

Pengembangan Website Backoffice Aplikasi Kasiran POS Menggunakan Framework Next.js

Agung Trylaksono*¹, Ramadhan Renaldy²

¹Informatika, Universitas Persatuan Guru Republik Indonesia, Kota Semarang, 23670107@upgris.ac.id

²Informatika, Universitas Persatuan Guru Republik Indonesia, Kota Semarang, ramadhanrenaldy@upgris.ac.id

Abstract.

The rapid growth of Point of Sale (POS) technology has driven the need for integrated backoffice management systems to support business operations. This study aims to develop a web-based backoffice system for the Kasiran POS application using the Next.js framework, which communicates with a Node.js and Express.js backend through RESTful API. The system was developed using the Agile method with an incremental approach, covering requirements analysis, system design, implementation, testing, and review cycles. The developed backoffice system successfully integrates 19 functional modules including dashboard, transaction history, financial management (accounts, income, expenses), product management (products, extra products, categories, discounts, taxes), operational management (employees, customers, tables, sales types, company, outlets), and sales reports. The system applies server-side rendering to ensure real-time data presentation and responsive interface across various screen sizes. Testing results confirm that all features function properly and meet user requirements. The implementation of Next.js proves effective in producing a structured, maintainable, and responsive backoffice system that meets professional software development standards.

Keywords: Next.js, RESTful API, Backoffice, Point of Sale, Agile, Node.js

Abstrak

Perkembangan teknologi Point of Sale (POS) yang pesat mendorong kebutuhan sistem manajemen backoffice yang terintegrasi untuk mendukung operasional bisnis. Penelitian ini bertujuan mengembangkan sistem backoffice berbasis web untuk aplikasi Kasiran POS menggunakan framework Next.js yang berkomunikasi dengan backend Node.js dan Express.js melalui RESTful API. Pengembangan dilakukan menggunakan metode Agile dengan pendekatan inkremental yang mencakup analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan evaluasi pada setiap siklusnya. Sistem backoffice yang dikembangkan berhasil mengintegrasikan 19 modul fungsional meliputi dashboard, riwayat transaksi, manajemen keuangan (rekening, uang masuk, uang keluar), manajemen produk (produk, extra produk, kategori, diskon, pajak), manajemen operasional (karyawan, pelanggan, meja, tipe penjualan, perusahaan, outlet), serta laporan penjualan. Sistem menerapkan mekanisme server-side rendering untuk memastikan penyajian data secara real-time dan antarmuka yang responsif di berbagai ukuran layar. Hasil pengujian menunjukkan seluruh fitur berjalan sesuai kebutuhan pengguna. Penerapan Next.js terbukti efektif dalam menghasilkan sistem backoffice yang terstruktur, mudah dipelihara, dan responsif sesuai standar pengembangan perangkat lunak profesional.

Kata Kunci: Next.js, RESTful API, Backoffice, Point of Sale, Agile, Node.js

1. Pendahuluan

Perkembangan teknologi informasi yang pesat dalam beberapa tahun terakhir telah membawa perubahan signifikan dalam pengelolaan operasional bisnis secara digital. Pelaku usaha dari berbagai skala, mulai dari usaha kecil hingga menengah, kini semakin membutuhkan sistem yang mampu membantu mereka dalam mengelola transaksi, produk, keuangan, dan laporan bisnis secara terintegrasi dan efisien. Salah satu solusi yang banyak digunakan adalah sistem kasir digital berbasis Point of Sale (POS). Pasar global sistem POS tercatat senilai USD 29,02 miliar pada tahun 2023 dan diproyeksikan tumbuh hingga USD

110,22 miliar pada tahun 2032 dengan CAGR sebesar 16,1%, didorong oleh meningkatnya adopsi solusi pembayaran digital di berbagai sektor bisnis [1]. Penerapan sistem POS terbukti mampu meningkatkan profitabilitas usaha sekaligus mendorong percepatan digitalisasi pada sektor UMKM [2].

Sistem POS modern tidak hanya berfungsi sebagai alat transaksi di titik penjualan, tetapi juga dilengkapi sistem manajemen backoffice yang memungkinkan pengelola usaha memantau dan mengendalikan seluruh operasional bisnisnya dari satu platform terpusat. Website backoffice berperan penting sebagai antarmuka administrasi yang menghubungkan data transaksi dengan kebutuhan manajerial seperti pengelolaan produk, laporan penjualan, manajemen karyawan, dan pengaturan keuangan. Tanpa sistem backoffice yang baik, pengelolaan data bisnis akan menjadi tidak efisien dan rentan terhadap kesalahan [3]. Perkembangan teknologi berbasis web telah mendorong berbagai lembaga dan perusahaan untuk mengadopsi sistem informasi yang mampu mengintegrasikan seluruh proses bisnis dalam satu platform yang dapat diakses secara real-time [4].

Beberapa penelitian terdahulu telah membahas pengembangan sistem berbasis web untuk kebutuhan manajemen bisnis. Ramadhani dkk. (2025) dalam penelitiannya berjudul "Pengembangan Aplikasi Point of Sale (POS) Berbasis Website Menggunakan Metodologi Agile untuk Manajemen Penjualan Toko Kue XYZ" berhasil merancang dan membangun sistem POS yang mampu mendukung transaksi penjualan, pengelolaan stok, dan pembuatan laporan secara real-time menggunakan metodologi Agile dengan pendekatan Scrum yang berjalan secara iteratif dan adaptif terhadap perubahan kebutuhan [5]. Penelitian tersebut menyimpulkan bahwa penerapan metodologi Agile terbukti efektif dalam pengembangan sistem POS berbasis web secara terstruktur dan responsif. Sementara itu, Kesuma dkk. (2024) dalam penelitiannya berjudul "Implementasi Next.js, TypeScript, dan Tailwind CSS untuk Pengembangan Aplikasi Frontend Sistem Inventory Perusahaan APAR (Studi Kasus: CV Indoka Surya Jaya)" berhasil membangun aplikasi frontend sistem inventory menggunakan Next.js dan menyimpulkan bahwa implementasi Next.js mampu menghasilkan antarmuka yang responsif, modern, serta meningkatkan efisiensi operasional secara signifikan [6].

Berdasarkan kedua penelitian tersebut, dapat disimpulkan bahwa kombinasi framework Next.js dan metodologi Agile merupakan pendekatan yang tepat dan telah terbukti efektif dalam pengembangan sistem informasi berbasis web. Next.js sebagai framework berbasis React.js yang dikembangkan oleh Vercel mendukung beberapa mekanisme rendering seperti Server-Side Rendering (SSR), Static Site Generation (SSG), dan Client-Side Rendering (CSR), serta menyediakan fitur file-based routing yang memungkinkan pengelolaan rute halaman secara lebih terstruktur tanpa konfigurasi tambahan [7]. Keunggulan-keunggulan ini menjadikan Next.js pilihan yang tepat untuk pengembangan sistem backoffice yang membutuhkan performa tinggi dan tampilan antarmuka yang konsisten di berbagai halaman.

Mengacu pada hal tersebut, penelitian ini mengembangkan website backoffice untuk aplikasi Kasiran POS menggunakan framework Next.js yang berkomunikasi dengan backend berbasis Node.js dan Express.js melalui RESTful API. Kasiran POS merupakan produk aplikasi kasir digital berbasis Android yang dirancang untuk membantu pelaku usaha dalam mengelola transaksi, produk, stok, dan operasional bisnis secara efisien [8]. Pengembangan dilakukan dengan metode Agile menggunakan pendekatan iteratif, di mana setiap fitur dikerjakan secara inkremental mulai dari analisis kebutuhan, perancangan, implementasi, hingga pengujian pada setiap siklusnya. Penelitian ini bertujuan menghasilkan sistem backoffice yang fungsional, responsif, dan sesuai dengan standar pengembangan perangkat lunak profesional.

2. Metode

Penelitian ini dilakukan untuk mengembangkan website backoffice aplikasi Kasiran POS menggunakan framework Next.js yang terhubung dengan backend berbasis Node.js dan Express.js. Pengembangan perangkat lunak dilakukan menggunakan metode Agile karena bersifat iteratif dan inkremental sehingga mampu mengakomodasi perubahan kebutuhan yang sering terjadi selama proses pengembangan berlangsung [9].



Gambar 2. 1 Siklus Metode Pengembangan Sistem Agile

Metode Agile digunakan sebagai pendekatan dalam pengembangan sistem pada penelitian ini. Berbeda dengan model Waterfall yang bersifat sekuensial, Agile menerapkan siklus pengembangan yang berulang, terdiri atas enam tahapan, yaitu Requirements, Design, Development, Testing, Deployment, dan Review. Keenam tahapan tersebut berjalan secara memutar (iteratif) untuk setiap fitur yang dikerjakan, sehingga setiap fitur dapat segera dievaluasi dan disempurnakan sebelum pengembangan berlanjut ke fitur berikutnya [9].

a. Requirements

Tahap requirements dilakukan untuk mengidentifikasi kebutuhan pengguna dan menentukan spesifikasi sistem yang akan dikembangkan, berdasarkan hasil observasi dan wawancara dengan pembimbing lapangan. Hasil analisis menunjukkan bahwa sistem harus mampu mendukung pengelolaan dashboard, riwayat transaksi, data keuangan, data produk, data karyawan dan pelanggan, konfigurasi penjualan, manajemen perusahaan dan outlet, serta laporan penjualan, dengan total 19 fitur yang harus dikembangkan pada sisi frontend backoffice.

b. Design

Tahap design dilakukan untuk menggambarkan struktur dan alur kerja sistem sebelum masuk ke tahap pengkodean. Perancangan menggunakan Unified Modeling Language (UML) yang terdiri dari Use Case Diagram untuk memetakan interaksi Owner dengan seluruh fitur sistem, Activity Diagram untuk menggambarkan alur kerja setiap modul, dan Class Diagram untuk merancang struktur data serta relasi antar entitas seperti Perusahaan, Outlet, Karyawan, Produk, dan Transaksi [10].

c. Development

Tahap development dilakukan dengan menerjemahkan hasil perancangan ke dalam bentuk halaman dan komponen antarmuka menggunakan framework Next.js. Next.js dipilih karena mendukung server-side rendering dan file-based routing yang menghasilkan struktur kode lebih terorganisir dibandingkan React.js murni [11]. Setiap halaman dibangun dengan menerapkan konsep reusable components seperti tabel data, modal form, dan komponen filter tanggal, sehingga mengurangi duplikasi kode dan menjaga konsistensi tampilan di seluruh halaman backoffice [12]. Pengembangan dilakukan secara bertahap per fitur, dimulai dari data dummy sebelum diintegrasikan dengan data sesungguhnya melalui RESTful API yang dikembangkan tim backend menggunakan Node.js dan Express.js.

d. Testing

Tahap testing dilakukan menggunakan metode Black Box Testing, yaitu pengujian yang berfokus pada fungsionalitas sistem tanpa memperhatikan struktur kode internal. Pengujian dilakukan secara mandiri menggunakan browser Google Chrome beserta Developer Tools untuk memeriksa kesesuaian tampilan pada berbagai ukuran layar, kebenaran interaksi komponen, serta pemantauan permintaan jaringan (network request) guna memastikan data dari API backend sesuai dengan format yang diharapkan. Pengujian juga dilakukan bersama mentor untuk memverifikasi kesesuaian fungsi sistem terhadap kebutuhan yang telah ditetapkan. Apabila ditemukan ketidaksesuaian, dilakukan penyesuaian pada implementasi sistem dan pengujian diulang hingga fitur dinyatakan sesuai dengan rancangan.

e. Deployment

Tahap deployment dilakukan untuk menerapkan fitur yang telah selesai diimplementasikan dan diuji ke lingkungan yang dapat diakses secara nyata. Setiap fitur yang telah dinyatakan sesuai pada tahap pengujian kemudian di-push ke repository menggunakan Git, dilanjutkan dengan proses build pada framework Next.js sebelum dijalankan pada environment yang telah disediakan. Proses deployment dilakukan secara bertahap per fitur sejalan dengan pendekatan inkremental pada metode Agile, sehingga setiap fitur yang sudah siap dapat segera diintegrasikan tanpa harus menunggu seluruh modul selesai dikembangkan.

f. Review

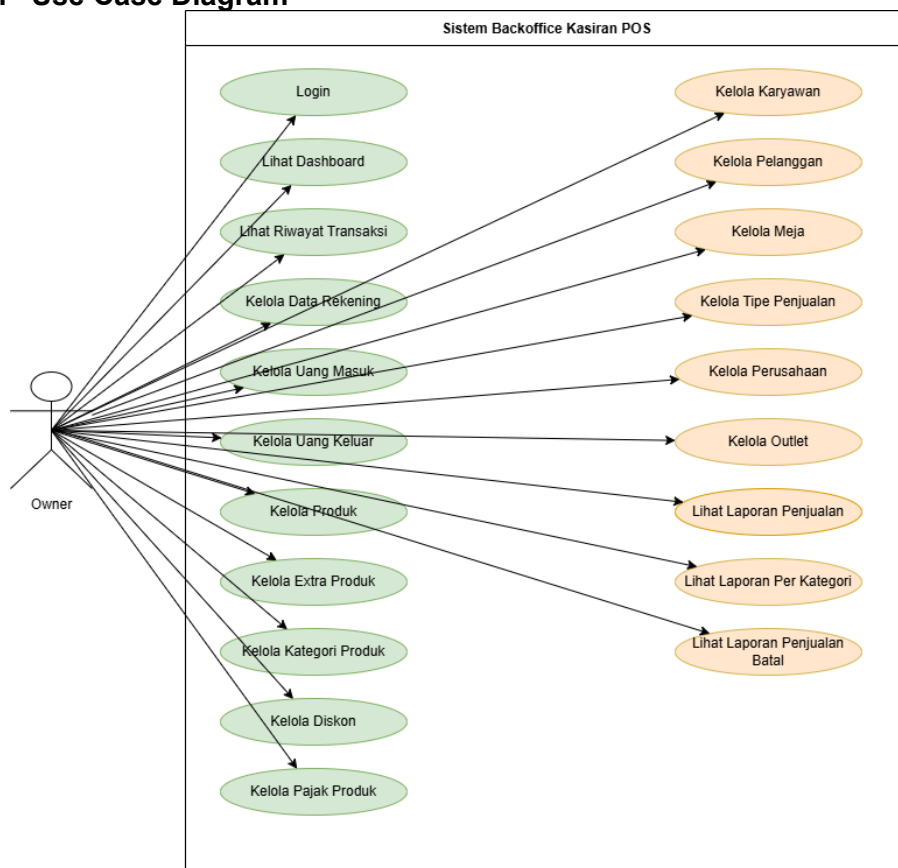
Tahap review dilakukan untuk mengevaluasi hasil deployment setiap fitur bersama pembimbing lapangan, guna memastikan kesesuaian fungsi sistem terhadap kebutuhan yang telah ditetapkan di awal siklus. Proses review melibatkan pengecekan langsung terhadap fitur yang telah berjalan pada environment, baik dari sisi fungsionalitas maupun kesesuaian tampilan antarmuka. Apabila ditemukan ketidaksesuaian atau masukan perbaikan dari pembimbing, catatan tersebut menjadi dasar untuk siklus iterasi berikutnya, sehingga fitur dapat disempurnakan secara berkelanjutan sebelum pengembangan berlanjut ke fitur lain [13].

3. Hasil dan Pembahasan

3.1. Penyajian Hasil

Website backoffice Kasiran POS dirancang untuk mendukung pengelolaan operasional bisnis secara terpusat dan real-time, dengan Owner sebagai aktor utama yang memiliki akses penuh terhadap seluruh modul — mulai dari dashboard, riwayat transaksi, keuangan, produk, data operasional, hingga laporan penjualan untuk mendukung pengambilan keputusan bisnis.

3.1.1 Use Case Diagram



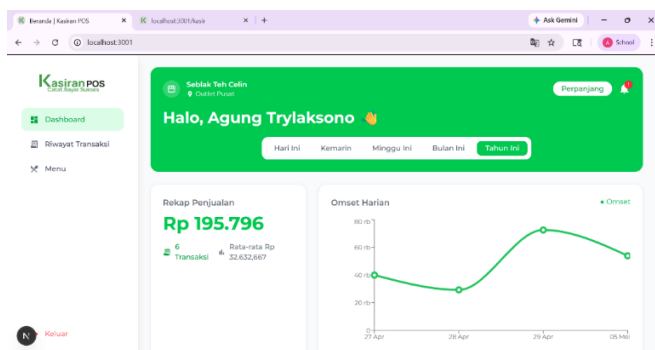
Gambar 3. 1 Use Case Diagram Backoffice Kasiran POS

Gambar 3.1 menggambarkan interaksi antara Owner dengan seluruh fitur yang tersedia pada sistem backoffice Kasiran POS. Diagram ini menunjukkan bahwa Owner memiliki akses penuh terhadap 19 use case utama yang mencakup seluruh aspek operasional bisnis. Use case tersebut meliputi Login untuk autentikasi pengguna, Lihat Dashboard untuk memantau performa penjualan secara real-time, dan Lihat Riwayat Transaksi untuk meninjau seluruh transaksi yang telah terjadi. Pada aspek keuangan, Owner dapat melakukan Kelola Data Rekening serta Kelola Uang Masuk dan Uang Keluar untuk mencatat arus kas bisnis. Pada aspek manajemen produk, tersedia use case Kelola Produk, Extra Produk, dan Kategori Produk, serta Kelola Diskon dan Pajak Produk untuk mengatur ketentuan transaksi. Selain itu, Owner juga dapat melakukan Kelola Karyawan dan Pelanggan, Kelola Meja dan Tipe Penjualan, serta Kelola Perusahaan dan Outlet untuk mendukung data operasional dan administratif. Sebagai penutup, sistem menyediakan use case Lihat Laporan Penjualan, Laporan Per Kategori, dan Laporan Penjualan Batal yang memungkinkan Owner mengevaluasi performa bisnis secara menyeluruh dari satu platform terpusat.

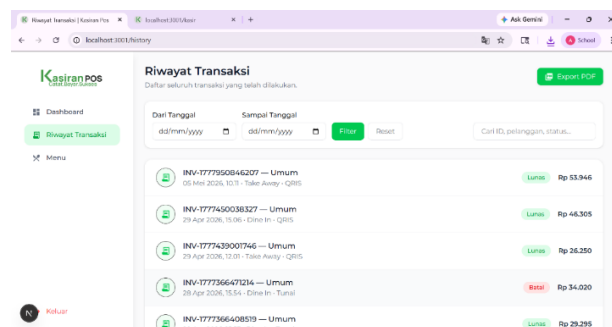
3.1.2 Class Diagram



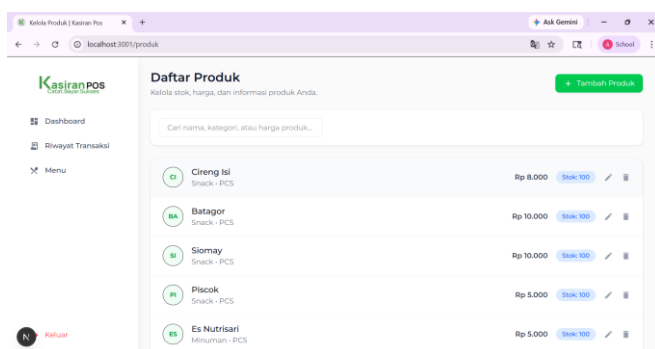
3.1.1 Implementasi Sistem



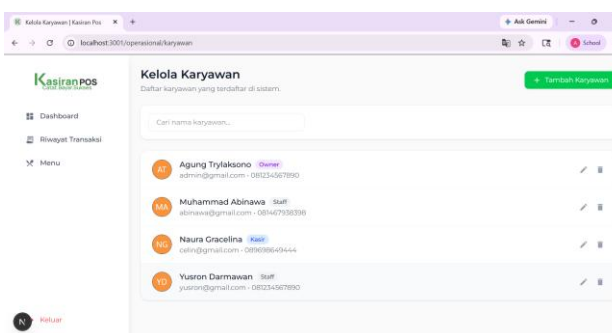
Gambar 3. 3 Halaman Dashboard



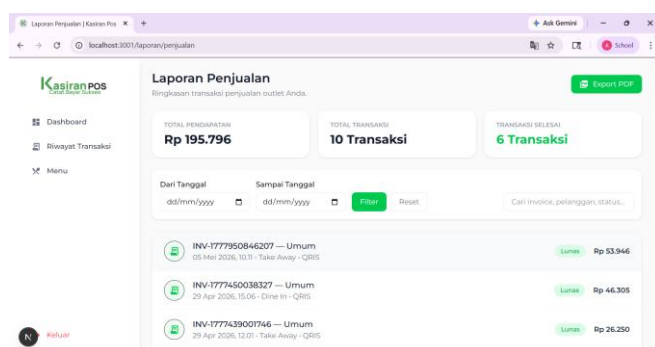
Gambar 3. 4 Halaman Riwayat Transaksi



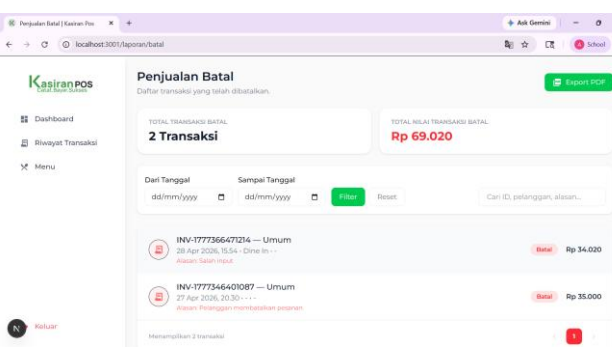
Gambar 3. 5 Halaman Produk



Gambar 3. 6 Halaman Karyawan



Gambar 3. 7 Halaman Laporan Penjualan



Gambar 3. 8 Halaman Laporan Penjualan Batal

Gambar 3.3 menampilkan halaman dashboard yang memuat rekap omset, jumlah transaksi, grafik omset harian, produk terlaris, serta rekap tipe penjualan dan metode pembayaran secara real-time. Gambar 3.4 menunjukkan halaman riwayat transaksi dengan filter tanggal, pencarian, modal detail, dan ekspor PDF. Gambar 3.5 menampilkan halaman pengelolaan produk untuk menambah, mengedit, dan menghapus data produk beserta kategori dan harga jualnya. Gambar 3.6 menunjukkan halaman pengelolaan karyawan untuk mendaftarkan karyawan baru beserta identitas, peran, dan akun login. Gambar 3.7 menampilkan halaman laporan penjualan berisi ringkasan pendapatan dan jumlah transaksi per periode, dilengkapi ekspor PDF, sementara Gambar 3.8 menunjukkan laporan penjualan batal yang mencatat transaksi dibatalkan beserta alasan dan karyawan yang memprosesnya. Seluruh 19 fitur yang ditargetkan berhasil dikembangkan, berfungsi sesuai rancangan, dan terhubung secara real-time dengan backend melalui RESTful API.

3.2. Pembahasan

Hasil penelitian menunjukkan bahwa website backoffice Kasiran POS berhasil dikembangkan menggunakan framework Next.js dengan seluruh fitur yang ditargetkan berfungsi sebagaimana mestinya. Pemilihan Next.js terbukti efektif karena mekanisme server-side rendering memungkinkan data dari API backend sudah tersedia saat halaman pertama kali dimuat, sementara fitur file-based routing mengurangi kompleksitas konfigurasi dibandingkan React.js murni yang memerlukan library tambahan seperti React Router [14]. Penerapan konsep reusable components pada komponen tabel data, modal form, dan filter tanggal juga terbukti mengurangi duplikasi kode dan menjaga konsistensi tampilan di seluruh halaman backoffice.

Penelitian ini memiliki kesamaan dengan penelitian Ramadhani dkk. yang menerapkan metodologi Agile dalam pengembangan sistem POS berbasis website, di mana pendekatan iteratif terbukti efektif mengakomodasi perubahan kebutuhan selama proses pengembangan. Tahap review yang dilakukan secara berkelanjutan pada setiap siklus, sebagaimana dijelaskan pada bagian Metode, memungkinkan setiap fitur dievaluasi secara cepat sehingga koreksi dapat dilakukan sebelum masuk ke pengembangan fitur berikutnya.

Proses integrasi antara frontend Next.js dan backend Node.js serta Express.js berjalan dengan baik melalui RESTful API, karena setiap operasi data dipetakan secara konsisten ke metode HTTP GET, POST, PUT, dan DELETE [15]. Penggunaan platform Stoplight sebagai dokumentasi API turut mempercepat proses integrasi, sehingga pengembangan frontend dan backend dapat dilakukan secara paralel tanpa banyak kendala koordinasi. Seluruh 19 fitur yang ditargetkan berhasil dikembangkan dan terhubung secara real-time dengan backend melalui RESTful API. Hasil pengujian menggunakan metode Black Box Testing menunjukkan bahwa seluruh fitur berfungsi sesuai dengan skenario yang diharapkan, mulai dari autentikasi, pengelolaan data, hingga ekspor laporan.

Meskipun demikian, terdapat beberapa keterbatasan pada sistem yang dikembangkan. Mekanisme autentikasi belum menerapkan token berbasis JSON Web Token (JWT) dengan refresh token, sehingga keamanan sesi pengguna masih dapat ditingkatkan. Halaman dengan volume data besar juga belum dioptimalkan menggunakan teknik lazy loading atau infinite scroll, serta sistem belum memiliki notifikasi real-time atas aktivitas transaksi terbaru. Keterbatasan-keterbatasan tersebut menjadi area pengembangan lanjutan pada iterasi berikutnya untuk meningkatkan keamanan dan performa sistem secara keseluruhan.

4. Kesimpulan

Penelitian ini berhasil merancang dan mengembangkan website backoffice aplikasi Kasiran POS menggunakan framework Next.js yang terhubung dengan backend berbasis Node.js dan Express.js melalui RESTful API. Sistem yang dikembangkan mampu mengintegrasikan seluruh kebutuhan operasional bisnis dalam satu platform terpusat, yang mencakup dashboard, riwayat transaksi, pengelolaan keuangan, pengelolaan produk, data operasional seperti karyawan dan pelanggan, konfigurasi penjualan, manajemen perusahaan dan outlet, serta berbagai laporan penjualan. Pengembangan dilakukan menggunakan metode Agile yang bersifat iteratif melalui enam tahapan utama, yaitu requirements, design, development, testing, deployment, dan review, sehingga setiap fitur dapat di-deploy secara bertahap dan dievaluasi secara berkelanjutan sesuai arahan pembimbing lapangan sebelum pengembangan berlanjut ke fitur berikutnya. Hasil pengujian menggunakan metode Black Box Testing menunjukkan bahwa seluruh 19 fitur yang ditargetkan berhasil diimplementasikan dan berfungsi sesuai dengan kebutuhan yang telah ditetapkan, dengan tampilan antarmuka yang konsisten dan responsif di berbagai ukuran layar.

Hasil penelitian menunjukkan kombinasi Next.js dan metode Agile efektif diterapkan dalam pengembangan sistem backoffice berbasis web, baik dari segi kecepatan maupun kualitas antarmuka. Integrasi RESTful API yang didokumentasikan melalui Stoplight turut mempercepat pengembangan karena memungkinkan tim frontend dan backend bekerja paralel. Pengembangan selanjutnya dapat menambahkan autentikasi JWT, teknik lazy loading

untuk data berskala besar, serta fitur notifikasi real-time guna meningkatkan kualitas dan keandalan sistem.

5. Referensi

- [1] Fortune Business Insights, "Point of Sale Market Size, Share & Industry Analysis," Fortune Business Insights.
- [2] N. I. Andriasari, "Aplikasi Point of Sale untuk Meningkatkan Profitabilitas dan Digitalisasi UMKM," *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, pp. 89–94, 2024.
- [3] R. Awang Sudrajat and E. Aryanny, "Perancangan Sistem Informasi Manajemen Toko (Point Of Sales) Berbasis Website di UMKM Hijab Nafisa," *Jurnal Manajemen Riset Inovasi*, vol. 1, no. 2, pp. 1–19, 2023.
- [4] J. Tarigan, M. Gultom, and T. Willay, "Perancangan Sistem Informasi Manajemen Penjualan Produk Berbasis Web pada PT Harapan Tani Permai," *Jurnal Teknologi Sistem Informasi*, vol. 5, no. 2, pp. 82–97, 2024, doi: 10.35957/jtsi.v5i2.8666.
- [5] I. Ramadhani, "Pengembangan Aplikasi Point of Sale (POS) Berbasis Website Menggunakan Metodologi Agile untuk Manajemen Penjualan Toko Kue XYZ," *ICIT Journal*, 2025, [Online]. Available: <https://ejournal.raharja.ac.id/icit/article/view/4220>
- [6] K. A. B. W. Kesuma, I. N. Y. A. Wijaya, and I. G. J. E. Putra, "Implementasi Next.js, Typescript, Dan Tailwind CSS Untuk Pengembangan Aplikasi Frontend Sistem Inventory Perusahaan APAR (Studi Kasus: CV Indoka Surya Jaya)," *Jikom: Jurnal Informatika dan Komputer*, vol. 14, no. 2, pp. 95–108, 2024, doi: 10.55794/jikom.v14i2.195.
- [7] R. Hanafi, A. Haq, and N. Agustin, "Comparison of Web Page Rendering Methods Based on Next.js Framework Using Page Loading Time Test," *Teknika*, vol. 13, no. 1, pp. 102–108, 2024, doi: 10.34148/teknika.v13i1.769.
- [8] Ribar ID, "Kasiran - Reliable Cashier App," 2025. [Online]. Available: <https://play.google.com/store/apps/details?id=com.Kasiran.AplikasiKasir>
- [9] C. Ramadhan, M. A. Senubekti, and D. Amalia, "Penerapan Metodologi Agile dalam Pengembangan Perangkat Lunak," *Router: Jurnal Teknik Informatika dan Terapan*, vol. 3, no. 2, pp. 10–15, 2025, doi: 10.62951/router.v3i2.411.
- [10] K. Nistrina and L. Sahidah, "Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Penerimaan Siswa Baru di SMK Marga Insan Kamil," *J-SIKA: Jurnal Sistem Informasi*, vol. 4, no. 1, pp. 17–23, 2022.
- [11] Vercel, "Next.js Documentation: App Router," 2023.
- [12] K. Juan and S. Budi, "Pengembangan Menu Digital Menggunakan ReactJs Implementasi Hasil Belajar Studi Independen di Frontend Engineering Program Ruangguru CAMP (Career Acceleration Bootcamp)," *Jurnal Strategi*, vol. 5, no. 1, pp. 130–138, 2023.

- [13] S. H. Nova, A. P. Widodo, and B. Warsito, “Analisis Metode Agile pada Pengembangan Sistem Informasi Berbasis Website: Systematic Literature Review,” *Techno.Com*, vol. 21, no. 1, pp. 139–148, 2022, doi: 10.33633/tc.v21i1.5659.
- [14] M. Olivia, “Implementasi Dan Evaluasi REST API Dalam Integrasi Sistem Lintas Platform,” *Jurnal Komputer*, vol. 3, no. 2, pp. 43–48, 2025.
- [15] S. Adha, “Implementasi Web Service Menggunakan RESTful API untuk Integrasi Sistem,” *JMIK: Jurnal Mahasiswa Ilmu Komputer*, vol. 5, no. 2, pp. 101–108, 2024, [Online]. Available: <https://scholar.ummetro.ac.id/index.php/IlmuKomputer/article/download/7286/2887/>