

**RANCANG BANGUN SISTEM INFORMASI MANAJEMEN
INVENTARIS BARANG BERBASIS WEB**

(Studi Kasus: Taman Kanak-Kanak Teratai Cirebon)



Diajukan oleh :
EDWARDO MANHAZ RIZQIA
5220311130

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS & TEKNOLOGI
UNIVERSITAS TEKNOLOGI YOGYAKARTA
2025**

LEMBAR PENGESAHAN
PROYEK PENGEMBANGAN SISTEM INFORMASI
RANCANG BANGUN SISTEM INFORMASI MANAJEMEN
INVENTARIS BARANG BERBASIS WEB
(Studi Kasus: Taman Kanak-Kanak Teratai Cirebon)

disusun oleh

Edwardo Manhaz Rizqia

5220311130

Telah diujikan

pada tanggal 26 Januari 2026

Ketua Penguji	Anggota Penguji
Nama Ketua Penguji/Pembimbing	Nama Anggota Penguji

Yogyakarta,
Ketua Program Studi Sistem Informasi
Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta

Adityo Permana Wibowo, S.Kom., M.Cs.
NIK 110116079

LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Edwardo Manhaz Rizqia

NIM : 5220311130

Program Studi : Sistem Informasi

Menyatakan bahwa laporan ISCP yang berjudul Rancang Bangun Sistem Informasi Manajemen Berbasis Web (Studi Kasus: Taman Kanak-Kanak Teratai Cirebon). Merupakan karya ilmiah asli saya dan belum pernah dipublikasi oleh orang lain, kecuali yang tertulis sebagai acuan dalam naskah ini dan disebutkan dalam daftar pustaka.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Yogyakarta, 12 Juni 2025

Edwardo Manhaz Rizqia

SURAT KETERANGAN ISCP

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga laporan ISCP ini dengan judul Rancang Bangun Sistem Informasi Manajemen Inventaris Barang Berbasis Web (Studi Kasus: Taman Kanak-Kanak Teratai Cirebon) dapat diselesaikan dengan baik. Laporan ini disusun sebagai salah satu syarat untuk melaksanakan ISCP pada Program Studi Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta.

Penulis mengucapkan terima kasih kepada pihak-pihak yang telah memberikan dukungan, khususnya kepada:

1. Dr. Bambang Moertono S., MM., Akt., CA., sebagai Rektor Universitas Teknologi Yogyakarta.
2. Dr. Endy Marlina, M.T., sebagai Dekan Fakultas Sains & Teknologi, Universitas Teknologi Yogyakarta.
3. Adityo Permana Wibowo, S.Kom., M.Cs., sebagai Program Studi Sistem Informasi.
4. Sutarman,M.Kom, Ph.D., sebagai Dosen Pembimbing ISCP
5. Bapak/Ibu Dosen Pembimbing yang telah memberikan arahan dan masukan.
6. Pihak Taman Kanak-Kanak Teratai Cirebon yang telah bersedia menjadi mitra ISCP.

Penulis menyadari bahwa Laporan ini masih memiliki kekurangan. Oleh karena itu, penulis sangat terbuka terhadap kritik dan saran untuk perbaikan di masa mendatang. Semoga Laporan ini dapat memberikan kontribusi positif dalam pengembangan sistem informasi dan menjadi manfaat bagi pihak yang berkepentingan.

Yogyakarta, 12 Juni 2025

Penulis

ABSTRAK

Perkembangan teknologi informasi mendorong efisiensi dalam berbagai sektor, termasuk pendidikan. Taman Kanak-Kanak Teratai Cirebon selama ini mengelola inventaris barang secara manual yang rentan terhadap kesalahan pencatatan, kehilangan data, dan keterbatasan akses informasi. Untuk mengatasi permasalahan tersebut, dirancang dan dibangun sebuah Sistem Informasi Manajemen Inventaris Barang berbasis web. Sistem ini dikembangkan menggunakan framework Laravel dan basis data MySQL, serta diimplementasikan dengan pendekatan berbasis peran (admin, petugas, dan kepala sekolah). Fitur utama sistem meliputi pencatatan barang masuk dan keluar, monitoring stok, laporan bulanan, dan pengelolaan pengguna. Pengujian sistem dilakukan menggunakan metode black-box yang menunjukkan bahwa seluruh fungsi berjalan sesuai spesifikasi, dengan respon cepat dan tampilan antarmuka yang user-friendly. Hasil implementasi membuktikan bahwa sistem ini mampu meningkatkan efisiensi, akurasi, dan transparansi dalam pengelolaan inventaris di lingkungan sekolah.

Kata kunci: Sistem Informasi, Inventaris Barang, Web, Laravel, TK Teratai Cirebon

ABSTRACT

The advancement of information technology has significantly improved efficiency across various sectors, including education. Taman Kanak-Kanak Teratai Cirebon previously managed its inventory manually, which often led to data inaccuracies, loss of records, and limited access to information. To address these issues, a web-based Inventory Management Information System was designed and developed. The system was built using the Laravel framework and MySQL database, implementing role-based access for administrators, inventory staff, and school principals. Key features include item inflow and outflow recording, stock monitoring, monthly reporting, and user management. The system was tested using black-box testing, and the results showed that all features functioned correctly with fast response time and a user-friendly interface. The implementation proved that the system successfully enhances efficiency, accuracy, and transparency in managing school inventory.

Keywords: *Information System, Inventory, Web-Based, Laravel, TK Teratai Cirebon*

DAFTAR ISI

LEMBAR PERNYATAAN	iii
SURAT KETERANGAN ISCP	iv
KATA PENGANTAR.....	v
ABSTRAK	vi
<i>ABSTRACT</i>	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	x
DAFTAR GAMBAR	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan ISCP	2
1.5 Manfaat ISCP	3
1.6 Tahapan ISCP	3
1.7 Sistematika Penulisan	5
BAB II KAJIAN HASIL PENELITIAN DAN LANDASAN TEORI	8
2.1 Kajian Hasil Penelitian.....	8
2.2 Landasan Teori	10
2.2.1 Sistem.....	10
2.2.2 Informasi	10
2.2.3 Sistem Informasi	10
2.2.4 Rancang Bangun	11
2.2.5 Flowchart	11
2.2.6 Sistem Informasi Inventory.....	12
2.2.7 Website	13
2.2.8 <i>Unified Modelling Language</i>	13
2.2.9 Pengujian Blackbox	16
BAB III TINJAUAN UMUM INSTANSI	18
3.1 Gambaran Umum	18
3.1.1 Visi &Misi	19
3.2 Struktur Organisasi.....	20
3.3 Produk / Jasa	22
3.4 Proses Bisnis	22
3.4.1 Alur Bisnis.....	23
BAB IV ANALISIS DAN PERANCANGAN	25
4.1 Analisis Sitem	25
4.2 Analisa Kebutuhan	25
4.3 Perancangan Sistem	26
4.3.1 Perancangan Model Proses	26
4.3.2 Perancangan Model Data	54
4.3.3 Perancangan Fisik Basis Data	55
4.3.4 Perancangan Relasi Antar Tabel.....	58
4.3.5 Perancangan Antarmuka.....	58

BAB V IMPLEMENTASI DAN PEMBAHASAN SISTEM.....	63
5.1. Implementasi.....	63
5.1.1 Implementasi Basis Data.....	63
5.1.2 Implementasi Sistem	66
5.2 Pembahasan.....	71
5.2.1 Pembahasan Basis Data	71
5.2.2. Pembahasan Sistem.....	81
5.2.3 Pengujian.....	90
BAB IV PENUTUP	96
6.1 Simpulan	96
6.2 Saran	97
DAFTAR PUSTAKA	99

DAFTAR TABEL

Tabel 2.1 Simbol Flowchart.....	11
Tabel 2.2 Simbol Use Case Diagram.....	14
Tabel 2.3 Simbol Class Diagram	15
Tabel 2.4 Simbol Activity Diagram.....	15
Tabel 2.5 Simbol Squence Diagram	16
Tabel 4.1 Tabel Users	55
Tabel 4. 2 Tabel Kategori.....	55
Tabel 4. 3 Tabel Lokasi.....	56

DAFTAR GAMBAR

Gambar 3.1 Bangunan TK Teratai	18
Gambar 3.2 Denah Lokasi	19
Gambar 3.3 Bagan Struktur Organisasi	20
Gambar 4.1 Use Case Diagram	27
Gambar 4.2 Activity Diagram Admin Sistem Login	28
Gambar 4.3 Activity Diagram Admin Sistem Kelola Kategori Barang	29
Gambar 4.4 Activity Diagram Admin Sistem Monitoring Aktivitas Pengguna ..	30
Gambar 4.5 Activity Diagram Admin Ssitem Reset Password	31
Gambar 4.6 Activity Diagram Admin Sistem <i>Backup</i> Dan Restore Data	32
Gambar 4.7 Activity Diagram Petugas Inventaris Edit Data Barang	33
Gambar 4.8 Activity Diagram Petugas Inventaris Cek Stok Barang.....	34
Gambar 4.9 Activity Diagram Petugas Inventaris Lihat Barang Rusak Atau Hilang	35
Gambar 4.10 Petugas Inventaris Cetak Laporan Bulanan Inventaris	36
Gambar 4.11 Activity Diagram Kepala Sekolah Lihat Laporan.....	37
Gambar 4.12 Activity Diagram Kepala Sekolah Lihat Data Barang Rusak Atau Hilang.....	38
Gambar 4.13 Activity Diagram Kepala Sekolah Monitoring Stok Barang	39
Gambar 4.14 Activity Diagram Kepala Sekolah Approve Permintaan Pengadaan Barang	40
Gambar 4.15 Squence Diagram Admin Sistem Login	41
Gambar 4.16 Squence Diagram Admin Ssitem Memilih Kategori Barang.....	42
Gambar 4.17 Squence Diagram Admin Sistem Monitoring Barang	43
Gambar 4.18 Squence Diagram Admin Sistem Reset Password.....	44
Gambar 4.19 Squence Diagram Admin Buka Menu	45
Gambar 4.20 Squence Diagram Petugas Inventaris Edit Barang	46
Gambar 4.21 Squence Diagram Petugas Inventaris Cek Stok Barang	47
Gambar 4.22 Squence Diagram Petugas Inventaris Melihat Barang Rusak AtauHilang	48
Gambar 4.23 Squence Diagram Petugas Inventaris Cetak Laporan.....	49
Gambar 4.24 Squence Diagram Petugas Inventaris Melihat Laporan.....	50
Gambar 4.25 Squence DIagram Kepala Sekolah Melihat Laporan.....	51
Gambar 4.26 DIagram Kepala Sekolah Lihat Barang Rusak Atau Hilang	52
Gambar 4.27 DIagram Kepala Sekolah Approve Pengadaan Barang	53
Gambar 4.28 Class Diagram.....	54
Gambar 4.29 Antarmuka Kelola Data Barang.....	59
Gambar 4.30 Antarmuka Kelola Data Pengguna	60
Gambar 4.31 Antarmuka Kelola Laporan	61
Gambar 4.32 Antarmuka Halaman Dashboard.....	61
Gambar 4.33 Antarmuka Halaman Aktivitas Log	62
Gambar 5.1 Tabel Pada Database	66

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang pesat telah membawa dampak signifikan dalam berbagai aspek kehidupan, termasuk dalam pengelolaan administrasi di sektor pendidikan. Salah satu bentuk pemanfaatan teknologi informasi adalah dalam sistem manajemen inventaris barang, yang berperan penting dalam memastikan ketersediaan dan pemeliharaan sarana serta prasarana pendidikan.

Taman Kanak-Kanak Teratai Cirebon merupakan salah satu institusi pendidikan yang memiliki kebutuhan untuk meningkatkan efisiensi dalam pengelolaan inventory barang. Saat ini, pencatatan dan pengelolaan barang di sekolah masih dilakukan secara manual dengan menggunakan dokumen fisik dan pencatatan tertulis. Metode ini sering kali menghadirkan berbagai permasalahan, seperti kesalahan dalam pencatatan, kesulitan dalam melacak ketersediaan barang, risiko kehilangan atau kerusakan dokumen, serta keterbatasan akses informasi bagi pihak yang berkepentingan.

Untuk mengatasi permasalahan tersebut, diperlukan sebuah sistem informasi inventaris barang berbasis web yang dapat mengelola data barang secara lebih modern dan efisien. Sistem ini diharapkan dapat memberikan berbagai manfaat, seperti mempermudah pencatatan dan pelacakan barang, mempercepat proses inventarisasi, mengurangi risiko kehilangan atau kesalahan pencatatan, serta memungkinkan akses informasi secara *real-time* bagi pihak yang bertanggung jawab.

Dengan adanya sistem informasi berbasis web ini, TK Teratai Cirebon dapat meningkatkan kualitas pengelolaan sarana dan prasarana pendidikan serta menciptakan administrasi yang lebih tertata dan transparan. Oleh karena itu, penelitian ini bertujuan untuk melakukan mengimplementasikan dan mengevaluasi sistem informasi inventory barang berbasis web yang dapat memenuhi kebutuhan

operasional sekolah sekaligus memberikan kemudahan dalam pengelolaan data inventaris.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut diperoleh rumusan masalah tersebut, penulis menyusun masalah yakni bagaimana merancang dan membangun sistem inventaris barang berbasis web yang dapat memenuhi kebutuhan Taman Kanak-Kanak Teratai Cirebon?

1.3 Batasan Masalah

Agar penelitian ini dapat dilakukan dengan fokus dan terarah, beberapa batasan masalah ditentukan sebagai berikut:

1. Sistem informasi yang dirancang hanya berfokus pada proses pengelolaan inventory barang di Taman Kanak-Kanak Teratai Cirebon, mulai dari pencatatan barang masuk, pencatatan barang keluar, hingga monitoring stok barang.
2. Sistem hanya mencakup fitur utama, seperti pencatatan barang masuk dan keluar, pengelolaan kategori barang, pencarian dan pelacakan stok barang, serta pembuatan laporan inventaris.

1.4 Tujuan ISCP

Tujuan dari pelaksanaan ISCP ini adalah sebagai berikut:

1. Menganalisis kebutuhan sistem informasi inventory barang di Taman Kanak-Kanak Teratai Cirebon, dengan mengidentifikasi kendala yang dihadapi dalam proses pencatatan dan pengelolaan inventaris secara manual saat ini.
2. Merancang sistem informasi inventory barang berbasis web yang dapat mempermudah dan mempercepat proses pencatatan barang masuk dan keluar, pemantauan stok, serta pembuatan laporan inventaris.
3. Membangun sistem informasi inventory barang sesuai dengan rancangan yang telah dihasilkan.

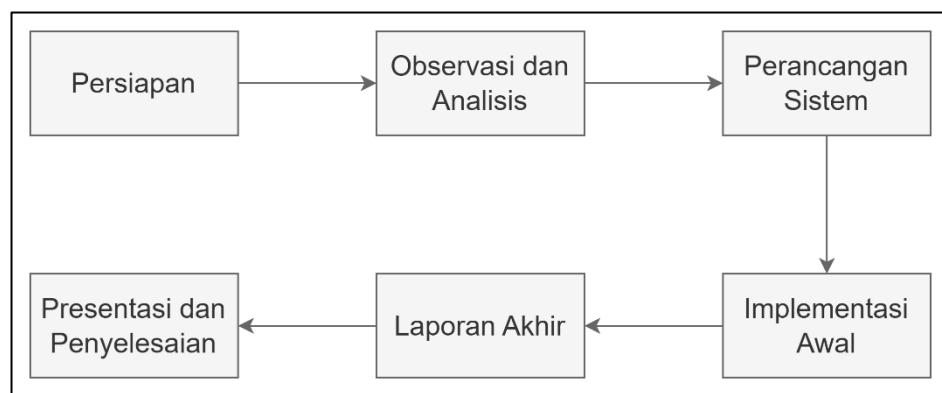
1.5 Manfaat ISCP

Pelaksanaan ISCP ini diharapkan memberikan manfaat sebagai berikut:

1. Mempermudah dan mempercepat proses pencatatan barang masuk dan keluar secara digital.
2. Memungkinkan staf sekolah untuk memantau stok barang secara *real-time* tanpa harus melakukan pencatatan manual yang rentan terhadap kesalahan.
3. Meningkatkan efisiensi dalam pengelolaan sarana dan prasarana sekolah dengan sistem yang lebih terstruktur dan terdokumentasi dengan baik.
4. Mengurangi risiko kehilangan atau kekeliruan dalam pencatatan inventaris barang karena data tersimpan dalam sistem berbasis digital.
5. Mempermudah pembuatan laporan inventaris secara otomatis, sehingga menghemat waktu dan tenaga staf administrasi.
6. Meningkatkan transparansi dalam pengelolaan barang, sehingga memudahkan pihak manajemen dalam mengambil keputusan terkait pengadaan atau pemeliharaan barang.

1.6 Tahapan ISCP

Pelaksanaan ISCP di lakukan secara bertahap sesuai dengan alur gambar berikut :



Gambar 1.6 Tahapan ISCP

Berdasarkan **Gambar 1.6** penjabaran tahapan ISCP sebagai berikut:

1. Persiapan

Tahap ini mencakup kegiatan awal seperti menyusun laporan ISCP, mengidentifikasi tujuan proyek, dan mendapatkan persetujuan dari pihak terkait. Mengumpulkan informasi awal tentang proses yang akan dianalisis. Selain itu, koordinasi dengan instansi terkait dilakukan untuk menetapkan jadwal dan detail kegiatan.

2. Observasi dan Analisis

Pada tahap ini, dilakukan pengamatan langsung terhadap sistem inventaris manual yang berjalan di TK Teratai Cirebon. Wawancara dengan petugas sekolah, guru, dan kepala sekolah dilakukan untuk mengidentifikasi kebutuhan serta kendala dalam pengelolaan inventaris barang. Data yang diperoleh digunakan untuk menganalisis kekurangan sistem lama dan mengidentifikasi peluang perbaikan yang dapat diimplementasikan dalam sistem baru.

3. Perancangan Sistem

Berdasarkan hasil analisis, dilakukan perancangan sistem informasi manajemen inventaris barang yang mencakup dokumentasi kebutuhan pengguna. Diagram UML, seperti use case diagram dan activity diagram, digunakan untuk memvisualisasikan alur sistem yang baru. Selain itu, rancangan prototipe antarmuka disusun untuk memberikan gambaran awal tampilan sistem kepada pengguna sebelum implementasi.

4. Implementasi Awal

Mengembangkan sistem informasi berdasarkan desain yang telah dirancang sebelumnya. Prototipe atau versi awal sistem dibuat untuk diuji oleh pengguna, seperti petugas inventaris atau staf administrasi sekolah. Uji coba ini membantu dalam mengidentifikasi kesalahan serta fitur yang perlu diperbaiki. Hasil dari implementasi awal menjadi dasar untuk penyempurnaan sistem hingga siap digunakan secara penuh.

5. Laporan Akhir

Setelah uji coba, menyusun laporan ISCP yang mencakup latar belakang, analisis, perancangan, dan implementasi sistem. Laporan ini juga memuat kesimpulan dan saran perbaikan berdasarkan hasil ISCP.

6. Presentasi dan Penyelesaian

Tahap terakhir adalah mempresentasikan hasil ISCP kepada pihak kampus dan instansi terkait. Masukan dari pembimbing dan pengguna digunakan untuk melakukan revisi akhir pada sistem atau laporan.

1.7 Sistematika Penulisan

Untuk memberikan gambaran menyeluruh dan terstruktur mengenai isi dari laporan ISCP ini, penulisan disusun dalam beberapa bab utama yang masing-masing membahas aspek-aspek penting terkait kegiatan ISCP. Sistematika penulisan laporan ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini memuat uraian mengenai latar belakang permasalahan yang menjadi dasar dilakukannya ISCP, rumusan masalah yang hendak diselesaikan, batasan masalah guna menjaga fokus pembahasan, serta tujuan dan manfaat yang diharapkan dari pelaksanaan ISCP. Selain itu, bab ini juga memuat sistematika penulisan sebagai panduan struktur dokumen secara keseluruhan.

BAB II KAJIAN HASIL PENELITIAN DAN LANDASAN TEORI

Bab ini berisi kajian terhadap penelitian-penelitian terdahulu yang relevan sebagai referensi dan pembanding terhadap penelitian yang akan dilakukan. Selain itu, dibahas pula landasan teori yang mendasari penyusunan sistem informasi, meliputi konsep sistem, informasi, sistem informasi, rancang bangun sistem, flowchart, sistem informasi inventory, website, dan pemodelan UML yang akan digunakan dalam proses perancangan.

BAB III TINJAUAN UMUM INSTANSI

Dalam bab ini dijelaskan mengenai profil dari instansi tempat ISCP dilakukan, dalam hal ini Taman Kanak-Kanak Teratai Cirebon. Pembahasan mencakup gambaran umum sekolah, visi dan misi, struktur organisasi, serta proses bisnis utama yang dijalankan oleh institusi. Informasi ini penting sebagai konteks awal dalam merancang sistem yang sesuai dengan kebutuhan instansi.

BAB IV METODE ISCP

Bab ini menguraikan metodologi ISCP yang meliputi tahapan-tahapan pelaksanaan, mulai dari tahap persiapan, observasi dan analisis kebutuhan, perancangan sistem, implementasi awal, hingga penyusunan laporan akhir. Selain itu, dijelaskan pula sumber data yang digunakan dalam ISCP serta jadwal kegiatan yang dirancang untuk memastikan proyek berjalan sesuai rencana dan waktu yang ditentukan.

BAB V IMPLEMENTASI DAN PEMBAHASAN SISTEM

Bab ini menyajikan uraian detail mengenai pelaksanaan penerapan sistem informasi manajemen inventaris barang di TK Teratai Cirebon beserta analisis hasilnya. Paparan diawali dengan penjelasan lingkungan implementasi spesifikasi perangkat keras, perangkat lunak, serta konfigurasi jaringan yang digunakan. Selanjutnya ditampilkan tangkapan layar antarmuka utama dan modul-modul penting sistem, disertai penjelasan alur kerjanya. Bagian uji fungsional memeriksa setiap fitur, mulai dari pencatatan barang masuk atau keluar hingga pembuatan laporan stok, menggunakan skenario pengujian black-box. Hasil pengujian dianalisis melalui metrik akurasi data, kecepatan proses, dan tingkat kebergunaan.

BAB IV PENUTUP

Bab terakhir merangkum keseluruhan temuan ISCP sekaligus menawarkan rekomendasi pengembangan lebih lanjut. Bagian kesimpulan memuat pemenuhan tujuan proyek, manfaat nyata yang dirasakan pihak sekolah, serta kontribusi akademis penelitian. Selanjutnya disajikan saran teknis seperti penambahan modul notifikasi otomatis, integrasi perangkat mobile, dan penguatan keamanan data dan saran non teknis, di antaranya pelatihan rutin bagi staf serta penetapan kebijakan pemeliharaan sistem. Bab ini juga memetakan rencana pengembangan ke depan agar sistem dapat ditingkatkan skalabilitasnya dan menjadi dasar bagi riset lanjutan terkait manajemen inventaris di lingkungan pendidikan.

BAB II

KAJIAN HASIL PENELITIAN DAN LANDASAN TEORI

2.1 Kajian Hasil Penelitian

Dalam penelitian oleh Murod dkk., (2024), penelitiannya mengembangkan sistem informasi manajemen pergudangan untuk PT. Jazeera Inti Sukses, yang sebelumnya mengandalkan kartu stok dan Excel, menyebabkan ketidakefisienan dalam pengelolaan stok barang. Sistem yang dirancang menggunakan pendekatan prototyping dengan framework Laravel bertujuan untuk meningkatkan akurasi data barang masuk dan keluar serta mempermudah pengelolaan stok dan pemesanan. Evaluasi dilakukan dengan metode Webuse, sementara pemodelan sistem menggunakan UML dengan pendekatan ICONIX. Hasil pengujian blackbox menunjukkan sistem berfungsi sesuai harapan, dengan skor usabilitas 0,89 (baik), diharapkan dapat membantu manajer, kepala gudang, dan operator dalam mengelola gudang secara lebih efisien.

Dalam penelitian oleh Dwi Putri dkk., (2022), ini dilakukan di SMP Negeri 01 Runjung Agung yang masih mengelola inventaris barang secara manual menggunakan buku, menyebabkan kesulitan dalam pencatatan dan pelaporan. Untuk mengatasi masalah ini, penelitian mengembangkan sistem informasi inventaris berbasis website menggunakan PHP, MySQL, dan framework CodeIgniter dengan metode prototype. Sistem ini dirancang untuk membantu staf sarana dan prasarana dalam mencatat data barang masuk, jumlah barang, barang rusak, serta menyusun laporan secara efisien dan *real-time*. Hasil pengujian dengan metode black box menunjukkan bahwa sistem yang dikembangkan dalam penelitian ini berjalan dengan baik, mampu mengolah data, dan menyajikan informasi secara akurat.

Dalam penelitian oleh Sidik dkk., (2024), Sekolah Menengah Kejuruan Bina Putera Nusantara masih menggunakan pencatatan manual untuk inventaris

barang, yang kurang efektif dan efisien. Untuk mengatasi permasalahan ini, dikembangkan sistem informasi inventaris barang menggunakan metode Agile, dengan tahapan perencanaan, analisis dan desain, assembling, coding, testing, serta release. Pengujian dilakukan menggunakan *System Usability Scale* (SUS) untuk mengukur efektivitas sistem, dengan hasil rata-rata skor 72,75 yang termasuk dalam kategori "Good". Sistem ini diharapkan dapat meningkatkan efisiensi dalam pencatatan dan pengelolaan inventaris barang di sekolah.

Penelitian oleh Riza dkk., (2024), ini dilakukan di Kantor Perbekel Desa Kediri Tabanan yang masih mengelola inventaris barang secara manual, sehingga rentan terhadap kesalahan dan kehilangan data. Untuk mengatasi masalah ini, penelitian mengembangkan sistem informasi pengelolaan inventaris berbasis web menggunakan metodologi waterfall dan framework Laravel. Tahapan penelitian mencakup analisis kebutuhan, perancangan sistem, implementasi, pengujian dengan metode Black Box Testing, serta evaluasi. Hasil penelitian menunjukkan bahwa sistem ini mampu meningkatkan efisiensi, keakuratan, transparansi, dan akuntabilitas dalam pengelolaan inventaris barang, serta meningkatkan kepuasan pengguna dalam pencatatan dan pemantauan aset secara lebih terstruktur.

Dalam penelitian oleh Triana dkk., (2024), dikembangkan sistem informasi persediaan barang berbasis web dikembangkan untuk mengatasi kendala pencatatan manual di Toko Kuat yang masih menggunakan perhitungan fisik dan nota biasa, menyebabkan keterlambatan informasi stok serta potensi kesalahan data. Dengan metode Waterfall dan model Unified Modeling Language (UML), sistem ini dirancang menggunakan PHP dan MySQL untuk meningkatkan efisiensi pengelolaan barang masuk, barang keluar, serta laporan persediaan. Implementasi sistem ini memungkinkan pemilik toko memantau stok secara *real-time*, mempercepat pencarian data barang dan supplier, serta mengurangi risiko kehilangan atau ketidaksesuaian data, sehingga meningkatkan efektivitas operasional toko.

2.2 Landasan Teori

2.2.1 Sistem

Sistem adalah serangkaian elemen yang saling terhubung untuk mencapai tujuan tertentu. Elemen-elemen dalam sistem mencakup *input*, proses, *output*, dan pengendalian, yang semuanya bekerja secara sinergis untuk menghasilkan hasil yang diinginkan (Mukhlis dkk., 2024). Sistem dapat berbentuk fisik, seperti perangkat keras, atau abstrak, seperti prosedur dan kebijakan. Setiap elemen memiliki peran penting dalam mendukung kinerja keseluruhan sistem. Sistem yang dirancang dengan baik harus memiliki kejelasan tujuan, efisiensi proses, dan kemampuan adaptasi terhadap perubahan lingkungan. Hal ini memungkinkan sistem untuk tetap relevan dan bermanfaat dalam berbagai konteks, baik organisasi maupun teknologi.

2.2.2 Informasi

Informasi merupakan hasil pengolahan data mentah menjadi bentuk yang lebih bermakna dan berguna. Mukhlis dkk. (2024) menjelaskan bahwa informasi berkualitas harus memenuhi tiga kriteria utama, yaitu relevansi, akurasi, dan ketepatan waktu. Informasi digunakan untuk mendukung pengambilan keputusan di berbagai level, mulai dari operasional hingga strategis. Misalnya, data yang diolah dalam sistem informasi manajemen dapat membantu manajer dalam menganalisis tren, memprediksi hasil, dan mengevaluasi performa. Oleh karena itu, informasi berfungsi sebagai dasar penting untuk mendukung fungsi manajemen dan operasional yang efektif.

2.2.3 Sistem Informasi

Sistem informasi adalah kombinasi teknologi, manusia, dan proses yang dirancang untuk mengelola data menjadi informasi yang relevan. Menurut Mukhlis dkk. (2024), sistem informasi mendukung aktivitas organisasi dengan menyediakan informasi yang akurat dan dapat diandalkan untuk pengambilan keputusan. Komponen utama sistem informasi meliputi perangkat keras, perangkat lunak, basis data, jaringan, dan pengguna. Sistem ini memainkan peran penting dalam meningkatkan efisiensi operasional, mempercepat proses bisnis, dan

mengoptimalkan pengelolaan data. Dengan pengelolaan yang tepat, sistem informasi dapat menjadi aset strategis dalam lingkungan bisnis modern.

2.2.4 Rancang Bangun

Menurut Arifianto, dkk (2023), rancang bangun adalah kegiatan menerjemahkan hasil analisa kedalam bentuk paket perangkat lunak, kemudian menciptakan sistem tersebut atau memperbaiki sistem yang sudah ada. Dalam konteks sistem informasi, rancang bangun bertujuan untuk menciptakan sistem yang dapat meningkatkan efisiensi dan efektivitas dalam pengelolaan data serta mendukung pengambilan keputusan yang lebih akurat. Proses ini umumnya dimulai dengan analisis kebutuhan, yang mencakup pengumpulan data dan identifikasi permasalahan pada sistem yang telah ada. Selanjutnya, dilakukan perancangan yang meliputi desain sistem, model data, serta antarmuka pengguna, yang bertujuan untuk menghasilkan sistem yang sesuai dengan kebutuhan pengguna.

2.2.5 Flowchart

Menurut Kus Indrani Listyoningrum dkk., (2023), flowchart adalah suatu teknik untuk menggambarkan urutan logika dari suatu prosedur penyelesaian masalah. Flowchart sering digunakan dalam pemrograman, analisis sistem, serta dokumentasi prosedur operasional. Simbol-simbol standar digunakan dalam flowchart untuk merepresentasikan berbagai jenis operasi, seperti proses, keputusan, input/output, dan konektor. Dengan menggunakan flowchart, pengguna dapat memahami alur proses dengan lebih mudah, mengidentifikasi kesalahan, serta meningkatkan efisiensi dalam perancangan sistem.

Tabel 2.1 Simbol Flowchart

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Input/Output</i>	Sebagai media masukan dan keluaran dari data

Tabel 2.1 Lanjutan

NO	GAMBAR	NAMA	KETERANGAN
2		<i>Process</i>	Menggambarkan proses transformasi dari data masuk menjadi data keluar
3		<i>Predifined proses</i>	Menggambarkan proses yang masih berisi proses lain di dalamnya
4		<i>Preparation</i>	Sebagai pemberian nilai awal
5		<i>Start/End</i>	Sebagai awal dan akhir program
6		<i>Connector</i>	Sebagai penghubung satu halaman
7		<i>Decision</i>	Sebagai media untuk melakukan pemilihan
8		<i>Off-Page Connector</i>	Sebagai penghubung beda halaman

2.2.6 Sistem Informasi Inventory

Menurut (Irfan dkk., 2023) sistem informasi inventory merupakan sebuah hal keharusan yang harus dimiliki sebuah perusahaan ataupun usaha yang bergerak pada bidang penjualan dan pembelian agar dapat mengolah data transaksi yang berkaitan dengan persedian dapat dilakukan secara efektif dan efisien. Dengan sistem informasi inventory, proses pengelolaan stok menjadi lebih otomatis, mengurangi kesalahan pencatatan, meningkatkan akurasi data, serta mempercepat pengambilan keputusan terkait pengadaan barang. Fungsi utama sistem ini meliputi

pemantauan stok, pencatatan transaksi barang, analisis tren penggunaan, serta pengoptimalan proses pengadaan. Berdasarkan metode pengelolaannya, sistem inventory dapat bersifat manual, berbasis komputer, atau berbasis cloud. Komponen utama dalam sistem ini mencakup database untuk penyimpanan data, user interface sebagai antarmuka pengguna, laporan dan analisis untuk pemantauan stok, serta fitur keamanan guna melindungi akses dan data dari manipulasi. Dengan penerapan sistem informasi inventory, organisasi dapat mengelola stok barang secara lebih sistematis, akurat, dan efisien, sehingga membantu meningkatkan produktivitas serta mengurangi pemborosan sumber daya.

2.2.7 Website

Website adalah kumpulan halaman digital yang saling terhubung dan berfungsi sebagai antarmuka untuk mengakses informasi secara *online*. Menurut Mukhlis dkk. (2024), website dalam sistem informasi memiliki peran sentral sebagai media interaktif untuk berkomunikasi dengan pengguna. Dalam konteks sistem pendaftaran siswa, website digunakan untuk menampilkan formulir pendaftaran, memproses data, dan memberikan notifikasi secara *real-time*. Keunggulan website terletak pada aksesibilitasnya, yang memungkinkan pengguna untuk mengakses sistem dari mana saja dan kapan saja melalui perangkat yang terhubung ke internet. Selain itu, website dapat diintegrasikan dengan berbagai fitur keamanan untuk melindungi data pengguna dari akses yang tidak sah.

2.2.8 Unified Modelling Language

Unified Modeling Language (UML) adalah alat pemodelan yang digunakan untuk merancang dan mendokumentasikan sistem berbasis objek. Wahyuni dkk. (2024) menjelaskan bahwa UML membantu menggambarkan struktur dan interaksi sistem, seperti proses registrasi online dan pengelolaan data. Alat ini mencakup berbagai diagram, seperti *use case*, *class*, dan *activity* diagram, yang digunakan untuk mendeskripsikan kebutuhan sistem secara visual. Dengan menggunakan UML, pengembang dapat memastikan bahwa sistem yang dirancang memenuhi kebutuhan pengguna dan dapat diimplementasikan secara efisien. UML menjadi

standar dalam pengembangan perangkat lunak karena kemampuannya menjelaskan detail teknis dengan cara yang terstruktur.

Untuk memperjelas elemen-elemen yang membentuk use case diagram pada sistem yang dirancang, rincian simbolnya disajikan pada Tabel 2.2 berikut.

Tabel 2.2 Simbol Use Case Diagram

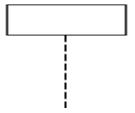
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Representasi entitas eksternal yang berinteraksi dengan sistem.
2		<i>Dependency</i>	Menunjukkan bahwa elemen bergantung pada elemen lainnya.
3		<i>Generalization</i>	Hubungan pewarisan antara elemen yang lebih spesifik dengan yang lebih umum.
4		<i>Include</i>	Use case menyertakan perilaku use case lain secara wajib.
5		<i>Extend</i>	Use case memperluas perilaku use case lain secara opsional
6		<i>Association</i>	Hubungan langsung antara aktor dan use case.
7		<i>System</i>	Menandai batas sistem yang mencakup semua use case.
8		<i>Use Case</i>	Representasi fungsi atau layanan yang disediakan oleh sistem.
9		<i>Collaboration</i>	Menunjukkan cara elemen bekerja bersama untuk mencapai tujuan tertentu.

Tabel 2.2 Lanjutan

NO	GAMBAR	NAMA	KETERANGAN
10		<i>Note</i>	Menyediakan penjelasan tambahan atau catatan untuk elemen diagram.

Setelah memahami simbol use case, langkah selanjutnya adalah mengenali simbol-simbol yang membangun class diagram, sebagaimana dirangkum pada **Tabel 2.3.**

Tabel 2.3 Simbol Class Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Merepresentasikan keberadaan sebuah objek atau aktor dalam waktu tertentu selama interaksi berlangsung.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Adapun simbol-simbol yang digunakan dalam activity diagram dipaparkan lebih lanjut pada **Tabel 2.4** untuk memberikan gambaran alur aktivitas secara detail.

Tabel 2.4 Simbol Activity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain

Tabel 2.4 Lanjutan

NO	GAMBAR	NAMA	KETERANGAN
2		Action	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		Initial Node	Bagaimana objek dibentuk atau diawali.
4		Actifity Final Node	Bagaimana objek dibentuk dan dihancurkan
5		Fork Node	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Untuk melengkapi pemahaman mengenai interaksi antar-objek dalam sistem, detail simbol yang digunakan pada sequence diagram dirangkum pada **Tabel 2.5** berikut.

Tabel 2.5 Simbol Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		Activity	Memperlihatkan bagaimana masing masing kelas antarmuka saling berinteraksi satu sama lain
2		Action	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		Initial Node	Bagaimana objek dibentuk atau diawali
4		Actifity Final Node	Bagaimana objek dibentuk dan dihancurkan
5		Decision Node	State dari sistem yang mengambil keputusan

2.2.9 Pengujian Blackbox

Pengujian Black Box merupakan metode pengujian perangkat lunak yang digunakan untuk mengevaluasi fungsi sistem berdasarkan spesifikasi yang telah ditentukan tanpa memperhatikan struktur internal atau kode sumber aplikasi.

Metode ini menilai kesesuaian antara data masukan yang diberikan pengguna dengan keluaran yang dihasilkan sistem, sehingga fokus utama pengujian berada pada perilaku fungsional sistem (Lubis, 2024).

Pendekatan Black Box dilakukan dengan menyusun skenario pengujian yang mencakup kondisi input valid maupun tidak valid. Tujuan dari pengujian ini adalah untuk memastikan sistem mampu memproses data dengan benar, menampilkan hasil yang sesuai, serta memberikan pesan kesalahan yang tepat ketika terjadi kesalahan input atau proses (Kartono dkk., 2024).

Pengujian Black Box banyak digunakan pada tahap pengujian sistem dan penerimaan pengguna (*user acceptance testing*) karena mencerminkan sudut pandang pengguna akhir. Metode ini efektif untuk mengidentifikasi kesalahan fungsi, antarmuka, dan validasi data, serta memastikan sistem telah memenuhi kebutuhan fungsional sebelum diimplementasikan secara operasional (Pratama dkk., 2023).

Dalam penelitian ini, pengujian Black Box diterapkan untuk menguji fitur-fitur utama sistem informasi yang dibangun, sehingga dapat memastikan bahwa setiap fungsi berjalan sesuai dengan kebutuhan pengguna dan spesifikasi sistem yang telah dirancang.

BAB III

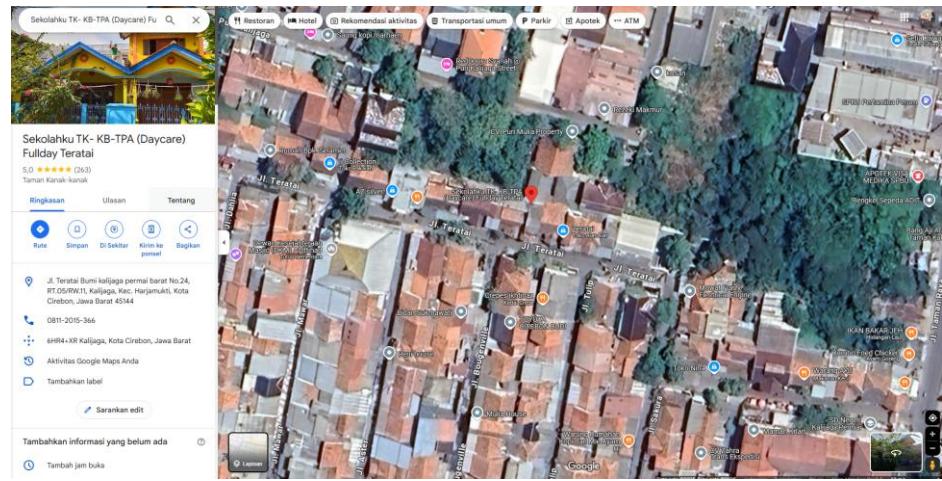
TINJAUAN UMUM INSTANSI

3.1 Gambaran Umum



Gambar 3.1 Bangunan TK Teratai

Gambar 3.1 menunjukkan Taman Kanak-Kanak Teratai Cirebon, sebuah lembaga pendidikan anak usia dini yang beralamat di Jl. Teratai No. 24, Kalijaga Permai, Kota Cirebon, Jawa Barat. Denah lokasi sekolah ini tersedia pada **Gambar 3.2**. Taman Kanak-Kanak (TK) Teratai merupakan sebuah lembaga pendidikan anak usia dini yang memiliki visi untuk membentuk generasi yang unggul dalam sikap, tindakan, kesehatan, kecerdasan, kreativitas, kemandirian, serta berakhhlak mulia. TK Teratai berkomitmen untuk menciptakan pembelajaran yang inovatif, kreatif, dan menyenangkan, serta membangun karakter anak dengan perilaku berbasis akhlak mulia.



Gambar 3. 2 Denah Lokasi

3.1.1 Visi & Misi

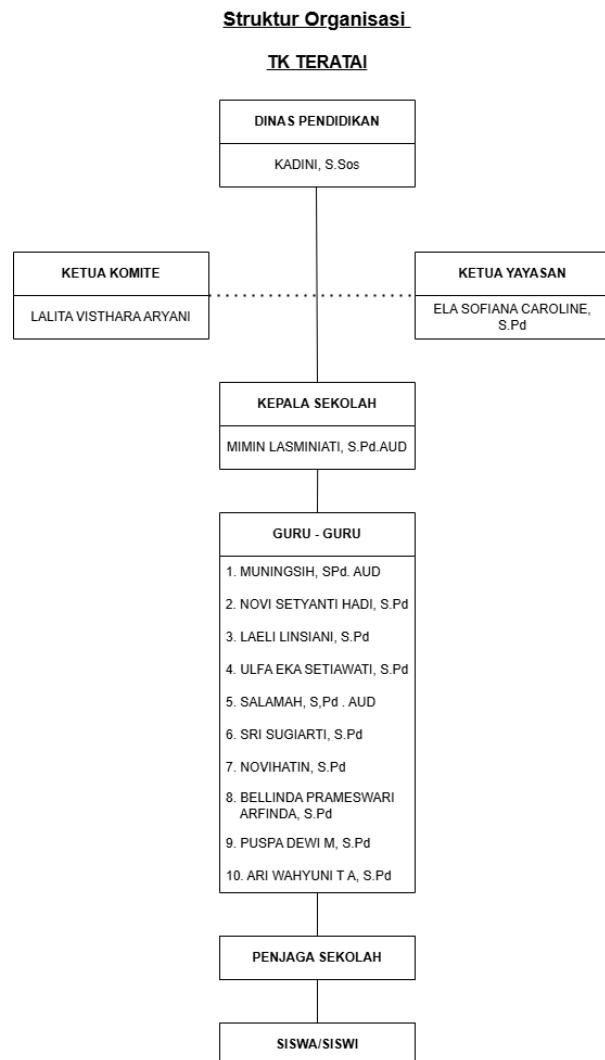
Visi

Membentuk generasi yang unggul dalam bertindak dan unggul dalam bersikap, sehat, cerdas, kreatif, mandiri, dan berakhhlakul karimah.

Misi

1. Menciptakan iklim pembelajaran yang kreatif, inovatif, dan menyenangkan.
2. Membangun pembiasaan berperilaku akhlak mulia dengan hidup sehat, mandiri, jujur, dan kompetitif.
3. Memfasilitasi sarana dan prasarana pembelajaran yang memadai.
4. Membangun kerja sama dengan wali murid, masyarakat sekitar, dan lingkungan dalam rangka pengelolaan PAUD yang berdaya saing.

3.2 Struktur Organisasi



Gambar 3.3 Bagan Struktur Organisasi

Berdasarkan **Gambar 3.3**, berikut adalah tugas masing-masing jabatan dalam struktur organisasi:

1.1. Dinas Pendidikan

Bertugas untuk mengawasi dan membina operasional lembaga pendidikan, termasuk menetapkan kebijakan, mengelola program

pendidikan, serta memastikan kualitas pendidikan sesuai standar nasional.

1.2. Ketua Yayasan

Memimpin yayasan yang menaungi sekolah dan bertanggung jawab atas pengelolaan keuangan, kebijakan strategis, serta pengembangan fasilitas dan kualitas pendidikan. Selain itu, yayasan juga bertindak sebagai penghubung antara sekolah dan pihak eksternal.

1.3. Ketua Komite

Berperan sebagai perwakilan orang tua siswa dan masyarakat dalam mendukung kebijakan sekolah, termasuk memberikan saran, membantu penggalangan dana, serta menjalin hubungan kerja sama yang baik antara pihak sekolah dan orang tua.

1.4. Kepala Sekolah

Mengelola seluruh kegiatan operasional sekolah, mulai dari perencanaan, pelaksanaan, hingga evaluasi program pembelajaran. Kepala sekolah juga bertanggung jawab atas pembinaan guru dan staf, pengelolaan administrasi, serta menjaga hubungan baik dengan yayasan, komite, dan dinas pendidikan.

1.5. Guru-Guru

Berperan memberikan pembelajaran kepada siswa sesuai kurikulum, memantau perkembangan siswa, memberikan evaluasi, serta berkomunikasi dengan orang tua terkait perkembangan anak.

1.6. Penjaga Sekolah

Berperan menjaga keamanan, kebersihan, dan ketertiban lingkungan sekolah, baik selama jam sekolah berlangsung maupun di luar jam operasional.

1.7. Siswa/Siswi

Para siswa/siswi belajar melalui bermain, berinteraksi, dan berlatih kemandirian dalam lingkungan yang menyenangkan dan mendidik. Mengikuti kegiatan pembelajaran seperti menyanyi, menggambar, bermain peran, dan berhitung dasar.

3.3 Produk / Jasa

Taman kanak kanak teratai menyediakan fasilitas yang bisa di gunakan untuk para siswa/siswi, maka dari itu menyediakan sebuah sistem informasi manajemen inventaris barang berbasis web yang dirancang secara khusus untuk memenuhi kebutuhan TK Teratai Cirebon dalam mengelola berbagai aset dan perlengkapan sekolah. Sistem ini bertujuan untuk mempermudah pencatatan, pemantauan, dan pelaporan inventaris barang seperti alat peraga, peralatan belajar, serta kebutuhan operasional lainnya secara digital. Dengan adanya sistem ini, proses administrasi yang sebelumnya dilakukan secara manual dapat dilakukan dengan lebih cepat, akurat, dan efisien.

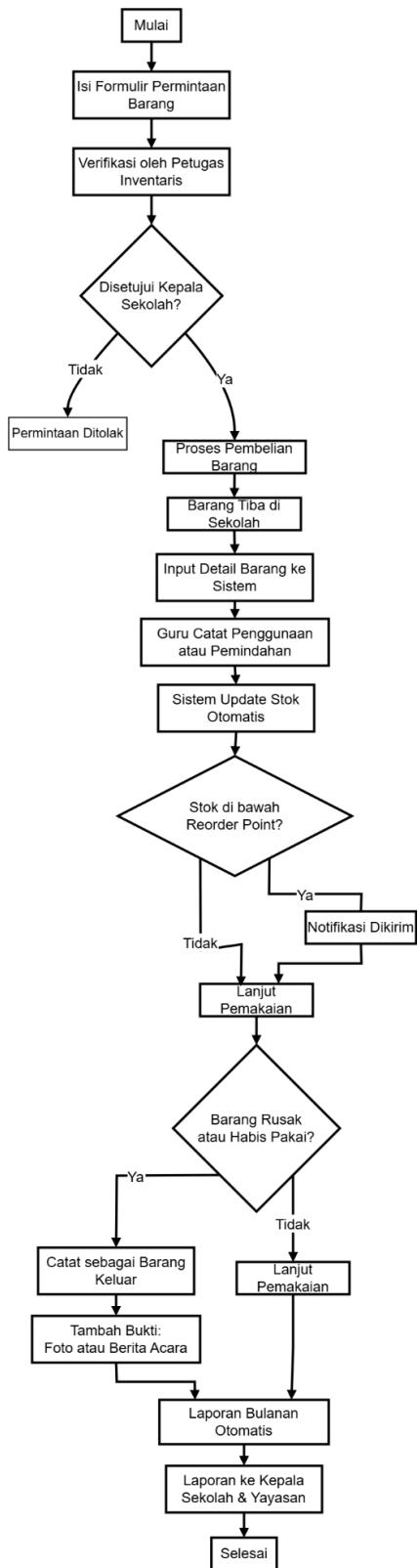
Sistem ini juga menyediakan fitur manajemen pengguna dengan hak akses berbeda berdasarkan peran seperti admin atau petugas inventaris, sehingga keamanan dan kontrol data tetap terjaga. Selain fitur pencatatan barang masuk dan keluar, sistem juga mampu menghasilkan laporan stok dan riwayat penggunaan barang yang dapat diakses kapan saja. Layanan dalam proyek ini mencakup pengembangan sistem, implementasi di lingkungan sekolah, pelatihan penggunaan sistem untuk staf terkait, serta dukungan teknis dan pemeliharaan sistem pasca implementasi agar sistem dapat terus berjalan optimal.

3.4 Proses Bisnis

Bagian ini menjelaskan rangkaian aktivitas utama yang berlangsung di TK Teratai Cirebon terkait pengelolaan inventaris barang. Proses bisnis dimulai dari perencanaan kebutuhan sarana prasarana, dilanjutkan pengadaan, pencatatan barang masuk, penggunaan harian di kelas, hingga penarikan atau penghapusan barang yang rusak. Setiap tahap melibatkan beberapa peran: kepala sekolah sebagai penanggung jawab kebijakan, petugas inventaris sebagai operator sistem, guru sebagai pemakai barang, serta yayasan sebagai pihak persetujuan anggaran. Dengan memetakan proses bisnis, diharapkan sistem informasi yang dibangun dapat mengotomatisasi alur kerja, meminimalkan pencatatan manual, dan menyediakan data stok secara *real-time*.

3.4.1 Alur Bisnis

Alur bisnis inventaris dimulai ketika guru atau staf mengajukan permintaan barang melalui formulir elektronik. Permintaan tersebut diverifikasi oleh petugas inventaris dan, jika disetujui kepala sekolah, diteruskan ke bagian pengadaan/yayasan untuk proses pembelian. Setelah barang tiba, petugas inventaris melakukan pengecekan fisik, mencatat detail barang (kode, kategori, jumlah, lokasi penyimpanan) ke dalam sistem, dan mencetak label inventaris. Barang kemudian didistribusikan ke kelas atau ruang penyimpanan sesuai kebutuhan. Selama masa penggunaan, guru mencatat pengeluaran barang via sistem setiap kali barang dipakai atau dipindahkan. Sistem otomatis memperbarui stok dan menampilkan peringatan reorder point bila jumlah minimum tercapai. Barang yang rusak atau habis pakai dicatat dalam modul barang keluar, disertai alasan dan bukti pendukung (foto atau berita acara). Laporan persediaan bulanan dihasilkan secara otomatis dan dikirim ke kepala sekolah serta yayasan untuk keperluan audit dan perencanaan anggaran selanjutnya. Adapun diagram flowchart alur bisnis sistem inventaris terdapat pada gambar 3.4 dibawah ini.

**Gambar 3.4** Flowchart Alur Bisnis

BAB IV

ANALISIS DAN PERANCANGAN

4.1 Analisis Sitem

Analisis sistem dilakukan untuk memahami alur pengelolaan inventaris barang di TK Teratai Cirebon dan mengidentifikasi area yang dapat ditingkatkan melalui pemanfaatan teknologi informasi. Beberapa tantangan yang ditemukan meliputi kesulitan dalam pelacakan stok barang, keterlambatan dalam penyusunan laporan, serta kebutuhan akan akses informasi yang lebih cepat dan akurat oleh berbagai pihak terkait.

Analisis dilakukan dengan menelaah alur kerja yang dilaporkan oleh petugas inventaris dan kepala sekolah, serta melalui studi dokumen dan referensi kebijakan internal. Hasil analisis menunjukkan adanya kebutuhan sistem yang mendukung pengelolaan data barang secara terpusat, pelaporan yang dapat diakses lintas peran pengguna, dan pencatatan transaksi yang terdokumentasi dengan baik.

4.2 Analisa Kebutuhan

Kebutuhan sistem yang berhasil dirangkum terdiri dari dua aspek, yaitu:

1. Admin Sistem

- a. Mengelola data pengguna (tambah, ubah, hapus akun).
- b. Melakukan reset password pengguna lain.
- c. Melihat dan memantau log aktivitas seluruh pengguna.
- d. Melakukan *backup* dan restore data sistem secara berkala.
- e. Mengakses seluruh fitur sistem untuk keperluan administrasi dan pemeliharaan.

2. Petugas Inventaris

- a. Mencatat barang masuk dan keluar melalui formulir sistem.

- b. Mengelola data barang (input data baru, edit, dan update status barang).
 - c. Melihat dan mencetak laporan inventaris bulanan.
 - d. Memantau stok barang secara *real-time*.
 - e. Melihat daftar barang rusak atau hilang.
 - f. Menyampaikan permintaan pengadaan barang kepada kepala sekolah.
3. Kepala Sekolah
- a. Melihat laporan inventaris secara berkala (bulanan, barang rusak, barang hilang).
 - b. Memantau kondisi stok barang untuk keperluan operasional sekolah.
 - c. Menyetujui atau menolak permintaan pengadaan barang dari petugas inventaris.
 - d. Melakukan monitoring terhadap pengelolaan inventaris tanpa mengubah data.

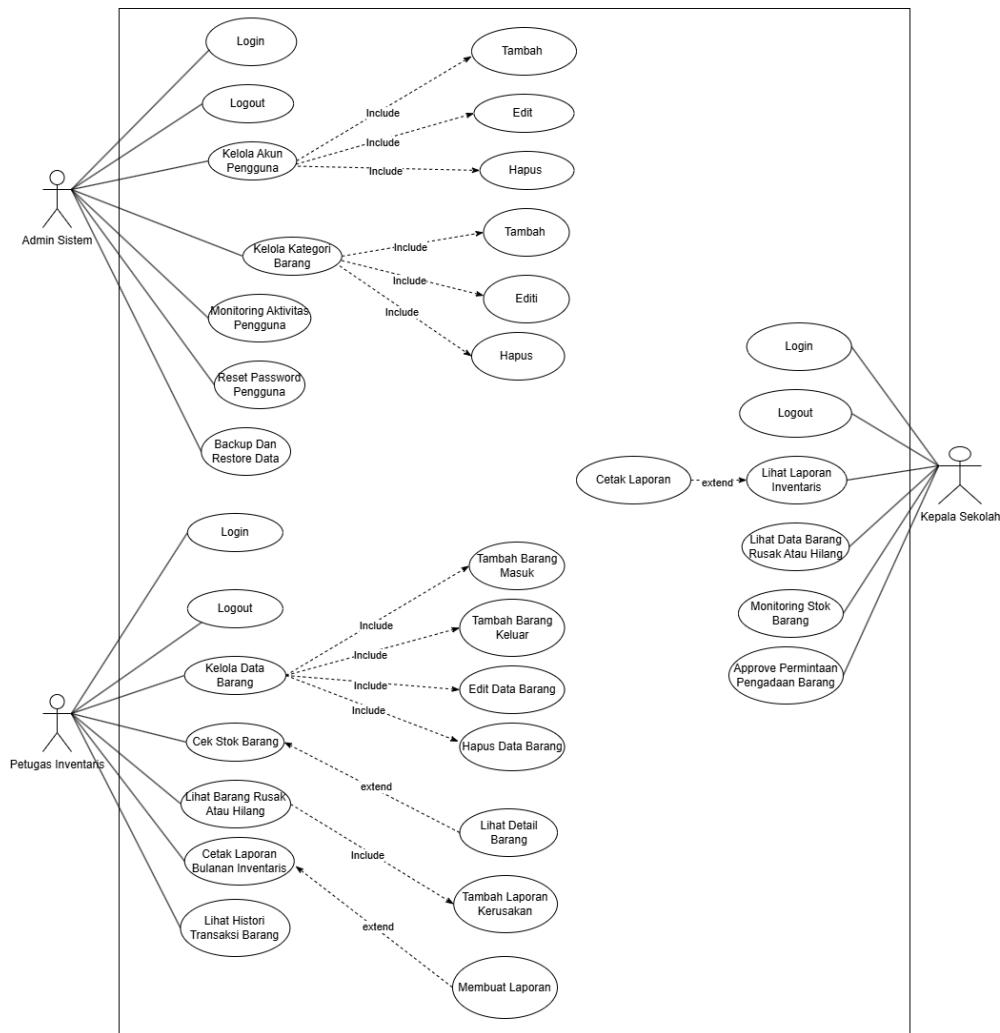
4.3 Perancangan Sistem

4.3.1 Perancangan Model Proses

Model proses menggambarkan bagaimana alur data mengalir dalam sistem. Perancangan sistem dilakukan sebagai dasar implementasi sistem informasi inventaris barang. Diagram UML digunakan untuk memastikan sistem yang dikembangkan sesuai dengan kebutuhan pengguna sebelum diimplementasikan secara penuh. Perancangan dilakukan dengan menggunakan diagram UML, seperti *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram*.

1. Diagram Use Case

Diagram Use Case ini menggambarkan interaksi antara aktor (pengguna) dan sistem informasi manajemen inventaris barang. Diagram ini menampilkan berbagai use case yang mencakup pengelolaan kategori barang, monitoring aktivitas pengguna, dan pembuatan laporan oleh admin, petugas inventaris, dan kepala sekolah.



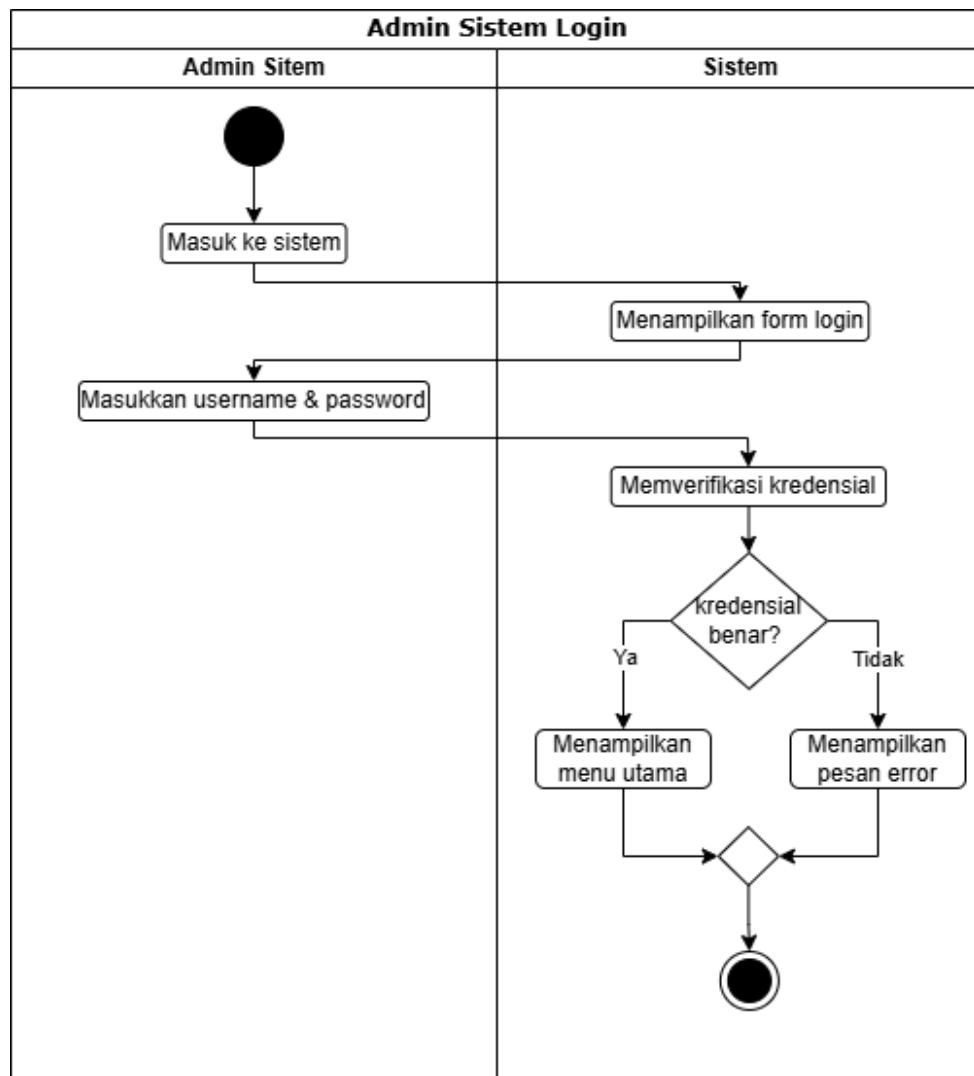
Gambar 4. 1 Use Case Diagram

2. Activity Diagram

Activity Diagram menggambarkan alur aktivitas yang dilakukan oleh berbagai peran dalam sistem, termasuk admin, petugas inventaris, dan kepala sekolah. Setiap diagram aktivitas memetakan langkah-langkah yang terlibat dalam proses seperti login sistem, pemilihan kategori barang, pemantauan aktivitas, serta reset password.

a. Activity Diagram Admin Sistem Login.

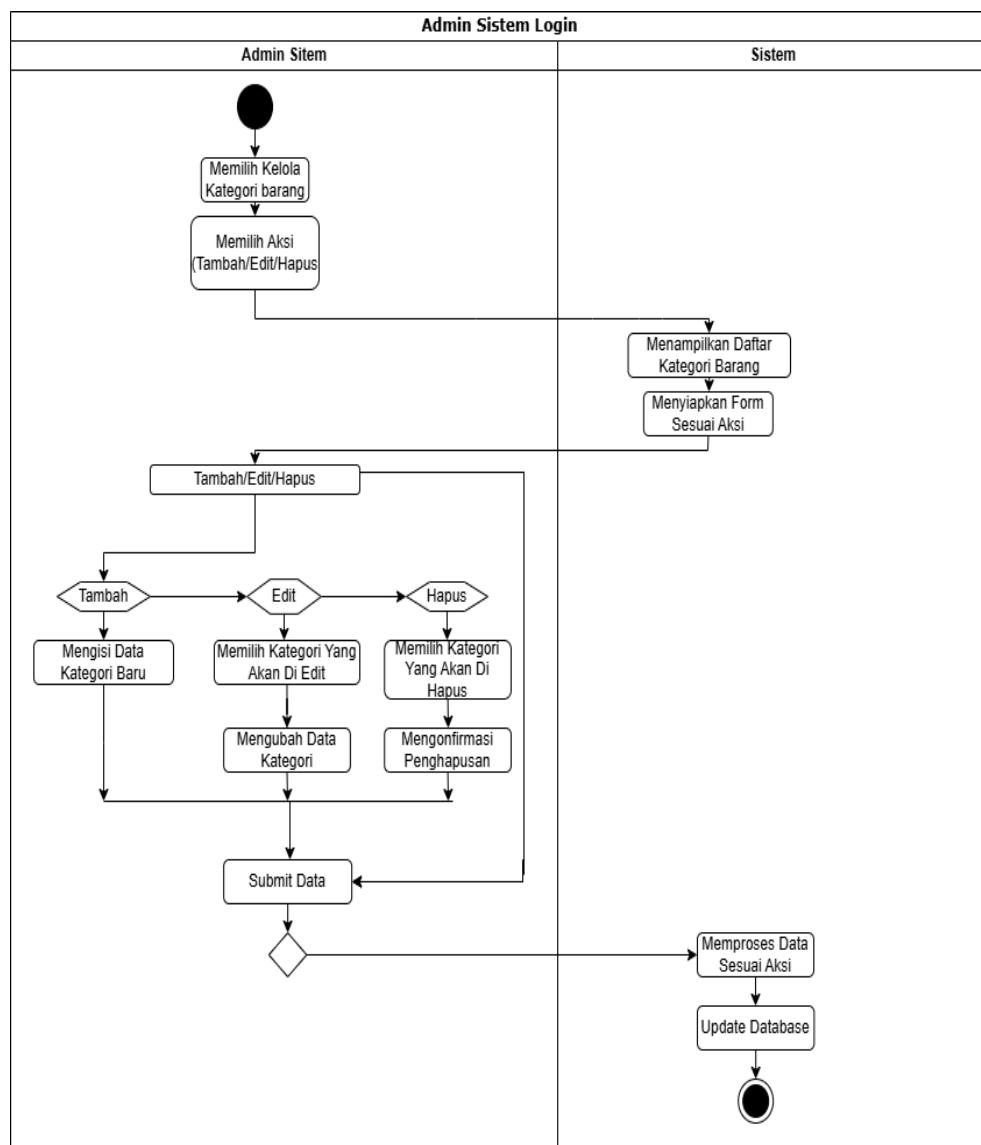
Diagram ini menggambarkan alur aktivitas yang terjadi saat admin melakukan login ke dalam sistem, mulai dari memasukkan username dan password hingga berhasil masuk ke dashboard sistem.



Gambar 4. 2 Activity Diagram Admin Sistem Login

b. Activity Diagram Admin Sistem Memilih Kategori Barang.

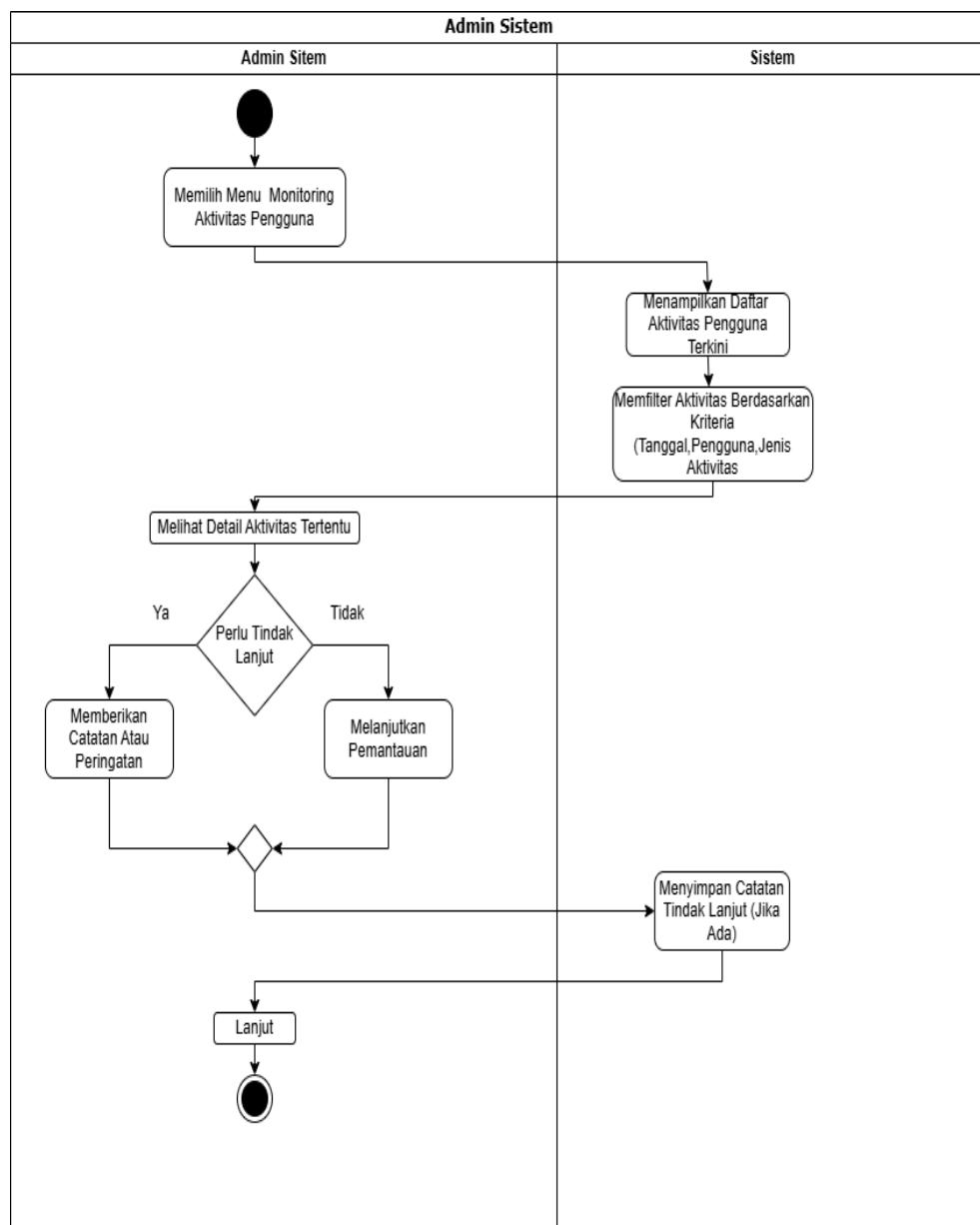
Diagram ini menggambarkan aktivitas yang dilakukan admin dalam memilih kategori barang untuk dikelola dalam sistem, termasuk proses penambahan atau pembaruan kategori barang.



Gambar 4. 3 Activity Diagram Admin Sistem Kelola Kategori Barang

c. Activity Diagram Admin Sistem Monitoring Aktivitas Pengguna

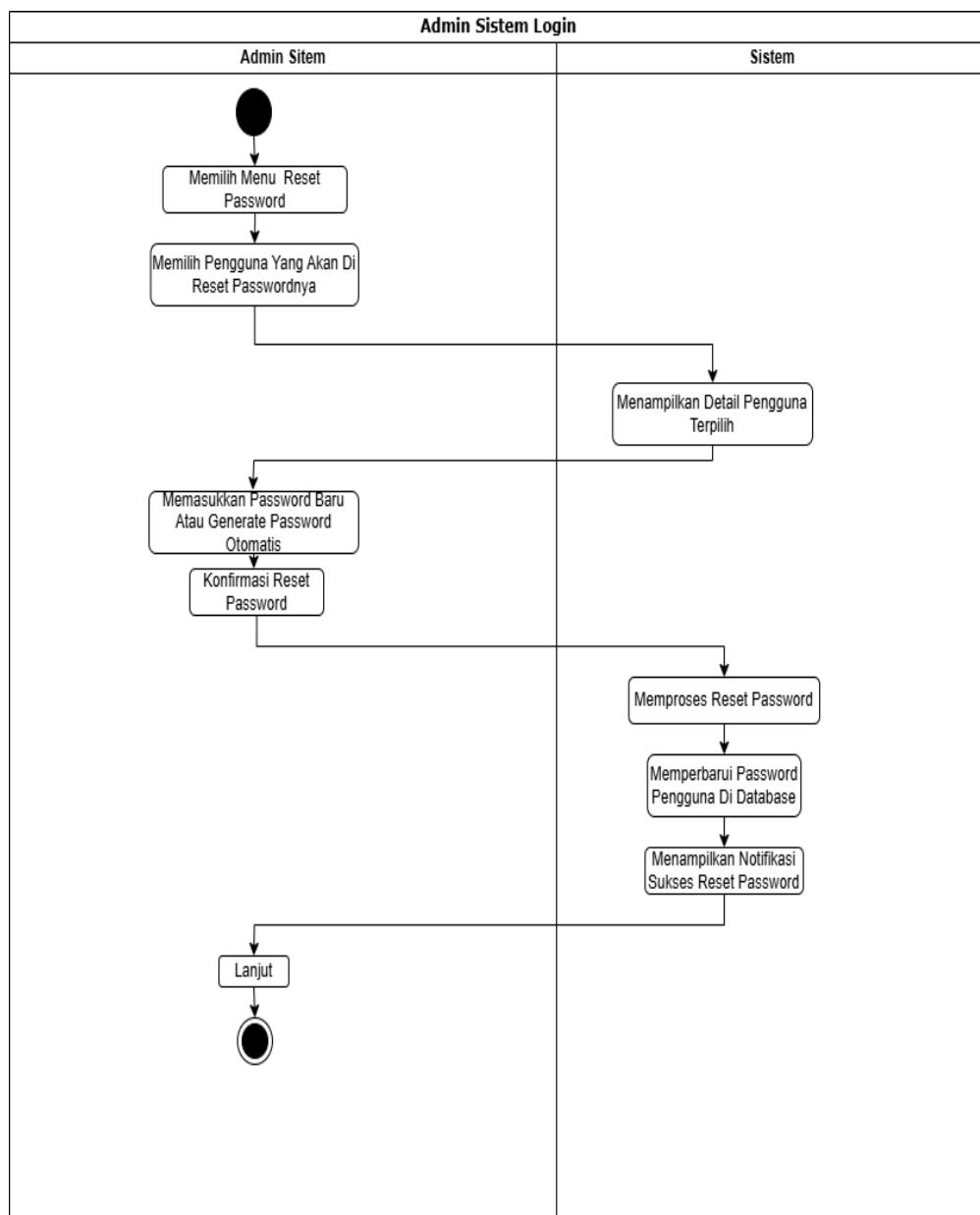
Diagram ini menunjukkan bagaimana admin memantau aktivitas pengguna dalam sistem untuk memastikan bahwa penggunaan sistem berjalan sesuai prosedur yang ditetapkan.



Gambar 4. 4 Activity Diagram Admin Sistem Monitoring Aktivitas Pengguna

d. Activity Diagram Admin Sistem Reset Password.

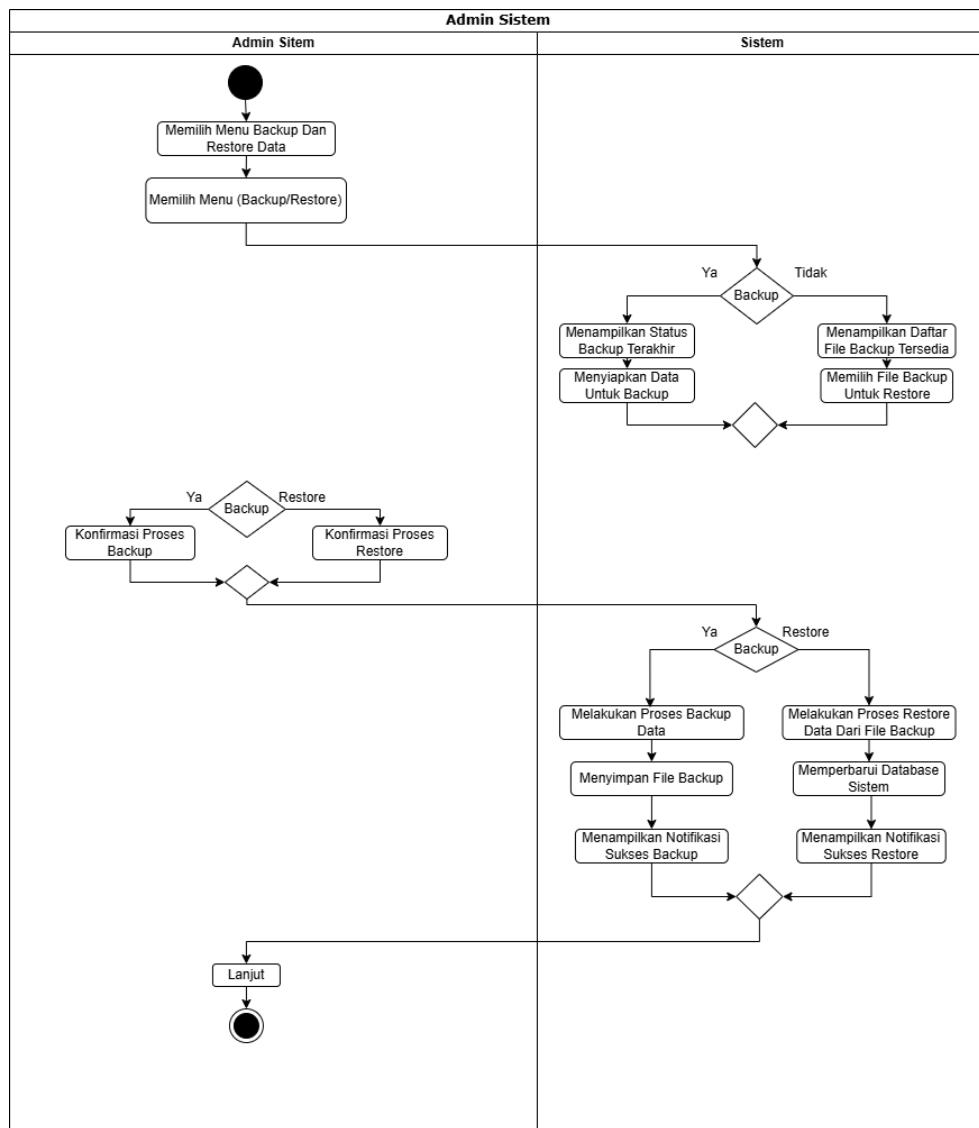
Diagram ini menggambarkan proses yang dilakukan admin untuk mereset password pengguna guna memastikan keamanan dan kontrol atas akun pengguna.



Gambar 4. 5 Activity Diagram Admin Ssitem Reset Password

e. Activity Diagram Admin Sistem *Backup* Dan *Restore* Data.

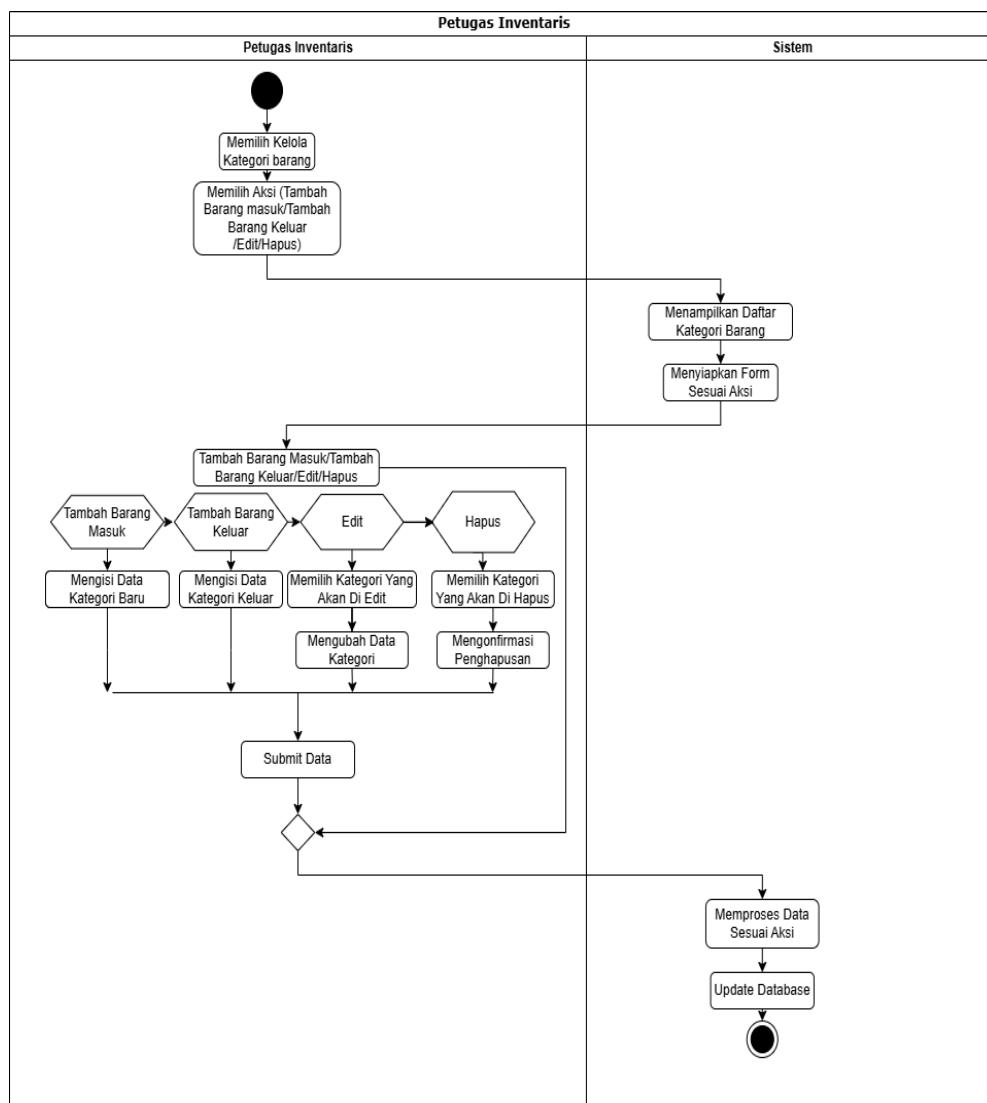
Diagram ini menggambarkan langkah-langkah yang dilakukan admin dalam melakukan *backup* dan *restore* data untuk menjaga integritas dan ketersediaan informasi sistem.



Gambar 4. 6 Activity Diagram Admin Sistem *Backup* Dan *Restore* Data

f. Activity Diagram Petugas Inventaris Kelola Kategori Barang.

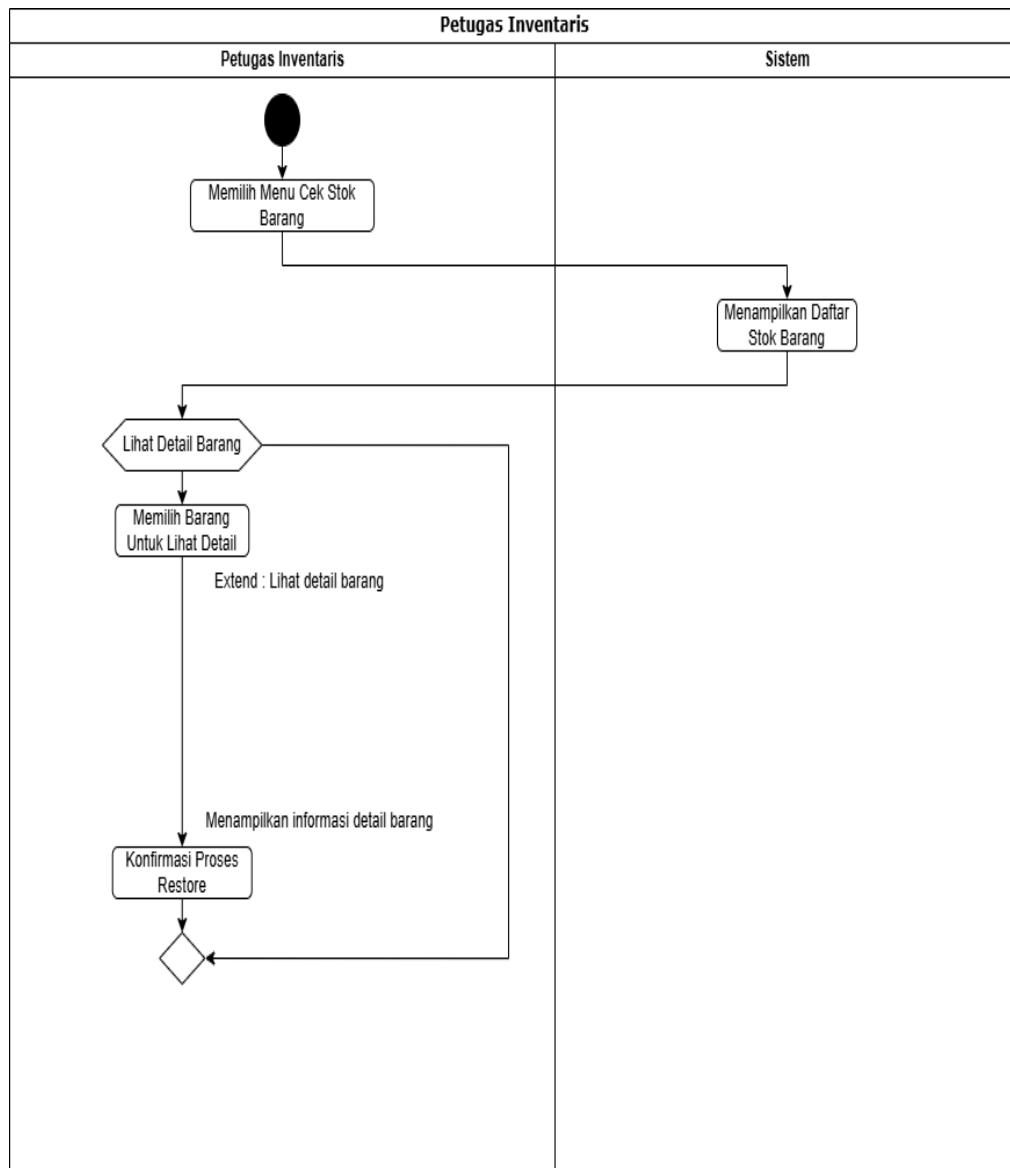
Diagram ini menggambarkan aktivitas petugas inventaris dalam mengelola kategori barang, termasuk pencatatan kategori baru atau pengeditan kategori barang yang sudah ada.



Gambar 4. 7 Activity Diagram Petugas Inventaris Edit Data Barang

g. Activity Diagram Petugas Inventaris Memilih Menu Cek Stok Barang.

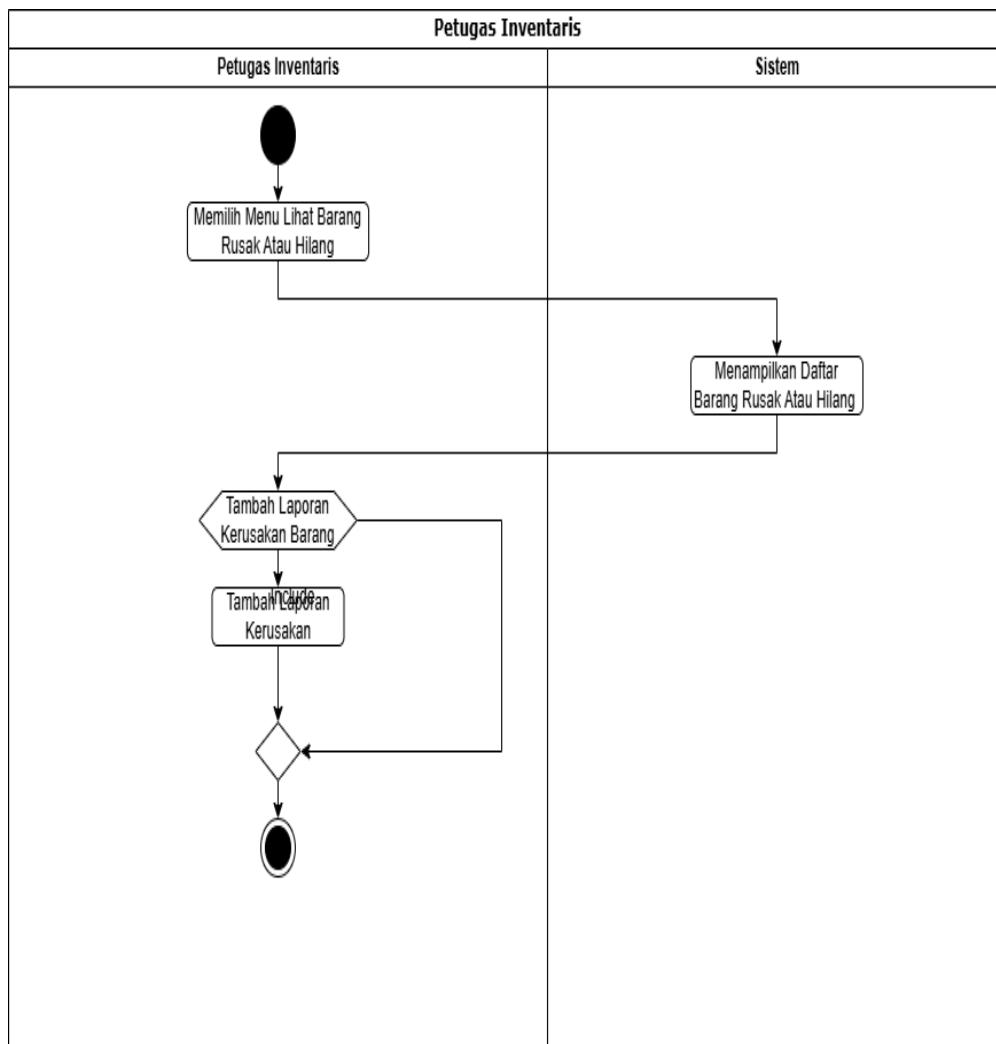
Diagram ini menunjukkan bagaimana petugas inventaris memilih menu untuk memeriksa stok barang yang tersedia di dalam sistem.



Gambar 4. 8 Activity Diagram Petugas Inventaris Cek Stok Barang

h. Activity Diagram Petugas Inventaris Lihat Barang Rusak Atau Hilang.

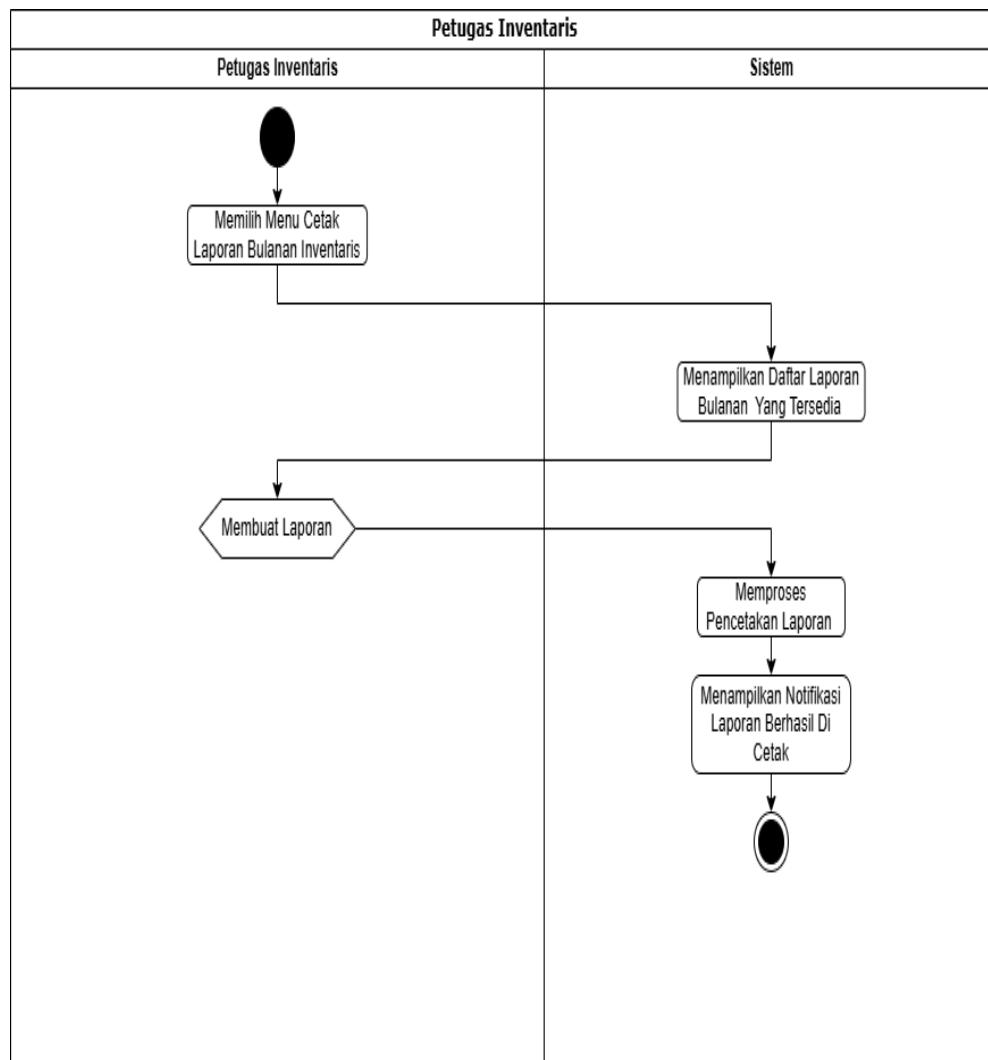
Diagram ini menggambarkan alur yang diikuti petugas inventaris untuk melihat data barang yang rusak atau hilang, serta memperbarui status barang tersebut di sistem.



Gambar 4. 9 Activity Diagram Petugas Inventaris Lihat Barang Rusak Atau Hilang

i. Activity Diagram Petugas Inventaris Cetak Laporan Bulanan.

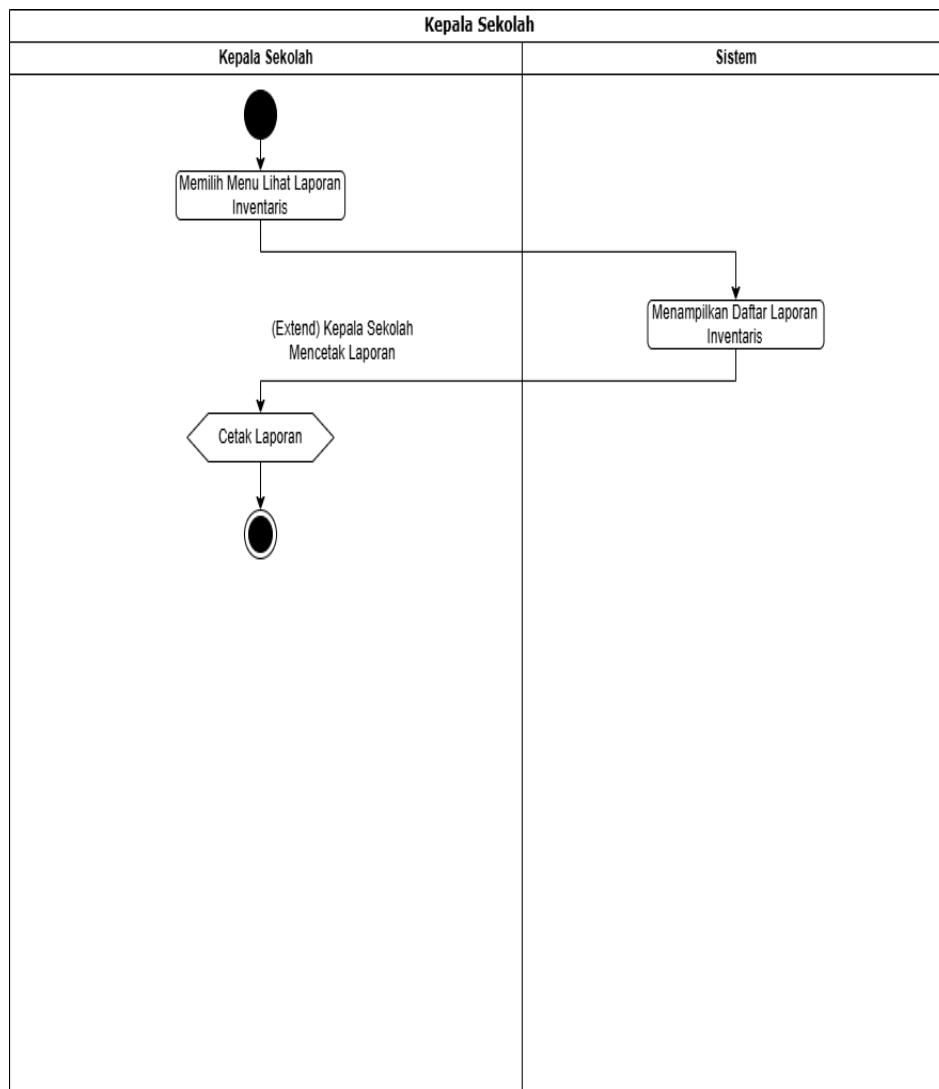
Diagram ini menunjukkan bagaimana petugas inventaris mencetak laporan bulanan mengenai status stok barang, barang yang digunakan, serta barang yang rusak atau hilang.



Gambar 4. 10 Petugas Inventaris Cetak Laporan Bulanan Inventaris

j. Activity Diagram Kepala Sekolah Melihat Laporan Inventaris.

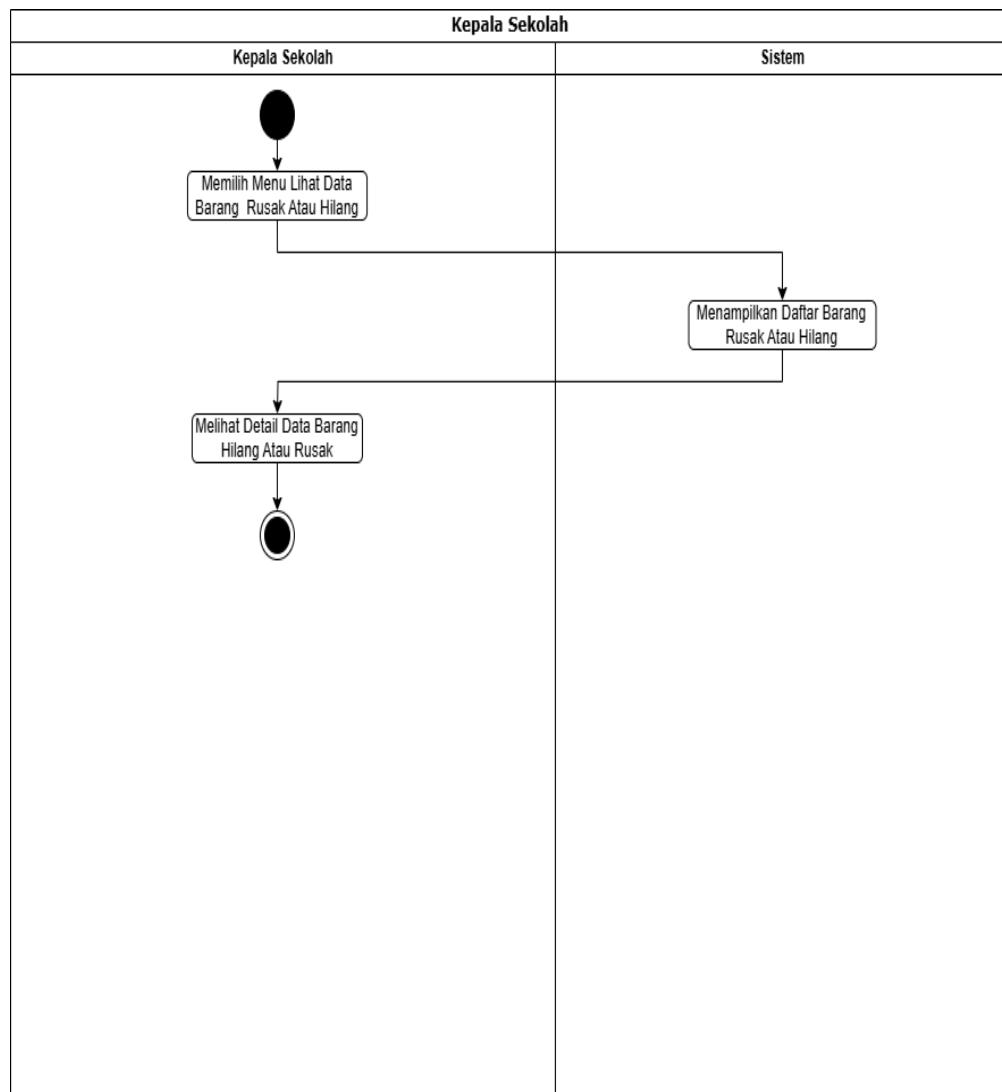
Diagram ini menggambarkan bagaimana kepala sekolah melihat laporan inventaris untuk memantau keadaan stok dan pengelolaan barang yang ada.



Gambar 4. 11 Activity Diagram Kepala Sekolah Lihat Laporan

k. Activity Diagram Kepala Sekolah Melihat Data Barang Rusak Atau Hilang.

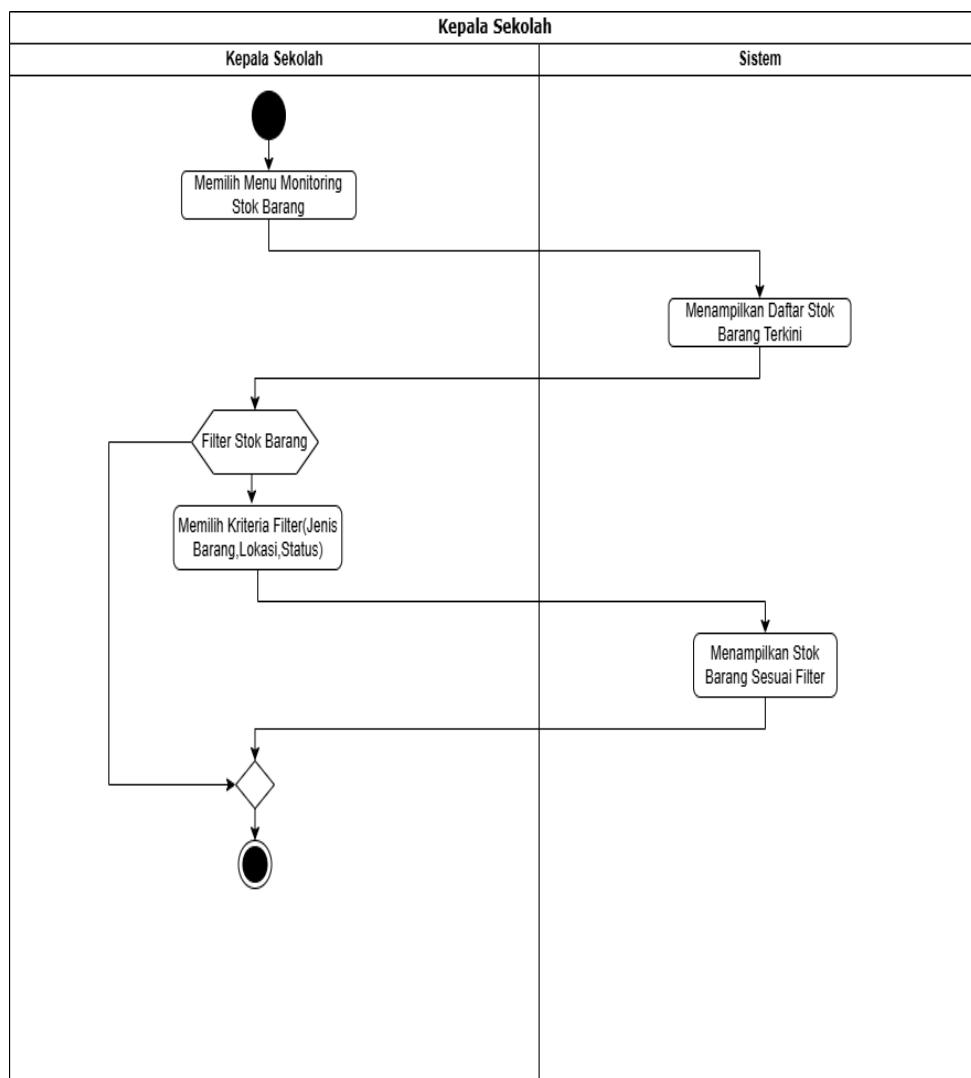
Diagram ini menunjukkan bagaimana kepala sekolah memeriksa data barang yang rusak atau hilang untuk memastikan pengelolaan yang tepat.



Gambar 4. 12 Activity Diagram Kepala Sekolah Lihat Data Barang Rusak Atau Hilang

1. Activity Diagram Kepala Sekolah Monitoring Stok Barang.

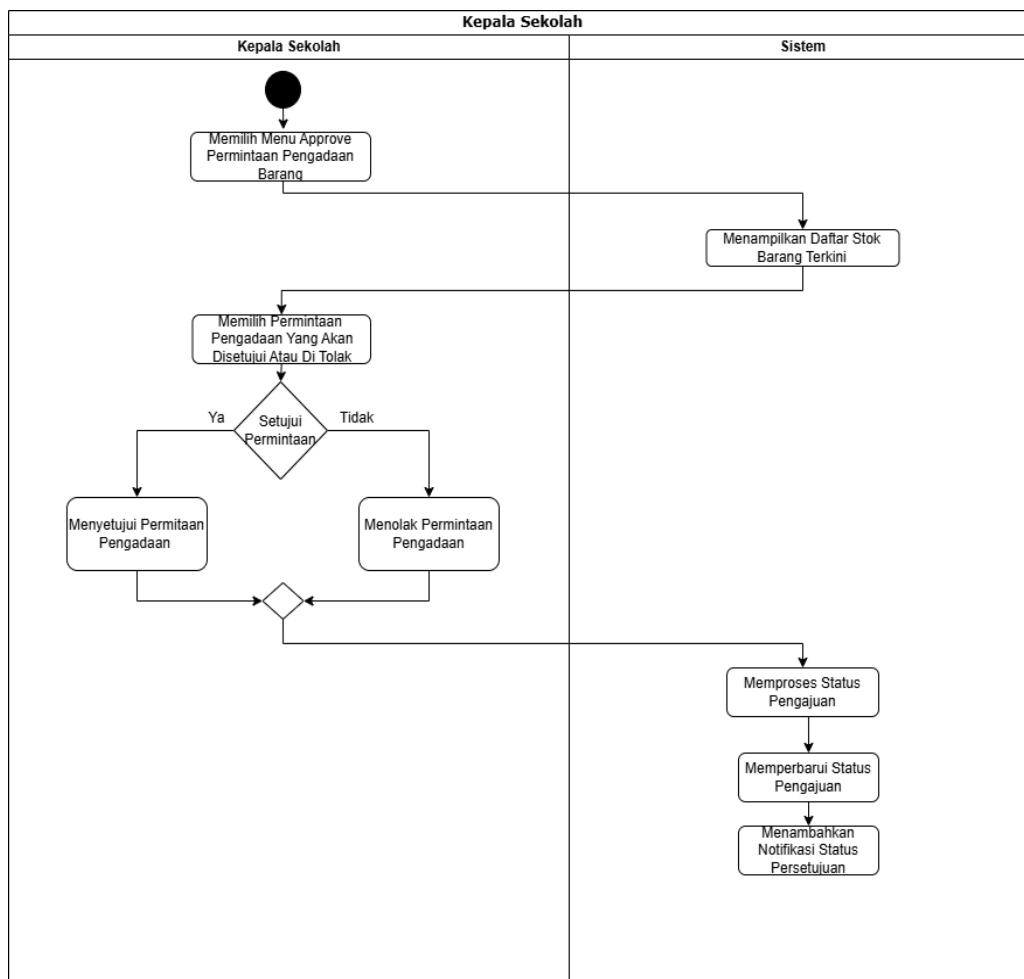
Diagram ini menggambarkan aktivitas kepala sekolah dalam memantau stok barang yang tersedia, termasuk mengidentifikasi kebutuhan pengadaan barang baru atau penggantian barang yang rusak.



Gambar 4. 13 Activity Diagram Kepala Sekolah Monitoring Stok Barang

m. Activity Diagram Kepala Sekolah Approve Permintaan Pengadaan Barang.

Diagram ini menunjukkan bagaimana kepala sekolah menyetujui permintaan pengadaan barang yang diajukan oleh staf atau guru, setelah memverifikasi ketersediaan anggaran.



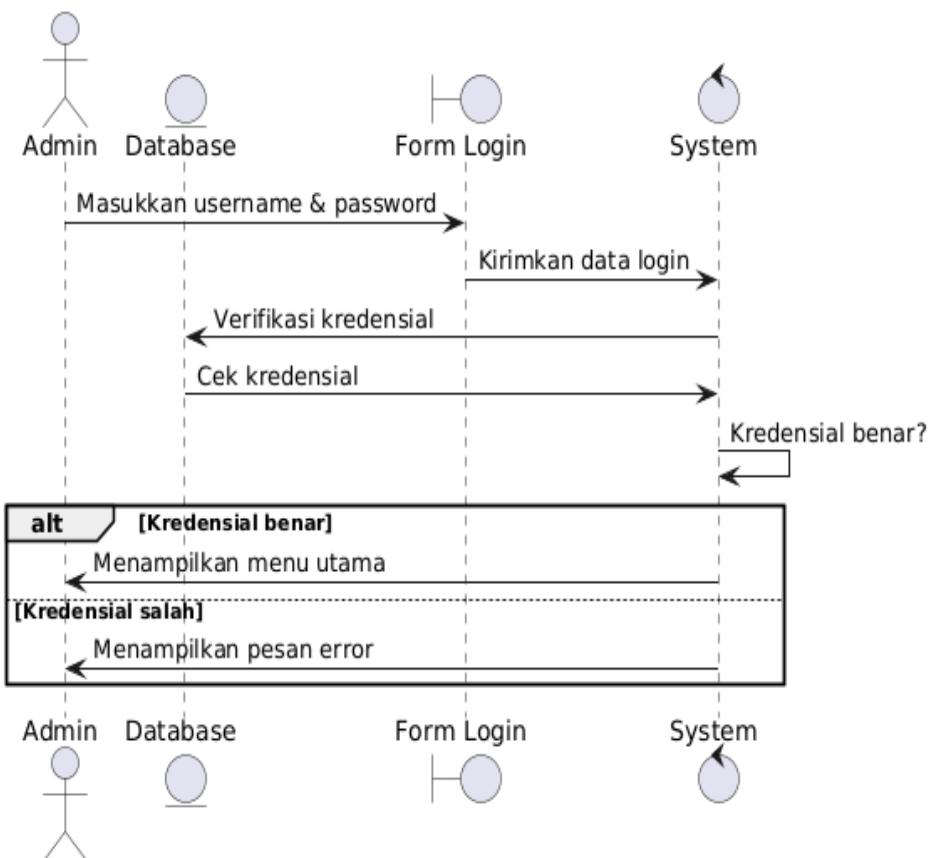
Gambar 4. 14 Activity Diagram Kepala Sekolah Approve Permintaan Pengadaan Barang

3. Sequence Diagram

Sequence Diagram ini menggambarkan urutan interaksi antar objek dalam sistem, memperlihatkan alur komunikasi antara admin, petugas inventaris, dan kepala sekolah dalam proses login, pengelolaan kategori barang, monitoring aktivitas pengguna, serta *backup* data.

a. Sequence Diagram Admin Sistem Login

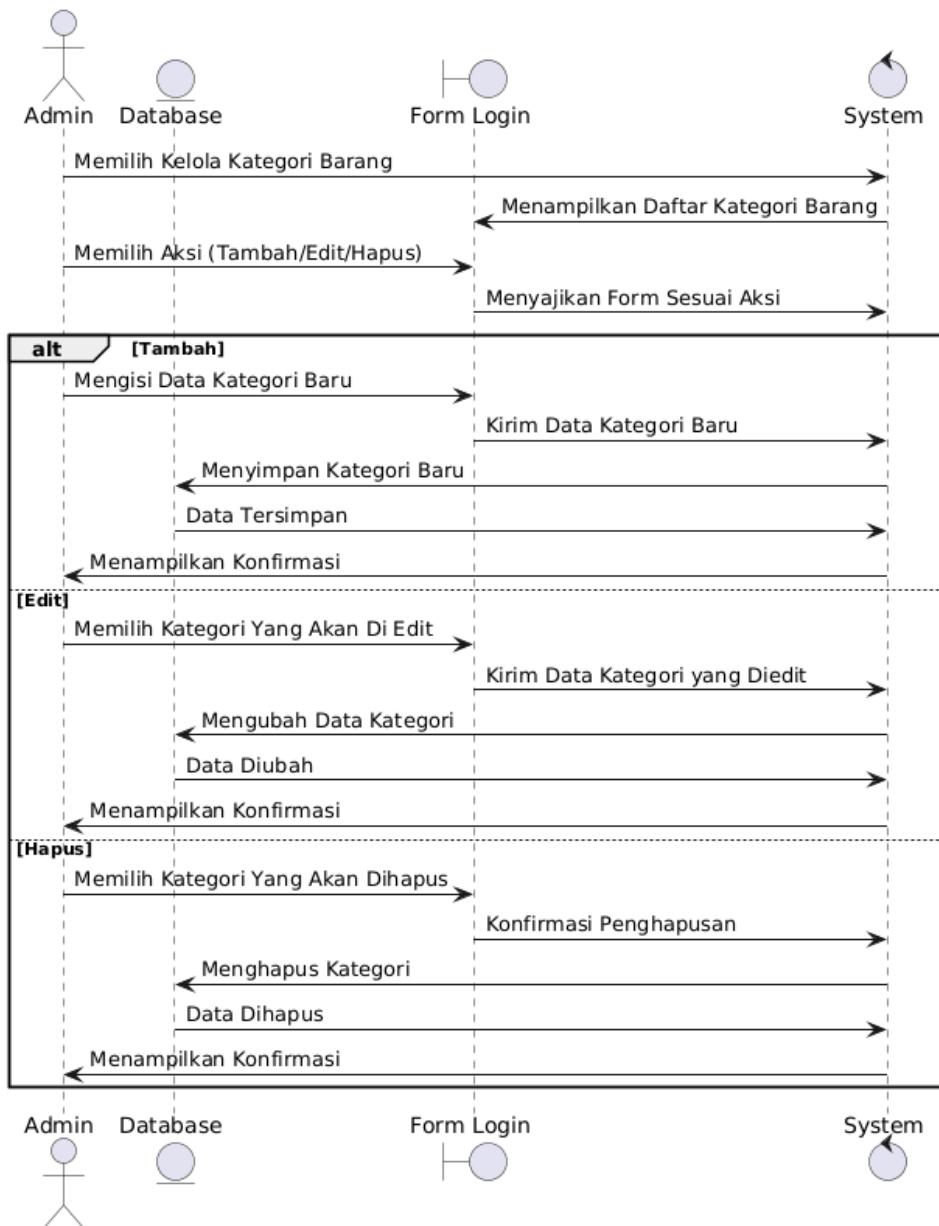
Diagram ini menggambarkan urutan interaksi antara admin dan sistem saat melakukan proses login untuk mengakses sistem manajemen inventaris barang.



Gambar 4. 15 Squence Diagram Admin Sistem Login

b. Sequence Diagram Admin Sistem Kelola Kategori Barang

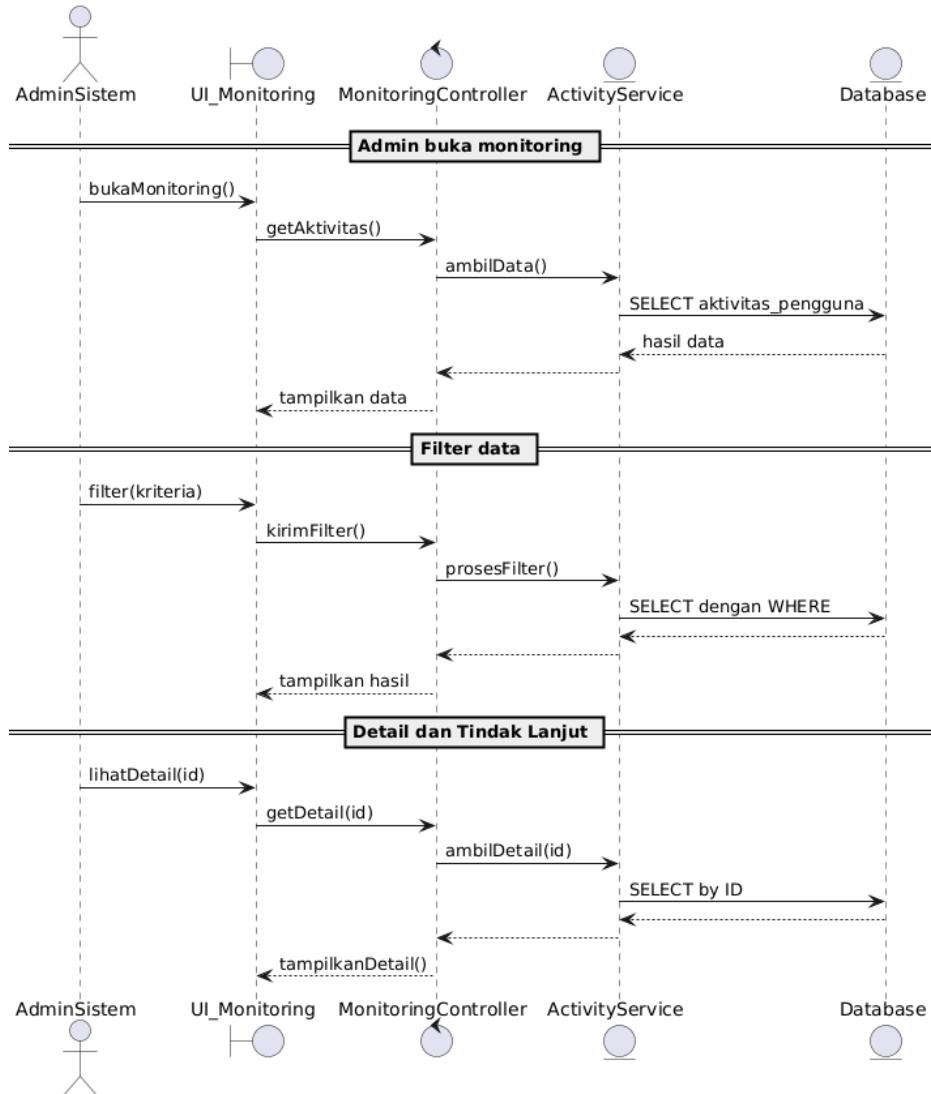
Diagram ini memperlihatkan bagaimana admin mengelola kategori barang, baik untuk menambah, mengubah, atau menghapus kategori barang yang ada.



Gambar 4. 16 Sequence Diagram Admin Ssitem Memilih Kategori Barang

c. Sequence Diagram Admin Sistem Monitroing Aktivitas Pengguna

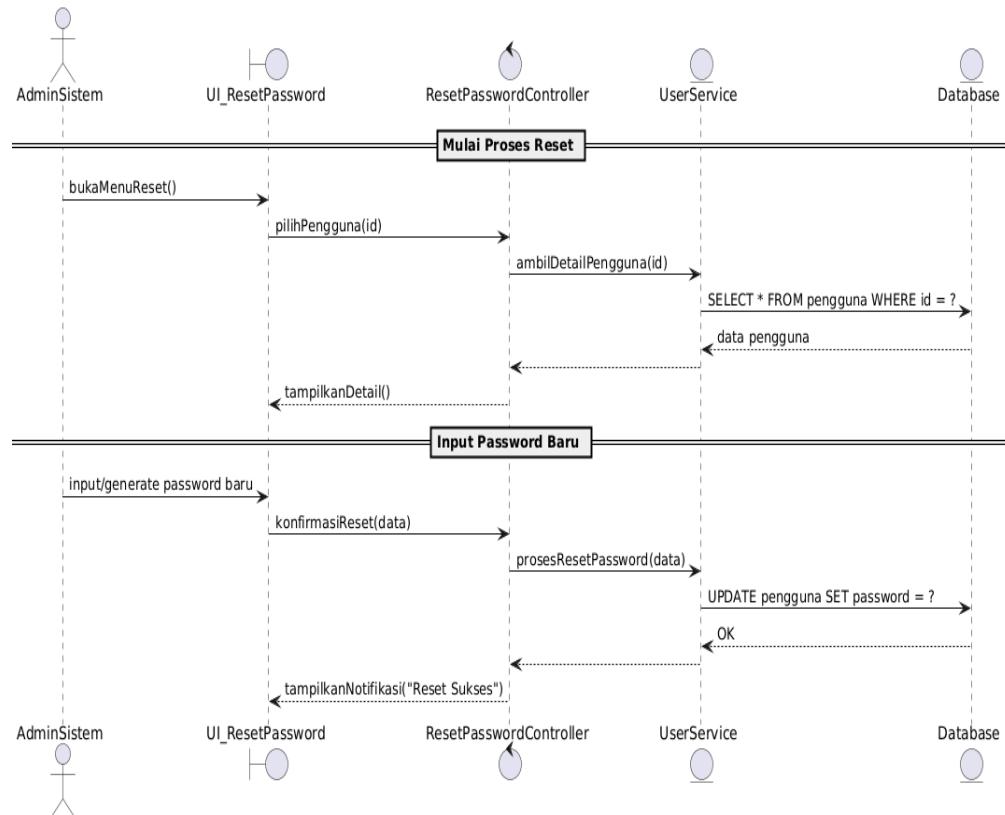
Diagram ini menggambarkan urutan interaksi antara admin dan sistem untuk memantau aktivitas pengguna dalam sistem.



Gambar 4. 17 Squence Diagram Admin Sistem Monitoring Barang

d. Squence Diagram Admin Sistem Reset Password

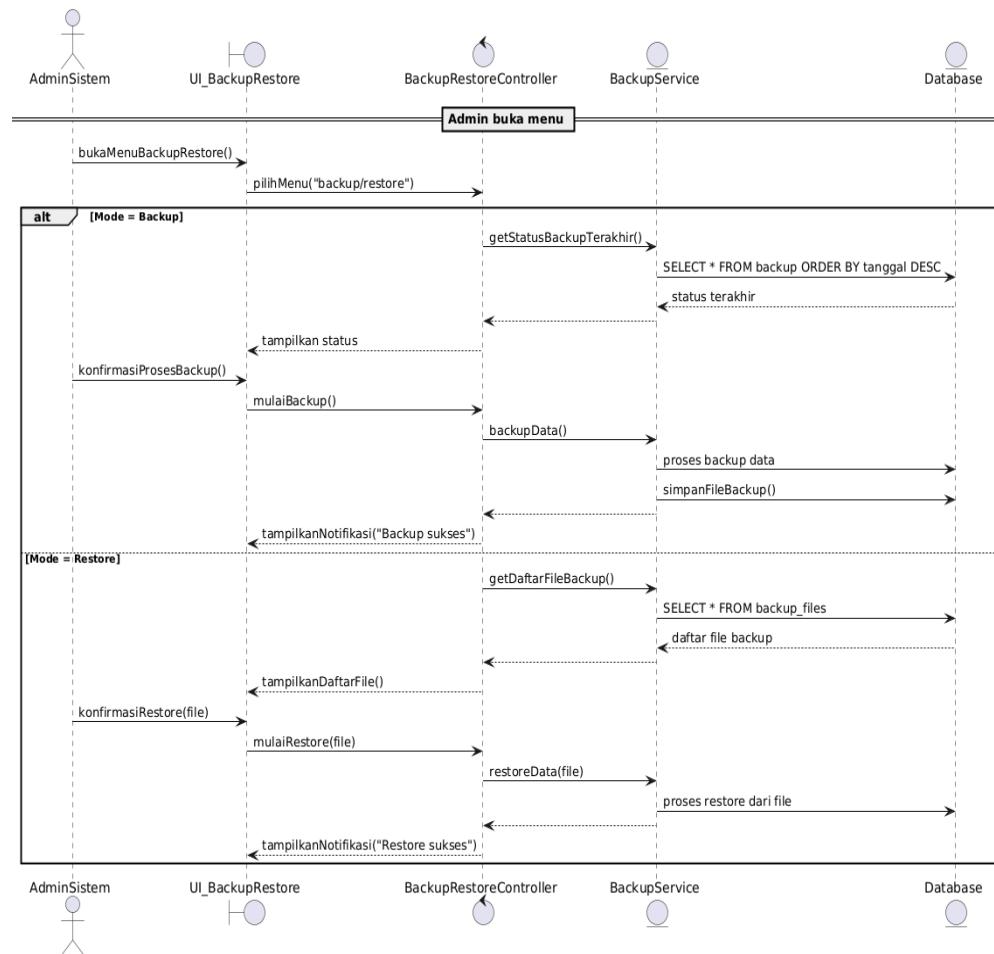
Diagram ini menunjukkan urutan proses reset password oleh admin untuk memastikan keamanan akses pengguna dalam sistem.



Gambar 4. 18 Squence Diagram Admin Sistem Reset Password

e. Sequence Diagram Petugas Inventaris *Backup* Dan Restore Data

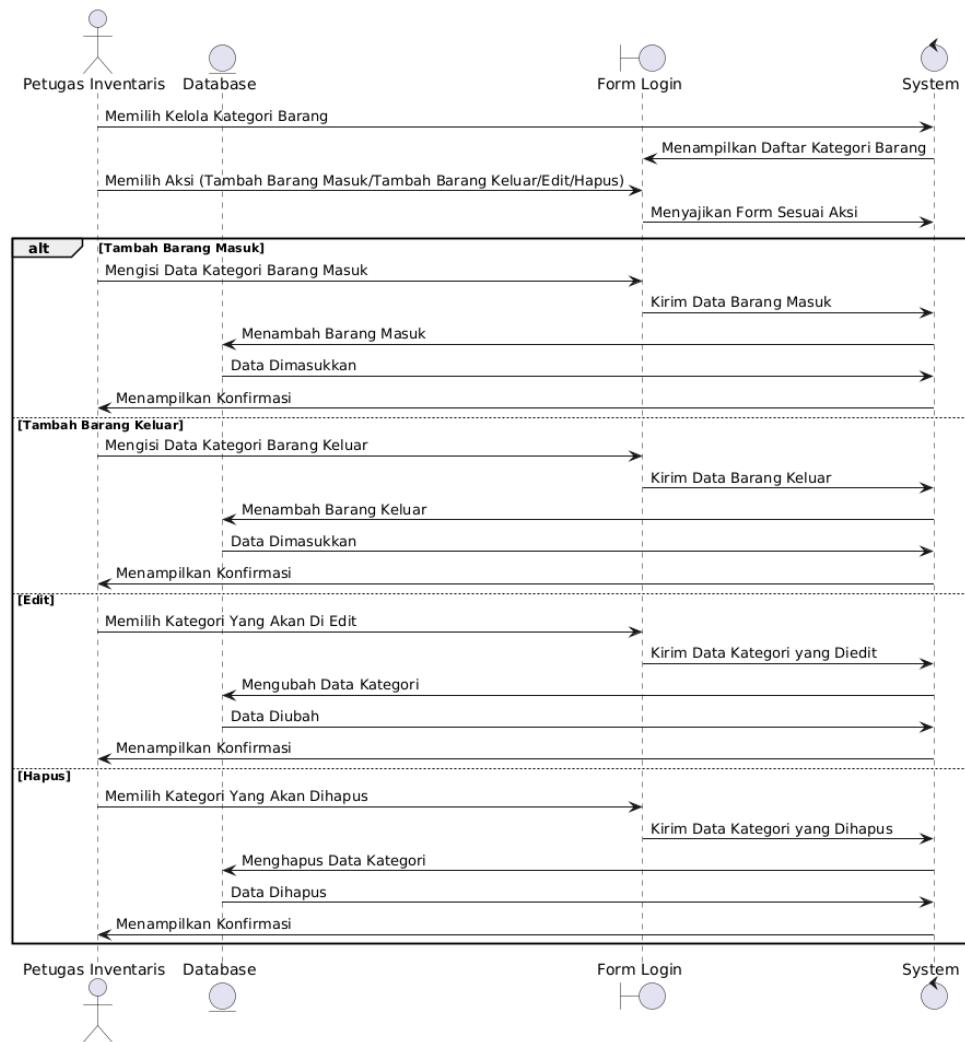
Diagram ini menggambarkan interaksi antara petugas inventaris dan sistem dalam melakukan *backup* dan restore data untuk memastikan data inventaris tersimpan dengan aman.



Gambar 4. 19 Squence Diagram Admin Buka Menu

f. Squence Diagram Petugas Inventaris Kelola Kategori Barang

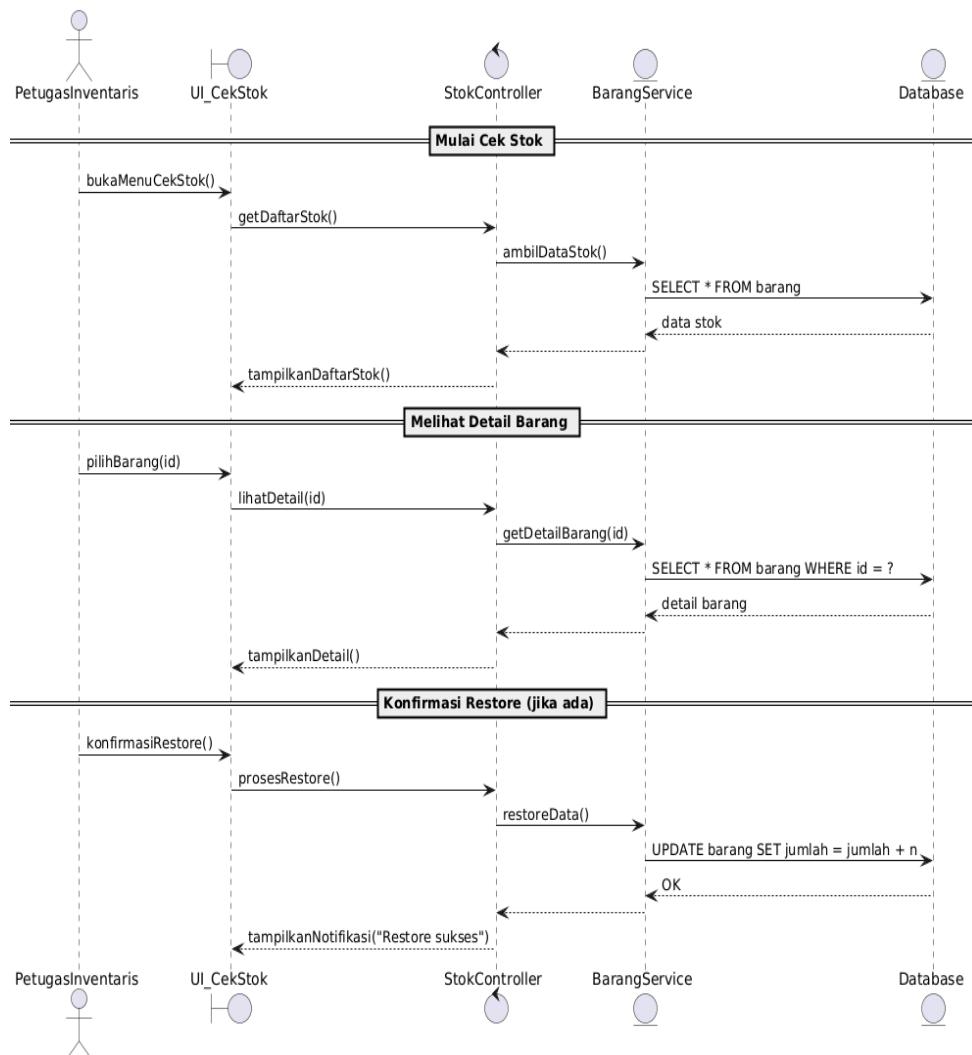
Diagram ini memperlihatkan interaksi antara petugas inventaris dan sistem dalam mengelola kategori barang, termasuk pencatatan barang baru atau perubahan status barang.



Gambar 4. 20 Squence Diagram Petugas Inventaris Edit Barang

g. Squence Diagram Petugas Inventaris Cek Stok Barang

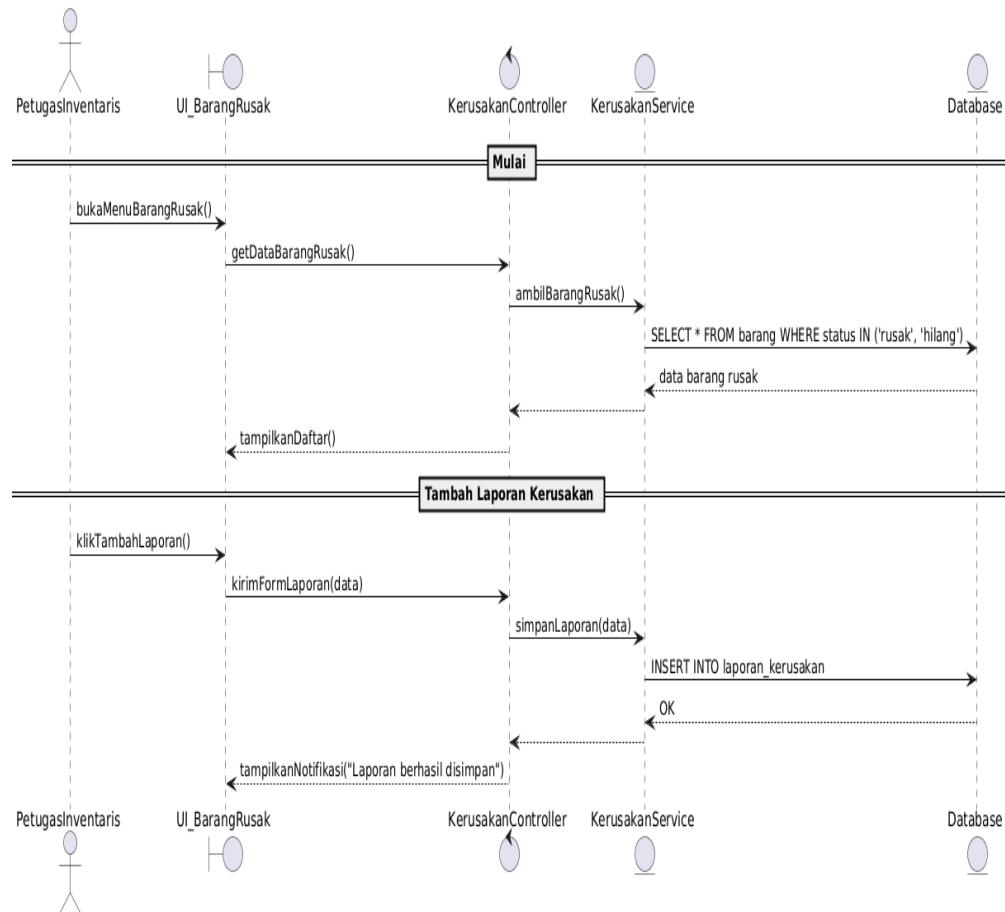
Diagram ini menggambarkan urutan interaksi petugas inventaris dalam memeriksa stok barang yang ada, memastikan data yang ditampilkan akurat dan terkini



Gambar 4. 21 Squence Diagram Petugas Inventaris Cek Stok Barang

h. Squence Diagram Petugas Inventaris Melihat Barang Rusak Atau Hilang

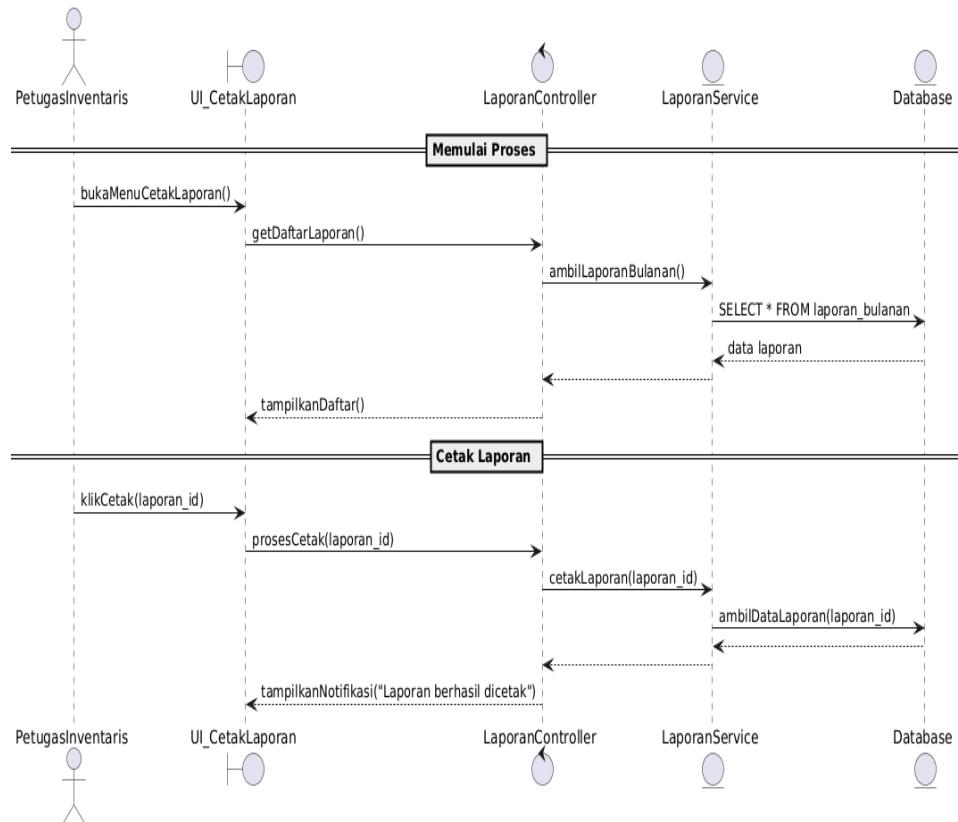
Diagram ini menunjukkan bagaimana petugas inventaris melihat daftar barang rusak atau hilang dan memperbarui statusnya di sistem.



Gambar 4. 22 Squence Diagram Petugas Inventaaris Melihat Barang Rusak Atau Hilang

i. Squence Diagram Petugas Inventaris Cetak Laporan Bulanan

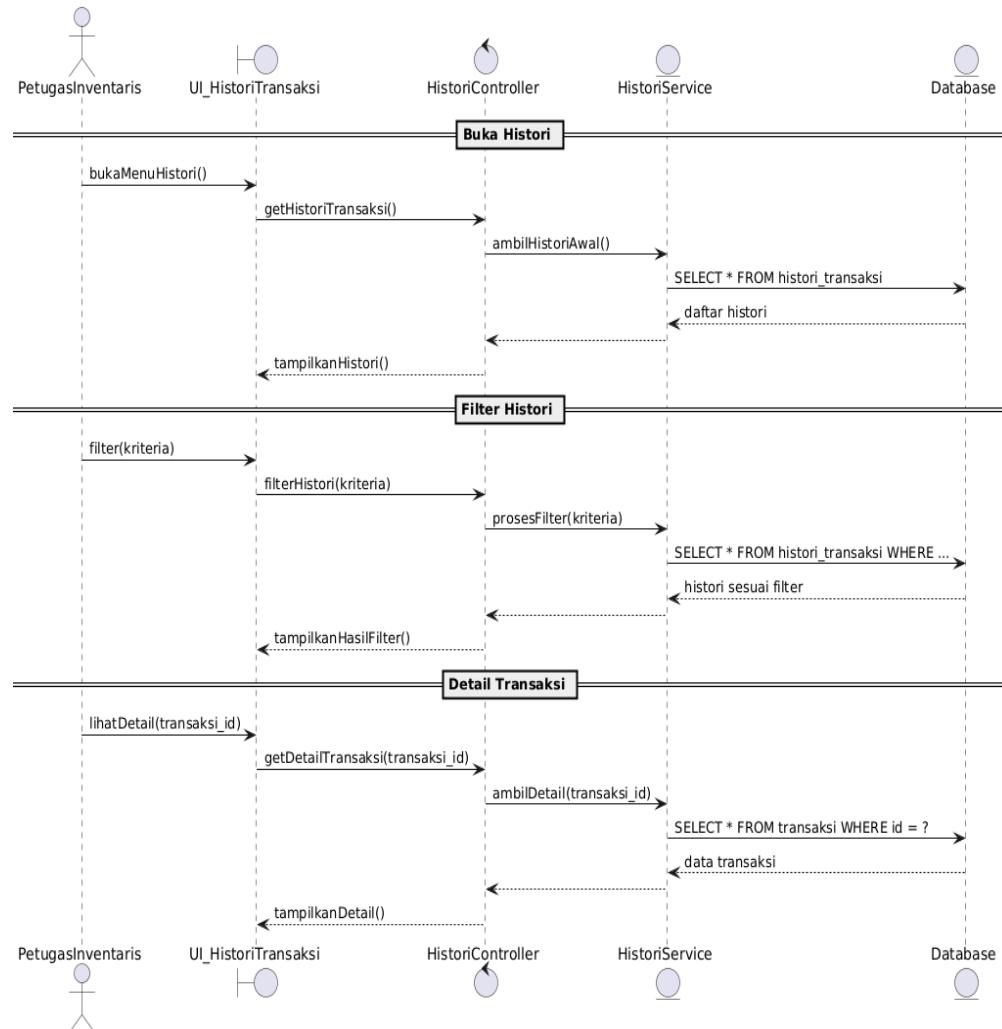
Diagram ini menggambarkan proses yang dilakukan petugas inventaris untuk mencetak laporan bulanan terkait status stok barang.



Gambar 4. 23 Squence Diagram Petugas Inventaris Cetak Laporan

j. Squence Diagram Kepala Sekolah Melihat Laporan Inventaris

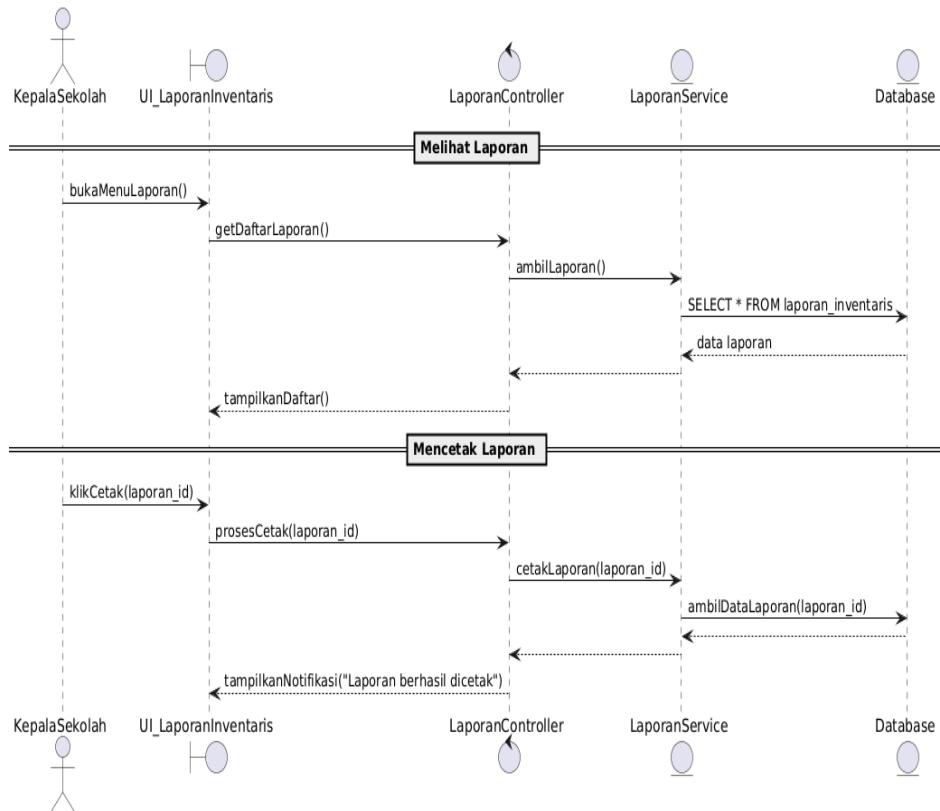
Diagram ini menunjukkan bagaimana kepala sekolah melihat laporan inventaris untuk memantau keadaan stok dan pengelolaan barang.



Gambar 4. 24 Squence Diagram Petugas Inventaris Melihat Laporan

k. Squence Diagram Kepala Sekolah Melihat Data Barang Rusak Atau Hilang

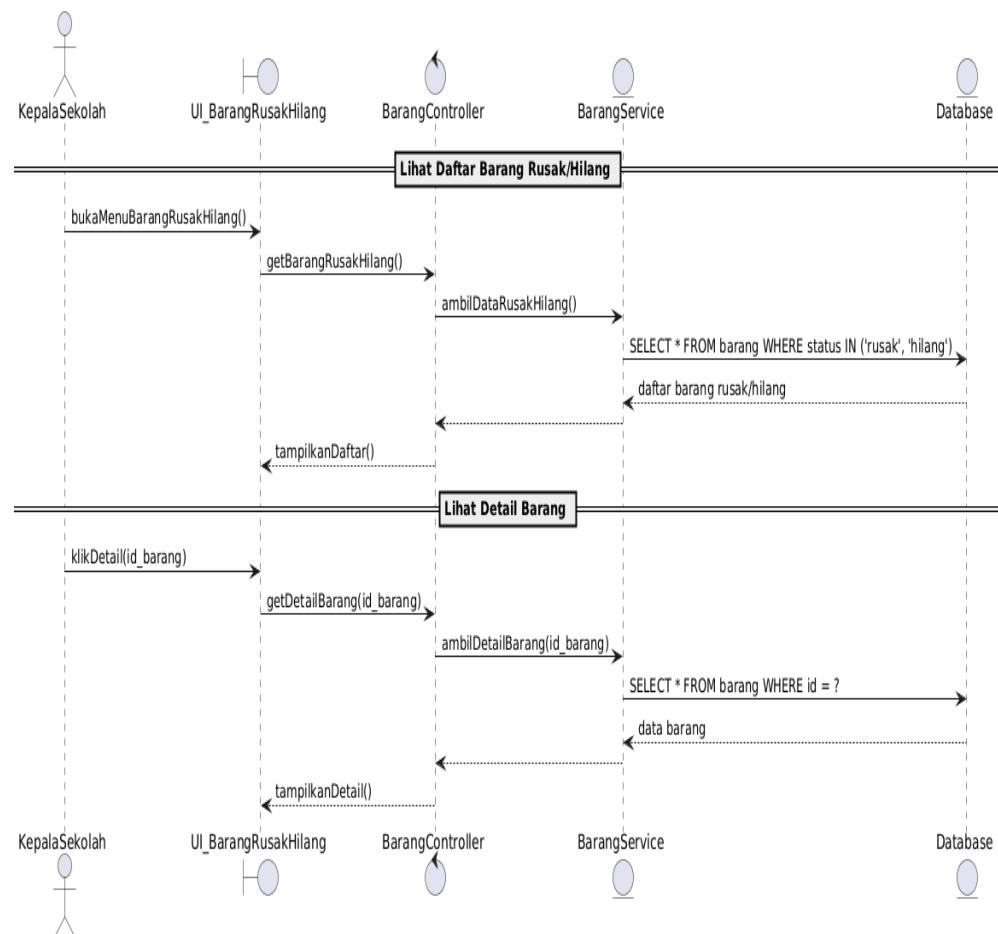
Diagram ini menggambarkan bagaimana kepala sekolah melihat data barang yang rusak atau hilang untuk memastikan pengelolaan dan penggantian barang yang tepat.



Gambar 4. 25 Squence DIagram Kepala Sekolah Melihat Laporan

1. Sequence Diagram Kepala Sekolah Monitoring Stok Barang

Diagram ini menggambarkan urutan interaksi dalam pemantauan stok barang oleh kepala sekolah, termasuk pengambilan keputusan terkait pengadaan barang baru.



Gambar 4. 26 Diagram Kepala Sekolah Lihat Barang Rusak Atau Hilang

m. Sequence Diagram Kepala Sekolah Approve Permintaan Pengadaan Barang

Diagram ini menunjukkan bagaimana kepala sekolah menyetujui atau menolak permintaan pengadaan barang berdasarkan analisis kebutuhan dan anggaran yang tersedia.



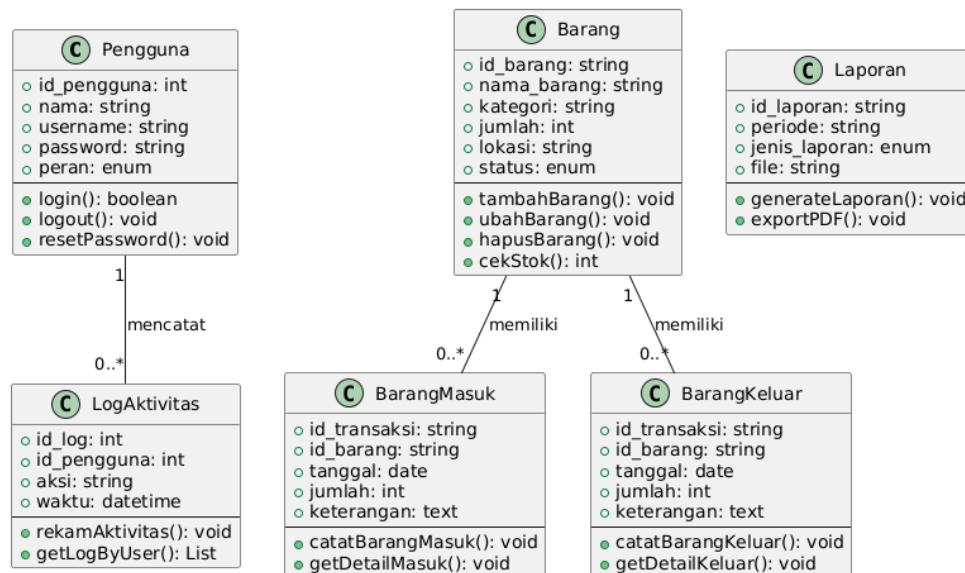
Gambar 4. 27 DIagram Kepala Sekolah Approve Pengadaan Barang

4. Class Diagram

Sistem ini dibuat untuk membantu TK TERATAI dalam mencatat dan mengelola barang secara digital. Di dalamnya, ada pengguna seperti admin atau petugas yang bisa login dan otomatis mencatat aktivitas yang mereka lakukan ke dalam log.

Setiap barang disimpan lengkap dengan nama, jumlah, lokasi, dan status. Barang ini bisa masuk atau keluar, dan semua transaksi tersebut tercatat sebagai barang masuk atau barang keluar.

Sistem juga bisa membuat laporan seperti laporan bulanan atau laporan kerusakan, yang bisa diunduh dalam bentuk PDF. Semua data saling terhubung agar proses pencatatan lebih mudah, rapi, dan tidak perlu dilakukan secara manual.



Gambar 4. 28 Class Diagram

4.3.2 Perancangan Model Data

Perancangan model data menggunakan Entity Relationship Diagram (ERD) yang menggambarkan entitas, atribut, dan relasi antar data. Entitas yang digunakan antara lain: pengguna, barang, transaksi barang masuk, transaksi barang keluar, laporan, dan log aktivitas.

4.3.3 Perancangan Fisik Basis Data

Basis data dirancang menggunakan MySQL, dengan nama basis data: tk_warehouse. Contoh struktur tabel:

1. Tabel Users

Tabel pengguna menyimpan data login dan profil semua pengguna sistem. Setiap pengguna memiliki kredensial unik serta peran untuk mengatur hak akses terhadap fitur sistem.

Tabel 4.1 Tabel Users

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO INCREMENT
2	name	VARCHAR(255)	NOT NULL
3	email	VARCHAR(255)	UNIQUE, NOT NULL
4	email verified at	TIMESTAMP	NULLABLE
5	password	VARCHAR(255)	NOT NULL
6	remember token	VARCHAR(100)	NULLABLE
7	is active	BOOLEAN	DEFAULT(true)
8	created at	TIMESTAMP	NOT NULL
9	updated at	TIMESTAMP	NOT NULL
10	deleted at	TIMESTAMP	NULLABLE

2. Tabel Kategori

Tabel kategori digunakan untuk mengelompokkan barang berdasarkan jenis atau fungsi, seperti alat tulis, elektronik, atau perlengkapan kantor, agar pengelolaan inventaris lebih terstruktur.

Tabel 4.2 Tabel Kategori

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO INCREMENT
2	nama kategori	VARCHAR(100)	NOT NULL
3	deskripsi	TEXT	NULLABLE
4	created at	TIMESTAMP	NOT NULL
5	updated at	TIMESTAMP	NOT NULL
6	deleted at	TIMESTAMP	NULLABLE

3. Tabel Lokasi

Tabel lokasi menyimpan informasi tempat penyimpanan barang seperti gedung, lantai, dan ruangan. Data ini memudahkan proses pencarian serta pengelolaan aset fisik.

Tabel 4. 3 Tabel Lokasi

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO_INCREMENT
2	nama_lokasi	VARCHAR(100)	NOT NULL
3	gedung	VARCHAR(50)	NULLABLE
4	lantai	VARCHAR(20)	NULLABLE
5	keterangan	TEXT	NULLABLE
6	created_at	TIMESTAMP	NOT NULL
7	updated_at	TIMESTAMP	NOT NULL
8	deleted_at	TIMESTAMP	NULLABLE

4. Tabel Barang

Tabel barang menyimpan seluruh data aset atau inventaris yang dimiliki. Data ini mencakup informasi kategori, lokasi, stok, serta kondisi barang untuk memudahkan pemantauan dan pengelolaan inventaris secara *real-time*.

Tabel 4. 4 Tabel Barang

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO_INCREMENT
2	kode_barang	VARCHAR(50)	UNIQUE, NOT NULL
3	nama_barang	VARCHAR(200)	NOT NULL
4	kategori_id	BIGINT	FK → kategori
5	lokasi_id	BIGINT	FK → lokasi
6	jumlah_stok	INT	DEFAULT(0)
7	reorder_point	INT	DEFAULT(10)
8	satuan	VARCHAR(50)	DEFAULT('pcs')
9	status	ENUM	DEFAULT('baik')
10	deskripsi	TEXT	NULLABLE
11	harga_satuan	DECIMAL(15,2)	NULLABLE
12	merk	VARCHAR(100)	NULLABLE
13	tanggal_pembelian	DATE	NULLABLE
14	created_at	TIMESTAMP	NOT NULL
15	updated_at	TIMESTAMP	NOT NULL
16	deleted_at	TIMESTAMP	NULLABLE

5. Tabel Barang Rusak

Tabel barang rusak mencatat aset yang mengalami kerusakan, baik sebagian maupun keseluruhan. Data ini digunakan untuk evaluasi kondisi barang dan pertanggungjawaban penggunaan aset.

Tabel 4. 6 Tabel Barang Rusak

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO_INCREMENT
2	barang_id	BIGINT	FK → barang

3	jumlah	INT	NOT NULL
4	tanggal_kejadian	DATE	NOT NULL
5	keterangan	TEXT	NULLABLE
6	penanggung_jawab	VARCHAR(100)	NOT NULL
7	user_id	BIGINT	FK → users
8	created_at	TIMESTAMP	NOT NULL
9	updated_at	TIMESTAMP	NOT NULL

6. Tabel Transaksi Barang

Tabel transaksi barang mencatat setiap aktivitas keluar atau masuk barang, termasuk proses persetujuan. Tabel ini berfungsi sebagai histori pergerakan stok dan alat kontrol inventaris.

Tabel 4. 7 Tabel Transaksi Barang

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO_INCREMENT
2	kode_transaksi	VARCHAR(50)	UNIQUE, NOT NULL
3	barang_id	BIGINT	FK → barang
4	tipe_transaksi	ENUM	NOT NULL
5	jumlah	INT	NOT NULL
6	tanggal_transaksi	DATE	NOT NULL
7	penanggung_jawab	VARCHAR(100)	NULLABLE
8	keterangan	TEXT	NULLABLE
9	user_id	BIGINT	FK → users
10	approved_by	BIGINT	FK → users
11	approved_at	TIMESTAMP	NULLABLE
12	approval_status	VARCHAR(20)	DEFAULT('pending')
13	approval_notes	TEXT	NULLABLE
14	created_at	TIMESTAMP	NOT NULL
15	updated_at	TIMESTAMP	NOT NULL

7. Tabel Backup Logs

Tabel log aktivitas digunakan untuk mencatat seluruh aktivitas pengguna dalam sistem, seperti login, tambah data, ubah data, hapus data, dan aktivitas penting lainnya. Data ini berfungsi sebagai audit trail untuk keamanan dan evaluasi sistem.

Tabel 4. 4Tabel Backup Logs

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO_INCREMENT
2	user_id	BIGINT	FK → users
3	jenis_aktivitas	ENUM	NOT NULL
4	nama_tabel	VARCHAR(100)	NULLABLE
5	record_id	BIGINT	NULLABLE

6	deskripsi	TEXT	NOT NULL
7	perubahan_data	JSON	NULLABLE
8	ip_address	VARCHAR(45)	NULLABLE
9	user_agent	TEXT	NULLABLE
10	created_at	TIMESTAMP	NOT NULL
11	updated_at	TIMESTAMP	NOT NULL

8. Tabel Log Aktifitas

Tabel *backup logs* mencatat seluruh aktivitas pencadangan (*backup*) database, termasuk informasi file, status proses, dan pengguna yang melakukan *backup*. Tabel ini membantu dalam monitoring keamanan dan pemulihan data.

Tabel 4. 5Tabel Log Aktivitas

No.	Nama Field	Tipe Data	Constraint
1	id	BIGINT	PK, AUTO_INCREMENT
2	user_id	BIGINT	FK → users
3	filename	VARCHAR(255)	NULLABLE
4	format	VARCHAR(50)	DEFAULT('sql')
5	file_size	BIGINT	NULLABLE
6	status	ENUM	DEFAULT('pending')
7	notes	TEXT	NULLABLE
8	created_at	TIMESTAMP	NOT NULL
9	updated_at	TIMESTAMP	NOT NULL
10	deleted_at	TIMESTAMP	NULLABLE

4.3.4 Perancangan Relasi Antar Tabel

Relasi antar tabel:

- pengguna ke log_aktivitas (one-to-many)
- barang ke barang_masuk dan barang_keluar (one-to-many)
- laporan terkait periode dan jenis laporan (standalone)

4.3.5 Perancangan Antarmuka

Perancangan antarmuka dilakukan untuk masing-masing peran pengguna:

- Petugas Inventaris:
 - Menu barang masuk/keluar
 - Form pencatatan transaksi
 - Monitoring stok
 - Cetak laporan bulanan
 - Laporan barang rusak/hilang

2. Admin Sistem:
 - a. Kelola pengguna
 - b. Reset password
 - c. *Backup* dan restore data
 - d. Monitoring aktivitas pengguna

3. Kepala Sekolah:
 - a. Lihat laporan bulanan
 - b. Monitoring stok barang
 - c. Lihat data barang rusak/hilang

a. Halaman Data Barang

Pada halaman ini, pengguna dapat melihat, menambah, dan memperbarui data barang. Informasi seperti nama barang, jumlah, lokasi penyimpanan, dan status barang ditampilkan secara sistematis.

Nama	Kondisi	Jumlah
Meja Guru	Bogus	15
Kursi Guru	Bogus	15
Meja Murid	Bogus	80
Kursi Murid	Bogus	150
AC*	Bogus	15
Kipas	Rusak	8
Papan Tulis	Bogus	8
Spidol*	Bogus	16
Alat Peraga Anak	Rusak	95
Mainan AP Luar	Bogus	20
Karpet	Bogus	16

Gambar 4.29 Antarmuka Kelola Data Barang

b. Halaman Pengguna

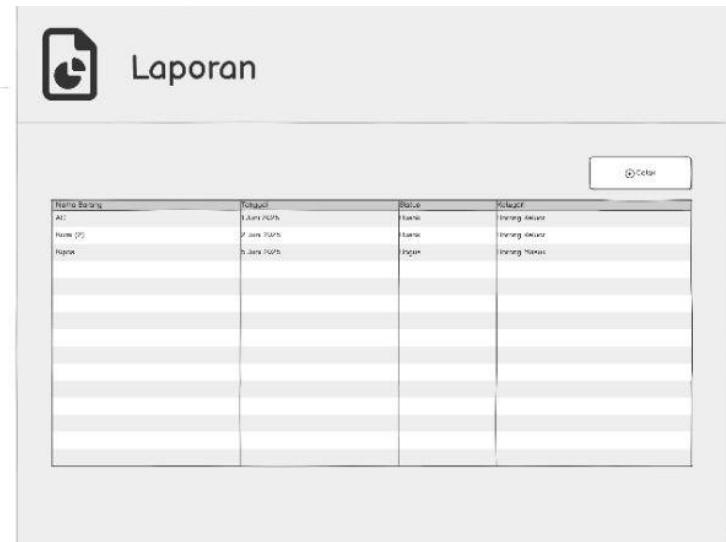
Halaman ini digunakan oleh admin untuk mengelola akun pengguna. Fitur yang tersedia mencakup penambahan, pengubahan, penghapusan akun, serta pengaturan peran dan reset password.

Username	Peran
Budiono Siregar	Petugas Inventaris

Gambar 4.30 Antarmuka Kelola Data Pengguna

c. Halaman Laporan

Halaman ini memungkinkan pengguna untuk membuat dan mengunduh laporan inventaris berdasarkan periode dan jenis (bulanan, kerusakan, atau kehilangan). Laporan dapat dicetak dalam format PDF.



Gambar 4.31 Antarmuka Kelola Laporan

d. Halaman Dashboard

Tampilan ini menyajikan dashboard utama setelah login. Informasi penting seperti jumlah total barang, aktivitas terakhir, dan peringatan stok minimum ditampilkan secara ringkas dan *real-time*.



Gambar 4.32 Antarmuka Halaman Dashboard

e. Halaman Aktivitas Log

Halaman ini mencatat seluruh aktivitas pengguna di dalam sistem. Fitur ini berguna untuk pemantauan dan audit, mencatat waktu dan jenis tindakan yang dilakukan oleh setiap pengguna.

The screenshot displays the 'Aktivitas Log' (Activity Log) page of the TK TERATAI application. At the top left is the logo 'TK TERATAI'. On the left side, there is a sidebar menu with the following items: Dashboard, Pengguna, Inventaris, Laporan, and Aktivitas log. The 'Aktivitas log' item is highlighted with a blue background. The main content area is titled 'Aktivitas Log' and features a circular icon with a clock and a double-headed arrow. Below the title is a table with three columns: 'Nama Pengguna', 'Tanggal', and 'Aksi'. The table contains four rows of data:

Nama Pengguna	Tanggal	Aksi
Administrator	7 Jun 2019	Mengakses APL
Administrator	7 Jun 2019	Mengakses APL
Administrator	8 Jun 2019	Kembali Online

Gambar 4.33 Antarmuka Halaman Aktivitas Log

BAB V

IMPLEMENTASI DAN PEMBAHASAN SISTEM

5.1. Implementasi

5.1.1 Implementasi Basis Data

a. Buat Basis Data

Membuat database baru dengan nama tk_warehouse yang digunakan sebagai basis penyimpanan seluruh data sistem. Berikut query yang di gunakan untuk membuat dengan nama tk_warehouse.

```
CREATE DATABASE tk_warehouse;
```

b. Buat Tabel Pengguna

Membuat tabel pengguna yang menyimpan data akun pengguna sistem, termasuk informasi login, status akun, dan waktu aktivitas.

```
CREATE TABLE users (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    email VARCHAR(255) NOT NULL UNIQUE,
    email_verified_at TIMESTAMP NULL,
    password VARCHAR(255) NOT NULL,
    remember_token VARCHAR(100) NULL,
    is_active BOOLEAN DEFAULT TRUE,
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    deleted_at TIMESTAMP NULL
);
```

c. Buat Tabel Kategori

Tabel kategori digunakan untuk mengelompokkan barang berdasarkan jenis atau klasifikasinya.

```
CREATE TABLE kategori (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    nama_kategori VARCHAR(100) NOT NULL,
    deskripsi TEXT NULL,
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    deleted_at TIMESTAMP NULL
);
```

d. Buat Tabel Lokasi

Tabel lokasi digunakan untuk menyimpan informasi lokasi penyimpanan barang seperti gedung, lantai, dan ruangan.

```
CREATE TABLE lokasi (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    nama_lokasi VARCHAR(100) NOT NULL,
    gedung VARCHAR(50) NULL,
    lantai VARCHAR(20) NULL,
    keterangan TEXT NULL,
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    deleted_at TIMESTAMP NULL
);
```

e. Buat Tabel Barang

Tabel barang berfungsi untuk menyimpan data inventaris barang yang dikelola dalam sistem.

```
CREATE TABLE barang (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    kode_barang VARCHAR(50) NOT NULL UNIQUE,
    nama_barang VARCHAR(200) NOT NULL,
    kategori_id BIGINT NOT NULL,
    lokasi_id BIGINT NOT NULL,
    jumlah_stok INT DEFAULT 0,
    reorder_point INT DEFAULT 10,
    satuan VARCHAR(50) DEFAULT 'pcs',
    status ENUM('baik', 'rusak', 'hilang') DEFAULT 'baik',
    deskripsi TEXT NULL,
    harga_satuan DECIMAL(15,2) NULL,
    merk VARCHAR(100) NULL,
    tanggal_pembelian DATE NULL,
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    deleted_at TIMESTAMP NULL,
    FOREIGN KEY (kategori_id) REFERENCES kategori(id),
    FOREIGN KEY (lokasi_id) REFERENCES lokasi(id)
);
```

f. Buat Tabel Barang Rusak

Tabel ini digunakan untuk mencatat barang yang mengalami kerusakan.

```
CREATE TABLE barang_rusaks (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    barang_id BIGINT NOT NULL,
    jumlah INT NOT NULL,
    tanggal_kejadian DATE NOT NULL,
    keterangan TEXT NULL,
    penanggung_jawab VARCHAR(100) NOT NULL,
    user_id BIGINT NOT NULL,
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    FOREIGN KEY (barang_id) REFERENCES barang(id),
    FOREIGN KEY (user_id) REFERENCES users(id)
);
```

g. Buat Tabel Transaksi Barang

Tabel ini mencatat semua transaksi keluar dan masuk barang.

```

CREATE TABLE transaksi_barang (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    kode_transaksi VARCHAR(50) NOT NULL UNIQUE,
    barang_id BIGINT NOT NULL,
    tipe_transaksi ENUM('masuk', 'keluar') NOT NULL,
    jumlah INT NOT NULL,
    tanggal_transaksi DATE NOT NULL,
    penanggung_jawab VARCHAR(100) NULL,
    keterangan TEXT NULL,
    user_id BIGINT NOT NULL,
    approved_by BIGINT NULL,
    approved_at TIMESTAMP NULL,
    approval_status VARCHAR(20) DEFAULT 'pending',
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    FOREIGN KEY (barang_id) REFERENCES barang(id),
    FOREIGN KEY (user_id) REFERENCES users(id),
    FOREIGN KEY (approved_by) REFERENCES users(id)
);

```

h. Buat Tabel Log Aktivitas

Tabel ini mencatat seluruh aktivitas pengguna di dalam sistem.

```

CREATE TABLE log_aktivitas (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    user_id BIGINT NOT NULL,
    jenis_aktivitas ENUM('login', 'logout', 'create',
    'update', 'delete', 'view') NOT NULL,
    nama_tabel VARCHAR(100) NULL,
    record_id BIGINT NULL,
    deskripsi TEXT NOT NULL,
    perubahan_data JSON NULL,
    ip_address VARCHAR(45) NULL,
    user_agent TEXT NULL,
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    FOREIGN KEY (user_id) REFERENCES users(id)
);

```

i. Buat Tabel Backup Logs

Tabel ini menyimpan riwayat proses *backup* database.

```

CREATE TABLE backup_logs (
    id BIGINT AUTO_INCREMENT PRIMARY KEY,
    user_id BIGINT NOT NULL,
    filename VARCHAR(255) NULL,
    format VARCHAR(50) DEFAULT 'sql',
    file_size BIGINT NULL,
    status ENUM('success', 'failed', 'pending') DEFAULT
    'pending',
    notes TEXT NULL,
    created_at TIMESTAMP NOT NULL,
    updated_at TIMESTAMP NOT NULL,
    deleted_at TIMESTAMP NULL,
    FOREIGN KEY (user_id) REFERENCES users(id)
);

```

Query DDL untuk masing-masing tabel sudah dituliskan di dalam sistem, termasuk foreign key untuk menjaga relasi antar entitas.

j. Hasil Tabel

Table	Action
backup_logs	
barang	
barang_rusaks	
kategori	
log_aktivitas	
lokasi	
transaksi_barang	
users	

Gambar 5.1 Tabel Pada Database

5.1.2 Implementasi Sistem

Sistem dikembangkan menggunakan framework Laravel Library Filament. Model-model Eloquent digunakan untuk mempermudah manipulasi data:

a. Data Barang

Modul ini menampilkan daftar seluruh barang inventaris dengan informasi lengkap. Pengguna dapat melihat kode barang, nama, kategori, jumlah stok, lokasi penyimpanan, dan status kondisi barang. Sistem memberikan penanda visual berupa warna status: hijau untuk "baik", kuning untuk "rusak", dan merah untuk "hilang". Fitur pencarian dan pengurutan memudahkan pengguna menemukan barang spesifik, serta memungkinkan pengeditan data langsung melalui tabel.

```
public static function table(Table $table): Table
{
    return $table->columns([
        Tables\Columns\TextColumn::make('kode_barang')
            ->label('Kode')
            ->searchable()
            ->sortable(),
        Tables\Columns\TextColumn::make('nama_barang')
            ->label('Nama Barang')
    ]);
}
```

```

->searchable(),
Tables\Columns\TextColumn::make('kategori.nama_kategori')
->label('Kategori'),
Tables\Columns\TextColumn::make('jumlah_stok')
->label('Jumlah')
->sortable(),

Tables\Columns\TextColumn::make('lokasi.nama_lokasi')
->label('Lokasi'),

Tables\Columns\BadgeColumn::make('status')
->label('Status')
->colors([
    'success' => 'baik',
    'warning' => 'rusak',
    'danger' => 'hilang',
])
->sortable(),
])
->filters([])
->actions([
    Tables\Actions>EditAction::make(),
])
->bulkActions([
    Tables\Actions\BulkActionGroup::make([
        Tables\Actions>DeleteBulkAction::make(),
    ]),
])
->defaultSort('nama_barang');

```

b. Transaksi Barang

Fitur ini digunakan untuk mencatat dan menampilkan riwayat seluruh transaksi barang masuk maupun keluar. Setiap transaksi dilengkapi dengan ID unik, referensi barang, tanggal pelaksanaan, jumlah barang, dan keterangan detail. Status transaksi ditandai dengan badge berwarna (hijau untuk masuk, merah untuk keluar) sehingga mudah dibedakan. Data dapat diurutkan berdasarkan tanggal terbaru, diedit untuk koreksi, atau dihapus jika diperlukan.

```

public static function table(Table $table): Table
{
    return $table->columns([

```

```

Tables\Columns\TextColumn::make('kode_transaksi')
    ->label('ID')
    ->searchable()
    ->sortable(),

Tables\Columns\TextColumn::make('barang.nama_barang')
    ->label('Barang')
    ->searchable(),

Tables\Columns\TextColumn::make('tanggal_transaksi')
    ->label('Tanggal')
    ->date()
    ->sortable(),

Tables\Columns\TextColumn::make('jumlah')
    ->label('Jumlah')
    ->sortable(),

Tables\Columns\BadgeColumn::make('tipe_transaksi')
    ->label('Jenis')
    ->colors([
        'success' => 'masuk',
        'danger' => 'keluar',
    ])
    ->formatStateUsing(fn($state) => ucfirst($state)),

Tables\Columns\TextColumn::make('keterangan')
    ->label('Keterangan')
    ->limit(30)
    ->wrap(),
])

->filters([])
->actions([
    Tables\Actions>EditAction::make(),
])
->bulkActions([
    Tables\Actions\BulkActionGroup::make([
        Tables\Actions>DeleteBulkAction::make(),
    ]),
])
->defaultSort('tanggal_transaksi', 'desc');
}

```

c. Log Aktivitas

Modul audit log mencatat semua tindakan pengguna dalam sistem seperti login, pembuatan data, pengeditan, atau penghapusan. Sistem menyimpan informasi detail meliputi pengguna yang melakukan aksi, jenis aktivitas, waktu pelaksanaan, tabel yang terpengaruh, dan deskripsi perubahan data. Log ini mendukung monitoring keamanan sistem, tracing perubahan data untuk compliance audit, dan membantu dalam investigasi jika

terjadi kesalahan atau aktivitas mencurigakan. Log ditampilkan dalam urutan terbaru terlebih dahulu dan hanya dapat dilihat (read-only).

```
public static function table(Table $table): Table
{
    return $table->columns([
        Tables\Columns\TextColumn::make('user.name')
            ->label('Pengguna')
            ->searchable(),

        Tables\Columns\TextColumn::make('jenis_aktivitas')
            ->label('Aksi')
            ->formatStateUsing(fn($state) =>
ucfirst(str_replace('_', ' ', $state)))
            ->searchable(),

        Tables\Columns\TextColumn::make('nama_tabel')
            ->label('Tabel')
            ->searchable(),

        Tables\Columns\TextColumn::make('created_at')
            ->label('Waktu')
            ->dateTime('d M Y, H:i:s')
            ->sortable(),
    ])
    ->filters([])
    ->actions([]) // Read-only, tidak ada edit/delete
    ->bulkActions([]) // Tidak ada bulk action
    ->defaultSort('created_at', 'desc');
}
```

d. Dashboard

Dashboard berfungsi sebagai landing page dan ringkasan statistik utama sistem inventaris. Tampilan dirancang untuk memberikan overview cepat kepada pengguna tentang kondisi inventaris secara keseluruhan.

Informasi yang disajikan mencakup jumlah total barang aktif dalam sistem, jumlah transaksi barang masuk, dan jumlah transaksi barang keluar. Setiap statistik ditampilkan dalam bentuk kartu (card) yang dilengkapi dengan ikon visual, deskripsi singkat, dan penanda warna untuk membedakan jenis data dan memudahkan interpretasi.

```
class StatsOverviewWidget extends BaseWidget
{
    protected function getCards(): array
    {
        return [
            Card::make('Total Barang', Barang::count())
                ->description('Data inventaris aktif')
```

```

        ->icon('heroicon-o-archive-box') ,

                Card::make('Barang Masuk' ,
TransaksiBarang::where('tipe_transaksi', 'masuk')-
>count())
                ->description('Transaksi masuk')
                ->icon('heroicon-o-arrow-down-tray')
                ->color('success') ,

                Card::make('Barang Keluar' ,
TransaksiBarang::where('tipe_transaksi', 'keluar')-
>count())
                ->description('Transaksi keluar')
                ->icon('heroicon-o-arrow-up-tray')
                ->color('danger') ,
            ];
        }
    }
}

```

e. Manajemen Pengguna

Modul User digunakan oleh administrator untuk mengelola akun pengguna sistem secara terpusat. Admin dapat menambah pengguna baru, mengubah data pengguna (nama dan email), serta menghapus akun yang tidak diperlukan. Setiap pengguna memiliki role/peran yang menentukan level akses dan izin pada fitur tertentu. Role ditampilkan menggunakan badge berwarna berbeda (hijau untuk admin, biru untuk petugas inventaris, kuning untuk kepala sekolah) untuk memudahkan identifikasi. Sistem juga mendukung pencarian pengguna berdasarkan nama atau email.

```

public static function table(Table $table): Table
{
    return $table->columns([
        Tables\Columns\TextColumn::make('name')
            ->label('Nama')
            ->searchable(),
        Tables\Columns\TextColumn::make('email')
            ->label('Email')
            ->searchable(),
        Tables\Columns\TextColumn::make('roles.name')
            ->label('Role')
            ->badge()
            ->colors([
                'success' => 'admin',
                'info' => 'petugas_inventaris',
                'warning' => 'kepala_sekolah',
            ])
    ]);
}

```

```

        ->formatStateUsing(fn($state) =>
        ucwords(str_replace('_', ' ', $state))),  

    ])  

->filters([])  

->actions([
    Tables\Actions>EditAction::make(),
])  

->bulkActions([
    Tables\Actions>BulkActionGroup::make([
        Tables\Actions>DeleteBulkAction::make(),
    ]),
])  

->defaultSort('name');

```

f. *Backup & Restore Database*

Modul *backup* memfasilitasi administrator dalam membuat *backup* database dan melakukan restore ketika diperlukan. Fitur ini penting untuk disaster recovery dan business continuity. Admin dapat membuat *backup* baru dengan sekali klik, melihat riwayat semua *backup* yang telah dibuat (dengan nama file dan ukuran), serta melakukan *restore* dari file *backup* lama. Sistem mencatat siapa yang membuat *backup*, kapan waktu pembuatan, status proses (*success/failed*), dan pesan error jika terjadi kegagalan. *Backup* file disimpan dengan format .sql di directory storage/backups. Untuk kodenya saya sertakan di lampiran.

5.2 Pembahasan

5.2.1 Pembahasan Basis Data

Pembahasan basis data yang telah dibangun dapat dilakukan dengan cara menjabarkan penggunaan *Data Manipulation Language* (DML) pada data sesuai kebutuhan organisasi/instansi/perusahaan. DML mencakup operasi-operasi dasar yakni *Create* (INSERT), *Read* (SELECT), *Update* (UPDATE), dan *Delete* (DELETE) yang diimplementasikan melalui *Eloquent ORM* dalam Laravel.

5.2.1.1. Create Data Barang

Operasi create (*INSERT*) digunakan untuk menambahkan data barang baru ke dalam sistem inventaris. Data yang disimpan meliputi kode barang,

nama, kategori, lokasi, stok awal, *reorder point*, satuan, status, dan informasi tambahan seperti harga dan merek. Setiap penambahan data barang harus dilakukan melalui form validasi untuk memastikan integritas dan konsistensi data dalam database.

```
// Create single record
$barang = Barang::create([
    'kode_barang' => 'BRG001',
    'nama_barang' => 'Pensil HB',
    'kategori_id' => 1,
    'lokasi_id' => 2,
    'jumlah_stok' => 100,
    'reorder_point' => 10,
    'satuan' => 'pcs',
    'status' => 'baik',
    'harga_satuan' => 2000.00,
    'merk' => 'Faber Castell',
    'tanggal_pembelian' => now(),
]);
;

// Using fill method
$barang = new Barang();
$barang->fill([
    'kode_barang' => 'BRG001',
    'nama_barang' => 'Pensil HB',
    'kategori_id' => 1,
])->save();

// Bulk create
Barang::insert([
    [
        'kode_barang' => 'BRG001',
        'nama_barang' => 'Pensil HB',
        'kategori_id' => 1,
        'created_at' => now(),
        'updated_at' => now(),
    ],
    [
        'kode_barang' => 'BRG002',
        'nama_barang' => 'Penghapus',
        'kategori_id' => 1,
        'created_at' => now(),
        'updated_at' => now(),
    ],
]);
;
```

5.2.1.2. Read Data Barang

Operasi read (*SELECT*) bertujuan untuk mengambil dan menampilkan data barang dari *database* sesuai kebutuhan. Data dapat diambil secara

keseluruhan, dengan filter kondisi tertentu, atau dengan relasi ke tabel lain seperti kategori dan lokasi. Operasi ini mendukung *pagination*, *sorting*, dan *filtering* untuk memudahkan navigasi data dalam skala besar.

```
// Get all barang
$barang = Barang::all();

// Get with pagination
$barang = Barang::paginate(15);

// Get specific columns
$barang = Barang::select('id', 'nama_barang', 'jumlah_stok')->get();

// Get with where clause
$barang = Barang::where('status', 'baik')
    ->where('jumlah_stok', '>', 0)
    ->get();

// Get with relationship
$barang = Barang::with('kategori', 'lokasi')->get();

// Get single record
$barang = Barang::find(1);
$barang = Barang::where('kode_barang', 'BRG001')->first();

// Get with filtering and sorting
$barang = Barang::where('kategori_id', 1)
    ->orderBy('nama_barang', 'asc')
    ->paginate(10);

// Custom scope
$barang = Barang::stokRendah()->get(); // Stok <= reorder_point
$barang = Barang::byStatus('rusak')->get(); // Filter by status

// Count records
$total = Barang::count();
$stokRendah = Barang::where('jumlah_stok', '<', 10)->count();
```

5.2.1.3. Update Data Barang

Operasi update (*UPDATE*) memungkinkan perubahan data barang yang sudah tersimpan di database. Update dapat dilakukan pada satu atau beberapa kolom, baik untuk satu *record* maupun *multiple records* sekaligus. Setiap update mencatat waktu perubahan secara otomatis melalui timestamp dan harus melewati validasi sebelum disimpan.

```

$barang = Barang::find(1);
$barang->update([
    'nama_barang' => 'Pensil 2B',
    'jumlah_stok' => 150,
    'harga_satuan' => 2500.00,
]);
// Update with save method
$barang = Barang::find(1);
$barang->nama_barang = 'Pensil 2B';
$barang->jumlah_stok = 150;
$barang->save();

// Update multiple records with where
Barang::where('kategori_id', 1)
->update([
    'status' => 'baik',
    'updated_at' => now(),
]);
// Increment/Decrement
Barang::find(1)->increment('jumlah_stok', 50); // Tambah stok
Barang::find(1)->decrement('jumlah_stok', 20); // Kurangi stok

// Update using firstOrCreate
$barang = Barang::firstOrCreate(
    ['kode_barang' => 'BRG001'],
    ['nama_barang' => 'Pensil HB', 'kategori_id' => 1]
);

// Update using updateOrCreate
$barang = Barang::updateOrCreate(
    ['kode_barang' => 'BRG001'],
    ['nama_barang' => 'Pensil HB Updated', 'jumlah_stok' => 200]
);

```

5.2.1.4. Delete Data Barang

Operasi delete (DELETE) digunakan untuk menghapus data barang dari *database*. Sistem menggunakan *soft delete* sehingga data tidak benar-benar terhapus melainkan ditandai dengan *timestamp deleted_at*. Fitur ini memungkinkan pemulihan data jika terjadi kesalahan penghapusan dan mendukung *audit trail*

```

// Soft delete single record
$barang = Barang::find(1);
$barang->delete();

```

```

// Force delete (permanent)
$barang = Barang::find(1);
$barang->forceDelete();

// Delete multiple records
Barang::where('status', 'hilang')->delete();

// Restore soft deleted record
$barang = Barang::withTrashed()->find(1);
$barang->restore();

// Permanently delete trashed record
$barang = Barang::onlyTrashed()->find(1);
$barang->forceDelete();

// Query only trashed records
$trashedBarang = Barang::onlyTrashed()->get();

// Query including trashed records
$allBarang = Barang::withTrashed()->get();

// Force delete multiple
Barang::where('kategori_id', 1)->forceDelete();

// Restore multiple
Barang::onlyTrashed()
    ->where('kategori_id', 1)
    ->restore();

```

5.2.1.5. Create Data Kategori

Operasi create pada tabel kategori bertujuan menambahkan jenis/klasifikasi barang baru ke dalam sistem. Setiap kategori diberi nama dan deskripsi untuk memudahkan pengelompokan inventaris. Kategori merupakan data master yang harus ada sebelum membuat data barang untuk menjaga integritas relasi *foreign key*

```

// Create kategori
$kategori = Kategori::create([
    'nama_kategori' => 'Alat Tulis',
    'deskripsi' => 'Peralatan tulis kantor dan sekolah',
]);

// Using fill and save
$kategori = new Kategori();
$kategori->fill([
    'nama_kategori' => 'Peralatan Kantor',
    'deskripsi' => 'Perlengkapan umum kantor',
])->save();

```

```
// Bulk insert kategori
Kategori::insert([
    [
        'nama_kategori' => 'Alat Tulis',
        'deskripsi' => 'Peralatan tulis kantor',
        'created_at' => now(),
        'updated_at' => now(),
    ],
    [
        'nama_kategori' => 'Peralatan Teknologi',
        'deskripsi' => 'Perangkat dan aksesori teknologi',
        'created_at' => now(),
        'updated_at' => now(),
    ],
    [
        'nama_kategori' => 'Perlengkapan Laboratorium',
        'deskripsi' => 'Alat dan bahan laboratorium',
        'created_at' => now(),
        'updated_at' => now(),
    ],
]);
]) ;
```

5.2.1.6. Read Data Kategori

Operasi read pada kategori mengambil data jenis-jenis barang dari database untuk ditampilkan dalam form *dropdown* atau tabel. Data kategori dapat diambil seluruhnya atau dengan relasi ke tabel barang untuk melihat berapa banyak barang dalam setiap kategori. Kategori sering digunakan sebagai filter dalam query barang untuk pencarian dan reporting.

```
// Get all kategori
$kategori = Kategori::all();

// Get with relationship count
$kategori = Kategori::withCount('barang')->get();

// Get kategori with barang
$kategori = Kategori::with('barang')->get();

// Get with pagination
$kategori = Kategori::paginate(10);

// Search by name
$kategori      =      Kategori::where('nama_kategori',      'like',
'%Alat%')
->get();

// Get single kategori
$kategori = Kategori::find(1);
```

```

$kategori = Kategori::where('nama_kategori', 'Alat Tulis')->first();

// Get kategori with active barang only
$kategori = Kategori::with([
    'barang' => function($query) {
        $query->where('status', 'baik');
    }
])->get();

// Get kategori ordered by name
$kategori = Kategori::orderBy('nama_kategori', 'asc')->get();

```

5.2.1.7. Update Data Kategori

Operasi update pada kategori mengubah nama atau deskripsi dari jenis barang yang sudah terdaftar. Update pada kategori dapat mempengaruhi semua barang yang terkait karena relasi foreign key. Perubahan nama kategori harus dilakukan dengan hati-hati untuk memastikan konsistensi data dan tidak mengganggu proses bisnis

```

// Update kategori
$kategori = Kategori::find(1);
$kategori->update([
    'nama_kategori' => 'Perlengkapan Tulis',
    'deskripsi' => 'Semua peralatan menulis dan menggambar',
]);

// Update with save
$kategori = Kategori::find(1);
$kategori->nama_kategori = 'Perlengkapan Tulis';
$kategori->deskripsi = 'Alat menulis dan menggambar';
$kategori->save();

// Update multiple kategori
Kategori::where('nama_kategori', 'like', 'Alat%')
    ->update(['deskripsi' => 'Updated description']);

// updateOrCreate
$kategori = Kategori::updateOrCreate(
    ['nama_kategori' => 'Alat Tulis'],
    ['deskripsi' => 'Peralatan tulis kantor dan sekolah updated']
);

```

5.2.1.8. Create Data Kategori Barang

Operasi create transaksi mencatat setiap pergerakan barang masuk atau keluar dari gudang. Transaksi yang dibuat meliputi jenis transaksi (masuk/keluar), barang yang terlibat, jumlah, tanggal, penanggung jawab, dan keterangan. Setiap transaksi secara otomatis mengurangi atau menambah stok barang untuk menjaga akurasi inventaris *real-time*

```
// Create transaksi barang masuk
$transaksi = TransaksiBarang::create([
    'kode_transaksi' => 'TRX-IN-001',
    'barang_id' => 1,
    'tipe_transaksi' => 'masuk',
    'jumlah' => 50,
    'tanggal_transaksi' => now()->toDateString(),
    'penanggung_jawab' => 'Budi Santoso',
    'keterangan' => 'Pembelian dari supplier',
    'user_id' => 1,
    'approval_status' => 'pending',
]);

// Create transaksi barang keluar
$transaksi = TransaksiBarang::create([
    'kode_transaksi' => 'TRX-OUT-001',
    'barang_id' => 2,
    'tipe_transaksi' => 'keluar',
    'jumlah' => 10,
    'tanggal_transaksi' => now()->toDateString(),
    'penanggung_jawab' => 'Andi Wijaya',
    'keterangan' => 'Penggunaan untuk kegiatan',
    'user_id' => 2,
    'approval_status' => 'pending',
]);

// Create with auto approval
$transaksi = TransaksiBarang::create([
    'kode_transaksi' => 'TRX-001',
    'barang_id' => 1,
    'tipe_transaksi' => 'masuk',
    'jumlah' => 100,
    'tanggal_transaksi' => now()->toDateString(),
    'user_id' => 1,
    'approved_by' => 3,
    'approved_at' => now(),
    'approval_status' => 'approved',
]);
```

5.2.1.9. Read Data Transaksi Barang

Operasi read transaksi mengambil riwayat pergerakan barang untuk keperluan pelaporan dan monitoring. Data transaksi dapat ditampilkan dengan filter berdasarkan jenis transaksi, periode, status *approval*, atau barang spesifik. Operasi ini mendukung grouping dan aggregation untuk analisis tren pergerakan barang dalam kurun waktu tertentu

```
// Create single record
$barang = Barang::create([
    'kode_barang' => 'BRG001',
    'nama_barang' => 'Pensil HB',
    'kategori_id' => 1,
    'lokasi_id' => 2,
    'jumlah_stok' => 100,
    'reorder_point' => 10,
    'satuan' => 'pcs',
    'status' => 'baik',
    'harga_satuan' => 2000.00,
    'merk' => 'Faber Castell',
    'tanggal_pembelian' => now(),
]);
;

// Using fill method
$barang = new Barang();
$barang->fill([
    'kode_barang' => 'BRG001',
    'nama_barang' => 'Pensil HB',
    'kategori_id' => 1,
])->save();

// Bulk create
Barang::insert([
    [
        'kode_barang' => 'BRG001',
        'nama_barang' => 'Pensil HB',
        'kategori_id' => 1,
        'created_at' => now(),
        'updated_at' => now(),
    ],
    [
        'kode_barang' => 'BRG002',
        'nama_barang' => 'Penghapus',
        'kategori_id' => 1,
        'created_at' => now(),
        'updated_at' => now(),
    ],
]);
// Get all transaksi
$transaksi = TransaksiBarang::all();

// Get with relationships
```

```

$transaksi      = TransaksiBarang::with('barang',      'user',
'approvedBy')
->paginate(15);

// Get transaksi masuk
$masuk = TransaksiBarang::masuk()->get();

// Get transaksi keluar
$keluar = TransaksiBarang::keluar()->get();

// Get transaksi pending approval
$pending = TransaksiBarang::pending()->get();

// Get transaksi approved
$approved      = TransaksiBarang::where('approval_status',
'approved')->get();

// Get transaksi by period
$transaksi      =
TransaksiBarang::whereBetween('tanggal_transaksi', [
    now()->startOfMonth(),
    now()->endOfMonth()
])->get();

// Get transaksi by barang
$transaksi = TransaksiBarang::where('barang_id', 1)
->orderBy('tanggal_transaksi', 'desc')
->paginate(10);

// Get transaksi with sum jumlah
$total = TransaksiBarang::where('barang_id', 1)
->where('tipe_transaksi', 'masuk')
->sum('jumlah');

// Get grouped by barang
$transaksi = TransaksiBarang::selectRaw(
    'barang_id, tipe_transaksi, SUM(jumlah) as total'
)
->groupBy('barang_id', 'tipe_transaksi')
->get();

```

5.2.1.10. Create Data Log Aktivitas

Operasi create log aktivitas mencatat setiap aksi pengguna dalam sistem secara otomatis melalui observer. Log mencakup jenis aktivitas (login, create, update, delete), user yang melakukan aksi, tabel yang terpengaruh, dan perubahan data detail dalam format JSON. Sistem logging ini mendukung keamanan dan audit trail untuk compliance requirement organisasi

```
// Create log aktivitas
```

```

$log = LogAktivitas::create([
    'user_id' => 1,
    'jenis_aktivitas' => 'create',
    'nama_tabel' => 'barang',
    'record_id' => 1,
    'deskripsi' => 'Membuat data barang baru: Pensil HB',
    'perubahan_data' => json_encode([
        'kode_barang' => 'BRG001',
        'nama_barang' => 'Pensil HB',
        'jumlah_stok' => 100,
    ]),
    'ip_address' => '192.168.1.1',
    'user_agent' => 'Mozilla/5.0...',
]);

```

// Create update log

```

$log = LogAktivitas::create([
    'user_id' => 2,
    'jenis_aktivitas' => 'update',
    'nama_tabel' => 'barang',
    'record_id' => 1,
    'deskripsi' => 'Update stok barang',
    'perubahan_data' => json_encode([
        'old' => ['jumlah_stok' => 100],
        'new' => ['jumlah_stok' => 150],
    ]),
    'ip_address' => request()->ip(),
    'user_agent' => request()->userAgent(),
]);

```

// Create delete log

```

$log = LogAktivitas::create([
    'user_id' => 3,
    'jenis_aktivitas' => 'delete',
    'nama_tabel' => 'barang',
    'record_id' => 5,
    'deskripsi' => 'Menghapus data barang: Penghapus',
    'ip_address' => request()->ip(),
    'user_agent' => request()->userAgent(),
]);

```

5.2.2. Pembahasan Sistem

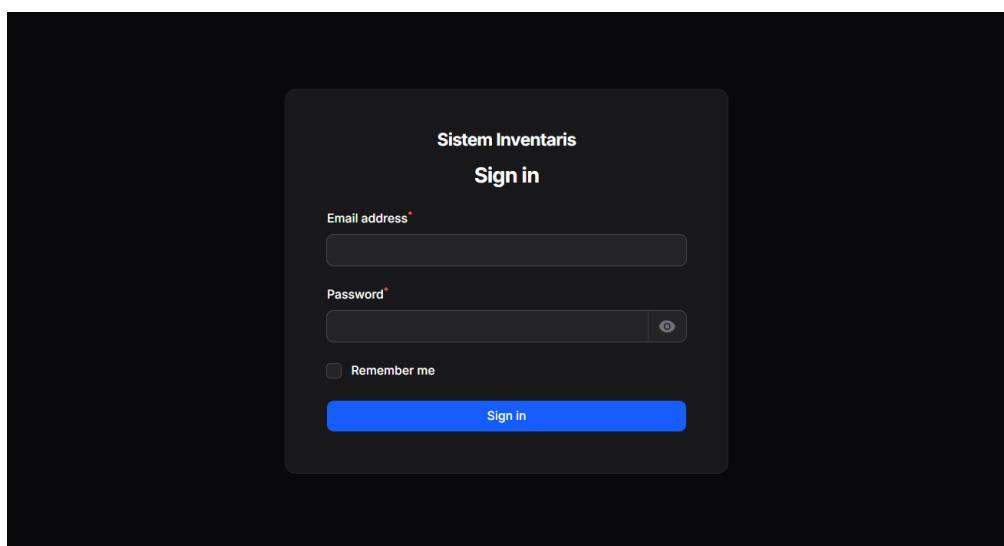
Antarmuka sistem terbagi menjadi tiga role:

- Admin: dapat memantau log aktivitas, reset password, dan *backup-restore* data.
- Petugas inventaris: dapat mencatat barang masuk/keluar, memeriksa stok, serta mencetak laporan bulanan.
- Kepala sekolah: dapat memantau laporan inventaris dan menyetujui permintaan pengadaan barang.

Setiap fitur diimplementasikan menggunakan validasi form dan alert untuk meningkatkan keandalan sistem.

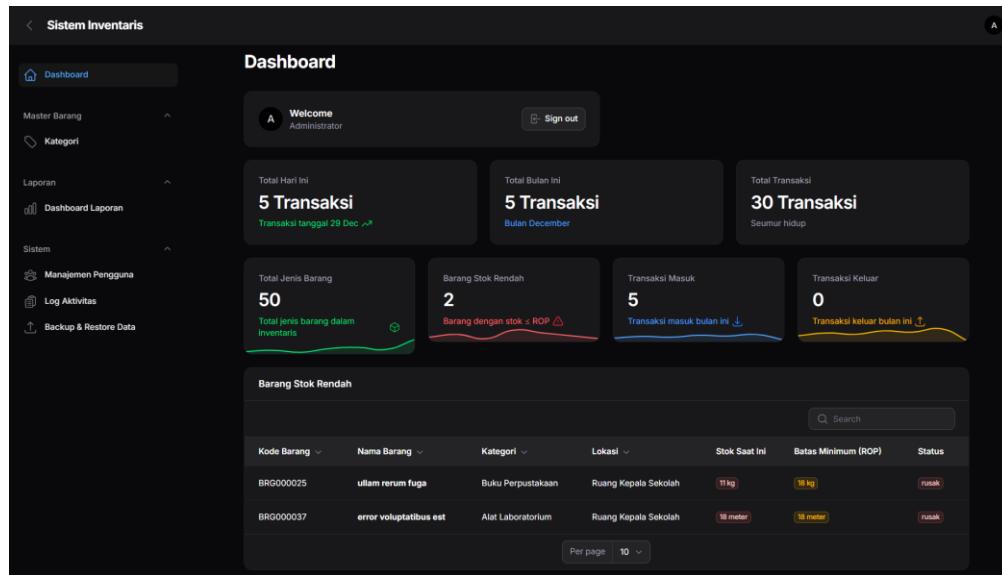
5.2.1.11. Halaman Login

Halaman login merupakan pintu masuk utama sistem inventaris barang sekolah yang memastikan setiap pengguna terauthentikasi dengan benar sebelum mengakses aplikasi. Pada halaman ini, pengguna (admin, petugas inventaris, atau kepala sekolah) memasukkan kredensial mereka berupa email dan *password*, kemudian sistem akan memvalidasi data tersebut dengan *database*. Jika login berhasil, user akan diarahkan ke dashboard admin sesuai dengan *rolenya*, dan sistem akan mencatat waktu login untuk audit trail. Fitur keamanan seperti *password hashing* dan *rate limiting* juga diimplementasikan untuk mencegah *brute force attack*.



5.2.1.12. Halaman Dashboard Admin

Halaman dashboard admin adalah pusat kontrol administrator yang menampilkan ringkasan statistik menyeluruh mengenai kinerja sistem inventaris. Dashboard ini menampilkan *widget-widget* penting seperti total barang, total kategori, total lokasi, dan total transaksi dalam sistem. Admin dapat melihat grafik *trend* transaksi, *monitoring* stok barang yang rendah, dan *alert* untuk barang dengan kondisi rusak. Setiap widget dapat diklik untuk *drill-down* ke laporan yang lebih detail dan memanage data terkait.



5.2.1.13. Halaman Manajemen Kategori Barang

Halaman manajemen kategori memudahkan admin mengorganisir barang ke dalam kategori-kategori yang logis dan mudah dipahami oleh petugas inventaris. Admin dapat membuat kategori baru, edit nama/deskripsi kategori, atau menghapus kategori yang sudah tidak digunakan. Sistem mencegah penghapusan kategori yang masih memiliki barang yang teraut. Fitur *soft delete* memastikan data kategori dapat dipulihkan jika terjadi penghapusan yang tidak disengaja, dan setiap perubahan tercatat dalam log aktivitas untuk audit purposes.

Nama Kategori	Keterangan	Jumlah Barang	Action
Alat Kebersihan	Sapu, pel, pembersih lantai, lap pel	4 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Alat Laboratorium	Peralatan praktikum sains dan teknologi	6 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Alat Tulis Kantor	Kertas, pulpen, spidol, penghapus, tinta printer	7 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Buku Perpustakaan	Buku pelajaran, buku referensi, novel	8 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Elektronik	Peralatan elektronik seperti komputer, printer, pr...	6 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit

5.2.1.14. Halaman Manajemen Lokasi Penyimpanan

Halaman manajemen lokasi memudahkan admin mengelola berbagai tempat penyimpanan barang di sekolah seperti gudang, ruang kelas, laboratorium, perpustakaan, dll. Admin dapat membuat lokasi baru, edit informasi lokasi (nama, gedung, lantai, keterangan), atau menghapus lokasi yang sudah tidak digunakan. Setiap lokasi menampilkan informasi detail seperti nama lokasi, gedung, lantai, dan deskripsi lokasi. Sistem mencegah penghapusan lokasi yang masih memiliki barang yang disimpan di dalamnya.

Nama Lokasi	Keterangan	Jumlah Barang	Action
Gudang Penyimpanan	Gudang utama untuk penyimpanan barang inventaris	6 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Laboratorium Komputer	Lab komputer untuk praktikum siswa	8 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Perpustakaan	Ruang perpustakaan sekolah	11 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Ruang Kelas	Ruang belajar mengajar siswa	11 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit
Ruang Kepala Sekolah	Ruang kerja kepala sekolah	14 barang	<input type="radio"/> View <input checked="" type="checkbox"/> Edit

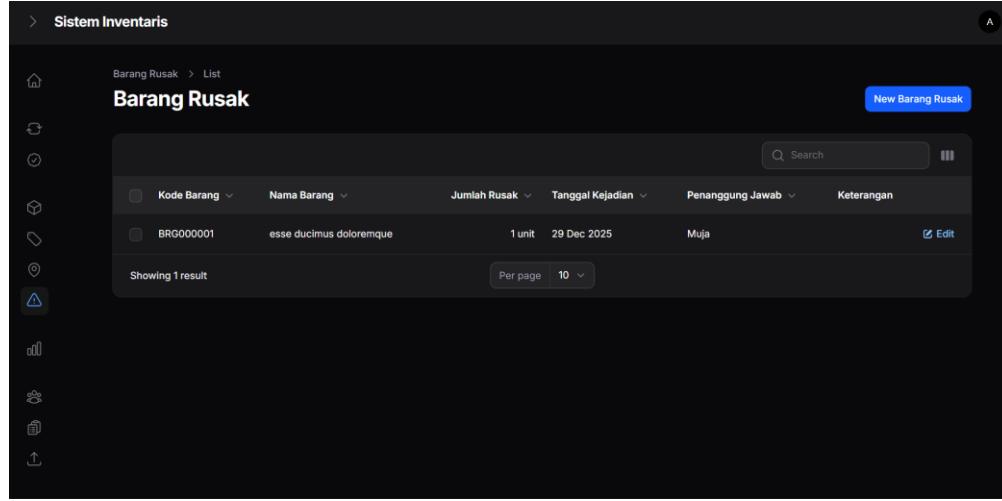
5.2.1.15. Halaman Manajemen Data Barang

Halaman manajemen barang adalah dashboard lengkap untuk admin mengelola semua data barang inventaris yang dimiliki sekolah. Admin dapat melihat tabel barang dengan kolom lengkap seperti kode barang, nama, kategori, lokasi, stok (jumlah), *reorder point*, satuan, harga, merk, status kondisi, dan tanggal pembelian. Fitur pencarian dan filter memudahkan navigasi dalam katalog barang yang besar. Admin dapat melakukan operasi seperti tambah barang baru, edit informasi barang, melihat detail barang, atau menghapus barang. Setiap barang menampilkan status kondisi (baik/rusak/hilang) dan informasi stok minimum yang harus di-maintain.

Kode	Nama Barang	Kategori	Lokasi	Stok	Merk	Harga	Status
BRG000030	magni autem excepturi	Alat Tulis Kantor	Gudang Penyimpanan	71 unit	Dell	IDR 276,098.00	Baik
BRG000034	nihil voluptatem dolorem	Furniture	Ruang Kepala Sekolah	43 unit	LG	IDR 206,584.00	Baik
BRG000035	quis cumque consequatur	Furniture	Gudang Penyimpanan	100 unit	LG	IDR 188,635.00	Baik
BRG000044	quo enim voluptatibus	Furniture	Perpustakan	63 unit	Brother	IDR 475,131.00	Baik
BRG000045	aut delectus ut	Alat Laboratorium	Laboratorium Komputer	128 unit	HP	IDR 114,638.00	Baik

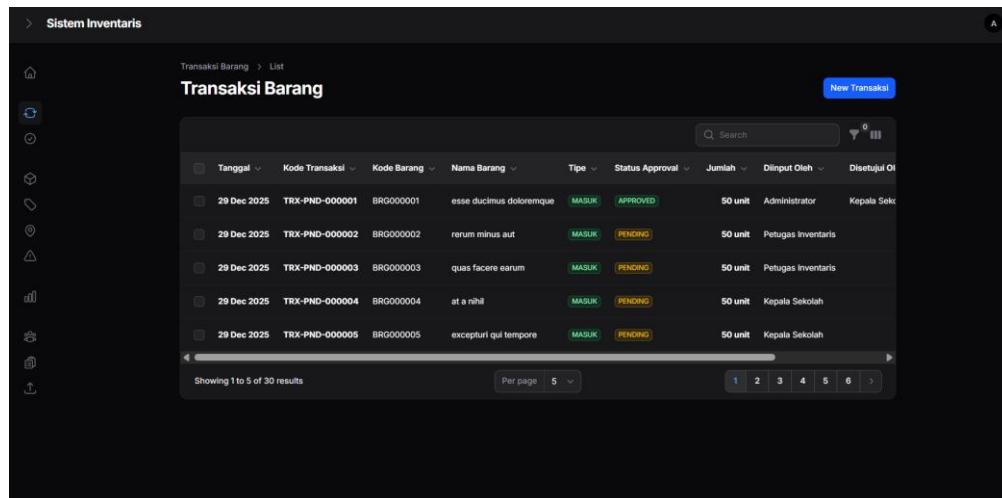
5.2.1.16. Halaman Manajemen Barang Rusak

Halaman manajemen barang rusak memberikan *visibility* khusus terhadap barang yang mengalami kerusakan dan perlu ditangani. Admin dapat melihat daftar barang rusak dengan informasi seperti barang yang rusak, lokasi ditemukan, deskripsi kerusakan, tanggal pelaporan, dan status penanganan (pending/diperbaiki/dibuang). Admin dapat melakukan operasi seperti tambah laporan barang rusak baru, edit status penanganan, atau melihat detail barang rusak. Sistem mengintegrasikan dengan master barang untuk mencegah penjualan barang yang rusak.



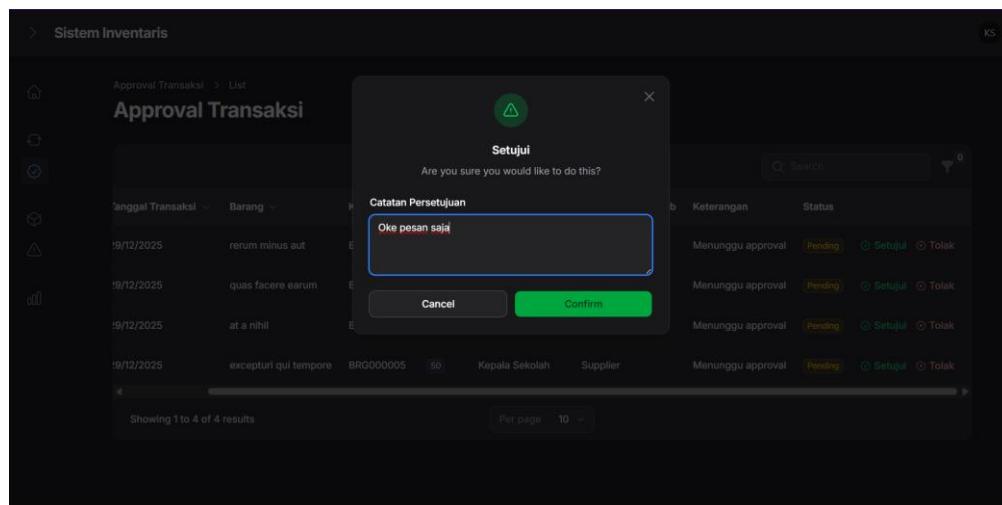
5.2.1.17.Halaman Manajemen Transaksi Barang

Halaman manajemen transaksi adalah dashboard lengkap untuk admin mengelola semua pergerakan barang masuk dan keluar dari gudang. Admin dapat melihat daftar transaksi dengan informasi lengkap seperti kode transaksi, jenis transaksi (masuk/keluar), barang terlibat, jumlah, tanggal transaksi, penanggung jawab, keterangan, dan status *approval*. Fitur pencarian dan filter memudahkan admin menemukan transaksi spesifik berdasarkan periode, jenis, atau status. Admin dapat melakukan operasi seperti tambah transaksi baru, edit transaksi, melihat detail, atau melihat *approval* history. Sistem otomatis validasi untuk mencegah transaksi keluar yang melebihi stok yang tersedia.



5.2.1.18. Halaman Persetujuan Transaksi

Halaman persetujuan transaksi adalah *workflow approval* untuk semua transaksi barang yang dibuat oleh petugas inventaris dengan status *approval* yang masih pending. Admin/Kepala Sekolah dapat melihat daftar transaksi pending yang menunggu persetujuan, lengkap dengan detail transaksi seperti barang, jumlah, dan keterangan. Mereka dapat melakukan *review* dan kemudian *approve* atau *reject* transaksi. Jika *approve*, stok barang akan terupdate dan transaksi menjadi valid. Jika *reject*, transaksi akan kembali ke status draft dan petugas dapat merevisi.



5.2.1.19. Halaman Log Aktivitas Sistem

Halaman log aktivitas menampilkan audit trail lengkap dari semua aksi yang dilakukan pengguna dalam sistem untuk keperluan server dan *security*. Admin dapat melihat daftar lengkap log dengan informasi seperti user yang melakukan aksi, jenis aktivitas (*create/read/update/delete/login*), tabel yang terpengaruh, *record ID*, deskripsi aktivitas, perubahan data (dalam format JSON), IP *address*, user agent, dan *timestamp*. Fitur filter memudahkan pencarian berdasarkan user, jenis aktivitas, tabel, atau periode waktu. Setiap log adalah *read-only* untuk menjaga integritas audit trail.

Tanggal	Aktivitas	Tabel	User	Role	Deskripsi
28 Aug 2025 15:22:29	DELETE	Lokasi	Kepala Sekolah		Beatae saepe perspiciatis aut voluptate...
21 Aug 2025 03:58:42	LOGIN	User	Kepala Sekolah		Ratione voluptas eveniet ea saepe a volu...
12 Aug 2025 14:32:00	UPDATE	Kategori	Administrator		Autem commodi quibusdam soluta alias mag...
09 Aug 2025 20:38:35	LOGIN	User	Administrator		In quasi sed aut incidenti ure qui non.
05 Aug 2025 17:19:55	CREATE	Lokasi	Petugas Inventaris		Laboriosam eaque soluta voluptatem liber...

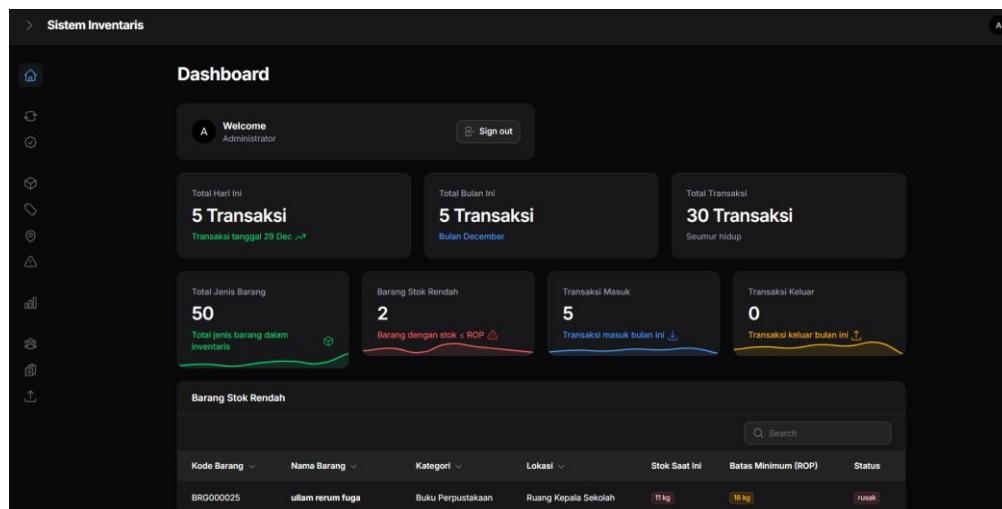
5.2.1.20. Halaman Manajemen Pengguna

Halaman manajemen pengguna memungkinkan admin untuk melakukan CRUD terhadap akun pengguna sistem, baik itu petugas inventaris, kepala sekolah, maupun admin lainnya. Admin dapat melihat daftar lengkap pengguna dengan informasi nama, email, role, status aktif, dan waktu terakhir login. Fitur pencarian dan filter memudahkan admin menemukan user tertentu, sementara tombol action memungkinkan edit profil, *assign* roles, atau menghapus akun user. Sistem menggunakan *Spatie Permission* untuk *flexible* role dan *permission management*. Setiap perubahan terhadap *user* tercatat dalam log aktivitas untuk *security* dan *compliance*.

Nama Pengguna	Email	Peran	Aktif
Administrator	admin@inventaris.test	Admin Sistem	<input checked="" type="checkbox"/> Edit
Petugas Inventaris	petugas@inventaris.test	Petugas Inventaris	<input checked="" type="checkbox"/> Edit
Kepala Sekolah	kepala@inventaris.test	Kepala Sekolah	<input checked="" type="checkbox"/> Edit

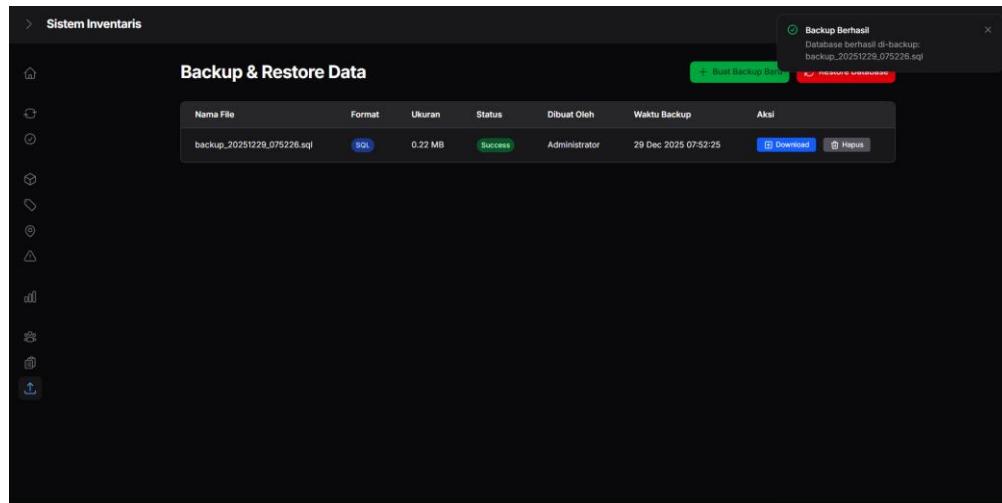
5.2.1.21. Halaman Dahboard Laporan Analytics

Halaman dashboard laporan adalah halaman *analytics* komprehensif yang menampilkan statistik dan *insight* mendalam dari data inventaris. Halaman ini menampilkan berbagai *widget analytics* seperti total barang per kategori, distribusi barang per lokasi, *trend* transaksi masuk vs keluar, barang dengan stok rendah, dan statistik barang rusak. Admin dapat melihat visualisasi data dalam bentuk *charts* dan *tables* yang interaktif, dan melakukan filter berdasarkan periode waktu. Data *analytics* ini membantu admin membuat keputusan strategis terkait *procurement* dan *inventory management*.



5.2.1.22. Halaman Backup Dan Restore

Halaman *backup* & restore memberikan admin kemampuan untuk membuat *backup database* sistem dan melakukan restore dari *backup* file yang sebelumnya dibuat. Admin dapat melihat *history* semua *backup* yang telah dibuat dengan informasi seperti filename, ukuran file, waktu *backup*, status (*success/failed*), dan user yang melakukan *backup*. Fitur utama di halaman ini adalah tombol "*Create Backup*" untuk membuat *backup* baru dan "*Upload & Restore*" untuk *restore* dari file *backup*. Sistem *backup* dilakukan dalam format SQL dan disimpan di storage/*backups* dengan timestamped filename untuk *easy reference*.



5.2.3 Pengujian

Pengujian black box dilakukan untuk memastikan setiap fitur sistem inventaris berjalan sesuai kebutuhan fungsional. Pengujian dilakukan dengan memberikan *input* tertentu dan mengamati *output* yang dihasilkan tanpa melihat logika internal sistem.

5.2.2.1. Pengujian Fitur Login

Pengujian fitur login dilakukan untuk memastikan bahwa sistem hanya dapat diakses oleh pengguna yang memiliki kredensial dan peran yang valid. Pengujian ini mencakup skenario login berhasil, kesalahan autentikasi, serta pembatasan akses berdasarkan hak pengguna.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Login Admin valid	<u>admin@inventaris.test</u> / password	Berhasil masuk dashboard admin	✓
1.2	Login Petugas valid	<u>petugas@inventaris.test</u> / password	Berhasil masuk dashboard petugas	✓
1.3	Login Kepala valid	<u>kepala@inventaris.test</u> / password	Berhasil masuk dashboard kepala sekolah	✓
1.4	Password salah	<u>admin@inventaris.test</u> / wrongpassword	Login ditolak, pesan kesalahan tampil	✓

5.2.2.2. Pengujian Fitur Manajemen Barang

Pengujian pada fitur manajemen barang bertujuan untuk memastikan sistem mampu mengelola data barang secara akurat. Pengujian dilakukan

terhadap proses penambahan, perubahan, dan validasi data barang agar konsistensi dan keandalan data inventaris tetap terjaga.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Tambah barang baru	Meja Guru, Furniture, Stok 10	Barang berhasil ditambahkan	✓
1.2	Kode barang custom	FRN-MJ-001, Meja Siswa, Stok 50	Barang tersimpan dengan kode custom	✓
1.3	Update nama barang	Meja Kayu → Meja Kayu Jati	Nama barang diperbarui	✓
1.4	Update stok barang	Stok 10 → 15	Stok berhasil diperbarui	✓

5.2.2.3. Pengujian Fitur Manajemen Kategori

Pengujian fitur manajemen kategori dilakukan untuk memastikan sistem mampu mengelompokkan data barang secara terstruktur. Pengujian mencakup proses penambahan, perubahan, dan penghapusan kategori serta validasi data agar tidak terjadi duplikasi atau kesalahan pengelompokan.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Tambah kategori	Elektronik, Kode ELK	Kategori berhasil ditambahkan	✓
1.2	Kode kategori kosong	Nama kategori tanpa kode	Validasi error muncul	✓
1.3	Update kategori	Elektronik → Peralatan Elektronik	Nama kategori diperbarui	✓
1.4	Hapus kategori kosong	Kategori tanpa barang	Kategori berhasil dihapus	✓

5.2.2.4. Pengujian Fitur Manajemenn Lokasi

Pengujian pada fitur manajemen lokasi bertujuan untuk memastikan pendataan lokasi penyimpanan barang berjalan dengan baik. Pengujian dilakukan terhadap proses penambahan, pembaruan, dan penghapusan lokasi guna menjaga ketepatan informasi posisi barang.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Tambah lokasi	Ruang Guru, Kode RG	Lokasi berhasil ditambahkan	✓
1.2	Kode lokasi kosong	Nama lokasi tanpa kode	Validasi error muncul	✓

1.3	Update lokasi	Ruang Guru → Ruang Guru Lt.2	Lokasi diperbarui	✓
1.4	Hapus lokasi kosong	Lokasi tanpa barang	Lokasi berhasil dihapus	✓

5.2.2.5. Pengujian Fitur Transaksi Barang

Pengujian fitur transaksi barang dilakukan untuk memastikan sistem mampu mencatat pergerakan barang masuk dan keluar secara akurat. Pengujian difokuskan pada validasi jumlah barang, perubahan stok, serta pembatasan transaksi agar tidak melebihi stok tersedia.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Transaksi barang masuk	Barang A, Qty 10	Stok bertambah 10	✓
1.2	Transaksi barang keluar valid	Barang A, Qty 5	Stok berkurang 5	✓
1.3	Qty keluar melebihi stok	Barang A, Qty 20	Sistem menolak transaksi	✓
1.4	Qty bernilai nol	Barang A, Qty 0	Validasi error ditampilkan	✓

5.2.2.6. Pengujian Fitur Barang Rusak

Pengujian fitur barang rusak bertujuan memastikan sistem mampu mencatat kondisi barang yang mengalami kerusakan. Pengujian mencakup pencatatan laporan, perubahan status kerusakan, serta penyesuaian stok agar data inventaris tetap akurat.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Laporkan barang rusak	Barang A, Qty rusak 2	Laporan tersimpan	✓
1.2	Update status perbaikan	Status: Dalam Perbaikan	Status berhasil diperbarui	✓
1.3	Barang dihapuskan	Status: Dihapuskan	Stok berkurang	✓
1.4	Qty rusak melebihi stok	Qty rusak > stok	Sistem menolak input	✓

5.2.2.7. Pengujian Fitur Manajemen Pengguna

Pengujian fitur manajemen pengguna dilakukan untuk memastikan pengelolaan akun dan hak akses berjalan dengan baik. Pengujian meliputi

penambahan pengguna, perubahan peran, serta pembatasan akses sesuai dengan *role* yang dimiliki.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Tambah pengguna baru	Nama, Email, Role	Akun pengguna berhasil dibuat	✓
1.2	Update role pengguna	Kasir → Admin	Hak akses berubah sesuai role	✓
1.3	Email duplikat	Email sudah terdaftar	Sistem menolak input	✓
1.4	Nonaktifkan pengguna	Status: Aktif → Nonaktif	Pengguna tidak dapat login	✓

5.2.2.8. Pengujian Fitur Log Aktivitas

Pengujian fitur log aktivitas bertujuan memastikan sistem mencatat setiap aktivitas penting pengguna. Pengujian difokuskan pada pencatatan aktivitas otomatis, penyaringan log berdasarkan kriteria tertentu, serta pembatasan akses log hanya untuk pengguna berwenang.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Log tambah barang	Tambah barang baru	Log aktivitas tercatat	✓
1.2	Log update stok	Update stok barang	Log perubahan tersimpan	✓
1.3	Filter log berdasarkan user	User Admin	Log sesuai user ditampilkan	✓
1.4	Non-admin akses log	User Petugas	Akses ditolak	✓

5.2.2.9. Pengujian Fitur Backup Dan Restore

Pengujian fitur *backup* dan *restore* dilakukan untuk memastikan sistem mampu menjaga keamanan data inventaris. Pengujian difokuskan pada proses pencadangan data, pemulihan data dari *file* cadangan, serta pembatasan akses fitur hanya untuk pengguna berwenang.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Backup manual	Klik menu <i>backup</i>	File <i>backup</i> berhasil dibuat	✓
1.2	Backup dengan timestamp	Generate otomatis	File tersimpan dengan nama unik	✓
1.3	Restore data	File <i>backup</i> dipilih	Data kembali sesuai <i>backup</i>	✓
1.4	Akses non-admin	User petugas	Akses <i>backup</i> ditolak	✓

5.2.2.10. Pengujian Fitur Dashboard Laporan

Pengujian dashboard dan laporan bertujuan memastikan sistem mampu menyajikan informasi inventaris secara ringkas dan akurat. Pengujian dilakukan terhadap tampilan statistik, filter laporan, serta proses ekspor data ke format dokumen.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Akses dashboard admin	Login admin	Statistik inventaris tampil	✓
1.2	Laporan stok barang	Filter semua kategori	Data stok ditampilkan	✓
1.3	Laporan transaksi bulanan	Bulan tertentu	Laporan sesuai periode	✓
1.4	Export laporan	PDF / Excel	File laporan berhasil diunduh	✓

5.2.2.11. Pengujian Fitur *Approval* Transaksi

Pengujian fitur *approval* transaksi dilakukan untuk memastikan proses persetujuan berjalan sesuai alur yang ditetapkan. Pengujian mencakup pemberian persetujuan, penolakan transaksi, serta pembatasan hak *approval* berdasarkan peran pengguna.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Lihat transaksi pending	User kepala sekolah	Daftar transaksi pending tampil	✓
1.2	Approve transaksi	Klik approve	Status berubah menjadi approved	✓
1.3	Reject transaksi	Klik reject	Status berubah menjadi rejected	✓
1.4	Non-authorized approve	User petugas	Tombol approve tidak tersedia	✓

5.2.2.12. Pengujian Fitur Pencarian Dan Filter

Pengujian fitur pencarian dan filter bertujuan memastikan sistem memudahkan pengguna dalam menemukan data. Pengujian difokuskan pada pencarian berdasarkan kata kunci, penyaringan data berdasarkan kategori dan tanggal, serta penanganan hasil pencarian kosong.

No	Skenario Uji	Input	Output yang Diharapkan	Hasil
1.1	Pencarian data barang	Kata kunci nama barang	Data sesuai kata kunci tampil	✓
1.2	Filter berdasarkan kategori	Pilih kategori tertentu	Data terfilter sesuai kategori	✓
1.3	Filter berdasarkan tanggal	Rentang tanggal	Data sesuai periode tampil	✓
1.4	Pencarian tanpa hasil	Kata kunci tidak ada	Pesan “data tidak ditemukan”	✓

BAB IV

PENUTUP

6.1 Simpulan

Berdasarkan hasil analisis, perancangan, implementasi, dan pengujian yang telah dilakukan pada Sistem Informasi Manajemen Inventaris Barang berbasis web di Taman Kanak-Kanak Teratai Cirebon, dapat disimpulkan beberapa hal sebagai berikut:

1. Sistem informasi manajemen inventaris barang berhasil dirancang dan diimplementasikan sesuai dengan kebutuhan pengguna, khususnya dalam membantu proses pencatatan barang masuk, barang keluar, pemantauan stok, serta pengelolaan data inventaris secara terstruktur dan terkomputerisasi.
2. Sistem yang dibangun mampu meningkatkan efisiensi dan efektivitas pengelolaan inventaris dibandingkan dengan metode manual yang sebelumnya digunakan. Hal ini ditunjukkan dengan kemudahan akses data, kecepatan dalam pencarian informasi, serta pengurangan risiko kesalahan pencatatan.
3. Penerapan sistem berbasis web memungkinkan akses data secara *real-time* oleh pihak yang berwenang, seperti admin, petugas inventaris, dan kepala sekolah, sehingga mendukung proses pengambilan keputusan yang lebih cepat dan akurat.
4. Fitur-fitur yang tersedia, seperti pengelolaan data barang, pencatatan barang rusak, *monitoring* stok, pembuatan laporan, serta pencatatan aktivitas pengguna, telah berjalan sesuai dengan kebutuhan yang dirancang dan membantu meningkatkan transparansi pengelolaan inventaris.

5. Berdasarkan hasil pengujian menggunakan metode *black-box*, seluruh fungsi sistem dapat berjalan dengan baik sesuai dengan skenario yang telah direncanakan, sehingga sistem dinyatakan layak untuk digunakan dalam lingkungan operasional Taman Kanak-Kanak Teratai Cirebon.

6.2 Saran

Berdasarkan hasil pengembangan dan keterbatasan yang ditemukan selama proses perancangan dan implementasi sistem, terdapat beberapa saran yang dapat dijadikan pertimbangan untuk pengembangan selanjutnya, yaitu:

1. Sistem dapat dikembangkan lebih lanjut dengan menambahkan fitur notifikasi otomatis, seperti pemberitahuan stok menipis atau pengingat pemeliharaan barang, agar pengelolaan inventaris menjadi lebih proaktif.
2. Diperlukan peningkatan pada aspek keamanan sistem, seperti penerapan enkripsi data yang lebih kuat, autentikasi dua faktor (*two-factor authentication*), serta pengelolaan hak akses pengguna yang lebih detail.
3. Sistem dapat dikembangkan agar mendukung akses melalui perangkat *mobile*, sehingga pengguna dapat melakukan *monitoring* dan pengelolaan inventaris secara lebih fleksibel.
4. Diperlukan pelatihan berkala bagi pengguna sistem, khususnya petugas inventaris dan admin, agar pemanfaatan sistem dapat berjalan optimal dan meminimalkan kesalahan operasional.
5. Pengembangan selanjutnya dapat menambahkan fitur analisis dan visualisasi data, seperti grafik penggunaan barang dan laporan statistik, guna membantu pihak manajemen dalam pengambilan keputusan strategis.

Dengan adanya pengembangan lanjutan tersebut, diharapkan sistem informasi manajemen inventaris barang dapat terus berkembang, memiliki performa yang

lebih baik, serta memberikan manfaat yang maksimal bagi Taman Kanak-Kanak Teratai Cirebon.

DAFTAR PUSTAKA

- Arifianto, Endra Y., Anas, M., Galisong, R. D., Hulukati, S. A., Sholilah, Q. (2023). K3 Kegawatdaruratan Perguruan Tinggi. Yogyakarta: CV BINTANG SEMESTA MEDIA.
- Dwi Putri, R., Andryani, R., Studi Sistem Informasi, P., Ilmu Komputer, F., Bina Darma, U., Jenderal Yani No, J. A., & Kode Pos, P. (2022). Rancang Bangun Sistem Informasi Inventaris Barang Pada Smp Negeri 01 Runjung Agung Berbasis Website. *Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika*.
- Indarwati, I., et al. (2023). *Pendidikan anak usia dini*. PT Sada Kurnia Pustaka.
- Irfan, M., Siregar, H., & Handoko, J. T. (2023). Pengembangan Dan Integrasi Aplikasi Prediksi Jumlah Gagal Produksi PC Meggunakan Metode Triple Exponential Smoothing Pada Sistem Aplikasi Produksi Di PT Tera Data Indonusa,Tbk. *Institut Informatika dan Bisnis Darmajaya*, 80–96.
- Kartono, F. K., Nursaadah, S., Nugroho, M. R., Tama, D. A., Mashudi, F. A., Wicaksono, A., & Nasir, M. (2024). Pengujian Black Box testing pada sistem website menggunakan pendekatan boundary value analysis. *Jurnal Kridatama Sains dan Teknologi*, 6(2), 754–766.
- Kus Indrani Listyoningrum, Danise Yunaini Fenida, & Nurhasan Hamidi. (2023). Inovasi Berkelanjutan dalam Bisnis: Manfaatkan Flowchart untuk Mengoptimalkan Nilai Limbah Perusahaan. *Jurnal Informasi Pengabdian Masyarakat*, 1(4), 100–112. <https://doi.org/10.47861/jipm-nalanda.v1i4.552>
- Lubis, A. S. (2024). Pengujian aplikasi berbasis web menggunakan metode Black Box Testing. *Journal Aira*, 3(1), 45–52.
- Mukhlis, I. R., et al. (2024). *Sistem informasi: Teori dan implementasi sistem informasi di berbagai bidang*. Sonpedia Publishing.
- Murod, A., Hadiwiyanti, R., & Kartika, D. S. Y. (2024). Rancang Bangun Sistem Informasi Manajemen Persediaan Barang Menggunakan Framework Laravel (Studi Kasus : Pt. Jazeera Inti Sukses). *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(3). <https://doi.org/10.23960/jitet.v12i3.4706>
- Pratama, S. D., Lasimin, L., & Dadaprawira, M. N. (2023). Pengujian Black Box Testing pada aplikasi berbasis website menggunakan metode equivalence partitioning. *Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD*, 6(2), 560–569.
- Riza, N. P., Damayanti, A., Harsemadi, G., Nym, N., & Januhari, U. (2024). Rancang Bangun Sistem Informasi Pengelolaan Inventaris Barang Pada

Kantor Perbekel Desa Kediri Tabanan. Jurnal Informatika dan Teknik Elektro Terapan.

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>

Triana, I., Nugroho, A., & Meisak, D. (2024). *Jurnal Informatika Dan Rekayasa Komputer (JAKAKOM) Perancangan Sistem Informasi Persediaan Barang Berbasis Web Pada Toko Kuat*. *Jurnal Informatika Dan Rekayasa Komputer*, 4(1). <https://doi.org/10.33998/jakakom.v4i1>

Wahyuni, E. I., et al. (2024). Analisis perancangan sistem informasi pendaftaran siswa baru TK Putiek Nanggroe berbasis web menggunakan UML. *Jurnal Teknik Industri, Universitas Malikussaleh*.

Wahyuni, E. I., Gani, S. A., Aryanto, H., Siregar, A. K., & Aini, Q. (2024). Analisis perancangan sistem informasi pendaftaran siswa baru TK Putiek Nanggroe berbasis web menggunakan Unified Modeling Language. *Jurnal Teknik Industri, Universitas Malikussaleh*, 855–86

LAMPIRAN

1. Sintaks *Backup* dan *Restore* Database

```
public static function table(Table $table): Table
{
    return $table->columns([
        Tables\Columns\TextColumn::make('filename')
            ->label('Nama File')
            ->searchable()
            ->sortable(),

        Tables\Columns\TextColumn::make('file_size')
            ->label('Ukuran')
            ->formatStateUsing(fn($state) => $state ?
                number_format($state / 1024, 2) . ' KB' : '-')
            ->sortable(),

        Tables\Columns\TextColumn::make('format')
            ->label('Format')
            ->badge()
            ->colors([
                'info' => 'sql',
                'warning' => 'json',
            ]),
    ]),

    Tables\Columns\BadgeColumn::make('status')
        ->label('Status')
        ->colors([
            'success' => 'success',
            'danger' => 'failed',
            'warning' => 'pending',
        ])
        ->formatStateUsing(fn($state) =>
ucfirst($state))
        ->sortable(),

    Tables\Columns\TextColumn::make('created_at'
        )
            ->label('Tanggal')
            ->dateTime('d M Y, H:i')
            ->sortable(),

    Tables\Columns\TextColumn::make('user.name')
        ->label('Dibuat Oleh')
        ->searchable(),
    ])
    ->filters([])
    ->actions([
        Tables\Actions\Action::make('download')
            ->label('Download')
            ->icon('heroicon-o-arrow-down-tray')
            ->action(fn($record) => response() -
>download(
                storage_path('backups/' . $record-
>filename),
            )
        )
    ])
}
```

```

        $record->filename
    ))
    ->visible(fn($record) => $record->status
===== 'success' && file_exists(storage_path('backups/' .
$record->filename))),
    Tables\Actions\DeleteAction::make(),
])
->bulkActions([
    Tables\Actions\BulkActionGroup::make([
        Tables\Actions\DeleteBulkAction::make(),
    ]),
])
->defaultSort('created_at', 'desc');
}
```
Implementasi Header Actions (Buat Backup & Restore):
```
```php
protected function getHeaderActions(): array
{
 return [
 Action::make('createBackup')
 ->label('Buat Backup Baru')
 ->icon('heroicon-o-plus')
 ->color('success')
 ->action(fn () => $this->createBackup())
 ->requiresConfirmation()
 ->modalHeading('Buat Backup Baru')
 ->modalDescription('Ini akan membuat
backup database baru. Proses ini mungkin memerlukan
beberapa menit untuk database besar.')
 ->modalSubmitActionLabel('Ya, Buat
Backup'),
 Action::make('restoreDatabase')
 ->label('Restore Database')
 ->icon('heroicon-o-arrow-path')
 ->color('danger')
 ->visible(fn () => Auth::user() ?-
>hasPermissionTo('restore_system') ?? false)
 ->form([
 FileUpload::make('sql_file')
 ->label('Upload File Backup
(.sql)')
 ->maxSize(102400)
 ->required()
 ->rules([
 function () {
 return function (string
$attribute, $value, \Closure $fail) {
```

```

```

$extension =
strtolower(pathinfo($value->getClientOriginalName(),
PATHINFO_EXTENSION));
if ($extension !==
'sql') {
    $fail('File harus
berformat .sql');
}
},
])
->helperText('Pilih file
backup .sql yang ingin direstore. Proses ini akan menghapus
semua data yang ada!')
->disk('local')
->directory('temp-restores')
->visibility('private')
->storeFiles(false),
])
->action(function (array $data) {
    $uploadedFile = $data['sql_file'];
    if (is_array($uploadedFile)) {
        $uploadedFile = $uploadedFile[0];
    }
    $this-
>restoreFromUpload($uploadedFile);
})
->requiresConfirmation()
->modalHeading('Restore Database dari
File')
->modalDescription('⚠ PERINGATAN: Proses
ini akan menghapus SEMUA data yang ada di database dan
menggantinya dengan data dari file backup. Pastikan Anda
sudah membuat backup terlebih dahulu!')
->modalSubmitActionLabel('Ya, Restore
Database')
->modalIcon('heroicon-o-exclamation-
triangle'),
];
}

public function createBackup(): void
{
    try {
        $backupLog = BackupLog::create([
            'user_id' => Auth::id(),
            'status' => 'pending',
            'format' => 'sql',
        ]);
        \Illuminate\Support\Facades\Artisan::call(
            'db:backup');
        $backupDir = storage_path('backups');
    }
}

```

```

$files      =
\Illuminate\Support\Facades\File::files($backupDir);

if (empty($files)) {
    throw new \Exception('Tidak ada file
backup yang ditemukan.');
}

usort($files, function ($a, $b) {
    return $b->getMTime() - $a->getMTime();
});

$latestFile = $files[0];
$filename   = basename($latestFile-
>getPathname());
$fileSize   = $latestFile->getSize();

$backupLog->update([
    'filename' => $filename,
    'file_size' => $fileSize,
    'status' => 'success',
]);

Notification::make()
    ->title('Backup Berhasil')
    ->body("Database berhasil di-backup:
{$filename}")
    ->success()
    ->send();

$this->loadBackups();
} catch (\Exception $e) {
    if (isset($backupLog)) {
        $backupLog->update([
            'status' => 'failed',
            'notes' => $e->getMessage(),
        ]);
    }

    Notification::make()
        ->title('Backup Gagal')
        ->body($e->getMessage())
        ->danger()
        ->send();
}
}

public function restoreFromUpload($uploadedFile):
void
{
    try {
        if (!Auth::user()->hasPermissionTo('restore_system')) {
            throw new \Exception('Anda tidak memiliki
izin untuk restore database.');
    }
}

```

```

        }

        if (is_string($uploadedFile)) {
            $tempPath = $uploadedFile;
            $originalName = basename($uploadedFile);
        } else {
            $tempPath = $uploadedFile->getRealPath();
            $originalName = $uploadedFile-
>getClientOriginalName();
        }

                if (!
\Illuminate\Support\Facades\File::exists($tempPath)) {
            throw new \Exception('File backup tidak
ditemukan.');
        }

                $fileSize =
\Illuminate\Support\Facades\File::size($tempPath);
        $userId = Auth::id();

                $sql =
\Illuminate\Support\Facades\File::get($tempPath);

        if (empty(trim($sql))) {
            throw new \Exception('File backup kosong
atau tidak valid.');
        }

                \Illuminate\Support\Facades\DB::unprepared(
'SET FOREIGN_KEY_CHECKS=0');
                \Illuminate\Support\Facades\DB::unprepared(
$sql);
                \Illuminate\Support\Facades\DB::unprepared(
'SET FOREIGN_KEY_CHECKS=1');

        BackupLog::create([
            'user_id' => $userId,
            'filename' => $originalName,
            'file_size' => $fileSize,
            'format' => 'sql',
            'status' => 'success',
            'notes' => 'Restore berhasil dari file
upload: '.$originalName,
        ]);

                \Illuminate\Support\Facades\Artisan::call(
'cache:clear');
                \Illuminate\Support\Facades\Artisan::call(
'config:clear');
                \Illuminate\Support\Facades\Artisan::call(
'view:clear');

        Notification::make()
            ->title('Restore Berhasil')
    
```

```
->body('Database berhasil di-restore dari  
file: '.$originalName)  
->success()  
->duration(3000)  
->send();  
  
$this->loadBackups();  
  
$this-  
>redirect(route('filament.admin.pages.backup-manager'),  
navigate: false);  
  
} catch (\Exception $e) {  
    try {  
        \Illuminate\Support\Facades\DB::unprepa-  
red('SET FOREIGN_KEY_CHECKS=1');  
        } catch (\Exception $ignored) {  
    }  
  
    Notification::make()  
        ->title('Restore Gagal')  
        ->body('Error: '.$e->getMessage())  
        ->danger()  
        ->duration(10000)  
        ->send();  
    }  
}
```