


Perancangan Sistem Informasi Penjualan Dan Stok Berbasis Web *Multi-Cabang* Pada Mochi *Petshop*

DOI: <http://dx.doi.org/10.35889/jutisi.v15i3.3743>

Creative Commons License 4.0 (CC BY – NC) 

Nabil Syawaludin Prima^{1*}, Triana Elizabeth²

Sistem Informasi, Universitas Multi Data Palembang, Palembang, Indonesia

*e-mail *Corresponding Author*: nabilsyawaludin108@gmail.com

Abstract

This study focuses on Mochi Petshop, a business for pet equipment managing multiple physical stores in Palembang. At present, cashier recording and inter-store inventory management are not yet computerized, often triggering slow data recapitulation, item quantity discrepancies, and obstacles in monitoring daily performance. This study aims to design a multi-branch, web-based sales and inventory information system using the Laravel framework. This research adopts the Rapid Application Development (RAD) model as its development approach. The workflow of this method encompasses four primary phases: initial specification planning, interface design, program coding process, and full operational transition. The developed system supports the management of sales transactions, stock-in, stock-out, inter-branch stock transfers, product returns, and real-time reporting. The Black Box Testing results indicate that all system features operate according to the specified requirements, proving that the system is capable of enhancing data management effectiveness and operational efficiency at Mochi Petshop.

Keywords: Sales information system; Inventory; Multi-branch; Laravel; RAD

Abstrak

Penelitian ini mengambil studi kasus pada Mochi Petshop, sebuah bisnis perlengkapan hewan yang mengelola sejumlah gerai fisik di Palembang. Saat ini, pencatatan kasir dan manajemen persediaan antartoko belum terkomputerisasi, sehingga sering memicu rekapitulasi data yang lambat, selisih jumlah barang, serta kendala dalam pengawasan kinerja harian. Penelitian ini bertujuan merancang sistem informasi penjualan dan stok berbasis web *multi-cabang* menggunakan *framework Laravel*. Penelitian ini mengadopsi model kerja *Rapid Application Development (RAD)* sebagai pendekatan pengembangannya. Alur metode ini mencakup empat fase utama, yakni: perencanaan spesifikasi awal, perancangan antarmuka, proses pengkodean program, serta transisi operasional penuh. Sistem yang dibangun mendukung pengelolaan transaksi penjualan, stok masuk, stok keluar, perpindahan barang antar cabang, retur barang, dan laporan secara *real-time*. Hasil pengujian *Black Box Testing* menunjukkan seluruh fitur sistem berjalan sesuai kebutuhan sehingga sistem mampu meningkatkan efektivitas pengelolaan data dan efisiensi operasional pada Mochi *Petshop*.

Kata kunci: Sistem informasi penjualan; Persediaan barang; Multi-cabang; Laravel; RAD

1. Pendahuluan

Perkembangan teknologi informasi mendorong pelaku usaha untuk mengelola proses bisnis secara lebih cepat, akurat, dan terintegrasi. Secara konseptual, sistem informasi merupakan kesatuan berbagai elemen terintegrasi yang bertugas menghimpun, memproses, menyimpan, sekaligus menyalurkan data krusial guna menunjang tata kelola harian dan penentuan kebijakan sebuah institusi [1]. Dalam konteks penjualan dan persediaan, sistem berbasis web dapat mempercepat pencatatan transaksi, mempermudah pemantauan stok, serta mengurangi ketergantungan pada rekapitulasi manual karena data dapat diakses dari berbagai lokasi [2]. Kebutuhan tersebut semakin penting pada usaha kecil dan menengah yang memiliki transaksi harian tinggi serta variasi produk yang cukup banyak.

Mochi Petshop merupakan usaha penjualan kebutuhan hewan peliharaan yang berdiri sejak tahun 2019 dan memiliki tiga lokasi operasional di Kota Palembang. Produk yang dijual meliputi makanan kucing dan anjing, vitamin, obat hewan, pasir kucing, kandang, *litter box*, serta berbagai aksesoris hewan. Berdasarkan hasil observasi, proses transaksi penjualan dan pencatatan stok pada setiap cabang masih dilakukan menggunakan buku dan file sederhana. Dengan jumlah produk sekitar 200 item dan transaksi pada toko pusat yang dapat mencapai 100 sampai 150 transaksi per hari, pencatatan manual menimbulkan kendala dalam pencarian data, penyusunan laporan, dan pengendalian stok.

Permasalahan tersebut berkaitan langsung dengan konsep sistem informasi persediaan. Dalam konteks inventaris, persediaan merujuk pada sekumpulan barang fisik yang menuntut pengawasan ketat lewat dokumentasi sirkulasi masuk-keluar serta pemantauan sisa kuantitas, demi menjaga stabilitas pelayanan toko [3]. Pada *Mochi Petshop*, pencatatan stok yang masih tersebar di setiap cabang membuat owner sulit mengetahui ketersediaan barang secara menyeluruh. Selain itu, retur pelanggan dan perpindahan barang antar cabang belum terdokumentasi secara sistematis, sehingga riwayat perubahan stok tidak selalu dapat ditelusuri secara akurat.

Konsep sistem informasi *multi-cabang* menjadi relevan karena objek penelitian memiliki lebih dari satu lokasi operasional. Sistem *multi-cabang* dirancang untuk mengintegrasikan data dari beberapa cabang dalam satu *platform* terpusat sehingga manajemen dapat memantau kinerja, mengontrol operasional, dan memperoleh informasi secara lebih efisien [4]. Penelitian Widiastuti dkk. [5] juga menunjukkan bahwa rancangan *point of sales* terintegrasi dapat menghubungkan data transaksi dan stok antar cabang secara *real-time*. Temuan tersebut menegaskan bahwa integrasi cabang merupakan kebutuhan penting bagi usaha yang tidak lagi beroperasi pada satu titik layanan.

Sejumlah penelitian terdahulu telah mengkaji sistem penjualan dan stok berbasis web. Tinjauan terhadap beberapa studi terdahulu memperkuat urgensi penelitian ini. Riset yang dilakukan oleh Mulyana dan Rusmawan [6] serta Auditira dkk. [7] membuktikan bahwa aplikasi kasir digital efektif dalam mengefisienkan rekapitulasi pembayaran dan pemantauan barang, meski fokusnya hanya pada gerai tunggal. Di sisi lain, Salsabila dan Andryana [8] mengeksplorasi pengelolaan inventaris petshop menggunakan metode FEFO guna meminimalisir produk yang telah melewati masa pakainya. Sementara itu, kajian Afriani dan Gusriyanti [9] menyoroti peningkatan validitas data melalui digitalisasi layanan perawatan hewan.

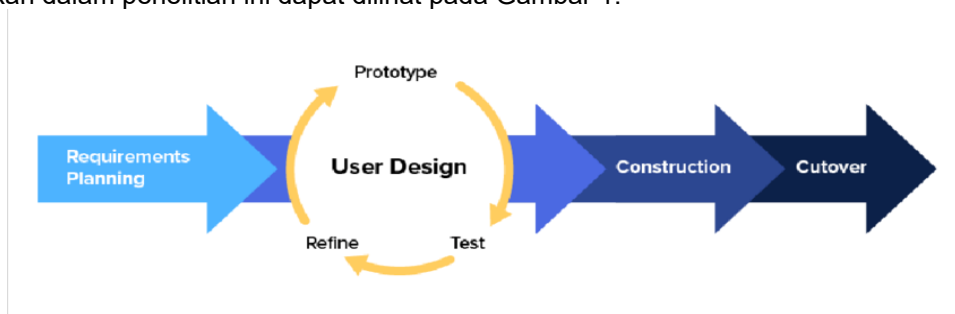
Pada aspek rekayasa perangkat lunak, pemilihan Laravel didasari oleh implementasi pola *Model-View-Controller (MVC)*. Pola arsitektur ini memecah struktur sistem menjadi bagian pemrosesan data, antarmuka visual, dan logika pengendali, yang bermuara pada kemudahan pemeliharaan program di masa mendatang [10]. Penggunaan *Laravel* juga dinilai dapat mempercepat pembangunan aplikasi berbasis web melalui dukungan *routing*, *autentikasi*, dan integrasi basis data [11]. Pendekatan *Rapid Application Development (RAD)* diimplementasikan dengan pertimbangan bahwa model ini mendukung siklus pembuatan perangkat lunak yang lebih singkat, berulang secara kontinu, dan mengutamakan masukan langsung dari calon pengguna [12]. Widodo dkk. [13] menyatakan bahwa RAD dapat meningkatkan fleksibilitas pengembangan sistem berbasis web, sedangkan Fahmi Rosyadi dan Sembiring [14] menunjukkan bahwa RAD sesuai untuk pengembangan sistem *POS* terintegrasi karena kebutuhan pengguna dapat disesuaikan selama proses pembangunan.

Berdasarkan uraian tersebut, masih terdapat celah penelitian pada pengembangan sistem informasi penjualan dan stok yang secara khusus menyesuaikan kebutuhan *petshop multi-cabang*. Penelitian sebelumnya umumnya berfokus pada satu lokasi, pengelolaan stok secara umum, atau rancangan integrasi cabang tanpa penekanan pada retur barang, transfer antar cabang, dan pembagian proses kerja berdasarkan peran pengguna. Oleh karena itu, penelitian ini bertujuan untuk merancang dan membangun sistem informasi penjualan dan stok berbasis web *multi-cabang* pada *Mochi Petshop*. Kebaruan penelitian terletak pada integrasi aktivitas operasional *petshop* dalam satu platform terpusat yang mencakup transaksi penjualan, stok masuk, stok keluar, retur barang, transfer barang antar cabang, *shift* kasir, diskon, serta laporan penjualan dan stok secara *real-time*.

2. Metodologi

Penelitian ini mengadopsi model *kerja Rapid Application Development (RAD)* sebagai pendekatan pengembangannya. Alur metode ini mencakup empat fase utama, yakni:

perencanaan spesifikasi awal, perancangan antarmuka, proses pengkodean program, serta transisi operasional penuh. Objek penelitian adalah Mochi *Petshop* Palembang. Pengumpulan data dilakukan melalui observasi langsung, wawancara dengan *Owner* dan kepala cabang, serta studi dokumentasi terhadap proses bisnis yang berjalan. Adapun diagram model RAD yang digunakan dalam penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Metode *Rapid Application Development* (RAD)

2.1 Requirement Planning

Tahap *requirement* planning dilakukan untuk mengidentifikasi permasalahan dan kebutuhan sistem berdasarkan proses operasional yang berjalan pada Mochi *Petshop*. Pada tahap ini, peneliti melakukan observasi terhadap proses penjualan, pengecekan stok, pengelolaan barang masuk, perpindahan barang antar cabang, pencatatan retur, serta penyusunan laporan. Wawancara juga dilakukan kepada *Owner* dan kepala cabang untuk mengetahui kendala utama yang dihadapi dan harapan terhadap sistem yang akan dibangun.

Hasil identifikasi menunjukkan bahwa sistem yang dibutuhkan harus mampu mengelola data pengguna, cabang, kategori, satuan, *supplier*, produk, stok masuk, stok keluar, transaksi penjualan, retur barang, transfer barang antar cabang, *shift* kasir, diskon, dan laporan. Kebutuhan nonfungsional yang diperhatikan meliputi kemudahan penggunaan, pembagian hak akses, penyimpanan data terpusat, serta kemampuan sistem menampilkan informasi secara *real-time* sesuai data yang dimasukkan oleh pengguna.

Tabel 1. Kebutuhan Fungsional Sistem

No.	Kebutuhan Fungsional	Aktor Utama	Keluaran Sistem
1	Autentikasi pengguna berdasarkan <i>Username</i> dan password	<i>Owner</i> , kepala cabang, kasir	Pengguna masuk ke <i>dashboard</i> sesuai <i>role</i>
2	Pengelolaan data pengguna dan cabang	<i>Owner</i>	Data pengguna, <i>role</i> , dan cabang tersimpan
3	Pengelolaan kategori, satuan, <i>supplier</i> , dan produk	Kepala cabang	<i>Data master</i> produk tersedia untuk transaksi dan stok
4	Pencatatan stok masuk dan stok keluar	Kepala cabang	Riwayat pergerakan stok tersimpan
5	Pencatatan transaksi penjualan dan pencetakan nota	Kasir	Transaksi tersimpan dan nota dapat dicetak
6	Pencatatan retur barang dari pelanggan	Kasir, kepala cabang	Retur tercatat dan stok diperbarui setelah validasi
7	Pencatatan transfer barang antar cabang	Kepala cabang	Status transfer dan perubahan stok tercatat
8	Pengelolaan <i>shift</i> kasir	Kasir, kepala cabang	Riwayat modal awal, pendapatan, dan selisih kas tersimpan
9	Pengajuan dan persetujuan diskon produk	Kepala cabang, <i>Owner</i>	Status diskon tercatat dan digunakan pada transaksi
10	Penyajian laporan penjualan, stok, dan produk terlaris	<i>Owner</i> , kepala cabang	Laporan tampil berdasarkan cabang dan periode

Kebutuhan fungsional pada Tabel 1 digunakan sebagai acuan dalam perancangan modul sistem. Setiap kebutuhan dihubungkan dengan aktor yang berwenang agar sistem tidak hanya memenuhi proses bisnis, tetapi juga mendukung pengendalian akses pada lingkungan *multi-cabang*.

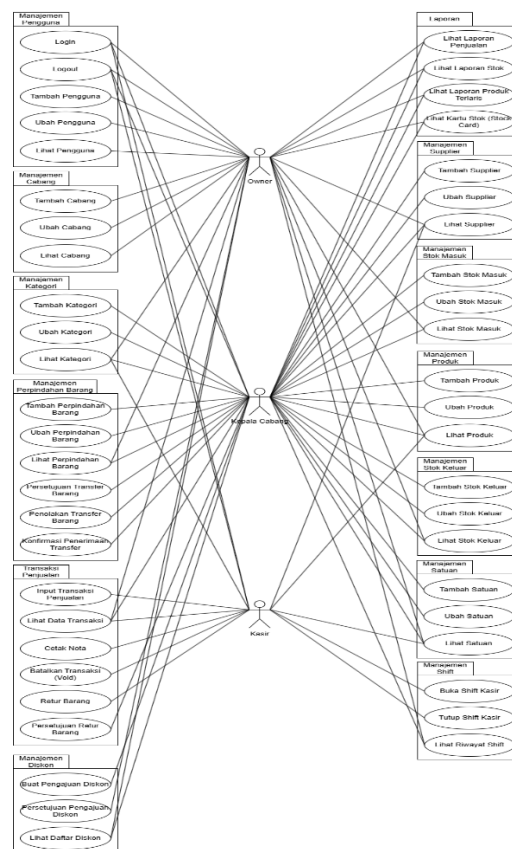
2.2 User Design

Tahap *User design* dilakukan untuk menerjemahkan kebutuhan pengguna ke dalam rancangan sistem. Perancangan dilakukan menggunakan *Use Case Diagram*, *Data Flow*

Diagram, dan Entity Relationship Diagram. Pemodelan interaksi pengguna terhadap fitur aplikasi direpresentasikan melalui Use Case Diagram. Untuk memetakan rute pertukaran informasi antara entitas eksternal dengan sistem, peneliti memanfaatkan Data Flow Diagram (DFD). Sedangkan arsitektur penyimpanan data yang merekam seluruh riwayat transaksi dan logistic disusun menggunakan Entity Relationship Diagram (ERD).

2.2.1 Use Case Diagram

Use Case Diagram menggambarkan fungsi-fungsi utama yang dapat diakses oleh setiap aktor dalam sistem. Aktor yang terlibat terdiri dari Owner, Kepala Cabang, dan Kasir. Owner memiliki hak akses untuk mengelola data pengguna dan cabang, melihat laporan seluruh cabang, serta menyetujui pengajuan diskon. Kepala Cabang berperan dalam pengelolaan data operasional cabang, seperti produk, kategori, satuan, *supplier*, stok masuk, stok keluar, perpindahan barang, serta validasi retur. Kasir berperan dalam membuka dan menutup *shift*, mencatat transaksi penjualan, mencetak nota, membatalkan transaksi sesuai ketentuan, dan mencatat retur barang. Gambaran hubungan antara aktor dan fungsi sistem ditampilkan pada Gambar 2.



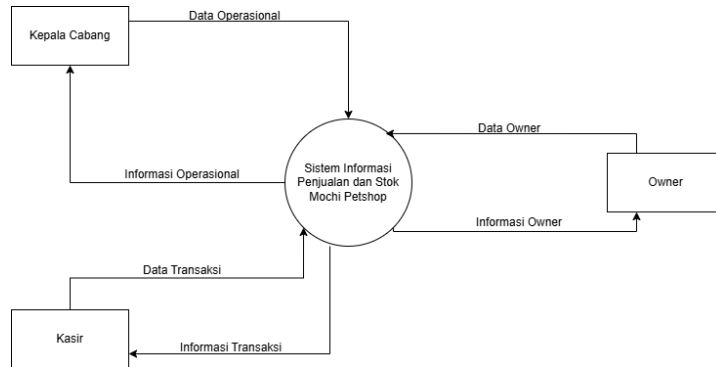
Gambar 2. Use case Diagram

2.2.2 Data Flow Diagram

Data Flow Diagram Level 0 digunakan untuk menunjukkan aliran data utama antara sistem dengan entitas eksternal. Pada rancangan ini, sistem diposisikan sebagai satu proses utama, yaitu Sistem Informasi Penjualan dan Stok Mochi *Petshop*, yang berinteraksi dengan tiga entitas eksternal: Owner, Kepala Cabang, dan Kasir.

Owner mengirimkan data *login*, data pengguna, data cabang, persetujuan pengajuan diskon, dan permintaan laporan. Sebagai keluaran, sistem memberikan status *login*, informasi pengguna, informasi cabang, laporan penjualan, laporan stok, laporan produk terlaris, dan kartu stok. Kepala Cabang mengirimkan data operasional cabang yang mencakup data *login*, data produk, kategori, satuan, *supplier*, stok masuk, stok keluar, perpindahan barang, persetujuan retur, persetujuan transfer, pengajuan diskon, dan permintaan laporan cabang. Sistem kemudian menghasilkan informasi operasional berupa status *login*, informasi master data, informasi stok,

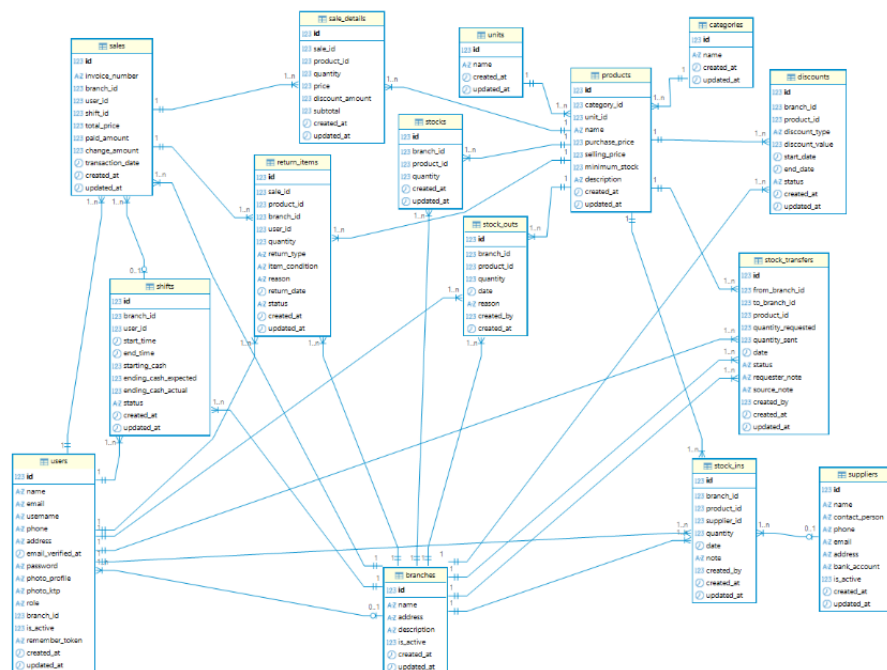
status retur dan transfer, status pengajuan diskon, serta laporan cabang. Kasir mengirimkan data transaksi kasir berupa data *login*, buka *shift*, tutup *shift*, transaksi penjualan, pembatalan transaksi, dan retur barang. Sistem memberikan keluaran berupa status *login*, informasi produk dan stok, nota transaksi, riwayat transaksi, status retur, dan riwayat *shift*. Aliran data tersebut menunjukkan bahwa setiap aktor memiliki ruang interaksi yang berbeda sesuai kewenangannya. Bentuk DFD Level 0 sistem ditampilkan pada Gambar 3.



Gambar 3. Data Flow Diagram Level 0

2.2.3 Entity Relationship Diagram

Entity Relationship Diagram digunakan untuk menggambarkan struktur basis data dan hubungan antar entitas yang digunakan dalam sistem. Relasi sistem ini memiliki 15 entitas utama yang berpusat pada entitas cabang (branches) yang melacak gerai fisik dan mengikat entitas pengguna (Users), stok ketersediaan barang (stocks) di setiap cabang, diskon promosi produk (discounts), sesi kerja kasir (shifts), transaksi penjualan (sales) beserta rincian itemnya (sale_details), pengajuan retur pelanggan (return_items), serta pencatatan pergerakan inventaris baik masuk dari supplier (stock_ins), keluar karena penyesuaian kerusakan (stock_outs), maupun mutasi antar-cabang (stock_transfers). Seluruh relasi dirancang menggunakan hubungan satu-ke-banyak (One-to-Many) untuk mengaitkan entitas induk dan tabel perantara untuk menjembatani relasi banyak-ke-banyak (Many-to-Many), guna memastikan alur pemotongan stok otomatis berjalan dinamis dan seluruh aktivitas transaksi tercatat secara aman dengan hak akses yang teratur. Rancangan ERD ditampilkan pada Gambar 4.



Gambar 4. Entity Relationship Diagram (ERD)

2.3 Construction

Tahap *construction* merupakan proses pembangunan sistem berdasarkan rancangan yang telah disetujui. Sistem dikembangkan menggunakan PHP dengan *framework* Laravel dan basis data MySQL, serta menerapkan pola *Model-View-Controller* agar logika aplikasi, tampilan antarmuka, dan pengelolaan data tersusun lebih terstruktur. Pengembangan dilakukan secara bertahap, mulai dari autentikasi pengguna, master data, transaksi penjualan, pengelolaan stok, retur barang, transfer antar cabang, hingga laporan. Pada tahap ini, setiap fitur juga diuji menggunakan *Black Box Testing* berdasarkan skenario masukan dan keluaran yang telah ditentukan, kemudian dilakukan perbaikan apabila ditemukan fungsi yang belum sesuai kebutuhan.

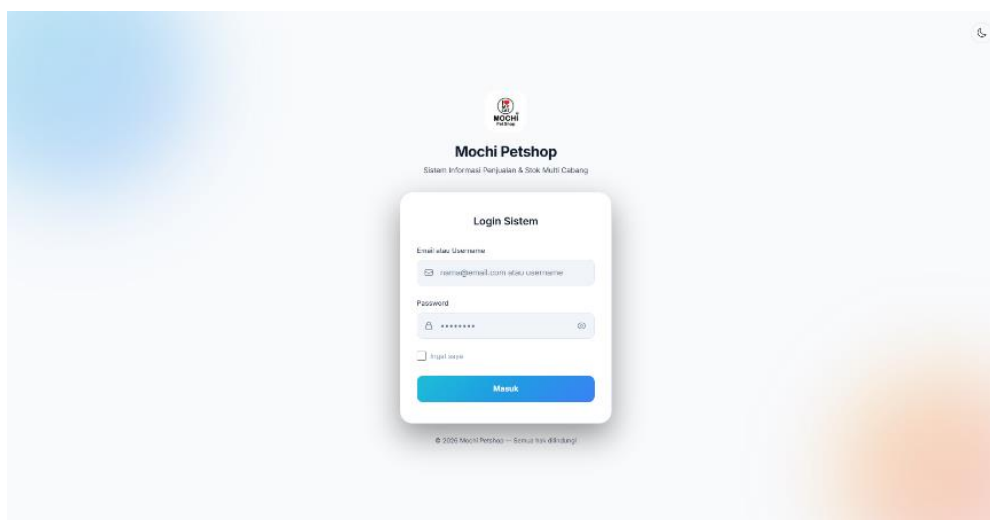
2.4 Cutover

Tahap *cutover* merupakan tahap akhir yang mencakup pengujian keseluruhan, evaluasi hasil, dan persiapan penggunaan sistem. Sistem diperiksa untuk memastikan seluruh fitur dapat digunakan oleh *Owner*, kepala cabang, dan kasir sesuai hak akses masing-masing. Evaluasi dilakukan dengan membandingkan kebutuhan awal dengan hasil implementasi. Luaran dari tahap ini adalah aplikasi sistem informasi penjualan dan stok berbasis web *multi-cabang* pada Mochi *Petshop* yang dapat membantu pengelolaan transaksi, stok, retur barang, transfer antar cabang, dan laporan secara terpusat.

3. Hasil dan Pembahasan

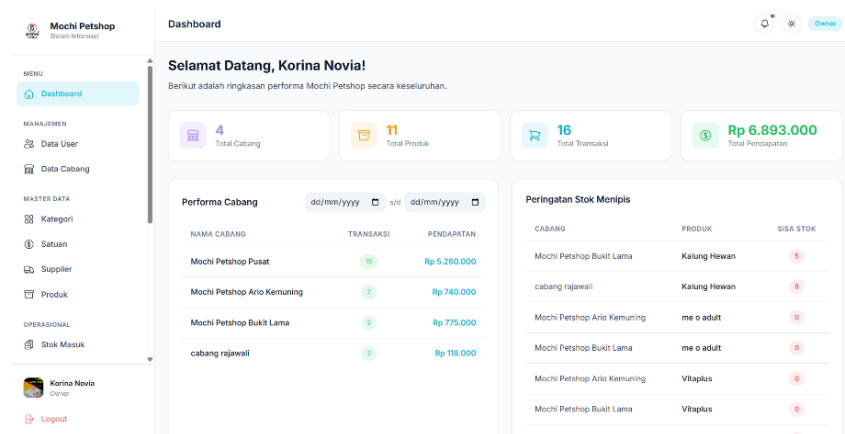
3.1 Implementasi Sistem

Halaman *login* berfungsi sebagai gerbang keamanan utama sebelum pengguna mengakses fitur aplikasi. Kredensial berupa nama pengguna atau surel beserta kata sandi yang valid diwajibkan pada tahap ini. Setelah proses autentikasi berhasil, sistem akan mengarahkan pengguna ke halaman utama sesuai dengan *role* masing-masing, baik sebagai *Owner*, Kepala Cabang, maupun Kasir.



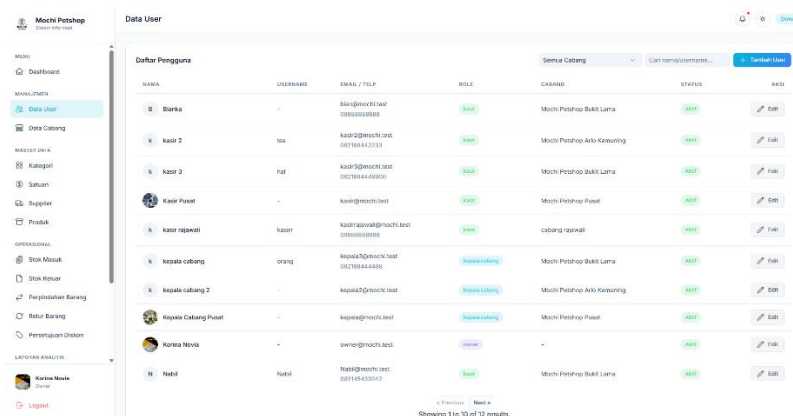
Gambar 5. Halaman *Login*

Dashboard Owner digunakan untuk menampilkan ringkasan informasi operasional Mochi *Petshop* secara keseluruhan. Melalui halaman ini, *Owner* dapat melihat jumlah cabang, total produk, jumlah transaksi, total pendapatan, performa penjualan tiap cabang, serta peringatan stok menipis. Informasi tersebut membantu *Owner* dalam melakukan pemantauan kondisi usaha secara cepat tanpa harus membuka setiap menu laporan secara terpisah.



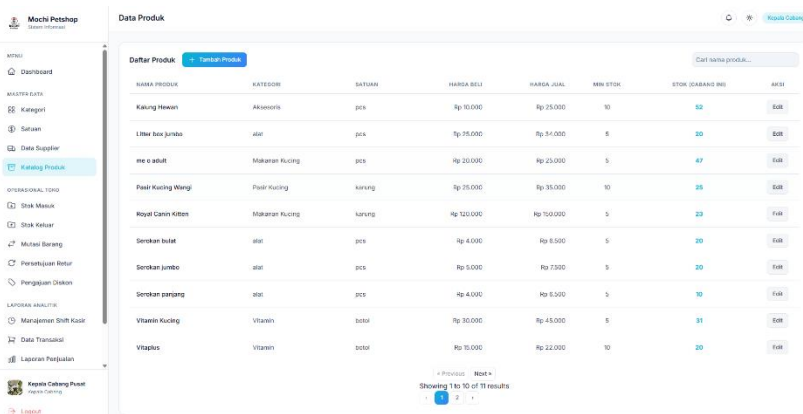
Gambar 6. Halaman Dashboard Owner

Halaman data *User* memfasilitasi pengaturan profil setiap karyawan yang diberikan wewenang untuk mengoperasikan *platform*. Fitur ini mencakup penambahan, perubahan, dan peninjauan data pengguna berdasarkan nama, email, nomor telepon, *role*, cabang, serta status akun. Pengelolaan data *User* menjadi penting karena sistem menerapkan pembagian hak akses antara *Owner*, Kepala Cabang, dan Kasir.



Gambar 7. Halaman Data User

Halaman Katalog Produk menyajikan rincian komprehensif mengenai seluruh persediaan item yang didistribusikan di setiap gerai. Informasi yang ditampilkan meliputi nama produk, kategori, satuan, harga beli, harga jual, stok minimum, serta jumlah stok yang tersedia. Fitur ini mendukung Kepala Cabang dalam memantau dan mengelola data produk agar proses penjualan dan pencatatan stok dapat berjalan lebih akurat.



Gambar 8. Halaman Katalog Produk

Halaman stok masuk digunakan untuk mencatat penambahan stok barang pada cabang. Data yang dicatat meliputi tanggal, nama produk, *supplier*, jumlah barang masuk, dan keterangan. Pencatatan stok masuk diperlukan agar setiap penerimaan barang dapat terdokumentasi dengan baik dan jumlah stok pada sistem selalu sesuai dengan kondisi barang di toko.

TANGGAL	PRODUK	SUPPLIER	JUMLAH	KETERANGAN	AKSI
26 May 2026 10:43 WIB	Vitaplus	Supplier Utama Mochi Petshop	+25	-	Edit Keterangan
26 May 2026 10:43 WIB	Serokan panjang	Supplier Utama Mochi Petshop	+10	-	Edit Keterangan
26 May 2026 00:02 WIB	Litter box jumbo	Pt Celentang Pet	+20	-	Edit Keterangan
26 May 2026 00:02 WIB	Serokan bulat	Pt Celentang Pet	+20	-	Edit Keterangan
26 May 2026 00:00 WIB	Serokan jumbo	Pt Celentang Pet	+20	-	Edit Keterangan
25 May 2026 23:06 WIB	Kalung Hewan	Pt Celentang Pet	+5	-	Edit Keterangan
25 May 2026 23:06 WIB	Pasir Kucing Wangi	Pt Celentang Pet	+5	-	Edit Keterangan
24 May 2026 18:22 WIB	me o adult	Pt Celentang Pet	+20	batch test	Edit Keterangan
24 May 2026 14:00 WIB	Pasir Kucing Wangi	Supplier Utama Mochi Petshop	+20	batch pertama	Edit Keterangan
24 May 2026					Edit

Gambar 9. Halaman Stok Masuk

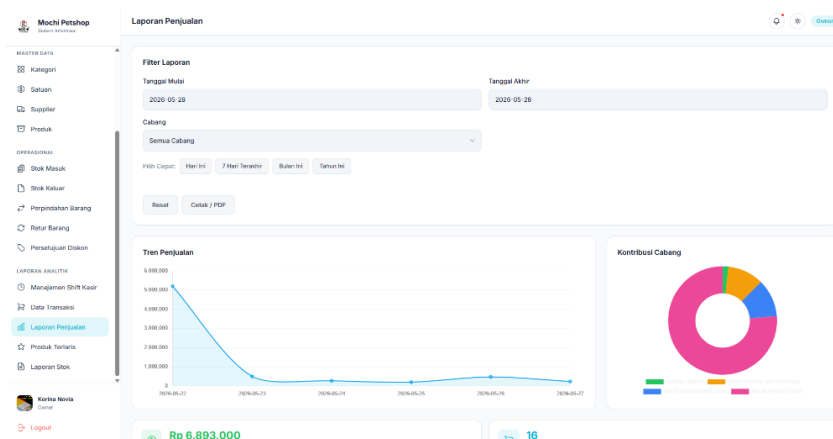
Halaman Kasir digunakan oleh Kasir untuk melakukan proses transaksi penjualan. Pada halaman ini, Kasir dapat memilih produk, memasukkan jumlah pembelian, melihat total tagihan, mencatat uang pembayaran, serta memproses transaksi. Fitur ini dirancang untuk mempercepat pelayanan kepada pelanggan sekaligus menghasilkan data transaksi yang otomatis tersimpan ke dalam sistem.

Produk	Harga	Stok
Whiskas Adult	Rp 55.000	Stok: 20 pack
Royal Canin Kitten	Rp 150.000	Stok: 20 kemasan
Pasir Kucing Wangi	Rp 35.000	Stok: 20 kemasan
Vitamin Kucing	Rp 45.000	Stok: 20 bender
Kalung Hewan	Rp 35.000	Stok: 52 pcs
me o adult	Rp 25.000	Stok: 47 pcs
Vitaplus	Rp 22.000	Stok: 20 botol
Serokan bulat	Rp 6.500	Stok: 20 pcs
Serokan panjang	Rp 6.500	Stok: 10 pcs
Serokan jumbo	Rp 7.500	Stok: 20 pcs
Litter box jumbo	Rp 34.000	Stok: 20 pcs

Koranjang	Jumlah
Pasir Kucing Wangi Rp 35.000	1
Serokan bulat Rp 6.500	2
Total Tagihan	Rp 48.000
Uang Bayar (Tunai)	50.000
Kemaaian	Rp 2.000

Gambar 10. Halaman Kasir

Halaman laporan penjualan digunakan untuk menampilkan informasi penjualan berdasarkan periode dan cabang tertentu. Laporan ini menyajikan data seperti tren penjualan, kontribusi cabang, total pendapatan, jumlah transaksi, dan detail transaksi. Dengan adanya laporan penjualan, *Owner* maupun Kepala Cabang dapat melakukan evaluasi terhadap performa penjualan secara lebih terukur.



Gambar 11. Halaman Laporan Penjualan

3.2 Pengujian Sistem

Tahap evaluasi fungsionalitas aplikasi dieksekusi melalui pendekatan Black Box Testing. Titik berat pengujian ini terletak pada proses validasi antara hasil keluaran program dengan spesifikasi kebutuhan yang telah dirumuskan pada tahap awal. Setiap skenario diuji berdasarkan aktivitas utama yang dilakukan oleh *Owner*, kepala cabang, dan kasir. Hasil pengujian disajikan pada Tabel 3. Kolom status dapat disesuaikan kembali setelah pengujian aktual dilakukan pada sistem yang telah diimplementasikan.

Tabel 3. Hasil Black Box Testing

No.	Skenario Pengujian	Input / Aksi	Hasil yang Diharapkan	Status
1	Login pengguna	Memasukkan <i>Username</i> dan <i>password valid</i>	Sistem menampilkan <i>dashboard</i> sesuai <i>role</i> pengguna	Sesuai
2	Login gagal	Memasukkan data <i>login tidak valid</i>	Sistem menolak akses dan menampilkan pesan kesalahan	Sesuai
3	Kelola pengguna	<i>Owner</i> menambah atau mengubah data pengguna	Data pengguna tersimpan dan hak akses dapat digunakan	Sesuai
4	Kelola cabang	<i>Owner</i> menambah atau mengubah data cabang	Data cabang tersimpan dan dapat dipilih pada proses terkait	Sesuai
5	Kelola kategori dan satuan	Kepala cabang menambah kategori atau satuan	Data kategori dan satuan tersimpan dalam sistem	Sesuai
6	Kelola <i>supplier</i>	Kepala cabang menambah data <i>supplier</i>	Data <i>supplier</i> tersimpan dan dapat digunakan pada stok masuk	Sesuai
7	Kelola produk	Kepala cabang menambah atau mengubah produk	Data produk tersimpan lengkap dengan kategori, satuan, dan harga	Sesuai
8	Input stok masuk	Kepala cabang mencatat barang masuk	Stok cabang bertambah dan riwayat stok masuk tersimpan	Sesuai
9	Input stok keluar	Kepala cabang mencatat barang keluar non-penjualan	Stok cabang berkurang dan riwayat stok keluar tersimpan	Sesuai
10	Transaksi penjualan	Kasir memilih produk dan menyimpan transaksi	Transaksi tersimpan, total dihitung, dan stok berkurang	Sesuai
11	Cetak nota	Kasir menekan tombol cetak nota	Sistem menghasilkan nota transaksi penjualan	Sesuai
12	Void transaksi	Kasir membatalkan transaksi sesuai batas waktu	Transaksi dibatalkan dan stok dikembalikan	Sesuai
13	Retur barang	Kasir mencatat retur pelanggan	Data retur tersimpan dan menunggu validasi kepala cabang	Sesuai
14	Persetujuan retur	Kepala cabang menyetujui atau menolak retur	Status retur berubah dan stok diperbarui sesuai kondisi barang	Sesuai
15	Pengajuan transfer barang	Kepala cabang mengajukan perpindahan barang	Data transfer tersimpan dengan status pengajuan	Sesuai

No.	Skenario Pengujian	Input / Aksi	Hasil yang Diharapkan	Status
16	Persetujuan transfer	Cabang sumber menyetujui transfer barang	Stok cabang sumber berkurang sesuai jumlah disetujui	Sesuai
17	Konfirmasi penerimaan transfer	Cabang tujuan mengonfirmasi barang diterima	Stok cabang tujuan bertambah dan status transfer selesai	Sesuai
18	Buka <i>shift</i> kasir	Kasir memasukkan modal awal	Sesi <i>shift</i> aktif dan transaksi dapat dilakukan	Sesuai
19	Tutup <i>shift</i> kasir	Kasir memasukkan nominal kas aktual	Sistem menghitung selisih kas dan menyimpan riwayat <i>shift</i>	Sesuai
20	Laporan penjualan	Owner atau kepala cabang memilih periode laporan	Sistem menampilkan laporan sesuai cabang dan periode	Sesuai
21	Laporan stok	Owner atau kepala cabang membuka laporan stok	Sistem menampilkan stok terkini dan riwayat pergerakan barang	Sesuai
22	<i>Logout</i>	Pengguna menekan tombol <i>logout</i>	Sesi pengguna berakhir dan sistem kembali ke halaman <i>login</i>	Sesuai

3.3 Pembahasan

Hasil implementasi menunjukkan bahwa sistem yang dibangun mampu menjawab tujuan penelitian, yaitu menghasilkan sistem informasi penjualan dan stok berbasis web *multi-cabang* yang mengintegrasikan data transaksi, stok, retur, transfer antar cabang, dan laporan. Pada kondisi awal, data setiap cabang dicatat secara terpisah sehingga owner tidak dapat memperoleh informasi penjualan dan ketersediaan stok secara menyeluruh. Setelah sistem diterapkan, setiap aktivitas operasional tersimpan dalam basis data terpusat dan dapat ditampilkan berdasarkan cabang maupun periode. Temuan ini selaras dengan konsep sistem informasi yang menempatkan data sebagai sumber informasi manajerial untuk mendukung operasi dan pengambilan keputusan [1]. Dengan demikian, hasil penelitian tidak hanya menghasilkan aplikasi transaksi, tetapi juga membentuk mekanisme pengelolaan informasi yang lebih terstruktur.

Peningkatan paling nyata terdapat pada integrasi transaksi penjualan dengan perubahan stok. Pada saat kasir menyimpan transaksi, sistem menghitung total pembayaran dan mengurangi stok pada cabang terkait secara otomatis. Mekanisme ini mengurangi risiko perbedaan antara catatan penjualan dan saldo stok karena proses pencatatan tidak lagi dilakukan dalam dua media berbeda. Hasil tersebut mendukung temuan Angellin dkk. [2] bahwa sistem informasi penjualan dan persediaan berbasis web dapat meningkatkan ketepatan pencatatan transaksi dan mempermudah pemantauan stok secara *real-time*. Dalam konteks *Mochi Petshop*, manfaat tersebut menjadi lebih penting karena jumlah produk cukup banyak dan transaksi harian pada toko pusat relatif tinggi.

Dari sisi pengelolaan persediaan, sistem mendukung pencatatan stok masuk, stok keluar, retur, dan transfer antar cabang. Setiap perubahan stok memiliki riwayat yang dapat ditelusuri sehingga kepala cabang dan owner dapat mengetahui sumber perubahan jumlah barang. Hal ini sesuai dengan prinsip sistem informasi persediaan yang menuntut pencatatan barang masuk, barang keluar, dan saldo persediaan secara konsisten [3]. Fitur retur juga memberikan kontribusi praktis karena barang yang dikembalikan pelanggan tidak langsung memengaruhi stok sebelum divalidasi oleh kepala cabang. Mekanisme validasi tersebut penting untuk membedakan retur yang masih layak dijual, retur rusak, atau retur yang membutuhkan pengembalian dana.

Fitur transfer barang antar cabang menjadi bagian penting karena karakteristik usaha *Mochi Petshop* berbeda dari sistem penjualan satu lokasi. Dalam praktik operasional, perpindahan barang sering dilakukan untuk menyeimbangkan ketersediaan stok antar cabang. Sebelum sistem dibangun, koordinasi transfer dilakukan melalui komunikasi manual sehingga riwayat pengiriman dan penerimaan barang sulit dilacak. Melalui sistem, proses transfer memiliki status pengajuan, persetujuan, dan konfirmasi penerimaan. Hasil ini memperkuat konsep sistem *multi-cabang* yang menekankan integrasi data dari beberapa lokasi dalam satu platform terpusat [4]. Hasil penelitian juga sejalan dengan Widiastuti dkk. [5] yang menyatakan bahwa integrasi cabang mempermudah pemantauan transaksi dan stok secara menyeluruh. Perbedaan utama penelitian ini terletak pada penyesuaian proses transfer dengan kebutuhan *petshop*, termasuk pengaruh transfer terhadap stok cabang sumber dan cabang tujuan.

Jika dibandingkan dengan penelitian Mulyana dan Rusmawan [6], sistem yang dikembangkan sama-sama mendukung pencatatan transaksi dan laporan penjualan berbasis

web. Namun, penelitian ini memiliki cakupan lebih luas karena mencakup *multi-cabang*, retur, transfer barang, dan *shift* kasir. Dibandingkan dengan Auditira dkk. [7], penelitian ini memiliki kesamaan pada konteks *petshop*, tetapi sistem Cora *Petshop* masih berfokus pada satu lokasi usaha. Penelitian Salsabila dan Andryana [8] memiliki kedekatan pada pengelolaan persediaan *petshop*, khususnya pencatatan stok dan barang kedaluwarsa, sedangkan penelitian ini menambahkan integrasi antar cabang dan transaksi penjualan. Penelitian Afriani dan Ayu Gusriyanti [9] menunjukkan pentingnya sistem penjualan dan jasa pada *petshop*, tetapi belum membahas distribusi stok antar cabang. Dengan demikian, penelitian ini mendukung temuan sebelumnya sekaligus memberikan perluasan pada aspek integrasi operasional *petshop multi-cabang*.

Pemilihan *Laravel* sebagai *framework* pengembangan berkontribusi pada keteraturan struktur sistem. Arsitektur *Model-View-Controller* memungkinkan pemisahan antara logika bisnis, tampilan, dan pengelolaan data sehingga setiap modul dapat dikembangkan dan dipelihara secara lebih sistematis [10]. Hal ini tampak pada pemisahan modul *otentikasi*, master data, transaksi, persediaan, transfer, retur, dan laporan. Fitur bawaan *Laravel* seperti *routing*, *middleware*, dan integrasi basis data juga membantu penerapan pembatasan akses berdasarkan peran pengguna [11]. Faktor ini menjelaskan mengapa sistem dapat mengakomodasi tiga peran utama tanpa mencampurkan kewenangan *owner*, kepala cabang, dan kasir.

Metode RAD juga berpengaruh terhadap hasil yang diperoleh. Proses *requirement planning* membantu mengidentifikasi kebutuhan utama dari proses bisnis yang berjalan, sedangkan *user design* memungkinkan rancangan fungsi disesuaikan dengan aktivitas pengguna. Tahap *construction* kemudian menerjemahkan rancangan tersebut menjadi fitur aplikasi, dan *cutover* digunakan untuk memeriksa kesesuaian sistem terhadap kebutuhan awal. Pola ini sesuai dengan karakter RAD yang menekankan pembangunan cepat, iterasi, dan umpan balik pengguna [12]. Hasil ini juga mendukung Widodo dkk. [13] serta Fahmi Rosyadi dan Sembiring [14] yang menyatakan bahwa RAD sesuai untuk pengembangan sistem berbasis web dan POS karena kebutuhan dapat dievaluasi secara bertahap selama proses pembangunan.

Hasil *Black Box Testing* menunjukkan bahwa seluruh skenario utama berstatus sesuai. Artinya, fitur yang diuji dapat menghasilkan keluaran yang sama dengan kebutuhan fungsional yang telah didefinisikan, mulai dari *login*, master data, transaksi, stok, retur, transfer, *shift*, laporan, hingga *logout*. Hasil ini memperlihatkan bahwa sistem telah memenuhi aspek fungsional pada tahap pengujian. Akan tetapi, status sesuai pada *Black Box Testing* tidak dapat langsung diartikan bahwa sistem telah optimal dalam seluruh kondisi operasional. Pengujian tersebut belum mengukur performa ketika jumlah transaksi sangat besar, kecepatan akses pada jaringan yang tidak stabil, maupun tingkat penerimaan pengguna dalam jangka panjang.

Secara rasional, keberhasilan sistem dalam meningkatkan keteraturan data disebabkan oleh tiga faktor. Pertama, penyimpanan data dilakukan secara terpusat sehingga data dari setiap cabang tidak lagi terpisah dalam buku atau file berbeda. Kedua, setiap proses utama dikaitkan dengan perubahan data stok dan transaksi sehingga informasi yang ditampilkan lebih konsisten. Ketiga, pembagian kewenangan pengguna membatasi perubahan data hanya pada aktor yang relevan. Ketiga faktor tersebut juga ditemukan pada penelitian Andriasari dkk. [15], yang menunjukkan bahwa digitalisasi POS dapat meningkatkan efisiensi dan akurasi data pada UMKM. Pada penelitian ini, prinsip tersebut diterapkan pada konteks *petshop multi-cabang* dengan kebutuhan distribusi barang yang lebih kompleks.

Kontribusi penelitian ini terhadap bidang sistem informasi terletak pada model implementasi sistem penjualan dan stok untuk usaha *petshop* yang memiliki lebih dari satu cabang. Penelitian ini menunjukkan bahwa integrasi transaksi, persediaan, retur, transfer, dan laporan dapat ditempatkan dalam satu arsitektur sistem berbasis web tanpa menghilangkan tanggung jawab pengguna. Secara praktis, hasil penelitian dapat membantu owner melakukan monitoring cabang, kepala cabang mengendalikan stok, dan kasir mencatat transaksi secara lebih cepat. Secara akademik, penelitian ini memperkaya kajian sistem informasi ritel skala usaha kecil dan menengah dengan menambahkan konteks operasional *multi-cabang* dan karakteristik produk *petshop*.

Meskipun demikian, penelitian ini memiliki beberapa keterbatasan. Sistem belum membahas integrasi pembayaran *online*, integrasi dengan perangkat POS khusus, analisis prediktif untuk perencanaan stok, dan pengujian beban sistem pada *volume* data besar. Keakuratan informasi juga tetap bergantung pada ketepatan input pengguna, sehingga kesalahan input masih dapat memengaruhi laporan. Peluang penelitian berikutnya dapat

diarahkan pada integrasi *barcode scanner*, pencatatan *batch* dan tanggal kedaluwarsa, *dashboard* analitik, peramalan kebutuhan stok, audit log yang lebih rinci, serta evaluasi *usability* dan *user acceptance* setelah sistem digunakan secara penuh oleh seluruh cabang. Pengembangan tersebut penting agar sistem tidak hanya berfungsi sebagai alat pencatatan, tetapi juga menjadi pendukung keputusan operasional yang lebih adaptif.

4. Simpulan

Berdasarkan hasil penelitian, sistem informasi penjualan dan stok berbasis web *multi-cabang* pada Mochi *Petshop* berhasil dirancang untuk membantu pengelolaan transaksi, stok barang, retur, perpindahan barang antar cabang, dan laporan secara terpusat. Sistem ini memberikan solusi terhadap permasalahan pencatatan manual yang menyebabkan keterlambatan pencarian data, ketidaksesuaian stok, dan kesulitan monitoring operasional. Penerapan metode *Rapid Application Development* memungkinkan proses pengembangan dilakukan secara bertahap dengan menyesuaikan kebutuhan pengguna. Hasil *pengujian Black Box Testing* menunjukkan bahwa fitur utama sistem berjalan sesuai kebutuhan, sehingga sistem berpotensi meningkatkan efisiensi operasional, akurasi data, dan kemudahan pengambilan keputusan pada Mochi *Petshop*. Pengembangan selanjutnya dapat diarahkan pada peningkatan keamanan sistem, integrasi perangkat cetak nota, serta penambahan fitur analisis penjualan untuk mendukung perencanaan stok yang lebih baik.

Daftar Referensi

- [1] K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, 16th ed. New York, NY, USA: Pearson Education, 2018.
- [2] K. Angellin, R. S. Oetama, and M. Amri, "Web-Based Inventory and Sales Information System: Indonesian Micro Small Medium Enterprise Case Study," *JOINS (Journal of Information System)*, vol. 8, no. 1, pp. 57–66, 2023, doi: 10.33633/joins.v8i1.7977.
- [3] A. Rohman and H. D. Bhakti, "Perancangan Sistem Informasi Persediaan Barang Berbasis Web," *Syntax Literate: Jurnal Ilmiah Indonesia*, vol. 7, no. 9, pp. 15304–15313, 2022, doi: 10.36418/syntax-literate.v7i9.14255.
- [4] A. Husna Ahadi and T. Djauhari, "Pengembangan Sistem Informasi Manajemen Hotel Multi-Cabang Berbasis Web dengan Framework Laravel," *Djtechno: Jurnal Teknologi Informasi*, vol. 6, no. 2, pp. 758–773, 2025, doi: 10.46576/djtechno.v6i2.7139.
- [5] P. A. Widiastuti, N. W. Utami, and N. M. Estiyanti, "Perancangan Sistem Point of Sales (POS) Terintegrasi pada UD. Akor Nature Bag," *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 10, no. 1, pp. 171–184, 2021, doi: 10.35889/jutisi.v10i2.659.
- [6] A. Mulyana and U. Rusmawan, "Rancang Bangun Sistem Informasi Point of Sale (POS) Berbasis Web: Studi Kasus Toko Andorio," *Majalah Ilmiah UNIKOM*, vol. 21, no. 1, pp. 43–50, 2023, doi: 10.34010/miu.v21i1.10689.
- [7] D. Auditira, A. Triayudi, and D. Hidayatullah, "Implementasi Point of Sale pada Cora Petshop Menggunakan Metode Agile dan Scrum Framework," *Journal of Information System Research (JOSH)*, vol. 4, no. 2, pp. 532–542, 2023, doi: 10.47065/josh.v4i2.2966.
- [8] S. Salsabila and S. Andryana, "SIRBA: Sistem Informasi Persediaan Barang Petshop Berbasis Web dengan Metode FEFO dan Model FAST," *Jurnal Sistem dan Teknologi Informasi (JustIN)*, vol. 10, no. 4, pp. 495–504, 2022, doi: 10.26418/justin.v10i4.52641.
- [9] M. Afriani and D. A. Gusriyanti, "Perancangan Sistem Informasi Penjualan dan Jasa Grooming pada Linda Petshop Jambi," *Jurnal Informatika dan Rekayasa Komputer (JAKAKOM)*, vol. 3, no. 2, pp. 591–599, 2023, doi: 10.33998/jakakom.2023.3.2.876.
- [10] L. Rahmawati and S. Sumarsono, "Desain Pengembangan Website dengan Arsitektur Model View Controller pada Framework Laravel," *Jurnal Teknologi dan Sistem Informasi Bisnis*, vol. 6, no. 4, pp. 785–790, 2024, doi: 10.47233/jteksis.v6i4.1497.
- [11] A. Surahmat and R. Darmawan, "Application of SDLC Method and Laravel Framework in the Development of Web-Based Company Profile Information Systems," *JISI: Jurnal Ilmiah Sistem Informasi*, vol. 2, no. 1, pp. 20–25, 2024, doi: 10.70429/sjis.v2i1.101.
- [12] D.K. Ramadhani, & S. Widiono, "Penerapan metode RAD dalam pengembangan sistem pemesanan berbasis mobile di Toko Hurip," *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 13, no. 3, pp. 1966-1976, 2024.

- [13] E. Widodo, R. W. Setiawan, M. N. A. Dasra, and Y. A. Singgalen, "Enhancing Website Management Through Expertise and Rapid Application Development Frameworks," *Journal of Information Systems and Informatics*, vol. 6, no. 2, pp. 781–796, 2024, doi: 10.51519/journalisi.v6i2.725.
- [14] M. F. Rosyadi and F. Sembiring, "Penerapan Rapid Application Development (RAD) dalam Penerapan Sistem Point of Sale (POS) Terintegrasi di Klinik Apotek Nusa Putra Farmedika," *ZONAsi: Jurnal Sistem Informasi*, vol. 7, no. 2, pp. 734–746, 2025, doi: 10.31849/12vhsk98.
- [15] S. Andriasari, N. Nizam, I. A. Nurhasanah, and K. S. Wulandari, "Aplikasi Point of Sale untuk Meningkatkan Profitabilitas dan Digitalisasi UMKM," *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, vol. 14, no. 2, pp. 89–97, 2024, doi: 10.36448/expert.v14i2.3974.