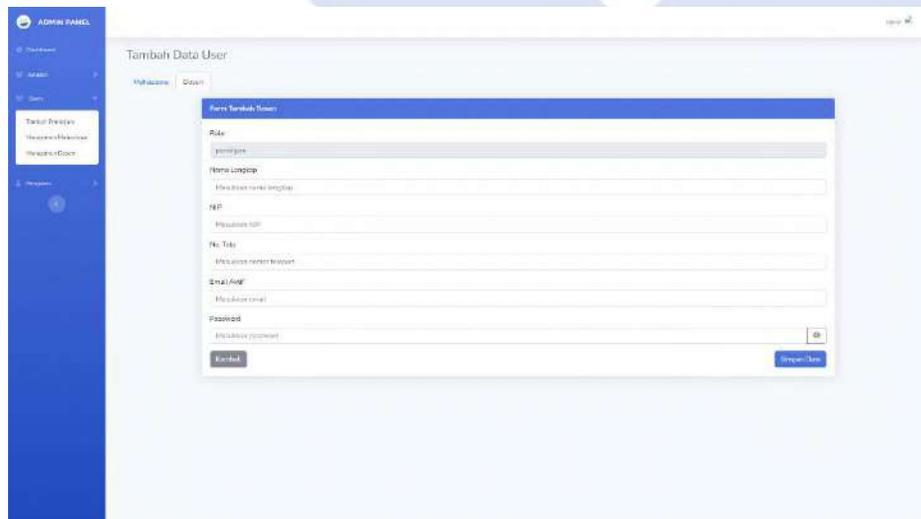
Sistem menyediakan dua jenis peminjam, yaitu mahasiswa dan dosen. Oleh karena itu, pengelolaan data peminjam dibagi ke dalam beberapa halaman:

- 1) Halaman Tambah Mahasiswa berfungsi untuk menambahkan peminjam dengan atribut seperti nama lengkap, NIM, program studi, dan kelas.



Gambar 4.3-15 Tampilan Halaman Tambah Mahasiswa

- 2) Halaman Tambah Dosen digunakan untuk menambahkan peminjam yang berasal dari kalangan dosen, dengan data seperti nama, NIP, dan jurusan.



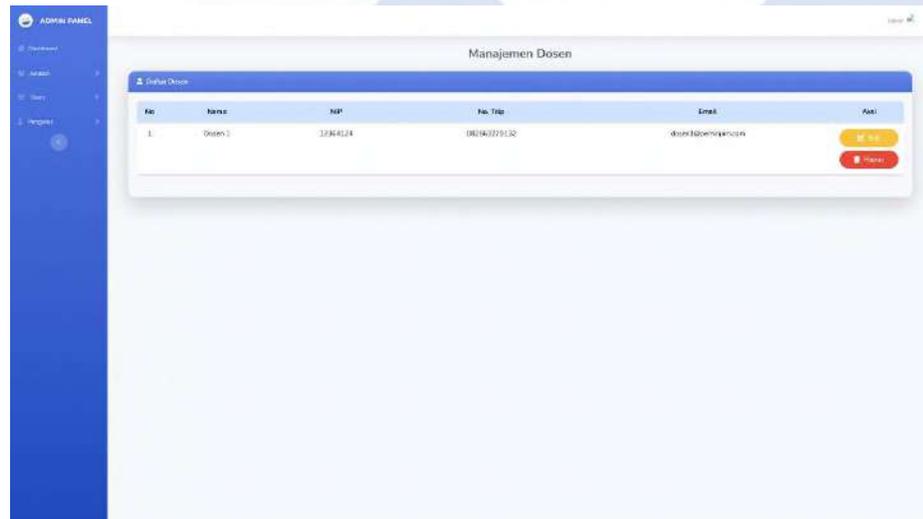
Gambar 4.3-16 Tampilan Halaman Tambah Dosen

- 3) Halaman Mahasiswa menampilkan seluruh daftar mahasiswa yang telah terdaftar sebagai peminjam, lengkap dengan fitur pencarian, pengeditan, dan penghapusan data.



Gambar 4.3-17 Tampilan Halaman Mahasiswa

- 4) Halaman Dosen menampilkan data peminjam dari kalangan dosen dengan fitur yang serupa.

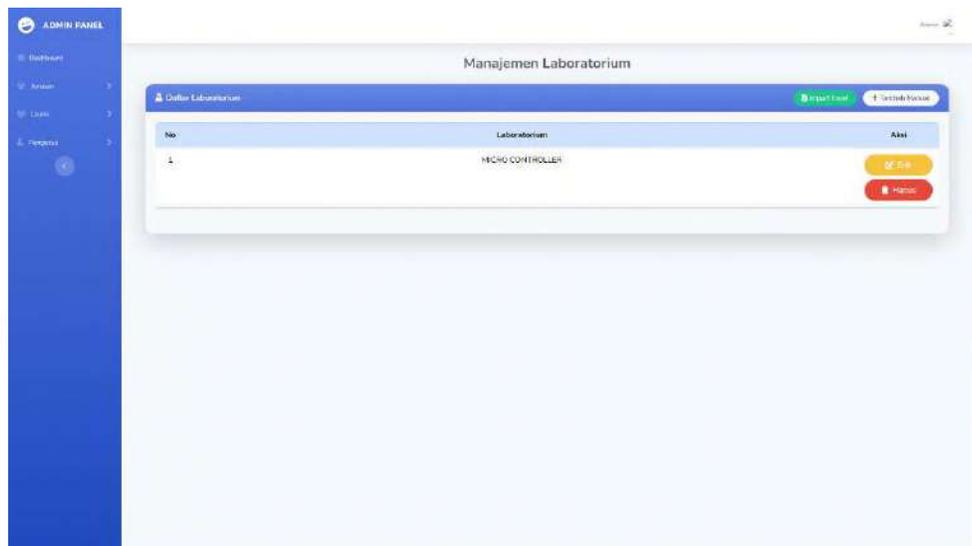


Gambar 4.3-18 Tampilan Halaman Dosen

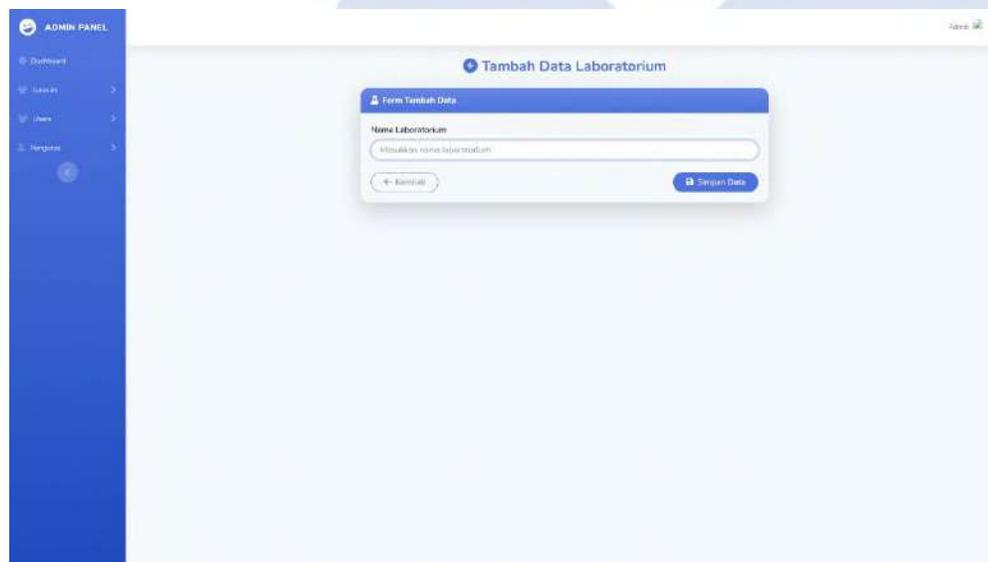
g) Laboratorium

Halaman Laboratorium digunakan untuk mendata laboratorium yang tersedia dalam institusi. Data ini dibutuhkan untuk mengelompokkan alat, pengguna, dan pengurus sesuai lokasi laboratorium masing-masing.

Admin dapat menambahkan nama laboratorium dan mengelola informasi terkait, sehingga struktur laboratorium dapat tercatat secara rinci.



Gambar 4.3-19 Tampilan Halaman Laboratorium

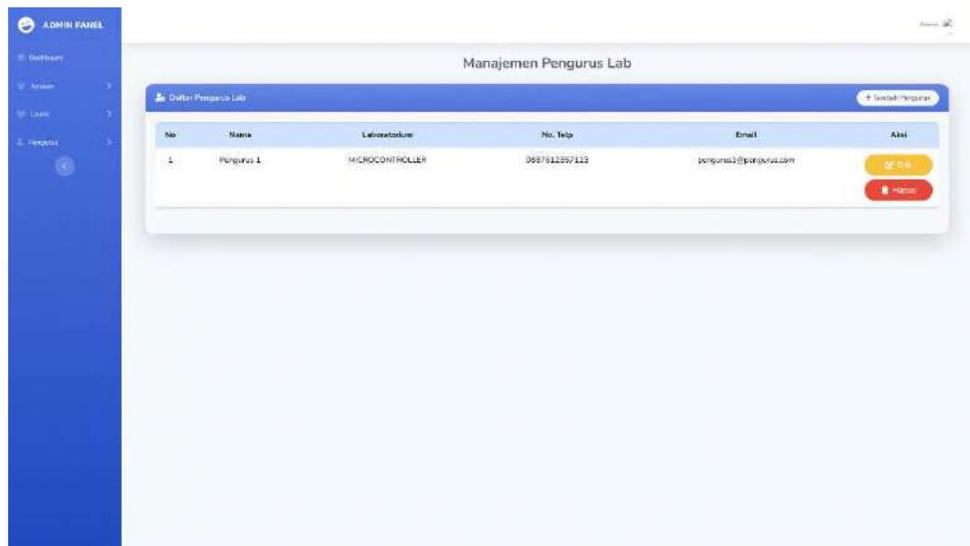


Gambar 4.3-20 Tampilan Halaman Form Laboratorium

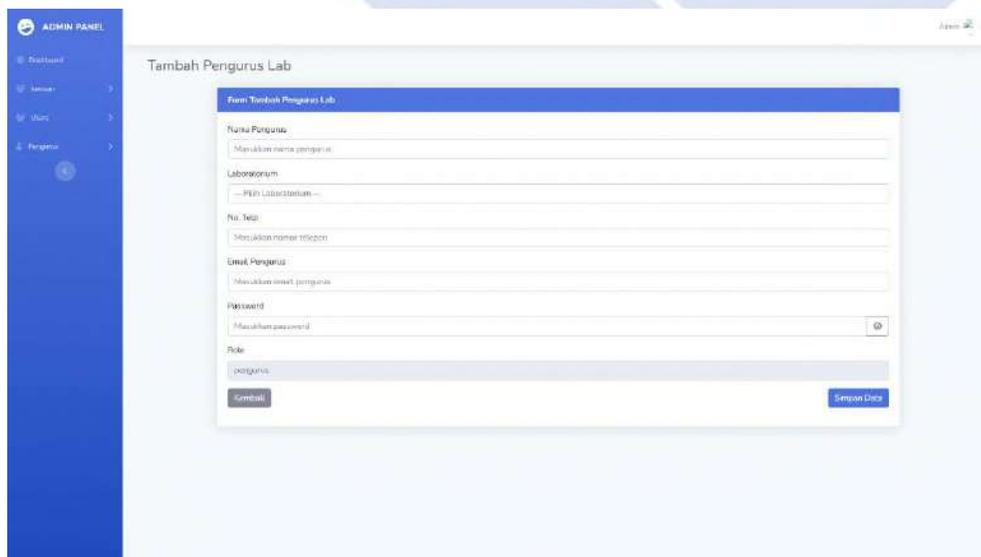
h) Pengurus Laboratorium

Halaman ini menampilkan daftar pengurus laboratorium (*Pranata Laboratorium Pendidikan – PLP*) dan digunakan untuk menambahkan, mengelola, serta menetapkan pengurus ke laboratorium tertentu.

Setiap pengurus akan dikaitkan dengan satu atau lebih laboratorium. Pengurus ini memiliki hak akses khusus dalam sistem, termasuk mengelola data alat, validasi peminjaman, dan pelaporan.



Gambar 4.3-21 Tampilan Halaman Pengurus Laboratorium



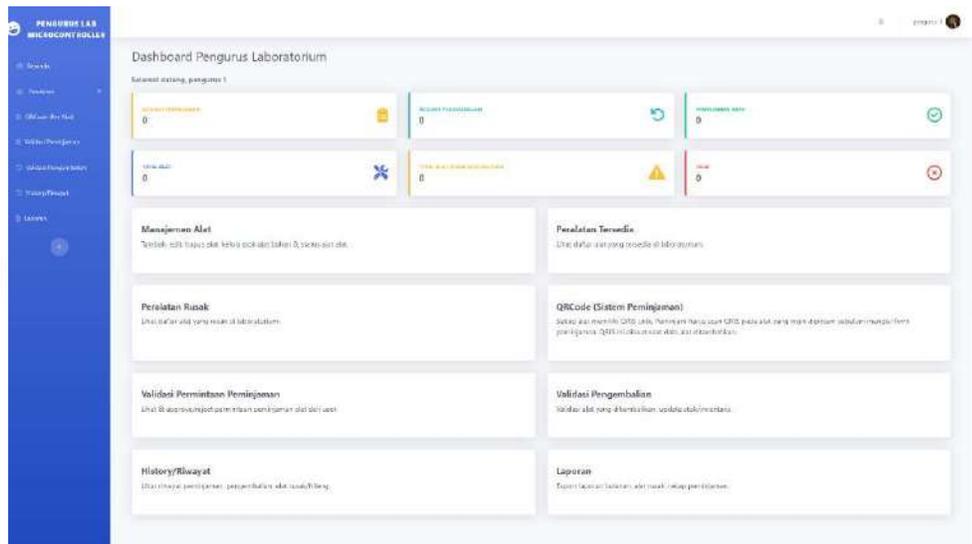
Gambar 4.3-22 Tampilan Halaman Form Pengurus Laboratorium

3. Pengurus Laboratorium

a) Dashboard Pengurus Laboratorium

Dashboard ini merupakan halaman utama bagi Pengurus Laboratorium setelah berhasil *login*. Halaman ini menyajikan navigasi cepat ke fitur-fitur utama seperti manajemen alat, validasi peminjaman dan pengembalian, serta akses ke laporan.

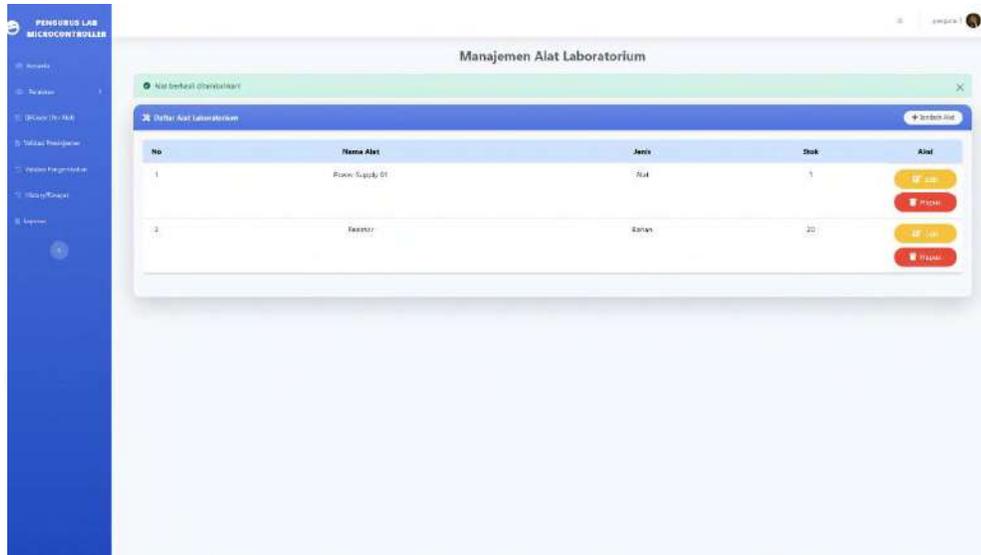
Tampilan dashboard dirancang sederhana namun fungsional, sehingga mempermudah pengurus untuk mengelola aktivitas laboratorium secara efisien.



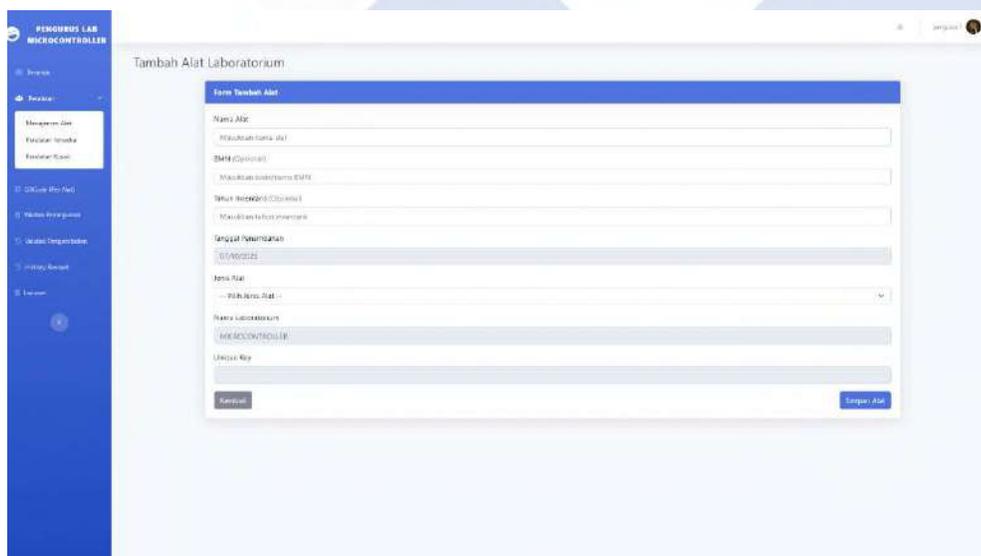
Gambar 4.3-23 Tampilan Halaman Dashboard Pengurus Laboratorium

b) Manajemen Alat

Halaman ini digunakan oleh pengurus untuk mengelola data alat laboratorium, mulai dari menambahkan, mengubah, hingga menghapus informasi alat. Setiap entri alat akan terhubung dengan data laboratorium, status ketersediaan, serta identitas QRCode.



Gambar 4.3-24 Tampilan Halaman Manajemen Alat

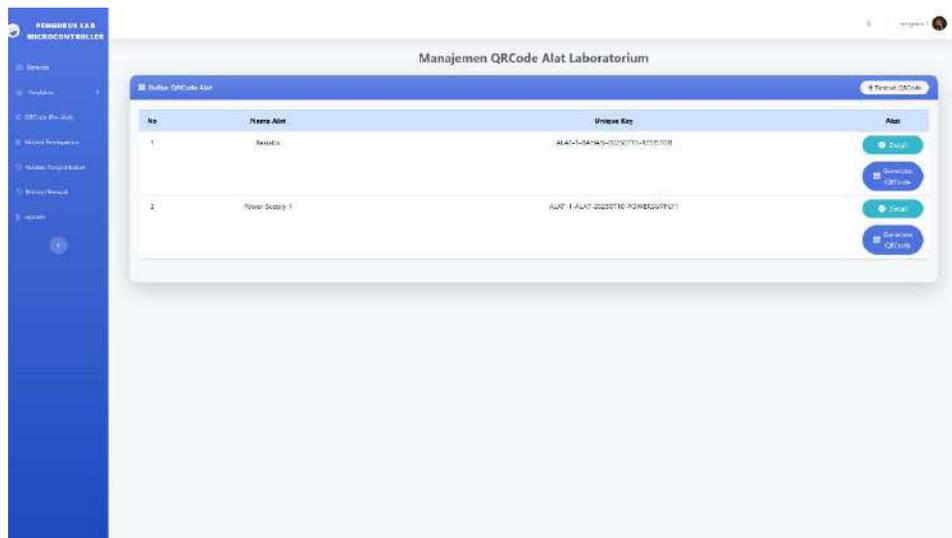


Gambar 4.3-25 Tampilan Halaman Form Manajemen Alat

c) Generate QRCode

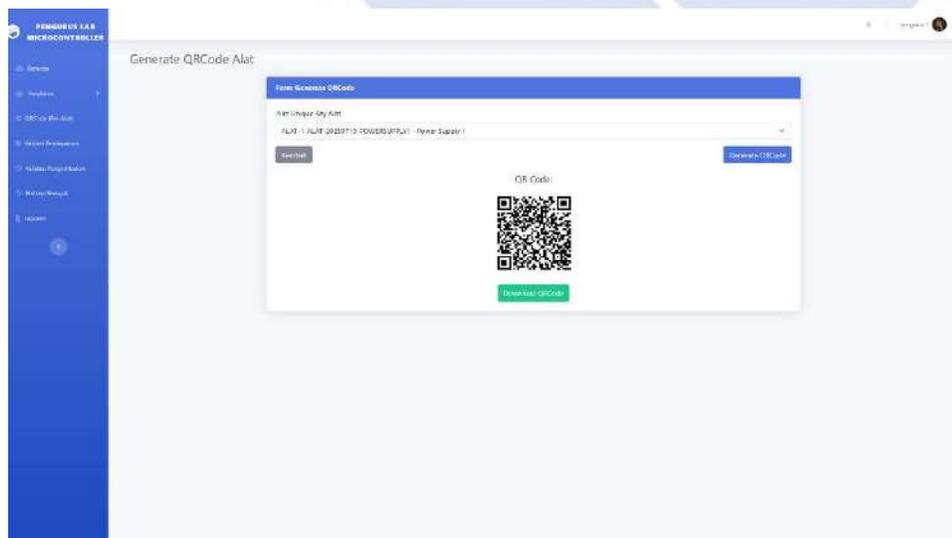
Fitur ini memungkinkan pengurus menghasilkan QRCode untuk setiap alat yang telah didaftarkan. QRCode digunakan sebagai identifikasi digital alat saat proses peminjaman maupun pengembalian.

- 1) Halaman Generate QRCode digunakan untuk memproses pembuatan QR berdasarkan data alat yang dipilih.



Gambar 4.3-26 Tampilan Halaman Generate QRCode

- 2) Halaman Generate QRCode digunakan untuk memproses pembuatan QR berdasarkan data alat yang dipilih.



Gambar 4.3-27 Tampilan Halaman QRCode

d) Peralatan Tersedia

Halaman ini menampilkan daftar alat yang dalam kondisi siap digunakan dan tidak sedang dipinjam. Fitur ini membantu pengurus memantau inventaris aktif secara real-time dan mempermudah validasi ketersediaan alat saat pengajuan peminjaman.

No.	Nama Alat	Jenis	Stok Tersedia	Unique Key	Status
1	Rakibila	Bahan	25	ALAT-1-BAHAN-2020110-RESISTOR	Tersedia
2	Kawat Tuggy 1	Alat	1	ALAT-1-ALAT-2020110-1A-POROS-DRUM-PT	Tersedia

Gambar 4.3-28 Tampilan Halaman Peralatan Tersedia

e) Peralatan Rusak

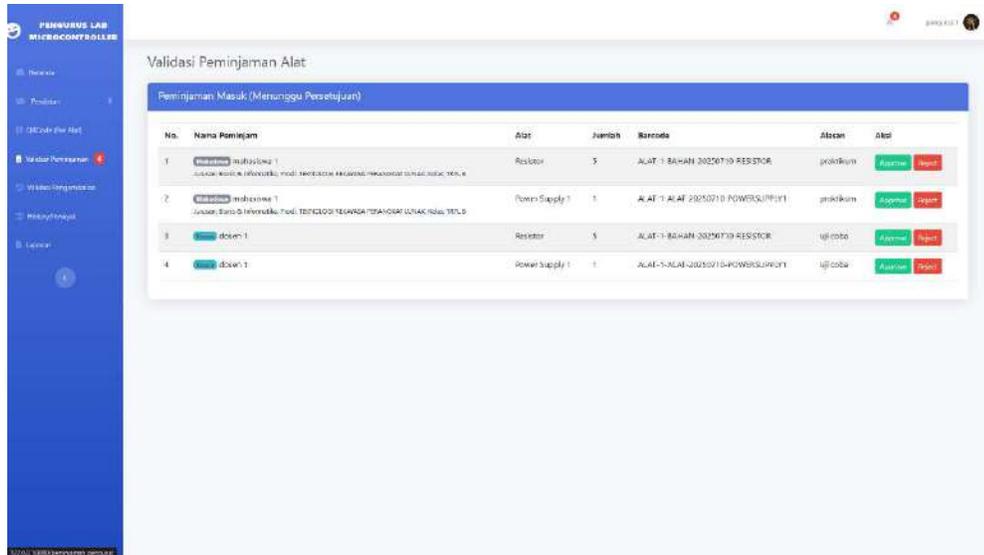
Menampilkan data alat yang dilaporkan dalam kondisi rusak. Data ini penting untuk keperluan perawatan, penghapusan, atau penggantian alat. Pengurus juga dapat menandai kategori kerusakan sebagai ringan, atau berat.

No.	Nama Alat	Jumlah Rusak (Berat)	Jumlah Rusak Berat	Jumlah Bisa Dipakai	Tanggal Rusak	Status	Aksi
1.	Rozdor	1	0	0	10-07-2025 10:50 WIB	Detail kerusakan	Tambah Perbaikan
2.	Kalioter	1	1	1	10-07-2025 10:50 WIB	Detail kerusakan	Tambah Perbaikan

Gambar 4.3-29 Tampilan Halaman Peralatan Rusak

f) Validasi Peminjaman

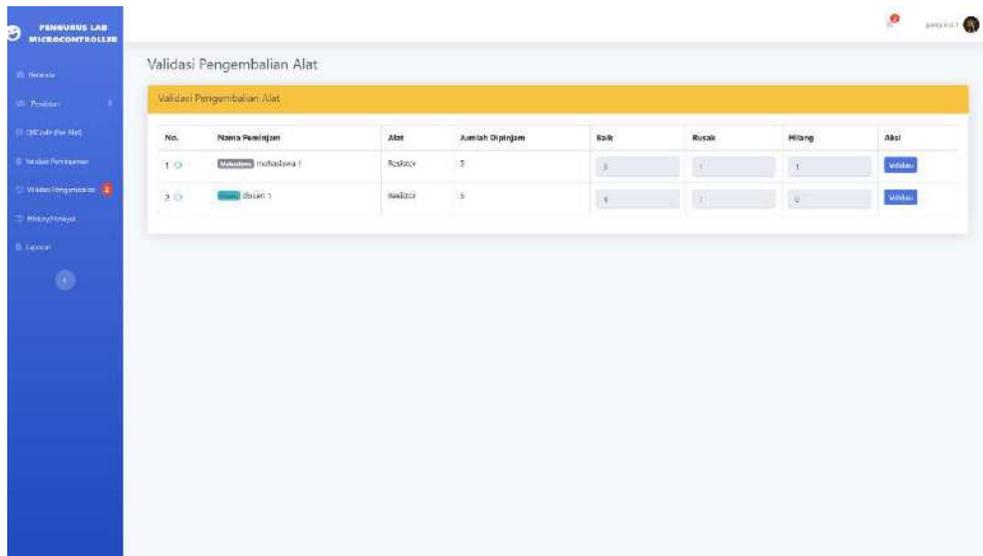
Fitur ini digunakan untuk menyetujui atau menolak pengajuan peminjaman yang masuk dari peminjam. Setiap permintaan disertai informasi detail alat, peminjam, dan tanggal yang diajukan. Sistem menyediakan tombol aksi untuk konfirmasi cepat.



Gambar 4.3-30 Tampilan Halaman Validasi Peminjaman

g) Validasi Pengembalian

Halaman ini digunakan untuk memverifikasi pengembalian alat oleh peminjam. Validasi dilakukan dengan memindai QRCode alat dan memeriksa kondisi alat saat dikembalikan. Sistem juga mencatat jika ada keterlambatan atau kerusakan.



Gambar 4.3-31 Tampilan Halaman Validasi Pengembalian

h) History/Riwayat Peminjaman

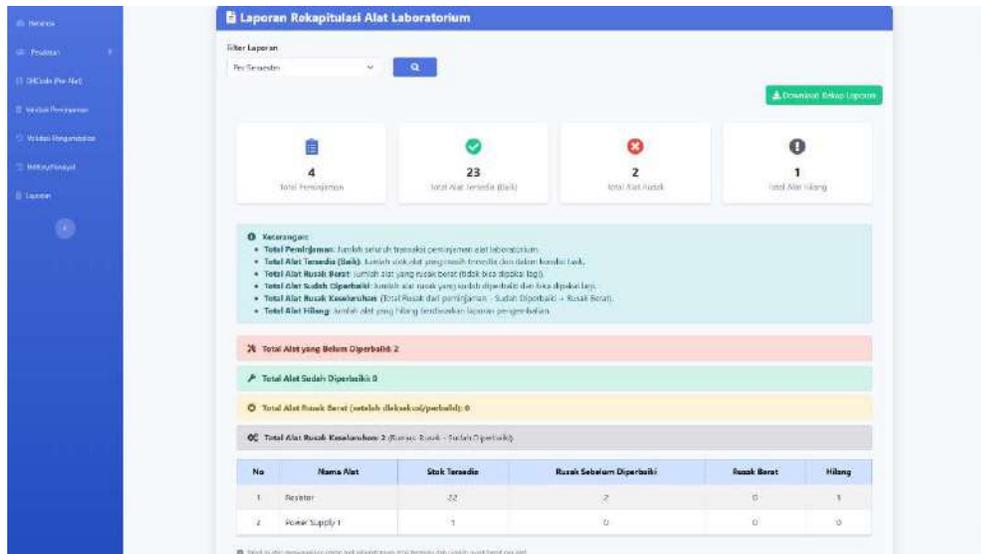
Menampilkan seluruh riwayat peminjaman alat oleh pengguna. Pengurus dapat melihat informasi lengkap berupa tanggal, alat yang dipinjam, status pengembalian, serta keterlambatan jika ada. Fitur filter tanggal juga tersedia untuk memudahkan pencarian data historis.

No.	Nama Peminjam	Alat	Jumlah	Tanggal Pinjam	Batas Waktu	Waktu Pengembalian	Status / Detail Pengembalian
1	oson 1	Power Supply 1	1	11-07-2025 08:40 WIB	11-07-2025 16:00 WIB	-	Terdapat
2	oson 1	Resistor	5	11-07-2025 08:45 WIB	11-07-2025 16:00 WIB	10-07-2025 20:02	Lampirkan Cekoran: foto 0, Ruok 1, barang 0
3	helobeswa 1	Power Supply 1	1	11-07-2025 08:44 WIB	11-07-2025 16:00 WIB	-	Terdapat
4	mdhobeswa 1	Resistor	5	11-07-2025 08:42 WIB	11-07-2025 16:00 WIB	10-07-2025 20:02	Lampirkan Cekoran: foto 0, Ruok 1, barang 1

Gambar 4.3-32 Tampilan Halaman History Pengurus Laboratorium

i) Laporan

Fitur laporan digunakan untuk menghasilkan dokumen peminjaman, pengembalian, dan data alat rusak dalam format PDF atau Excel. Laporan dapat difilter berdasarkan periode waktu tertentu, sehingga memudahkan kebutuhan dokumentasi dan rekapitulasi laboratorium.

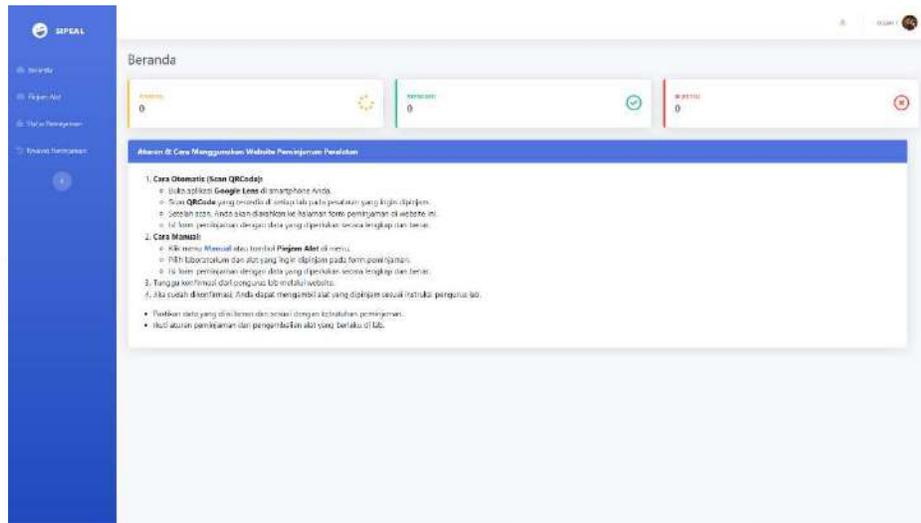


Gambar 4.3-33 Tampilan Halaman Laporan

4. Peminjam

a) Dashboard Peminjam

Halaman Dashboard Peminjam merupakan tampilan utama yang diakses oleh pengguna dengan peran sebagai peminjam (mahasiswa atau dosen) setelah berhasil melakukan *login*. Tampilan ini memberikan navigasi ringkas menuju fitur-fitur utama yang dapat digunakan oleh peminjam, seperti peminjaman alat, pengecekan status, pengembalian, dan riwayat peminjaman.

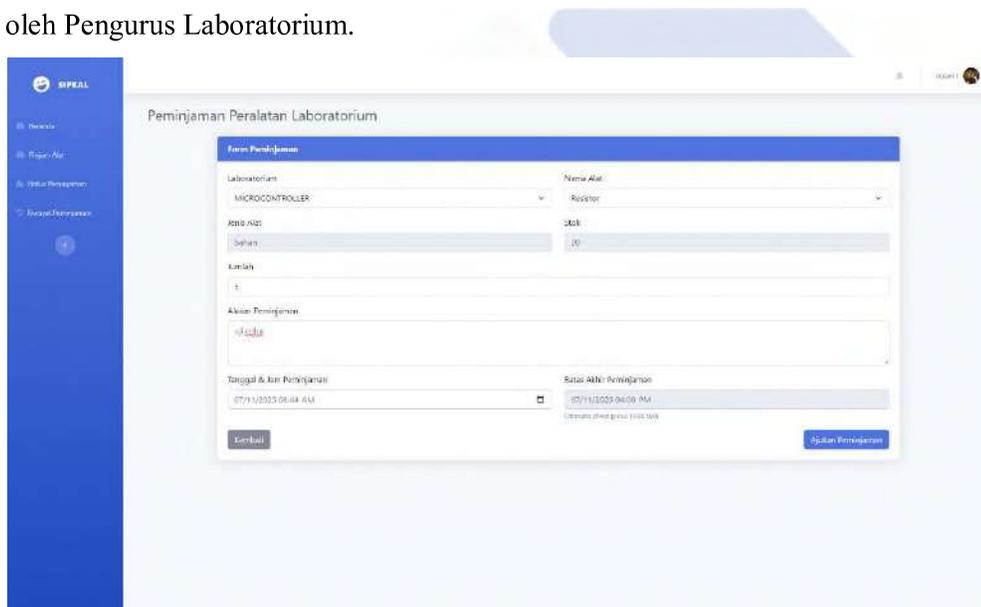


Gambar 4.3-34 Tampilan Halaman Dashboard Peminjam

b) Pinjam Alat

Fitur ini memungkinkan peminjam untuk melakukan permohonan peminjaman alat laboratorium secara langsung melalui sistem. Peminjam dapat memilih nama alat, menentukan tanggal peminjaman dan pengembalian, serta mengisi formulir yang telah terintegrasi dengan basis data inventaris alat.

Selain melalui pencarian manual, peminjam juga dapat meminjam alat dengan cara memindai (*scan*) kode *QR* yang terpasang pada alat, sehingga sistem akan secara otomatis menampilkan detail alat terkait. Setelah data permohonan dikirim, sistem akan menyimpan informasi tersebut dan menampilkannya dalam daftar validasi oleh Pengurus Laboratorium.



Gambar 4.3-35 Tampilan Halaman Pinjam Alat

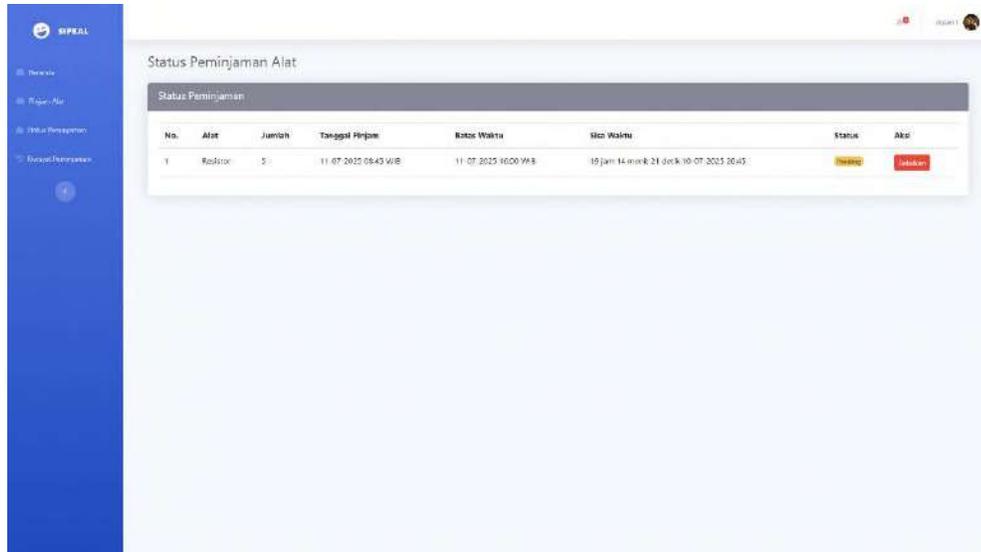


Gambar 4.3-36 Tampilan Halaman Scan QRCode

c) Status Peminjaman

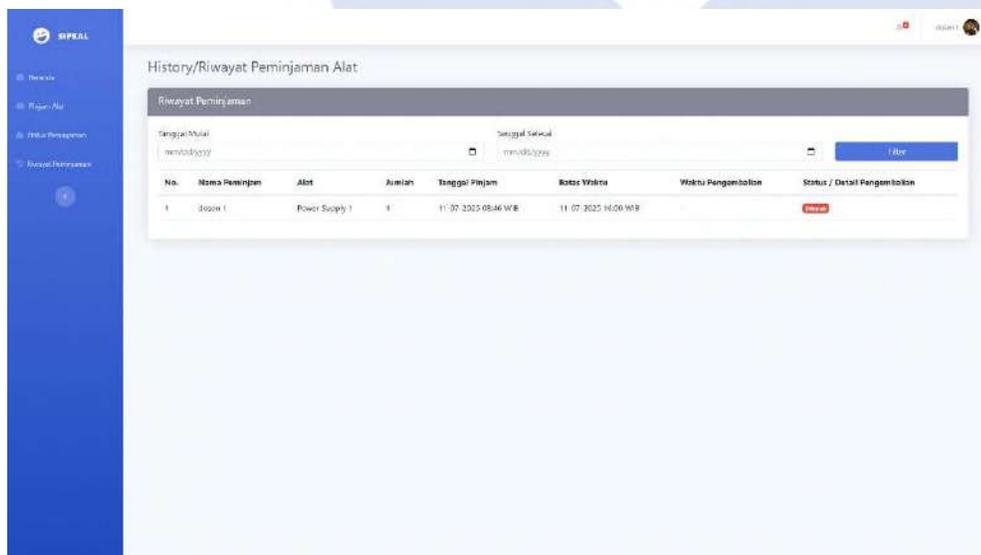
Halaman Status Peminjaman digunakan untuk melihat perkembangan permintaan peminjaman yang telah diajukan. Status yang tersedia dalam sistem terdiri dari:

- 1) Pending – menandakan bahwa permintaan masih menunggu validasi dari pengurus laboratorium.



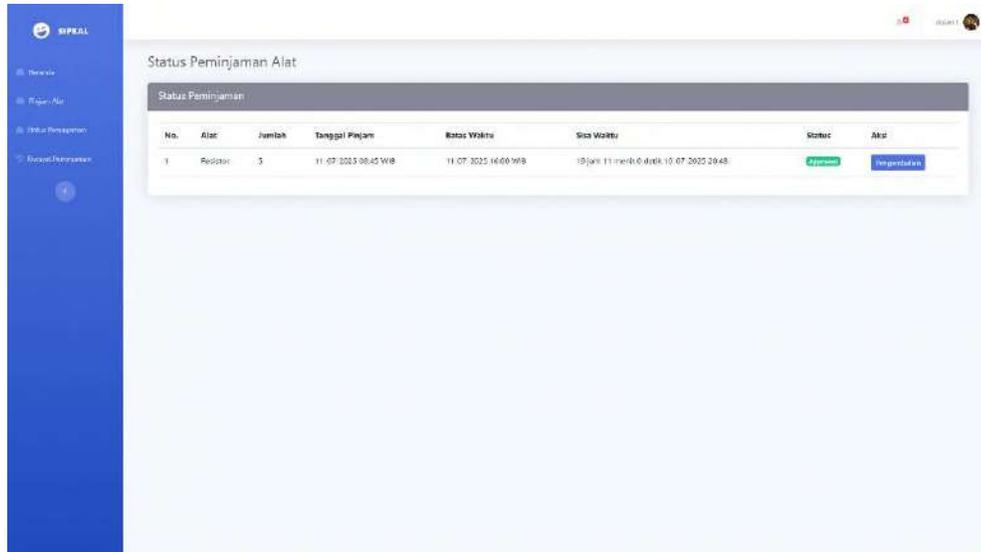
Gambar 4.3-37 Tampilan Halaman Status Peminjaman (pending)

- 2) Reject – menandakan bahwa permintaan telah ditolak.



Gambar 4.3-38 Tampilan Halaman Status Peminjaman (reject)

3) Approve – menandakan bahwa permintaan telah disetujui dan alat siap dipinjam.

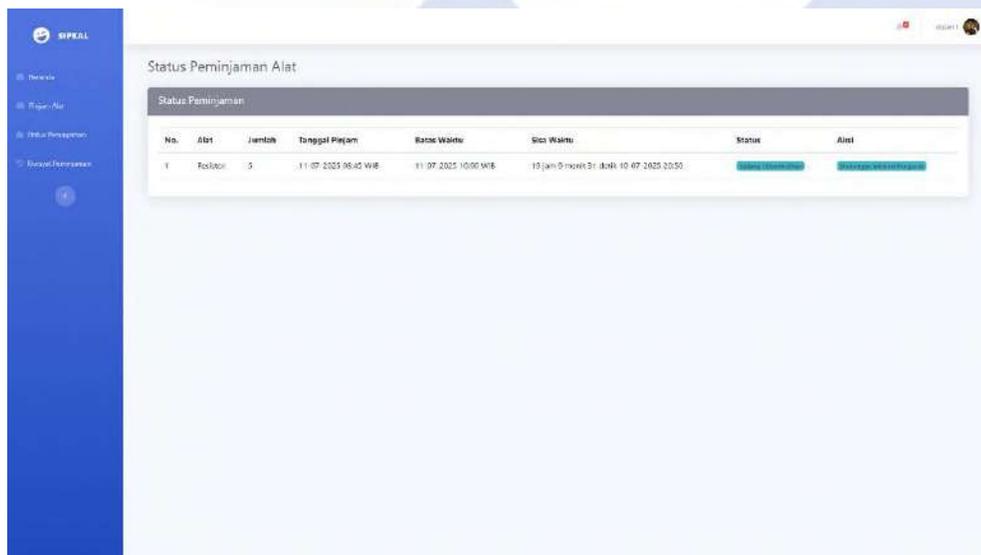


The screenshot shows the 'Status Peminjaman Alat' page in the SIPRAL application. The page features a blue sidebar with navigation options: 'Beranda', 'Peminjaman', 'Status Peminjaman', and 'Profil Peminjaman'. The main content area displays a table with the following data:

No.	Alat	Jumlah	Tanggal Pinjam	Rata-Rata Waktu	Sisa Waktu	Status	Aksi
1	Reaktor	5	11-07-2025 08:45 WIB	11-07-2025 16:00 WIB	19 jam 11 menit 6 detik 10-07-2025 20:48	Disetujui	Wajibada

Gambar 4.3-39 Tampilan Halaman Status Peminjaman (Approve)

4) Sedang dikembalikan – menandakan bahwa alat sedang dalam proses pengembalian dan validasi oleh pengurus.



The screenshot shows the 'Status Peminjaman Alat' page in the SIPRAL application. The page features a blue sidebar with navigation options: 'Beranda', 'Peminjaman', 'Status Peminjaman', and 'Profil Peminjaman'. The main content area displays a table with the following data:

No.	Alat	Jumlah	Tanggal Pinjam	Rata-Rata Waktu	Sisa Waktu	Status	Aksi
1	Reaktor	5	11-07-2025 08:45 WIB	11-07-2025 10:00 WIB	19 jam 0 menit 31 detik 10-07-2025 20:50	Sedang Dikembalikan	Wajibada

Gambar 4.3-40 Tampilan Halaman Status Peminjaman (sedang dikembalikan)

d) Pengembalian Ala

Pada halaman ini, peminjam dapat melakukan proses pengembalian alat dengan cara mengisi formulir yang tersedia. Data yang dimasukkan meliputi informasi alat, tanggal pengembalian, serta kondisi alat jika terdapat kerusakan ringan atau berat. Data pengembalian ini akan diteruskan kepada pengurus laboratorium untuk divalidasi.

The screenshot shows a web application interface for returning laboratory equipment. The page title is 'Pengembalian Peralatan Laboratorium'. The form is titled 'Form Pengembalian' and contains the following fields and options:

- Uniqur Key Alat:** ALKD-1-BUKAN-20250716-RESISTOR
- Nama Alat:** Resistor
- Aprio Alat:** Bahara
- Kantah Di Pinjam:** 1
- Kantah Diembalik:** Three radio buttons: Baik (selected), Rusak, and Hilang.
- catatan (Optional):** A text area containing the text 'Resistor yang rusak'.
- Nama Peminjam:** Siswa 1
- Tanggal Peminjaman:** 11 07 2023 06:45 AM
- Tanggal & Jam Pengembalian:** 07/10/2023 08:56:59A

At the bottom of the form, there are two buttons: 'Kembali' and 'Selesaikan Pengembalian'.

Gambar 4.3-41 Tampilan Halaman Pengembalian Alat

e) History/Riwayat Peminjaman

Halaman ini menampilkan daftar lengkap seluruh aktivitas peminjaman yang telah dilakukan oleh pengguna. Informasi yang ditampilkan meliputi nama alat, tanggal pinjam, tanggal kembali, status peminjaman, dan tindakan yang dilakukan.

No.	Nama Peminjam	Alat	Jumlah	Tanggal Pinjam	Batas Waktu	Waktu Pengembalian	Status / Detail Pengembalian
1	idosen 1	Power Supply 1	1	11-07-2025 08:40 WIB	11-07-2025 16:00 WIB		Terdapat
2	idosen 1	Resistor	5	11-07-2025 08:40 WIB	11-07-2025 16:00 WIB	10-07-2025 25:52	Batas Waktu Cetakan: 046, 0; RUMAH: 0; Hibang: 0

Gambar 4.3-42 Tampilan Halaman History Peminjaman

f) Notifikasi Email Keterlambatan\

Sistem dilengkapi dengan fitur *notifikasi otomatis* yang dikirim melalui surat elektronik (*email*) kepada pengguna apabila alat yang dipinjam tidak dikembalikan sesuai batas waktu yang telah ditentukan.



Gambar 4.3-43 Tampilan Notifikasi Email Keterlambatan

4.4. Hasil Pengujian

4.4.1. Umpan Balik Pengguna

Proses pengujian tidak hanya dilakukan untuk memastikan fungsi sistem berjalan dengan benar, tetapi juga untuk memperoleh masukan dari pengguna (*end-user*) terkait kenyamanan, kebutuhan, serta pengembangan sistem di masa mendatang.

Umpan balik dikumpulkan dari tiga jenis peran pengguna, yaitu admin, pengurus laboratorium (PLP), dan peminjam (dosen/mahasiswa).

Berikut merupakan umpan balik pengguna yang diperoleh selama pengujian sistem

Table 4.4-1 Umpan Balik Pengguna

No.	Umpan Balik Pengguna	Tanggapan
1.	Jenjang pendidikan dan program studi terpisah pada <i>input</i> data pengguna	Disarankan untuk digabung menjadi satu kolom, misalnya “D4 Teknologi Rekayasa Perangkat Lunak”
2.	Tidak tersedia <i>input</i> kode BMN dan tahun inventaris saat menambahkan alat	Perlu penambahan kolom Kode BMN dan Tahun Inventaris pada formulir <i>input</i> alat
3.	Riwayat peminjaman belum bisa difilter berdasarkan rentang waktu	Tambahkan fitur filter riwayat berdasarkan tanggal, bulan, atau tahun
4.	Tidak ada indikator sisa waktu peminjaman	Disarankan sistem menampilkan sisa waktu peminjaman agar pengguna mengetahui batas pengembalian
5.	Laporan hanya menampilkan daftar peminjaman	Laporan sebaiknya dilengkapi dengan jumlah peminjam, serta bisa difilter per bulan/semester dan menampilkan jumlah alat rusak
6.	Tanggal peminjaman tidak otomatis muncul di form pengembalian	Sebaiknya tanggal peminjaman ditampilkan otomatis sebagai referensi saat proses pengembalian
7.	Keterangan kerusakan alat hanya bersifat umum	Tambahkan klasifikasi jenis kerusakan menjadi: Ringan dan Berat

4.4.2. Pengujian Black Box

Pengujian *Black Box* dilakukan untuk mengevaluasi fungsionalitas sistem berdasarkan keluaran (*output*) yang dihasilkan dari sejumlah masukan (*input*) tertentu, tanpa memperhatikan struktur internal atau kode program sistem. Metode ini berfokus pada pengujian dari sudut pandang pengguna dengan tujuan memastikan bahwa setiap fitur berjalan sesuai dengan yang diharapkan.

Pengujian dilakukan dalam kondisi nyata (*daring*) setelah sistem berhasil diimplementasikan ke *server hosting*. Seluruh modul utama diuji menggunakan skenario yang telah dirancang sebelumnya berdasarkan kebutuhan pengguna.

Fitur-fitur utama yang di uji antara lain:

- 1) *Login* dan autentikasi peran pengguna
- 2) Pengajuan dan verifikasi peminjaman
- 3) Pengembalian alat melalui pemindaian QRCode
- 4) Pengiriman *notifikasi email* keterlambatan
- 5) Pengelolaan data alat dan pengguna
- 6) Pengunduhan laporan

Berikut adalah hasil pengujian oleh dua orang ahli sistem:

Table 4.4-2 Hasil Pengujian Black Box

No.	Fitur yang Diuji	Skor Ahli 1	Skor Ahli 2
1.	Dapat <i>login</i> sesuai hak akses (peminjam, pengurus, admin)	5	5
2.	Menolak <i>login</i> dengan username/password salah	5	5
3.	Pengurus dapat menyetujui atau menolak permintaan	5	4
4.	Pengurus dapat melihat ketersediaan atau jumlah stok alat	5	4
5.	Pengurus dapat menambah, mengedit, atau hapus data alat	5	4
6.	Menampilkan data dan menu sesuai peran	5	4
7.	Tombol <i>logout</i> berfungsi dan mengakhiri sesi pengguna	5	5
8.	Dapat menghasilkan QRCode untuk setiap alat	5	4
9.	Pesan error informatif saat terjadi kesalahan	5	4
10.	Dapat mendownload riwayat peminjaman atau laporan lab dalam bentuk dokumen (PDF/Excel)	5	5
Total Skor		50	44

Pengujian *Black Box* juga melibatkan evaluasi sistem oleh dua orang ahli yang menguji fungsionalitas dari sepuluh fitur inti. Penilaian dilakukan menggunakan skala Likert 1–5, dengan hasil sebagai berikut:

- 1) Ahli 1 memberikan skor sempurna, yaitu 50 dari 50 poin, yang menunjukkan bahwa semua fitur telah berjalan optimal dan sesuai dengan tujuan sistem.
- 2) Ahli 2 memberikan skor 44 dari 50 poin, dengan catatan pada fitur seperti validasi permintaan, pengelolaan data alat, dan tampilan antarmuka berdasarkan peran, yang dianggap masih memiliki ruang untuk peningkatan.

Rata-rata nilai evaluasi mencapai 47 poin atau 94%, yang mengindikasikan sistem telah berfungsi dengan baik dan layak digunakan.

Meskipun demikian, ditemukan satu *bug minor* pada formulir peminjaman. Jika terjadi kesalahan *input*, seperti tanggal atau alat tidak sesuai, sistem tidak secara otomatis memuat ulang daftar alat. Pengguna perlu menyegarkan halaman secara manual agar dapat memilih ulang.

Kendala ini tidak berdampak signifikan terhadap fungsi utama sistem dan dapat diperbaiki pada pengembangan selanjutnya. Secara keseluruhan, pengujian *Black Box* menunjukkan bahwa sistem telah memenuhi kriteria fungsional dan berjalan stabil di lingkungan nyata.

4.4.3. Pengujian *User Acceptance Testing* (UAT)

Pengujian *User Acceptance Testing* (UAT) dilakukan untuk menilai tingkat penerimaan pengguna terhadap system yang telah dikembangkan. Pengujian ini dilakukan oleh 30 responden yang terdiri dari berbagai peran pengguna, yaitu peminjam (20 responden), pengurus laboratorium (6 responden), dan admin(4 responden).

Parameter yang digunakan dalam pengujian ini dijelaskan pada:

1. Tabel 3.7.2: Parameter untuk *Perceived Usefulness* (PU)
2. Tabel 3.7.3: Parameter untuk *Perceived Ease of Use* (PEOU)

3. Tabel 3.7.4: Parameter untuk *General Acceptance (GA)*

Rumus Perhitungan

Untuk memperoleh skor total dan persentase penerimaan, digunakan rumus berikut:

1) Total Skor

$$\Sigma(\text{Jumlah Responden pada tiap skor} \times \text{Bobot Skor})$$

2) Skor Maksimum

$$\text{Skor Maksimum} = \text{Jumlah Responden} \times \text{Jumlah Pertanyaan} \times 5$$

3) Rata-Rata

$$\text{Rata - rata} = \frac{\text{Total Skor}}{\text{Jumlah Responden} \times \text{Jumlah Pertanyaan}}$$

4) Persentase Kepuasan

$$\text{Persentase} = \frac{\text{Total Skor}}{\text{Skor Maksimum}} \times 100\%$$

Hasil Penelitian *Perceived Usefulness (PU)*

Table 4.4-3 Hasil Penelitian *Perceived Usefulness (PU)*

Role	Jumlah Responden	Skor Maksimum	Total Skor	Rata - rata	Persentase
Peminjam	20 orang	500	440	4.4	88%
Pengurus Laboratorium	6 orang	150	135	4.5	90%
Admin	4 orang	100	89	4.45	89%
Total	30 orang	750	664	4.43	88,53%

Nilai rata-rata sebesar 4,43 dari 5 (88,5%) pada aspek *Perceived Usefulness* menunjukkan bahwa sistem dipandang efektif dan relevan dalam menunjang aktivitas peminjaman alat laboratorium oleh seluruh peran pengguna.

Hasil Penelitian *Perceived Ease of Use (PEOU)*

Table 4.4-4 Hasil Penelitian Perceived Ease of Use (PEOU)

Role	Jumlah Responden	Skor Maksimum	Total Skor	Rata - rata	Persentase
Peminjam	20 orang	500	455	4.55	91%
Pengurus Laboratorium	6 orang	150	136	4.53	90,67%
Admin	4 orang	100	90	4.5	90%
Total	30 orang	750	681	4.54	90,8%

Aspek *Perceived Ease of Use* memperoleh nilai rata-rata tertinggi sebesar 4,54 (90,8%), yang menunjukkan bahwa sistem dianggap mudah digunakan, dengan antarmuka yang intuitif dan navigasi yang jelas.

Hasil Penelitian *General Acceptance (GA)*

Table 4.4-5 Hasil Penelitian General Acceptance (GA)

Role	Jumlah Responden	Skor Maksimum	Total Skor	Rata - rata	Persentase
Peminjam	20 orang	900	800	4.55	88,89%
Pengurus Laboratorium	6 orang	270	244	4.53	90,37%
Admin	4 orang	180	160	4.5	88,89%
Total	30 orang	1350	1204	4.46	89,19%

Skor *General Acceptance* sebesar 89,2% mengindikasikan bahwa sistem telah memenuhi kriteria kelayakan implementasi secara menyeluruh, mencakup aspek validasi akses, stabilitas sistem, serta pengelolaan fungsi sesuai peran pengguna.

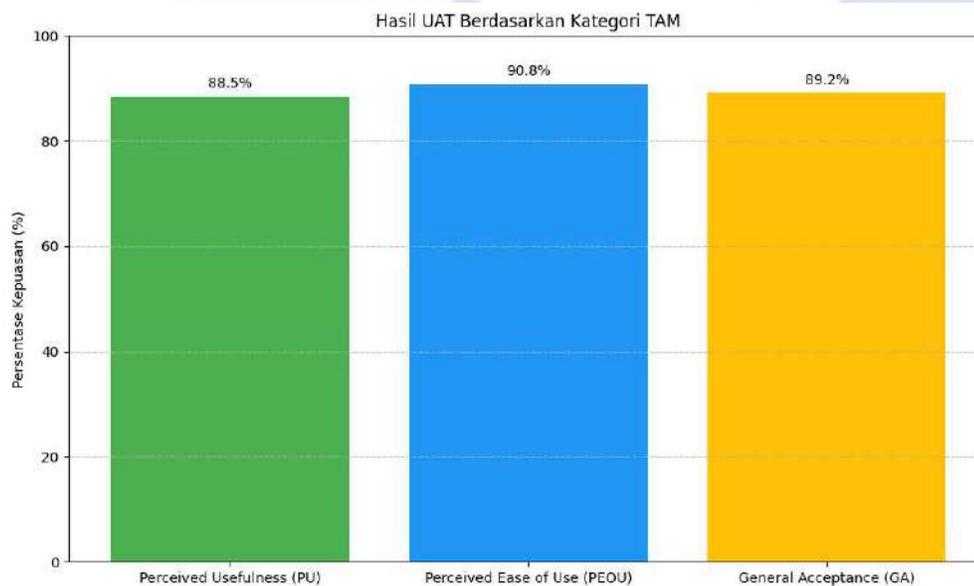
Berdasarkan pengolahan data dari seluruh responden, diperoleh hasil sebagai berikut:

Table 4.4-6 Hasil Pengujian UAT

Kategori Penilaian	Rata-rata Skor	Persentase Kepuasan
Perceived Usefulness	4,42	88,5%
Perceived Ease of Use	4,54	90,8%
General Acceptance	4,46	89,2%

Secara umum, tingkat kepuasan pengguna berada dalam rentang 88,5% hingga 90,8%, yang masuk dalam kategori “Sangat Baik”. Beberapa tanggapan juga menyoroti keunggulan pada fitur notifikasi email otomatis, tampilan antarmuka yang sederhana, serta sistem berbasis peran yang memperjelas alur proses peminjaman alat laboratorium.

Beberapa tanggapan pengguna menyatakan bahwa fitur *notifikasi email*, tampilan sederhana, dan sistem berbasis peran yang jelas sangat memudahkan proses peminjaman alat laboratorium.



Gambar 4.4-1 Hasil Pengujian UAT

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil pengembangan dan pengujian yang telah dilakukan, diperoleh kesimpulan sebagai berikut:

1. Pengembangan sistem informasi manajemen peminjaman peralatan laboratorium menggunakan metode *Rapid Application Development* (RAD) telah berhasil menyelesaikan permasalahan utama pada sistem manual sebelumnya, seperti ketidakteraturan pencatatan, keterlambatan pengembalian, dan minimnya transparansi. Perubahan yang dihasilkan adalah sistem digital yang memungkinkan pencatatan otomatis, pelacakan status alat secara real-time, dan pengingat otomatis, sehingga pengelolaan alat menjadi lebih efisien dan terstruktur.
2. Implementasi teknologi berbasis *web* dengan Laravel dan PHP menghasilkan sistem yang responsif, mudah digunakan, serta mampu memfasilitasi peminjaman dan pengembalian alat secara daring dengan dukungan QRCode dan notifikasi otomatis. Dampaknya, alur kerja laboratorium menjadi lebih efisien dan transparan.
3. Hasil pengujian *Black Box* menunjukkan bahwa seluruh fitur sistem berjalan sesuai dengan spesifikasi fungsional. Tidak ditemukan kesalahan logika dalam skenario pengujian yang dilakukan secara daring.
4. Hasil pengujian User Acceptance Testing (UAT) menunjukkan bahwa sistem memperoleh tingkat kepuasan pengguna yang sangat baik, dengan persentase kepuasan berada pada rentang 88,5% hingga 90,8%. Berdasarkan tiga kategori penilaian, *Perceived Usefulness* memperoleh kepuasan sebesar 88,5%, *Perceived Ease of Use* sebesar 90,8%, dan *General Acceptance* sebesar 89,2%. Temuan ini mencerminkan bahwa pengguna merasa sistem

mudah digunakan, tampilannya jelas, dan proses peminjaman alat menjadi lebih tertib serta terstruktur.

5. Sistem telah di-*hosting* dan dapat digunakan secara operasional, meskipun masih terdapat potensi *bug* minor akibat keterbatasan waktu pengembangan.

5.2. Saran

Adapun beberapa saran yang dapat dijadikan acuan untuk pengembangan dan peningkatan sistem di masa mendatang adalah sebagai berikut:

1. Pengembangan versi mobile (*mobile responsive* atau *native app*) agar sistem dapat diakses lebih optimal melalui perangkat gawai.
2. Peningkatan keamanan dan skalabilitas dengan implementasi autentikasi dua faktor, enkripsi data, serta pemantauan aktivitas pengguna.
3. Peningkatan layanan notifikasi, seperti integrasi dengan layanan *WhatsApp API* atau *push notification*, agar pengingat keterlambatan lebih cepat diterima.
4. Lakukan pembaruan berkala untuk memperbaiki kekurangan minor dan menyempurnakan fitur sesuai umpan balik pengguna.

DAFTAR PUSTAKA

- Aji, C. P., Havis, A., Ridwan, Mulia, P. N., & Atmaja, S. A. (2025). Pengujian Black Box pada Fitur Pendaftaran Online RSUD Kota Bandung dengan Teknik Boundary Value Analysis. *Journal on Pustaka Cendekia Informatika*, 115-124.
- Aliyah, Hartono, N., & Muin, A. A. (2025). Penggunaan User Acceptance Testing (UAT) Pada Pengujian Sistem Informasi Pengelolaan Keuangan Dan Inventaris Barang. *Jurnal Sains dan Teknologi Informasi*, 84-100.
- Andri, C., & Oktaviati, R. (2025). Implementasi Sistem Pelaporan Kerusakan Lab Komputer Berbasis QRCode dengan Pendekatan Agile pada Mahasiswa. *Jurnal Ilmu Komputer dan Sistem Informasi*, 78-91.
- Anggar, B., Anshor, A. H., & Hadikristanto, W. (2024). Implementation Web-Based Qr-Code Information System Design in Warehouse Inventory Management System Using Rapid Application Development (RAD) Method at PT Dharma Precision Parts. *Formosa Journal of Computer and Information Science (FJCIS)*, Vol.3, No.2 81-90 .
- Aryani, Y., Aqil, I., & Paramita, B. (2024). Penerapan Unified Modeling Language (UML) pada Digitalisasi Sistem Informasi Perpustakaan. *Digital Transformation Technology*, 1032-1040.
- Ashari, H., & Burhan, M. I. (2023). Monitoring Peralatan Berbasis QRCode Pada Pembelajaran di Laboratorium. *Jurnal Kajian Pendidikan dan Kepramukaan*, 51-59.
- Dewi, A. U., & Voutama, A. (2024). IMPLEMENTASI UML DALAM PERANCANGAN SISTEM INFORMASI KEUANGAN MASJID AL-AMANA BERBASIS WEB. *Jurnal Mahasiswa Teknik Informatika*, 10300-10308.
- Eriya, Agus, S., Maulana, H., & Risna, S. (2020). Sistem Manajemen Inventaris Laboratorium Otomatis Menggunakan Barcode. *JURNAL MULTINETICS*, VOL. 6 NO. 2 149-156.
- Hamdani, D., & Saptanji, R. V. (2020). Implementasi Sistem Informasi Geografis untuk Pemetaan Sebaran Jumlah Penduduk di Kota Cimahi. *Jurnal Manajemen Informatika*, 161-170.

- Hayati, R., Sa'idah, S., Zuraini, Halimah, A. H., Talindong, A., Damayanti, N., . . . Yourint, S. (2025). *MANAJEMEN SARANA DAN PRASARANA PENDIDIKAN*. Banten: PT SADA KURNIA PUSTAKA .
- Jailani, A., & Yaqin, M. A. (2024). Pengujian Aplikasi Sistem Informasi Akademik menggunakan Metode Blackbox dengan Teknik Boundary Value Analysis. *Journal Automation Computer Information System*, 60-66.
- Jaya, D. K., Susilowati, L. E., & Akhdiyati, H. R. (2023). SISTEM INFORMASI QUICK RESPONSE CODE (QR CODE) SEDERHANA UNTUK MENGOPTIMALKAN INVENTARISASI PEMINJAMAN DAN PENGGUNAAN ALAT-ALAT LABORATORIUM DI JURUSAN ILMU TANAH. *JURNAL ABDI INSANI*, 771-783.
- Kenneth C. Laudon, J. P. (2021). *Management Information Systems: Managing the Digital Firm*. New York: Pearson Education.
- Kuswandari, F., Susanti, R., & Grey, M. A. (2023). Analisis Pengelolaan Laboratorium Berbasis e-TLM QRCode dan Pengembangan Fitur Kalibrasi Secara Online pada Sistem Inventarisasi Peralatan Laboratorium. *INDONESIAN JOURNAL OF LABORATORY*, 246-252.
- Madurapperuma, I., Shafana, M., & Sabani, M. (2022). State-of-Art Frameworks for Front-end and Back-end Web Development. *Proceedings of Papers, 2nd International Conference on Science and Technology*, 62-67.
- Nikmah, R., Hamidah, & Sukanti, S. (2023). Efektivitas Sistem Peminjaman Alat Online pada Masa Pandemi Covid-19. *INDONESIAN JOURNAL OF LABORATORY*, 167-176.
- Parlika, R., Afifudin, M., Pradana, I. A., Wiratama, Y. D., & Holis., M. N. (2023). STUDI LITERATUR EFISIENSI MODEL RAPID APPLICATION DEVELOPMENT DALAM PENGEMBANGAN PERANGKAT LUNAK (2014-2022). *Jurnal Sistem dan Teknologi Informasi*, Vol. 12 No. 1 24-30.
- Purnomo, S., & Alijoyob, F. A. (2024). Sistem Peminjaman Barang Menggunakan QRCode Berbasis Aplikasi Android. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 322-328.
- Suhanda, A., Rahmi, A., Efrinalia, W., & Ariska, M. (2025). Optimalisasi Pengelolaan Smart Lab Berbasis ME-QR (Quick Response) pada Sistem Inventarisasi Peralatan dan Absensi Laboratorium Animal House. *Jurnal Pengelolaan Laboratorium Pendidikan*, 1-11.

- Sumartono, Y. V., Susanto, A., Huda, N., & Agustin, I. D. (2024). Rancang Bangun Sistem Peminjaman Alat Menggunakan Barcode Berbasis Website Di Laboratorium Jurusan Teknik Mesin Politeknik Negeri Balikpapan. *JURNAL SAINS TERAPAN*, VOL. 10 NO.2 143-150.
- Susanti, H., Asmoro, C. P., & Maemunah, I. (2023). Penerapan Sistem Peminjaman Alat Laboratorium pada SiDal (Sistem BigData Laboratorium). *Jurnal Edukasi dan Penelitian Informatika*, Vol. 9 No. 3, 747-752.



DAFTAR RIWAYAT HIDUP

1. Data Pribadi

Nama Lengkap : Putra Baruna Wijaya
Tempat & Tanggal Lahir : Sungailiat, 11 Februari 2003
Alamat Rumah : Jl. Imam Bonjol no.71B
Telp/HP : +62 852-1099-2963
Email : bwp0721@gmail.com
Jenis Kelamin : Laki-Laki
Agama : Islam



2. Riwayat Pendidikan

SD Negeri 10 Sungailiat	2009 - 2015
SMP Negeri 2 Sungailiat	2015 - 2018
SMA Negeri 1 Sungailiat	2018 - 2021

3. Pendidikan Non-Formal

Sungailiat, 18 Juli 2025

Putra Baruna Wijaya