



SISTEM INFORMASI POINT OF SALES BERBASIS CODEIGNITER

Teguh Hayatul Atni Hadi ¹, Rikie Kartadie²

¹ Program Studi Informatika, Universitas Teknologi Digital Indonesia, ² Program Studi Teknik Komputer, Universitas Teknologi Digital Indonesia

Jl. Raya Janti Karang Jambe No. 143 Yogyakarta 55198, Indonesia

¹ teguh.hayatul@students.utdi.ac.id, ² rikie@utdi.ac.id

Abstract

The transformation of traditional business operations to digital systems has become crucial for small and medium enterprises (SMEs) to remain competitive in the modern economy. This study develops a web-based Point of Sales (POS) information system using the CodeIgniter framework to address critical operational inefficiencies in SME retail operations, specifically at ROS Minimarket. The research tackles significant challenges including manual transaction recording resulting in 15% error rates, 3-5 minute transaction processing times, and 2-3 day delays in financial reporting. Using the Waterfall development methodology, the system integrates real-time inventory management, automated transaction processing with barcode scanning, comprehensive financial reporting, and multi-user role-based access control. Implementation results demonstrate substantial operational improvements: 60% reduction in transaction processing time (from 3-5 minutes to 1-2 minutes), 86% decrease in recording errors (from 15% to 2%), and 99% improvement in reporting efficiency (from 2-3 days to real-time). System performance testing shows 1.2-second average response time with 99.9% availability, supporting up to 20 concurrent users. Security implementation includes multi-layered protection with input validation, XSS prevention, CSRF protection, and bcrypt password encryption. User acceptance testing achieved 4.45/5.0 satisfaction rating with 100% functional test success rate. This research contributes to SME digital transformation by providing an accessible, efficient, and secure POS solution that enhances operational competitiveness and business sustainability, thereby offering a novel scientific contribution in the development of integrated information systems for SMEs.

Keywords: Information System, Point of Sales, CodeIgniter, Waterfall. Multi-user, System Security.

Abstrak

Transformasi operasi bisnis tradisional ke sistem digital telah menjadi krusial bagi usaha kecil menengah (UKM) untuk tetap kompetitif dalam ekonomi modern. Penelitian ini mengembangkan sistem informasi Point of Sales (POS) berbasis web menggunakan framework CodeIgniter untuk mengatasi inefisiensi operasional kritis dalam operasi ritel UKM, khususnya pada ROS Minimarket. Penelitian menangani tantangan signifikan termasuk pencatatan transaksi manual yang menghasilkan tingkat kesalahan 15%, waktu pemrosesan transaksi 3-5 menit, dan keterlambatan 2-3 hari dalam pelaporan keuangan. Menggunakan metodologi pengembangan Waterfall, sistem mengintegrasikan manajemen inventori real-time, pemrosesan transaksi otomatis dengan pemindaian barcode, pelaporan keuangan komprehensif, dan kontrol akses berbasis peran multi-user. Hasil implementasi menunjukkan peningkatan operasional yang substansial: pengurangan 60% waktu pemrosesan transaksi (dari 3-5 menit menjadi 1-2 menit), penurunan 86% kesalahan pencatatan (dari 15% menjadi 2%), dan peningkatan 99% efisiensi pelaporan (dari 2-3 hari menjadi real-time). Pengujian kinerja sistem menunjukkan waktu respons rata-rata 1,2 detik dengan ketersediaan 99,9%, mendukung hingga 20 pengguna bersamaan. Implementasi keamanan mencakup perlindungan berlapis dengan validasi input, pencegahan XSS, perlindungan CSRF, dan enkripsi password bcrypt. Pengujian penerimaan pengguna



mencapai rating kepuasan 4,45/5,0 dengan tingkat keberhasilan uji fungsional 100%. Penelitian ini berkontribusi pada transformasi digital UKM dengan menyediakan solusi POS yang dapat diakses, efisien, dan aman yang meningkatkan daya saing operasional dan keberlanjutan bisnis, sekaligus menjadi kontribusi ilmiah dalam pengembangan sistem informasi terintegrasi untuk UKM.

Kata Kunci: *Sistem Informasi, Point of Sales, CodeIgniter, Waterfall. Multi-user, Sistem Keamanan*

1. PENDAHULUAN

ROS Minimarket merupakan sebuah Usaha Kecil Menengah (UKM) yang bergerak di bidang perdagangan ritel sembako, makanan ringan, dan minuman kemasan. Minimarket ini terletak di Masbagik, Lombok Timur, dan telah melayani masyarakat sekitar selama beberapa tahun. Namun, dalam pelaksanaan operasional hariannya, khususnya dalam pengelolaan transaksi penjualan dan stok barang, ROS Minimarket masih menggunakan sistem manual seperti pencatatan dengan buku tulis dan perhitungan menggunakan kalkulator. Hal ini menimbulkan berbagai permasalahan seperti kesalahan pencatatan, keterlambatan pembuatan laporan penjualan, dan kesulitan dalam pelacakan inventaris.

Seiring berkembangnya teknologi informasi, berbagai aspek bisnis mengalami transformasi ke arah digital [1]. Teknologi informasi adalah seperangkat alat yang digunakan untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan, koordinasi, dan analisis dalam sebuah organisasi. Salah satu aplikasi teknologi informasi dalam bisnis ritel adalah sistem *Point of Sales* (POS) [2].

POS atau sistem titik penjualan adalah sistem informasi yang digunakan untuk mencatat transaksi penjualan, mengelola stok barang, serta menghasilkan laporan keuangan [3]. Sistem informasi POS memungkinkan pengumpulan data transaksi secara *real-time*, yang memudahkan manajemen dalam membuat keputusan berbasis data [4]. Implementasi POS sangat penting bagi bisnis ritel karena mampu meningkatkan efisiensi kerja, mengurangi risiko kesalahan manusia, dan mempercepat proses transaksi pelanggan [5].

Sistem Informasi POS telah menjadi bagian tak terpisahkan dari transformasi digital pada sektor ritel. POS modern mengintegrasikan

proses transaksi, *inventory*, dan laporan bisnis ke dalam satu sistem terpusat, memungkinkan pengambilan keputusan yang lebih cepat dan akurat [6]. Berbagai penelitian menunjukkan bahwa efisiensi sistem POS sangat ditentukan oleh arsitektur teknologinya dan *platform* yang digunakan.

Berdasarkan kajian literatur yang telah dilakukan, masih terdapat beberapa celah (*gap*) dalam pengembangan sistem *Point of Sale* (POS) khususnya untuk diterapkan pada sektor Usaha Kecil dan Menengah (UKM). Penelitian Mehetabel dkk. [7] memang menekankan aspek keamanan dengan menerapkan enkripsi AES-256 untuk melindungi data transaksi, namun sistem yang dikembangkan belum mengintegrasikan manajemen inventaris secara *real-time*. Hal ini menjadi kelemahan karena manajemen stok yang tidak terintegrasi dapat menyulitkan pelacakan barang dan menimbulkan ketidaksesuaian antara stok fisik dan sistem. Sementara itu, penelitian oleh Widiyanti dan Tisnawati [2] berhasil mengembangkan sistem POS berbasis desktop menggunakan *Visual Basic*, tetapi pendekatan tersebut memiliki keterbatasan dari sisi aksesibilitas dan mobilitas. Pengguna hanya dapat mengakses sistem dari perangkat tertentu di lokasi tetap, sehingga kurang fleksibel untuk kebutuhan operasional yang dinamis, terutama di era digital saat ini.

Fachrozy, M. I., & Batubara [8] mencoba menjawab tantangan tersebut dengan mengembangkan aplikasi POS berbasis *mobile* Android yang menawarkan kemudahan akses melalui perangkat seluler. Namun, sistem ini belum mengakomodasi fitur pelaporan yang komprehensif dan tidak mendukung manajemen multi-user, padahal kedua fitur ini sangat penting untuk mendukung pengambilan keputusan dan koordinasi antar staf dalam sebuah usaha ritel. Selain itu, sebagian besar penelitian terdahulu lebih berfokus pada pengembangan sistem POS untuk skala enterprise atau perusahaan besar,



seperti yang terlihat pada pendekatan teknologi dan fitur yang digunakan. Hal ini menyebabkan kurangnya penyesuaian dengan kebutuhan UKM yang memiliki keterbatasan sumber daya baik dari sisi infrastruktur, anggaran, maupun kompetensi teknis.

Dengan demikian, dapat disimpulkan bahwa masih terdapat beberapa kekosongan dalam pengembangan sistem POS yang benar-benar sesuai untuk UKM, khususnya dari aspek integrasi fitur yang komprehensif, fleksibilitas akses dari berbagai perangkat, penyediaan laporan *real-time*, serta adaptasi terhadap keterbatasan yang dimiliki pelaku usaha kecil. Gap ini menjadi dasar penting bagi pengembangan sistem POS yang lebih responsif dan relevan terhadap kebutuhan operasional bisnis seperti yang dijalankan oleh ROS Minimarket.

Untuk menjawab gap penelitian tersebut, diperlukan sistem informasi kasir berbasis web yang mampu mengintegrasikan proses transaksi, manajemen inventori, dan pelaporan keuangan dalam satu *platform* yang dapat diakses dari berbagai perangkat. Penggunaan *framework* dalam pengembangan sistem menjadi pilihan logis untuk mempercepat proses pembangunan aplikasi [9]. Salah satu *framework* yang banyak digunakan dalam pengembangan aplikasi web adalah CodeIgniter [10].

Framework CodeIgniter telah dilengkapi dengan mekanisme *input filtering*, *query binding*, dan *XSS filtering* yang mampu menangkal hingga 98% percobaan penetrasi berbasis web [11]. Kemampuan *caching* CodeIgniter mampu mengurangi *load server* hingga 40% dengan menggunakan *library caching* bawaan [12]. CodeIgniter juga dapat menangani beban tinggi dengan tetap mempertahankan *response time* di bawah 2 detik untuk hingga 1000 pengguna secara bersamaan [13].

Penelitian ini bertujuan untuk merancang dan membangun sistem informasi *Point of Sales* (POS) berbasis web menggunakan *framework* CodeIgniter yang spesifik untuk kebutuhan UKM seperti ROS Minimarket. Fokus utama dari penelitian ini adalah pada pengembangan fitur-fitur terintegrasi meliputi manajemen inventori *real-time*, pencatatan transaksi penjualan, pelaporan keuangan komprehensif, dan sistem multi-user dengan tingkat akses yang berbeda, yang terintegrasi dalam satu sistem informasi

yang efisien, aman, dan dapat diakses dari berbagai perangkat.

2. TINJAUAN PUSTAKA

Penelitian mengenai pengembangan sistem Point of Sale (POS) untuk mendukung kebutuhan bisnis skala kecil dan menengah telah banyak dilakukan dalam berbagai pendekatan dan platform teknologi. Berdasarkan platform, penelitian dapat dikelompokkan menjadi POS berbasis desktop, mobile, dan web.

Penelitian oleh Widiyanti dan Tisnawati [2] berhasil mengembangkan sistem POS berbasis desktop menggunakan Visual Basic. Meskipun efektif dalam pengelolaan data, pendekatan ini memiliki keterbatasan dari sisi aksesibilitas dan mobilitas, karena pengguna hanya dapat mengakses sistem dari perangkat tertentu di lokasi tetap.

Fachrozy, M. I., & Batubara [8] mencoba menjawab tantangan mobilitas dengan mengembangkan aplikasi POS berbasis mobile Android. Sistem ini menawarkan kemudahan akses melalui perangkat seluler, namun belum mengakomodasi fitur pelaporan yang komprehensif dan tidak mendukung manajemen multi-user, yang penting untuk koordinasi staf dalam usaha ritel. Penelitian Mulyani dan Najiyah [9] juga mengembangkan aplikasi penjualan berbasis scanner berbasis Android.

Marroquín et al. [14] mengeksplorasi praktik terbaik dalam integrasi PayPal pada sistem POS berbasis PHP, dengan fokus pada audit dan penguatan aspek keamanan transaksi. Sibagariang & Sinaga [15] mengidentifikasi masalah ketidakefisienan proses transaksi penjualan pada sistem manual. Penelitian menggunakan metode waterfall dengan *framework* CodeIgniter untuk mengembangkan sistem POS berbasis web. Saan [16] mengatasi permasalahan operasional warung kopi yang masih menggunakan proses manual dalam pencatatan transaksi. Metode pengembangan menggunakan SDLC dengan *framework* CodeIgniter berbasis web. Ramadhan [17] menyelesaikan masalah ketidakefisienan transaksi pada pasar swalayan yang masih menggunakan sistem konvensional. Metode penelitian menggunakan *framework* CodeIgniter dengan arsitektur MVC untuk mengoptimalkan proses penjualan. Hasil penelitian membuktikan sistem dapat meningkatkan kecepatan transaksi

dan akurasi pencatatan pada lingkungan ritel besar.

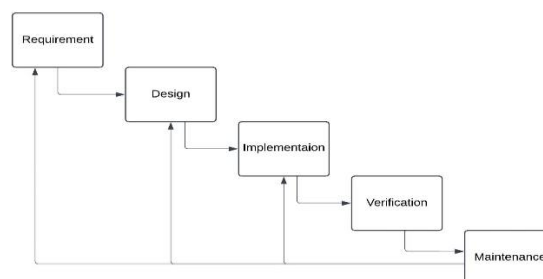
Lim & Ridho [18] mengidentifikasi permasalahan pengelolaan inventori manual pada CV POWERSHOP yang menyebabkan kesulitan monitoring stok barang. Penelitian menggunakan metodologi UML dengan bahasa pemrograman PHP dan XAMPP sebagai database. Hasil implementasi menunjukkan peningkatan efisiensi pengelolaan data inventori dan kemudahan akses informasi stok real-time. Maydianto [19] mengatasi permasalahan CV *Powershop* yang masih menggunakan proses penjualan konvensional melalui media sosial tanpa sistem persediaan. Penelitian menggunakan framework CodeIgniter dengan metode pengembangan terstruktur. Sistem yang dihasilkan mampu mengintegrasikan proses penjualan online dengan manajemen inventory yang efektif.

Berdasarkan kajian literatur tersebut, dapat disimpulkan bahwa penelitian sebelumnya telah berhasil menggarap berbagai aspek penting sistem POS, mulai dari antarmuka yang ramah pengguna, manajemen inventaris real-time, integrasi pembayaran daring, hingga optimalisasi keamanan. Namun demikian, mayoritas studi masih mengembangkan solusi secara terpisah untuk masing-masing fitur. Penelitian terkini belum banyak yang menyatukan seluruh kebutuhan UKM mulai dari multi-store, real-time reporting, inventory adjustment, keamanan berlapis, hingga analitik prediktif ke dalam satu platform POS berbasis CodeIgniter secara holistik dan modular. Penelitian ini diharapkan dapat berkontribusi dalam pengembangan sistem POS berbasis CodeIgniter yang mengintegrasikan secara komprehensif manajemen inventori real-time, pencatatan transaksi penjualan, pelaporan keuangan komprehensif, dan sistem multi-user dengan tingkat akses yang berbeda, yang terintegrasi dalam satu sistem informasi yang efisien, aman, dan dapat diakses dari berbagai perangkat, khusus untuk kebutuhan UKM khususnya ROS Minimarket.

3. METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan dalam pengembangan sistem informasi POS (*Point of Sales*) ini adalah model *Waterfall*, yang merupakan salah satu model pengembangan perangkat lunak yang sistematis dan berurutan.

Model ini dipilih karena memberikan pendekatan yang terstruktur, di mana setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya. Tahapan dalam metode ini meliputi analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan [20].

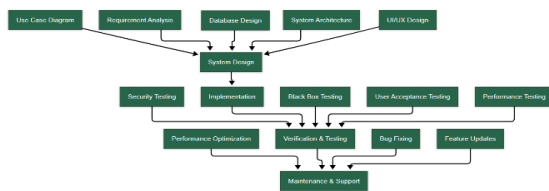


Gambar 1. Metode *Waterfall*

Pemilihan model Waterfall untuk proyek pengembangan sistem POS UKM didasari oleh analisis karakteristik proyek dan kebutuhan spesifik ROS Minimarket. *Stabilitas requirement* menjadi faktor kunci karena sistem POS memiliki kebutuhan yang relatif stabil dan dapat didefinisikan dengan baik sejak awal, tidak seperti aplikasi yang sering mengalami perubahan requirement. Keterbatasan budget UKM membuat predictable cost dari model Waterfall sangat menguntungkan, memungkinkan penetapan fixed price contract yang memberikan certainty bagi kedua belah pihak dan mengurangi risiko cost overrun.

Tingkat pemahaman klien dan compliance requirement menjadi pertimbangan penting lainnya. ROS Minimarket sebagai UKM memiliki keterbatasan technical knowledge, sehingga model Waterfall dengan milestone yang jelas memberikan comfort level yang diperlukan untuk memahami progress proyek. Sistem POS harus memenuhi berbagai regulasi perpajakan dan pencatatan transaksi, dimana pendekatan sequential memungkinkan analisis compliance yang thorough di fase awal. Aspek risk management menjadi crucial mengingat dampak langsung sistem terhadap operasional harian, dan model Waterfall dengan testing phase yang dedicated memberikan assurance untuk meminimalkan production risk melalui comprehensive testing sebelum go-live.

3.1. Skema Alur Penelitian



Gambar 2. Skema Alur Penelitian

3.2. Requirement Analysis

Tahap pertama dalam metode *Waterfall* adalah Analisis Kebutuhan (*Requirement Analysis*), yang mencakup studi menyeluruh terhadap kebutuhan pengguna dan kelayakan sistem, baik dari aspek teknis maupun teknologi [21]. Dalam tahap ini dilakukan:

1. Observasi Langsung: Pengamatan sistem manual yang sedang berjalan di ROS Minimarket selama 2 minggu.
2. Wawancara: Wawancara dengan pemilik dan kasir untuk mengidentifikasi kebutuhan sistem.
3. Analisis Proses Bisnis: Mapping seluruh proses transaksi, *inventory management*, dan pelaporan.
4. Identifikasi Masalah: Dokumentasi seluruh permasalahan dalam sistem manual *existing*.

Hasil dari tahap ini berupa dokumen spesifikasi kebutuhan perangkat lunak *Software Requirement Specification* (SRS), yang mendefinisikan seluruh fitur dan fungsi yang akan diimplementasikan dalam sistem.

3.3. System Design

Tahap selanjutnya adalah Perancangan Sistem (*System Design*), yang bertujuan untuk mengonversi kebutuhan yang telah dikumpulkan ke dalam bentuk blueprint perangkat lunak sebelum proses pengkodean dimulai [22]. Pada tahap ini, dilakukan:

1. Database Design: Perancangan struktur database menggunakan *Data Flow Diagram* (DFD).
2. System Architecture: Perancangan arsitektur sistem menggunakan pattern MVC (*Model-View-Controller*).
3. UI/UX Design: Perancangan mockup dan wireframe untuk *user interface*.
4. Use Case Design: Pembuatan *use case diagram* dan *activity diagram*.

3.4. Implementation

Setelah tahap perancangan selesai, tahap berikutnya adalah Implementasi (*Implementation*), yaitu proses di mana sistem

yang telah dirancang diterjemahkan ke dalam kode program menggunakan *framework* CodeIgniter [23]. Teknologi yang digunakan:

1. Backend: PHP 7.4+ dengan CodeIgniter 4.x.
2. Frontend: HTML5, CSS3, JavaScript ES6, Bootstrap 5.
3. Database: MySQL 8.0.
4. Additional Libraries: jQuery, Chart.js, DataTables.

3.5. Verification dan Testing

Tahap selanjutnya adalah Verifikasi dan Pengujian (*Verification dan Testing*), yang dilakukan untuk memastikan bahwa perangkat lunak yang dikembangkan berfungsi sesuai dengan spesifikasi yang telah ditentukan serta bebas dari kesalahan atau bug [24]. Metode pengujian yang digunakan:

1. *Black Box Testing*: Pengujian fungsionalitas sistem tanpa melihat *source code*.
2. *User Acceptance Testing* (UAT): Pengujian oleh *end-user* untuk validasi kebutuhan bisnis.
3. *Performance Testing*: Pengujian kinerja sistem dengan berbagai beban kerja.
4. *Security Testing*: Pengujian keamanan sistem terhadap berbagai ancaman.

3.6. Maintenance dan Support

Tahap terakhir dalam model *Waterfall* adalah Pemeliharaan dan Perawatan (*Maintenance dan Support*), yang dilakukan setelah sistem berhasil diimplementasikan untuk memastikan bahwa sistem tetap berjalan dengan optimal [17].

4. HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk memahami fitur dan spesifikasi yang diperlukan dalam sistem informasi POS berbasis CodeIgniter [25]. Studi awal dilakukan melalui observasi langsung terhadap sistem kasir di ROS Minimarket selama 2 minggu dan wawancara dengan pemilik dan kasir.

4.1.1 Hasil Observasi dan Wawancara

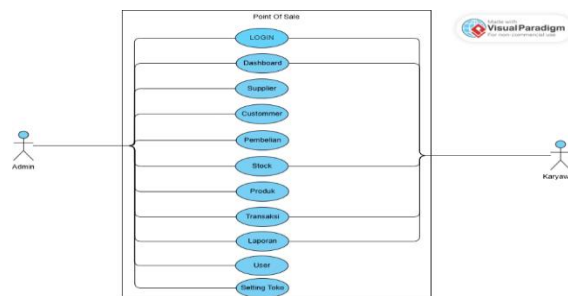
Berdasarkan hasil observasi dan wawancara yang telah dilakukan, ditemukan beberapa permasalahan utama dalam proses operasional sistem penjualan yang masih dilakukan secara manual. Rata-rata waktu yang dibutuhkan untuk menyelesaikan satu transaksi berkisar antara 3 hingga 5 menit per pelanggan, yang menyebabkan antrean panjang dan efisiensi kerja yang rendah. Selain itu, terdapat tingkat

kesalahan pencatatan yang cukup tinggi, yaitu sekitar 15% dari total transaksi harian. Laporan penjualan pun mengalami keterlambatan, dengan rata-rata waktu penyajian 2 hingga 3 hari setelah akhir periode pelaporan. Permasalahan lainnya adalah kehilangan data transaksi yang mencapai 5% per bulan, serta kesulitan dalam melakukan pelacakan stok barang akibat proses inventory count yang masih dilakukan secara manual dan membutuhkan waktu antara 6 hingga 8 jam.

Dari berbagai permasalahan tersebut, diidentifikasi sejumlah kebutuhan fungsional yang perlu dipenuhi oleh sistem yang akan dikembangkan. Kebutuhan tersebut meliputi sistem *login* dan manajemen pengguna dengan akses berbasis peran (*role-based access*), pengelolaan data master seperti produk, kategori, dan supplier, serta dukungan transaksi penjualan melalui pemindaian *barcode*. Selain itu, sistem juga harus mendukung manajemen inventori secara *real-time*, menyediakan fitur pelaporan harian, mingguan, dan bulanan, serta memiliki kemampuan untuk melakukan backup dan restore data. Sementara itu, kebutuhan non-fungsional yang teridentifikasi mencakup waktu respon transaksi yang kurang dari 2 detik, tingkat ketersediaan sistem sebesar 99.9% (dengan downtime maksimal 8.76 jam per tahun), dukungan terhadap hingga 10 pengguna yang mengakses secara bersamaan, keamanan data dengan enkripsi, serta antarmuka pengguna yang ramah dan mudah dipelajari dengan kurva belajar kurang dari 30 menit.

4.1.2 Use Case Diagram

Diagram *Use Case* pada sistem *Point of Sale* (POS) menggambarkan interaksi antara pengguna dengan berbagai fungsionalitas sistem. Diagram ini penting karena menyajikan gambaran menyeluruh mengenai bagaimana sistem beroperasi dari perspektif pengguna, tanpa masuk ke aspek teknis. Dalam kasus ini, sistem POS dirancang untuk mendukung operasional toko retail modern dengan melibatkan dua aktor utama admin dan karyawan yang berinteraksi dengan sebelas fitur inti seperti *login*, dashboard, transaksi, laporan, dan pengaturan toko.



Gambar 3. Use Case Diagram

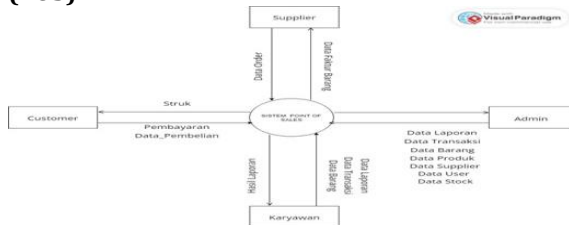
Analisis terhadap diagram menunjukkan bahwa admin memiliki akses penuh terhadap seluruh fitur sistem, termasuk pengelolaan pengguna dan konfigurasi toko, sementara karyawan berperan dalam aktivitas operasional seperti melayani transaksi dan manajemen stok. Meski dalam diagram semua fitur terlihat dapat diakses oleh kedua aktor, implementasi sebenarnya dapat menerapkan pembatasan hak akses sesuai peran. Setiap fitur yang tersedia, mulai dari *login* hingga laporan dan pengaturan toko, dirancang saling terintegrasi untuk menciptakan sistem yang efisien dan mendukung pengambilan keputusan strategis.

Seluruh *use case* dalam sistem POS ini saling berhubungan membentuk ekosistem yang mendukung efisiensi operasional, konsistensi data, dan kemudahan analisis bisnis. Integrasi antar modul seperti *supplier* dengan pembelian, produk dengan transaksi, dan customer dengan laporan, menunjukkan adanya sinergi antarfungsi yang kuat. Dengan pendekatan *user-centric* dan fleksibilitas dalam pengaturan otorisasi, diagram *use case* ini tidak hanya mendefinisikan fungsionalitas, tetapi juga mencerminkan pemahaman terhadap kebutuhan pengguna dan dinamika bisnis retail.

4.2 Perancangan Sistem

Sistem *Point of Sale* (POS) yang dirancang menggunakan arsitektur *Model-View-Controller* (MVC) yang didukung oleh *framework* CodeIgniter. Dalam tahap perancangan ini, sistem dibagi menjadi beberapa komponen utama yang mencakup antarmuka pengguna, struktur basis data, dan alur transaksi. Tujuannya adalah untuk menghasilkan sistem yang modular, mudah dikembangkan, dan terstruktur dengan baik. Desain sistem ini mendukung integrasi antara logika bisnis, lapisan presentasi, dan pengelolaan data secara efisien [26].

4.2.1 Diagram Konteks Sistem Point Of Sale (POS)



Gambar 4. Diagram Konteks Sistem POS

Sistem *Point of Sale* (POS) merupakan jantung operasional sebuah toko atau bisnis retail yang menghubungkan berbagai pihak dalam proses penjualan dan manajemen inventori. Dalam diagram konteks ini, sistem POS berperan sebagai pusat koordinasi yang mengelola aliran informasi antara empat entitas eksternal utama.

Supplier sebagai pemasok barang memiliki hubungan fundamental dengan sistem POS melalui penyediaan data barang. Informasi yang diberikan *supplier* mencakup detail produk, spesifikasi, harga, dan ketersediaan stok. Data ini menjadi dasar bagi sistem untuk mengelola inventori dan menentukan produk apa saja yang tersedia untuk dijual kepada pelanggan.

Customer atau pelanggan merupakan *end-user* dari sistem POS yang menerima output berupa struk pembelian. Struk ini bukan sekadar bukti transaksi, tetapi juga berisi informasi detail mengenai produk yang dibeli, harga, jumlah, total pembayaran, dan informasi relevan lainnya yang diperlukan pelanggan untuk keperluan administrasi atau klaim garansi.

Karyawan bertindak sebagai operator utama sistem POS dalam aktivitas sehari-hari. Mereka berinteraksi dengan sistem melalui proses *login* untuk autentikasi, kemudian menggunakan interface sistem untuk melakukan berbagai transaksi penjualan. Data transaksi yang dihasilkan dari aktivitas karyawan menjadi input penting bagi sistem untuk mengupdate stok, menghitung omzet, dan menghasilkan laporan.

Admin memiliki peran strategis dalam pengelolaan sistem POS secara keseluruhan. Mereka menerima berbagai jenis laporan dari sistem, termasuk data laporan penjualan, data transaksi harian, data barang dan inventori, data keuangan, informasi supplier, data pengguna sistem, dan data stok *real-time*. Informasi komprehensif ini memungkinkan admin untuk membuat keputusan bisnis yang tepat,

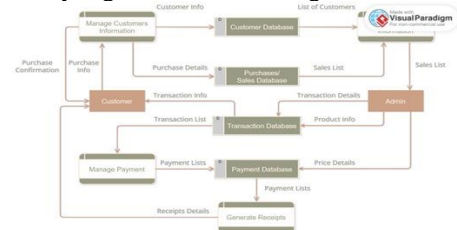
melakukan analisis performa, dan merencanakan strategi ke depan.

Aliran data dalam sistem ini bersifat sirkular dan saling mendukung. Supplier menyediakan informasi barang yang dikelola sistem, karyawan mengoperasikan sistem untuk melayani customer, customer menerima bukti transaksi, dan admin mendapatkan insight menyeluruh dari seluruh aktivitas sistem. Integrasi ini menciptakan ekosistem bisnis yang efisien dan termonitor dengan baik.

Sistem POS ini dirancang untuk mengotomatisasi proses penjualan, mengurangi kesalahan manual, mempercepat layanan pelanggan, dan menyediakan data akurat untuk pengambilan keputusan bisnis. Dengan adanya sistem terpusat ini, semua stakeholder dapat berinteraksi secara harmonis dalam mencapai tujuan bisnis yang optimal.

4.2.2 Data Flow Diagram (DFD)

Sistem POS dirancang dengan alur data yang menggambarkan interaksi antara *external entity*, *process*, dan *data store*. DFD level 1 menunjukkan gambaran umum sistem dengan empat *external entity* utama: Customer, Admin, Manager, dan Sales List yang berinteraksi dengan sistem POS.



Gambar 5. Data Flow Diagram Level 1

Berdasarkan diagram diatas menunjukkan alur data dari Customer melakukan *Purchase* ke sistem, Admin mengelola *Customer Database* dan *Transaction Database*, Manager mengakses *Sales List* dan *Reports*, serta interaksi dengan *Payment Database* untuk *Generate Receipts*.

Komponen utama dalam DFD meliputi:

1. *External Entity*: Customer (pelanggan yang melakukan transaksi), Admin (pengelola sistem), Manager (pengakses laporan), Sales List (output penjualan).
2. *Data Store*: Customer Database, Transaction Database, Payment Database.
3. *Process*: Manage Customers Information, Purchase Confirmation, Transaction Info, Manage Payment, Generate Receipts.

4. *Data Flow: Customer Info, Purchase Details, Transaction List, Payment Lists, Receipts Details, Product Info, Price Details.*

Alur data dimulai dari Customer yang memberikan *Purchase Confirmation* ke sistem, kemudian sistem memproses *Transaction Info* dan menyimpannya ke *Transaction Database*. Admin mengelola *Customer Database* dan dapat mengakses *Purchase/Sales Database*. Manager memiliki akses ke *Sales List* dan dapat melihat *Product Info* serta *Price Details*. Sistem juga terintegrasi dengan *Payment Database* untuk memproses pembayaran dan *Generate Receipts*.

4.2.2 System Architecture

Arsitektur sistem didesain menggunakan pola MVC, yang memisahkan tanggung jawab komponen ke dalam tiga lapisan. Lapisan *Model* bertugas menangani logika bisnis serta operasi basis data, sedangkan *View* bertanggung jawab terhadap antarmuka pengguna dan penyajian informasi. Sementara itu, *Controller* mengatur alur komunikasi antara *Model* dan *View*, serta menangani permintaan dari pengguna. Dengan pendekatan ini, sistem menjadi lebih terorganisir dan mudah dalam hal pemeliharaan serta pengembangan fitur di masa mendatang.

4.3 Implementasi

Tahap implementasi merupakan proses pengkodean berdasarkan rancangan sistem yang telah dibuat sebelumnya. Sistem dikembangkan menggunakan teknologi yang sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan pada tahap desain. Selama proses ini, berbagai fitur utama direalisasikan dan diuji agar dapat berfungsi sesuai harapan [27].

4.3.1 Fitur-Fitur yang Diimplementasikan

Sistem informasi kasir berbasis web yang dikembangkan memiliki beragam fitur terintegrasi yang dirancang untuk mengoptimalkan operasional bisnis UKM. Aspek manajemen transaksi menjadi inti dari sistem ini, dimana setiap transaksi penjualan dapat diproses secara real-time dengan pembaruan stok otomatis melalui teknologi barcode scanning. Sistem mendukung fleksibilitas metode pembayaran tunai, kartu, dan transfer digital, serta menghasilkan struk pembayaran otomatis dengan format yang dapat disesuaikan sesuai kebutuhan bisnis.

Pengelolaan inventori dan sistem pelaporan terintegrasi menyediakan pelacakan stok real-time dengan notifikasi otomatis untuk batas

minimum stok dan penerapan metode FIFO untuk rotasi produk optimal. Dashboard keuangan menampilkan informasi revenue, profit, dan cash flow secara real-time, sementara laporan penjualan harian memberikan breakdown detail per kategori produk. Sistem juga dilengkapi dengan manajemen multi-user berlapis untuk berbagai level akses, fitur shift management dengan tracking waktu kerja, serta aspek keamanan komprehensif yang mencakup enkripsi data, sistem backup otomatis, dan audit trail untuk memastikan akuntabilitas operasional.

Fitur pertama yang diimplementasikan adalah sistem autentikasi dan otorisasi, yang mencakup *login* dengan manajemen sesi dan kontrol akses berbasis peran (Admin, Kasir, Manager), serta enkripsi password menggunakan algoritma bcrypt. Fitur berikutnya adalah manajemen data master, yang mendukung operasi CRUD untuk data produk, kategori, dan supplier, lengkap dengan fitur pembuatan barcode dan unggah gambar produk. Untuk transaksi penjualan, sistem menyediakan antarmuka POS dengan pemindaian barcode, perhitungan harga secara *real-time*, dukungan berbagai metode pembayaran, serta pencetakan struk. Fitur manajemen inventori juga disediakan, dengan kemampuan pelacakan stok secara *real-time*, pemberitahuan untuk stok rendah, penyesuaian stok, dan pengelolaan data supplier. Terakhir, sistem pelaporan mendukung pembuatan laporan penjualan harian, mingguan, dan bulanan, laporan inventori, analisis produk terlaris, serta analitik pendapatan dalam bentuk grafik interaktif.

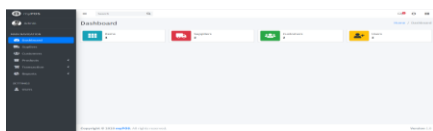
4.3.2 Screenshot Implementasi



Gambar 6. Tampilan Halaman *Login*

Gambar 6 menunjukkan halaman *login* yang dirancang untuk memastikan akses pengguna yang aman. Pada halaman ini, pengguna harus memasukkan kredensial berupa *username* dan password yang telah terdaftar dalam sistem. Implementasi sistem *login* ini menggunakan autentikasi berbasis sesi, sehingga hanya pengguna yang memiliki izin yang dapat mengakses sistem. Dengan fitur ini, keamanan

data transaksi dapat terjaga, dan hanya pengguna yang memiliki wewenang tertentu yang bisa mengelola informasi dalam sistem.



Gambar 7. Tampilan Dashboard

Gambar 7 menampilkan halaman dashboard yang menjadi pusat informasi bagi pengguna. Dashboard ini menyajikan data transaksi secara *real-time*, seperti jumlah transaksi harian, produk terlaris, serta total pendapatan dalam periode tertentu. Desain dashboard dibuat dengan tampilan yang informatif dan mudah dipahami oleh pengguna, sehingga pemilik usaha dapat langsung melihat performa penjualan mereka tanpa harus melakukan pencarian data secara manual.



Gambar 8. Form Tambah Pengguna

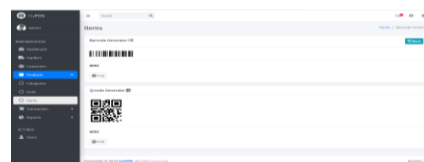
Gambar 8 menunjukkan fitur tambah pengguna yang dibuat menggunakan modal dan AJAX. Fitur ini memungkinkan admin untuk menambahkan pengguna baru tanpa perlu melakukan refresh halaman, sehingga mempercepat dan meningkatkan efisiensi dalam pengelolaan data pengguna. Dengan adanya fitur ini, administrator dapat dengan mudah mengatur hak akses bagi setiap pengguna, memastikan bahwa hanya pengguna tertentu yang memiliki izin untuk mengelola berbagai fitur dalam sistem.



Gambar 9. Konfirmasi Penghapusan Data

Gambar 9 menunjukkan fitur konfirmasi penghapusan data yang menggunakan *Sweet Alert 2*. Fitur ini dirancang untuk mencegah penghapusan data yang tidak disengaja. Sebelum pengguna menghapus data, sistem akan menampilkan jendela konfirmasi yang meminta persetujuan akhir. Hal ini membantu pengguna menghindari kehilangan data yang tidak

diinginkan dan meningkatkan keamanan dalam pengelolaan informasi.



Gambar 10. Barcode dan QR Code Generator

Gambar 10 menunjukkan fitur *Barcode dan QR Code Generator*, yang memudahkan dalam identifikasi produk dan mempercepat proses transaksi di kasir. Dengan adanya fitur ini, setiap produk dapat memiliki kode unik yang dapat dipindai menggunakan *scanner*, sehingga mempercepat proses *checkout* pelanggan dan mengurangi kemungkinan kesalahan input data oleh kasir.

4.4 Pengujian

Setelah proses implementasi sistem selesai dilaksanakan, tahap selanjutnya adalah melakukan pengujian secara menyeluruh untuk memastikan bahwa sistem berfungsi sesuai dengan yang diharapkan dan memiliki tingkat keandalan yang tinggi. Pengujian dilakukan melalui beberapa metode, yakni *black box testing*, *user acceptance testing* (UAT), dan *performance testing*, guna mengevaluasi berbagai aspek mulai dari fungsionalitas, pengalaman pengguna, hingga performa sistem dalam kondisi beban tinggi [28].

4.4.1 Black Box Testing

Pengujian fungsional menggunakan metode *black box testing* difokuskan pada pengujian input dan output sistem tanpa melihat kode program secara langsung. Beberapa fitur utama yang diuji antara lain sistem *login*, penambahan produk, proses transaksi, dan pembuatan laporan [29]. Berikut adalah tabel hasil *Black Box Testing*:

Tabel 1. Hasil *Black Box Testing*

No	Fitur	Test Case	Expected Result	Actual Result	Status
1	Login	Valid credentials	Login berhasil, redirect ke dashboard	Sesuai expected	Sesuai
2	Login	Invalid credentials	Error message, tetap di	Sesuai expected	Sesuai



			halaman <i>login</i>		
3	<i>Add Product</i>	Valid data	Produk tersimpan, muncul di daftar	Sesuai expected	Sesuai
4	<i>Process Transaction</i>	Normal transaction	Transaksi berhasil, stock terupdate	Sesuai expected	Sesuai
5	<i>Generate Report</i>	Select date range	Report ter-generate sesuai periode	Sesuai expected	Sesuai

Dari hasil pengujian, sistem menunjukkan performa yang sangat baik. Saat diuji dengan kredensial yang valid, sistem berhasil melakukan *login* dan langsung mengarahkan pengguna ke halaman *dashboard*. Sebaliknya, jika menggunakan kredensial tidak valid, sistem menampilkan pesan *error* dan tetap berada di halaman *login* sebagaimana mestinya. Penambahan data produk dengan input yang valid berhasil menyimpan data dan menampilkannya di daftar produk. Proses transaksi normal pun berjalan lancar, dengan stok produk yang otomatis terbaru setelah transaksi selesai. Fitur pelaporan juga berhasil menghasilkan laporan berdasarkan rentang tanggal yang dipilih. Secara keseluruhan, dari 25 skenario pengujian yang dijalankan, seluruhnya berhasil melewati uji coba tanpa masalah, menghasilkan tingkat keberhasilan 100%.

4.4.2 User Acceptance Testing (UAT)

Untuk mengukur kepuasan pengguna akhir terhadap sistem yang dikembangkan, dilakukan *User Acceptance Testing* (UAT) yang melibatkan pemilik minimarket dan kasir. Mereka menguji sistem secara langsung dan memberikan penilaian terhadap beberapa aspek [30]. Berikut adalah rangkuman hasil *User Acceptance Testing* (UAT):

Tabel 2. Hasil *User Acceptance Testing* (UAT)

Aspek	Skor (1-5)	Keterangan
Kemudahan Penggunaan	4.5	<i>Interface intuitive</i> , mudah dipelajari
Kecepatan Sistem	4.3	<i>Response time</i> rata-rata 1.2

		detik
Keakuratan Data	4.8	Tidak ada kesalahan kalkulasi ditemukan
Fitur Kelengkapan	4.2	Semua kebutuhan dasar terpenuhi
Overall Satisfaction	4.45	Sangat memuaskan

Berdasarkan hasil pengujian, sistem memperoleh skor rata-rata yang sangat baik. Kemudahan penggunaan mendapat nilai 4.5 dari 5, dengan catatan bahwa antarmuka sistem dinilai intuitif dan mudah dipelajari. Kecepatan sistem mendapat skor 4.3, dengan waktu respons rata-rata sebesar 1,2 detik per transaksi. Dalam hal keakuratan data, skor mencapai 4.8 karena selama pengujian tidak ditemukan adanya kesalahan kalkulasi. Kelengkapan fitur memperoleh nilai 4.2 karena semua kebutuhan dasar operasional telah tersedia dan berfungsi dengan baik. Secara keseluruhan, tingkat kepuasan pengguna mencapai skor 4.45 yang menunjukkan bahwa sistem sudah memenuhi ekspektasi mereka. Beberapa tanggapan positif dari pengguna antara lain menyebutkan bahwa sistem sangat mudah digunakan, laporan otomatis sangat membantu sehingga tidak perlu lagi melakukan perhitungan manual, dan penggunaan pemindai barcode mampu mempercepat proses transaksi hingga tiga kali lipat dibandingkan sebelumnya.

4.4.3 Performance Testing

Pengujian performa dilakukan melalui skenario *load testing* dan *stress testing* untuk mengevaluasi bagaimana sistem merespon dalam kondisi beban normal maupun ekstrem [31]. Dalam pengujian *load testing*, sistem diuji dengan 10 pengguna yang mengakses secara bersamaan. Hasilnya menunjukkan waktu respon rata-rata sebesar 1.2 detik, waktu respon maksimum mencapai 2.8 detik, tanpa adanya error (0% error rate), serta throughput sebesar 45 permintaan per detik. Selanjutnya, pada *stress testing*, sistem menunjukkan stabilitas hingga pada jumlah maksimum 25 pengguna bersamaan. Sistem tetap berjalan stabil hingga 20 pengguna aktif secara bersamaan, dengan penggunaan memori puncak tercatat sebesar 512 MB RAM. Hasil ini



menunjukkan bahwa sistem mampu beroperasi secara konsisten dalam kondisi beban tinggi dan memiliki performa yang andal untuk operasional skala kecil hingga menengah.

4.5 Pembahasan

4.5.1 Analisis Peningkatan Efisiensi

Implementasi sistem POS berbasis CodeIgniter pada ROS Minimarket menunjukkan peningkatan efisiensi operasional yang signifikan dibandingkan dengan sistem manual yang sebelumnya digunakan. Evaluasi dilakukan dengan membandingkan performa sistem lama dan baru pada empat metrik utama: waktu per transaksi, tingkat error, waktu pembuatan laporan, dan waktu inventory count. Berikut merupakan ringkasan tabel yang menunjukkan perbedaan efisiensi Sistem Manual dan POS.

Tabel 3. Perbandingan Efisiensi Sistem Manual dan POS

Metrik	Sistem Manual	Sistem POS	Improvement
Waktu per Transaksi	3-5 menit	1-2 menit	60% lebih cepat
Tingkat Error	15%	2%	86% pengurangan error
Waktu Buat Laporan	2-3 hari	<i>Real-time</i>	99% lebih cepat
Waktu Inventory Count	6-8 jam	30 menit	93% lebih cepat

Implementasi sistem POS berbasis CodeIgniter pada ROS Minimarket menunjukkan peningkatan efisiensi operasional yang signifikan. Waktu per transaksi berkurang 60% dari 3-5 menit menjadi 1-2 menit melalui barcode scanning dan perhitungan otomatis. Tingkat kesalahan pencatatan turun drastis dari 15% menjadi 2% berkat otomatisasi proses dan validasi input real-time. Pembuatan laporan yang sebelumnya membutuhkan 2-3 hari kini dapat dilakukan secara real-time, sementara proses inventory count dipercepat dari 6-8 jam menjadi hanya 30 menit.

4.5.2 Analisis Keamanan

Aspek keamanan merupakan prioritas Sistem menerapkan keamanan berlapis untuk melindungi data sensitif transaksi dan informasi bisnis. Implementasi meliputi input validation komprehensif untuk mencegah SQL injection, automatic filtering untuk perlindungan XSS, dan

token-based authentication untuk mencegah CSRF. Session security diterapkan dengan timeout otomatis dan regenerasi session ID, sementara password security menggunakan algoritma bcrypt dengan salt untuk mencegah rainbow table attack.

4.5.3 Analisis Skalabilitas

Desain sistem mempertimbangkan pertumbuhan bisnis masa depan melalui database design yang ternormalisasi dengan proper indexing, arsitektur modular dengan loose coupling untuk kemudahan maintenance, dan performance optimization melalui query optimization dan caching mechanism. Sistem juga dirancang dengan multi-tenant readiness untuk mendukung ekspansi ke multiple store dalam satu instance aplikasi dengan tenant isolation dan resource sharing yang efisien.

5. Kesimpulan dan Saran

Berdasarkan hasil penelitian dan implementasi yang telah dilakukan, sistem Point of Sale (POS) berbasis CodeIgniter berhasil dikembangkan dengan tingkat kepuasan pengguna 4.45/5.0 dan tingkat keberhasilan pengujian 100%. Sistem ini terbukti meningkatkan efisiensi operasional secara signifikan dengan pengurangan waktu transaksi sebesar 60%, tingkat error sebesar 86%, dan percepatan pembuatan laporan hingga 99% dibandingkan sistem manual, serta telah mengintegrasikan fitur-fitur seperti manajemen inventory *real-time*, *barcode scanning*, *reporting* otomatis, dan *role-based access control* dengan kinerja optimal. Penelitian ini secara ilmiah berkontribusi dalam menyediakan model sistem informasi POS yang efisien, aman, dan skalabel, yang secara khusus disesuaikan untuk kebutuhan operasional Usaha Kecil dan Menengah (UKM) di era digital.

Untuk pengembangan selanjutnya, disarankan untuk menambahkan integrasi sistem pembayaran digital (misalnya QRIS atau e-wallet), implementasi mobile application (native Android/iOS) untuk fleksibilitas akses yang lebih tinggi, dan pengembangan dashboard analytics yang lebih komprehensif dengan fitur prediksi penjualan dan analisis tren. Selain itu, perlu dilakukan optimisasi performa sistem secara berkala untuk mendukung peningkatan volume transaksi di masa depan, serta implementasi



backup dan disaster recovery plan yang lebih optimal untuk menjamin keberlanjutan

DAFTAR PUSTAKA

- [1] R. A. Purba dkk, "Aplikasi teknologi informasi: teori dan implementasi," 2020.
- [2] V. Widiyanti dan R. Tisnawati, "Perancangan Sistem Informasi Point of Sale di PT. Abercode Software Berbasis Web," *Jurnal Sosial Teknologi*, vol. 4, no. 10, hlm. 838–850, 2024.
- [3] B. Y. Geni, D. Ramayanti, dan A. Ratnasari, "IMPLEMENTASI SISTEM POIN OF SALE TERINTEGRASI BERBASIS PYTHON," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 4, hlm. 4387–4393, 2024.
- [4] M. Rahmatuloh, M. F. A. Kamil, dan S. Nirwan, "RANCANG BANGUN APLIKASI POINT OF SALE TOKO FASHION BERBASIS WEBSITE," *Jurnal Teknik Informatika*, vol. 16, no. 3, hlm. 127–135, 2024.
- [5] W. M. Kansha, "Analisis Perbandingan Framework Codeigniter Dan Laravel Dalam Pengembangan Web Application," *Jurnal Teknik Informatika*, vol. 9, no. 1, hlm. 27–33, 2023.
- [6] A. Sani, L. Judijanto, A. Y. Rukmana, K. A. Dwiwijaya, I. W. Suparno, dan A. Andrianingsih, *Sistem Informasi untuk Transformasi Bisnis Modern*. PT. Sonpedia Publishing Indonesia, 2025.
- [7] A. A. Mehetabel, W. F. Senjaya, dan S. L. Liliawati, "Penerapan Framework CodeIgniter 4 dan Bootstrap dalam Pengembangan Situs Reservasi Aula Karunia Global School," *Jurnal STRATEGI-Jurnal Maranatha*, vol. 6, no. 1, hlm. 1–11, 2024.
- [8] M. I. Fachrozy dan R. O. Batubara, "Aplikasi Berbasis Mobile Pada Perancangan Point Of Sale (POS)," dipresentasikan pada PROSIDING SEMINAR NASIONAL MULTI DISIPLIN ILMU (SENADIMU), 2024, hlm. 1446–1458.
- [9] N. S. Mulyani dan I. Najiyah, "Rancang Bangun Aplikasi Penjualan Menggunakan Barcode Scanner Berbasis Android," *Jurnal Responsif: Riset Sains dan Informatika*, vol. 4, no. 1, hlm. 49–55, 2022.
- [10] A. Ammar, M. Z. Maulana, T. J. Novella, S. Hawari, dan A. Nuraminah, "Perancangan operasional.
- Dan Implementasi Catering Ordering System 'Hena Catering' Menggunakan Rapid Application Development," dipresentasikan pada Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya, 2021, hlm. 31–39.
- [11] R. A. Purba dkk, *Pengembangan Sistem Informasi: Analisis, Pemodelan, dan Perangkat Lunak*. Yayasan Kita Menulis, 2022.
- [12] C. Cakrawati, A. F. Rochim, dan I. P. Windasari, "Rancang Bangun dan Implementasi Sistem Informasi Manajemen Penjualan Berbasis Website Menggunakan Kerangka Kerja CodeIgniter pada Kafe Dangau Kopi di Daerah Gunung Pati, Kota Semarang, Jawa Tengah," *Jurnal Teknik Komputer*, vol. 2, no. 1, hlm. 31–38, 2023.
- [13] Y. Aprizal dan E. Hartati, "Implementasi Metode Waterfall dalam Merancang Sistem Digitalisasi Ruang Baca," *JUPITER: Jurnal Penelitian Ilmu dan Teknologi Komputer*, vol. 15, no. 2, hlm. 1191–1203, 2023.
- [14] I. Marroquín, G. Casco, dan M. Cardona, "Best Practices for PayPal Integration: An Audit and Security-Based Approach," dipresentasikan pada 2024 IEEE Central America and Panama Student Conference (CONESCAPAN), IEEE, 2024, hlm. 1–4.
- [15] S. Sibagariang dan M. E. P. Sinaga, "Rancang Bangun Aplikasi Point Of Sales Menggunakan Framework Codeigniter," *JURNAL MAHAJANA INFORMASI*, vol. 8, no. 1, hlm. 26–35, 2023.
- [16] M. Y. Saan, Y. A. Dalimunthe, dan D. Irwan, "Rancang Bangun Aplikasi Point of Sale Penjualan Kopi Dengan Menggunakan Framework Codeigniter Berbasis Web," *Syntax J. Softw. Eng. Comput. Sci. Inf. Technol*, vol. 4, no. 1, hlm. 314–319, 2023.
- [17] M. G. Ramadhan, "Perancangan Sistem Informasi Pos (Point of Sales) Berbasis Web Dengan Menggunakan Framework Codeigniter Pada Pasar Swalayan," *Electrician: Jurnal Rekayasa dan Teknologi Elektro*, vol. 14, no. 3, hlm. 76–83, 2020.
- [18] M. Lim dan M. R. Ridho, "Rancang Bangun Sistem Informasi Point of Sale Dengan Framework Codeigniter Pada CV Powershop," *Computer And Science*



- Industrial Engineering (COMASIE)*, vol. 4, no. 2, hlm. 46–55, 2021.
- [19] M. Maydianto, “Rancang Bangun Sistem Informasi Point of Sale Dengan Framework Codeigniter Pada Cv Powershop,” 2021.
- [20] F. Adhim dan M. A. Murtadho, “Perancangan Aplikasi Cetak Dokumen Online Berbasis Android Di Biner Jombang,” *Jurnal Informatika dan Rekayasa Elektronik*, vol. 3, no. 1, hlm. 85–90, 2020.
- [21] R. Hafsari, E. Arribe, dan A. Fernando, “Analisis dan perancangan sistem informasi pendaftaran online pt. Medianusa permana net,” *vol*, vol. 7, hlm. 66–78, 2024.
- [22] M. A. A. Syaputra dan A. Voutama, “Rancang Bangun Aplikasi Pemesanan Cuci Motor & Mobil Berbasis Website,” *Jurnal Informatika dan Rekayasa Elektronik*, vol. 7, no. 1, hlm. 88–97, 2024.
- [23] P. Kustianto, S. Auliana, B. R. S. Permana, A. Rohman, dan A. Munawir, “Penerapan Framework Codeigniter 3 pada Aplikasi Penjualan di Toko Kue Kering Miya Jaya Serang,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 1, hlm. 831–836, 2025.
- [24] S. Idayanti, “Analisis kesesuaian P5P2RA dengan prinsip pelaksanaan dan dampaknya terhadap perilaku peserta didik,” *Wawasan: Jurnal Kediklatan Balai Diklat Keagamaan Jakarta*, vol. 4, no. 1, hlm. 48–66, 2023.
- [25] A. R. Kamila, G. H. Derhass, D. A. Rabbani, J. F. Andry, dan F. S. Lee, “Aplikasi Absensi Berbasis Android Pada Sekolah Boarding Sebagai Transformasi Digital Bidang Pendidikan,” *Nuansa Informatika*, vol. 18, no. 2, hlm. 26–34, 2024.
- [26] K. Hasan, A. Husna, M. Muchlis, D. Fitri, dan Z. Zulfadli, “Transformasi komunikasi massa era digital antara peluang dan tantangan,” *JPP Jurnal Politik dan Pemerintahan*, vol. 8, no. 1, hlm. 41–55, 2023.
- [27] A. Wibowo, “Internet of Things (IoT) dalam Ekonomi dan Bisnis Digital,” *Penerbit Yayasan Prima Agus Teknik*, hlm. 1–94, 2023.
- [28] F. Sinlae, E. Irwanda, Z. Maulana, dan V. E. Syahputra, “Penggunaan Framework Laravel dalam Membangun Aplikasi Website Berbasis PHP,” *Jurnal Siber Multi Disiplin*, vol. 2, no. 2, hlm. 119–132, 2024.
- [29] B. Y. Geni, A. Supriyadi, H. Khotimah, dan W. I. Yanti, “Rancang Bangun Company Profile Berbasis Web Menggunakan Metode Waterfall (Studi Kasus: APM Frozen Food),” *Jurnal RESTIKOM: Riset Teknik Informatika dan Komputer*, vol. 6, no. 1, hlm. 75–85, 2024.
- [30] A. Aliyah, N. Hartono, dan A. A. Muin, “Penggunaan User Acceptance Testing (UAT) pada pengujian sistem informasi pengelolaan keuangan dan inventaris barang,” *Switch: Jurnal Sains dan Teknologi Informasi*, vol. 3, no. 1, hlm. 84–100, 2025.
- [31] S. S. Raweyai dan I. R. Widiarsari, “Performance Testing Of Academic Website Using Load Testing Method Supported By Apache JMeterTM At XYZ University,” *Jurnal Teknik Informatika (Jutif)*, vol. 5, no. 3, hlm. 721–730, 2024.