

PENGELOLAAN INVENTARIS TERINTEGRASI SISTEM INFORMASI MANAJEMEN INVENTARIS BERBASIS WEB MENGUNAKAN ARSITEKTUR MERN STACK

Maghfirah Fufi Fauziah Zen ^{*1}, Dwi Sartika Simatupang ²

Email: ¹ mffzen7799@gmail.com, ² dwi.sartika@esaunggul.ac.id

^{1,2} Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Esa Unggul

Abstrak

Pengelolaan inventaris merupakan aspek penting dalam mendukung efektivitas operasional dan akuntabilitas organisasi. Namun, masih banyak instansi yang mengelola inventaris secara manual atau semi-terkomputerisasi, sehingga menimbulkan permasalahan seperti ketidaktepatan data stok, keterlambatan pelaporan, lemahnya pengendalian internal, serta keterbatasan akses informasi secara real-time. Penelitian ini bertujuan untuk merancang dan mengimplementasikan Sistem Informasi Manajemen Inventaris berbasis web menggunakan arsitektur MERN Stack (MongoDB, Express.js, React, dan Node.js) sebagai solusi pengelolaan inventaris yang terintegrasi dan andal. Metode penelitian yang digunakan adalah pendekatan rekayasa perangkat lunak dengan tahapan System Development Life Cycle (SDLC), yang meliputi analisis kebutuhan, perancangan sistem, implementasi, dan pengujian. Sistem yang dikembangkan mencakup modul autentikasi dan otorisasi berbasis peran (*Role-Based Access Control*), pengelolaan data master barang, transaksi barang masuk dan keluar, pemantauan stok, pelaporan inventaris, serta audit trail untuk pencatatan aktivitas pengguna. Pengujian sistem dilakukan menggunakan metode Black Box Testing untuk memverifikasi kesesuaian fungsi sistem terhadap kebutuhan yang telah ditetapkan. Hasil pengujian menunjukkan bahwa seluruh modul utama sistem berjalan sesuai dengan spesifikasi fungsional, mampu memproses input valid maupun tidak valid secara tepat, serta menjaga konsistensi dan keandalan data inventaris. Dengan demikian, sistem informasi yang dikembangkan dinilai layak digunakan sebagai solusi digital untuk meningkatkan efisiensi operasional, akurasi data, serta transparansi dan akuntabilitas pengelolaan inventaris dalam organisasi.

Kata kunci: sistem informasi inventaris, aplikasi web, MERN Stack, manajemen inventaris, black box testing.

Abstract

Inventory management is a critical aspect of organisational operations, supporting efficiency, accountability, and informed decision-making. However, many institutions still rely on manual or semi-computerised inventory practices, which often result in inaccurate stock data, reporting, weak internal controls, and limited real-time information. This study aims to design and implement a web-based Inventory Management Information System using the MERN Stack (MongoDB, Express.js, React, and Node.js) as an integrated and reliable solution. The research adopts a software engineering approach based on the System Development Life Cycle (SDLC), comprising requirements analysis, system design, implementation, and testing. The system includes role-based authentication and authorisation (*Role-Based Access Control*), master data management, inbound and outbound inventory transactions, real-time stock monitoring, inventory reporting, and an audit trail for user activities. System testing was conducted using the Black Box Testing to evaluate functional compliance with predefined requirements. The testing results indicate that all core system modules operate as expected, correctly handling both valid and invalid inputs while maintaining data consistency and reliability. The findings demonstrate that the proposed system effectively improves operational efficiency, enhances inventory data accuracy, and strengthens transparency and accountability in inventory processes. Therefore, the developed web-based inventory management system is suitable as a digital solution to support effective and controlled inventory management.

Keywords: inventory management system, web-based application, MERN Stack, Role-Based Access Control, black box testing.

1. Pendahuluan

Pengelolaan inventaris merupakan aktivitas fundamental yang berperan penting dalam mendukung keberlangsungan operasional, efisiensi kerja, dan akuntabilitas organisasi. Inventaris tidak hanya merepresentasikan aset fisik, tetapi juga menjadi sumber data strategis yang digunakan dalam perencanaan, pengendalian, serta pengambilan keputusan manajerial. Ketidaktepatan dalam pengelolaan inventaris dapat menimbulkan berbagai permasalahan, seperti ketidaksesuaian data

stok, kehilangan barang, pemborosan anggaran, dan menurunnya efisiensi operasional organisasi (Romney & Steinbart, 2018). Oleh karena itu, diperlukan sistem informasi yang mampu mengelola inventaris secara terstruktur, akurat, dan mudah diakses.

Dalam praktiknya, banyak organisasi masih mengandalkan metode manual atau semi-terkomputerisasi dalam pengelolaan inventaris, seperti pencatatan berbasis dokumen fisik atau lembar kerja spreadsheet. Pendekatan ini memiliki keterbatasan dalam menjaga konsistensi data, menyediakan histori transaksi

yang lengkap, serta mendukung akses multi-pengguna secara bersamaan. Selain itu, sistem konvensional umumnya belum dilengkapi dengan mekanisme kontrol akses dan pencatatan aktivitas pengguna yang memadai, sehingga berpotensi menimbulkan kesalahan pencatatan dan lemahnya pengendalian internal (Hidayat et al., 2020). Kondisi tersebut menunjukkan perlunya sistem inventaris berbasis teknologi informasi yang terintegrasi dan andal.

Perkembangan teknologi web modern membuka peluang untuk mengembangkan sistem informasi berbasis web yang fleksibel dan skalabel. Salah satu pendekatan yang banyak digunakan adalah arsitektur MERN Stack, yang terdiri atas MongoDB, Express.js, React, dan Node.js. Arsitektur ini memungkinkan pengembangan aplikasi web secara end-to-end menggunakan bahasa JavaScript, serta mendukung pemisahan antara antarmuka pengguna, logika aplikasi, dan pengelolaan basis data. Pemisahan tersebut meningkatkan modularitas dan kemudahan pemeliharaan sistem (Flanagan, 2020). Selain itu, MongoDB sebagai basis data NoSQL memberikan fleksibilitas dalam pengelolaan data inventaris yang bersifat dinamis dan berorientasi dokumen (Banker, 2021).

Berdasarkan kondisi tersebut, permasalahan utama dalam penelitian ini meliputi rendahnya akurasi dan keterpaduan data inventaris, keterbatasan sistem konvensional dalam menyediakan informasi stok secara real-time, serta belum optimalnya penerapan kontrol akses dan audit trail. Oleh karena itu, penelitian ini bertujuan merancang dan mengimplementasikan Sistem Informasi Manajemen Inventaris berbasis web menggunakan arsitektur MERN Stack yang mampu mengelola data master barang, transaksi barang masuk dan keluar, serta menerapkan autentikasi dan kontrol akses berbasis peran. Penelitian ini diharapkan memberikan kontribusi teoretis dalam pengembangan sistem informasi dan manfaat praktis berupa peningkatan efisiensi, akurasi, serta transparansi pengelolaan inventaris di lingkungan organisasi.

2. Tinjauan Pustaka

Sistem informasi manajemen inventaris merupakan bagian integral dari sistem informasi organisasi yang berfungsi mengelola

data barang secara terstruktur, mulai dari pencatatan, pemantauan, hingga pelaporan inventaris. Inventaris dipahami sebagai aset berwujud yang mendukung aktivitas operasional dan harus dikelola secara sistematis agar memberikan nilai guna yang optimal. Sistem informasi inventaris berperan penting dalam menjaga akurasi data, mengurangi risiko kehilangan atau ketidaksesuaian stok, serta mendukung pengambilan keputusan berbasis data (Romney & Steinbart, 2018). Dengan dukungan teknologi informasi, pengelolaan inventaris dapat dilakukan secara lebih efisien, transparan, dan terdokumentasi.

Seiring meningkatnya kebutuhan akan akses informasi yang cepat dan fleksibel, sistem informasi berbasis web menjadi solusi yang banyak diadopsi oleh organisasi. Sistem berbasis web memungkinkan pengguna mengakses aplikasi melalui peramban tanpa ketergantungan pada perangkat atau sistem operasi tertentu. Keunggulan utama pendekatan ini terletak pada kemudahan distribusi, pemeliharaan terpusat, serta dukungan akses multi-pengguna secara simultan. Penelitian sebelumnya menunjukkan bahwa penerapan sistem inventaris berbasis web mampu meningkatkan efisiensi proses administrasi dan memperbaiki konsistensi data dibandingkan sistem manual atau berbasis desktop (Hidayat et al., 2020).

Dalam pengembangan aplikasi web modern, pemilihan arsitektur dan teknologi pengembangan menjadi faktor kunci yang memengaruhi kualitas dan keberlanjutan sistem. Arsitektur MERN Stack merupakan pendekatan yang banyak digunakan karena mengintegrasikan teknologi berbasis JavaScript secara end-to-end. MERN Stack terdiri atas MongoDB sebagai basis data NoSQL, Express.js sebagai kerangka kerja backend, React sebagai pustaka antarmuka pengguna, dan Node.js sebagai lingkungan eksekusi sisi server. Pendekatan ini mendukung pengembangan sistem yang modular, skalabel, dan mudah dikembangkan sesuai kebutuhan organisasi (Flanagan, 2020).

MongoDB sebagai basis data berorientasi dokumen menyimpan data dalam format JSON-like (BSON), sehingga memberikan fleksibilitas tinggi dalam menangani data inventaris yang bersifat dinamis dan terus berkembang. Dibandingkan basis data relasional, MongoDB lebih adaptif terhadap perubahan skema dan sesuai untuk aplikasi

dengan kebutuhan transaksi serta pencatatan histori yang intensif (Banker, 2021). Pada sisi antarmuka pengguna, React mendukung pembuatan tampilan yang interaktif dan responsif melalui konsep komponen, yang meningkatkan keterbacaan kode, kemudahan pemeliharaan, serta pengalaman pengguna. Sementara itu, Node.js dan Express.js menyediakan lingkungan server-side yang ringan dan efisien untuk pengelolaan permintaan klien dan logika bisnis aplikasi. Selain aspek teknis, pengelolaan inventaris memerlukan mekanisme pengendalian internal yang memadai. *Role-Based Access Control* (RBAC) digunakan untuk membatasi hak akses pengguna berdasarkan peran, sehingga mengurangi risiko kesalahan dan penyalahgunaan sistem (Sandhu et al., 1996). Audit trail juga berperan penting dalam mencatat seluruh aktivitas pengguna dan transaksi inventaris, sehingga meningkatkan transparansi, akuntabilitas, dan keandalan sistem informasi (Behl & Behl, 2017). Berdasarkan tinjauan tersebut, sistem informasi manajemen inventaris berbasis web dengan arsitektur MERN Stack, RBAC, dan audit trail merupakan pendekatan yang relevan untuk memenuhi kebutuhan pengelolaan inventaris modern.

3. Metodologi Penelitian

Penelitian ini menggunakan pendekatan rekayasa perangkat lunak dengan tujuan menghasilkan artefak berupa sistem informasi manajemen inventaris berbasis web yang fungsional dan teruji. Pendekatan ini menekankan proses sistematis mulai dari analisis kebutuhan hingga pengujian sistem, sehingga solusi yang dikembangkan tidak hanya berfungsi secara teknis, tetapi juga sesuai dengan kebutuhan pengguna dan konteks operasional organisasi (Pressman & Maxim, 2020). Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *System Development Life Cycle* (SDLC). SDLC dipilih karena menyediakan tahapan pengembangan yang terstruktur dan mudah dikendalikan, serta banyak digunakan dalam penelitian sistem informasi dan pengembangan aplikasi berskala kecil hingga menengah. Tahapan SDLC yang diterapkan dalam penelitian ini meliputi analisis kebutuhan, perancangan sistem, implementasi, dan pengujian (Sommerville, 2016). Pendekatan ini memungkinkan

identifikasi permasalahan secara sistematis dan meminimalkan risiko kegagalan sistem pada tahap implementasi.



Gambar 1. System Development Life Cycle (Pressman & Maxim, 2020)

Tahap analisis kebutuhan bertujuan untuk mengidentifikasi kebutuhan fungsional dan non-fungsional sistem. Kebutuhan fungsional mencakup pengelolaan data master inventaris, transaksi barang masuk dan keluar, pemantauan stok, serta penyajian laporan inventaris. Sementara itu, kebutuhan non-fungsional meliputi aspek keamanan, kemudahan penggunaan, kinerja sistem, dan keandalan aplikasi. Analisis kebutuhan merupakan tahapan krusial karena menjadi dasar dalam menentukan rancangan sistem yang tepat dan sesuai dengan kebutuhan pengguna (Satzinger et al., 2020).

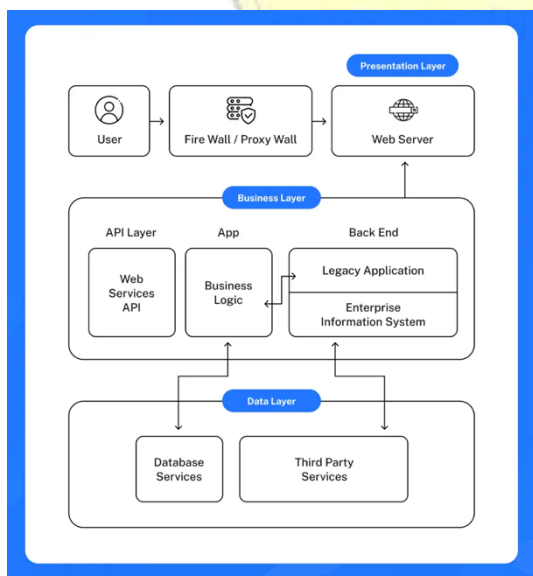
Tahap perancangan sistem dilakukan dengan menerapkan pendekatan arsitektur berlapis (*layered architecture* – gambar 2), yang memisahkan komponen antarmuka pengguna, logika aplikasi, dan pengelolaan basis data. Pendekatan ini bertujuan untuk meningkatkan modularitas, skalabilitas, dan kemudahan pemeliharaan sistem. Pada tahap ini juga dirancang model basis data konseptual, alur proses bisnis, serta mekanisme kontrol akses berbasis peran RBAC untuk mengatur hak akses pengguna sesuai kewenangannya (Bass et al., 2021). Perancangan sistem dilakukan sebagai jembatan antara kebutuhan pengguna dan implementasi teknis.

Tahap implementasi sistem dilakukan dengan mengembangkan aplikasi web (gambar 2.) menggunakan arsitektur MERN Stack. React digunakan untuk membangun antarmuka

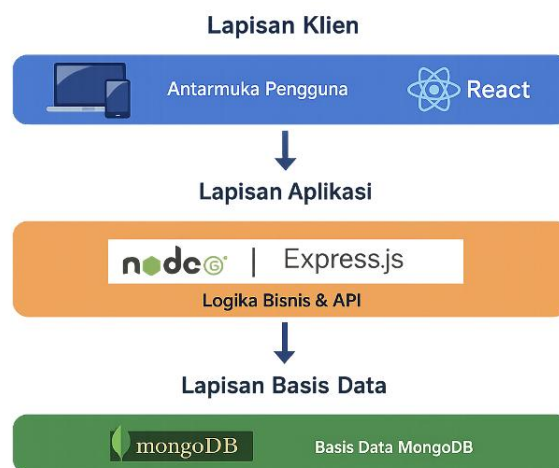
pengguna yang interaktif dan responsif, Node.js dan Express.js digunakan untuk mengelola layanan backend dan API, sedangkan MongoDB digunakan sebagai basis data untuk menyimpan data inventaris dan transaksi. Pemilihan MERN Stack didasarkan pada kemampuannya dalam mendukung pengembangan aplikasi web yang bersifat modular, efisien, dan mudah dikembangkan secara berkelanjutan (Flanagan, 2020; Banker, 2021).

Tahap pengujian sistem bertujuan untuk memastikan bahwa sistem yang dikembangkan telah memenuhi kebutuhan yang ditetapkan. Pengujian dilakukan menggunakan metode black box testing untuk memverifikasi fungsi-fungsi utama sistem tanpa melihat struktur kode internal. Selain itu, dilakukan pengujian hak akses pengguna untuk memastikan mekanisme RBAC berjalan sesuai perancangan, serta validasi transaksi inventaris untuk menjamin akurasi data stok. Pengujian fungsional merupakan pendekatan yang umum digunakan dalam evaluasi sistem informasi berbasis aplikasi karena berfokus pada kesesuaian sistem dengan kebutuhan pengguna (Rosa & Shalahuddin, 2018).

Melalui tahapan metodologi tersebut, penelitian ini berupaya menghasilkan sistem informasi manajemen inventaris yang tidak hanya dapat diimplementasikan secara teknis, tetapi juga memiliki dasar metodologis yang kuat dan dapat dipertanggungjawabkan secara ilmiah.



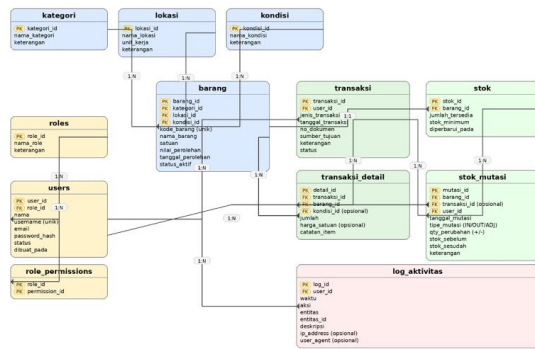
Gambar 2. Layer Arsitektur Aplikasi Web (Flanagan, 2020).



Gambar 3. Diagram Arsitektur Sistem Tiga Lapis (Bass et al., 2021)

Arsitektur sistem yang dikembangkan menerapkan pendekatan arsitektur berlapis (gambar 3.) yang memisahkan sistem ke dalam lapisan klien, lapisan aplikasi, dan lapisan basis data. Lapisan klien berfungsi sebagai antarmuka pengguna berbasis web yang dibangun menggunakan React untuk menangani interaksi pengguna dan penyajian informasi secara responsif. Lapisan aplikasi berperan sebagai pengolah logika bisnis dan layanan aplikasi yang diimplementasikan menggunakan Node.js dan Express.js melalui mekanisme RESTful API, termasuk pengelolaan autentikasi, otorisasi berbasis peran, serta pemrosesan transaksi inventaris. Sementara itu, lapisan basis data menggunakan MongoDB sebagai basis data berorientasi dokumen untuk menyimpan data inventaris, transaksi, dan audit trail secara terpusat. Pemisahan fungsi pada setiap lapisan bertujuan untuk meningkatkan modularitas, skalabilitas, dan kemudahan pemeliharaan sistem, serta mengurangi ketergantungan antar komponen sistem sehingga mendukung pengembangan dan pengelolaan sistem informasi berbasis web yang lebih efisien dan andal (Bass et al., 2021; Flanagan, 2020).

3.1 Entity Relationship Diagram (ERD)



Gambar 4. ERD Model SIMBI

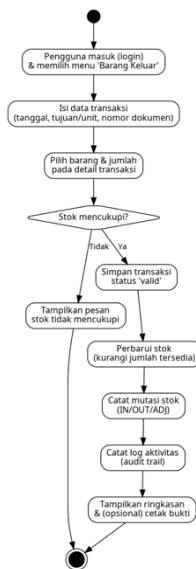
Diagram ERD Sistem Informasi Manajemen Barang Inventaris (SIMBI) memodelkan struktur data terintegrasi yang mencakup entitas master (kategori, lokasi, kondisi, dan barang), entitas pengguna dan hak akses (roles, users, dan role_permissions), serta entitas operasional (transaksi, transaksi_detail, stok, stok_mutasi, dan log_aktivitas) yang saling berelasi secara konsisten untuk mendukung seluruh proses pengelolaan inventaris. Setiap barang diklasifikasikan berdasarkan kategori, lokasi, dan kondisi, kemudian dikelola melalui transaksi barang masuk, keluar, atau penyesuaian yang dicatat pada tabel transaksi dan transaksi_detail, sementara perubahan kuantitas dicerminkan secara real-time pada tabel stok dan direkam secara historis pada stok_mutasi. Seluruh aktivitas pengguna, termasuk perubahan data dan proses transaksi, terekam dalam log_aktivitas untuk menjamin transparansi dan akuntabilitas, sedangkan mekanisme RBAC memastikan setiap pengguna hanya dapat mengakses fungsi sesuai perannya, sehingga keseluruhan skema basis data ini mendukung integritas, keterlacakan, dan keandalan sistem inventaris secara menyeluruh.

3.2 Unified Modelling Language (UML)



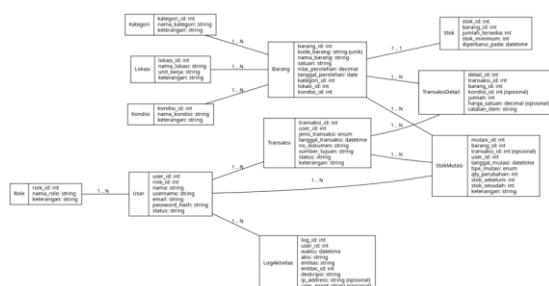
Gambar 5. Use Case Diagram

Diagram Use Case SIMBI menggambarkan interaksi antara tiga aktor utama - admin, operator, dan pimpinan - dengan fungsi-fungsi inti sistem pengelolaan inventaris. Admin memiliki kewenangan penuh untuk mengelola pengguna dan hak akses, mengelola data referensi (kategori, lokasi, dan kondisi), mengelola data barang, memantau stok, menghasilkan dan mengeksport laporan, serta meninjau log aktivitas. Operator berfokus pada proses operasional harian, yaitu mencatat barang masuk, mencatat barang keluar, melakukan penyesuaian stok, mengelola data barang, serta memantau kondisi stok. Pimpinan berperan pada tingkat pengawasan dan pengambilan keputusan dengan mengakses laporan, mengeksport hasil laporan, dan memantau aktivitas sistem. Seluruh aktor harus melalui proses autentikasi (login) sebelum mengakses sistem, sementara fitur ekspor laporan dimodelkan sebagai bagian terintegrasi dari proses pelaporan, sehingga diagram ini menegaskan pembagian peran, alur akses, dan batasan tanggung jawab pengguna dalam sistem secara jelas dan terstruktur.



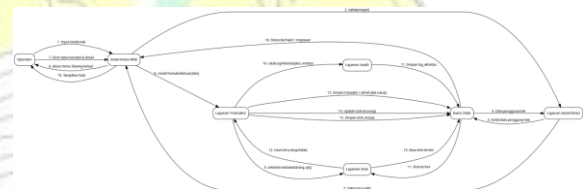
Gambar 6. Activity Diagram

Diagram aktivitas ini memodelkan alur operasional pencatatan barang keluar dalam SIMBI, yang dimulai ketika pengguna berhasil masuk ke sistem dan memilih menu “Barang Keluar”, kemudian mengisi data transaksi berupa tanggal, tujuan atau unit penerima, serta nomor dokumen, dilanjutkan dengan pemilihan barang dan jumlah pada detail transaksi. Sistem selanjutnya melakukan pengecekan ketersediaan stok, di mana apabila stok tidak mencukupi maka sistem menampilkan pesan peringatan dan proses dihentikan, sedangkan jika stok mencukupi maka transaksi disimpan dengan status valid, stok barang diperbarui dengan mengurangi jumlah tersedia, mutasi stok dicatat sebagai riwayat perubahan, serta aktivitas pengguna direkam dalam audit trail. Proses diakhiri dengan penayangan ringkasan transaksi kepada pengguna dan opsi pencetakan bukti, sehingga diagram ini menegaskan kontrol validasi stok, konsistensi data, dan keterlacakan aktivitas dalam setiap transaksi barang keluar.



Gambar 7. Class Diagram

Diagram kelas SIMBI merepresentasikan struktur objek sistem yang memodelkan entitas utama seperti Barang, Transaksi, Stok, dan User beserta relasi antar kelas yang mendukung proses pengelolaan inventaris secara terintegrasi. Kelas Barang berasosiasi dengan Kategori, Lokasi, dan Kondisi sebagai referensi klasifikasi, serta terhubung dengan Stok untuk merepresentasikan jumlah persediaan terkini dan dengan TransaksiDetail serta StokMutasi untuk mencatat pergerakan barang. Kelas Transaksi berfungsi sebagai induk pencatatan aktivitas operasional yang ditautkan dengan pengguna serta detail transaksi, sedangkan kelas User, Role, dan LogAktivitas mengatur autentikasi, otorisasi, serta keterlacakan aktivitas pengguna. Secara keseluruhan, diagram ini menunjukkan pemisahan tanggung jawab antar kelas sekaligus keterkaitan struktural yang memastikan integritas data, konsistensi proses bisnis, dan dukungan terhadap auditabilitas sistem.



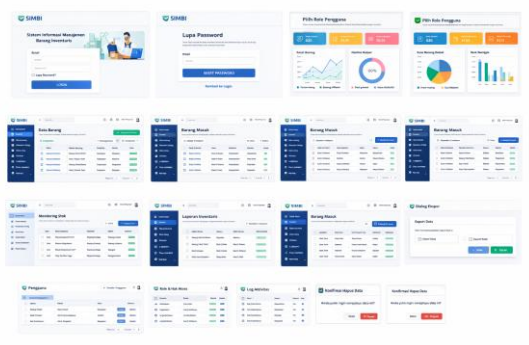
Gambar 8. Sequence Diagram

Diagram sekuens ini menggambarkan interaksi berurutan antara operator, antarmuka web, layanan autentikasi, layanan transaksi, layanan stok, layanan audit, dan basis data dalam proses pencatatan barang keluar. Proses dimulai saat operator memasukkan kredensial untuk divalidasi, setelah itu sistem memberikan akses ke menu barang keluar dan operator mengirimkan data transaksi beserta detail barang. Layanan transaksi kemudian memeriksa ketersediaan stok melalui layanan stok yang membaca data stok terkini dari basis data, dan apabila stok mencukupi maka transaksi beserta detailnya disimpan, stok diperbarui, mutasi stok dicatat, serta aktivitas pengguna direkam oleh layanan audit. Terakhir, sistem mengembalikan status keberhasilan dan ringkasan transaksi kepada operator, sehingga diagram ini menegaskan keterpaduan antar layanan, alur kontrol proses, dan konsistensi data dalam setiap transaksi barang keluar.

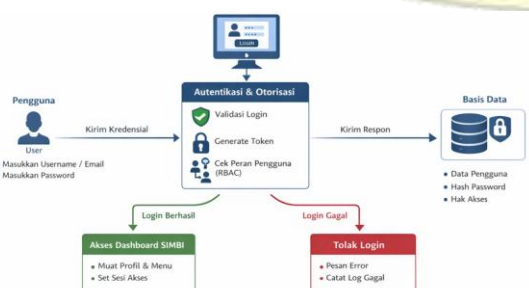
4. Hasil dan Implementasi Database Sistem (Mockup)

4.1 Mockup SIMBI dan Pembahasan

Mockup SIMBI (gambar 9.) dirancang sebagai aplikasi web terintegrasi yang mendukung seluruh siklus pengelolaan inventaris mulai dari autentikasi pengguna, pengelolaan data master, pencatatan transaksi barang masuk dan keluar, penyesuaian stok, hingga pelaporan dan audit aktivitas secara terpusat. Antarmuka disusun secara modular dengan navigasi berbasis peran (admin, operator, dan pimpinan) sehingga setiap pengguna memperoleh akses sesuai kewenangannya, sementara *dashboard* menyajikan ringkasan kondisi inventaris secara visual dan real-time. Setiap proses operasional, seperti input transaksi dan pembaruan stok, dilengkapi validasi serta jejak audit untuk menjamin akurasi, transparansi, dan akuntabilitas data, sedangkan fitur monitoring stok minimum, notifikasi, serta ekspor laporan ke format standar memastikan sistem tidak hanya mendukung kebutuhan operasional harian tetapi juga pengambilan keputusan manajerial dan pengendalian internal.



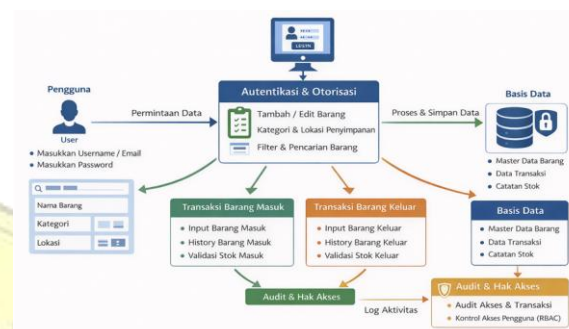
Gambar 9. Mockup SIMBI



Gambar 10. Modul login & Dashboard

Diagram modul login SIMBI (gambar 10.) menggambarkan alur autentikasi pengguna

yang dimulai dari pengisian kredensial, proses validasi data pengguna, dan pengecekan hak akses berbasis peran. Jika autentikasi berhasil, sistem memberikan akses ke dashboard, sedangkan kegagalan login akan ditolak dan dicatat dalam log aktivitas sistem.



Gambar 11. Modul Data Barang & Transaksi

Modul data barang dan transaksi pada SIMBI berfungsi mengelola data master barang, kategori, dan lokasi penyimpanan serta mencatat transaksi barang masuk dan barang keluar. Modul ini memastikan validasi stok, pencatatan histori transaksi, dan pengendalian akses pengguna secara terintegrasi.

4.2 Pengujian Analisis Metode Pengujian Black Box Testing

Pengujian sistem pada penelitian ini dilakukan menggunakan metode Black Box Testing, yaitu pendekatan pengujian perangkat lunak yang berfokus pada evaluasi fungsionalitas sistem berdasarkan spesifikasi kebutuhan tanpa memperhatikan struktur internal kode program. Metode ini dipilih karena sesuai untuk menilai kesesuaian antara kebutuhan pengguna dengan perilaku sistem yang dihasilkan, khususnya pada sistem informasi manajemen inventaris berbasis web seperti SIMBI.

Pada modul login (gambar 10.), pengujian Black Box dilakukan dengan menguji berbagai skenario input kredensial pengguna, seperti kombinasi username/email dan kata sandi yang valid maupun tidak valid. Pengujian ini bertujuan untuk memastikan bahwa sistem mampu memvalidasi data pengguna secara benar, memberikan respons yang sesuai, serta menerapkan mekanisme autentikasi dan otorisasi berbasis peran. Hasil pengujian menunjukkan bahwa sistem dapat mengizinkan akses pengguna yang memiliki kredensial valid dan menolak akses pengguna dengan kredensial

tidak sesuai, disertai pesan kesalahan yang informatif serta pencatatan log kegagalan login sebagai bagian dari audit trail.

Pada modul data barang (gambar 11.), Black Box Testing difokuskan pada pengujian fungsi tambah, ubah, hapus, dan pencarian data barang. Pengujian dilakukan dengan memberikan input data sesuai dan tidak sesuai dengan aturan sistem, seperti data kosong atau format yang tidak valid. Hasil pengujian menunjukkan bahwa sistem mampu melakukan validasi input dengan baik, menyimpan data master barang secara benar ke dalam basis data, serta menampilkan data yang telah diperbarui secara real-time pada antarmuka pengguna.

Selanjutnya, pada modul transaksi barang masuk dan barang keluar (gambar 11.), pengujian dilakukan untuk memastikan bahwa proses pencatatan transaksi berjalan sesuai dengan aturan bisnis yang ditetapkan. Pengujian mencakup skenario input jumlah barang, validasi ketersediaan stok, serta pencatatan histori transaksi. Hasil pengujian menunjukkan bahwa sistem mampu menolak transaksi barang keluar apabila stok tidak mencukupi dan secara otomatis memperbarui

catatan stok serta histori transaksi apabila transaksi berhasil dilakukan.

Selain itu, pengujian Black Box juga diterapkan pada modul dashboard dan laporan (gambar 9. & 10.), dengan memverifikasi kesesuaian data ringkasan, grafik, dan laporan inventaris yang ditampilkan dengan data yang tersimpan dalam basis data. Pengujian ini memastikan bahwa informasi yang disajikan pada dashboard mencerminkan kondisi inventaris terkini dan dapat digunakan sebagai dasar pengambilan keputusan.

Berdasarkan hasil pengujian Black Box Testing pada seluruh modul utama, dapat disimpulkan bahwa SIMBI telah memenuhi kebutuhan fungsional yang ditetapkan. Sistem mampu merespons setiap input pengguna secara tepat, menjaga konsistensi data, serta mendukung proses pengelolaan inventaris secara terintegrasi dan andal. Dengan demikian, metode Black Box Testing terbukti efektif dalam memvalidasi kinerja fungsional sistem informasi manajemen inventaris berbasis web yang dikembangkan.

Tabel 1. Hasil Pengujian Black Box SIMBI

No	Modul	Skenario Pengujian	Data Uji (Input)	Hasil yang Diharapkan	Hasil & Status
1	Login	Login dengan kredensial valid	Username & password valid	Sistem menerima login dan menampilkan dashboard sesuai peran	Sesuai & Valid
2	Login	Login dengan password salah	Username valid, password salah	Sistem menolak login dan menampilkan pesan kesalahan	Sesuai & Valid
3	Login	Login dengan data kosong	Field username & password kosong	Sistem menolak login dan menampilkan peringatan	Sesuai & Valid
4	Login	Hak akses pengguna	User dengan peran tertentu	Menu dan fitur sesuai peran pengguna	Sesuai & Valid
5	Data Barang	Tambah data barang	Data barang lengkap dan valid	Data barang tersimpan dalam basis data	Sesuai & Valid
6	Data Barang	Tambah data barang tidak lengkap	Field wajib dikosongkan	Sistem menolak penyimpanan data	Sesuai & Valid
7	Data Barang	Ubah data barang	Perubahan nama/kategori barang	Data barang berhasil diperbarui	Sesuai & Valid
8	Data Barang	Hapus data barang	Pilih data barang tertentu	Data barang terhapus dari sistem	Sesuai & Valid
9	Data Barang	Pencarian data barang	Kata kunci nama/kategori	Sistem menampilkan data sesuai pencarian	Sesuai & Valid
10	Barang Masuk	Input transaksi barang masuk	Jumlah barang valid	Stok bertambah dan histori tercatat	Sesuai & Valid
11	Barang Masuk	Input jumlah tidak valid	Jumlah nol/negatif	Sistem menolak transaksi	Sesuai & Valid
12	Barang Keluar	Input transaksi barang keluar	Jumlah \leq stok tersedia	Stok berkurang dan histori tercatat	Sesuai & Valid
13	Barang Keluar	Stok tidak mencukupi	Jumlah $>$ stok tersedia	Sistem menolak transaksi	Sesuai & Valid
14	Dashboard	Tampilkan ringkasan data	Data inventaris tersedia	Informasi stok dan transaksi tampil benar	Sesuai & Valid
15	Laporan	Generate laporan inventaris	Periode laporan dipilih	Laporan tampil sesuai periode	Sesuai & Valid
16	Ekspor Data	Ekspor laporan	Pilih format PDF/Excel	File laporan berhasil diunduh	Sesuai & Valid
17	Audit Trail	Pencatatan aktivitas	Login & transaksi dilakukan	Log aktivitas tercatat otomatis	Sesuai & Valid

Tabel 1., hasil pengujian Black Box menunjukkan bahwa seluruh fungsi utama

SIMBI, mulai dari modul login, pengelolaan data barang, transaksi inventaris, hingga pelaporan dan audit trail, berjalan sesuai dengan kebutuhan fungsional yang ditetapkan. Sistem mampu menangani input valid maupun tidak valid secara tepat, sehingga dinilai andal dan layak digunakan dalam mendukung pengelolaan inventaris berbasis web.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Penelitian ini berhasil merancang dan mengimplementasikan Sistem Informasi Manajemen Inventaris Berbasis Web menggunakan arsitektur MERN Stack sebagai solusi pengelolaan inventaris yang terintegrasi. Sistem yang dikembangkan mampu mengelola data master barang, transaksi barang masuk dan keluar, pemantauan stok, serta pelaporan inventaris secara real-time dan terpusat. Penerapan arsitektur berlapis yang memisahkan lapisan klien, aplikasi, dan basis data terbukti meningkatkan modularitas, kemudahan pemeliharaan, serta skalabilitas sistem. Implementasi mekanisme autentikasi dan otorisasi berbasis peran (RBAC) serta audit trail memungkinkan pengendalian akses pengguna dan keterlacakan aktivitas sistem secara menyeluruh, sehingga mendukung transparansi dan akuntabilitas pengelolaan inventaris. Hasil pengujian menggunakan metode Black Box Testing menunjukkan bahwa seluruh fungsi utama sistem berjalan sesuai dengan kebutuhan fungsional yang telah ditetapkan. Sistem mampu memproses input valid dan menolak input tidak valid secara tepat, menjaga konsistensi data, serta mendukung proses operasional inventaris secara andal. Dengan demikian, SIMBI yang dikembangkan dinilai layak digunakan sebagai solusi digital untuk meningkatkan efisiensi, akurasi, dan pengendalian internal dalam pengelolaan inventaris organisasi.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, pengembangan sistem selanjutnya dapat diarahkan pada beberapa aspek. Pertama, integrasi teknologi pendukung seperti barcode atau RFID dapat dipertimbangkan untuk meningkatkan kecepatan dan akurasi proses pencatatan barang. Kedua, penambahan fitur

analitik dan visualisasi data inventaris yang lebih lanjut, seperti prediksi kebutuhan stok atau tren penggunaan barang, dapat membantu pengambilan keputusan manajerial. Ketiga, pengujian sistem pada skala organisasi yang lebih besar serta evaluasi kinerja non-fungsional, seperti performa dan keamanan, perlu dilakukan untuk memastikan sistem tetap optimal dalam kondisi beban tinggi. Selain itu, pengembangan aplikasi versi mobile dapat menjadi alternatif untuk meningkatkan fleksibilitas akses pengguna terhadap sistem.

Daftar Pustaka

- Banker, K. (2021). *MongoDB in action* (2nd ed.). Manning Publications. <https://www.manning.com>
- Bass, L., Clements, P., & Kazman, R. (2021). *Software architecture in practice* (4th ed.). Addison-Wesley.
- Behl, A., & Behl, K. (2017). *Cyberwar: The next threat to national security and what to do about it*. Oxford University Press.
- Flanagan, D. (2020). *JavaScript: The definitive guide* (7th ed.). O'Reilly Media.
- Hidayat, R., Nugroho, A., & Prasetyo, E. (2020). Sistem informasi inventaris barang berbasis web pada instansi pemerintah daerah. *Jurnal Teknologi Informasi dan Ilmu Komputer*, 7(3), 523–530. <https://doi.org/10.25126/jtiik.202073210>
- Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.
- Romney, M. B., & Steinbart, P. J. (2018). *Accounting information systems* (14th ed.). Pearson Education.
- Rosa, A. S., & Shalahuddin, M. (2018). *Rekayasa perangkat lunak terstruktur dan berorientasi objek*. Informatika Bandung.
- Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). Role-based access control models. *IEEE Computer*, 29(2), 38–47. <https://doi.org/10.1109/2.485845>
- Satzinger, J. W., Jackson, R. B., & Burd, S. D. (2020). *Systems analysis and design in a changing world* (7th ed.). Cengage Learning.
- Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson Education.