

PERANCANGAN DAN IMPLEMENTASI SISTEM PERSEDIAAN BARANG BERBASIS WEB MENGGUNAKAN METODE FIFO DAN *REORDER POINT* PADA TOKO RAJA AKI

Muhammad Alief Adhitya Pratama; Khanun Roisatul Ummah
Program Studi Teknik Informatika, Fakultas Komunikasi dan Informatika
Universitas Muhammadiyah Surakarta

Abstrak

Pencatatan stok barang yang masih dilakukan secara manual kerap menimbulkan berbagai persoalan, mulai dari data persediaan yang tidak akurat, kekeliruan dalam mencatat transaksi, hingga proses pelaporan yang memakan waktu lama. Kondisi serupa dijumpai pada Toko Raja Aki, yang hingga saat ini masih mengandalkan buku catatan sebagai sarana pendataan barang. Berangkat dari permasalahan tersebut, penelitian ini dilakukan untuk merancang dan membangun sebuah sistem manajemen persediaan berbasis web guna mendukung pengelolaan stok yang lebih efektif, dengan menerapkan metode First In First Out (FIFO) dan Reorder Point (ROP). Pengembangan sistem dilakukan menggunakan pendekatan Waterfall, yang mencakup tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, hingga pemeliharaan. Dari sisi teknologi, sistem ini dibangun dengan framework Next.js, memanfaatkan PostgreSQL yang di-hosting pada layanan Neon Database, dan di-deploy menggunakan platform Vercel. Penerapan metode FIFO berfungsi untuk mengatur urutan keluarnya barang berdasarkan waktu kedatangannya, sementara metode ROP digunakan sebagai dasar pemicu notifikasi otomatis ketika jumlah stok telah mencapai ambang batas minimum. Berdasarkan hasil implementasi, sistem yang dikembangkan terbukti mampu mendukung proses pengelolaan persediaan agar berjalan lebih terstruktur dan efisien. Dari hasil pengujian Blackbox, diperoleh skor rata-rata sebesar 3,67 yang tergolong dalam kategori Sangat Baik. Sementara itu, evaluasi menggunakan System Usability Scale (SUS) yang melibatkan 30 responden menghasilkan skor 85, yang berada pada kategori Acceptable dengan predikat Grade A serta tingkat penilaian Excellent. Temuan ini mengindikasikan bahwa sistem yang dibangun memiliki tingkat kemudahan penggunaan dan pemahaman yang tinggi, serta mampu menjawab kebutuhan pengguna dalam proses pengelolaan persediaan barang.

Kata kunci: FIFO, Next.js, Reorder Point, Sistem Persediaan Barang, Waterfall.

Abstract

Manual stock recording often leads to various issues, ranging from inaccurate inventory data and transaction recording errors to time-consuming reporting processes. Toko Raja Aki faces similar challenges, as it currently relies on logbooks for inventory tracking. Addressing these issues, this study aims to design and develop a web-based inventory management system to facilitate more effective stock management by implementing the First-In-First-Out (FIFO) and Reorder Point (ROP) methods. The system was developed using the Waterfall approach, encompassing requirements analysis, system design, implementation, testing, and maintenance. Technically, the system was built using the Next.js

framework, utilizes PostgreSQL hosted on Neon Database, and is deployed via the Vercel platform. The FIFO method manages the sequence of stock outflows based on arrival times, while the ROP method triggers automatic notifications when stock levels reach a minimum threshold. Implementation results demonstrate that the developed system effectively supports a more structured and efficient inventory management process. Black-box testing yielded an average score of 3.67, placing it in the "Very Good" category. Meanwhile, a System Usability Scale (SUS) evaluation involving 30 respondents resulted in a score of 85, falling into the "Acceptable" category with a "Grade A" rating and an "Excellent" assessment level. These findings indicate that the system is highly usable and intuitive, effectively meeting user needs for inventory management.

Keywords: *FIFO, Inventory Management System, Next.js, Reorder Point, Waterfall.*

1. PENDAHULUAN

Toko Raja Aki merupakan usaha ritel yang berlokasi di Kota Bontang dan bergerak di bidang penjualan akumulator (aki) kendaraan. Aktivitas operasional toko mencakup penerimaan barang dari pemasok, transaksi penjualan kepada pelanggan, serta pengelolaan berbagai jenis dan merek aki. Tingginya aktivitas tersebut menuntut adanya sistem pengelolaan persediaan yang akurat dan efisien. Namun, proses administrasi di Toko Raja Aki masih dilakukan secara manual menggunakan buku besar sehingga pengelolaan data persediaan belum berjalan secara optimal.

Penggunaan sistem pencatatan konvensional menimbulkan berbagai permasalahan, antara lain ketidaksesuaian jumlah stok, kesulitan menelusuri riwayat transaksi, serta lambatnya penyusunan laporan penjualan. Kondisi serupa juga banyak ditemukan pada usaha skala kecil. Gultom dan Maryam (2020) serta Setyawan dan Maryam (2021) menyatakan bahwa pencatatan berbasis kertas memiliki risiko tinggi terhadap *human error* dan kurang tertibnya administrasi, yang dapat menyebabkan kerugian material hingga 10–15% (Alam et al., 2023).

Selain permasalahan pencatatan, pengelolaan persediaan aki memerlukan pengaturan alur keluar barang yang tepat agar kualitas produk tetap terjaga. Oleh karena itu, metode **First In First Out (FIFO)** diterapkan untuk memastikan barang yang pertama kali diterima menjadi prioritas dalam proses penjualan. Menurut Ristiandi (2021), penerapan FIFO dapat mengurangi risiko penurunan kualitas produk akibat penyimpanan yang terlalu lama. Pelaksanaan metode tersebut akan lebih efektif apabila didukung oleh sistem berbasis web karena proses

pemantauan urutan barang dapat dilakukan secara otomatis.

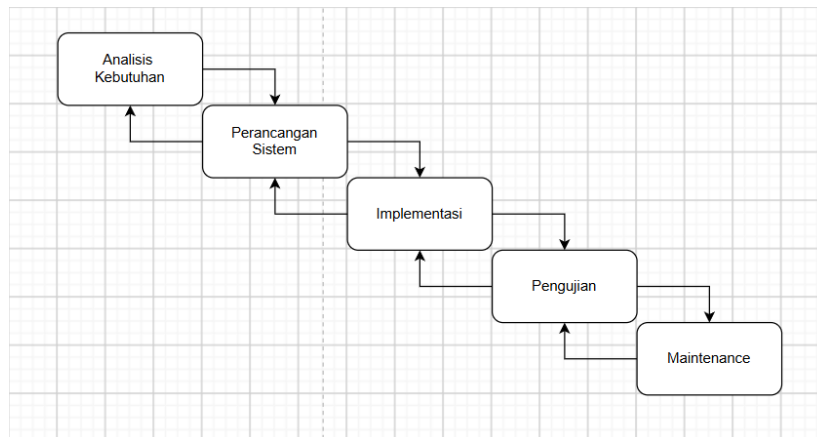
Permasalahan lain yang dihadapi Toko Raja Aki adalah menentukan waktu yang tepat untuk melakukan pemesanan kembali sehingga tidak terjadi kekosongan stok (*stockout*). Salah satu pendekatan yang dapat digunakan adalah metode **Reorder Point (ROP)**. Indrayuni et al. (2023) menjelaskan bahwa penerapan ROP dalam sistem memungkinkan penentuan titik pemesanan kembali dilakukan secara lebih akurat berdasarkan rata-rata penjualan dan *lead time*. Pendapat tersebut didukung oleh Panigrahi et al. (2024), yang menyatakan bahwa pengelolaan persediaan dengan parameter yang jelas mampu menjaga ketersediaan produk sekaligus meningkatkan kepuasan pelanggan.

Perkembangan teknologi juga mendorong pelaku UMKM untuk mulai beralih ke sistem digital. Agboola et al. (2022) mengemukakan bahwa penggunaan platform berbasis web dapat membantu UMKM beradaptasi dengan kebutuhan bisnis modern. Selain menjaga kesesuaian antara stok fisik dan data pada sistem (Azizah & Nurgiyatna, 2021), sistem berbasis web juga mampu menyajikan informasi secara *real-time* (Syari & Supriyono, 2020) serta mendukung pengambilan keputusan melalui penyediaan data yang lebih transparan (Holloway, 2024). Oleh karena itu, pengembangan sistem persediaan barang berbasis web yang mengintegrasikan metode **FIFO** dan **ROP** diharapkan mampu meningkatkan efektivitas pengelolaan stok, mengurangi potensi kerugian, serta memperkuat daya saing Toko Raja Aki di era digital.

2. METODE

Pengembangan sistem pengelolaan persediaan barang di Toko Raja Aki, penelitian ini menggunakan model **Waterfall** sebagai metode pengembangan perangkat lunak. Model yang diperkenalkan oleh Royce pada tahun 1970 ini menerapkan tahapan pengembangan secara berurutan dan sistematis. Pemilihan metode **Waterfall** didasarkan pada karakteristiknya yang terstruktur sehingga setiap kebutuhan sistem, termasuk implementasi metode **First In First Out (FIFO)** dan **Reorder Point (ROP)**, dapat dianalisis serta dirancang secara menyeluruh sebelum proses pengkodean dilakukan (Nugraha et al., 2023). Selain itu, menurut Sari dan Jannah (2022), model **Waterfall** sesuai diterapkan pada proyek yang memiliki kebutuhan sistem yang telah terdefinisi dengan jelas karena setiap tahapan harus diselesaikan dan diverifikasi terlebih dahulu sebelum dilanjutkan ke tahap pengembangan berikutnya.

Alur pengembangan sistem secara sistematis dalam penelitian ini diilustrasikan pada Gambar 1 mengenai Langkah-langkah metode *Waterfall* berikut:



Gambar 1. Langkah-langkah Metode *Waterfall*

Berdasarkan urutan pada Gambar 1, tahapan penelitian dilaksanakan sebagai berikut:

2.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan sebagai dasar dalam pengembangan sistem untuk mengidentifikasi permasalahan pada proses yang sedang berjalan serta menentukan kebutuhan sistem yang akan dikembangkan. Pengumpulan informasi dilakukan melalui observasi terhadap proses pengelolaan persediaan di Toko Raja Aki, disertai wawancara dengan pemilik usaha dan admin guna memperoleh pemahaman mengenai alur kerja serta kendala yang dihadapi dalam kegiatan operasional.

Berdasarkan hasil analisis tersebut, kebutuhan sistem dikelompokkan menjadi dua bagian utama. Sommerville (2011) menjelaskan bahwa kebutuhan sistem terdiri atas **kebutuhan fungsional**, yang menjelaskan layanan dan fungsi yang harus disediakan oleh sistem, serta **kebutuhan nonfungsional**, yang berkaitan dengan karakteristik, batasan, dan kualitas sistem, seperti performa, keamanan, serta kemudahan akses. Adapun kebutuhan sistem yang diidentifikasi dalam penelitian ini adalah sebagai berikut:

1. Kebutuhan Fungsional

Kebutuhan fungsional mendefinisikan fitur-fitur utama dan layanan yang harus disediakan oleh sistem, meliputi:

- Manajemen Akses: Menyediakan fitur login dan logout untuk menjamin keamanan akses bagi pemilik dan pegawai.
- Manajemen Inventori (FIFO): Sistem mampu mengelola data barang (tambah, ubah, hapus, lihat) serta menghitung pengeluaran stok secara otomatis menggunakan logika *first in first out* (FIFO).

- Pencatatan Transaksi: Mendokumentasikan setiap aktivitas barang masuk (pembelian) dan barang keluar (penjualan) secara digital.
- Perhitungan Pemesanan Kembali (ROP): Sistem dapat menghitung *reorder point* (ROP) secara presisi berdasarkan histori penjualan dan waktu tunggu (lead time).
- Sistem Notifikasi: Memberikan peringatan dini (notifikasi) kepada pengguna ketika jumlah stok telah mencapai batas minimum ROP.
- Pelaporan dan Visualisasi: Menyajikan laporan transaksi yang akurat serta grafik perkembangan stok untuk memudahkan pemantauan tren bisnis.

2. Kebutuhan Nonfungsional

Kebutuhan nonfungsional mencakup karakteristik serta batasan teknis yang harus dipenuhi oleh sistem untuk mendukung kinerja dan memberikan pengalaman penggunaan yang optimal, meliputi:

- Aksesibilitas: Sistem dikembangkan berbasis web agar dapat diakses secara fleksibel melalui berbagai peramban (browser).
- User Experience (UX): Memiliki antarmuka yang intuitif, sederhana, dan mudah dioperasikan oleh pengguna dengan berbagai tingkat literasi digital.
- Performa dan Akurasi: Mampu memproses dan menampilkan data dalam waktu singkat dengan tingkat akurasi perhitungan yang tinggi.
- Keamanan Data: Menjamin integritas data melalui sistem autentikasi pengguna yang ketat untuk mencegah akses dari pihak yang tidak berwenang.

Seluruh poin kebutuhan ini menjadi fondasi utama dalam menentukan arah perancangan sistem pada tahap berikutnya, sehingga solusi yang dihasilkan benar-benar menjawab tantangan efisiensi di Toko Raja Aki.

2.2 Perancangan Sistem

Tahap perancangan sistem merupakan fase krusial untuk mentransformasikan kebutuhan fungsional ke dalam model teknis yang terstruktur. Perancangan ini berfungsi sebagai cetak biru (blueprint) dalam membangun sistem agar sesuai dengan alur bisnis yang diharapkan.

2.2.1 Use Case Diagram

Use Case Diagram digunakan untuk memetakan ruang lingkup sistem serta menggambarkan bagaimana para pengguna (aktor) berinteraksi dengan fitur-fitur yang tersedia. Menurut

Booch, Rumbaugh, dan Jacobson (1999), use case diagram merupakan bagian dari Unified Modeling Language (UML) yang digunakan untuk mendeskripsikan fungsionalitas sistem dari sudut pandang pengguna secara visual. Melalui diagram ini, alur kerja dan batasan hak akses dalam sistem Toko Raja Aki dapat dipahami dengan mudah.

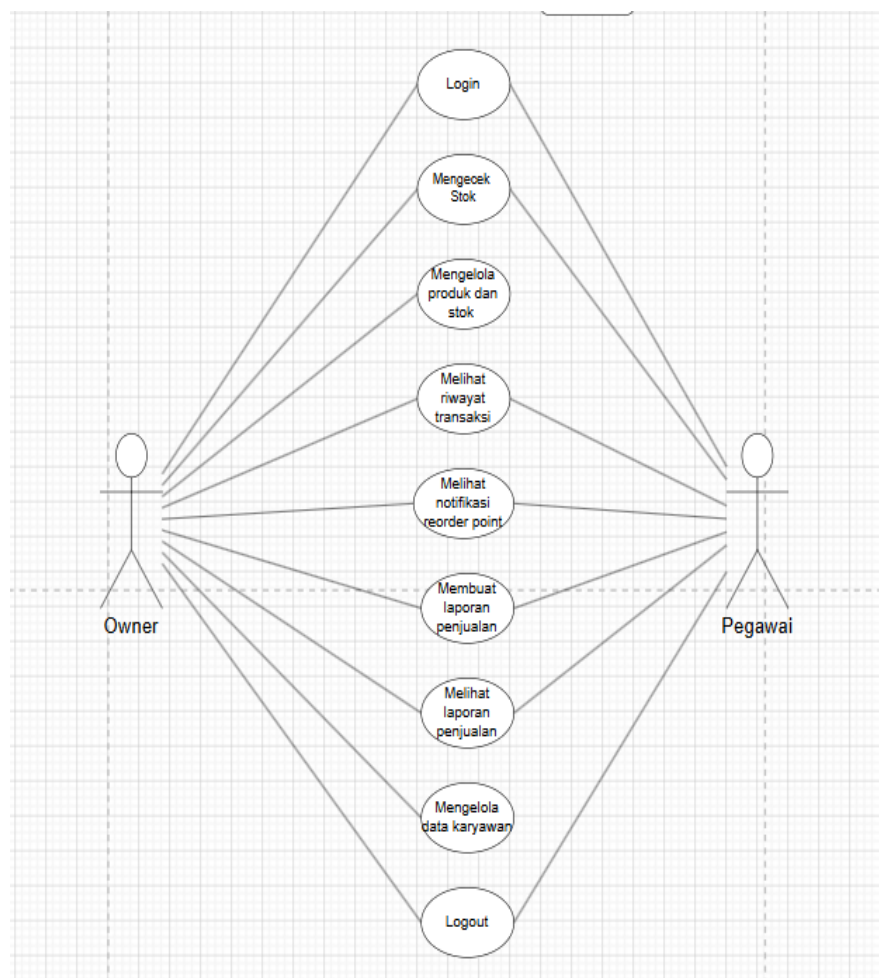
Sebagaimana digambarkan pada Gambar 2, sistem ini melibatkan dua aktor utama dengan pembagian tugas yang jelas:

1. *Owner*

Memegang kendali penuh atas seluruh fungsi sistem. Selain bisa melakukan semua pekerjaan pegawai, *Owner* memiliki otoritas khusus untuk mengelola data pengguna (akun pegawai), mengatur parameter sistem seperti batas minimal stok (*Reorder point*), serta memantau laporan perkembangan bisnis secara menyeluruh untuk keperluan pengambilan keputusan.

2. Pegawai

Fokus pada aktivitas operasional harian di toko. Tugas utamanya meliputi pencatatan barang masuk dari pemasok, mendata barang keluar (penjualan), serta memantau ketersediaan stok aki secara real-time.



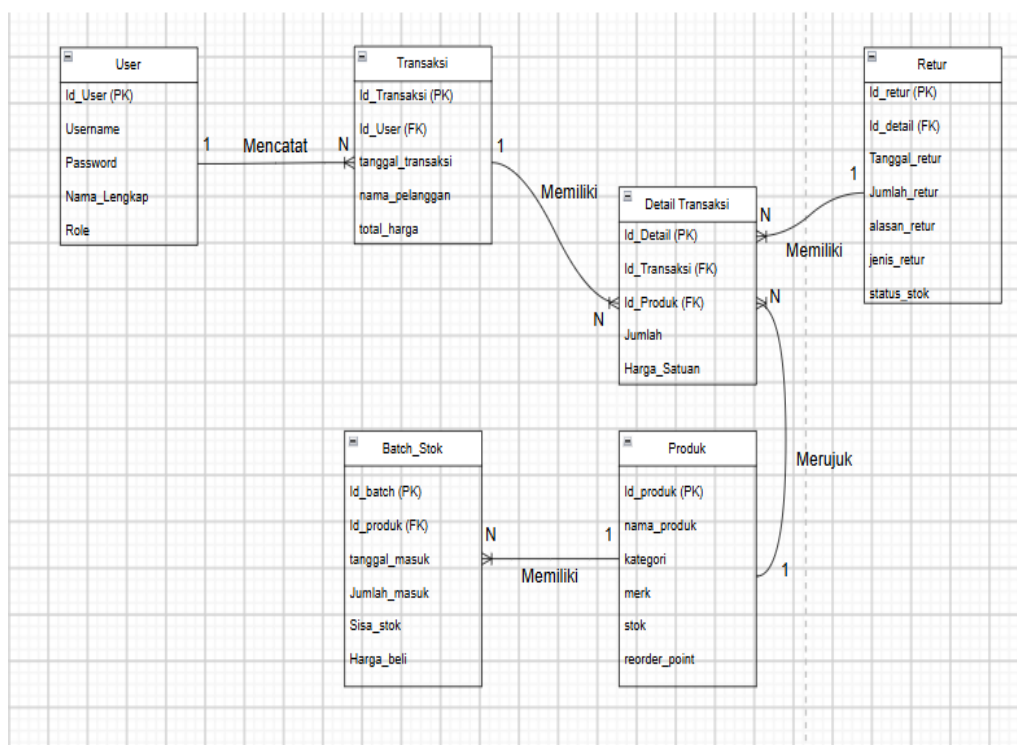
Gambar 2. *Use Case Diagram* Sistem Web Toko Raja Aki

Berdasarkan pada gambar 2, pemisahan peran antara *Owner* dan Pegawai ini bertujuan untuk menjaga keamanan data sekaligus memastikan setiap bagian operasional di Toko Raja Aki berjalan lebih teratur.

2.2.2 Entity Relationship Diagram (ERD)

Entity Relationship Diagram merupakan teknik pemodelan *database* yang pertama kali diusulkan oleh Chen (1976) untuk merepresentasikan entitas, atribut, dan relasi antardata dalam sebuah sistem secara konseptual. ERD pada Gambar 3 merupakan skema konseptual yang memetakan aliran data dan hubungan antartabel dalam *database*. Struktur yang ditunjukkan mengintegrasikan entitas utama seperti data pengguna, kategori, barang, serta riwayat transaksi masuk dan keluar.

Relasi antar-entitas memastikan seluruh aktivitas operasional yang dilakukan *Owner* maupun Pegawai tercatat secara otomatis dan saling terhubung. Dengan susunan *database* pada Gambar 3, setiap perubahan stok dan transaksi dapat terdokumentasi secara sistematis, sehingga laporan yang dihasilkan untuk pemilik toko menjadi lebih akurat dan konsisten.



Gambar 3. *Entity Relationship Diagram*(ERD)

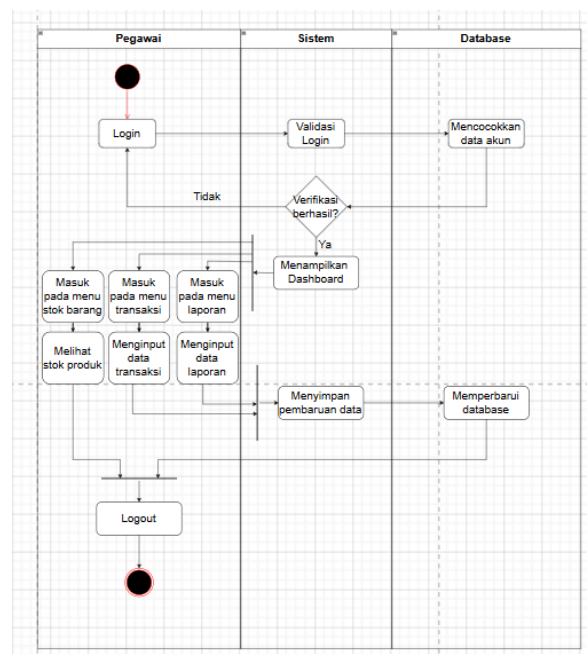
2.2.3 Activity Diagram

Activity Diagram dimanfaatkan untuk menggambarkan rangkaian aktivitas yang terjadi

dalam suatu sistem. Fowler (2004) menjelaskan bahwa Activity Diagram merupakan bagian dari *Unified Modeling Language* (UML) yang digunakan untuk memodelkan alur proses atau logika prosedural, baik secara sekuensial maupun paralel. Dalam penelitian ini, setiap diagram aktivitas dibuat berdasarkan peran aktor yang terlibat sehingga proses operasional sistem dapat divisualisasikan dengan lebih sistematis dan mudah dipahami.

1. Activity Diagram Pegawai

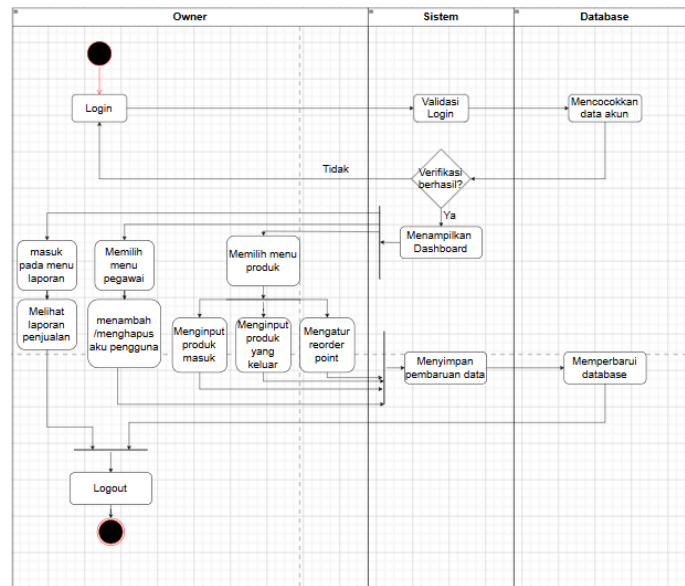
Sebagaimana ditunjukkan pada Gambar 4, alur kerja pegawai berfokus pada aktivitas harian toko. Proses yang tergambar dalam Gambar 4 dimulai dari input transaksi penjualan dan barang masuk, yang kemudian diproses oleh sistem untuk memperbarui stok secara real-time menggunakan metode FIFO. Alur pada Gambar 4 ini berakhir saat data berhasil tersimpan dengan akurat di dalam *database*.



Gambar 4. Activity Diagram Pegawai

2. Activity Diagram Owner

Sementara itu, Gambar 5 memaparkan alur kerja khusus bagi pemilik toko. Berdasarkan urutan pada Gambar 5, aktivitas dimulai saat *Owner* mengakses menu laporan atau manajemen pengguna. Sistem dalam Gambar 5 kemudian akan menyajikan ringkasan data transaksi dan notifikasi stok minimum (*Reorder point*) yang valid sebagai dasar pengambilan keputusan bisnis yang cepat.



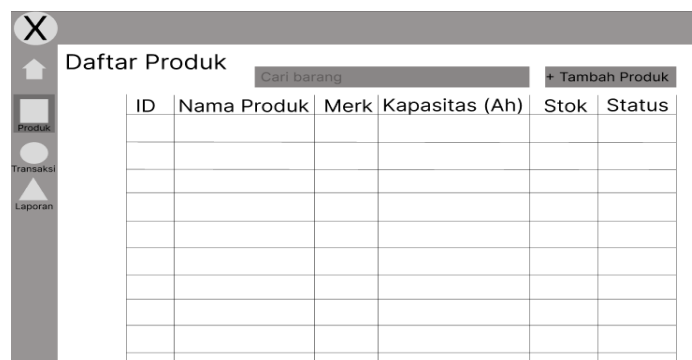
Gambar 5. Activity Diagram Owner

2.2.4 Perancangan Antarmuka (Wireframe)

Perancangan antarmuka menggunakan *Wireframe* merupakan bagian dari pendekatan user-centered design, di mana *Wireframe* berfungsi sebagai cetak biru visual yang menggambarkan tata letak dan struktur antarmuka sebelum implementasi dilakukan (Garrett, 2011). Sistem Toko Raja Aki memiliki beberapa halaman utama yang dirancang untuk mendukung efisiensi manajemen stok dan transaksi sebagai berikut:

1. Halaman Produk

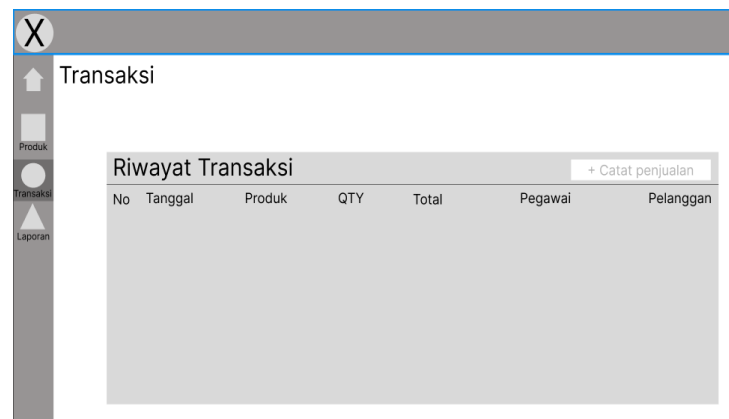
Sebagaimana disajikan pada Gambar 6, halaman ini berfungsi sebagai pusat kendali data barang. Melalui tampilan pada Gambar 6, *Owner* dan Pegawai dapat mengelola informasi aki secara detail serta memantau status stok kritis yang telah terintegrasi dengan perhitungan *Reorder point* (ROP).



Gambar 6. Wireframe Halaman Produk

2. Halaman Transaksi

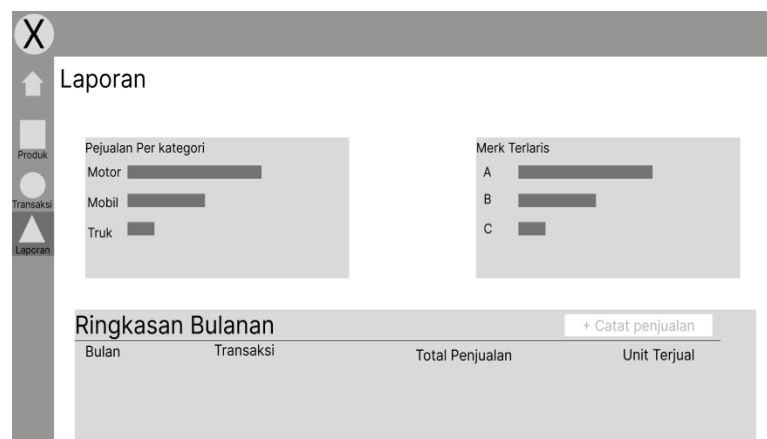
Mengacu pada rancangan di Gambar 7, halaman ini digunakan untuk mendokumentasikan setiap aktivitas barang masuk dan keluar. Sistem dalam Gambar 7 secara otomatis menerapkan logika FIFO, sehingga stok aki yang masuk lebih awal akan diprioritaskan dalam pencatatan penjualan guna menjaga sirkulasi barang yang optimal.



Gambar 7. *Wireframe* Halaman Transaksi

3. Halaman Laporan

Seperti yang ditunjukkan pada Gambar 8, halaman ini menyediakan ringkasan seluruh aktivitas operasional toko. Melalui sajian data pada Gambar 8, *Owner* dapat melihat grafik perkembangan bisnis dan laporan saldo stok akhir secara akurat untuk mendukung pengambilan keputusan strategis



Gambar 8. *Wireframe* Halaman Laporan

2.3 Implementasi

Tahap implementasi merupakan fase realisasi teknis berdasarkan seluruh rancangan yang telah disusun sebelumnya. Pada tahap ini, sistem Toko Raja Aki mulai dikembangkan secara menyeluruh dengan cakupan sebagai berikut:

- **Pembangunan Database:** Mengonversi rancangan ERD menjadi struktur *database* fisik untuk menyimpan seluruh data operasional secara terorganisir.
- **Pengembangan Antarmuka & Logika:** Membangun modul pengelolaan data barang serta transaksi masuk dan keluar menggunakan bahasa pemrograman dan *Framework* yang relevan.
- **Integrasi Sistem:** Menyatukan seluruh modul fungsional ke dalam satu ekosistem aplikasi yang utuh agar dapat beroperasi secara sinkron.

Dalam proses ini, terdapat dua algoritma utama yang diimplementasikan untuk menjamin efisiensi manajemen stok:

1. Implementasi Algoritma FIFO

Sistem menerapkan algoritma FIFO sebagai aturan utama dalam sirkulasi barang. Secara teknis, setiap stok aki yang masuk ke gudang akan dicatat berdasarkan urutan tanggal penerimaan. Saat terjadi transaksi penjualan, logika sistem secara otomatis akan memprioritaskan pengeluaran stok yang memiliki tanggal masuk paling awal. Penerapan ini sangat krusial untuk menjaga kualitas barang dan menghindari risiko kerusakan akibat penyimpanan yang terlalu lama (stok lama).

2. Implementasi Fitur Reorder Point (ROP)

Untuk mencegah terjadinya kehabisan stok (*stockout*), sistem menerapkan konsep *Reorder Point* (ROP) sebagai acuan dalam proses pemesanan kembali barang. Pada sistem ini, nilai *Reorder Point* ditetapkan oleh *Owner* melalui pengaturan batas minimum stok pada setiap produk. Nilai tersebut selanjutnya digunakan sebagai indikator untuk menentukan kapan suatu produk perlu dipesan kembali. Sistem memantau ketersediaan stok secara *real-time*. Ketika jumlah stok telah mencapai atau berada di bawah batas minimum yang ditentukan, produk akan secara otomatis ditampilkan pada halaman **Pesan Ulang** disertai notifikasi kepada *Owner*. Mekanisme ini membantu pengguna mengetahui produk yang perlu segera dipesan kembali sehingga risiko kehabisan stok dapat diminimalkan dan ketersediaan barang tetap terjaga. Fitur *Reorder Point* diimplementasikan pada sistem persediaan barang berbasis web yang dikembangkan menggunakan *framework* Next.js. Aplikasi dideploy secara *serverless* melalui platform Vercel, sedangkan pengelolaan data dilakukan menggunakan Neon *Database* berbasis PostgreSQL. Kombinasi teknologi tersebut memungkinkan data persediaan diperbarui secara *real-time*, sehingga informasi stok dan notifikasi pemesanan kembali selalu sesuai dengan kondisi terkini. Sebelum diimplementasikan, seluruh fitur telah

melalui pengujian internal untuk memastikan fungsionalitas sistem berjalan dengan baik dan mendukung proses pengelolaan persediaan secara efektif.

2.4 Pengujian

Tahap pengujian merupakan langkah krusial untuk memastikan bahwa seluruh fungsionalitas sistem telah selaras dengan kebutuhan yang ditetapkan pada tahap analisis. Strategi pengujian yang diterapkan dalam sistem ini mencakup dua pendekatan utama:

1. Pengujian Fungsional (*Blackbox Testing*)

Metode *Blackbox Testing* yang pertama kali diperkenalkan secara komprehensif oleh Myers (1979) mendefinisikannya sebagai pendekatan pengujian yang berfokus pada validasi output berdasarkan input tanpa memperhatikan struktur kode internal program. Metode ini digunakan untuk memvalidasi performa fitur-fitur sistem tanpa mengintervensi struktur kode program. Fokus utamanya adalah memastikan input yang diberikan menghasilkan output yang akurat. Ruang lingkup pengujian ini meliputi:

- Autentikasi: Validasi keamanan hak akses pada halaman login.
- Manajemen Inventaris: Pengujian fungsi tambah, ubah, dan hapus pada data barang.
- Logika Transaksi: Verifikasi akurasi pencatatan barang masuk dan keluar.
- Otomasi FIFO: Memastikan urutan pengeluaran barang sesuai dengan tanggal masuk tertua.
- Akurasi ROP: Pengujian ketepatan munculnya notifikasi saat stok mencapai titik minimum.
- Visualisasi Data: Pemeriksaan integritas data pada pembuatan laporan serta grafik stok.

2. Pengujian Pengguna (*Usability Testing*)

Untuk menjamin sistem tidak hanya berfungsi tetapi juga nyaman digunakan, dilakukan pengujian usability dengan standar *System Usability Scale* (SUS) yang dikembangkan oleh Brooke (1996), yaitu kuesioner terstandar berisi 10 butir pernyataan yang menghasilkan skor 0–100 untuk menentukan tingkat kegunaan sistem secara cepat dan reliabel. Fokus utama pengujian ini adalah mendapatkan perspektif objektif dari *Owner* dan Pegawai mengenai:

- Efisiensi Kerja: Kelancaran dalam mengelola stok dan transaksi.
- Kemudahan Akses: Seberapa cepat menu navigasi dapat dipahami tanpa bantuan eksternal.
- Kenyamanan Visual: Dampak penggunaan tema Industrial Dark Mode terhadap kenyamanan mata pengguna.

- Akurasi Operasional: Kejelasan sistem dalam mengarahkan pengguna saat terjadi kesalahan input.

Evaluasi dilakukan melalui kuesioner terstandar yang menghasilkan nilai akhir untuk menentukan apakah sistem Toko Raja Aki sudah masuk kategori *Acceptable* dan ergonomis untuk digunakan sehari-hari.

2.5 Maintenance

Tahap pemeliharaan bertujuan untuk menjamin keberlanjutan dan performa optimal sistem setelah diimplementasikan di Toko Raja Aki. Langkah-langkah pemeliharaan yang dilakukan meliputi:

- Perbaikan (*Bug Fixing*): Menangani kesalahan teknis atau kekeliruan logika yang muncul selama penggunaan operasional sehari-hari.
- Pembaruan Adaptif: Menyesuaikan sistem terhadap perubahan perangkat keras, pembaruan *database*, atau kebutuhan fitur baru di masa mendatang.
- Optimasi Performa: Melakukan pemeliharaan rutin pada *database* agar proses pencarian stok dan penyajian laporan tetap cepat dan stabil.
- Keamanan Data: Melakukan pencadangan (*backup*) data secara berkala untuk melindungi riwayat transaksi dan informasi stok dari risiko kehilangan atau kerusakan.

Melalui prosedur ini, sistem diharapkan tetap reliabel dalam mendukung manajemen stok FIFO dan perhitungan ROP bagi *Owner* dan Pegawai.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Penelitian

Penelitian ini menghasilkan sebuah aplikasi pengelolaan persediaan barang berbasis web yang telah berhasil dikembangkan dan diimplementasikan pada lingkungan produksi. Sistem dibangun menggunakan *Framework* Next.js, kemudian dideploy melalui platform Vercel dengan dukungan *database* PostgreSQL serverless yang disediakan oleh Neon *Database*. Pada bagian berikut akan dijelaskan implementasi antarmuka sistem, fungsi dari masing-masing halaman, serta hasil pengujian yang telah dilakukan.

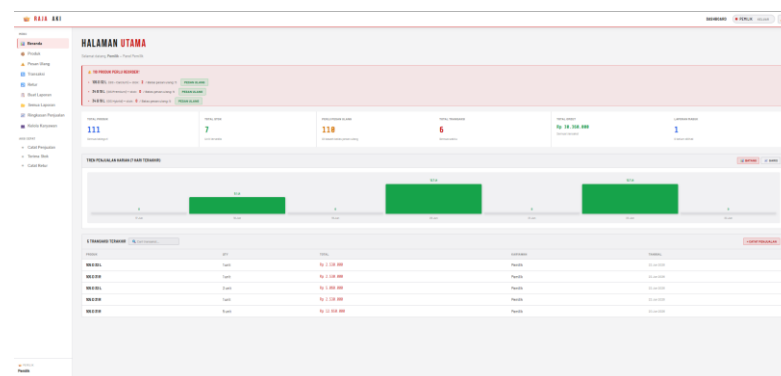
3.1.1 Antarmuka Sistem dan Fungsi Halaman

Aplikasi dirancang dengan konsep tampilan Industrial Dark Mode yang bertujuan meningkatkan kenyamanan pengguna ketika mengoperasikan sistem dalam waktu yang

cukup lama, khususnya pada lingkungan kerja toko. Adapun halaman-halaman utama yang tersedia dalam sistem meliputi:

1. Halaman *Dashboard*

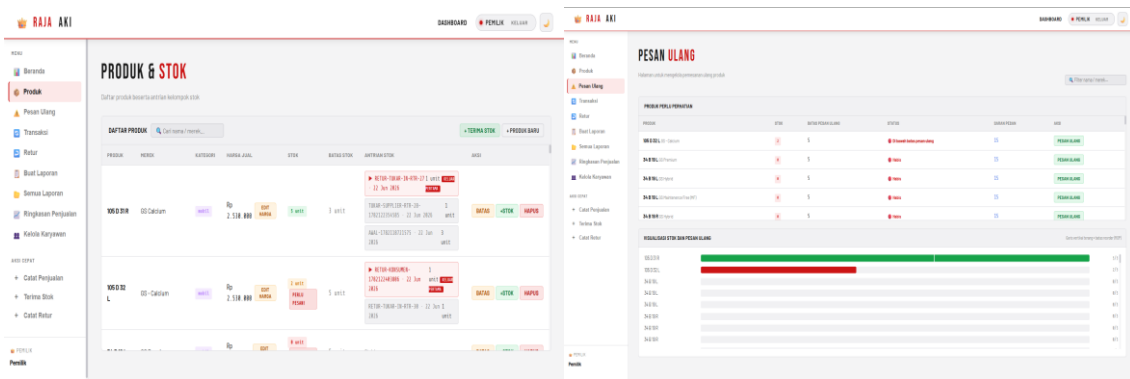
Halaman *dashboard* merupakan tampilan awal yang diakses pengguna setelah proses autentikasi berhasil dilakukan. Seperti yang ditunjukkan pada Gambar 9, halaman ini menyajikan ringkasan informasi penting, meliputi jumlah produk yang tersedia, lima transaksi terbaru, serta informasi mengenai kondisi stok barang. Selain itu, sistem secara otomatis memberikan notifikasi kepada *Owner* apabila jumlah persediaan suatu produk telah mencapai atau berada di bawah nilai *Reorder Point* (ROP). Notifikasi tersebut berfungsi sebagai pengingat agar proses pemesanan kembali dapat segera dilakukan sehingga ketersediaan stok tetap terjaga.



Gambar 9 . Halaman *Dashboard*

2. Halaman Manajemen Produk dan Halaman Pesan Ulang

Berdasarkan Gambar 10, halaman manajemen produk digunakan untuk mengelola data barang yang tersimpan dalam sistem. Melalui halaman ini, pengguna dapat menambahkan, mengubah, menghapus, dan melihat detail produk. *Owner* juga dapat mencatat barang yang baru diterima dari pemasok sehingga stok dapat diperbarui secara otomatis. Informasi yang dikelola meliputi nama produk, merek, kapasitas aki (Ah), kategori kendaraan, serta batas minimum stok yang digunakan dalam perhitungan *Reorder Point* (ROP). Selain itu, sistem menyediakan halaman Pesan Ulang yang menampilkan produk yang telah mencapai batas pemesanan kembali, sehingga memudahkan pengguna dalam memantau dan melakukan pemesanan ulang stok tepat waktu.



Gambar 10. Halaman Produk & Pesan Ulang

3. Halaman Transaksi

Halaman transaksi berfungsi untuk mencatat dan mengelola seluruh aktivitas pergerakan persediaan barang, baik barang masuk maupun barang keluar. Setiap transaksi penjualan yang dilakukan pelanggan akan tersimpan dan ditampilkan sebagai riwayat transaksi pada halaman ini, sebagaimana ditunjukkan pada Gambar 11. Proses transaksi sistem secara otomatis menerapkan metode FIFO (*First In First Out*) dengan memprioritaskan pengeluaran stok yang memiliki tanggal penerimaan paling awal. Penerapan metode ini bertujuan untuk menjaga perputaran persediaan agar berjalan dengan baik, meminimalkan risiko penumpukan stok lama, serta memastikan ketersediaan barang tetap terjaga secara optimal.

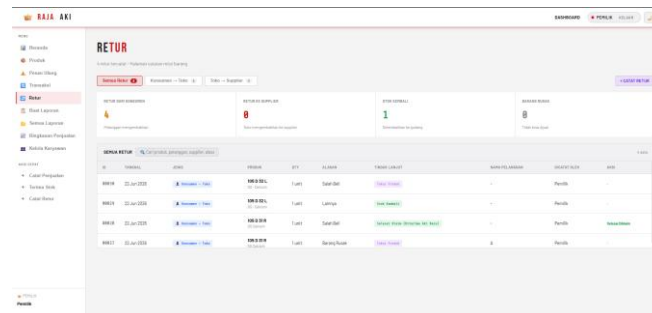
#	TANGGAL	PRODUK	QUANTITAS	STOK	TOTAL	KETERANGAN	REMARKS
00001	10.01.2020	YOGURT	1	1	Rp 2.500.000	Paid	
00002	10.01.2020	YOGURT	1	1	Rp 2.500.000	Paid	
00003	10.01.2020	YOGURT	2	2	Rp 5.000.000	Paid	

Gambar 11. Halaman Transaksi

4. Halaman Retur

Halaman retur barang berfungsi untuk mengelola seluruh proses pengembalian barang, baik yang berasal dari pelanggan maupun yang dikembalikan kepada pemasok. Seperti yang ditunjukkan pada Gambar 12, pengguna dapat mencatat berbagai informasi terkait retur, seperti nama produk, jumlah barang yang diretur, alasan pengembalian, serta tanggal terjadinya transaksi retur. Data yang tersimpan akan diproses secara otomatis oleh sistem untuk menyesuaikan jumlah stok sesuai dengan jenis retur yang dilakukan. Dengan adanya

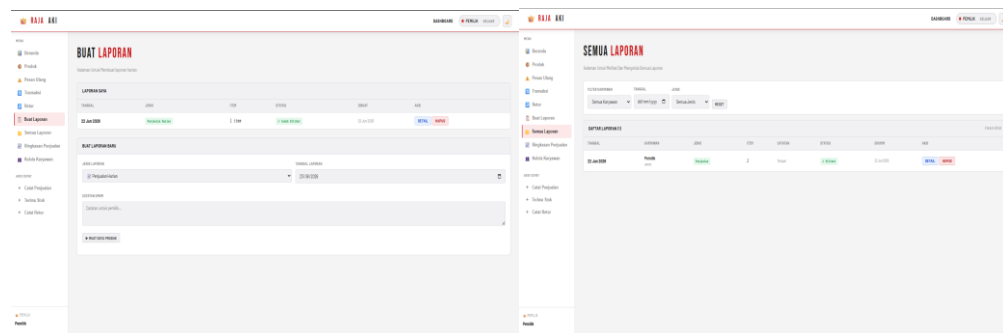
fitur ini, pengelolaan barang retur dapat dilakukan secara lebih terorganisir, terdokumentasi dengan baik, serta mendukung proses pengawasan dan audit persediaan secara lebih efektif.



Gambar 12. Halaman Retur

5. Halaman Laporan

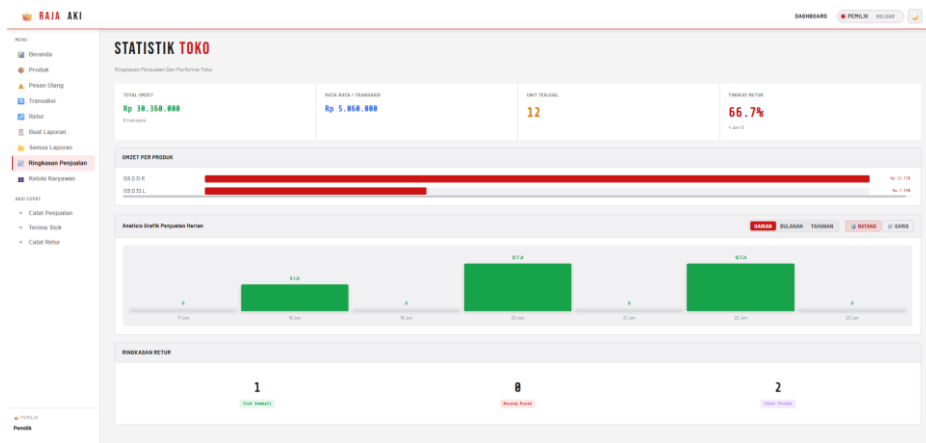
Halaman laporan digunakan sebagai sarana untuk menampilkan dan mengelola laporan yang dibuat oleh pengguna. Seperti yang terlihat pada Gambar 13, halaman ini menyajikan informasi berupa nama produk, harga, dan keterangan laporan. Selain menampilkan data laporan, sistem juga memungkinkan pengguna untuk membuat serta mengirimkan laporan secara langsung. Seluruh laporan yang telah dikirim akan tersimpan dan ditampilkan pada halaman laporan *Owner*, sehingga proses pemantauan dan pengelolaan laporan dapat dilakukan secara terpusat dan lebih terorganisir.



Gambar 13. Halaman Laporan

6. Halaman Ringkasan Penjualan

Halaman ringkasan penjualan berfungsi untuk menyajikan informasi mengenai hasil penjualan dan aktivitas operasional toko dalam bentuk visual yang mudah dipahami. Seperti yang ditunjukkan pada Gambar 14, halaman ini menampilkan berbagai informasi yang dapat digunakan oleh *Owner* untuk memantau perkembangan bisnis secara lebih efektif. Data yang disajikan juga membantu dalam proses analisis dan menjadi dasar dalam pengambilan keputusan yang berkaitan dengan pengelolaan persediaan maupun operasional toko.



Gambar 14. Halaman Ringkasan Penjualan

7. Halaman Manajemen Karyawan

Halaman manajemen karyawan hanya dapat diakses oleh *Owner* dan digunakan untuk mengelola data pengguna yang memiliki hak akses ke dalam sistem. Seperti yang terlihat pada Gambar 15, *Owner* dapat menambahkan, mengubah, maupun menghapus informasi pengguna, seperti nama lengkap, username, password, dan tingkat hak akses. Penerapan pembatasan akses ini bertujuan untuk meningkatkan keamanan sistem serta memastikan setiap pengguna hanya dapat mengakses fitur dan informasi sesuai dengan peran dan kewenangannya masing-masing.

The screenshot displays the 'KELOLA KARYAWAN' dashboard. It features a table with the following columns: 'NAMA LENGKAP', 'USERNAME', 'HAK AKSES / ROLE', 'STATUS AKTIF', and 'Aksi'. The table contains two rows: 'Pengguna' and 'Pemilik'. The 'Pengguna' row has a 'password' field and a 'status' field. The 'Pemilik' row has a 'password' field and a 'status' field. The 'Aksi' column contains buttons for 'EDIT' and 'HAPUS'.

Gambar 15. Halaman Manajemen Karyawan

3.2 Pembahasan

3.2.1 Hasil Pengujian Fungsional (*Blackbox*)

Pengujian *Blackbox* dilakukan untuk memverifikasi bahwa seluruh modul dan fitur pada website berfungsi sesuai dengan kebutuhan serta tujuan pengembangan sistem. Pengujian ini berfokus pada kesesuaian hasil keluaran berdasarkan masukan yang diberikan tanpa meninjau struktur atau kode program. Hasil pelaksanaan pengujian *Blackbox* pada sistem disajikan

pada Tabel 1.

Tabel 1. Pengujian *blackbox*

Modul	Scenario	Hasil yang diharapkan	Hasil di aplikasi
<i>Login</i>	Memasukkan <i>username</i> dan <i>password</i> benar	Masuk ke halaman <i>dashboard</i>	Valid
Menerima produk baru	Mengisi data produk dan semua data terisi	Data masuk ke <i>database</i> dan tampil di halaman produk	Valid
Mengubah data produk	Melakukan perubahan data produk dan semua data sudah diisi	Data berhasil diubah di <i>database</i> dan ditampilkan di halaman	Valid
Menghapus data produk	Melakukan penghapusan pada data	Data terhapus di <i>database</i> dan menghilang dari halaman	Valid
Menerima stok	Mengisi data produk yang ditambahkan stoknya	Data masuk ke <i>database</i> dan stok di halaman produk bertambah	Valid
Pesan ulang	Mengisi data produk yang memerlukan pesan ulang	Data ditambahkan ke <i>database</i> dan jumlah stok produk bertambah	Valid
Membuat laporan	Mengisi data yang diperlukan dan semua data sudah diisi	Data ditambahkan ke <i>database</i> dan ditampilkan di halaman laporan <i>Owner</i>	Valid
Melihat	Memilih laporan	Data ditampilkan di	Valid

laporan	yang ingin dilihat	halaman laporan <i>Owner</i>	
Manajemen akun pegawai	Menambahkan data pegawai dan semua data telah diisi	Data berhasil masuk ke <i>database</i> dan pegawai bisa melakukan aksi <i>login</i>	Valid
<i>Logout</i>	Keluar dari website	Kembali ke halaman <i>login</i>	Valid

3.2.2 Hasil Pengujian Usabilitas (SUS)

Pengujian *usability* melibatkan 30 responden dengan rentang usia dan latar belakang pekerjaan yang beragam. Mayoritas responden berusia 20–25 tahun (40%) dan berprofesi sebagai ASN/Pegawai Pemerintah (40%). Keberagaman karakteristik responden ini memberikan sudut pandang yang lebih luas dalam menilai tingkat kemudahan penggunaan sistem. Adapun daftar pertanyaan beserta skala nilai jawaban responden yang digunakan dalam pengujian *usability* ditampilkan pada Tabel 2.

Tabel 2. Daftar Pertanyaan Pengujian SUS

Kode	Daftar Pertanyaan	Skala
Q1	Saya berpikir akan menggunakan sistem ini lagi	1-5
Q2	Saya merasa sistem ini rumit untuk digunakan	1-5
Q3	Saya merasa sistem ini mudah digunakan	1-5
Q4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini	1-5
Q5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya	1-5

Q6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)	1-5
Q7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat	1-5
Q8	Saya merasa sistem ini membingungkan	1-5
Q9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini	1-5
10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini	1-5

Untuk mengetahui seberapa mudah pengguna mengakses dan menggunakan website Sistem Persediaan Barang di Toko Raja Aki, penelitian ini memakai metode System Usability Scale (SUS) yang diciptakan oleh Brooke pada tahun 1996. Metode SUS terdiri dari 10 pertanyaan yang digunakan untuk menilai sejauh mana suatu sistem mudah digunakan oleh pengguna. Daftar pertanyaan dan jawaban yang diberikan oleh responden terlihat di Tabel 2. Setiap pertanyaan memiliki lima opsi jawaban yang dinilai dengan skala, yaitu Sangat Tidak Setuju (STS) bernilai 1, Tidak Setuju (TS) bernilai 2, Netral (N) bernilai 3, Setuju (S) bernilai 4, dan Sangat Setuju (SS) bernilai 5. Perhitungan skor SUS dilakukan dengan mengubah setiap jawaban yang diberikan oleh responden menjadi angka yang merepresentasikan kontribusinya. Untuk soal nomor ganjil (1, 3, 5, 7, dan 9), skor yang diberikan adalah nilai jawaban dikurangi 1. Sementara itu, untuk pertanyaan bernomor genap (2, 4, 6, 8, dan 10), skor kontribusi didapatkan dengan mengurangi nilai jawaban dari angka 5. Setelah semua skor kontribusi dijumlahkan, hasilnya dikalikan dengan 2,5 sehingga mendapatkan nilai SUS yang berkisar antara 0 hingga 100. Rumus perhitungan SUS adalah sebagai berikut:

Skor Pertanyaan Ganjil :

$$x = \frac{\text{Skor}}{\text{Jawaban}} - 1$$

Skor Pertanyaan Genap :

$$x = 5 - \frac{Skor}{Jawaban}$$

Rumus skor SUS :

$$SUS = (\sum X) \times 2,5$$

Keterangan :

- (X) = skor kontribusi setiap pertanyaan
- $(\sum X)$ = jumlah seluruh skor kontribusi
- 2,5 = konstanta pengali untuk mengubah skor ke rentang 0–100
- SUS = nilai akhir System *Usability* Scale

Berdasarkan hasil pengisian kuesioner **System Usability Scale (SUS)** oleh 30 responden, diperoleh data jawaban untuk setiap butir pertanyaan yang kemudian diolah sesuai dengan metode perhitungan SUS. Rekapitulasi jawaban responden terhadap masing-masing pertanyaan disajikan pada **Tabel 3**. Hasil tersebut menjadi dasar dalam perhitungan skor SUS untuk mengetahui tingkat usability website Sistem Persediaan Barang pada Toko Raja Aki.

Tabel 3. Hasil pengujian SUS

[illegible]

4	4	4	4	4	4	4	4	4	4	40	100
3	3	4	3	3	2	3	2	3	3	29	73
4	0	4	0	4	0	4	0	4	0	20	50
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	0	4	4	4	4	4	4	36	90
4	4	4	1	4	4	4	4	4	1	34	85
4	3	4	3	4	3	4	4	4	4	37	93
1	3	2	3	1	3	2	3	3	2	23	58
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	1	4	0	3	4	3	4	4	2	29	73
4	4	4	4	4	4	4	4	4	4	40	100
4	3	4	1	4	2	3	3	4	2	30	75

4	4	4	3	3	4	3	4	3	2	34	85
4	3	4	0	4	4	4	4	4	3	34	85
4	4	4	4	4	4	4	4	4	4	40	100
4	3	4	3	4	0	4	3	4	3	32	80
3	2	4	2	4	0	4	3	4	0	26	65
3	3	4	3	4	3	4	3	4	3	34	85
3	3	4	3	4	3	1	3	4	3	31	78
3	4	4	4	4	2	4	3	4	4	36	90
3	3	3	4	3	3	4	4	4	4	35	88
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	3	3	3	4	4	3	4	36	90
3	3	3	1	3	3	2	3	3	1	25	63
3	1	3	1	3	1	3	1	3	1	20	50
4	3	4	4	3	3	2	4	4	4	35	88

4	4	4	4	4	4	4	4	4	4	40	100
Rata-rata hasil (Hasil Akhir)											85

Berdasarkan hasil pengolahan data kuesioner System Usability Scale (SUS), website yang dikembangkan memperoleh skor sebesar 85. Mengacu pada pedoman interpretasi yang disusun oleh Bangor, Kortum, dan Miller (2009), perolehan skor tersebut masuk dalam kategori Acceptable, dengan predikat Grade A, serta berada pada tingkat Adjective Rating Excellent. Skor ini juga berada di atas nilai rata-rata SUS secara umum, yaitu 68, sehingga dapat disimpulkan bahwa tingkat usability website tergolong sangat baik. Dengan demikian, dapat dikatakan bahwa Sistem Persediaan Barang pada Toko Raja Aki relatif mudah dipelajari, mudah dioperasikan oleh pengguna, dan mampu mendukung proses pengelolaan persediaan barang secara lebih efektif dan efisien. Acuan klasifikasi interpretasi skor SUS yang digunakan dalam penelitian ini dapat dilihat pada Tabel 4.

Tabel 4. Kategori Interpretasi Skor *System Usability Scale* (SUS)

Skor SUS	<i>Acceptability</i>	<i>Grade Scale</i>	<i>Adjective Rating</i>
$\geq 80,3$	<i>Acceptable</i>	A	<i>Excellent</i>
68 – 80,2	<i>Acceptable</i>	B	<i>Good</i>
51 – 67	<i>Marginal</i>	C-D	<i>Ok</i>
< 51	<i>Not Acceptable</i>	F	<i>Poor</i>

4. PENUTUP

Penelitian ini menciptakan sebuah website sistem manajemen persediaan yang sudah

digunakan di Toko Raja Aki sesuai dengan kebutuhan pengelolaan stok barang. Sistem menggunakan cara First In First Out (FIFO) untuk menentukan urutan mengeluarkan barang berdasarkan waktu barang masuk, sedangkan cara Reorder Point (ROP) digunakan sebagai pedoman untuk memberi pemberitahuan ketika stok barang sudah mencapai batas terendah. Menerapkan kedua metode tersebut membuat proses pengelolaan persediaan lebih teratur, lebih efektif, dan lebih hemat waktu. Pengujian dengan metode Blackbox Testing menunjukkan bahwa semua fitur berjalan sesuai dengan fungsi yang telah direncanakan dan tidak ditemukan kesalahan selama proses pengujian. Selain itu, penilaian menggunakan Skala Kepuasan Pengguna (SUS) terhadap 30 orang responden mendapatkan skor 85, yang masuk dalam kategori Acceptable, mendapatkan nilai Grade A, dan berada pada tingkat penilaian Excellent. Hasil itu menunjukkan bahwa sistem tersebut mudah dipahami, mudah digunakan, dan bisa membantu dalam mengelola stok barang dengan baik. Beberapa saran dari para responden ditimbang dan akan digunakan dalam pengembangan berikutnya, misalnya menambahkan fitur foto produk, memperbaiki fitur pendukung, serta memperbaiki tampilan agar lebih cocok untuk perangkat mobile. Pengembangan ini diharapkan bisa meningkatkan kualitas sistem dan juga menjadi acuan bagi penelitian di bidang sistem persediaan barang berbasis web selanjutnya.

DAFTAR PUSTAKA

- Agboola, F. F., Malgwi, Y. M., Mahmud, M. A., & Oguntoye, J. P. (2022). Development of a web-based platform for automating inventory management of a small and medium enterprise. *Fudma Journal of Sciences*, 6(5), 57–65.
- Alam, M. K., Hossain, M., & Islam, M. R. (2023). Inventory management systems of small and medium enterprises in Bangladesh. *Journal of Global Entrepreneurship Research*, 13(1), 1–12.
- Azizah, D. N., & Nurgiyatna, N. (2021). Pengembangan sistem inventory barang perusahaan dagang berbasis *website* (Studi kasus: CV. Agung Nugraha). *Emitor: Jurnal Teknik Elektro*, 21(1), 42–48.
- Bangor, A., Kortum, P., & Miller, J. (2009). *Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale*. *Journal of Usability Studies*, 4(3), 114–123.
- Booch, G., Rumbaugh, J., & Jacobson, I. (1999). *The unified modeling language user guide*. Addison-Wesley.

- Brooke, J. (1996). SUS: A quick and dirty *usability* scale. Dalam P. W. Jordan, B. Thomas, B. A. Weerdmeester, & I. L. McClelland (Eds.), *Usability evaluation in industry* (hlm. 189–194). Taylor & Francis.
- Chen, P. P. (1976). The entity-relationship model — Toward a unified view of data. *ACM Transactions on Database Systems*, 1(1), 9–36.
- Fowler, M. (2004). *UML distilled: A brief guide to the standard object modeling language* (3rd ed.). Addison-Wesley.
- Garrett, J. J. (2011). *The elements of user experience: User-centered design for the web and beyond* (2nd ed.). New Riders/Pearson Education.
- Gultom, M. M., & Maryam, M. (2020). Sistem informasi penjualan material bangunan pada toko bangunan berkah. *Jurnal Teknik Informatika (JUTIF)*, 1(2), 79–86.
- Holloway, S. (2024). Impact of digital transformation on inventory management: An exploration of supply chain practices. *Preprints.org*.
- Indrayuni, E., et al. (2023). Analisis pengendalian persediaan menggunakan metode *Reorder point* (ROP). *Jurnal Sistem Informasi*.
- Mat, T., Hashim, M., Saad, S., & Ismail, M. B. (2023). Inventory management practices among small and micro businesses during COVID-19 pandemic. *Accounting and Finance Research*, 12(4), 86.
- Myers, G. J. (1979). *The art of software testing*. John Wiley & Sons.
- Nugraha, et al. (2023). Implementasi model *waterfall* dalam pengembangan sistem informasi. *Jurnal Teknologi Informasi*.
- Panigrahi, R. R., Shrivastava, A. K., & Nudurupati, S. S. (2024). Impact of inventory management on SME performance: a systematic review. *International Journal of Productivity and Performance Management*, 73(9), 2901–2925.
- Ristiandi, R. (2021). Implementasi metode FIFO dalam sistem informasi persediaan barang. *Jurnal Teknologi Dan Sistem Informasi*.
- Royce, W. W. (1970). Managing the development of large software systems. *Proceedings of IEEE WESCON*, 26, 1–9.
- Sari, P. P., & Jannah, M. (2022). Pengembangan sistem informasi menggunakan metode *waterfall*. *Jurnal Informatika*.

- Setyawan, R., & Maryam, M. (2021). Sistem informasi penjualan alat elektronik berbasis web pada Toko Mandiri Elektronik Purwantoro. *Jurnal Digital Teknologi Informasi*, 4(1), 1–7.
- Sommerville, I. (2011). *Software engineering* (9th ed.). Addison-Wesley/Pearson Education.
- Syari, R. S., & Supriyono, H. (2020). Sistem informasi berbasis web untuk pengelolaan aset desa. *JPPM (Jurnal Pendidikan dan Pemberdayaan Masyarakat)*, 7(1), 45–56.
- Tanaman, M. T., De Leon, R. G., & Gabayan, R. F. (2023). Web-based inventory management system. *International Journal of Science and Advanced Information Technology*, 12(5), 44–48.
- Widiastuti, H., & Fatmawati, A. (2019). Sistem informasi produksi usaha mikro kecil menengah pada Zahroh Barokah. *Jurnal INSYPRO (Information System and Processing)*, 4(2), 1–7.