PENGEMBANGAN APLIKASI *MOBILE* UNTUK PENGELOLAAN DAN PEMINJAMAN INVENTARIS BARANG DI KAMPUS



Disusun Oleh:

N a m a : Vahri Maulana

NIM : 21523126

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

2025

HALAMAN PENGESAHAN DOSEN PEMBIMBING

PENGEMBANGAN APLIKASI *MOBILE* UNTUK PENGELOLAAN DAN PEMINJAMAN INVENTARIS BARANG DI KAMPUS

TUGAS AKHIR

Disusun Oleh:

N a m a	: Vahri Maulana
NIM	: 21523126

Yogyakarta, 15 Juli 2025

Pembimbing,

(Irving Vitra Paputungan, S.T., M.Sc., Ph.D.)

HALAMAN PENGESAHAN DOSEN PENGUJI

PENGEMBANGAN APLIKASI *MOBILE* UNTUK PENGELOLAAN DAN PEMINJAMAN INVENTARIS BARANG DI KAMPUS

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 2 September 2025

Tim Penguji

Irving Vitra Paputungan, S.T., M.Sc., Ph.D. .....

Anggota 1

Chanifah Indah Ratnasari, S.Kom., M.Kom. .....

Anggota 2

Rahadian Kurniawan, S.Kom., M.Kom. .....

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia

(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Vahri Maulana

NIM : 21523126

Tugas akhir dengan judul:

PENGEMBANGAN APLIKASI *MOBILE* UNTUK PENGELOLAAN DAN PEMINJAMAN INVENTARIS BARANG DI KAMPUS

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 15 Juli 2025



(Vahri Maulana)

HALAMAN PERSEMBAHAN

Tugas akhir ini saya persembahkan sebagai kontribusi kecil bagi perkembangan ilmu pengetahuan, khususnya di bidang teknologi informasi. Dengan harapan, tulisan ini tidak hanya menjadi bentuk pemenuhan akademik, tetapi juga dapat memberikan manfaat bagi siapapun yang membutuhkan referensi, inspirasi, maupun solusi atas permasalahan yang serupa. Semoga hasil akhir pemikiran dan kerja keras yang tercurah dalam karya ini dapat menjadi bagian dari upaya kolektif untuk memajukan dunia pendidikan dan teknologi di masa kini maupun masa depan.

HALAMAN MOTO

فَإِذَا فَرَغْتَ فَانْصَبْ

Apabila engkau telah selesai (dengan suatu kebajikan), teruslah bekerja keras (untuk kebajikan yang lain).

KATA PENGANTAR

Alhamdulillah, segala puji dan syukur penulis panjatkan ke hadirat Allah Swt. atas rahmat dan hidayah-Nya yang telah memungkinkan penulis menyelesaikan penulis menyelesaikan tugas akhir ini yang berjudul "Pengembangan Aplikasi *Mobile* Untuk Pengelolaan dan Peminjaman Inventaris Barang di Kampus". Tugas akhir ini merupakan salah satu syarat untuk menyelesaikan program studi di Prodi Informatika Universitas Islam Indonesia.

Penulis ini menyadari bahwa dalam proses penyusunan laporan ini, terdapat berbagai tantangan dan kesulitan yang dihadapi. Namun, berkat dukungan dan bantuan dari berbagai pihak, penulis dapat menyelesaikan tugas akhir ini dengan baik. Oleh karena itu, penulis ingin menyampaikan terima kasih yang tulus kepada:

1. Allah Swt. yang telah memberikan kekuatan, kesehatan, dan petunjuk-Nya selama proses penyusunan tugas akhir ini.
2. Kedua orang tua tercinta, yang selalu memberikan doa, dukungan, dan semangat yang tiada henti dalam setiap langkah penulis.
3. Kedua kakak saya tercinta, yang senantiasa memberikan dukungan moral, semangat, serta menjadi teladan dalam menjalani proses kehidupan dan pendidikan.
4. Prof. Fathul Wahid, S.T., M.Sc., Ph.D., selaku Rektor Universitas Islam Indonesia.
5. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Jurusan Informatika Universitas Islam Indonesia.
6. Bapak Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D., selaku Ketua Program Studi Informatika Program Sarjana Fakultas Teknologi Industri Universitas Islam Indonesia
7. Bapak Irving Vitra Paputungan, S.T., M.Sc., Ph.D., selaku dosen pembimbing yang telah memberikan bimbingan, arahan, dan saran yang sangat berharga dalam penyelesaian tugas akhir ini.
8. Bapak dan Ibu Dosen Informatika yang sudah mengajarkan ilmu dan memberikan bimbingan akademik selama masa perkuliahan.
9. Teman-teman terdekat yang selalu memberikan semangat: Wisnu, Angga, Octa, Fikri, Rafli, Agung, Maul, Ridzky, Luqman, Fadhlan, Wawan, Ahsan, Bima, dan Wildan.
10. Teman-teman seperjuangan lainnya yang selalu memberikan motivasi dan dukungan sepanjang proses penulisan tugas akhir.

Penulis berharap tugas akhir ini dapat memberikan manfaat serta menjadi kontribusi positif bagi perkembangan ilmu pengetahuan, khususnya di bidang teknologi informasi. Segala kritik dan saran yang membangun akan diterima dengan terbuka. Terima kasih disampaikan kepada semua pihak yang telah memberikan dukungan selama proses penyelesaian tugas akhir ini.

Yogyakarta, 15 Juli 2025



(Vahri Maulana)

SARI

Pengelolaan inventaris barang di lingkungan perguruan tinggi masih menghadapi tantangan signifikan akibat ketergantungan pada metode manual dan sistem berbasis *website*/*desktop* yang terbatas. Praktik pencatatan konvensional rentan menyebabkan kesalahan data, duplikasi data, dan ketidaksesuaian antara dokumen inventaris dengan kondisi aktual barang. Sistem digital yang ada pun seringkali gagal menyediakan akses *real-time* dan mekanisme terpadu untuk proses peminjaman, menghambat efisiensi operasional kampus.

Penelitian ini bertujuan untuk mengembangkan aplikasi *mobile* terintegrasi sebagai solusi atas permasalahan tersebut dengan menerapkan metode *waterfall.* Tahapan pengembangan dimulai dari identifikasi masalah melalui wawancara semi-terstruktur, dilanjutkan dengan perancangan sistem berbasis pemodelan diagram alur dan struktur *database.* Proses implementasi difokuskan pada pembangunan antarmuka intuitif yang mengakomodasi dua peran pengguna (admin dan peminjam) dalam satu platform terpadu menggunakan Flutter untuk antarmuka pengguna dan Firebase sebagai *platform backend.*

Pengujian *usability* dilakukan dengan pendekatan kuosioner berbasis *Positive* SUS (*System Usability Scale* versi positif). Hasil pengujian menunjukkan bahwa sistem yang dikembangkan memiliki kualitas *usability* yang baik, dengan skor rata-rata 85,83 untuk peminjam (kategori *Excellent*) dan 84,0 untuk admin (kategori *Excellent*). Analisis kelebihan dan kekurangan sistem mengindikasikan meskipun sistem ini efektif dalam meningkatkan efisiensi operasional dan transparansi pengelolaan aset, beberapa aspek seperti sistem notifikasi dan fitur manajemen masih memerlukan penyempurnaan.

Sistem yang dihasilkan diharapkan dapat menjadi alternatif praktis bagi institusi pendidikan untuk meningkatkan transparansi pengelolaan aset. Melalui integrasi fungsi inventaris dan peminjaman dalam ekosistem digital, aplikasi ini bertujuan mengurangi beban kerja administratif sekaligus menyediakan aksesibilitas layanan bagi seluruh pemangku kepentingan. Implementasi solusi ini di lingkungan kampus diharapkan dapat mendorong transformasi digital melalui peningkatan akurasi data, optimasi waktu operasional, dan peningkatan kepuasan bagi semua pemangku kepentingan.

Kata kunci: Aplikasi Mobile, Inventaris Kampus, Sistem Peminjaman, Flutter, Firebase, Metode Waterfall, dan Efisiensi Operasional.

GLOSARIUM

Backend	Komponen server yang menangani logika bisnis, basis data, autentikasi, dan integrasi layanan.
Black-Box Testing	Pengujian fungsionalitas sistem tanpa analisis kode internal.
Cache Network Image	Pustaka Flutter untuk menampilkan gambar dengan optimalisasi caching.
CRUD	Operasi dasar manajemen data (*create*, *read*, *update*, *delete*).
Database Realtime	Penyimpanan data dengan pembaruan instan ke semua perangkat.
Emulator/Simulator	*Software* untuk simulasi operasi aplikasi di lingkungan perangkat virtual.
Firebase	*Platform backend cloud* untuk database real-time dan autentikasi.
Flutter	*Framework open-source* Google untuk pengembangan aplikasi *mobile multiplatform*.
Frontend	Komponen aplikasi yang berinteraksi langsung dengan pengguna.
Git	Sistem kontrol versi untuk melacak perubahan kode.
HTTP *Request*	Protokol komunikasi untuk pertukaran data aplikasi-server.
Image Picker	Fitur untuk mengambil gambar dari kamera/galeri perangkat.
Intl Package	Pustaka format teks, tanggal, dan angka multibahasa.
Metode *Waterfall*	Pendekatan pengembangan sekuensial linear.
Positive SUS	Kuesioner pengukuran kuantitatif tingkat kegunaan sistem.
Use Case Diagram	Diagram UML untuk memetakan interaksi pengguna – sistem.
Usability Testing	Evaluasi kemudahan penggunaan sistem melalui pengukuran kuantitatif.
Visual Studio Code	IDE untuk penulisan kode program.
Wireframe	*Blueprint* statis tata letak antarmuka sebelum implementasi.

DAFTAR ISI

HALAMAN JUDUL i
HALAMAN PENGESAHAN DOSEN PEMBIMBING ii
HALAMAN PENGESAHAN DOSEN PENGUJI iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR iv
HALAMAN PERSEMBAHAN v
HALAMAN MOTO vi
KATA PENGANTAR vii
SARI ix
GLOSARIUM x
DAFTAR ISI xi
DAFTAR TABEL xiv
DAFTAR GAMBAR xv
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 4
1.3 Pertanyaan Penelitian 4
1.4 Batasan Masalah 5
1.5 Tujuan Penelitian 5
1.6 Manfaat Penelitian 6
1.7 Metodologi Penelitian 6
1.8 Sistematika Penulisan 7
BAB II KAJIAN PUSTAKA 9
2.1 Sistem Pengelolaan Inventaris 9
2.2 Penelitian Terdahulu 10
2.3 Kategorisasi Metodologi Pengembangan Sistem 11
2.4 Pembahasan Metodologi 19
2.5 *System Usability Scale* (SUS) dan Variasinya 21
BAB III METODOLOGI PENELITIAN 22
3.1 Pemilihan Metodologi 22
3.2 Diagram Pengembangan Sistem 23
3.3 Analisis Kebutuhan Sistem 24
3.4 Desain dan Perancangan Sistem 34

3.4.1 Gambaran Umum Sistem 34
3.4.2 *Use Case Diagram* 35
3.4.3 *Activity Diagram* 39
3.4.4 Relasi Tabel *Database* 47
3.4.5 Proses Bisnis Sistem 48
3.4.6 Perancangan Desain Antarmuka 50
3.5 Metode Pengujian dan Evaluasi 63
3.5.1 Pengujian Fungsional (*Black-Box Testing*) 63
3.5.2 Pengujian Usability (*System Usability Testing* Versi Positif) 63
BAB IV HASIL DAN PEMBAHASAN 64
4.1 Implementasi Antarmuka Pengguna 64
4.1.1 Tampilan *Splash Screen* Aplikasi 64
4.1.2 Tampilan Halaman Registrasi Pengguna 65
4.1.3 Tampilan Halaman *Login* Pengguna 66
4.1.4 Tampilan Halaman *Home* Peminjam 67
4.1.5 Tampilan Halaman Detail Barang 68
4.1.6 Tampilan Halaman *Cart* Peminjam 69
4.1.7 Tampilan Halaman *Checkout* Peminjam 70
4.1.8 Tampilan Halaman *Order Activity* Peminjam 71
4.1.9 Tampilan Halaman *Notification* Peminjam 74
4.1.10 Tampilan Halaman *Manage Items* Admin 76
4.1.11 Tampilan Halaman *Add Item* Admin 77
4.1.12 Tampilan Halaman Edit *Item* Admin 78
4.1.13 Tampilan Halaman *Loans Requested* Admin 79
4.1.14 Tampilan Halaman *Loans Activity* Admin 81
4.1.15 Tampilan Halaman *Dashboard* Admin 86
4.2 Pengujian Sistem 87
4.2.1 *Black Box Testing* 87
4.2.2 *Usability Testing* 93
4.2.3 Analisis Kelebihan dan Kekurangan Sistem 101
4.3 Kesesuaian Sistem dengan Aturan Peminjaman di UII 103
BAB V KESIMPULAN DAN SARAN 104
5.1 Kesimpulan 104
5.2 Saran 105

DAFTAR PUSTAKA 107
LAMPIRAN 111

DAFTAR TABEL

Tabel 2.1 Ringkasan Penelitian Terdahulu Sistem Inventaris dan Peminjaman Barang..... 10
Tabel 2.2 Perbandingan Metodologi Pengembangan Sistem Inventaris Barang Berdasarkan Penelitian Terdahulu..... 12
Tabel 2.3 Perbandingan Kelebihan dan Kekurangan Metodologi Pengembangan Sistem 16
Tabel 3.1 Demografi Responden Wawancara..... 25
Tabel 3.2 Daftar Pertanyaan Wawancara..... 26
Tabel 3.3 Identifikasi Masalah dari Hasil Wawancara..... 26
Tabel 3.4 *Use Case* Aktor 37
Tabel 4.1 Hasil Pengujian *Black-Box Register* 87
Tabel 4.2 Hasil Pengujian *Black-Box Login*..... 88
Tabel 4.3 Hasil Pengujian *Black-Box* Peminjaman Barang 89
Tabel 4.4 Hasil Pengujian *Black-Box* Pembatalan Pengajuan..... 90
Tabel 4.5 Hasil Pengujian *Black-Box* Admin Konfirmasi Pengajuan..... 90
Tabel 4.6 Hasil Pengujian *Black-Box* Pengembalian Barang..... 91
Tabel 4.7 Hasil Pengujian *Black-Box* CRUD Barang..... 91
Tabel 4.8 Hasil Pengujian *Black-Box* Pengiriman Notifikasi 92
Tabel 4.9 Pertanyaan *System Usability Scale* Untuk Peminjam 93
Tabel 4.10 Pertanyaan *System Usability Scale* Untuk Admin 94
Tabel 4.11 Demografi Responden Pengujian..... 95
Tabel 4.12 Hasil Pengujian Untuk Peminjam 99
Tabel 4.13 Hasil Pengujian Untuk Admin 99
Tabel 4.14 Hasil Perhitungan Skor SUS Untuk Peminjam..... 100
Tabel 4.15 Hasil Perhitungan Skor SUS Untuk Admin..... 100

DAFTAR GAMBAR

Gambar 3.1 Diagram Alur Pengembangan Sistem 23
Gambar 3.2 Alur Gambaran Umum Sistem 35
Gambar 3.3 *Use Case Diagram* 36
Gambar 3.4 *Activity Diagram* Proses Peminjaman Barang 39
Gambar 3.5 *Activity Diagram* Proses Pembatalan 40
Gambar 3.6 *Activity Diagram* Proses Pengembalian Barang 41
Gambar 3.7 *Activity Diagram* Proses Penambahan Barang 42
Gambar 3.8 *Activity Diagram* Proses Pengeditan Barang 43
Gambar 3.9 *Activity Diagram* Proses Hapus Barang 44
Gambar 3.10 *Activity Diagram* Proses Lihat Notifikasi 45
Gambar 3.11 *Activity Diagram* Proses Lihat Aktivitas Peminjaman 46
Gambar 3.12 Relasi Tabel *Database* 47
Gambar 3.13 *Business Process Model and Notation* (BPMN) 49
Gambar 3.14 Desain Antarmuka Layar Registrasi 50
Gambar 3.15 Desain Antarmuka Halaman *Login* 51
Gambar 3.16 Desain Antarmuka Halaman *Home* Peminjam 52
Gambar 3.17 Desain Antarmuka Halaman Detail Barang 53
Gambar 3.18 Desain Antarmuka Halaman *Cart* Peminjam 54
Gambar 3.19 Desain Antarmuka Halaman *Checkout* Peminjam 55
Gambar 3.20 Desain Antarmuka Halaman *Order Activity* Peminjam 56
Gambar 3.21 Desain Antarmuka Halaman *Notification* Peminjam 57
Gambar 3.22 Desain Antarmuka Halaman *Manage Items* Admin 58
Gambar 3.23 Desain Antarmuka Halaman *Add Item* 59
Gambar 3.24 Desain Antarmuka Halaman *Loans Requested* Admin 60
Gambar 3.25 Desain Antarmuka Halaman *Loans Activity* Admin 61
Gambar 3.26 Desain Antarmuka Halaman *Dashboard* Admin 62
Gambar 4.1 Tampilan Halaman *Splash Screen* Aplikasi 64
Gambar 4.2 Tampilan Halaman Registrasi Pengguna 65
Gambar 4.3 Tampilan Halaman *Login* Pengguna 66
Gambar 4.4 Tampilan Halaman Utama *Home* Pengguna 67
Gambar 4.5 Tampilan Halaman Detail Barang 68
Gambar 4.6 Tampilan Halaman *Cart* Peminjam 69

Gambar 4.7 Tampilan Halaman *Checkout* Peminjam 70
Gambar 4.8 Notifikasi Sistem Pasca-*Checkout* 70
Gambar 4.9 Tampilan Halaman Utama *Order Activity* Peminjam 71
Gambar 4.10 Tampilan *Card* Transaksi Dengan Status *Waiting* 71
Gambar 4.11 Tampilan *Card* Transaksi Dengan Status *Ongoing* 72
Gambar 4.12 Tampilan *Card* Transaksi Dengan Status *Completed* 72
Gambar 4.13 Tampilan *Card* Transaksi Dengan Status *Rejected* 73
Gambar 4.14 Tampilan *Card* Transaksi Dengan Status *Late* 73
Gambar 4.15 Tampilan Halaman Utama *Notification* Peminjam 74
Gambar 4.16 Tampilan *Card* Notifikasi *Order Accepted* 74
Gambar 4.17 Tampilan *Card* Notifikasi *Order Rejected* 75
Gambar 4.18 Tampilan *Card* Notifikasi *Return Reminder* 75
Gambar 4.19 Tampilan *Card* Notifikasi *Late Return* 75
Gambar 4.20 Tampilan Halaman *Manage Items* Admin 76
Gambar 4.21 Tampilan Halaman *Add Item* Admin 77
Gambar 4.22 Tampilan Halaman Edit *Item* Admin 78
Gambar 4.23 Tampilan Halaman *Loans Requested* Admin 79
Gambar 4.24 Tampilan Dialog Konfirmasi *Reject* 80
Gambar 4.25 Tampilan Dialog Konfirmasi *Approve* 80
Gambar 4.26 Tampilan Dialog Peringatan Stok Habis 80
Gambar 4.27 Tampilan Utama *Halaman Loans Activity* Admin 81
Gambar 4.28 Tampilan Halaman *Tab Ongoing Loans Activity* Admin 82
Gambar 4.29 Tampilan Halaman *Tab Completed Loans Activity* Admin 82
Gambar 4.30 Tampilan Halaman *Tab Rejected Loans Activity* Admin 83
Gambar 4.31 Tampilan Halaman *Tab Late Loans Activity* Admin 84
Gambar 4.32 Tampilan Halaman *Tab Nearing Deadline Loans Activity* Admin 85
Gambar 4.33 Tampilan *Dashboard* Admin 86

BAB I
PENDAHULUAN

1.1 Latar Belakang

Di era digital yang terus berkembang pesat ini, teknologi informasi dan sistem informasi telah menjadi fondasi utama yang mentransformasi berbagai sektor kehidupan, mulai dari dunia bisnis hingga institusi pendidikan. Transformasi ini mendorong setiap organisasi untuk mengadopsi solusi digital guna meningkatkan efisiensi dan efektivitas operasional mereka. Dalam konteks ini, pengembangan aplikasi *mobile* muncul sebagai salah satu inovasi disruptif paling signifikan, secara fundamental mengubah cara individu berinteraksi dengan perangkat lunak dan layanan (Patel, 2024) Seiring dengan dominasi perangkat *mobile* dalam kehidupan sehari-hari, aplikasi *mobile* kini menjadi standar baru untuk penyediaan layanan yang ingin menjangkau penggunanya secara cepat dan efektif, mengingat hampir setiap orang memiliki akses ke perangkat *mobile* (Patel, 2024)

Paralel dengan perkembangan teknologi ini, pengelolaan aset merupakan fungsi vital bagi setiap organisasi, termasuk institusi pendidikan tinggi, guna memastikan keberlanjutan operasional dan pemanfaatan sumber daya yang optimal. Aset, sebagai sumber daya ekonomi yang dikendalikan atau dimiliki, memerlukan proses pencatatan, pengumpulan data, pelaporan, dan dokumentasi yang akurat serta efisien (Setiawan dkk., 2019). Dalam konteks pengelolaan inventaris barang di kampus, aplikasi *mobile* menawarkan keunggulan dalam hal aksesibilitas, kemudahan penggunaan, dan kemampuan untuk menyediakan informasi secara *real-time*, yang seringkali tidak seoptimal pada sistem berbasis *desktop* atau *website* tradisional (Patel, 2024). Potensi ini membuka peluang besar untuk pengembangan aplikasi *mobile* yang secara spesifik dirancang untuk pengelolaan dan peminjaman inventaris barang di kampus, sehingga dapat mendukung proses tersebut secara efisien dan akurat.

Pemilihan *platform mobile* dalam penelitian ini didasarkan pada temuan empiris dari studi perbandingan pengalaman pengguna antara aplikasi *mobile* dan *website* yang menunjukkan dominasi penggunaan perangkat bergerak di Indonesia serta manfaat fungsional yang relevan dengan proses peminjaman. Mayoritas pengguna lebih aktif melalui perangkat seluler, dan dari hasil diskusi kelompok terarah para pengguna menilai notifikasi pada aplikasi *mobile* memudahkan interaksi karena memberikan pemberitahuan persetujuan maupun pengingat secara cepat (Prasetyaningsih & Muchtar, 2023). Dalam konteks inventaris kampus yang menuntut proses pengajuan, persetujuan, hingga pengingat pengembalian, kemampuan

push notification dan akses kapan saja/di mana saja menjadi krusial. Dari sisi performa, penelitian perbandingan PWA dan *Mobile Web* menunjukkan bahwa pada sumber daya halaman yang besar dan skenario akses berulang, arsitektur yang dioptimalkan untuk perangkat bergerak dapat memberikan waktu tampil halaman yang lebih baik, sementara dukungan *caching* mendorong ketersediaan informasi yang lebih responsif (Ridho dkk., 2018). Dengan landasan temuan-temuan tersebut, *platform mobile* dipilih agar sistem mampu menyediakan pengalaman yang mudah diakses, responsif, dan didukung notifikasi *real-time* yang selaras dengan kebutuhan operasional peminjaman inventaris di lingkungan kampus.

Secara ideal, pengelolaan inventaris barang di lingkungan kampus seharusnya berjalan secara efisien, transparan, dan mudah diakses oleh seluruh pihak yang berkepentingan, mulai dari staf pengelola hingga mahasiswa yang membutuhkan akses terhadap aset kampus. Sistem yang ideal akan memungkinkan pencatatan, pelacakan, dan pelaporan inventaris yang akurat, serta memfasilitasi proses peminjaman dan pengembalian barang dengan cepat dan tanpa hambatan. Hal ini sejalan dengan prinsip bahwa pengelolaan aset yang baik adalah kunci untuk menjaga efisiensi dan kelancaran operasional sehari-hari dalam sebuah institusi (Setiawan dkk., 2019).

Namun realitas di banyak institusi pendidikan menunjukkan bahwa proses inventarisasi aset masih menghadapi banyak berbagai kendala. Sebagai contoh, di kampus STTI NIIT I-TECH, pengelolaan inventaris masih dilakukan secara manual, dengan mengandalkan pencatatan dalam buku sebagai media utama (Aulia & Setiyani, 2024). Pendekatan manual ini sangat rentan terhadap risiko kesalahan manusia, seperti duplikasi data (redundansi), kehilangan data, ketidakakuratan informasi, dan kesulitan mencari data historis (Aulia & Setiyani, 2024). Kondisi serupa juga ditemukan di Universitas Malikussaleh, di mana pendataan masih bersifat manual dan tidak efisien, menyebabkan berkas yang disusun cenderung lama dan tidak efektif dikelola secara berkala (Retno dkk., 2023). Demikian pula di Universitas Widya Kartika, metode konvensional seperti pencatatan manual dengan kertas sering kali rentan terhadap kesalahan, tidak efisien, dan sulit dalam melacak perubahan inventaris (Hizkia dkk., 2024). Akibatnya, pemantauan pergerakan barang menjadi sulit, proses pelaporan inventaris menjadi lambat, dan potensi kesalahan dalam pencatatan meningkat secara signifikan. Kondisi ini tidak hanya menghambat efisiensi operasional, tetapi juga dapat menimbulkan kerugian dan ketidakpuasan di kalangan pengguna aset kampus.

Menyadari tantangan yang ada dalam pengelolaan inventaris manual, berbagai upaya telah dilakukan untuk mengembangkan sistem informasi inventaris berbasis teknologi guna

mengatasi keterbatasan tersebut. Aplikasi inventaris *website* telah dikembangkan untuk Kampus STTI NIIT I-TECH, mencakup fitur-fitur penting seperti pengelolaan gudang, ruangan, laboratorium, pencatatan data barang, serta pembaruan status dan jumlah barang (Aulia & Setiyani, 2024). Sistem Informasi Manajemen Inventaris Barang (SIMB) berbasis *website* juga dirancang untuk Kampus Widya Kartika dengan tujuan meningkatkan efisiensi, akurasi, dan keterjangkauan dalam pengelolaan inventaris (Hizkia dkk., 2024). Sementara itu, pengembangan sistem pendataan inventaris barang berbasis *desktop* untuk Program Studi Teknik Informatika Universitas Malikussaleh difokuskan pada mengatasi masalah pendataan manual (Retno dkk., 2023). Selain itu, sistem informasi inventaris berbasis *website* yang dirancang khusus untuk peminjaman dan pengembalian alat di laboratorium telah dikembangkan, memfasilitasi pengelolaan alat dan bahan habis pakai di lingkungan akademik (Wibowo dkk., 2021).

Meskipun solusi-solusi tersebut memberikan kontribusi signifikan dalam digitalisasi pengelolaan inventaris, masih terdapat kesenjangan dalam penyediaan solusi terintegrasi berbentuk aplikasi *mobile* yang secara spesifik dirancang untuk memfasilitasi proses pengelolaan dan peminjaman inventaris barang di lingkungan kampus secara komprehensif. Penelitian-penelitian sebelumnya cenderung berfokus pada aspek pengelolaan inventaris melalui platform *website*/*desktop* atau fungsi peminjaman di lingkungan terbatas, namun belum secara optimal mengintegrasikan kedua fungsi krusial ini dalam satu *platform mobile* yang terpadu. Keterbatasan aksesibilitas *real-time* pada solusi berbasis website/*desktop* serta fokus yang terbatas pada aset laboratorium atau inventarisasi umum tanpa mekanisme peminjaman menunjukkan perlunya pengembangan aplikasi *mobile* komprehensif.

Oleh karena itu, penelitian ini bertujuan mengisi kesenjangan tersebut dengan mengembangkan aplikasi *mobile* yang menyatukan pengelolaan inventaris efisien dan proses peminjaman terpadu mulai dari pengajuan permintaan, persetujuan, hingga pengembalian dalam satu platform yang dapat diakses kapan saja dan di mana saja oleh seluruh mahasiswa dengan antarmuka yang intuitif untuk meningkatkan pengalaman pengguna. Aplikasi ini diharapkan dapat membantu kampus dalam mempercepat proses peminjaman barang sehari-hari, mengurangi kesalahan pencatatan manual, dan memberikan akses yang lebih mudah bagi mahasiswa melalui perangkat *mobile*. Dengan cakupan yang terfokus pada kebutuhan dasar pengelolaan inventaris kampus, penelitian ini diharapkan dapat menjadi alternatif solusi praktis yang mudah diadopsi oleh institusi pendidikan.

Penelitian ini mengambil studi kasus di Universitas Islam Indonesia (UII)**.** Pada kenyataannya, pengelolaan inventaris di UII dilakukan secara desentralisasi, di mana setiap fakultas atau jurusan memiliki unit dan prosedur peminjaman inventaris masing-masing. Perbedaan aturan mencakup mekanisme persetujuan peminjaman, jangka waktu peminjaman, hingga adanya ketentuan denda keterlambatan pengembalian barang pada beberapa unit. Kondisi ini menunjukkan bahwa standar pengelolaan inventaris di UII masih bervariasi antar fakultas.

Aplikasi yang dikembangkan dalam penelitian ini dirancang untuk memberikan gambaran solusi berbasis mobile. Sistem ini bersifat tersentralisasi, dengan tujuan memudahkan pencatatan dan pelaporan inventaris secara terpadu. Namun, implementasi sistem tetap akan dibandingkan dengan aturan yang berlaku di UII, sehingga dapat diketahui kesesuaian maupun keterbatasannya dalam mendukung kebutuhan nyata di lingkungan kampus.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, berikut adalah rumusan masalah yang diidentifikasi dalam penelitian ini:

1. Pengelolaan inventaris di kampus masih mengandalkan metode manual yang rentan terhadap kesalahan pencatatan, duplikasi data, dan ketidaksesuaian antara catatan inventaris dengan kondisi fisik barang.
2. Sistem berbasis *website*/*desktop* yang tersedia belum optimal dalam menyediakan akses *real-time* dan dukungan terpadu untuk proses peminjaman barang, sehingga menghambat efisiensi operasional.
3. Pengguna (staf dan mahasiswa) memerlukan solusi *mobile* praktis yang dapat mengelola inventaris sekaligus memfasilitasi peminjaman barang secara terpadu, karena solusi yang ada saat ini belum mendukung kedua fungsi tersebut dengan baik.

1.3 Pertanyaan Penelitian

Berdasarkan rumusan masalah yang telah dijelaskan, berikut adalah pertanyaan penelitian yang disusun untuk menjawab setiap permasalahan yang ada:

1. Bagaimana cara mengembangkan sistem pengelolaan inventaris berbasis *mobile* yang dapat mengurangi kesalahan pencatatan manual, duplikasi data, dan ketidaksesuaian catatan dengan kondisi fisik barang di kampus?

2. Bagaimana merancang aplikasi *mobile* yang menyediakan akses *real-time* dan dukungan terpadu untuk proses peminjaman barang, mengatasi keterbatasan sistem berbasis *website*/*desktop* yang ada saat ini?
3. Bagaimana membangun aplikasi *mobile* praktis dengan antarmuka intuitif yang memfasilitasi pengelolaan inventaris sekaligus peminjaman barang secara terpadu untuk memenuhi kebutuhan spesifik pengguna di lingkungan kampus?

1.4 Batasan Masalah

Agar penelitian lebih terfokus, batasan masalah yang ditetapkan adalah sebagai berikut:

1. Aplikasi ini hanya mengelola peralatan laboratorium, alat olahraga, buku, elektronik, dan furnitur tanpa mencakup barang habis pakai atau aset di luar lingkungan kampus.
2. Pengembangan hanya mencakup pencatatan barang, peminjaman (pengajuan – persetujuan – pengembalian), pelacakan status, dan pengelolaan tugas terkait inventaris.
3. Penelitian ini tidak mencakup integrasi dengan sistem eksternal (akademik, keuangan) atau pembayaran denda.

1.5 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Mengembangkan sistem pengelolaan inventaris berbasis *mobile* yang dapat mengurangi kesalahan pencatatan, duplikasi data, dan mempercepat pengambilan keputusan dalam pengelolaan inventaris di kampus.
2. Merancang aplikasi *mobile* yang mendukung pengelolaan inventaris secara *real-time*, termasuk pelacakan status barang, dan pembaruan data yang akurat untuk meningkatkan efisiensi dan keakuratan manajemen inventaris.
3. Menciptakan antarmuka aplikasi yang intuitif dan mudah digunakan, agar pengguna, baik mahasiswa maupun staf kampus dapat mengelola inventaris dengan lebih nyaman dan efektif.

1.6 Manfaat Penelitian

Adapun manfaat dari penelitian ini diharapkan dapat memberikan kontribusi sebagai berikut:

1. Mempermudah dan meningkatkan kenyamanan dalam memantau, mencatat, dan mengelola inventaris dengan akurasi lebih tinggi melalui aplikasi *mobile*.
2. Mengurangi beban kerja terkait administrasi inventaris berkat fitur *real-time* yang memungkinkan pembaruan data langsung dan pelacakan status barang yang lebih efisien.
3. Meningkatkan efisiensi pengelolaan barang di kampus, sehingga meminimalkan kesalahan pencatatan dan duplikasi data serta mempercepat proses pengambilan keputusan terkait inventaris.

1.7 Metodologi Penelitian

Untuk mencapai keberhasilan dalam pengembangan aplikasi *mobile* ini, penelitian ini akan menggunakan strategi penelitian dan pengembangan. Pendekatan ini akan dipadukan dengan model pengembangan perangkat lunak *waterfall*, yang terdiri dari beberapa tahapan sebagai berikut:

1. Studi Literatur

   Penulis melakukan eksplorasi literatur terkait sistem pengelolaan inventaris, studi kasus sistem serupa, dan teknik pengembangan terkini. Analisis sumber dilakukan melalui kajian jurnal akademis dan dokumentasi teknis terkait sistem manajemen inventaris berbasis *mobile*.
2. Analisis Kebutuhan Sistem

   Penulis melakukan identifikasi kebutuhan melalui wawancara semi-terstruktur dengan *stakeholder* kunci. Analisis ini berfokus pada pemahaman mendalam tentang kebutuhan pengguna, tantangan operasional, dan fitur-fitur yang diharapkan dalam sistem digital. *Outputnya* adalah spesifikasi kebutuhan fungsional dan non-fungsional yang terstruktur.
3. Perancangan Sistem

   Penulis merancang desain teknis menyeluruh menggunakan berbagai alat desain. Hal ini mencakup pembuatan diagram alur proses bisnis yang memvisualisasikan siklus peminjaman barang, mulai dari pengajuan hingga pengembalian. Selain itu, dirancang

relasi tabel *database* untuk memodelkan struktur *database*. Untuk antarmuka pengguna, dibuat *wireframe* spesifik untuk dua peran (admin dan peminjam).

4. Implementasi

   Penulis mengembangkan aplikasi *mobile* menggunakan Flutter. Proses ini mencakup pengembangan antarmuka pengguna dan logika sistem. Aplikasi menggunakan Firebase sebagai penyimpan data untuk mengelola informasi penting seperti data pengguna, inventaris barang, dan transaksi peminjaman. Kode program ditulis sesuai desain yang telah dibuat untuk membuat aplikasi berfungsi dengan baik.

5. Pengujian Sistem

   Penulis melakukan pengujian sistem yang melibatkan dua pendekatan: (1) *Black-box Testing* untuk memvalidasi alur fungsional utama berdasarkan skenario penggunaan aktual seperti pengajuan peminjaman dan konfirmasi pengajuan, serta (2) *Usability Testing* menggunakan *System Usability Scale* versi positif (*Positive* SUS) untuk mengukur kepuasan pengguna terhadap kemudahan penggunaan.

1.8 Sistematika Penulisan

Struktur penulisan laporan yang digunakan dalam penyusunan tugas akhir ini dapat dijelaskan sebagai berikut:

BAB I PENDAHULUAN

Bab ini menguraikan latar belakang penelitian, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metode penelitian, serta sistematika penulisan tugas akhir secara rinci.

BAB II KAJIAN PUSTAKA

Bab ini berisi teori-teori dasar yang relevan sebagai fondasi dalam penelitian ini. Bab ini juga memgulas berbagai metode yang telah diterapkan dalam penelitian sebelumnya yang memiliki topik serupa. Setiap metode akan dibahas dengan mempertimbangkan kelebihan dan kekuarannya masing-masing, sehingga dapat menjadi acuan dalam memilih metode yang paling tepat untuk kebutuhan penelitian ini.

BAB III METODOLOGI PENELITIAN

Bab ini menguraikan metode yang digunakan, termasuk penerapan metode *waterfall* dalam pengembangan aplikasi, serta teknik-teknik pengembangan sistem yang mencakup

analisis kebutuhan untuk *input*, *output*, dan proses, serta perancangan desain untuk pengembangan sistem.

BAB IV HASIL DAN PEMBAHASAN

Bab ini memaparkan hasil penelitian yang telah dilakukan. Data dan informasi yang diperoleh ditampilkan dalam bentuk tabel dan gambar. Setiap hasil kemudian dibahas secara sistematis sesuai dengan tahapan penelitian.

BAB V KESIMPULAN DAN SARAN

Bab terakhir ini menyimpulkan secara ringkas mengenai temuan penting dari penelitian yang telah dilakukan, sesuai dengan tujuan dan rumusan masalah yang telah dirumuskan di awal. Selain itu, bab ini juga memberikan saran-saran yang bersifat membangun untuk pengembangan lebih lanjut dari aplikasi yang dikembangkan.

BAB II
KAJIAN PUSTAKA

2.1 Sistem Pengelolaan Inventaris

Sistem pengelolaan inventaris merupakan solusi terkomputerisasi untuk mencatat, memantau, dan mengendalikan alur barang secara efektif. Menurut (Chandra dkk., 2023), sistem ini sangat penting dalam lingkungan perguruan tinggi karena mampu mengelola asset seperti peralatan laboratorium, buku perpustakaan, dan barang inventaris lainnya, sekaligus memfasilitasi proses peminjaman yang terstruktur. Dengan sistem ini, perguruan tinggi dapat mengurangi kesalahan manusia yang sering terjadi pada pencatatan manual dan meningkatkan efisiensi operasional.

Perkembangan sistem pengelolaan inventaris telah mengalami transformasi signifikan dari metode konvensional ke berbasis digital. Seperti yang dijelaskan oleh (Chin Wei dkk., 2023), sistem modern dilengkapi dengan fitur-fitur canggih seperti pelacakan stok *real-time*, notifikasi stok rendah, dan laporan otomatis. Fitur-fitur ini tidak hanya meningkatkan akurasi data tetapi juga memudahkan pengelolaan inventaris dalam skala besar, seperti di lingkungan kampus yang melibatkan banyak pengguna dan aset. Selain itu, sistem ini juga mendukung analisis data untuk pengambilan keputusan yang lebih baik, seperti menentukan waktu yang tepat untuk restok barang.

Implementasi sistem pengelolaan inventaris di lingkungan kampus bahkan memberikan banyak manfaat, terutama dalam hal efisiensi waktu dan biaya. Menurut (Chandra dkk., 2023), sistem ini mampu menyederhanakan proses administrasi yang sebelumnya memakan waktu lama menjadi lebih cepat dan terintegrasi. Dengan adanya fitur peminjaman barang yang terkomputerisasi, mahasiswa dan staf dapat dengan mudah memantau status barang, mengajukan peminjaman, dan menerima notifikasi. Hal ini tidak hanya mengurangi beban kerja staf tetapi juga meningkatkan transparansi dan akuntabilitas dalam pengelolaan aset kampus.

2.2 Penelitian Terdahulu

Dalam mendukung penelitian ini, penulis meninjau beberapa studi sebelumnya yang berfokus pada pengembangan sistem inventaris dan peminjaman barang di lingkungan perguruan tinggi. Hasil kajian literatur memperlihatkan bahwa sebagian besar institusi pendidikan masih menghadapi permasalahan yang sama, yaitu proses inventarisasi dan peminjaman yang masih manual sehingga rentan terhadap kesalahan pencatatan, kehilangan data, serta keterlambatan dalam penyusunan laporan. Ringkasan penelitian terdahulu dapat dilihat pada table berikut:

Tabel 2.1 Ringkasan Penelitian Terdahulu Sistem Inventaris dan Peminjaman Barang

Judul Penelitian	Lokasi/Objek Penelitian	Temuan Utama	Kendala	Hasil Penelitian
Sistem Informasi Inventaris Untuk Peminjaman dan Pengembalian Alat di Laboratorium Teknik Informatika Polnep Berbasis Web (Wibowo dkk., 2021)	Laboratorium Teknik Informatika, Politeknik Negeri Pontianak	Sistem inventaris dan peminjaman masih berbasis kertas	Data sulit dikelola, formulir rentan hilang, laporan stok manual	Sistem berbasis *website* dengan QR *Code* & integrasi *database*
PENGEMBANGAN SISTEM APLIKASI MANAJEMEN BERBASIS ANDROID DI UNIVERSTIAS NEGERI GORONTALO (Yahya dkk., 2024)	Universitas Negeri Gorontalo	Aset kampus dikelola manual oleh satu unit perlengkapan	Tidak ada sistem terintegrasi, laporan lambat	Aplikasi manajemen aset berbasis Android (fokus pencatatan aset, bukan khusus peminjaman)
Rancang Bangun Sistem Informasi Manajemen Inventatis Barang Pada Kampus Widya Kartika (Hizkia dkk., 2024)	Universitas Widya Kartika, Surabaya	Inventaris dan peminjaman barang masih manual dengan kertas	Rentan salah, sulit dilacak	Sistem informasi manajemen inventoris berbasis *website*

Aplikasi Pengelolaan Inventaris Kampus STTI NIIT Bebasis Web (Aulia & Setiyani, 2024)	Kampus STTI NIIT I-Tech	Inventaris dicatat manual di buku	Duplikasi data, kehilangan arsip, sulit cari data historis	Sistem inventaris berbasis *website* untuk gudang, ruangan, dan laboratorium
Sistem Pendataan Inventaris Barang Pada Program Studi Teknik Informasika Universitas Malikussaleh	Prodi TI, Universitas Malikussaleh	Pendataan inventaris masih manual	Pencatatan lambat, sulit dikelola berkala	Sistem inventaris berbasis *desktop*

Berdasarkan hasil penelitian-penelitian tersebut, terlihat bahwa solusi yang ditawarkan umumnya masih terbatas pada sistem berbasis *website* atau *desktop* dengan fokus pada inventarisasi barang atau ruang lingkup unit tertentu. Hanya sebagian kecil yang mengembangkan aplikasi mobile, itupun lebih menekankan pada manajemen aset kampus secara umum tanpa memberikan alur terintegrasi antara pihak peminjam (mahasiswa/dosen) dan admin (pengelola inventaris).

Dengan demikian, penelitian ini berusaha mengisi celah tersebut dengan mengembangkan aplikasi *mobile* yang secara khusus dirancang untuk memfasilitasi proses peminjaman dan pengembalian inventaris di kampus secara *real-time* dan terintegrasi antara peminjam dan admin. Keunggulan ini diharapkan dapat meningkatkan efisiensi, transparansi, serta akurasi pengelolaan inventaris di lingkungan perguruan tinggi.

2.3 Kategorisasi Metodologi Pengembangan Sistem

Berikut adalah tabel perbandingan metodologi pengembangan sistem inventaris barang berdasarkan penelitian terlebih dahulu yang relevan. Tabel ini menunjukkan kesimpulan, aspek yang diperhatikan, serta relevansinya dari setiap metodologi.

Tabel 2.2 Perbandingan Metodologi Pengembangan Sistem Inventaris Barang Berdasarkan Penelitian Terdahulu

Metode	Judul Penelitian	Kesimpulan	Aspek yang Diperhatikan	Relevansi
Agile	- Penerapan Metode Agile Dalam Pemodelan Sistem Informasi Inventory Barang (Nuroji, 2023) - RANCANG BANGUN SISTEM INFORMASI INVENTARIS BARANG MENGGUNAKAN METODE AGILE DI SEKOLAH MENENGAH KEJURUAN BINA PUTERA NUSANTARA (Sidik dkk., 2024) - Rancang Bangun Sistem Informasi Inventory Menggunakan Metode Agile Software Development (Studi Kasus Toko Nada) (Widiarta dkk., 2023)	Ketiga penelitian secara konsisten menunjukkan bahwa metode a*gile* sangat efektif dan adaptif dalam pengembangan sistem informasi inventaris barang, menghasilkan sistem yang responsif, mudah digunakan, dan berkualitas tinggi (dibuktikan dengan pengujian fungsionalitas 100% dan nilai SUS "Good"), serta mampu meningkatkan efisiensi pengelolaan dan manajemen keuangan.	Fleksibilitas dalam pengembangan, iterasi cepat, adaptasi terhadap perubahan kebutuhan, kolaborasi intensif antara tim dan pengguna, fokus pada *user experience*, serta pengujian berkelanjutan untuk memastikan kualitas dan fungsionalitas sistem.	Relevan. Metode a*gile* terbukti ideal untuk pengembangan aplikasi pengelolaan dan peminjaman inventaris barang karena kemampuannya mengakomodasi perubahan, menghasilkan produk yang sesuai kebutuhan pengguna, dan meningkatkan efisiensi operasional dalam lingkungan yang dinamis.

Waterfall	- SISTEM INFORMASI INVENTORY BARANG MENGGUNAKAN METODE WATERFALL (Wijaya & Lomban, 2022) - Penerapan Metode Waterfall Untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang (Badrul, 2021) - PENGEMBANGAN APLIKASI SISTEM INFORMASI PADA CV WIJAYA LAS KEDIRI MENGGUNAKAN MODEL WATERFALL (Arianto dkk., 2021)	Ketiga penelitian secara konsisten menunjukkan bahwa metode w*aterfall* efektif dalam mengembangkan sistem informasi inventaris barang yang terstruktur, akurat, dan efisien. Sistem yang dihasilkan mampu menyederhanakan pengolahan data, mempermudah pencatatan dan pelaporan, serta meningkatkan pelayanan dan manajemen, dengan hasil pengujian fungsionalitas dan kelayakan yang positif.	Pendekatan sekuensial dan terstruktur dengan tahapan yang jelas (analisis kebutuhan, desain, implementasi, pengujian, pemeliharaan), fokus pada perencanaan awal yang komprehensif, dokumentasi yang lengkap di setiap fase, serta pengujian sistem yang sistematis untuk memastikan kualitas dan fungsionalitas.	Sangat relevan. Metode w*aterfall* sangat cocok untuk pengembangan aplikasi pengelolaan dan peminjaman inventaris barang, terutama jika kebutuhan sistem telah didefinisikan dengan sangat jelas dan stabil di awal proyek, serta membutuhkan dokumentasi yang kuat dan proses pengembangan yang linear untuk memastikan akurasi dan keandalan data inventaris.
Rapid Application Development (RAD)	- Implementasi Metode Rapid Application Development pada Pengembangan Aplikasi Inventory	Ketiga penelitian secara konsisten menunjukkan bahwa metode *Rapid Application Development* (RAD) sangat	Siklus pengembangan yang dipercepat dan fleksibel, penggunaan prototipe dan iterasi,	Relevan. Metode RAD sangat cocok untuk pengembangan aplikasi *mobile* pengelolaan dan

	Barang (Anisah, 2022) - Rancang Bangun Sistem Informasi Inventory Menggunakan Metode Rapid Application Development (Rianto & Amrin, 2023) - Analisis dan Perancangan Sistem Informasi Inventory Barang pada PT Cipta Rasa Multindo S (Nuryasin dkk., 2021)	efektif dalam mengembangkan sistem informasi inventaris barang. RAD memungkinkan pengembangan yang cepat, fleksibel, dan efisien, menghasilkan sistem yang akurat, mudah digunakan, dan mampu mengatasi masalah pengelolaan manual serta meningkatkan efisiensi operasional dan kepuasan pengguna.	kolaborasi intensif antara pengembang dan pengguna, fokus pada kecepatan dan efisiensi waktu, serta pengujian fungsionalitas untuk validasi sistem.	peminjaman inventaris barang di kampus, terutama jika ada kebutuhan mendesak untuk sistem yang berfungsi cepat, dengan fleksibilitas untuk mengakomodasi *feedback* awal dari pengguna kampus melalui prototipe.
Prototyping	- IMPLEMENTASI METODE PROTOTYPING PADA SISTEM INFORMASI PENGADAAN BARANG CETAKAN BERBASIS WEB (Ardiyansah dkk., 2021)	Ketiga penelitian secara konsisten menunjukkan bahwa metode p*rototyping* sangat efektif dalam mengembangkan sistem informasi inventaris barang. Metode ini memungkinkan	Siklus pengembangan evolusioner dan interaktif yang melibatkan komunikasi, perencanaan cepat, desain cepat, konstruksi prototipe, serta *deployment* dan	Sangat relevan. Metode p*rototyping* sangat cocok untuk pengembangan aplikasi *mobile* pengelolaan dan peminjaman inventaris barang di kampus,

	- Sistem Informasi Inventaris Berbasis Web Menggunakan Metode Prototype (Nurhadi & Muhammad Ridwan, 2022) - Penerapan Metode Prototype dalam Pengembangan Sistem Informasi Inventory Barang Berbasis Web (Aprilisa & Aulia, 2024)	pengembangan yang adaptif, meminimalkan kesalahan melalui evaluasi berkelanjutan dan umpan balik pengguna, serta menghasilkan sistem yang akurat, efisien, dan sesuai dengan kebutuhan riil pengguna dalam pengelolaan data dan pelaporan.	*feedback* berkelanjutan. Fokus pada validasi kebutuhan awal, pengujian berbasis pengguna, dan modifikasi sistem secara iteratif.	terutama untuk memvalidasi kebutuhan pengguna secara dini, mengakomodasi perubahan, dan memastikan sistem yang dihasilkan benar-benar intuitif dan fungsional bagi pengguna akhir.
User-Centered Design (UCD)	- PENGEMBANGAN SISTEM INVENTORY CONTROL DENGAN METODE USER-CENTERED DESIGN DAN EVALUASI USABILITY (Tarigan dkk., 2024) - Implementasi Metode User Centered Design (UCD) Pada Web Inventori Toko KickID (Ramadhan & Andrianingsih, 2024)	Ketiga penelitian secara konsisten menunjukkan bahwa metode *User-Centered Design* (UCD) sangat efektif dalam mengembangkan sistem informasi inventaris barang yang berpusat pada pengguna. Sistem yang dihasilkan tidak hanya fungsional dan efisien dalam pengelolaan stok dan data, tetapi	Penempatan pengguna sebagai fokus utama di setiap tahapan pengembangan (analisis kebutuhan, desain, evaluasi, implementasi), pengumpulan umpan balik pengguna secara berkelanjutan, desain yang intuitif dan mudah dipahami, serta	Relevan. Metode UCD sangat cocok untuk pengembangan aplikasi *mobile* pengelolaan dan peminjaman inventaris barang di kampus, karena memastikan sistem yang dibangun benar-benar memenuhi kebutuhan, preferensi, dan ekspektasi pengguna akhir

	- Perancangan Sistem Informasi Inventaris Barang Berbasis Website Menggunakan Metode User Centered Design: Studi Kasus BAZNAS Kota Depok (Sholahuddin Alfarisy, 2024)	juga sangat mudah digunakan dan memuaskan pengguna, dibuktikan dengan skor *usability* yang tinggi (SUS 80-90, SUMI 95).	pengujian *usability* yang komprehensif untuk memastikan kepuasan pengguna.	(mahasiswa, staf, dosen), sehingga menghasilkan pengalaman pengguna yang optimal dan adopsi sistem yang tinggi.

Setelah menganalisis berbagai penelitian terlebih dahulu mengenai metodologi pengembangan sistem inventaris barang melalui Tabel 2.2, langkah selanjutnya adalah melakukan evaluasi komparatif yang lebih mendalam terhadap karakter intrinsik setiap metodologi. Evaluasi ini difokuskan pada identifikasi kelebihan dan kekurangan masing-masing pendekatan dalam konteks pengembangan sistem.

Perbandingan ini penting untuk memberikan landasan objektif dalam pemilihan metodologi yang paling sesuai dengan karakteristik proyek pengembangan aplikasi *mobile* untuk pengelolaan dan peminjaman inventaris barang di lingkungan kampus. Berikut adalah analisis perbandingan sistematis yang merangkum aspek-aspek kritis setiap metodologi:

Tabel 2.3 Perbandingan Kelebihan dan Kekurangan Metodologi Pengembangan Sistem

Metode	Judul Penelitian	Kelebihan	Kekurangan
Agile	A comparative analysis of RAD and agile technique for management of computing graduation projects (Khan dkk., 2020)	Respons cepat terhadap perubahan	Biaya awal lebih tinggi
		Iterasi pendek memungkinkan perbaikan terus-menerus	Memerlukan komunikasi yang baik antara pengembang dan pelanggan
		Kualitas dan kepuasan pengguna lebih tinggi	Pengalaman tim yang baik sangat diperlukan
Waterfall	Waterfall methodology, prototyping and agile	Mudah dipahami dan digunakan	Tidak ada perangkat lunak yang berfungsi hingga tahap akhir

	development (Senarath, 2021)	Manajemen proyek lebih mudah karena proses yang kaku	Sulit melakukan perubahan setelah tahap pengujian
		Resiko lebih rendah pada proyek dengan spesifikasi yang jelas	Tidak cocok untuk proyek kompleks atau berubah-ubah
		Tahapan diproses dan diselesaikan satu per satu	Resiko dan ketidakpastian tinggi jika kebutuhan beresiko tinggi berubah
		Tahapan yang jelas, *milestone* dipahami dengan baik, mudah mengatur tugas, dan hasil terdokumentasi dengan baik	Integrasi dilakukan sebagai "*big-bang*" di akhir, tidak memungkinkan identifikasi tambahan teknologi atau bisnis lebih awal
Rapid Application Development (RAD)	A comparative analysis of RAD and agile technique for management of computing graduation projects (Khan dkk., 2020)	Proses lebih cepat dibandingkan w*aterfall*	Memerlukan banyak prototyping, yang dapat meningkatkan biaya
		Menggunakan prototipe untuk mendaptkan umpan balik pengguna lebih cepat	Sulit untuk memperluas prototipe pada proyek besar.
		Cocok untuk proyek kecil dengan kebutuhan yang jelas	Perubahan yang signifikan memerlukan banyak revisi
		Dokumentasi lebih ekstensif (13.33% lebih baik dari a*gile*)	Kurang fleksibel dibandingkan a*gile*
		Dapat memvisualisasikan produk aktual dan memodifikasi/mengadopsi fungsionalitas	Pengembangan prototipe dapat menjadi *overhead* tambahan dan membutuhkan banyak upaya ekstra

Prototyping	Waterfall methodology, prototyping, and agile development (Senarath, 2021)	Pengguna dapat melihat model sistem lebih awal	Pengguna mungkin bingung dengan perbedaan antara prototipe dan sistem final
		Melibatkan pengguna lebih aktif sejak awal	Analisis kebutuhan mungkin tidak cukup mendalam (risiko analisis kebutuhan tidak memadai)
		Mendeksi masalah lebih cepat	Menambah kompleksitas (jika ruang lingkup meluas)
		Biaya dan waktu lebih efisien (karena deteksi cacat lebih awal)	Pengembang mungkin mencoba menggunakan Kembali protipe yang ada meskipun tidak layak secara teknis
		Umpan balik pengguna lebih cepat tersedia, mengarah kek solusi yang lebih baik	Upaya yang diinvestasikan dalam membangun prototipe mungkin terlalu banyak jika tidak dipantau dengan benar
		Fungsi yang membingunkan atau sulit dapat diidentifikasi	
User Centered Design (UCD)	Implementation User-Centered Design in Agile Software Development: Systematic Literature Review (Shania dkk., 2023)	Memastikan bahwa pengembangan fokus pada kebutuhan pengguna	Memerlukan dokumentasi yang baik (kekurangan a*gile* yang perlu diatasi UCD)
		Mengurangi redundansi proses	Komunikasi yang intensif antara tim dan pemangku kepentingan diperlukan
		Meningkatkan komunikasi antara pemangku kepentingan	Memerlukan perancangan dan manajemen yang matang

		Meningkatkan kualitas produk (termasuk kesederhanaan dan kode berkualitas)	Pengambilan keputusan yang lambat dari manajemen dan kurangnya visi UX dapat menjadi hambatan
		Mengurangi kemungkinan kesalahpahaman persyaratan dan *bug*	Membutuhkan keterlibatan tinggi dari setiap anggota tim pengembangan
		Meningkatkan kepuasan pengguna dan ROI	Ketersediaan sumber daya dapat menjadi tantangan, terutama untuk *upfront design*

Tabel 2.3 mengungkapkan konfigurasi unik dari setiap metodologi, di mana keunggulan suatu pendekatan sering berbanding terbalik dengan kelemahanannya. *waterfall* menawarkan struktur jelas tapi kurang fleksibel, sementara *agile* fleksibel namun membutuhkan disiplin tinggi. RAD dan *prototyping* memberikan kecepatan dengan risiko ruang lingkup yang kurang terdefinisi, sedangkan UCD menjamin kepuasan pengguna meski dengan intensitas sumber daya yang lebih besar. Pola-pola ini memberikan landasan untuk analisis lebih mendalam tentang kesesuaian metodologi dengan karakteristik proyek sistem inventaris kampus.

2.4 Pembahasan Metodologi

Berdasarkan Tabel 2.2 dan Tabel 2.3, analisis terhadap berbagai metodologi pengembangan sistem informasi inventaris barang menunjukkan bahwa setiap pendekatan memiliki keunggulan dan kelemahan yang berbeda, serta relevansi yang bervariasi dalam konteks penelitian ini.

Metode *agile*, misalnya, dikenal karena fleksibilitasnya dan kemampuannya untuk merespons perubahan kebutuhan dengan cepat. Pendekatan ini menekankan iterasi pendek dan komunikasi intensif dengan pengguna, yang memungkinkan pengembang untuk terus memperbaiki sistem berdasarkan umpan balik. Namun, dalam konteks proyek ini, di mana cakupan kebutuhan sudah cukup jelas sejak awal, iterasi berulang yang menjadi ciri khas *agile* tidak sepenuhnya diperlukan. Selain itu, metode ini memerlukan biaya awal yang cukup tinggi dan pengalaman tim yang solid, yang menjadi tantangan ketika pengembangan dilakukan secara mandiri tanpa dukungan tim.

Metode *waterfall* menawarkan pendekatan sistematis dengan tahapan yang jelas, seperti analisis kebutuhan, perancangan, implementasi, dan pengujian. Pendekatan ini sangat cocok untuk proyek yang memiliki spesifikasi yang sudah didefinisikan dengan baik di awal dan tidak memerlukan banyak perubahan selama proses pengembangan. Meskipun *waterfall* memiliki kelemahan dalam hal fleksibilitas, dalam konteks sistem inventaris, di mana kebutuhan pengguna dapat diprediksi dan stabil, pendekatan ini memberikan struktur yang diperlukan untuk memastikan bahwa semua aspek sistem dapat dikelola dengan baik. Kelemahan utama *waterfall*, yaitu kurangnya respons terhadap perubahan, dapat diatasi dengan melakukan analisis kebutuhan yang mendalam di awal, sehingga meminimalkan risiko revisi besar di tahap akhir.

Rapid Application Development (RAD) juga menjadi pilihan yang menarik karena kecepatan pengembangannya dan keterlibatan pengguna sejak awal melalui *prototyping*. Namun, meskipun RAD mendukung pengembangan cepat, pendekatan ini kurang sesuai untuk proyek berskala besar atau dengan kebutuhan yang kompleks seperti sistem inventaris. Selain itu, RAD memerlukan banyak sumber daya untuk membuat protoipe, yang dapat meningkatkan biaya, terutama ketika pengembangan dilakukan secara mandiri.

Metode *prototyping* menekankan pada pengujian langsung oleh pengguna dengan menggunakan prototipe sebagai model awal sistem. Meskipun relevan untuk proyek dengan kebutuhan yang tidak sepenuhnya jelas di awal, dalam konteks penelitian ini, pendekatan *waterfall* lebih sesuai karena spesifikasi sistem inventaris sudah ditetapkan dengan jelas. *prototyping* juga memerlukan keterlibatan pengguna yang intensif, yang sulit dicapai dalam pengembangan yang dilakukan sendirian.

Terakhir, *User Centered Design* (UCD) berfokus pada pengalaman dan kebutuhan pengguna dengan memastikan desain antarmuka yang intuitif. Meskipun UCD sangat relevan untuk proyek yang memprioritaskan pengalaman pengguna, metodologi ini memerlukan dokumentasi yang intensif dan proses komunikasi yang mendalam. Hal ini menjadi tantangan pada proyek berskala kecil yang dikerjakan secara mandiri, di mana sumber daya dan waktu terbatas.

2.5 *System Usability Scale* (SUS) dan Variasinya

System Usability Scale (SUS) pertama kali diperkenalkan oleh John Brooke pada tahun 1986 sebagai instrumen cepat dan sederhana untuk mengukur *usability* suatu sistem berbasis kuesioner (Brooke, 1996). SUS terdiri dari sepuluh butir pernyataan dengan skala Likert 5 poin yang menghasilkan skor *usability* yang dapat diinterpretasikan secara kuantitatif.

Dalam perkembangannya, SUS memiliki beberapa variasi bentuk. Pertama, SUS standar, yaitu versi asli yang memuat lima pernyataan positif dan lima pernyataan negative (Brooke, 1996). Format ini menjaga keseimbangan jawaban, meskipun perbedaan polaritas kalimat kadang menimbulkan kebingungan bagi responden. Kedua, SUS positif (*all-positive* SUS), di mana semua pernyataan disusun dalam bentuk positif sehingga responden lebih mudah memahami pertanyaan tanpa perlu menyesuaikan arah skala (Sauro & Lewis, 2011). Ketiga, SUS singkat (*shortened* SUS) yang dikembangkan untuk meningkatkan efisiensi waktu dengan mengurangi jumlah item menjadi lima atau delapan, meskipun reliabilitasnya sedikit lebih rendah dibandingkan versi standar (Bangor dkk., 2008). Keempat, SUS kontekstual, yaitu modifikasi butir pertanyaan agar lebih sesuai dengan konteks sistem tertentu, misalnya pada aplikasi *mobile*, sistem kesehatan digital, atau *e-learning* (Sauro & Lewis, 2011).

BAB III
METODOLOGI PENELITIAN

3.1 Pemilihan Metodologi

Berdasarkan analisis komparatif terhadap berbagai metodologi pengembangan perangkat lunak yang dibahas pada Bab II, penelitian ini memilih metode *waterfall* sebagai pendekatan utama dalam pengembangan sistem. Pemilihan ini dilakukan dengan mempertimbangkan karakteristik sistem yang akan dibangun, ketersediaan sumber daya, serta tujuan penelitian yang menekankan pada akurasi data, dokumentasi lengkap, dan proses pengembangan yang terukur.

Alasan pertama adalah karena kebutuhan sistem inventaris yang dikembangkan dalam penelitian ini telah didefinisikan secara jelas sejak awal melalui hasil wawancara. Spesifikasi kebutuhan yang relatif stabil membuat metode *waterfall* lebih sesuai dibandingkan pendekatan lain yang menuntut fleksibilitas tinggi, seperti *agile* atau *prototyping*. Dengan adanya kebutuhan yang jelas sejak awal, pendekatan linear yang berurutan dapat mengurangi risiko perubahan mendasar di tengah jalannya pengembangan.

Alasan kedua adalah karena *waterfall* menyediakan struktur tahapan yang sistematis, yaitu dimulai dari analisis kebutuhan, perancangan, implementasi, hingga pengujian. Setiap tahap dalam *waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya, sehingga meminimalkan terjadinya kekeliruan yang dapat berakibat pada revisi besar di akhir. Dalam konteks sistem inventaris, kejelasan struktur ini sangat penting untuk memastikan bahwa seluruh proses pengelolaan barang, mulai dari pencatatan hingga peminjaman, dapat ditangani dengan baik dan terdokumentasi secara menyeluruh.

Alasan ketiga berkaitan dengan dokumentasi. *waterfall* menekankan pentingnya dokumentasi di setiap fase. Hal ini memberikan manfaat besar karena sistem inventaris kampus membutuhkan akurasi data dan catatan pengembangan yang lengkap. Dokumentasi tersebut dapat menjadi acuan apabila sistem perlu dikembangkan lebih lanjut atau dipelihara oleh pihak lain di masa depan. Dengan demikian, kesinambungan sistem lebih terjamin.

Selain itu, alasan keempat berkaitan dengan kondisi sumber daya penelitian. Proyek ini dikembangkan secara individu dengan keterbatasan waktu dan tenaga. Metode iteratif seperti *agile*, RAD, atau UCD memerlukan keterlibatan intensif antara pengembang dan pengguna secara berulang kali, yang sulit diterapkan dalam penelitian ini. Sementara itu, *waterfall* lebih

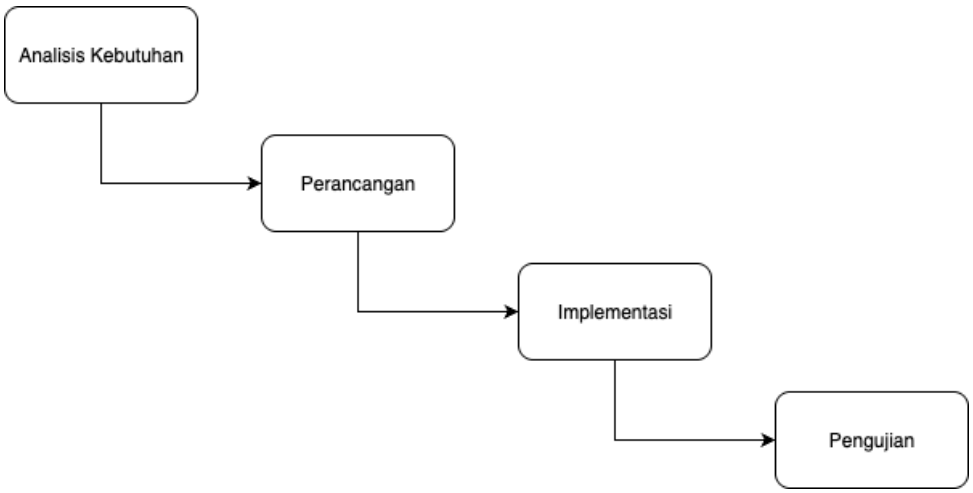
realistis karena fokus pada penyelesaian tahapan satu per satu secara sistematis, tanpa menuntut siklus komunikasi yang terlalu sering.

Kelemahan utama dari metode *waterfall* adalah kurang fleksibel dalam menghadapi perubahan kebutuhan di tengah jalan. Namun, dalam penelitian ini hal tersebut tidak menjadi kendala signifikan karena kebutuhan sistem sudah relatif stabil sejak awal. Analisis kebutuhan dilakukan untuk meminimalkan potensi perubahan besar di akhir pengembangan. Dengan demikian, keterbatasan tersebut masih dapat diantisipasi.

Berdasarkan pertimbangan-pertimbangan tersebut, metode *waterfall* dipandang paling sesuai untuk penelitian ini. Pendekatan ini mampu memberikan keseimbangan antara struktur yang jelas, dokumentasi yang kuat, serta manajemen risiko yang terkontrol. Dengan penerapan *waterfall*, aplikasi *mobile* pengelolaan dan peminjaman inventaris di kampus diharapkan dapat dibangun secara efisien, akurat, dan berkesinambungan sesuai dengan kebutuhan pengguna.

3.2 Diagram Pengembangan Sistem

Pengembangan sistem dalam penelitian ini mengikuti alur terstruktur seperti ditunjukkan pada Gambar 3.1. Tahapan dimulai dari analisis kebutuhan, perancangan, implementasi, hingga pengujian. Diagram ini menjadi kerangka kerja yang memandu seluruh proses pengembangan.



Gambar 3.1 Diagram Alur Pengembangan Sistem

Gambar 3.1 menunjukkan tahapan sistematis pengembangan sistem peminjaman inventaris ini. Proses diawali dengan Analisis Kebutuhan di mana dilakukan pengumpulan data melalui wawancara dengan *stakeholder* untuk mengidentifikasi masalah dan merumuskan solusi. Tahap ini menghasilkan spesifikasi kebutuhan sistem meliputi *input* data, proses, dan *output*.

Selanjutnya masuk ke tahap Perancangan yang mencakup pembuatan berbagai diagram pemodelan. *Use case diagram* dirancang untuk memetakan interaksi antara aktor (admin dan

peminjam) dengan sistem. *Activity diagram* dibuat untuk menggambarkan alur proses bisnis seperti peminjaman dan pengembalian barang. Relasi tabel *database* disusun untuk merancang struktur *database*, sementara *wireframe* antarmuka dirancang menggunakan Figma sebagai panduan visual.

Tahap Implementasi melibatkan pembangunan aplikasi nyata dengan menggunakan Flutter untuk *frontend* dan Firebase untuk *backend*. Pada fase-fase ini dikembangkan fitur-fitur inti seperti sistem autentikasi, manajemen inventaris, mekanisme peminjaman, serta sistem notifikasi.

Setelah Implementasi, dilakukan pengujian sistem menggunakan dua pendekatan utama. Pertama, *Blacbbox Testing* dilakukan untuk memverifikasi fungsionalitas sistem tanpa melihat kode internal. Kedua, *System Usability Scale* versi positif (Positivie SUS) digunakan untuk mengukur mengukur tingkat kepuasan sistem melalui kuosioner standar yang diisi oleh pengguna. Hasil SUS memberikan skor kuantitatif yang mencerminkan persepsi pengguna terhadap kemudahan penggunaan dan efisiensi sistem.

3.3 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dilakukan untuk merumuskan apa saja komponen yang diperlukan dalam pembangunan aplikasi. Data dikumpulkan melalui wawancara semi-terstruktur dengan staf Divisi Administrasi Umum dan Rumah Tangga (DAURT) dan mahasiswa.

Proses analisis kebutuhan dalam penelitian ini dilakukan melalui beberapa tahapan. Pertama, peneliti menyusun daftar pertanyaan wawancara semi-terstruktur berdasarkan literatur dan praktik umum pengelolaan inventaris. Kedua, wawancara dilakukan dengan dua kelompok responden utama, yaitu staf/admin sebagai pengelola inventaris dan mahasiswa sebagai pengguna. Ketiga, jawaban hasil wawancara kemudian dikategorikan menurut tema *input*, proses, *output*, dan kendala yang muncul. Keempat, hasil kategorisasi dianalisis untuk mengidentifikasi masalah utama serta merumuskan solusi sistem yang relevan. Hasil analisis inilah yang kemudian dirangkum dalam Tabel 3.1 (daftar pertanyaan wawancara) dan Tabel 3.2 (identifikasi masalah dan solusi).

a. Demografi Responden Wawancara

Responden wawancara pada penelitian ini terdiri dari dua kelompok, yaitu staf Divisi Administrasi Umum dan Rumah Tangga (DAURT) Fakultas Teknologi Industri (FTI) sebagai pengelola inventaris, serta mahasiswa sebagai pengguna layanan peminjaman barang. Data demografi responden disajikan pada tabel berikut:

Tabel 3.1 Demografi Responden Wawancara

Nama	Peran	Fakultas/Unit	Jenis Kelamin	Foto
Basuki Setiyawan	Admin	FTI - DAURT	Laki-laki	
Amiruddin Syah	Mahasiswa	FIAI – Pendidikan Agama Islam	Laki-laki	
M. Fauzan Afdillah Rahman	Mahasiswa	FH - Hukum	Laki-laki	
Fadhlan Putera Sopian	Mahasiswa	FTI - Informatika	Laki-laki	

Tiwi Esti Ariayanti	Mahasiswa	FPSB - Psikologi	Perempuan	

b. Metode Pengumpulan Kebutuhan

Pengumpulan data dilakukan melalui wawancara semi-terstruktur menggunakan panduan seperti dalam Tabel 3.2 berikut:

Tabel 3.2 Daftar Pertanyaan Wawancara

Responden	Fokus pertanyaan	Pertanyaan Kunci
Admin/Laboran	Proses Peminjaman	Bagaimana alur kerja peminjaman saat ini?
		Bagaimana pengecekan ketersediaan stok?
		Kendala apa yang sering muncul dalam validasi peminjaman?
	Proses Pengembalian	Bagaimana penanganan pengembalian barang?
	Pengelolaan Inventaris	Bagaimana sistem saat ini mencatat dan memantau stok barang?
		Apa kendala dalam memperbarui data inventaris?
Mahasiswa	Pengalaman Peminjaman	Bagaimana proses peminjaman saat ini?
		Kesulitan apa yang dialami?
	Kebutuhan sistem	Fitur apa yang diharapkan dalam sistem digital?

Hasil wawancara kemudian dianalisis menghasilkan identifikasi masalah dan solusi sebagai berikut:

Tabel 3.3 Identifikasi Masalah dari Hasil Wawancara

Pihak Terkait	Masalah yang Teridentifikasi	Solusi
Admin/Laboran	Proses validasi peminjaman manual memakan waktu dan beresiko *human-error* - Verifikasi stok harus dilakukan secara manual	Sistem validasi otomatis dengan pengecekan stok *real-time* - Auto check ketersediaan barang - Validasi data peminjam yang terintegrasi dengan profil nama Peminjam dan data yang telah diisi oleh Peminjam saat *checkout*, yang dapat

	- Pencocokan data peminjam melalui catatan manual - Tidak ada *warning* otomatis untuk stok habis	dilihat secara langsung saat ingin validasi peminjaman - *Approval/reject one-click.*
	Pengecekan ketersediaan stok secara manual tidak selalu akurat - Stok tidak terupdate secara *real-time* - Harus *cross-check* fisik vs catatan - Resiko kesalahan pencatatan tinggi	Sistem inventaris digital dengan *update* stok otomatis - Perubahan stok tercatat secara langsung saat terjadi transaksi - Stok berkurang ketika transaksi disetujui - Stok bertambah ketika pengembalian barang divalidasi
	Sulit melacak riwayat peminjaman - Data tersebar di catatan manual	*Database* terpusat dengan tracking lengkap - Semua transaksi tersimpan - *Advance filtering* berdasarkan status - *Search bar* untuk mempermudah pencarian
	Pencatata pengembalian manual menyebabkan *delay update* stok - Stok tidak langsung *update* - Harus *input* ulang ke catatan untuk memperbarui stok - Risiko *double entry*	Integrasi sistem pengembalian dengan *auto update* - *Auto update* status transaksi saat admin memvalidasi pengembalian - Tambah stok otomatis saat admin memvalidasi pengembalian
Peminjam	Kesulitan cek ketersediaan *real-time* - Harus datang langsung ke lokasi atau melalui chat	Tampilan barang-barang inventaris *live* dengan *checkout system* - *Real time* barang yang tersedia beserta informasi stoknya
	Sulit tracking peminjaman aktif	Personal *loans activity* untuk peminjam

	- Lupa barang yang dipinjam - Lupa dengan *deadline* pengembalian	- Halaman tentang seluruh transaksi yang pernah/sedang dilakukan - Menyediakan seluruh informasi terkait transaksi tersebut, meliputi status, *deadline* pengembalian, barang yang dipinjam, dan catatan admin saat pengajuan diterima/ditolak
	Kebutuhan tambahan dari wawancara Peminjam: Peminjam membutuhkan pengigat untuk pengembalian barang	Sistem notifikasi yang dapat dikirim oleh admin - Pengingat *deadline* pengembalian untuk transaksi yang sudah dekat dengan tenggat pengembalian

Tabel 3.3 menunjukkan solusi yang dirancang untuk mengatasi setiap masalah yang teridentifikasi. Solusi utama meliputi implementasi sistem validasi otomatis dengan pengecekan stok *real-time*, *database* terpusat untuk pelacakan transaksi, serta tampilan inventaris *live* bagi peminjam. Kebutuhan khusus yang muncul dari wawancara adalah sistem notifikasi pengingat pengembalian yang akan dikirim oleh admin saat mendekati *deadline*.

Analisis kebutuhan ini menjadi dasar dalam merancang sistem peminjaman berbasis digital yang akan mengatasi keterbatasan sistem manual sekaligus meningkatkan efisiensi operasional dan pengalaman pengguna.

Meskipun demikian, proses analisis kebutuhan ini masih memiliki keterbatasan. Pertanyaan wawancara yang digunakan belum sepenuhnya mendalam, sehingga beberapa aspek spesifik belum terungkap secara komprehensif. Selain itu, analisis hanya dilakukan pada lingkup terbatas dan belum mencakup seluruh fakultas di Universitas Islam Indonesia (UII), padahal setiap fakultas memiliki prosedur dan aturan peminjaman yang berbeda. Hal ini menyebabkan sistem yang dibangun lebih merepresentasikan kebutuhan sesuai dengan data yang berhasil dihimpun peneliti, bukan gambaran penuh dari keseluruhan kebijakan di tingkat universitas.

c. Kebutuhan *Input*

Berdasarkan solusi-solusi yang telah diidentifikasi dalam Tabel 3.3, sistem digital memerlukan berbagai data *input* untuk mendukung seluruh fungsi utamanya. Input-input ini dirancang untuk mengakomodasi kebutuhan operasional admin sekaligus

mempermudah proses peminjaman bagi peminjam, dengan tetap menjaga akurasi dan keandalan sistem. Kebutuhan *input* dalam sistem ini terdiri dari:

1. Data Akun Pengguna (Admin dan Peminjam):
    - Data Registrasi Akun: Nama lengkap, alamat email, dan kata sandi.
    - Data *Login*: Alamat email dan kata sandi.
2. Data Inventaris Barang (Dikelola oleh Admin):
    - Identitas Barang: Nama, kategori, dan deskripsi barang.
    - Kuantitas: Jumlah total stok barang.
    - Visual Barang: Foto barang (disimpan di Firebase Realtime Database dengan form Url).
3. Data Transaksi Peminjaman (Diinisialisasi oleh Peminjam):
    - Informasi Peminjam: Data pengguna seperti nama (diambil dari data profil pengguna yang sudah mendaftar), NIM, dan nomor telepon.
    - Jadwal Peminjaman: Tanggal dan waktu rencana pengambilan serta pengembalian.
    - Tujuan Peminjaman: Keterangan singkat.
    - Dokumen Peminjaman: Unggahan file untuk foto KTM/Foto Identitas dan dokumen pendukung (opsional).
4. Data Transaksi Pengembalian
    - Dari Peminjam: Peminjam membawa barang secara fisik ke petugas/admin.
    - Dari Admin: Input tindakan untuk menyelesaikan transaksi pada sistem dengan menekan tombol "*Mark as Completed*". Tindakan ini mengubah status transaksi dan memperbarui stok barang.
5. Data Notifikasi dan Pesan Admin
    - Admin Notes: Pesan dari admin terkait persetujuan, penolakan, atau informasi tambahan pada transaksi.
    - Notifikasi Pengembalian: Notifikasi yang dikirim oleh admin untuk mengigatkan peminjam mengembalikan barang yang sudah mendekati atau melewati batas periode waktu pengembalian

d. Kebutuhan Proses

Proses menggambarkan alur kerja logis dari sistem dalam mengelola data *input* menjadi *output* yang sesuai. Proses utama dalam sistem ini meliputi:

1. Proses Registrasi dan Autentikasi Pengguna :
    - Registrasi Akun Peminjam: Validasi input (nama, email, *password*) dan pembuatan akun baru untuk peminjam di Firebase Authentication
    - Autentikasi Pengguna (peminjam dan admin): *Login* pengguna dengan email dan *password* melalui Firebase Authentication. Akun admin telah ditentukan sebelumnya dengan email admin@gmail.com dengan *password* yang sudah ditentukan dan tidak melalui proses registrasi publik.
    - Pengelolaan Sesi *Login* Pengguna
2. Proses Pengelolaan Inventaris Barang (Admin):
    - CRUD (*Create*, *Read*, *Update*, *Delete*) data barang di Firebase.
    - Pembaruan otomatis jumlah stok saat peminjaman disetujui dan barang dikembalikan.
    - Implementasi validasi stok ketat sebelum persetujuan peminjaman, untuk mencegah persetujuan barang *out-of-stock*, checkItemBeforeApproval(), validateApprovalStock().
3. Proses Pencarian dan Penyaringan Barang dan Transaksi:
    - Peminjam dan admin dapat mencari barang berdasarkan nama atau kata kunci.
    - Peminjam dan admin dapat memfilter barang berdasarkan kategori.
    - Peminjam dan admin dapat mencari transaksi berdasarkan nama barang atau kata kunci.
    - Peminjam dan admin dapat memfilter transakasi berdasarkan status transaksi.
4. Proses Pengelolaan Keranjang Peminjaman (*Cart*):
    - Peminjam menambahkan barang yang diinginkan ke keranjang.
    - Peminjam dapat mengubah jumlah barang di keranjang atau menghapus barang.
    - Sistem menyimpan data keranjang per pengguna di Firebase untuk memastikan persistensi dan isolasi data antar pengguna.
5. Proses Pengajuan Peminjaman Barang:
    - Peminjam mengajukan peminjaman barang yang diinginkan ke keranjang.
    - Peminjam mengisi form pengajuan barang dengan mengisi NIM, no telepon, tanggal dan waktu pengambalian serta pengembalian, alasan peminjaman, dan foto identitas.

- Sistem mencatat permintaan peminjaman beserta detailnya.
- Peminjam dapat membatalkan pengajuan dengan catatan jika pengajuan tersebut belum disetujui oleh admin.

6. Proses Validasi Stok:
    - Sistem melakukan pengecekan ketersediaan stok barang sebelum admin dapat menyetujui permintaan (mencegah peminjaman barang yang stoknya habis).
7. Proses Persetujuan atau Penolakan Peminjaman (oleh Admin):
    - Admin meninjau permintaan peminjaman.
    - Admin menyetujui atau menolak permintaan, dengan opsi menambahkan catatan (admin *notes*).
    - Sistem memperbarui status peminjaman dan mengurangi stok barang jika disetujui.
8. Proses Pengelolaan Peminjaman Aktif:
    - Admin dapat memantau barang yang sedang dipinjam.
    - Sistem dapat mengidentifikasi peminjaman yang mendekati atau melewati batas waktu pengembalian.
9. Proses Pengembalian Barang:
    - Peminjam datang ke lokasi secara langsung untuk mengembalikan barang.
    - Admin menekan tombol "*Mark as Completed*" untuk menandai bahwa proses transaksi sudah selesai.
    - Sistem memperbarui status peminjaman menjadi "*Completed*" dan menambah kembali stok barang.
10. Proses Pengiriman Notifikasi (oleh Admin):
    - Notifikasi dari Admin (Status Peminjaman dan Pengingat): Notifikasi yang diinisialisasi oleh admin, seperti persetujuan, penolakan, pengingat batas waktu, atau keterlambatan pengembalian, akan muncul sebagai *pop-up* di aplikasi peminjam dan juga tersimpan di halaman daftar notifikasi.

e. Kebutuhan *Output*

Output adalah hasil akhir yang ditampilkan sistem kepada pengguna berdasarkan proses yang dilakukan. *Output* sistem ini meliputi:

1. Tampilan Daftar Inventaris Barang :
    - Galeri atau daftar barang beserta gambar, nama, dan sisa stok.
    - Detail lengkap barang saat item dipilih (deskripsi barang).
2. Tampilan Keranjang Peminjaman:
    - Daftar barang yang telah dipilih oleh peminjam beserta jumlahnya.
    - Total item dalam keranjang.
3. Informasi Status Transaksi Peminjaman:
    - Tampilan status untuk Peminjam (Menunggu Persetujuan, Ditolak, Sedang Dipinjam, Selesai, dan Terlambat.
    - Riwayat aktivitas peminjaman pengguna.
    - Catatan dari admin terkait transaksi (jika ada).
4. Notifikasi Sistem:
    - Pemberitahuan *real-time* di aplikasi mengenai pembaruan status peminjaman dan pengingat pengembalian.
5. Tampilan untuk Admin:
    - Daftar permintaan peminjaman yang memerlukan tindakan (persetujuan/penolakan).
    - Daftar semua transaksi peminjaman dengan berbagai status.
    - Antarmuka untuk pengelolaan data barang.

f. Kebutuhan Tampilan Antarmuka

Antarmuka aplikasi disusun agar intuitif, mudah digunakan, dan kompatibel dengan perangkat *mobile*. Tampilan untuk peminjam dan admin dirancang berbeda sesuai peran dan kebutuhan fungsional masing-masing.

1. Tampilan Antarmuka Peminjam:
    - Layar *Login* dan Registrasi: Form untuk masuk dan mendaftar akun.
    - Layar *Home* (Beranda): Menampilkan nama akun, bar pencarian, daftar kategori (*horizontal scroll*), dan *grid item* barang yang tersedia. Seluruh *layer* dapat di-*scroll*.

- Layar Detail Barang: Menampilkan gambar barang, nama, kategori, stok, deskripsi, dan tombol "*Add to Cart*".
- Layar Keranjang Peminjaman: Daftar *item* yang akan dipinjam, opsi mengubah jumlah, dan tombol untuk mengajukan peminjaman.
- Layar Formulir Pengajuan Peminjaman: *Form* untuk melengkapi detail pengajuan peminjaman seperti durasi peminjaman, alasan peminjaman, NIM, no telepon, foto identitas.
- Layar Notifikasi: Daftar notifikasi yang diterima pengguna.
- Layar Aktivitas Peminjaman: Riwayat transaksi peminjaman pengguna beserta statusnya.

2. Tampilan Antarmuka Admin:
   - Layar Pengelolaan Barang: Menampilkan daftar barang, opsi tambah, edit, dan hapus barang.
   - Layar *Form* Tambah/Edit Barang: Menampilkan sebuah *form* di mana admin dapat menambahkan atau mengubah semua detail barang.
   - Layar Pengelolaan Permintaan Peminjaman: Menampilkan daftar, detail permintaan, dan tombol setujui/tolak.
   - Layar Aktivitas Peminjaman: Menampilkan daftar semua transaksi peminjaman dengan filter berdasarkan status.

g. Analisis Kebutuhan Perangkat Keras

Kebutuhan perangkat dibagi menjadi dua: untuk pengembangan sistem dan menjalankan aplikasi.

1. Untuk Pengembangan Sistem (Apple MacBook Air M1 2020):
   - Prosesor: Apple M1 Chip.
   - RAM: 8 GB.
   - Peyimpanan: 256 GB.
   - Koneksi Internet: Dibutuhkan untuk mengunduh *library*, sinkronisasi dengan Firebase, dan riset.
2. Untuk Menjalankan Aplikasi (Perangkat Mobile):
   - Sistem Operasi: Android (versi 6.0 Marshmallow ke atas) atau IOS (versi 11.0 ke atas).
   - RAM: Minimal 2 GB.

- Penyimpanan Internal: Cukup ruang untuk instalasi aplikasi dan penyimpanan data *cache*.
- Koneksi Internet: Diperlukan untuk sinkronisasi data dengan Firebase secara *real-time*.

h. Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang dibutuhkan untuk merancang, membangun, dan menguji aplikasi ini meliputi:

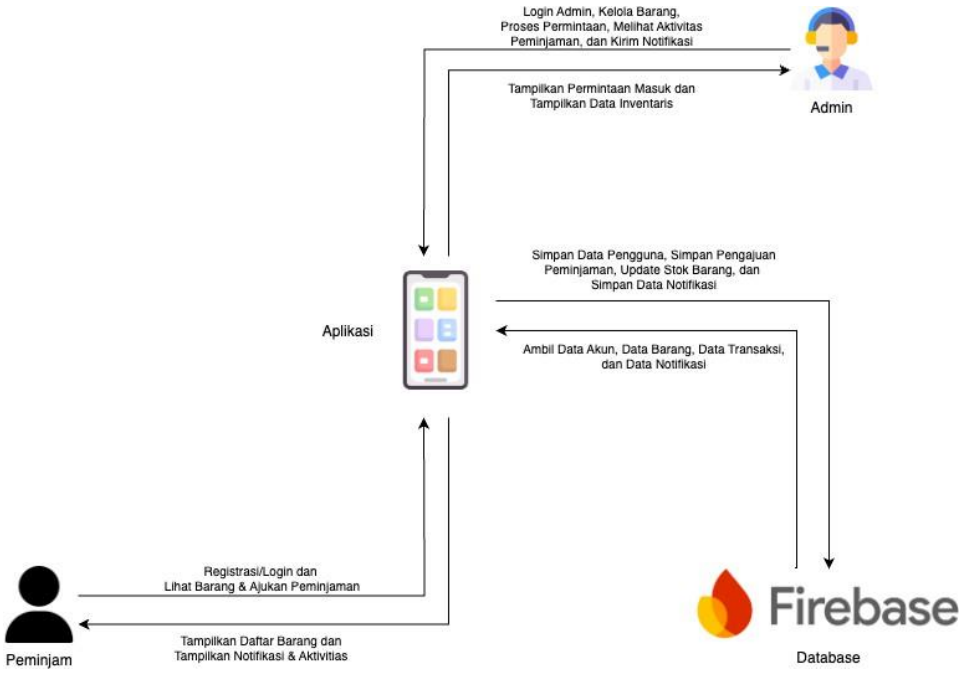
1. Sistem Operasi (untuk Pengembangan): macOS.
2. IDE: Visual Studio Code.
3. Bahasa Pemrograman: Dart.
4. *Framework* Aplikasi *Mobile*: Flutter SDK.
5. *Database Backend*: Firebase.
6. *Version Control System*: Git.
7. *Image Hosting Service*: Imgur API.
8. *Emulator*: via Xcode di macOS (untuk pengujian pada lingkungan iOS) dan via Android Studio (untuk pengujian pada lingkungan Android).
9. Web *Browser*: Google Chrome.
10. *Tools* Pendukung Flutter: Flutter Doctor.
    - *State Management*: provider.
    - Interaksi Perangkat dan Fitur: image_picker.
    - *Networking*: http.
    - Utilitas UI dan Data: intl, cached_network_image, dan path_provider.

3.4 Desain dan Perancangan Sistem

Bagian ini membahas perancangan sistem informasi yang telah dikembangkan, yang mencakup alur kerja sistem secara keseluruhan serta rancangan interaksi antara pengguna dan sistem melalui pemodelan diagram. Perancangan ini dilakukan agar implementasi dapat terstruktur, terarah, dan sesuai dengan kebutuhan pengguna yang telah dianalisis sebelumnya.

3.4.1 Gambaran Umum Sistem

Gambaran umum sistem disusun untuk memberikan representasi awal mengenai proses bisnis yang dijalankan oleh sistem informasi inventaris peminjaman ini. Pada Gambar 3.2 ditampilkan alur interaksi antar komponen utama, yaitu pengguna (peminjam dan admin), aplikasi *mobile*, dan layanan *backend* Firebase yang berperan sebagai pusat data.



Gambar 3.2 Alur Gambaran Umum Sistem

Peminjam berinteraksi langsung dengan aplikasi untuk melakukan registrasi, *login*, melihat daftar barang, dan mengajukan permintaan peminjaman. Setelah itu, pengguna dapat menerima notifikasi dari sistem serta memantau status dan riwayat transaksi mereka. Sementara itu, admin menggunakan aplikasi untuk *login*, melihat dan mengelola daftar barang, memproses permintaan peminjam, melihat aktivitas peminjaman, dan mengirim notifikasi pengingat pengembalian & pemberitahuan keterlambatan kepada peminjam.

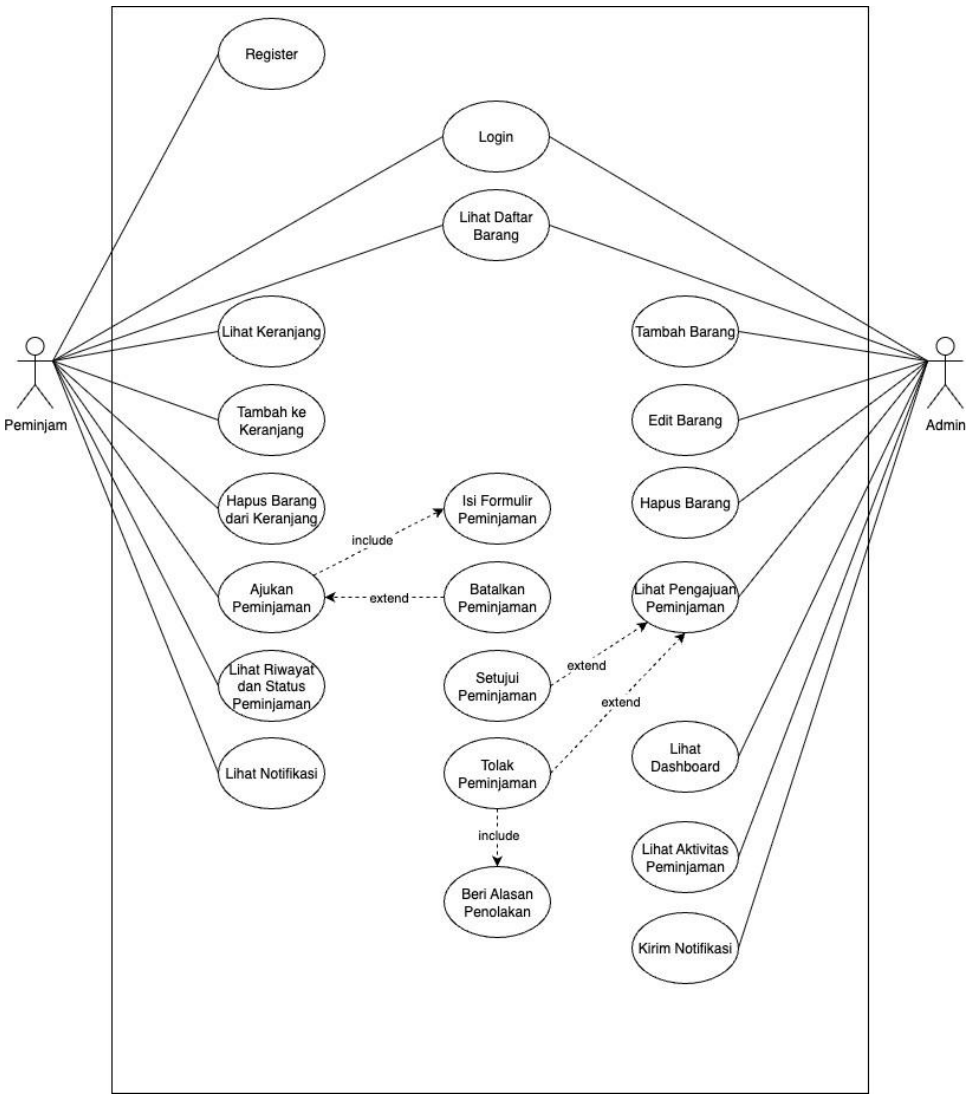
Aplikasi menjadi perantara utama antara pengguna dengan *database* Firebase. Aplikasi bertugas mengirim mengirim data dari pengguna ke *backend* (seperti data akun, permintaan peminjaman, dan perubahan stok), serta mengambil data untuk ditampilkan kembali dalam antarmuka pengguna. Firebase menyimpan seluruh data akun, barang, transaksi, dan notifikasi, serta mendukung sinkronisasi secara *real-time*.

3.4.2 *Use Case Diagram*

Use case diagram menggambarkan hubungan antara aktor (peminjam dan admin) dengan sistem serta fitur-fitur utama yang dapat diakses oleh masing-masing peran. Diagram ini penting untuk menunjukkan cakupan sistem dari sudut pandang pengguna serta batasan interaksi yang dimiliki setiap aktor.

Dalam penelitian ini, use case diagram dirancang untuk mendeskripsikan alur umum interaksi pengguna dengan sistem, mulai dari *login*, pengelolaan keranjang, pengajuan peminjaman, hingga proses persetujuan dan pengembalian barang. Diagram ini juga mencakup

fitur-fitur manajemen yang hanya dapat diakses oleh admin seperti menambah barang, mengelola transaksi, dan mengirim notifikasi. Ilustrasi *use case* ditampilkan pada Gambar 3.3.



Gambar 3.3 *Use Case Diagram*

Berdasarkan Gambar 3.3, terdapat dua aktor utama dalam sistem yaitu peminjam dan admin. Tabel 3.4 berikut menjelaskan rincian hubungan antara masing-masing aktor dan fitur yang digunakan

Tabel 3.4 *Use Case* Aktor

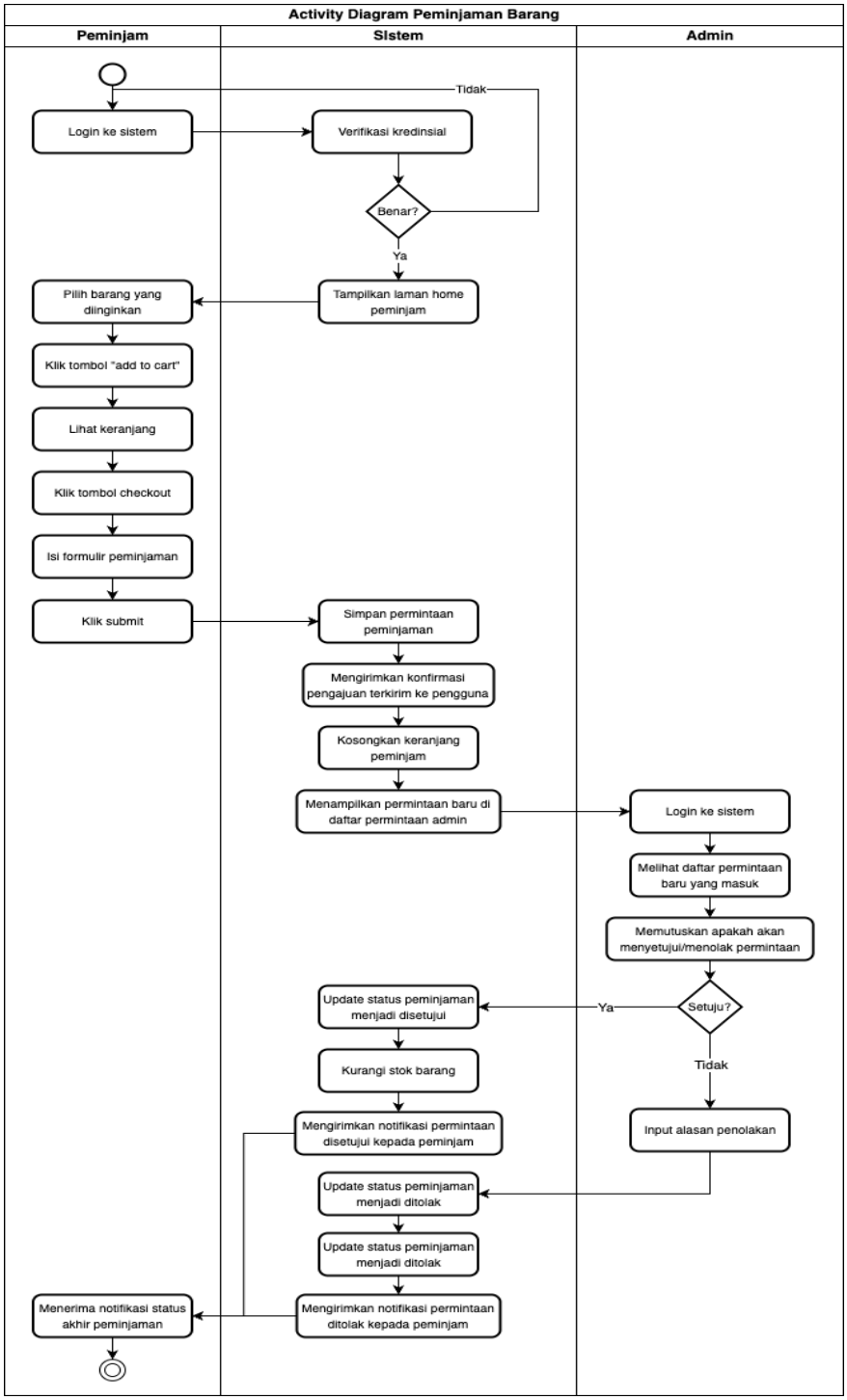
No	Aktor	*Use Case*	Deskripsi
1	Admin	*Login*	Admin melakukan proses autentikasi untuk masuk ke dalam sistem dengan memasukkan nama pengguna dan kata sandi. Setelah berhasil *login*, admin diarahkan ke *dashboard*.
		Lihat Daftar Barang	Admin mengakses menu inventaris untuk melihat daftar barang beserta informasi seperti nama, kategori, jumlah stok barang.
		Tambah Barang	Admin menambahkan barang baru ke dalam *database* inventaris dengan detail seperti nama, kategori, jumlah stok, deskripsi, dan gambar barang.
		Edit Barang	Admin dapat mengubah data barang yang telah terdaftar termasuk nama, kategori, jumlah stok, deskripsi, dan gambar barang.
		Hapus Barang	Admin menghapus data barang dari sistem apabila barang tidak lagi tersedia atau tidak digunakan.
		Lihat Pengajuan Peminjaman	Admin melihat daftar pengajuan peminjaman yang masuk dari peminjam, termasuk detail barang, peminjam, dan periode waktu peminjaman.
		Setujui Peminjaman	Admin menyetujui pengajuan peminjaman setelah memverifikasi detail permintaan dan ketersediaan barang.
		Tolak Peminjaman	Admin menolak pengajuan peminjaman apabila tidak sesuai atau barang tidak tersedia.
		Beri Alasan Penolakan	Admin memberikan alasan spesifik ketika menolak permintaan peminjaman agar dapat dikomunikasikan ke peminjam.
		Lihat Aktivitas Peminjaman	Admin memantau dan meninjau riwayat transaksi peminjaman dari semua peminjam.
		Kirim Notifikasi	Admin mengirimkan notifikasi ke peminjam, termasuk notifikasi pengingat pengembalian yang mendekati batas waktu dan peringatan bahwa barang sudah telat untuk dikembalikan (melebihi batas waktu periode yang sudah ditentukan sebelumnya).

		Lihat *Dashboard*	Admin melihat data statistik seperti total peminjaman, kategori dan barang yang paling sering dipinjam, total user aktif dan user yang baru mendaftar, dan prediksi permintaan selama 3 bulan ke depan.
2	Peminjam	Registrasi	Peminjam mendaftarkan akun baru dengan mengisi data pribadi seperti nama lengkap, email, dan *password*.
		Login	Peminjam masuk ke sistem menggunakan akun terdaftar.
		Lihat Daftar Barang	Peminjam melihat daftar barang yang tersedia beserta deskripsi dan stok barang.
		Lihat Keranjang	Peminjam melihat barang-barang yang dimasukkan ke dalam keranjang sebelum mengajukan peminjaman.
		Tambah ke Keranjang	Peminjam menambahkan barang yang diinginkan ke dalam keranjang sebelum mengisi formulir peminjaman.
		Hapus Barang dari Keranjang	Peminjam menghapus barang dari keranjang apabila tidak jadi dipinjam.
		Isi Formulir Peminjaman	Peminjam mengisi data peminjaman seperti data diri, periode peminjaman, alasan peminjaman, dan foto identitas.
		Ajukan Peminjaman	Peminjam mengirim permintaan pengiriman ke sistem setelah formulir lengkap.
		Batalkan Peminjaman	Peminjam membatalkan pengajuan yang masih dalam proses atau belum disetujui.
		Lihat Riwayat dan Status Peminjaman	Peminjam melihat daftar transaksi terdahulu dan status terkini dari permintaan peminjaman.
		Lihat Notifikasi	Peminjam melihat notifikasi terkait status pengajuan, pengingat, dan pesan dari admin.

3.4.3 *Activity Diagram*

a. Proses Peminjaman Barang

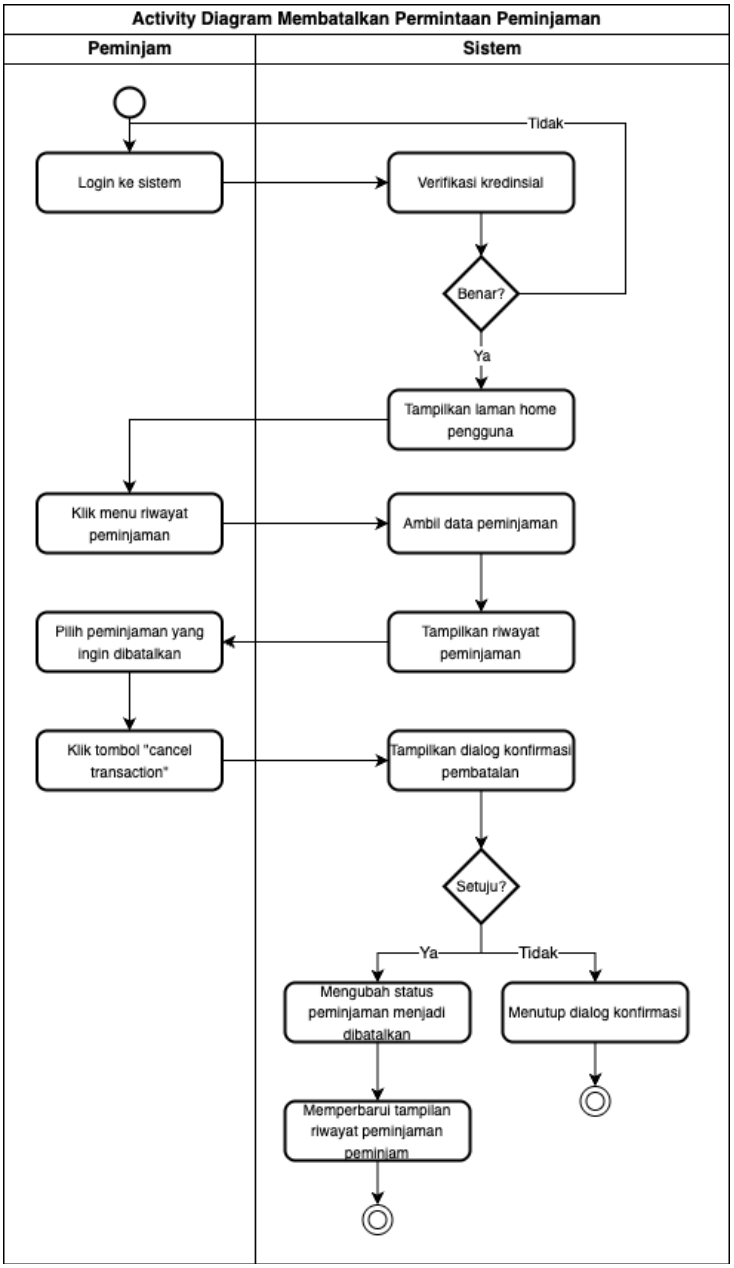
Activity diagram ini memperlihatkan urutan aktivitas dalam proses peminjaman barang yang dilakukan oleh peminjam hingga admin memberikan respon terhadap permintaan tersebut. Seluruh proses ini divisualisasikan secara dalam Gambar 3.4.



Gambar 3.4 *Activity Diagram* Proses Peminjaman Barang

b. Proses Pembatalan Permintaan Peminjaman Barang

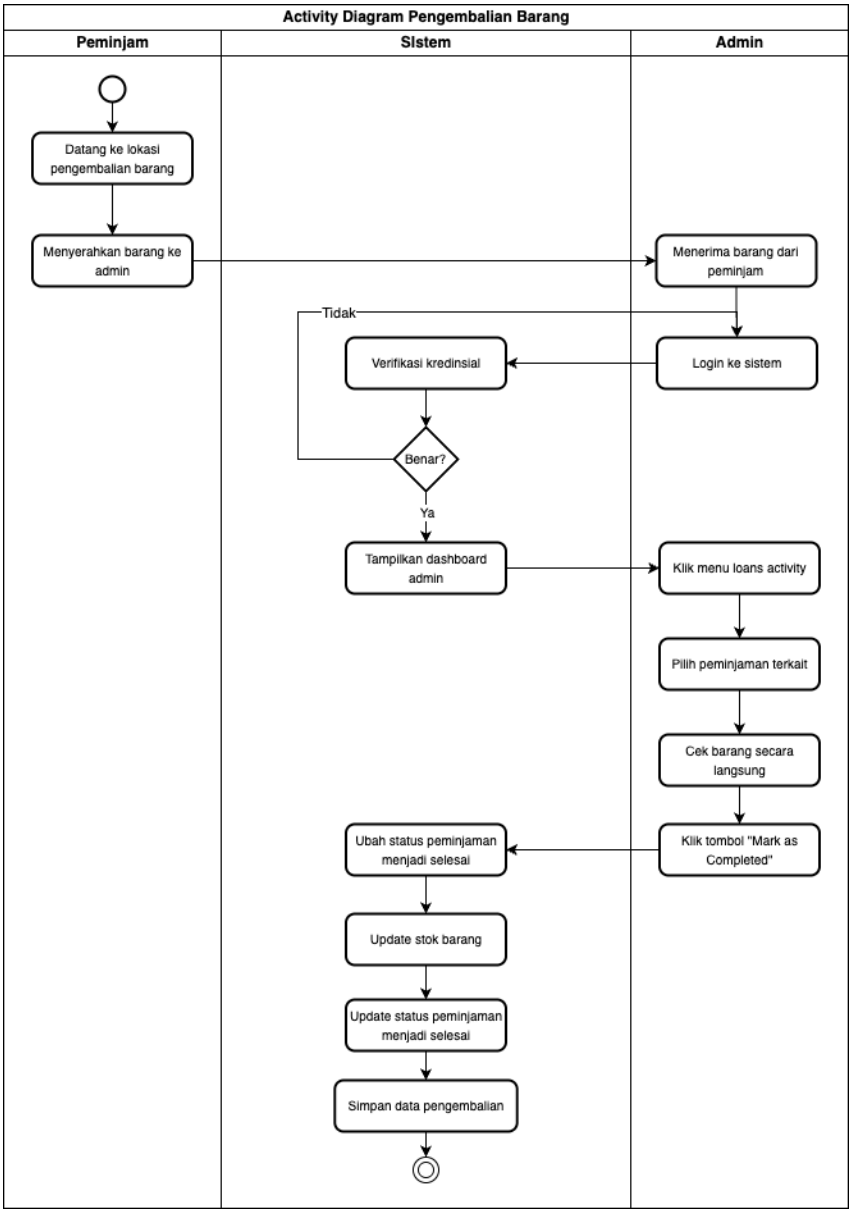
Activity diagram ini menjelaskan tahapan aktivitas yang dilakukan oleh peminjam ketika ingin membatalkan permintaan peminjaman yang masih dalam tahap status menunggu persetujuan. Proses ini digambarkan secara visual dalam Gambar 3.5



Gambar 3.5 *Activity Diagram* Proses Pembatalan

c. Proses Pengembalian Barang

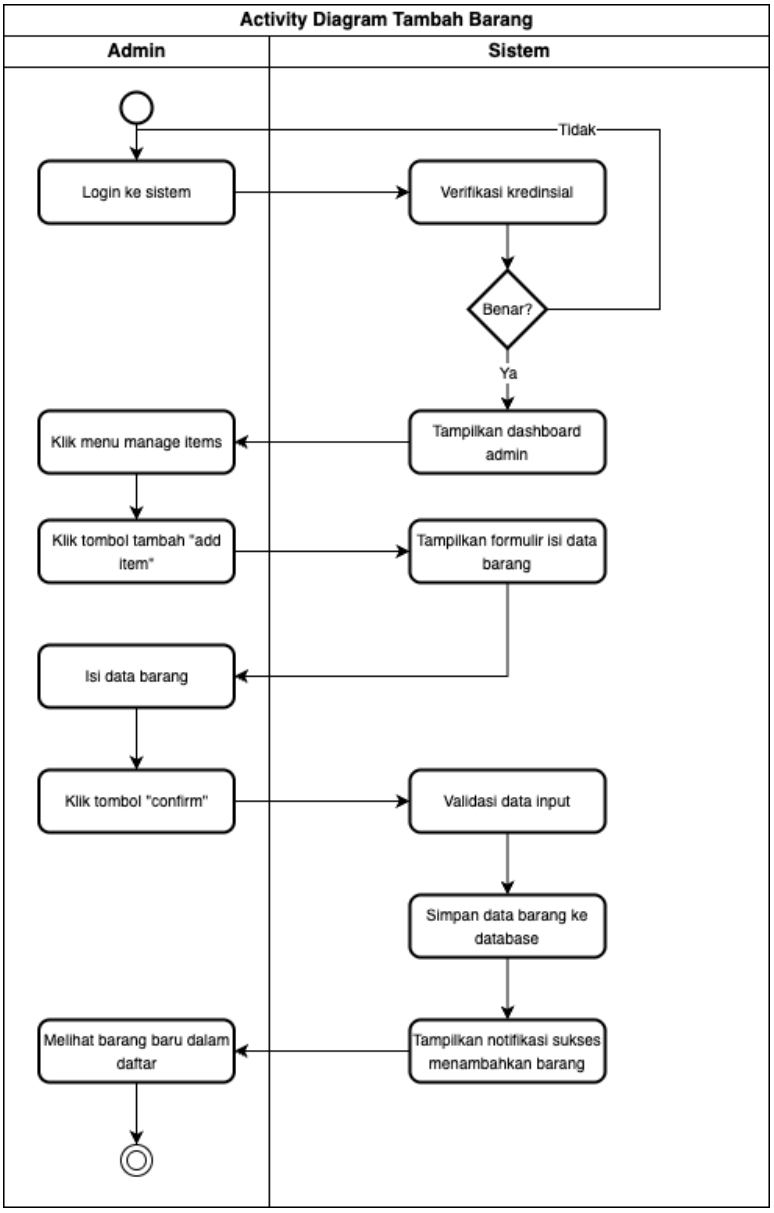
Activity diagram ini memperlihatkan urutas aktivitas dalam proses pengembalian barang oleh peminjam yang telah menyelesaikan masa peminjaman. Rangkaian seluruh aktivitas ini divisiualisasikan dalam Gambar 3.6.



Gambar 3.6 *Activity Diagram* Proses Pengembalian Barang

d. Proses Tambah Barang

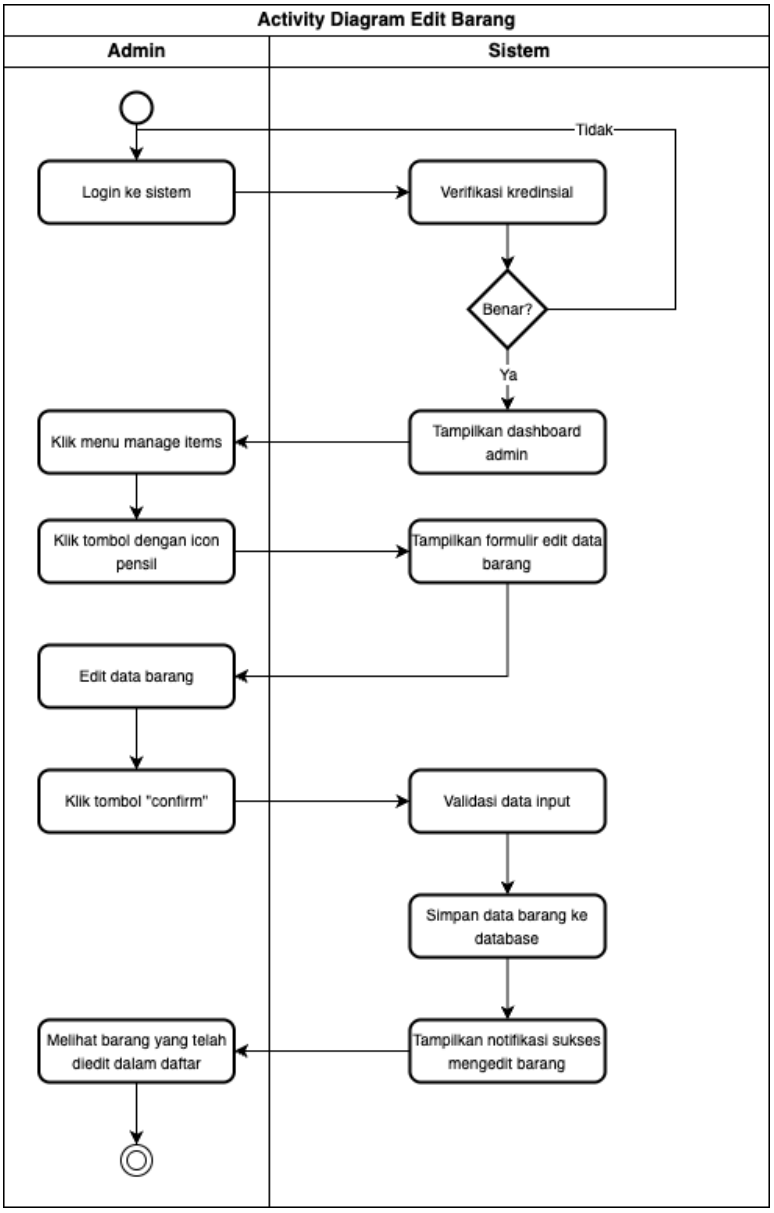
Activity diagram ini menggambarkan proses yang dilakukan oleh admin untuk menambahkan data barang baru ke dalam sistem. Fitur ini dirancang untuk untuk memastikan bahwa setiap barang yang ditambahkan memiliki informasi yang lengkap dan akurat agar dapat dikelola dengan baik dalam sistem inventaris. Visualisasi detail aktivitas dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Activity Diagram* Proses Penambahan Barang

e. Proses Edit Barang

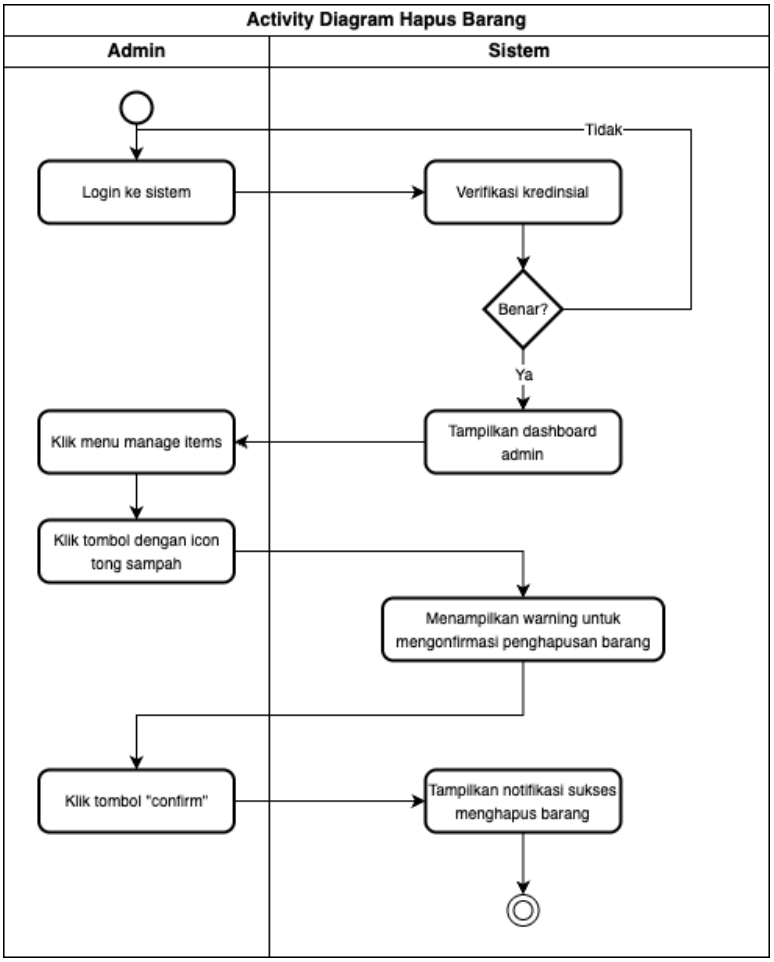
Activity diagram ini menjelaskan tahapan yang dilakukan oleh admin dalam memperbarui data barang yang telah tersimpan di dalam sistem. Rangkaian seluruh aktivitas divisualisasikan dalam Gambar 3.8.



Gambar 3.8 *Activity Diagram* Proses Pengeditan Barang

f. Proses Hapus Barang

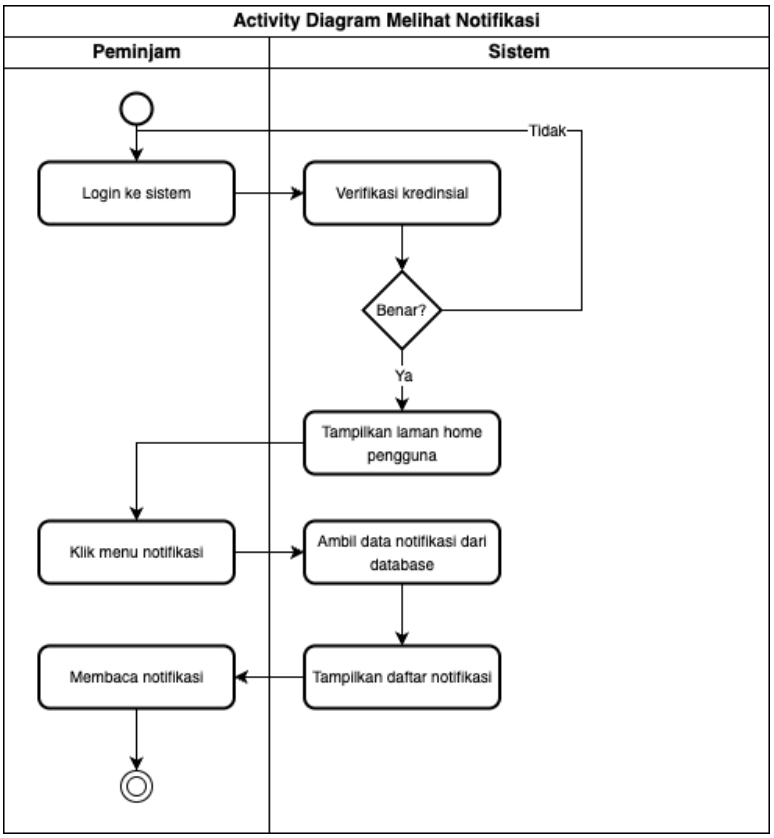
Activity diagram ini menjelaskan alur aktivitas ketika admin ingin menghapus barang dari daftar, hingga munculnya konfirmasi penghapusan. Diagram lengkap proses ini ditampilkan pada Gambar 3.9.



Gambar 3.9 *Activity Diagram* Proses Hapus Barang

g. Proses Lihat Notifikasi

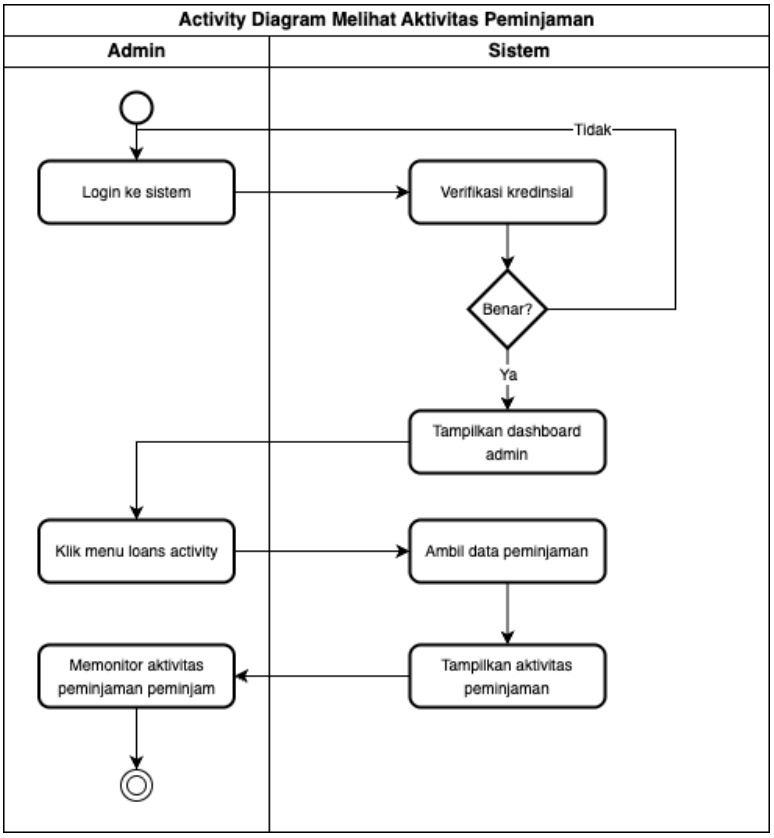
Activity diagram ini menggambarkan urutan proses yang dilakukan oleh peminjam ketika mengakses fitur notifikasi dalam aplikasi. Proses ini bertujuan untuk memastikan pengguna selalu mendapatkan informasi terkini terkait aktivitas peminjamannya. Visualisasi alur aktivitas ini ditampilkan pada Gambar 3.10.



Gambar 3.10 *Activity Diagram* Proses Lihat Notifikasi

h. Proses Lihat Aktivitas Peminjaman

Activity diagram ini memperlihatkan alur aktivitas admin saat mengakses halaman "*Loans Activity*" untuk memantau semua transaksi peminjaman. Visualisasi proses ini disajikan dalam Gambar 3.11.

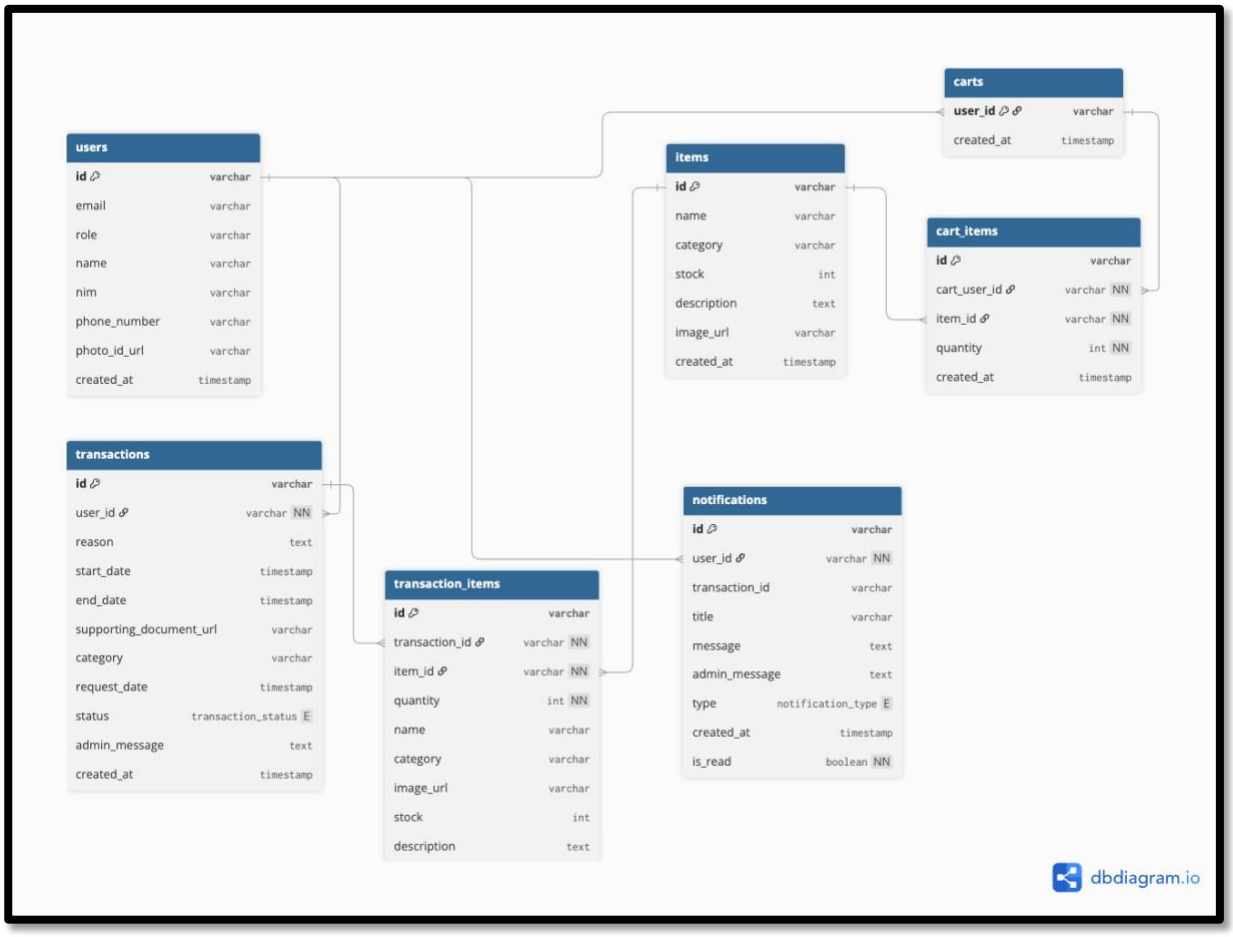


Gambar 3.11 *Activity Diagram* Proses Lihat Aktivitas Peminjaman

3.4.4 Relasi Tabel *Database*

Relasi tabel *database* digunakan untuk menggambarkan struktur basis data secara lebih teknis dengan menampilkan tabel-tabel utama beserta atribut dan hubungan antar tabel. Relasi ini menunjukkan bagaimana data pengguna, barang, transaksi, keranjang, dan notifikasi saling terhubung melalui *primary key* (PK) dan *foreign key* (FK). Dengan pemodelan ini, keterhubungan antar data dapat dipastikan konsisten dan mendukung integritas sistem.

Dalam penelitian ini, relasi tabel melibatkan beberapa entitas inti, yaitu Users, Items, Transactions, Transaction_Items, Carts, Cart_Items, dan Notifications. Masing-masing entitas memiliki atribut penting yang berfungsi dalam pencatatan data, sedangkan relasi antar entitas menunjukkan bagaimana data saling terhubung. Gambar 3.12 berikut memperlihatkan struktur tabel dan relasi yang digunakan dalam perancangan basis data aplikasi.



Gambar 3.12 Relasi Tabel *Database*

Penjelasan relasi antar kelas:

a. Tabel Users menyimpan informasi dasar pengguna dengan atribut seperti id, nama, email, nim, dan role. Satu pengguna dapat memiliki banyak transaksi (relasi *one-to-many* ke Transactions), serta menerima banyak notifikasi (relasi *one-to-many* ke Notifications).
b. Tabel Items menyimpan data inventaris berupa barang yang dapat dipinjam. Barang ini dapat direlasikan ke banyak transaksi melalui Transaction_Items, serta ke banyak keranjang melalui Cart_Items.
c. Tabel Transactions merepresentasikan proses peminjaman barang oleh pengguna. Setiap transaksi dapat terdiri dari beberapa barang, yang direpresentasikan pada Transaction_Items (relasi *one-to-many*).
d. Tabel Transaction_Items berfungsi sebagai tabel penghubung antara Transactions dan Items, mencatat detail barang yang dipinjam beserta jumlahnya.
e. Tabel Carts adalah keranjang sementara untuk menyimpan barang yang dipilih oleh pengguna sebelum diajukan menjadi transaksi. Setiap pengguna hanya memiliki satu keranjang (relasi *one-to-one* dengan Users).
f. Tabel Cart_Items berisi daftar barang yang ditambahkan ke dalam keranjang. Relasi *one-to-many* dengan Items memungkinkan satu barang dipilih berkali-kali di berbagai keranjang pengguna.
g. Tabel Notifications menyimpan notifikasi terkait status transaksi, pengingat pengembalian, maupun pesan admin. Tabel ini berelasi dengan Users (*one-to-many*), serta dapat dikaitkan ke transaksi tertentu.

3.4.5 Proses Bisnis Sistem

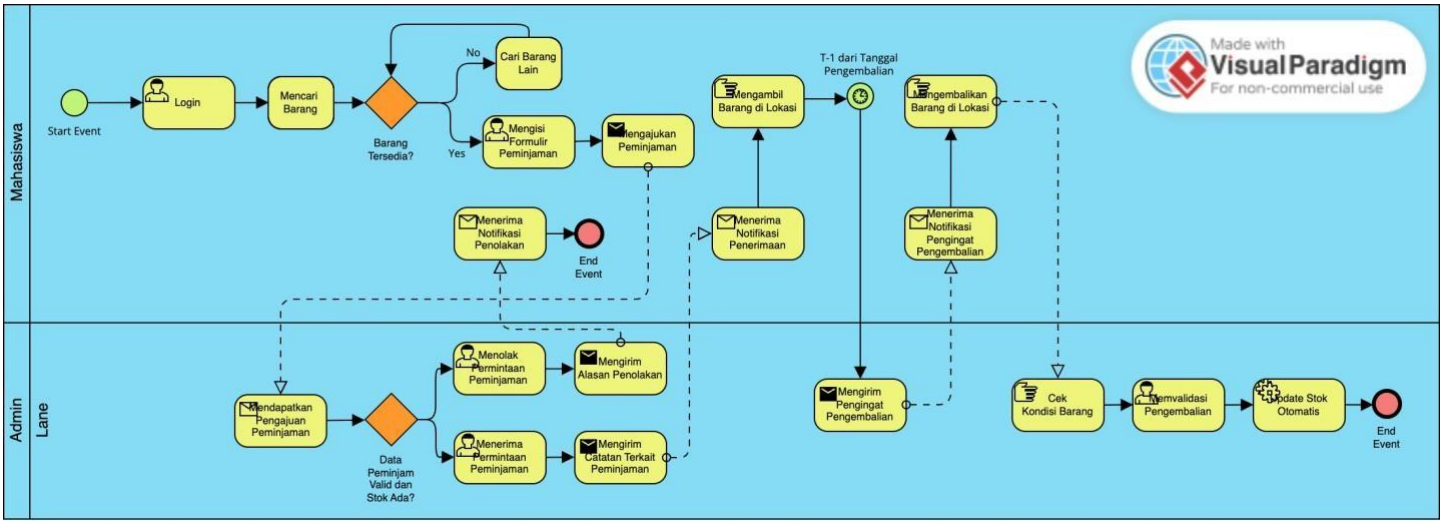
Proses bisnis dalam sistem informasi inventaris peminjaman barang ini dimodelkan menggunakan *Business Process Model and Notation* (BPMN). BPMN dipilih karena mampu menggambarkan alur proses secara terstruktur, mulai dari aktivitas pengguna hingga aktivitas admin.

Alur proses dimulai dari sisi mahasiswa yang melakukan *login*, mencari barang, dan mengajukan peminjaman. Apabila barang tersedia, mahasiswa mengisi formulir dan mengirim permintaan. Sebaliknya, jika barang tidak tersedia, mahasiswa diarahkan untuk mencari barang lain. Setelah permintaan dikirim, sistem akan meneruskan informasi ke admin untuk dilakukan verifikasi.

Pada sisi admin, pengajuan peminjaman yang masuk diverifikasi berdasarkan validitas data peminjam serta ketersediaan stok. Jika permintaan ditolak, admin mengirimkan alasan penolakan dan mahasiswa menerima notifikasi penolakan. Jika permintaan diterima, admin mengirimkan catatan terkait peminjaman, mahasiswa memperoleh notifikasi persetujuan, dan barang dapat diambil di lokasi.

Setelah barang diambil, sistem mengatur *timer event* pada H-1 sebelum tanggal pengembalian untuk memicu pengiriman notifikasi pengingat kepada mahasiswa. Mahasiswa yang menerima notifikasi ini diharapkan mengembalikan barang sesuai jadwal. Jika barang dikembalikan, admin kemudian melakukan pemeriksaan kondisi barang.

Tahap selanjutnya adalah validasi pengembalian oleh admin. Jika barang telah sesuai, maka sistem secara otomatis memperbarui jumlah stok yang tersedia di basis data. Dengan demikian, proses bisnis berakhir pada kondisi barang kembali tercatat, stok diperbarui, dan transaksi peminjaman selesai. Gambar 3.13 berikut menunjukkan pemodelan BPMN proses bisnis dalam sistem.



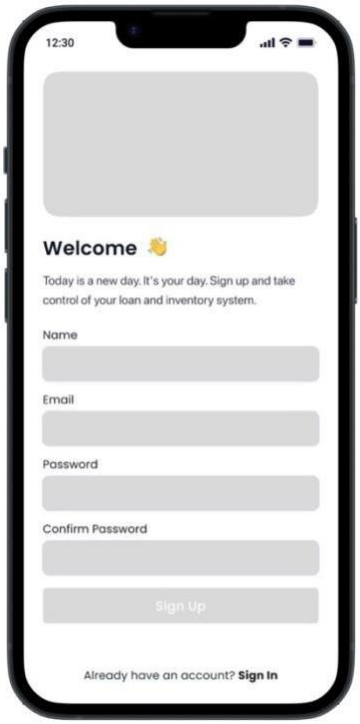
Gambar 3.13 *Business Process Model and Notation* (BPMN)

3.4.6 Perancangan Desain Antarmuka

Perancangan antarmuka pengguna (*user interface*) dalam penelitian ini menggunakan rancangan *wireframe* yang disusun melalui aplikasi desain Figma. *Wireframe* ini bertujuan untuk memberikan gambaran awal mengenai struktur tampilan dan alur penggunaan aplikasi. Gambar-gambar berikut menunjukkan rancangan *wireframe* yang dijadikan acuan dalam proses implementasi sistem.

a. Rancangan Desain Antarmuka Halaman Registrasi

Tata letak desain antarmuka untuk halaman registrasi dalam sistem ini dirancang khusus untuk memfasilitasi pembuatan akun baru untuk peminjam. Halaman ini menyediakan empat *input field* utama yang harus diisi oleh calon pengguna.

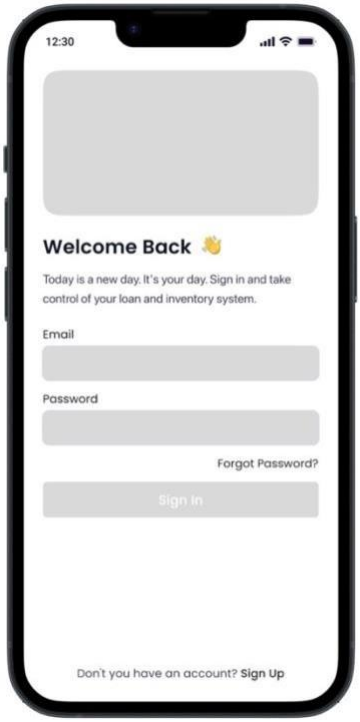


Gambar 3.14 Desain Antarmuka Layar Registrasi

Pada rancangan ini, peminjam baru diharapkan untuk melengkapi kolom-kolom yang tersedia dengan informasi berupa nama pengguna, alamat email, *password*, dan konfirmasi *password* yang diinginkan untuk mendaftarkan akun mereka ke dalam sistem.

b. Perancangan Desain Antarmuka Halaman *Login*

Rancangan desain antarmuka halaman *login* user ini menampilkan tata letak halaman masuk untuk pengguna. Pada halaman ini, pengguna akan menemukan dua *input field* utama.

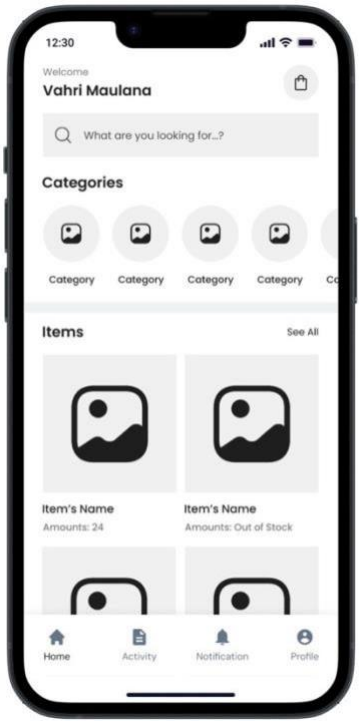


Gambar 3.15 Desain Antarmuka Halaman *Login*

Pada rancangan ini, pengguna diharapkan dapat menginput email dan *password*. Setelah itu, pengguna dapat menekan tombol "*Sign In*" untuk masuk halaman utama aplikasi.

c. Perancangan Desain Antarmuka Halaman *Home* Peminjam

Rancangan desain antarmuka halaman *home* peminjam ini menampilkan dasbor utama yang akan dilihat oleh pengguna dengan peran peminjam setelah berhasil masuk ke dalam sistem.

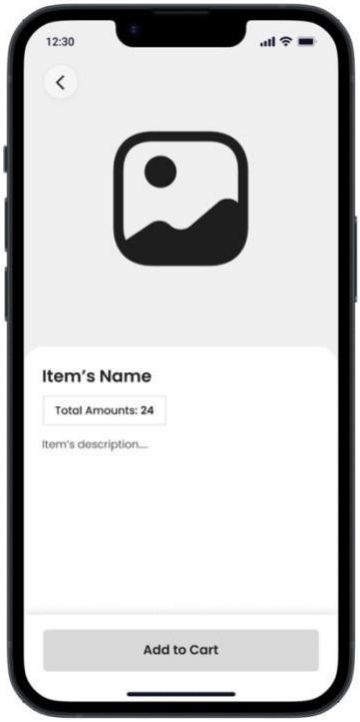


Gambar 3.16 Desain Antarmuka Halaman *Home* Peminjam

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) tombol *icon* keranjang di bagian atas untuk mengakses daftar barang peminjaman, (2) *search dan filter* kategori guna mempermudah pencarian barang, (3) daftar barang yang menampilkan gambar, nama, dan stok dalam bentuk *grid*, serta (4) *bottom navbar* yang berisi halaman *home*, *activity*, *notification*, dan *profile* untuk memandu pengguna secara intuitif.

d. Perancangan Desain Antarmuka Halaman Detail Barang

Rancangan desain antarmuka halaman detail barang menyajikan informasi lengkap tentang barang yang dipilih oleh peminjam.

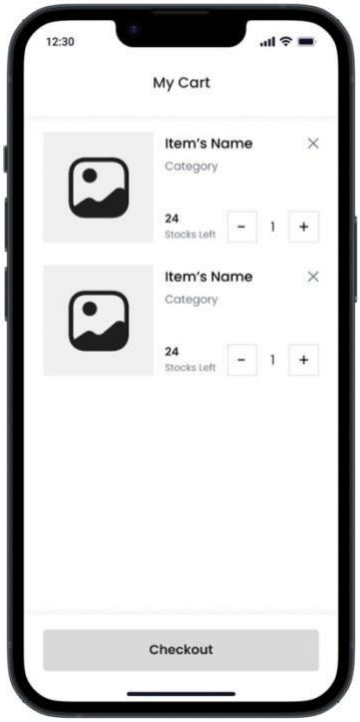


Gambar 3.17 Desain Antarmuka Halaman Detail Barang

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) *header* yang berisi tombol kembali, (2) konten utama yang menampilkan gambar, nama, stok, dan deskripsi barang, serta (3) tombol "*Add to Cart*" untuk menambahkan barang ke keranjang peminjam.

e. Perancangan Desain Antarmuka Halaman *Cart* Peminjam

Rancangan desain antarmuka halaman *cart* peminjam menampilkan daftar barang yang telah dipilih oleh peminjam untuk dipinjam beserta informasi terkait.

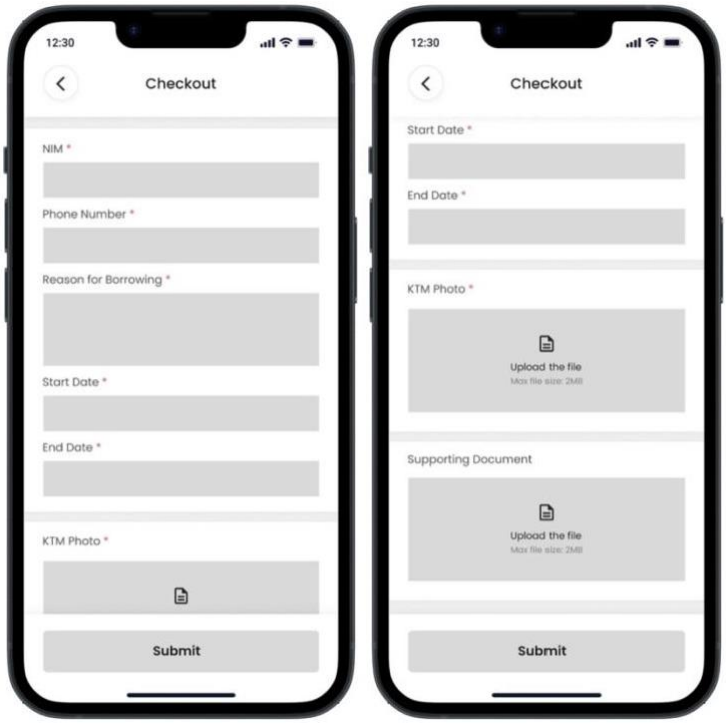


Gambar 3.18 Desain Antarmuka Halaman *Cart* Peminjam

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) *header* yang berisi judul "*My Cart*", (2) daftar barang yang menampilkan gambar, nama, kategori, dan stok tersedia dari setiap barang, (3) tombol *quantity adjustment* untuk mengubah jumlah peminjaman, serta (4) tombol "*Checkout*" sebagai *call-to-action* utama untuk melanjutkan proses peminjaman.

f. Perancangan Desain Antarmuka Halaman *Checkout* Peminjam

Rancangan desain antarmuka halaman *checkout* peminjam memfasilitasi proses akhir peminjaman barang oleh peminjam.

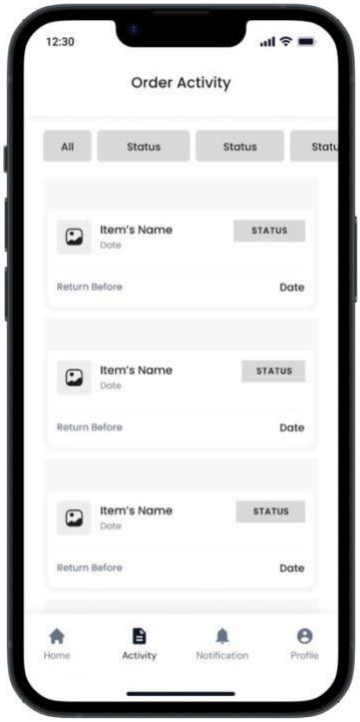


Gambar 3.19 Desain Antarmuka Halaman *Checkout* Peminjam

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) *form input* yang berisi *field* wajib seperti NIM, nomor telepon, alasan peminjaman, upload foto KTM, dan *upload* dokumen pendukung yang bersifat opsional, dan (2) tombol *submit* sebagai *call-to-action* utama untuk menyelesaikan proses *checkout*.

g. Perancangan Desain Antarmuka Halaman *Order Activity* Peminjam

Rancangan desain antarmuka halaman o*rder activity* peminjam menampilkan riwayat dan status transaksi peminjaman barang oleh peminjam.

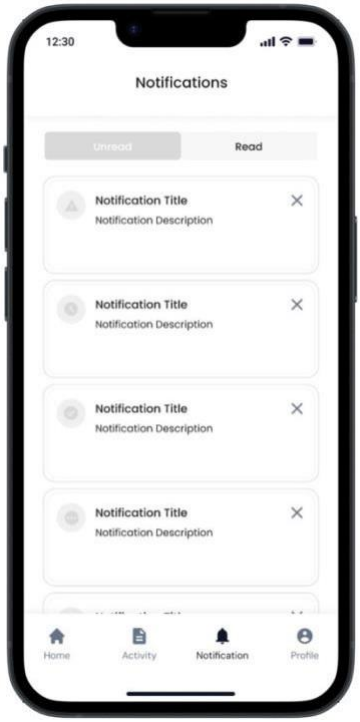


Gambar 3.20 Desain Antarmuka Halaman *Order Activity* Peminjam

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) filter status peminjaman, (2) daftar transaksi yang menampilkan nama barang, tanggal peminjaman, status persetujuan, serta tanggal pengembalian yang harus dipenuhi, dan (4) *bottom navbar* yang berisi halaman *home*, *activity*, *notification*, dan *profile* untuk memandu pengguna secara intuitif.

h. Perancangan Desain Antarmuka Halaman *Notification* Peminjam

Rancangan desain antarmuka halaman *notification* peminjam menyajikan berbagai pemberitahuan terkait aktivitas peminjaman kepada peminjam.

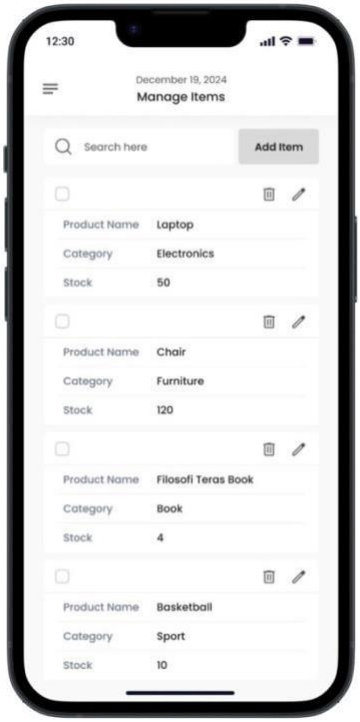


Gambar 3.21 Desain Antarmuka Halaman *Notification* Peminjam

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) segmentasi notifikasi menjadi *unread* dan *read* untuk memudahkan pengelolaan informasi, (2) daftar notifikasi yang menampilkan berbagai jenis pemberitahuan seperti *Late Return Notice*, *Return Reminder*, *Order Accepted*, *Order Completed*, dan *Order Rejected*, serta (3) *bottom navbar* yang berisi halaman *home*, *activity*, *notification*, dan *profile* untuk memandu pengguna secara intuitif.

i. Perancangan Desain Antarmuka Halaman *Manage Items* Admin

Rancangan desain antarmuka halaman *manage items* admin memungkinkan admin untuk mengelola daftar barang yang tersedia dalam sistem.

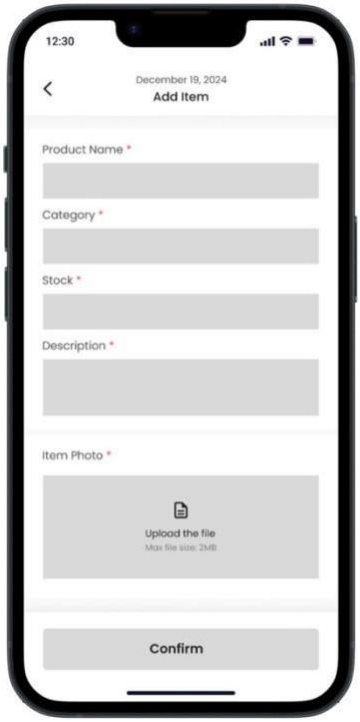


Gambar 3.22 Desain Antarmuka Halaman *Manage Items* Admin

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) *search bar* untuk memudahkan pencarian barang, (2) tombol "*Add Item*" untuk menambahkan barang baru ke sistem, dan (3) daftar barang yang menampilkan nama, kategori, dan stok produk. Setiap barang dalam dalam daftar dilengkapi dengan *icon* pensil untuk mengedit barang, dan *icon* tong sampah untuk menghapus barang.

j. Perancangan Desain Antarmuka Halaman *Add Item*

Rancangan desain antarmuka halaman *add item* memfasilitasi admin dalam menambahkan data barang baru ke dalam sistem.

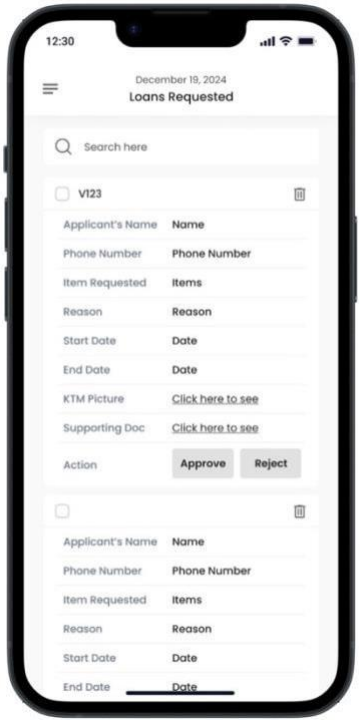


Gambar 3.23 Desain Antarmuka Halaman *Add Item*

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) *form input* yang berisi *field* wajib seperti nama, kategori, stok, deskripsi, dan foto barang, serta (2) tombol "*Confirm*" sebagai *call-to-action* untuk menyimpan data barang baru.

k. Perancangan Desain Antarmuka Halaman *Loans Requested* Admin

Rancangan desain antarmuka halaman *loans requested* admin memfasilitasi admin untuk memproses permohonan peminjaman barang dari peminjam.

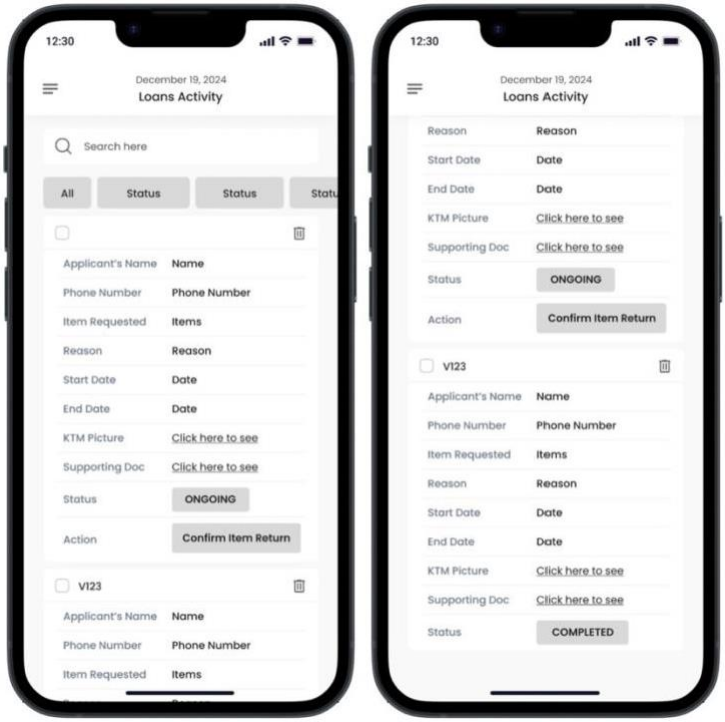


Gambar 3.24 Desain Antarmuka Halaman *Loans Requested* Admin

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) *search bar* untuk memfilter permintaan peminjaman, (2) daftar permohonan yang menampilkan informasi lengkap termasuk nama peminjam, nomor telepon, barang yang diajukan, alasan peminjaman, tanggal mulai dan akhir peminjaman, serta dokumen (KTM dan *supporting document*), dan (3) tombol aksi "*Approve*" dan "*Reject*" untuk setiap permohonan.

l. Perancangan Desain Antarmuka Halaman *Loans Activity* Admin

Rancangan desain antarmuka halaman *loans activity* admin memberikan gambaran menyeluruh tentang seluruh aktivitas peminjaman yang terjadi dalam sistem untuk keperluan monitoring.

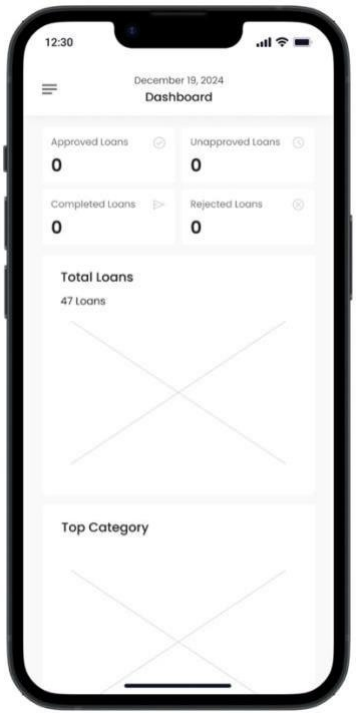


Gambar 3.25 Desain Antarmuka Halaman *Loans Activity* Admin

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) *search bar* di bagian atas untuk pencarian spesifik, (2) *tab* kategori untuk memfilter berdasarkan status peminjaman, serta (3) daftar lengkap permohonan peminjaman yang disajikan dalam format *card*. Setiap *card* menampilkan informasi detail meliputi: nama peminjam, nomor telepon peminjam, daftar barang yang dipinjam, alasan peminjaman, periode peminjaman, dokumen (KTM dan *supporting document*), status terkini, dan tombol aksi "*Confirm Item Return*" untuk peminjaman yang sedang berjalan.

m. Perancangan Desain Antarmuka Halaman *Dashboard* Admin

Rancangan desain antarmuka halaman *dashboard* admin menyajikan ringkasan statistik dan informasi penting terkait peminjaman barang untuk admin



Gambar 3.26 Desain Antarmuka Halaman *Dashboard* Admin

Rancangan antarmuka ini terdiri atas beberapa komponen utama, yaitu (1) statistik peminjaman yang mencakup *Approved Loans*, *Unapproved Loans*, *Completed Loans*, dan *Rejected Loans*, (2) grafik tren peminjaman bulanan, (3) daftar *top category* dan produk yang menampilkan kategori dan barang yang paling sering dipinjam, serta (4) informasi tanggal dan waktu terkini.

3.5 Metode Pengujian dan Evaluasi

3.5.1 Pengujian Fungsional (*Black-Box Testing*)

Pengujian fungsional dilakukan dengan menggunakan metode *Black-box Testing*. Tujuan pengujian ini adalah memastikan bahwa setiap fungsi utama pada aplikasi berjalan sesuai kebutuhan tanpa memperhatikan kode program. Cakupan pengujian meliputi autentikasi pengguna, proses CRUD (*Create*, *Read*, *Update*, *Delete*) data inventaris, mekanisme peminjaman melalui keranjang (*cart*) dan *checkout*, proses persetujuan atau penolakan peminjaman oleh admin, pembaruan stok inventaris, serta notifikasi kepada pengguna. Kriteria keberhasilan pengujian didasarkan pada apakah *output* yang dihasilkan sesuai dengan spesifikasi kebutuhan. Apabila semua fungsi menghasilkan keluaran sesuai ekspektasi, maka pengujian dinyatakan lulus. Sebaliknya, jika terdapat perbedaan atau kesalahan maka pengujian dinyatakan gagal. Dengan cara ini, kualitas fungsional aplikasi dapat divalidasi tanpa harus meninjau implementasi internal dari kode.

3.5.2 Pengujian Usability (*System Usability Testing* Versi Positif)

Dalam penelitian ini, instrumen yang digunakan untuk mengukur tingkat usability adalah *System Usability Scale* (SUS). Penelitian ini secara khusus menggunakan format SUS positif, di mana seluruh pernyataan disusun dalam bentuk positif. Pemilihan SUS positif didasarkan pada beberapa pertimbangan. Pertama, responden penelitian adalah mahasiswa yang berasal dari berbagai latar belakang, sehingga penggunaan pernyataan yang konsisten positif lebih mudah dipahami dan meminimalisasi kebingungan yang sering muncul akibat adanya *reverse statement* pada SUS standar. Kedua, penelitian-penelitian sebelumnya menunjukkan bahwa versi SUS positif tetap memiliki reliabilitas yang tinggi dan validitas yang setara dengan SUS standar, sekaligus dapat mengurangi kesalahan interpretasi jawaban (Sauro & Lewis, 2011). Ketiga, dengan menghilangkan pernyataan negatif, SUS positif mempercepat waktu pengisian kuesioner dan mengurangi beban kognitif responden, sehingga hasil data yang diperoleh lebih konsisten. Berdasarkan alasan tersebut, penggunaan SUS positif dianggap paling sesuai untuk konteks penelitian ini.

BAB IV
HASIL DAN PEMBAHASAN

4.1 Implementasi Antarmuka Pengguna

4.1.1 Tampilan *Splash Screen* Aplikasi

Halaman *splash screen* merupakan tampilan awal yang muncul ketika aplikasi dijalankan. Fungsi utama dari halaman ini adalah sebagai pengenal aplikasi dan penanda bahwa sistem sedang dimuat.

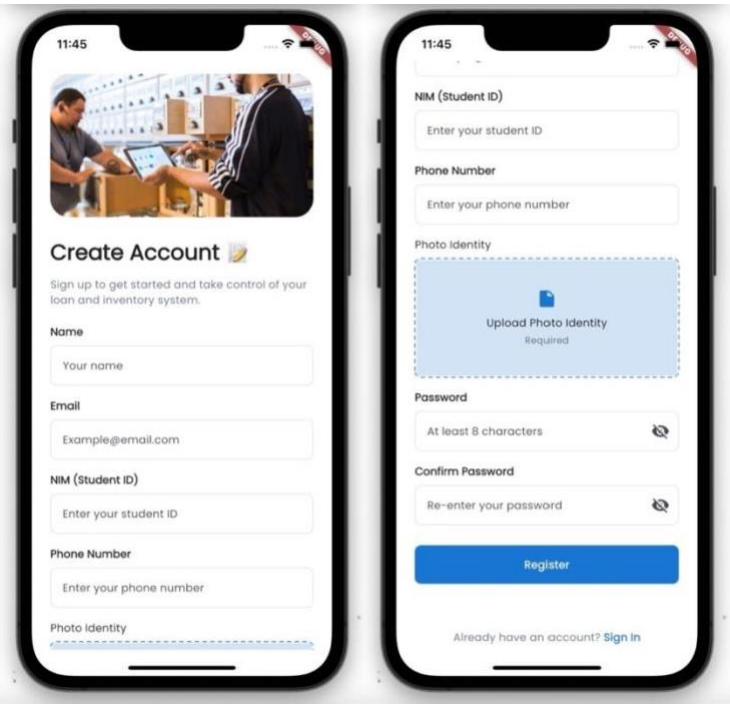


Gambar 4.1 Tampilan Halaman *Splash Screen* Aplikasi

Gambar 4.1 memperlihatkan antarmuka awal aplikasi yang menampilkan logo dan nomor versi. Setelah tahap ini selesai, pengguna akan diarahkan secara otomatis menuju halaman berikutnya, yaitu halaman *login* atau registrasi.

4.1.2 Tampilan Halaman Registrasi Pengguna

Antarmuka registrasi pengguna menyediakan *form* pendaftaran dengan empat *field input* berupa nama lengkap, email, password, dan konfirmasi password. Sistem akan memverifikasi kelengkapan dan validitas data sebelum memproses pembuatan akun baru. Tampilan halaman registrasi ditampilkan pada Gambar 4.2.

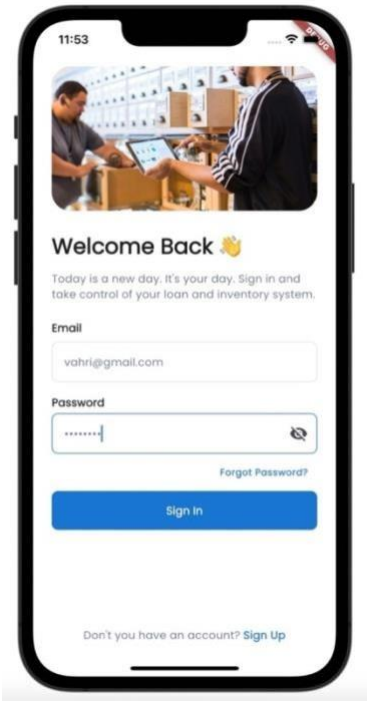


Gambar 4.2 Tampilan Halaman Registrasi Pengguna

Gambar 4.2 menampilkan formulir registrasi dari *input text* untuk data pribadi, *field* email dengan validasi format, NIM, nomor telepon, foto identitas, serta *dual field password* dengan pemeriksaan kesesuaian. Tombol *register* berfungsi untuk memproses pendaftaran setelah semua persyaratan terpenuhi, dilengkap dengan opsi navigasi ke halaman *login* bagi pengguna yang telah memiliki akun.

4.1.3 Tampilan Halaman *Login* Pengguna

Antarmuka *login* pengguna menyediakan *form* autentikasi dengan dua *field input* berupa email dan *password*. Sistem akan memverifikasi kredensial pengguna sebelum memberikan akses ke sistem.

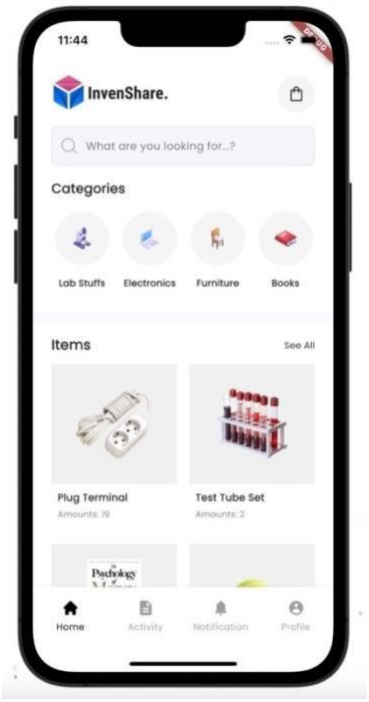


Gambar 4.3 Tampilan Halaman *Login* Pengguna

Gambar 4.3 menampilkan form *login* yang terdiri dari *input text* untuk email dan *password*, serta tombol *Sign In* untuk memproses autentikasi. Dilengkapi dengan navigasi ke halaman registrasi bagi pengguna yang belum memiliki akun.

4.1.4 Tampilan Halaman *Home* Peminjam

Antarmuka halaman *home* peminjam menyajikan *dashboard* utama dengan berbagai fitur navigasi dan informasi barang. Tampilan ini dirancang untuk memudahkan pengguna dalam menjelajahi sistem peminjaman. Tampilan halaman *home* peminjam ditampilkan pada Gambar 4.4.

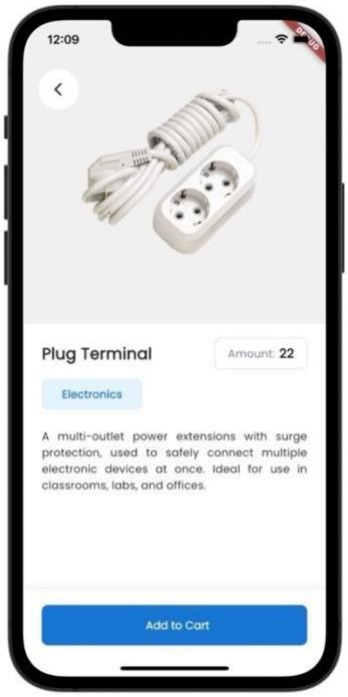


Gambar 4.4 Tampilan Halaman Utama *Home* Pengguna

Pada Gambar 4.4 menampilkan komponen: (1) *header* dengan logo aplikasi, (2) tombol keranjang di pojok kanan atas untuk mengakses daftar pinjaman, (3) *search bar* untuk pencarian barang, (4) filter kategori barang, (5) daftar item dalam bentuk *grid* dengan informasi stok, serta (6) *bottom navigation bar* berisi menu *Home*, *Activity*, *Notification*, dan *Profile*.

4.1.5 Tampilan Halaman Detail Barang

Antarmuka halaman detail barang menyajikan informasi lengkap mengenai spesifikasi dan ketersediaan barang yang akan dipinjam. Tampilan ini dirancang untuk memberikan gambaran komprehensif sebelum pengguna memutuskan untuk meminjam barang tersebut.

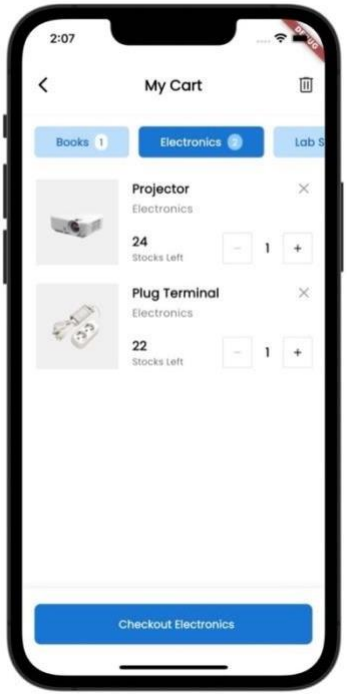


Gambar 4.5 Tampilan Halaman Detail Barang

Pada Gambar 4.5 ini menampilkan komponen utama berupa: (1) nama dan kategori barang, (2) deskripsi teknis yang menjelaskan fitur dan kegunaan barang, (3) informasi stok aktual yang diperbarui secara *real-time*, serta (4) tombol "*Add to Cart*" sebagai *call-to-action* utama untuk menambahkan barang ke keranjang.

4.1.6 Tampilan Halaman *Cart* Peminjam

Antarmuka halaman *cart* peminjam menyediakan ringkasan barang yang akan dipinjam beserta informasi stok terkini. Tampilan ini dirancang untuk memudahkan peminjam dalam mengelola daftar peminjaman sebelum proses *checkout*. Tampilan halaman *cart* peminjam ditampilkan pada Gambar 4.6.

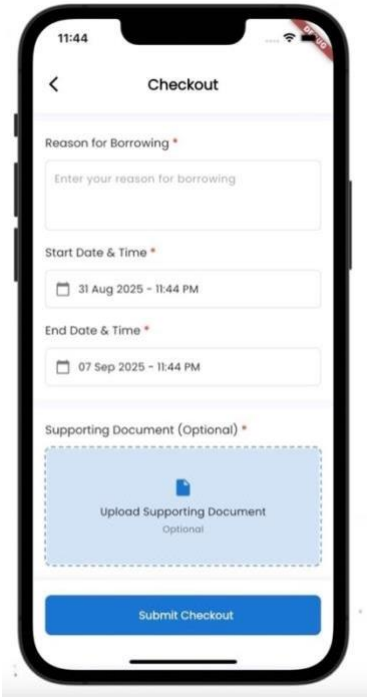


Gambar 4.6 Tampilan Halaman *Cart* Peminjam

Pada Gambar 4.6 ini menampilkan: (1) *header* "*My Cart*" dengan informasi jumlah barang per kategori, dengan pengguna hanya dapat memproses *checkout* untuk satu kategori dalam satu waktu, (2) daftar barang yang akan dipinjam dilengkapi dengan gambar, nama, kategori, dan stok tersedia, (3) tombol *adjust quantity* untuk mengubah jumlah peminjaman, (4) tombol [x] untuk menghapus barang tertentu, serta (5) tombol *checkout* untuk memproses peminjaman.

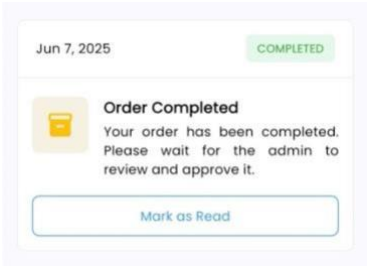
4.1.7 Tampilan Halaman *Checkout* Peminjam

Antarmuka halaman *checkout* peminjam dirancang sebagai proses bertahap yang memandu pengguna menyelesaikan transaksi peminjaman. Gambar 4.7 menampilkan formulir *checkout* yang memuat *field* seperti alasan peminjaman, periode peminjaman yang dilengkapi dengan *date picker, supporting document* yang bersifat opsional.



Gambar 4.7 Tampilan Halaman *Checkout* Peminjam

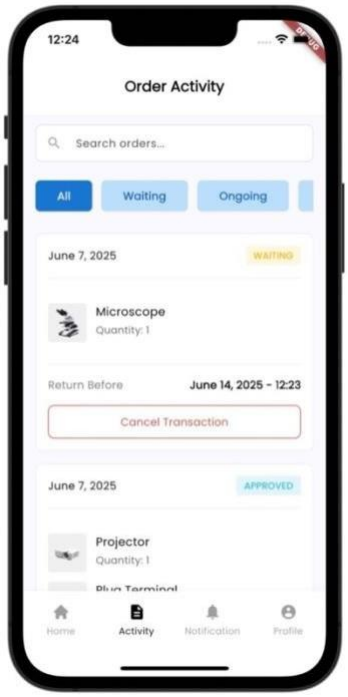
Proses *checkout* diakhiri dengan notifikasi otomatis seperti ditunjukkan pada Gambar 4.8, yang muncul di halaman *Notification* sebagai bukti transaksi.



Gambar 4.8 Notifikasi Sistem Pasca-*Checkout*

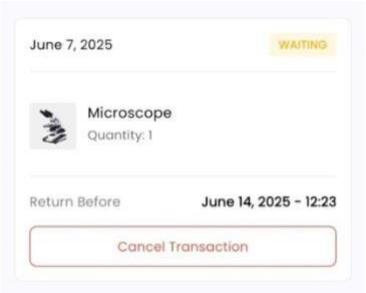
4.1.8 Tampilan Halaman *Order Activity* Peminjam

Antarmuka halaman *order activity* menyajikan ringkasan lengkap seluruh aktivitas peminjaman pengguna dalam berbagai status. Tampilan ini dirancang untuk memberikan pengelolaan riwayat transaksi yang komprehensif melalui antarmuka yang terorganisir.



Gambar 4.9 Tampilan Halaman Utama *Order Activity* Peminjam

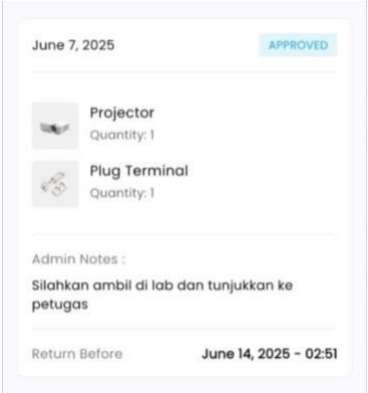
Gambar 4.9 menampilkan daftar kronologis transaksi yang dapat difilter melalui enam tab status (*All*, *Waiting*, *Ongoing*, *Completed*, *Rejected*, *Late*). Setiap entri transaksi ditampilkan dalam bentuk *card* yang konsisten dengan penanda status visual, mencakup informasi dasar seperti tanggal transaksi, barang yang dipinjam beserta jumlahnya, serta tenggat waktu pengembalian untuk status tertentu. *Search bar* di bagian atas memungkinkan pencarian spesifik berdasarkan kata kunci.



Gambar 4.10 Tampilan *Card* Transaksi Dengan Status *Waiting*

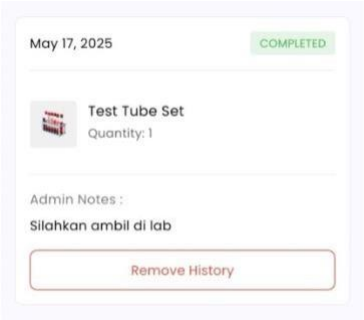
Gambar 4.10 menampilkan detail peminjaman yang sedang menunggu persetujuan admin, dengan label status *WAITING* bewarna kuning sebagai penanda visual. *Card* ini

menyertakan informasi barang yang diajukan, tombol *Cancel Transaction*, serta tenggat pengembalian yang sudah ditentukan sebelumnya saat proses *checkout*.



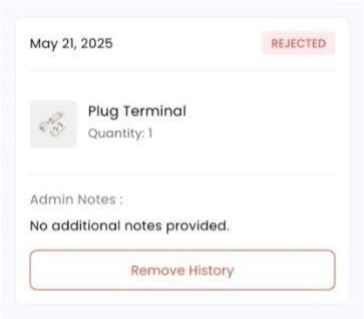
Gambar 4.11 Tampilan *Card* Transaksi Dengan Status *Ongoing*

Gambar 4.11 memperlihatkan transaksi yang telah disetujui dan sedang berjalan, ditandai dengan label *APPROVED* berwarna biru. Menampilkan daftar barang yang sedang dipinjam, catatan admin jika ada, serta tenggat pengembalian yang sudah ditentukan sebelumnya saat proses *checkout*.



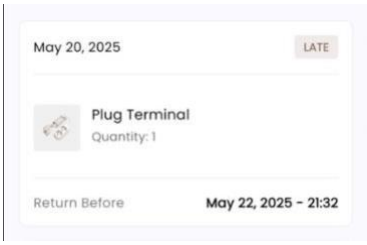
Gambar 4.12 Tampilan *Card* Transaksi Dengan Status *Completed*

Gambar 4.12 menyajikan riwayat transaksi yang telah selesai dengan label *COMPLETED* bewarna hijau. *Card* ini berisi detail barang yang dikembalikan beserta catatan admin jika ada. Pengguna dapat menghapus riwayat transaksi melalui opsi "*Remove History*".



Gambar 4.13 Tampilan *Card* Transaksi Dengan Status *Rejected*

Gambar 4.13 menampilkan transaksi yang tidak disetujui admin dengan label *REJECTED* berwarna merah. Secara eksplisit menampilkan alasan penolakan yang diberikan admin dan opsi penghapusan dari riwayat.

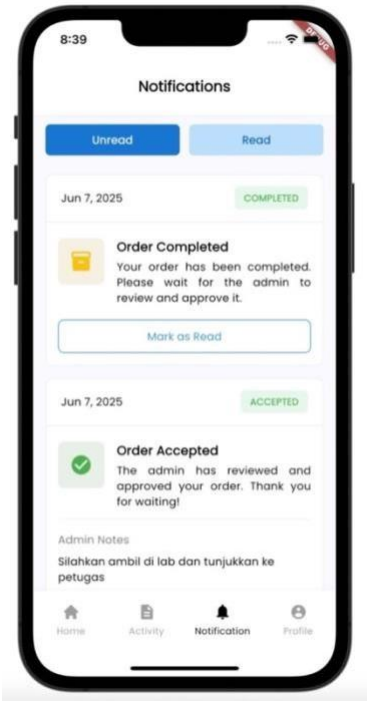


Gambar 4.14 Tampilan *Card* Transaksi Dengan Status *Late*

Gambar 4.14 dirancang khusus untuk menandai keterlambatan pengembalian dengan label *LATE* berwarna coklat. Menampilkan informasi barang yang terlambat dan tenggat waktu yang terlewat.

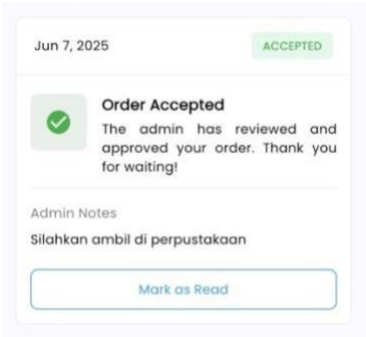
4.1.9 Tampilan Halaman *Notification* Peminjam

Antarmuka halaman *notification* menyajikan pusat pengelolaan seluruh notifikasi sistem terkait aktivitas peminjaman. Tampilan ini dirancang untuk memberikan peringatan ringkasan visual yang jelas tentang berbagai jenis pemberitahuan yang diterima pengguna.



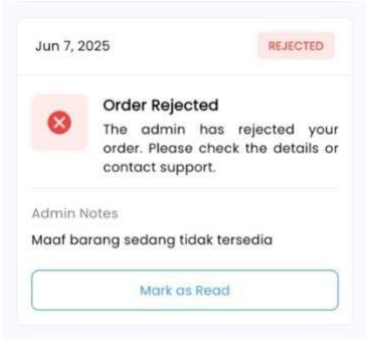
Gambar 4.15 Tampilan Halaman Utama *Notification* Peminjam

Gambar 4.15 menampilkan daftar terpusat semua notifikasi yang dikelompokkan berdasarkan status baca (*Unread*/*Read*) dan diurutkan secara kronologis. Setiap notifikasi mencakup informasi dasar seperti jenis notifikasi, tanggal notifikasi dibuat, dan pesan inti.



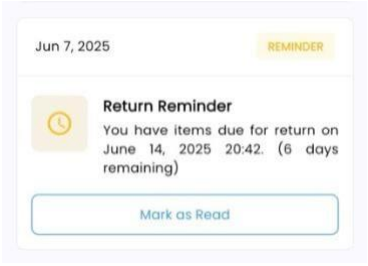
Gambar 4.16 Tampilan *Card* Notifikasi *Order Accepted*

Gambar 4.16 menampilkan *card* notifikasi persetujuan peminjaman dari admin, dengan penanda status *ACCEPTED* berwarna hijau. Card ini secara eksplisit menampilkan catatan admin tentang prosedur pengambilan barang dan tetap mempertahankan informasi dasar transaksi sebagai referensi.



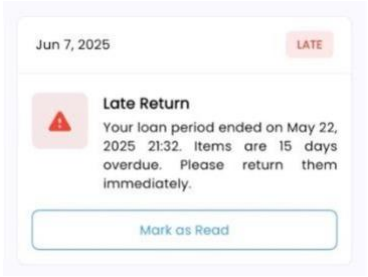
Gambar 4.17 Tampilan *Card* Notifikasi *Order Rejected*

Gambar 4.17 menampilkan card notifikasi penolakan peminjaman, dengan penanda status *REJECTED* berwarna merah. *Card* ini menonjolkan alasan penolakan dari admin dalam *section* khusus, memberikan transparansi proses pengajuan peminjaman.



Gambar 4.18 Tampilan *Card* Notifikasi *Return Reminder*

Gambar 4.18 menampilkan *card* notifikasi pengingat pengembalian, dengan penanda status *REMINDER* berwarna kuning. *Card* ini menyajikan informasi tenggat waktu pengembalian secara menonjol disertai hitungan mundur sisa waktu yang tersedia.

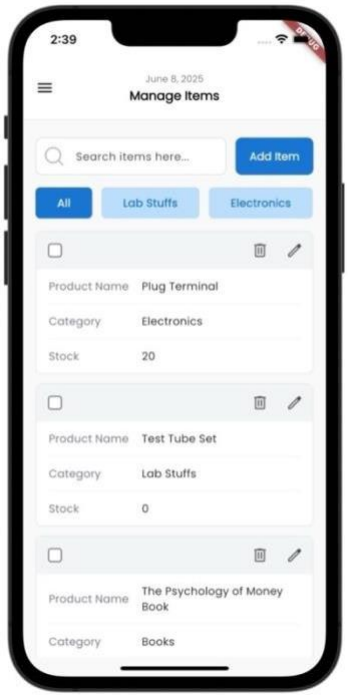


Gambar 4.19 Tampilan *Card* Notifikasi *Late Return*

Gambar 4.19 menampilkan *card* notifikasi keterlambatan pengembalian, dengan penanda status *LATE* berwarna merah. *Card* ini dirancang dengan penekanan visual pada jumlah hari keterlambatan dan instruksi *urgent* untuk pengembalian segera.

4.1.10 Tampilan Halaman *Manage Items* Admin

Antarmuka halaman *manage items* menyediakan fungsi lengkap untuk mengelola inventaris barang dalam sistem. Tampilan ini dirancang untuk memudahkan admin dalam melakukan operasi CRUD (*Create*, *Read*, *Update*, *Delete*) terhadap data barang.

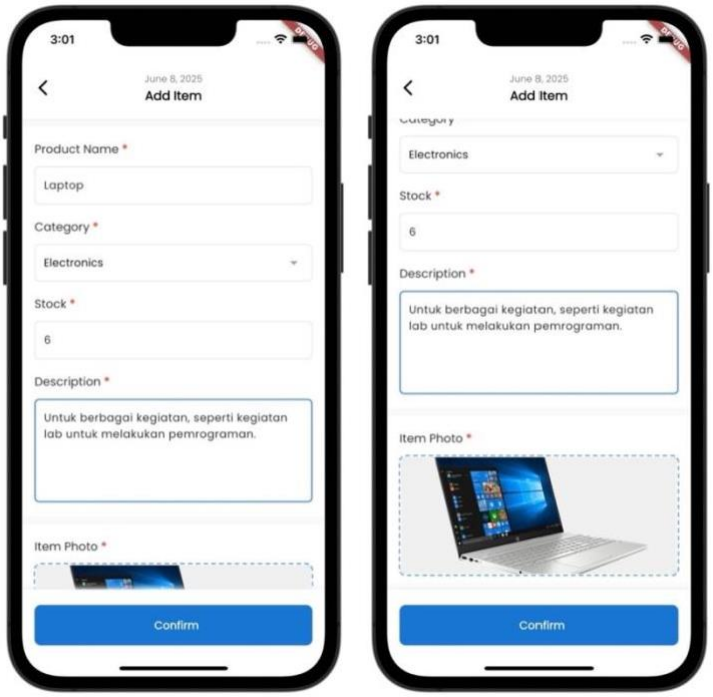


Gambar 4.20 Tampilan Halaman *Manage Items* Admin

Gambar 4.20 menampilkan daftar lengkap seluruh barang inventaris, admin dapat menyaring tampilan berdasarkan kategori utama melalui *tab* filter yang tersedia. Di bagian atas antarmuka, *search bar* memungkinkan pencarian instan berdasarkan kata kunci nama produk. Tombol "*Add Item*" memberikan akses langsung ke *form* penambaran bahang baru. Setiap item dilengkapi tombol edit (*icon* pensil) dan hapus (*icon* tong sampah).

4.1.11 Tampilan Halaman *Add Item* Admin

Antarmuka halaman *add item* admin dirancang untuk memudahkan admin menambahkan barang baru uke dalam sistem inventaris dengan validasi data yang ketat.

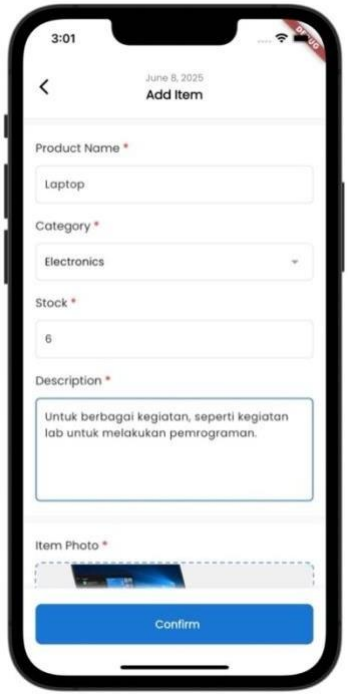


Gambar 4.21 Tampilan Halaman *Add Item* Admin

Gambar 4.21 menampilkan *form* penambahan barang baru dengan *field* wajib berupa *Product Name*, *Category* (*dropdown*), *Stock*, *Description*, dan *Item Photo*. Tombol *confirm* di bagian bawah berfungsi untuk memproses data setelah semua *field* terisi valid.

4.1.12 Tampilan Halaman Edit *Item* Admin

Antarmuka halaman edit *item* admin menyediakan fungsi lengkap untuk memodifikasi data barang yang telah terdaftar dalam sistem inventaris.

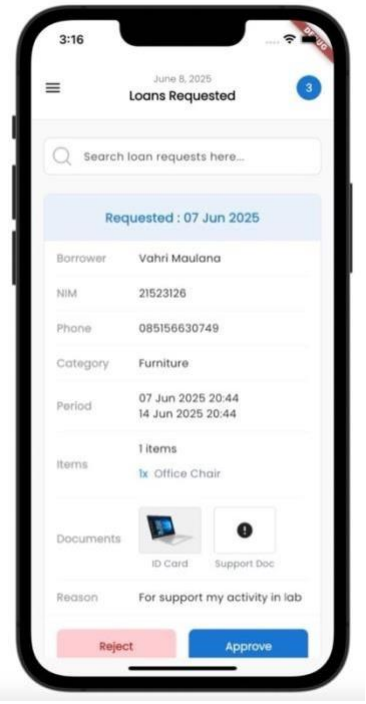


Gambar 4.22 Tampilan Halaman Edit *Item* Admin

Gambar 4.22 menampilkan *form* edit dengan data barang yang terisi otomatis sesuai item yang dipilih. *Form* init terdiri dari *field Procut Name*, *Category*, *Stock*, *Description*, dan *Item Photo*. Tombol "*Update*" di bagian bawah berfungsi untuk menyimpan perubahan setelah validasi data. Sistem secara otomatis menampilkan nilai-nilai nama sebagai referensi, memungkinkan admin melakukan perubahan selektif tanpa harus mengisi seluruh form dari awal.

4.1.13 Tampilan Halaman *Loans Requested* Admin

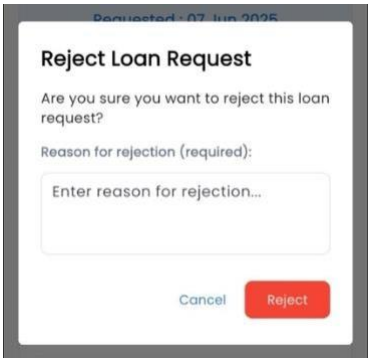
Antarmuka halaman *loans requested* admin menyediakan panel lengkap untuk memproses permohonan peminjaman barang dari peminjam. Tampilan ini dirancang untuk memudahkan admin dalam melakukan verifikasi dan persetujuan permintaan peminjaman.



Gambar 4.23 Tampilan Halaman *Loans Requested* Admin

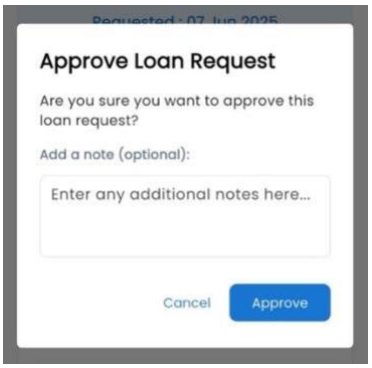
Gambar 4.23 menampilkan daftar permohonan peminjaman yang sedang menunggu persetujuan admin. Setiap permohonan ditampilkan dalam card terpisah yang mencakup informasi meliputi: identitas peminjam (Nama, NIM, Nomor Telepon, Foto KTM, Dokumen Tambahan yang bersifat opsional), detail peminjaman (Kategori Barang, Periode Peminjaman, Barang yang Dipinjam).

Bagian bawah setiap card menyertakan dua tombol aksi utama, yaitu *Reject* dan *Approve* untuk memproses permohonan. *Search bar* di bagian atas memungkinkan pencarian cepat berdasarkan kata kunci.



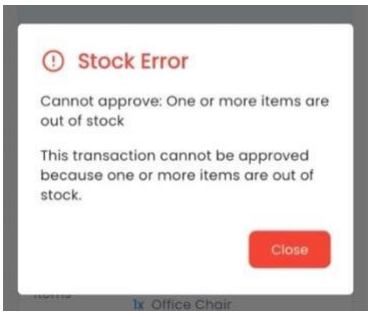
Gambar 4.24 Tampilan Dialog Konfirmasi *Reject*

Gambar 4.24 muncul ketika admin menekan tombol *Reject*, menampilkan *form* wajib pengisian alasan penolakan. Dialog ini memastikan admin memberikan penjelasan yang jelas kepada peminjam sebelum menolak permohonan.



Gambar 4.25 Tampilan Dialog Konfirmasi *Approve*

Gambar 4.25 ditampilkan saat admin menekan tombol *Approve*, menyediakan *field* opsional untuk catatan tambahan. Dialog ini memungkinkan admin menambahkan instruksi khusus tentang pengambilan barang sebelum menyetujui permohonan.

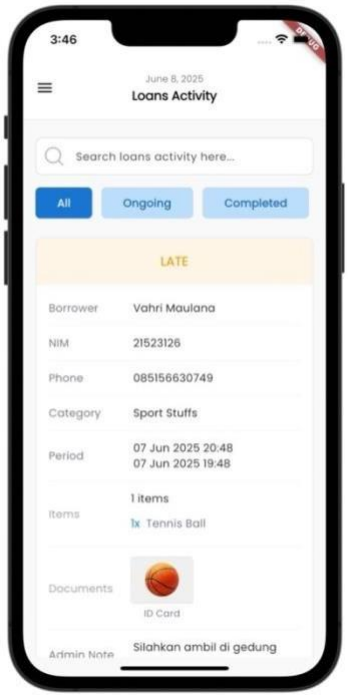


Gambar 4.26 Tampilan Dialog Peringatan Stok Habis

Gambar 4.26 menampilkan dialog peringatan sistem yang muncul otomatis ketika admin mencoba menyetujui peminjaman barang yang stoknya tidak tersedia.

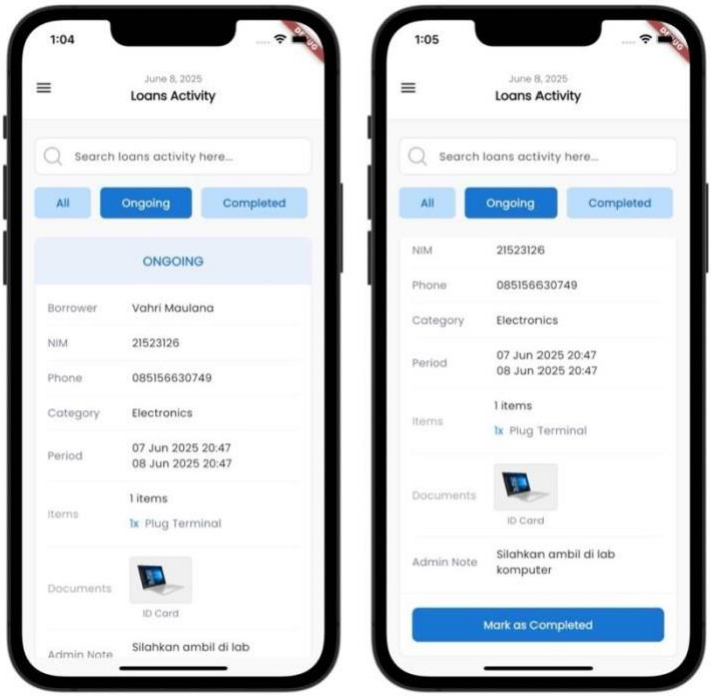
4.1.14 Tampilan Halaman *Loans Activity* Admin

Antarmuka halaman *loans activity* admin menyajikan ringkasan lengkap aktivitas peminjaman dalam sistem. Tampilan ini dirancang untuk memudahkan admin memantau perkembangan transaksi yang sedang berjalan.



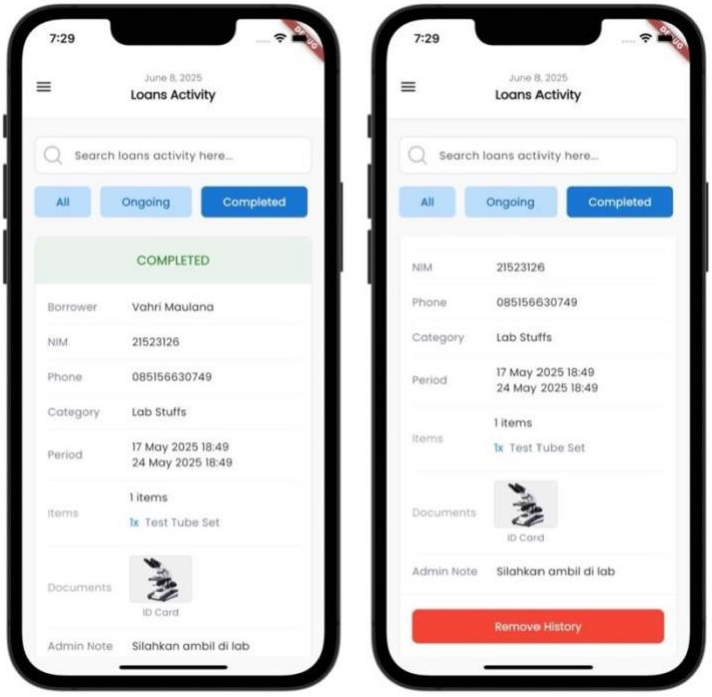
Gambar 4.27 Tampilan Utama *Halaman Loans Activity* Admin

Pada Gambar 4.27 menampilkan daftar transaksi yang dapat difilter berdasarkan status, yaitu: *Ongoing*, *Completed*, *Rejected*, *Late*, *Nearing Deadline*, dan *Late*. Setiap transaksi menampilkan status peminjaman, identitas peminjam, detail barang, dan catatan admin. *Search bar* memungkinkan pencarian cepat, sementara filter membantu menyortir berdasarkan status.



Gambar 4.28 Tampilan Halaman *Tab Ongoing Loans Activity* Admin

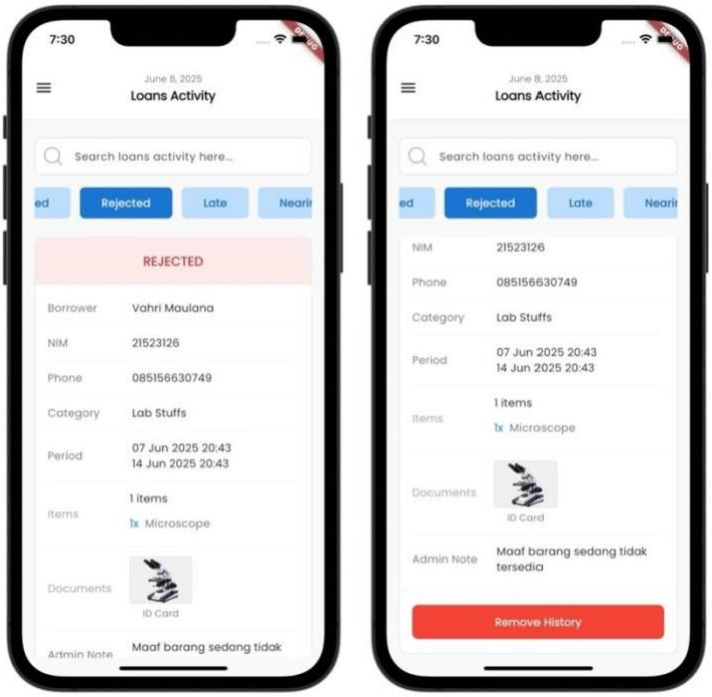
Gambar 4.28 menampilkan daftar transaksi peminjaman aktif dalam format *card* yang terorganisir, dengan setiap *card* menampilkan identitas peminjam, detail peminjaman, catatan admin, dan tombol "*Mark as Completed*" untuk penutupan transaksi.



Gambar 4.29 Tampilan Halaman *Tab Completed Loans Activity* Admin

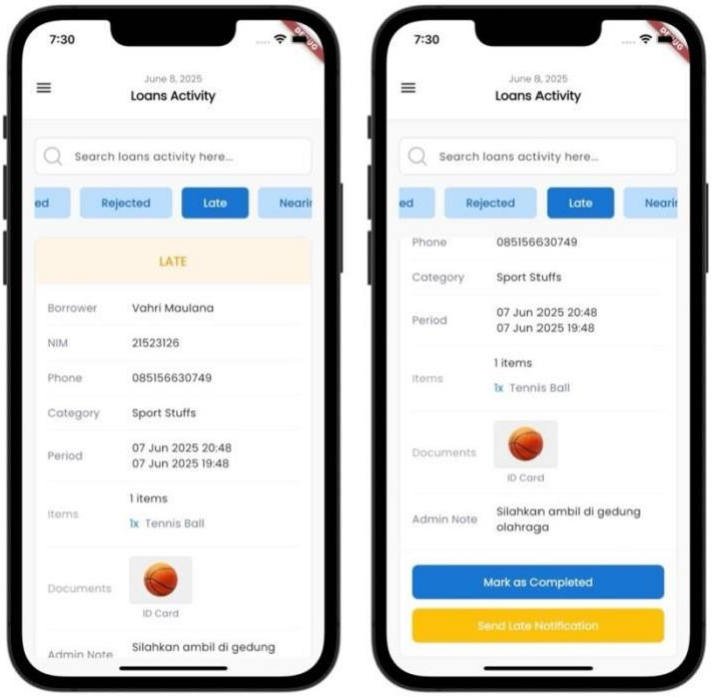
Gambar 4.29 menampilkan daftar transaksi yang telah selesai dengan status "*COMPLETED*" berwarna hijau, menampilkan identitas peminjam, detail peminjaman, catatan

admin, serta tombol "*Remove History*" untuk manajemen riwayat transaksi dengan tetap mempertahankan konsistensi desain *card*.



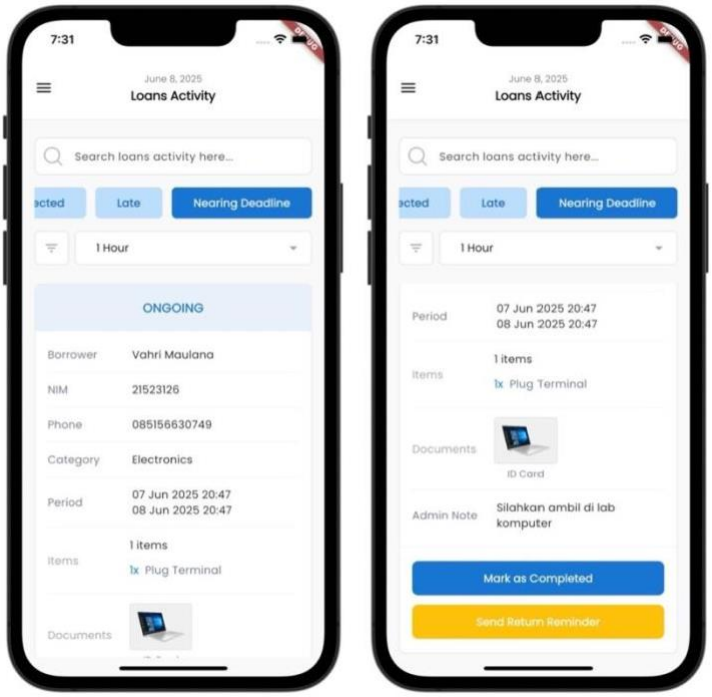
Gambar 4.30 Tampilan Halaman *Tab Rejected Loans Activity* Admin

Gambar 4.30 menampilkan daftar permohonan yang ditolak dengan status "*REJECTED*" berwarna merah, menampilkan informasi lengkap termasuk identitas peminjam, detail peminjaman, alasan penolakan dari admin, serta tombol "*Remove History*".



Gambar 4.31 Tampilan Halaman *Tab Late Loans Activity* Admin

Gambar 4.31 menampilkan transaksi peminjaman yang telah melewati tenggat waktu pengembalian dengan status "*LATE*" berwarna kuning. Setiap entri menampilkan data lengkap termasuk identitas peminjam, detail peminjaman yang terlambat dikembalikan, serta tombol "*Send Late Notification*".

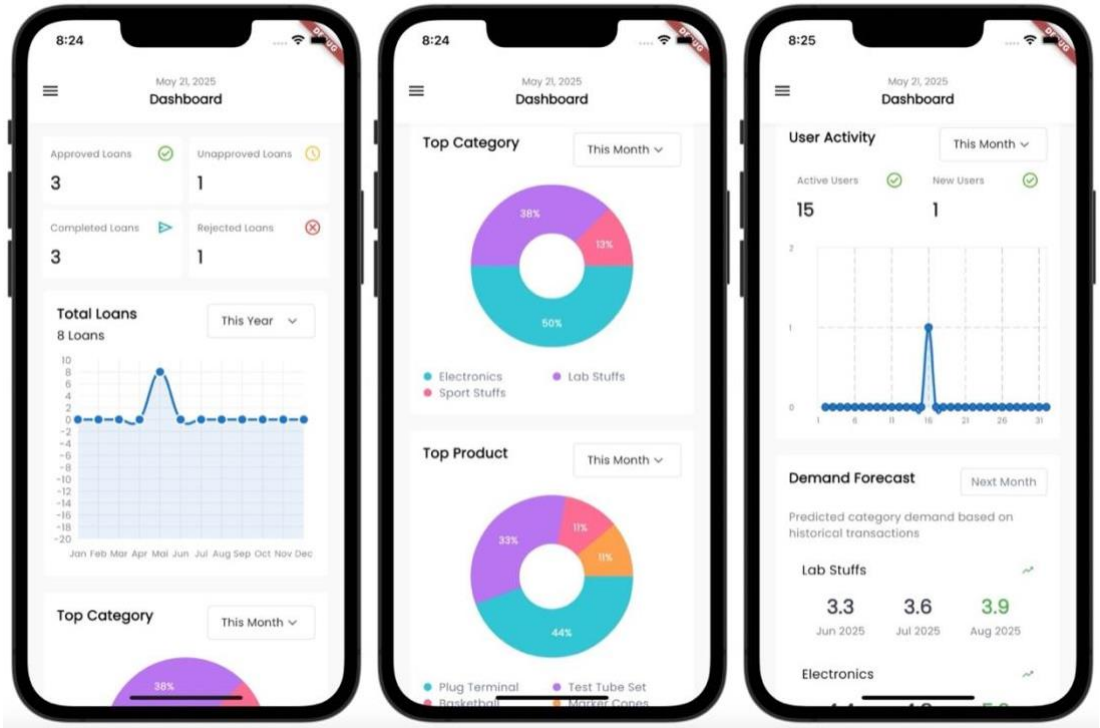


Gambar 4.32 Tampilan Halaman *Tab Nearing Deadline Loans Activity* Admin

Gambar 4.32 menampilkan transaksi peminjaman mendekati tenggat waktu pengembalian dengan filter berdasarkan sisa waktu pengembalian (3 *Days*, 2 *Days*, 1 *Days*, 12 *Hours*, 6 *Hours*, 3 *Hours*, dan 1 *Hour*). Setiap entri menampilkan informasi peminjam, detail peminjaman, dan tombol "*Send Return Reminder*" untuk mengirim notifikasi ke peminjam.

4.1.15 Tampilan Halaman *Dashboard* Admin

Antarmuka halaman *dashboard* admin menampilkan data ringkasan peminjaman dalam bentuk grafik, seperti total pinjaman, *user* aktif dan baru, kategori dan produk terpopuler, serta prediksi permintaan untuk tiga bulan ke depan.



Gambar 4.33 Tampilan *Dashboard* Admin

Pada Gambar 4.33 menampilkan Dashboard admin dengan empat bagian utama. Pada bagian *Top Category* dan *Top Product*, sistem menampilkan kategori produk terpopuler seperti *Electronics*, *Sport Stuffs*, dan *Lab Stuffs*, beserta daftar produk paling banyak dipinjam termasuk *Test Tube Set*, *Plug Terminal*, dan *Basketball*. Sementara Total *Loans* menampilkan grafik batang yang memvisualisasikan jumlah total peminjaman dengan puncak mencapai 8 *Loans*.

Dashboard ini juga mencakup *User Activity* (pengguna aktif dan baru), *Demand Forecast* (prediksi permintaan berdasarkan kategori), serta statistik peminjaman terbaru meliputi *Approved Loans*, *Completed Loans*, *Unapproved Loans*, dan *Rejected Loans*, menyediakan data komprehensif untuk pengambilan keputusan admin.

4.2 Pengujian Sistem

Pengujian sistem dilakukan untuk memvalidasi fungsionalitas aplikasi sesuai dengan kebutuhan pengguna. Pada subbab ini, pengujian dilakukan dengan pendekatan *Black-box Testing* untuk memverifikasi kebenaran fitur aplikasi tanpa memperhatikan struktur *internal code*, serta *usability testing* untuk mengevaluasi aspek kepuasan pengguna dari segi antarmuka (UI) dan pengalaman pengguna (UX).

4.2.1 *Black Box Testing*

Seperti yang telah dijelaskan pada bab sebelumnya, tahap pengujian awal ini difokuskan pada pemeriksaan fungsi sistem untuk memverifikasi fitur-fitur yang diimplementasikan telah sesuai dengan spesifikasi desain. Pendekatan *Black-box Testing* digunakan untuk mengevaluasi kinerja sistem melalui antarmuka pengguna (*front-end*) dan alur proses bisnis, tanpa melakukan analisis terhadap kode program. Rekapitulasi hasil pengujian yang berisi rangkuman *test case*, hasil yang diharapkan, dan output aktual disajikan dalam tabel-tabel berikut.

Tabel 4.1 Hasil Pengujian *Black-Box Register*

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.1 Registrasi	*Input* data *valid*	Nama: "Vahri", Email: "vahri@gmail.com", *Password*: "vahri123"	1. Data tersimpan di *database* 2. Notifikasi sukses muncul 3. Email verifikasi terkirim	Semua terpenuhi	*Pass*
	Email sudah terdaftar	Email: "vahri@gmail.com"	1. Sistem menolak registrasi 2. Pesan error: "*The email is already in use by another account*"	Semua terpenuhi	*Pass*
	Password terlalu pendek	*Password*: "vahri1"	1. Sistem menolak registrasi 2. Pesan error: "*Password must be at least 8 characters*"	Semua terpenuhi	*Pass*
	Password tidak sama	*Password*: "vahri123"	1. Sistem menolak registrasi		

	dengan konfirmasi *password*	Konfirmasi *Password*: "vahri1234"	2. Pesan error: "*Password do not match*"		
	Field wajib kosong	Nama: (kosong), Email: "vahri@gmail.com"	1. Sistem menolak registrasi 2. *Highlight* merah *field* wajib	Semua terpenuhi	*Pass*

Tabel 4.2 Hasil Pengujian *Black-Box Login*

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.2 Login	Kredensial valid Admin	Email: "admin@gmail.com" *Password*: "adminnyainvenshare"	1. *Redirect* ke dashboard Admin 2. *Session* terbentuk	Semua terpenuhi	*Pass*
	Kredensial valid Peminjam	Email: "vahri@gmail.com" *Password*: "vahri123"	1. *Redirect* ke home peminjam 2. *Session* terbentuk	Semua terpenuhi	*Pass*
	Email salah	Email: "mail@gmail.com"	Muncul pesan error: "*Invalid email or password. Please try again*"	Sesuai ekspektasi	*Pass*
	Password salah	Email: "vahri@gmail.com" *Password*: vahri1235	Muncul pesan error: "*Invalid email or password. Please try again*"	Sesuai ekspektasi	*Pass*
	Field kosong	Email: (Kosong) *Password*: (Kosong)	1. Sistem menolak *login* 2. *Highlight* merah *field* wajib	Semua terpenuhi	*Pass*

Tabel 4.3 Hasil Pengujian *Black-Box* Peminjaman Barang

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.3 Peminjaman	Tambah Barang ke Keranjang	Klik tombol "*Add to Cart*" pada "*Projector*"	Barang masuk ke keranjang	Sesuai ekspektasi	*Pass*
	Checkout per Kategori	Pilih kategori "*Electronic*"	Hanya tampilkan barang dalam kategori	Sesuai ekspektasi	*Pass*
	Submit Form Valid	NIM: "21523126" *Phone Number*: "085156630749" *Reason*: "*For Workshop*" *Start Date*: "10 *June* 12.52" *End Date*: "10 June 14.52" KTM *Pict*: "Vahri's KTM" *Supporting Document* (*Optional*): -	1. Dialog sukses muncul 2. Notifikasi terkirim ke halaman *Notification* peminjam 3. *History* pengajuan muncul di halaman *Activity* peminjam dengan status "*WAITING*" 4. Keranjang untuk kategori tersebut kosong 5. Pengajuan barang masuk ke halaman *Loans Requested* Admin	Semua terpenuhi	*Pass*
	Submit Form Kosong	*Field* "Reason" dikosongkan	*Highlight* merah *field* wajib ketika klik tombol "*Submit*"	Sesuai ekspektasi	*Pass*

Tabel 4.4 Hasil Pengujian *Black-Box* Pembatalan Pengajuan

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.4 Pembatalan	Peminjam batalkan (status: *WAITING*)	1. Klik tombol "*Cancel Transaction*" di halaman *Activity* 2. Konfirmasi dialog konfirmasi	Data pengajuan hilang di halaman *Loans Requested* Admin	Terpenuhi	*Pass*

Tabel 4.5 Hasil Pengujian *Black-Box* Admin Konfirmasi Pengajuan

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.5 Konfirmasi Pengajuan	Admin menyetujui pengajuan peminjaman (stok tersedia)	1. Klik tombol "*Approve*" 2. Isi catatan (opsional) *3. Submit*	1. Stok barang berkurang sesuai yang dipinjam 2. Notifikasi pengajuan diterima muncul di halaman *Notification* peminjam dan *notification center handphone* peminjam 3. Status: "*APPROVED*" (*Activity* Peminjam) & "*ONGOING*" (*Loans Activity* Admin)	Semua terpenuhi	*Pass*
	Admin menyetujui pengajuan peminjaman (stok habis)	Klik tombol "*Approve*"	1. Muncul dialog peringatan bahwa stok barang habis 2. Proses *approve* dibatalkan 3. Status tetap menjadi "*WAITING*" di halaman Activity Peminjam	Semua terpenuhi	*Pass*
	Admin menolak	1. Klik tombol "*Reject*"	1. Notifikasi pengajuan ditolak diterima muncul	Semua terpenuhi	*Pass*

	pengajuan peminjaman	2. Isi alasan penolakan (wajib) *3. Submit*	di halaman *Notification* peminjam *notification center handphone* peminjam 2. Status "*REJECTED*" (*Activity* Peminjam & *Loans Activity* admin) 3. Stok barang tidak berubah		

Tabel 4.6 Hasil Pengujian *Black-Box* Pengembalian Barang

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.6 Pengembalian	Admin mengonfirmasi barang pengembalian	Klik tombol "*Mark as Completed*" di halaman *Loans Activity*	3. Status transaksi berubah menjadi "*COMPLETED*" 4. Stok barang bertambah kembali sesuai dengan barang yang dipinjam	Semua terpenuhi	*Pass*

Tabel 4.7 Hasil Pengujian *Black-Box* CRUD Barang

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.7 CRUD Barang	Admin menambahkan barang baru (valid)	1. Akses menu "*Add Item*" 2. Isi form dengan data valid (nama, stok, kategori, deskripsi) *3. Submit*	1. Data barang tersimpan di *database* 2. *Message* barang berhasil ditambahkan muncul 3. Barang masuk ke daftar barang di halaman "*Manage Items*" 4. Barang muncul di halaman peminjam	Semua terpenuhi	*Pass*

	Admin menambahkan barang (data kosong)	1. Akses menu "*Add Item*" di halaman "*Manage Items*" 2. *Field* "Nama" dikosongkan	1. *Highlight* merah *field* wajib ketika klik tombol "*Submit*" 2. Data tidak tersimpan	Semua terpenuhi	*Pass*
	Admin mengedit data barang	1. Akses menu "*Manage Items*" 2. Klik *icon* pensil di barang yang ingin diedit 3. Ubah data (contoh: stok) *4. Submit*	1. Data barang terupdate di *database* 2. Message barang berhasil diedit muncul	Semua terpenuhi	*Pass*
	Admin menghapus barang	1. Akses menu "*Manage Items*" 2. Klik *icon* tong sampah di barang yang ingin dihapus 3. Konfirmasi dialog *pop-up*	1. Data barang dihapus dari *database* 2. *Message* barang berhasil dihapus muncul 3. Barang terhapus di daftar barang halaman "*Manage Items*" 4. Barang terhapus di halaman peminjam	Semua terpenuhi	*Pass*

Tabel 4.8 Hasil Pengujian *Black-Box* Pengiriman Notifikasi

Use Case	Skenario	*Input*	Hasil yang Diharapkan	Hasil Aktual	Status
UC.8 Pengiriman Notifikasi	Admin mengirim notifikasi pengingat pengembalian	1. Akses menu *Loans Activity* 2. Klik tombol "*Send Return Reminder*" pada transaksi di *tab* "*Nearing Deadline*"	1. Muncul *message* notifikasi berhasil dikirim 2. Peminjam menerima notifikasi di halaman *Notification* dan *notification center handphone*	Semua Terpenuhi	*Pass*

		3. Konfirmasi dialog konfirmasi			
	Admin mengirim notifikasi keterlambatan	1. Akses menu *Loans Activity* 2. Klik tombol "*Send Late Notification*" pada transaksi di tab "*Late*" 3. Konfirmasi dialog konfirmasi	1. Muncul *message* notifikasi berhasil dikirim 2. Peminjam menerima notifikasi di halaman *Notification* dan *notification center* di *handphone*	Semua Terpenuhi	*Pass*

4.2.2 *Usability Testing*

Pengujian usability pada sistem ini dilakukan dengan pendekatan kuesioner berbasis *Positive* SUS (*System Usability Scale* versi positif). Metode yang dikembangkan oleh Sauro dan Lewis. Metode ini dipilih karena memiliki validitas setara dengan SUS standar namun lebih sederhana dalam penerapan (Melissourgos George, 2023) Pengguna dari kedua peran (admin dan peminjam) memberikan penilaian menggunakan skala Likert 5 poin, mulai dari Sangat Tidak Setuju (1) hingga Sangat Setuju (5), tanpa menerapkan *reverse scoring* pada pertanyaan tertentu. Prosedur perhitungan Positive SUS dapat dilihat di poin-poin berikut:

a. Untuk setiap pertanyaan, kurangi 1 dari nilai responden (1 – 5 menjadi 0 – 4).
b. Jumlahkan skor semua pertanyaan.
c. Kalikan total dengan 2.5 untuk mendapatkan skor SUS (0 – 100).

Rumus:

$$\text{Skor SUS} = \left(\sum_{i=1}^{10} \text{Nilai } Q_i - 10\right) \times 2{,}5 \quad (1)$$

Tabel 4.9 Pertanyaan *System Usability Scale* Untuk Peminjam

Kode	Pertanyaan
Q1	Saya dapat meminjam barang dengan mudah tanpa bantuan.
Q2	Fitur pencarian barang bekerja dengan baik.
Q3	Proses pengajuan peminjaman mudah dipahami.
Q4	Notifikasi status peminjaman jelas dan tepat waktu.

Q5	Sistem menunjukkan ketersediaan barang secara *real-time*.
Q6	Saya dapat melihat riwayat peminjaman saya dengan lengkap.
Q7	Saya dapat dengan mudah kembali/membatalkan jika melakukan kesalahan.
Q8	Sistem memberikan konfirmasi yang jelas setelah pengajuan peminjaman.
Q9	Tampilan aplikasi tidak membingungkan.
Q10	Sistem ini lebih efisien daripada proses manual.

Tabel 4.10 Pertanyaan *System Usability Scale* Untuk Admin

Kode	Pertanyaan
Q1	Saya dapat memvalidasi peminjaman dengan cepat.
Q2	Fitur pelacakan stok *real-time* akurat.
Q3	Sistem memudahkan persetujuan/penolakan peminjaman.
Q4	Laporan riwayat transaksi lengkap dan mudah diakses.
Q5	Saya dapat mengelola data inventaris dengan efisien.
Q6	Notifikasi (pengingat pengembalian) efektif.
Q7	Antarmuka admin memudahkan tugas rutin.
Q8	Sistem mengurangi kesalahan pencatatan manual.
Q9	Saya puas dengan fitur manajemen pengguna.
Q10	Sistem ini menghemat waktu kerja administratif.

Pertanyaan SUS untuk peminjam (Tabel 4.9) secara khusus dirancang untuk mengevaluasi aspek-aspek utama pengalaman pengguna akhir, meliputi kemudahan proses peminjaman, efektivitas fitur pencarian, kejelasan notifikasi, dan efisiensi sistem secara keseluruhan. Sementara itu, kuosioner untuk admin (Tabel 4.10) difokuskan pada aspek administratif dan manajerial sistem, termasuk kecepatan dan kemudahan dalam memvalidasi peminjaman, akurasi pelacakan stok barang, serta efektivitas sistem dalam mendukung tugas-tugas administratif sehari-hari.

Pengujian ini melibatkan 20 responden yang terdiri dari 15 mahasiswa sebagai peminjam dan 5 staf DAURT FTI sebagai admin. Seluruh responden yang sebelumnya terlibat pada tahap wawancara analisis kebutuhan juga turut serta dalam pengujian aplikasi ini, sehingga hasil evaluasi dapat memperlihatkan kesinambungan antara kebutuhan awal dan pengalaman penggunaan sistem. Rincian demografi responden ditunjukkan pada Tabel 4.11.

Tabel 4.11 Demografi Responden Pengujian

Nama	Peran	Fakultas/Unit	Jenis Kelamin	Foto
Basuki Setiyawan	Admin	FTI - DAURT	Laki-laki	
Anonim	Admin	FTI – Staf Laboratorium Teknik Industri	Laki-laki	-
Hadi Liswanto	Admin	FTI – Staf Administrasi Teknik Kimia	Laki-laki	
Dwi Wahyu Santoso	Admin	FTI – Staf Laboratorium Teknik Industri	Laki-laki	
Anonim	Admin	FMIPA – Laboran Jurusan Statistika	Laki-laki	-
Muhammad Maulana Sahputra	Mahasiswa	FTI - Informatika	Laki-laki	

Rizqdwan Dhuhakbar Hendyutama	Mahasiswa	FTI - Informatika	Laki-laki	
Rabby Ahsan Matswaya	Mahasiswa	FTI - Informatika	Laki-laki	
Annisa Dyah Ayu	Mahasiswa	FK - Kedokteran	Perempuan	
Muhammad Luqman Hidayat	Mahasiswa	FTI - Informatika	Laki-laki	
Agung Prasetyo Abdjul	Mahasiswa	FTI - Informatika	Laki-laki	

Yoangga Achmad Dwi Pasanjaya	Mahasiswa	FTI - Informatika	Laki-laki	
Muhammad Rafli Fauzi	Mahasiswa	FTI - Informatika	Laki-laki	
Ridzky Putra Dwitama	Mahasiswa	FTI - Informatika	Laki-laki	
Fikri Achmad Fauzi	Mahasiswa	FTI - Informatika	Laki-laki	
Mutia Putri Gaisani	Mahasiswa	FTI – Teknik Industri	Perempuan	

Amiruddin Syah	Mahasiswa	FIAI – Pendidikan Agama Islam	Laki-laki	
M. Fauzan Afdillah Rahman	Mahasiswa	FH - Hukum	Laki-laki	
Fadhlan Putera Sopian	Mahasiswa	FTI - Informatika	Laki-laki	
Tiwi Esti Ariayanti	Mahasiswa	FPSB - Psikologi	Perempuan	

Hasil penilaian dari kedua kelompok pengguna ini kemudian dianalisis secara mendalam dengan mempertimbangkan berbagai dimensi pengalman pengguna, termasuk aspek fungsionalitas, kemudahan penggunaan, keandalan sistem, dan kepuasan pengguna secara keseluruhan. Rekapitulasi lengkap dari tanggapan responden disajikan secara terperinci dalam Tabel 4.12 untuk peminjam dan Tabel 4.13 untuk admin.

Tabel 4.12 Hasil Pengujian Untuk Peminjam

Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	4	5	4	4	5	5	4	4	5	5
R2	5	5	4	4	5	5	5	5	4	4
R3	4	5	4	3	5	5	5	4	4	5
R4	4	4	3	5	5	5	4	5	3	4
R5	5	5	4	5	5	5	5	4	4	5
R6	5	5	5	5	5	5	5	5	5	4
R7	4	5	5	5	5	5	5	5	5	5
R8	4	5	4	4	4	5	5	4	5	5
R9	4	5	5	4	4	3	5	5	4	5
R10	4	4	4	5	4	4	3	4	5	5
R11	4	4	4	4	3	4	4	3	4	4
R12	4	5	5	5	5	5	5	5	5	5
R13	4	5	4	4	4	5	4	5	4	4
R14	5	5	4	4	5	5	4	4	4	4
R15	4	4	5	5	4	5	5	4	5	4

Tabel 4.13 Hasil Pengujian Untuk Admin

Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	5	5	5	5	5	5	5	5	5	5
R2	4	4	4	4	4	4	2	3	4	3
R3	5	4	5	4	5	3	4	5	4	4
R4	5	5	4	5	4	5	4	5	5	5
R5	4	5	5	5	4	4	4	4	3	4

Berdasarkan data hasil pengujian yang telah dikumpulkan dari 15 responden peminjam dan 5 responden admin, berikut adalah hasil perhitungan menggunakan metode *Positive* SUS:

Tabel 4.14 Hasil Perhitungan Skor SUS Untuk Peminjam

Responden	Total Skor	Skor SUS	Kategori
R1	45 – 10 = 35	87,5	*Excellent*
R2	46 – 10 = 36	90,0	*Excellent*
R3	44 – 10 = 34	85,0	*Excellent*
R4	42 – 10 = 32	80,0	*Excellent*
R5	47 – 10 = 37	92,5	*Excellent*
R6	49 – 10 = 39	97,5	*Best Imaginable*
R7	48 – 10 = 38	95,0	*Best Imaginable*
R8	45 – 10 = 35	87,5	*Excellent*
R9	44 – 10 = 34	85,0	*Excellent*
R10	42 – 10 = 32	80,0	*Excellent*
R11	38 – 10 = 28	70,0	*Good*
R12	49 – 10 = 39	97,5	*Best Imagibanle*
R13	43 – 10 = 33	82,5	*Excellent*
R14	44 – 10 = 34	85,0	*Excellent*
R15	45 – 10 = 35	87,5	*Excellent*
Rata-Rata	34.73	85,83	*Excellent*

Tabel 4.15 Hasil Perhitungan Skor SUS Untuk Admin

Responden	Total Skor	Skor SUS	Kategori
R1	50 – 10 = 40	100,0	*Best Imaginable*
R2	36 – 10 = 26	65,0	*OK*
R3	43 – 10 = 33	82,5	*Excellent*
R4	47 – 10 = 37	92,5	*Excellent*
R5	42 – 10 = 32	80,0	*Excellent*
Rata-Rata	33.6	84,0	*Excellent*

Skor rata-rata pengujian *Positive* SUS menunjukkan bahwa sistem yang dikembangkan memiliki kualitas *usability* yang sangat baik dengan skor 85,83 untuk peminjam (kategori *Excellent*) dan 84,0 untuk admin (kategori *Excellent*), hasil ini mengindikasikan bahwa sistem secara keseluruhan telah memenuhi tujuan pengembangan dari segi kemudahan penggunaan dan kepuasan pengguna.

Dalam penelitian ini turut dikumpulkan masukan kualitatif dari pengguna untuk memperkaya evaluasi sistem. Masukan dari responden admin dapat dilihat pada Lampiran A, sementara tanggapan dari mahasiswa ditunjukkan pada Lampiran B. Masukan tersebut meliputi apresiasi terhadap kemudahan penggunaan sistem, usulan penambahan fitur deposit sebagai kontrol peminjaman, serta rekomendasi ekspor data peminjaman dalam format Excel untuk keperluan administrasi.

Pengujian juga didukung oleh sesi uji coba langsung bersama pengguna. Dokumentasi pengujian bersama staf DAURT FTI ditampilkan pada Lampiran C, sedangkan dokumentasi pengujian bersama mahasiswa ditunjukkan pada Lampiran D. Lampiran tersebut merekam proses demonstrasi sistem serta interaksi responden saat menguji fitur-fitur utama aplikasi.

4.2.3 Analisis Kelebihan dan Kekurangan Sistem

Berdasarkan hasil pengujian usability dengan *Positive* SUS, berikut adalah hasil analisis kelebihan dan kekurangan sistem:

1. Kelebihan Sistem:
    - Sistem memberikan pengalaman pengguna yang sangat baik bagi peminjam dengan skor rata-rata 85,83 (*Excellent*), terutama dalam:
        - Kemudahan proses peminjaman barang (Q1 peminjam rata-rata: 4,33).
        - Akurasi informasi ketersediaan barang *real-time* (Q5 peminjam rata-rata: 4,73).
        - Kelengkapan fitur riwayat peminjaman (Q6 peminjam rata-rata: 4,8)
    - Antarmuka admin yang efisien dengan skor rata-rata 84,0 (*Excellent*) dalam:
        - Proses validasi peminjaman yang cepat (Q1 admin rata-rata 4,6).
        - Pelacakan stok barang *real-time* (Q2 admin rata-rata: 4,6).
        - Penghematan waktu kerja administratif (Q5 admin rata-rata 4,4).

- Konsistensi performa sistem dengan:
    - 100% responden peminjam memberikan skor *Excellent* ke atas.
    - 80% responden admin memberikan skor *Excellent* ke atas.

2. Kekurangan Sistem:
    - Sistem notifikasi masih perlu peningkatan:
        - Notifikasi peminjaman (Q4 peminjam) mendapat skor terendah 3,93.
        - Notifikasi pengembalian (Q6 admin) mendapat skor 4,2 dengan 20% responden memberi nilai ≤ 3.
    - Beberapa fitur manajemen perlu penyempurnaan:
        - Antarmuka manajemen pengguna admin (Q9 admin) mendapat skor 4,2.
        - Fitur pembatalan transaksi (Q7 eminjam) mendapat skor 4,47.
    - Kurva pembelajaran untuk Admin:
        - 40% admin (R1 dan R5) memberi skor ≤ 4 pada fitur manajemen (Q5 admin dan Q9 admin), mengindikasikan kebutuhan pelatihan tambahan.

Berdasarkan analisis kelebihan dan kekurangan sistem, dapat disimpulkan bahwa sistem peminjaman barang ini secara keseluruhan telah mencapai tingkat *usability* yang sangat baik, dengan skor rata-rata 85,84 (*Excellent*) untuk pengguna peminjam dan 84,0 (*Excellent*) untuk admin. Kelebihan utama terletak pada kemudahan penggunaan, efisiensi proses peminjaman, dan efektivitas inventaris. Namun, beberapa aspek seperti sistem notifikasi, prosedur pembatalan transaksi, dan adaptasi fitur admin masih memerlukan penyempurnaan untuk meningkatkan pengalaman pengguna secara menyeluruh. Perbaikan pada area tersebut diharapkan dapat mengoptimalkan fungsionalitas sistem dan menjadikannya lebih intuitif bagi semua pengguna. Dengan demikian, sistem ini telah memenuhi tujuan pengembangan dari segi kegunaan, namun tetap memiliki ruang untuk pengembangan lebih lanjut guna mencapai tingkat kepuasan yang lebih tinggi.

4.3 Kesesuaian Sistem dengan Aturan Peminjaman di UII

Studi kasus penelitian ini dilakukan di Universitas Islam Indonesia (UII), di mana pengelolaan inventaris barang dijalankan secara desentralisasi. Setiap fakultas atau jurusan memiliki unit pengelolaan inventaris masing-masing, dengan aturan peminjaman yang tidak selalu sama. Sebagian unit di beberapa fakultas memberlakukan aturan denda keterlambatan apabila barang tidak dikembalikan sesuai waktu yang ditentukan.

Sistem yang dikembangkan dalam penelitian ini bersifat tersentralisasi dengan satu basis data inventaris. Pendekatan ini menyederhanakan pencatatan, pelacakan, dan pelaporan, namun belum sepenuhnya sesuai dengan kondisi nyata di UII. Pertama, sistem tidak mengakomodasi perbedaan pengelolaan antar fakultas karena semua data dipusatkan dalam satu *platform*. Kedua, sistem belum mendukung perhitungan denda keterlambatan, sehingga keterlambatan hanya tercatat melalui tanggal peminjaman dan pengembalian tanpa adanya konsekuensi administratif atau finansial.

Hal-hal ini menjadi catatan keterbatasan penelitian. Sistem yang dihasilkan tetap berfungsi sebagai prototipe aplikasi *mobile* untuk peminjaman inventaris, namun diperlukan pengembangan lebih lanjut agar selaras dengan aturan di UII. Misalnya, dengan menambahkan fitur multi-admin per fakultas serta modul denda otomatis sesuai ketentuan kampus.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilaksanakan, dapat ditarik beberapa kesimpulan sebagai berikut.

Pertama, penelitian ini berhasil mengidentifikasi bahwa pengelolaan inventaris di kampus yang masih menggunakan metode manual rentan menimbulkan berbagai kendala, seperti kesalahan pencatatan, duplikasi data, serta ketidaksesuaian antara catatan inventaris dengan kondisi fisik barang. Aplikasi yang dikembangkan mampu menjawab permasalahan tersebut dengan menyediakan sistem digital yang terintegrasi, sehingga pencatatan dan pembaruan data inventaris dapat dilakukan secara otomatis dan lebih akurat.

Kedua, keterbatasan sistem berbasis *website* atau *desktop* yang tidak sepenuhnya menyediakan akses *real-time* dan dukungan terpadu untuk proses peminjaman berhasil diatasi melalui pengembangan aplikasi *mobile*. Aplikasi ini dirancang dengan fitur utama berupa validasi stok secara *real-time*, persetujuan dan penolakan peminjaman, pengelolaan data inventaris, serta notifikasi pengingat pengembalian, sehingga mampu meningkatkan efisiensi operasional dan mengurangi hambatan yang sebelumnya ada pada sistem manual maupun berbasis *website*/*desktop*.

Ketiga, kebutuhan pengguna terhadap solusi *mobile* yang praktis dan terpadu telah terpenuhi melalui aplikasi yang dibangun dalam penelitian ini. Aplikasi tidak hanya memfasilitasi pengelolaan inventaris oleh admin, tetapi juga mendukung mahasiswa sebagai peminjam melalui antarmuka yang mudah digunakan, aksesibilitas kapan saja dan di mana saja, serta pengalaman penggunaan yang teruji melalui pengujian *System Usability Scale (*SUS*)*. Hasil pengujian menunjukkan bahwa aplikasi memiliki tingkat *usability* yang baik dan dapat diterima oleh pengguna.

Dengan demikian, seluruh rumusan masalah penelitian telah terjawab melalui analisis kebutuhan, perancangan, implementasi, dan evaluasi aplikasi *mobile* untuk pengelolaan dan peminjaman inventaris barang di kampus.

5.2 Saran

Berdasarkan temuan selama pengujian dan analisis keterbatasan sistem, saran pengembangan lanjutan meliputi:

1. Pendalaman Analisis Kebutuhan:
    - Melibatkan pihak DAURT di setiap fakultas UII dalam proses analisis untuk memperoleh gambaran lebih komprehensif mengenai aturan dan prosedur peminjaman.
    - Menyusun instrumen wawancara dengan pertanyaan yang lebih mendalam agar kebutuhan pengguna dapat tergali secara lebih akurat.
2. Fleksibilitas Peminjaman Lintas Kategori:

    Sistem saat ini masih membatasi peminjaman hanya pada satu kategori barang dalam setiap transaksi. Pengembangan ke depan sebaiknya memungkinkan peminjaman lintas kategori sekaligus, agar lebih sesuai dengan kebutuhan nyata di lapangan, mempermudah pengguna, mengurangi duplikasi proses, dan meningkatkan efisiensi operasional.
3. Peningkatan Notifikasi:
    - Menambahkan fitur notifikasi otomatis via email/SMS untuk status peminjaman dan pengingat pengembalian.
    - Implementasi notifikasi otomatis untuk pengingat pengembalian tanpa inisiasi admin.
    - Integrasi dengan kalender perangkat pengguna untuk alarm berbasis waktu.
4. Ekspansi integrasi sistem:
    - Integrasi dengan sistem kampus untuk validasi otomatis data mahasiswa dan NIM.
    - Penghubungan dengan sistem keuangan untuk otomasi perhitungan denda keterlambatan.
5. Pengembangan fitur efisiensi operasional:
    - Penambahan pemindaian *barcode*/*QR code* untuk mempercepat proses *checkout* dan pengembalian barang.
    - Modul analitik prediktif berbasis AI untuk perkiraan kebutuhan barang dan restok otomatis.

6. Optimasi pengalaman pengguna:
    - Tutorial interaktif khusus admin untuk mengurangi kurva pembelajaran fitur manajemen kompleks.
    - Redesain antarmuka fitur pembatalan transaksi agar lebih intuitif.
7. Peningkatan keamanan dan skalabilitas:
    - Enskripsi data *sensitive* (KTM/dokumen) dan penerapan autentikasi dua faktor (2FA).
    - *Load testing* untuk memastikan kinerja optimal dengan pengguna > 1000.

DAFTAR PUSTAKA

Anisah, S. (2022). Implementasi Metode Rapid Application Development pada Pengembangan Aplikasi Inventory Barang. *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, *7*(1). https://doi.org/10.30998/string.v7i1.13064

Aprilisa, S., & Aulia, R. (2024). Penerapan Metode Prototype dalam Pengembangan Sistem Informasi Inventory Barang Berbasis Web. *Jurnal Teknik Industri Terintegrasi*, *7*(1), 333–340. https://doi.org/10.31004/jutin.v7i1.24749

Ardiyansah, D., Pahlevi, O., & Santoso, T. (2021). IMPLEMENTASI METODE PROTOTYPING PADA SISTEM INFORMASI PENGADAAN BARANG CETAKAN BERBASIS WEB. *Hexagon Jurnal Teknik dan Sains*, *2*(2). https://doi.org/10.36761/hexagon.v2i2.1083

Arianto, R., Al Anam, A. K., Devi, B., & Rachman, A. (2021). PENGEMBANGAN APLIKASI SISTEM INFORMASI INVENTORY PADA CV WIJAYA LAS KEDIRI MENGGUNAKAN MODEL WATERFALL. *Jurnal SAINTIKOM (Jurnal Sains Manajemen Informatika dan Komputer)*, *20*(2). https://doi.org/10.53513/jis.v20i2.3749

Arvindkumar Patel, Dr. P. (2024a). An Analysis Of The Mobile App Development Industry. *Revista Electronica de Veterinaria*, 604–611. https://doi.org/10.69980/redvet.v25i1S.790

Aulia, R., & Setiyani, H. (2024). Aplikasi Pengelolaan Inventaris Kampus STTI NIIT Berbasis Website. *Prosiding TAU SNARS-TEK Seminar Nasional Rekayasa dan Teknologi*, *3*(1), 211–217. https://doi.org/10.47970/snarstek.v2i1.572

Badrul, M. (2021). Penerapan Metode waterfall untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang. *PROSISKO: Jurnal Pengembangan Riset dan Observasi Sistem Komputer*, *8*(2), 57–52. https://doi.org/10.30656/prosisko.v8i2.3852

Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An Empirical Evaluation of the System Usability Scale. *International Journal of Human-Computer Interaction*, *24*(6), 574–594. https://doi.org/10.1080/10447310802205776

Brooke, J. (1996). SUS -A quick and dirty usability scale Usability and context. *Usability evaluation in industry*, *189*(194).

Chin Wei, C., Ramiah, P., & Farhaini Razali, N. (2023). Inventory Management Systems (IMS). Dalam *Journal of Applied Technology and Innovation* (Vol. 7, Nomor 3).

Fisa Wisnu Wijaya, & Lomban, D. (2022). SISTEM INFORMASI INVENTORY BARANG MENGGUNAKAN METODE WATERFALL. *Jurnal Informatika, Teknologi dan Sains*, *4*(3). https://doi.org/10.51401/jinteks.v4i3.1963

Hizkia, D., Widianto, Y., & Kunci, K. (2024). Rancang Bangun Sistem Informasi Manajemen Inventaris Barang Pada Kampus Widya Kartika. Dalam *Surabaya Jurnal Sistem Cerdas dan Rekayasa (JSCR)* (Vol. 6, Nomor 2).

Khan, F. Q., Rasheed, S., Alsheshtawi, M., Ahmed, T. M., & Jan, S. (2020). A comparative analysis of RAD and agile technique for management of computing graduation projects. *Computers, Materials and Continua*, *64*(2). https://doi.org/10.32604/CMC.2020.010959

Melissourgos George. (2023, Januari 29). *The positive SUS: as good as the original*. Blucado.

Nurhadi, & Muhammad Ridwan. (2022). Sistem Informasi Inventaris Berbasis Web Menggunakan Metode Prototype. *Jurnal Multidisiplin Madani*, *2*(9). https://doi.org/10.55927/mudima.v2i9.1143

Nuroji, N. (2023). Penerapan Metode Agile Dalam Permodelan Sistem Informasi Inventory Barang. *Journal of Artificial Intelligence and Technology Information*, *1*(4).

Nuryasin, N., Saputra, A. H., Hardi, T., & Fadholur, I. (2021). Analisis dan Perancangan Sistem Informasi Inventory Barang pada PT Cipta Rasa Multindo S. *Applied Information System and Management (AISM)*, *2*(1). https://doi.org/10.15408/aism.v2i1.20205

Pankaj Chandra, Santosh Soni, Akanksha Gupta, Sushant Kumar, Kaushik Tiwary, & Deepak Kant Netam. (2023). Design and Implementation of Inventory Management System for University. *European Chemical Bulletin*.

Prasetyaningsih, S., & Muchtar, S. P. N. (2023). Analisis Perbandingan User Experience pada Website dan Aplikasi Mobile Shopee Menggunakan UEQ. *JTIM : Jurnal Teknologi Informasi dan Multimedia*, *5*(3). https://doi.org/10.35746/jtim.v5i3.326

Ramadhan, J. A., & Andrianingsih, A. (2024). *Implementasi Metode User Centered Design (UCD) Pada Web Inventori Toko KickID* (Vol. 13, Nomor 1).

Retno, S., Agusniar, C., & Sinambela, I. S. (2023). Sistem Pendataan Inventaris Barang Pada Program Studi Teknik Informatika Universitas Malikussaleh. *Sisfo: Jurnal Ilmiah Sistem Informasi*, *7*(2), 29. https://doi.org/10.29103/sisfo.v7i2.13944

Rianto, H., & Amrin, A. (2023). Rancang Bangun Sistem Informasi Inventory Menggunakan Metode Rapid Application Development. *INSANTEK*, *4*(1). https://doi.org/10.31294/instk.v4i1.1942

Ridho, M. R., Pinandito, A., & Dewi, R. K. (2018). Perbandingan Performa Progressive Web Apps dan Mobile Web Terkait Waktu Respon, Penggunaan Memori dan Penggunaan Media Penyimpanan. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, *2*(10).

Sauro, J., & Lewis, J. R. (2011). When designing usability questionnaires, does it hurt to be positive? *Conference on Human Factors in Computing Systems - Proceedings*. https://doi.org/10.1145/1978942.1979266

Senarath, U. S. (2021). Waterfall methodology, prototyping and agile development. *Tech. Rep.*, *June*.

Setiawan, R., Kurniadi, D., Aulawi, H., & Kurniawati, R. (2019). Asset management information system for higher education. *Journal of Physics: Conference Series*, *1402*(2), 022083. https://doi.org/10.1088/1742-6596/1402/2/022083

Shania, M., Raharjo, T., & Nur Fitriani, A. (2023). Implementation User-Centered Design in Agile Software Development: Systematic Literature Review. *Indonesian Journal of Multidisciplinary Science*, *2*(7). https://doi.org/10.55324/ijoms.v2i7.480

Sholahuddin Alfarisy. (2024). *Perancangan Sistem Informasi Inventaris Barang Berbasis Website Menggunakan Metode User Centered Design: Studi Kasus BAZNAS Kota Depok* [Sekolah Tinggi Teknologi Terpadu Nurul Fikri]. https://repository.nurulfikri.ac.id/id/eprint/600

Sidik, M. P., Supriatman, A., Supriatman, A., Ramadhan, T. I., & Ramadhan, T. I. (2024). RANCANG BANGUN SISTEM INFORMASI INVENTARIS BARANG MENGGUNAKAN METODE AGILE DI SEKOLAH MENENGAH KEJURUAN BINA PUTERA NUSANTARA. *Jurnal Informatika dan Teknik Elektro Terapan*, *12*(3). https://doi.org/10.23960/jitet.v12i3.4370

Tarigan, E., Damayanti, D., & Pratama, F. (2024). PENGEMBANGAN SISTEM INVENTORY CONTROL DENGAN METODE USER-CENTERED DESIGN DAN EVALUASI USABILITY. Dalam *Jurnal Jaringan Sistem Informasi Robotik (JSR)* (Vol. 8, Nomor 2). http://ojsamik.amikmitragama.ac.id

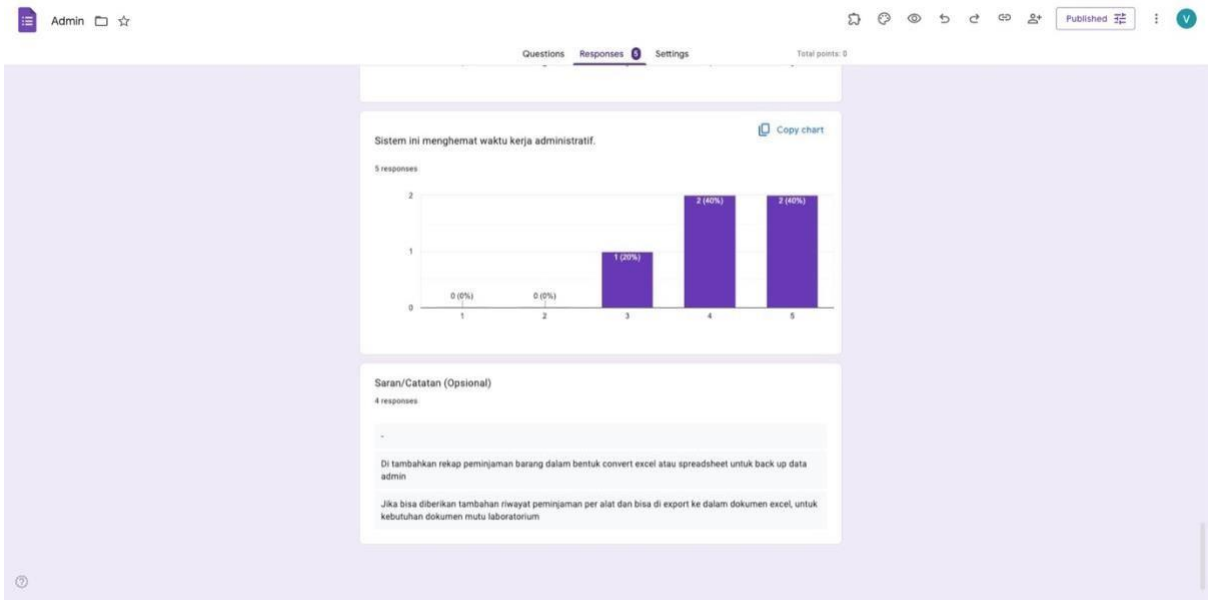
Wibowo, F., Rolansa, F., & Bowo Atmojo, T. (2021). Sistem Informasi Inventaris Untuk Peminjaman dan Pengembalian Alat di Laboratorium Teknik Informatika Polnep Berbasis Web. *Jurnal ELIT*, *2*(2). https://doi.org/10.31573/elit.v2i2.206

Widiarta, I. M., Mulyanto, Y., & Sutrianto, A. (2023). Rancang Bangun Sistem Informasi Inventory Menggunakan Metode Agile Software Development (Studi Kasus Toko Nada). *Digital Transformation Technology (Digitech)*, *3*(Maret).

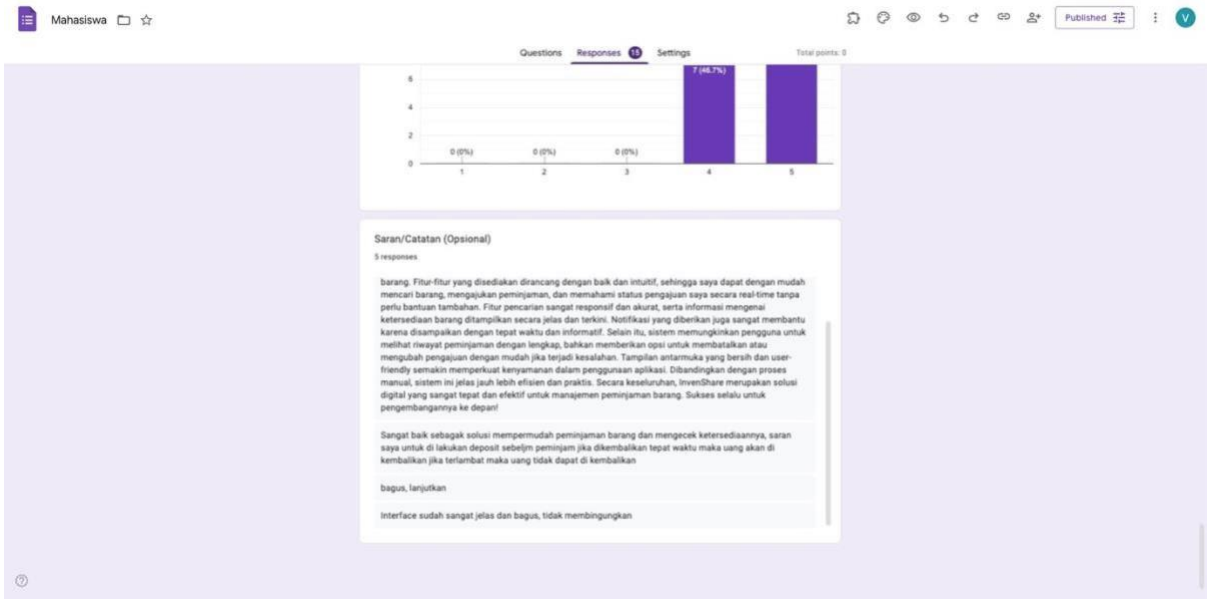
Yahya, A., Amali, L. N., & Padiku, I. R. (2024). PENGEMBANGAN SISTEM APLIKASI MANAJEMEN ASET BERBASIS ANDROID DI UNIVERSITAS NEGERI GORONTALO. *Diffusion: Journal of Systems and Information Technology*, *4*, 92–102.

LAMPIRAN

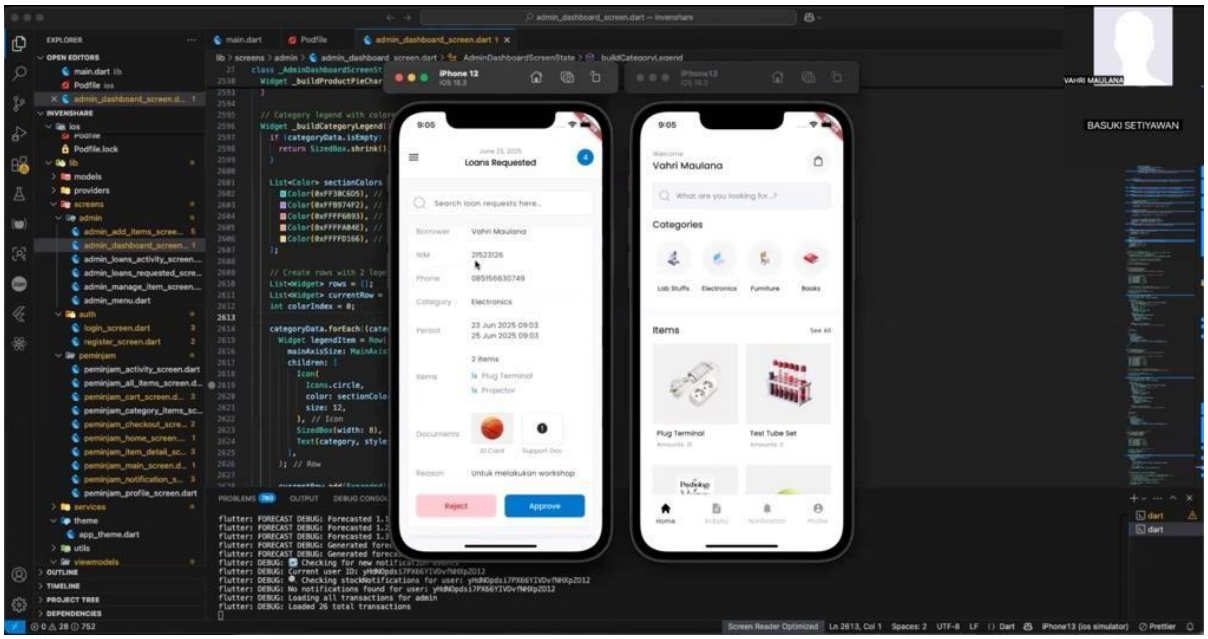
Lampiran A. *Capture* Masukan Mengenai Sistem dari Responden Admin



Lampiran B. *Capture* Masukan Mengenai Sistem dari Responden Mahasiswa

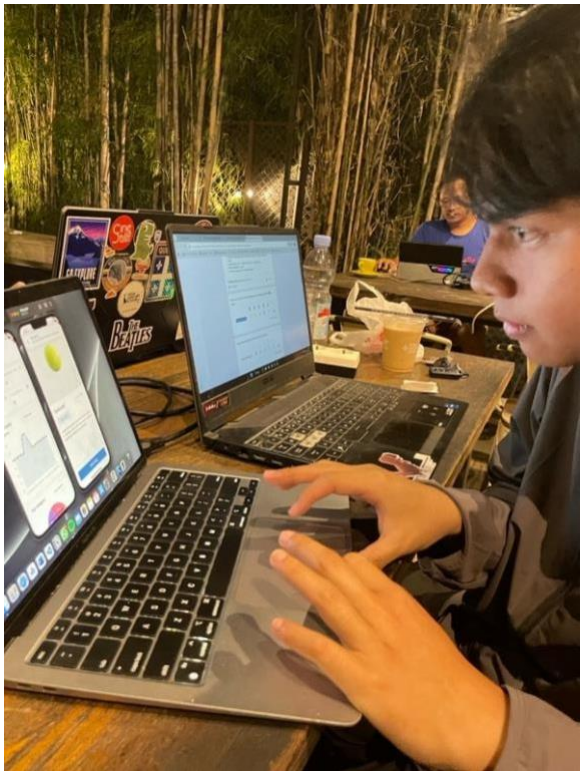


Lampiran C. Pengujian Bersama Staf DAURT FTI



Lampiran D. Pengujian Bersama Mahasiswa




The Beatles

