



Perancangan Sistem E-Commerce MotoPart Berbasis Web dengan Metode Waterfall

Masdar Helmi¹, Ahmad Wahyudi², Bagus Rifai³, Luthfi Ardeansyah R⁴

¹⁻⁴ Universitas Duta Bangsa Surakarta, Indonesia

Alamat: Jl. Bhayangkara No.55, Tipes, Kec. Serengan, Kota Surakarta, Jawa Tengah 57154

Korespondensi penulis: Masdarhelmi23@email.com

Abstract. *This research aims to develop MotoPart, an online marketplace application specifically designed to meet the demand for motorcycle spare parts in Indonesia. The background to the development of this application is the high demand for motorcycle spare parts along with the increasing number of two-wheeled vehicles, as well as various problems still frequently encountered in conventional distribution systems. These problems include limited access to genuine products, inefficiencies in the transaction process, and limited information regarding product availability. MotoPart was developed with the main goal of facilitating digital spare part buying and selling transactions that are easier, safer, and more efficient, as well as building an active, trusted, and integrated automotive ecosystem. In this research, a comprehensive system requirements analysis was conducted, covering functional requirements such as user registration and authentication, product management by sellers, a product search and filter system, a shopping cart feature, digital payments, delivery services, a user review system, notifications, transaction management, and an admin panel for monitoring. Meanwhile, non-functional requirements were also outlined to ensure system performance and reliability, including the required hardware and software specifications. The system design was carried out using a Data Flow Diagram (DFD) approach, starting from the context diagram to level 1 to illustrate the data flow and processes that occur. Furthermore, the database design was designed in detail with tables such as users, products, orders, and order items, to support systematic data management. With the development of MotoPart, it is hoped that it can provide solutions to obstacles in spare parts distribution, guarantee product authenticity, and become part of the digital transformation in the Indonesian automotive sector.*

Keywords: *E-commerce, Laravel, Information System, Waterfall, MotoPart.*

Abstrak. Penelitian ini bertujuan untuk mengembangkan MotoPart, sebuah aplikasi lokapasar (marketplace) berbasis daring yang secara khusus dirancang untuk memenuhi kebutuhan akan suku cadang sepeda motor di Indonesia. Latar belakang pengembangan aplikasi ini adalah tingginya permintaan terhadap onderdil motor seiring dengan meningkatnya jumlah kendaraan roda dua, serta berbagai permasalahan yang masih sering ditemui pada sistem distribusi konvensional. Permasalahan tersebut antara lain meliputi keterbatasan akses terhadap produk asli, inefisiensi dalam proses transaksi, serta keterbatasan informasi mengenai ketersediaan barang. MotoPart dikembangkan dengan tujuan utama untuk memfasilitasi transaksi jual beli onderdil secara digital yang lebih mudah, aman, dan efisien, serta membangun ekosistem otomotif yang aktif, terpercaya, dan terintegrasi. Dalam penelitian ini, dilakukan analisis kebutuhan sistem secara menyeluruh yang mencakup kebutuhan fungsional seperti registrasi dan autentikasi pengguna, manajemen produk oleh penjual, sistem pencarian dan filter produk, fitur keranjang belanja, pembayaran digital, layanan pengiriman, sistem ulasan pengguna, pemberitahuan (notification), pengelolaan transaksi, serta panel admin untuk pengawasan. Sementara itu, kebutuhan non-fungsional juga dijabarkan untuk memastikan performa dan keandalan sistem, termasuk spesifikasi perangkat keras dan lunak yang dibutuhkan. Perancangan sistem dilakukan dengan pendekatan Diagram Alir Data (Data Flow Diagram/DFD) mulai dari diagram konteks hingga level 1 untuk menggambarkan alur data dan proses yang terjadi. Selain itu, desain basis data dirancang secara detail dengan tabel-tabel seperti pengguna, produk, pesanan, dan item pesanan, guna mendukung pengelolaan data yang sistematis. Dengan dikembangkannya MotoPart, diharapkan dapat memberikan solusi atas hambatan dalam distribusi suku cadang, menjamin keaslian produk, serta menjadi bagian dari transformasi digital di sektor otomotif Indonesia.

Kata kunci: E-commerce, Laravel, Sistem Informasi, Waterfall, MotoPart.

1. LATAR BELAKANG

Banyak industri berubah karena kemajuan teknologi informasi, termasuk penjualan suku cadang kendaraan bermotor. Digitalisasi operasi bisnis melalui platform e-commerce memungkinkan bisnis untuk meningkatkan kinerja layanan, memperluas jangkauan pasar, dan memudahkan pelanggan untuk mendapatkan barang yang dibutuhkan secara cepat dan tepat. Dalam situasi seperti ini, sistem penjualan berbasis web menawarkan solusi praktis untuk mengatasi keterbatasan distribusi dan akses produk. Menurut penelitian Al-Amin dan Mariana (2022), sistem informasi penjualan suku cadang motor berbasis web dapat meningkatkan kepuasan pelanggan, mempercepat transaksi, dan memperbaiki pencatatan stok .

Pengembangan sistem e-commerce untuk industri otomotif telah dibahas dalam beberapa penelitian. Valentino, Nastiti, dan Yuliana (2022) menggunakan metode Waterfall untuk merancang sistem penjualan dan layanan cepat di bengkel motor berbasis web. Metode ini terbukti efektif dalam mendokumentasikan proses pengembangan secara sistematis . Sebaliknya, Nugroho dan Yuliazmi (2021) menggunakan strategi Business Model Canvas (BMC) dan optimasi SEO untuk menjual sparepart motor dalam e-commerce berbasis WordPress. Namun, mereka belum membahas integrasi sistem pencarian berdasarkan spesifikasi kendaraan. Ini menunjukkan bahwa solusi saat ini belum sepenuhnya menangani kompleksitas distribusi dan struktur katalog suku cadang di industri suku cadang kendaraan roda dua .

Seiring dengan peningkatan permintaan pasar, sistem e-commerce yang lebih komprehensif menjadi lebih penting. Maliki dan Nudin (2023) mengembangkan sistem e-commerce untuk UMKM sepeda motor yang memungkinkan pemesanan jasa dan penjualan sparepart melalui website, tetapi belum mempelajari aspek manajemen inventori teknis. Effenti, Setiawan, dan Setiawan (2022) menemukan bahwa perencanaan persediaan yang lebih akurat dapat dicapai dengan menggunakan metode Activity Based Costing (ABC) berbasis web. Studi serupa oleh Siregar dan Napitupulu (2023) juga menekankan bahwa sistem manajemen stok berbasis web sangat penting untuk mengontrol ketersediaan barang dan membuat keputusan pembelian yang lebih baik.

Sebaliknya, pilihan metode pengembangan perangkat lunak memengaruhi kualitas sistem yang dibuat. Studi oleh Ramadhan dan Fitriani (2021) menemukan bahwa pendekatan Waterfall cocok untuk membangun sistem dengan kebutuhan yang terdokumentasi dan tahapan proses yang terstruktur. Metodologi ini mendukung proses validasi tiap tahapan, mulai dari analisis kebutuhan hingga implementasi. Ini sangat penting untuk sistem penjualan suku

cadang yang memiliki dependensi antar fitur. Selain itu, Putra dan Wahyudi (2022) menunjukkan bahwa penerapan pendekatan terstruktur dalam sistem jual-beli sparepart kendaraan berbasis web dapat mempercepat transaksi dan memastikan data produk tetap aman.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa masih ada ruang untuk penelitian terkait dengan perancangan sistem e-commerce berbasis web yang dimaksudkan untuk menjual suku cadang motor dengan integrasi manajemen stok yang efektif, pencarian berdasarkan spesifikasi kendaraan, dan fitur katalog teknis. Oleh karena itu, penelitian ini bertujuan untuk menggunakan metode Waterfall untuk merancang sistem e-commerce berbasis web bernama MotoPart. Diharapkan sistem ini dapat meningkatkan efisiensi transaksi digital, memudahkan pencarian produk oleh pengguna berdasarkan jenis kendaraan mereka, dan membantu mengelola inventori secara terstruktur dan terdokumentasi.

2. KAJIAN TEORITIS

Sistem Informasi dan E-Business

Sistem informasi diartikan sebagai gabungan terstruktur dari manusia, perangkat keras, perangkat lunak, jaringan komunikasi, dan sumber daya data yang berfungsi untuk mengumpulkan, memproses, dan menyebarkan informasi dalam sebuah organisasi. Tujuan implementasi e-business adalah untuk memastikan bahwa data sumber daya manusia, keuangan, manajemen rantai pasokan, atau manajemen logistik diolah dengan cepat dan akurat (Yanuardi dkk., 2020). Dalam konteks aplikasi MotoPart, sistem informasi berperan sentral dalam mengelola seluruh data, mulai dari pengguna, produk, hingga transaksi. Integrasi Aplikasi Korporasi (IAK) menjadi relevan karena sebuah aplikasi *marketplace* membutuhkan konektivitas yang mulus antara berbagai modul (misalnya manajemen produk, pembayaran, pengiriman) dan potensi integrasi dengan sistem eksternal seperti *payment gateway*. E-bisnis juga memberi pelanggan peluang untuk lebih menyesuaikan dan menyesuaikan pengalaman mereka, sehingga meningkatkan kepuasan dan loyalitas pelanggan (Fatah dkk., 2023). Konsep *E-Business* sangat berkaitan erat dengan pengembangan MotoPart, di mana *E-Business* melibatkan pemanfaatan teknologi informasi dan komunikasi untuk mendukung keseluruhan aktivitas bisnis, tidak hanya jual beli, tetapi juga proses internal dan eksternal. Melalui *E-Business*, MotoPart diharapkan dapat mengoptimalkan efisiensi operasional, memperluas jangkauan pasar, dan meningkatkan layanan pelanggan secara daring.

Perancangan Sistem dengan Data Flow Diagram (DFD)

Perancangan sistem adalah fase esensial dalam siklus pengembangan sistem yang berfokus pada bagaimana sistem akan bekerja untuk memenuhi kebutuhan yang telah ditetapkan. Suatu diagram yang disebut DFD menunjukkan aliran data dari proses sistem informasi (Mirwansyah dkk., 2023). Data Flow Diagram (DFD) menunjukkan bagaimana data bergerak dalam sistem informasi (Saputra dkk., 2024). Dalam perancangan aplikasi MotoPart, DFD digunakan untuk memetakan aliran informasi antara Pembeli, Admin, dan sistem, mulai dari proses registrasi, pengelolaan produk, transaksi pembelian, hingga pelaporan. Penggunaan diagram konteks, DFD Level 0, dan DFD Level 1 memecah kompleksitas sistem menjadi bagian-bagian yang lebih mudah dipahami dan dikelola.

Basis Data Relasional

Basis data adalah bagian penting dari sistem informasi modern karena digunakan untuk menyimpan, mengatur, dan menyediakan informasi dengan cepat. Salah satu jenis basis data yang paling sering digunakan adalah basis data relasional. Basis data ini menyimpan data dalam bentuk tabel-tabel yang saling terhubung melalui kunci utama (primary key) dan kunci asing (foreign key). Dengan cara ini, data bisa diambil dan disusun secara rapi dan teratur. Basis data relasional dibuat dengan prinsip normalisasi, yaitu cara untuk menghindari pengulangan data dan menjaga keakuratan data (Thoib & Candra, 2025). Setiap objek penting dalam sistem, seperti pengguna, produk, pesanan, dan detail pesanan, disimpan dalam tabel tersendiri. Tabel-tabel ini saling terhubung untuk menunjukkan hubungan antara pengguna dan transaksi. Dalam sistem seperti marketplace MotoPart, model ini membantu mengelola data pengguna, daftar produk, dan riwayat transaksi dengan rapi dan efisien.

Sistem manajemen basis data relasional seperti MySQL dan MariaDB banyak dipakai karena dapat menjalankan transaksi dengan aman dan konsisten (sesuai prinsip ACID), bisa menangani pertumbuhan data yang besar, dan mudah dihubungkan dengan aplikasi web (Yuliani & Wahyuni, 2022). Penggunaan indeks dan hubungan antar tabel juga sangat penting untuk mempercepat pencarian data, terutama pada sistem yang memiliki banyak data dan transaksi. Menjaga keterkaitan data antar tabel (integritas referensial) dan memeriksa struktur basis data (validasi skema) sangat penting agar tidak terjadi ketidaksesuaian atau kesalahan data. Dalam proses pengembangan, biasanya dimulai dengan membuat model awal seperti Entity Relationship Diagram (ERD) untuk menggambarkan hubungan antar data (Wathani dkk., 2025). Setelah itu, model ini diterjemahkan ke dalam bentuk yang lebih teknis, yaitu tabel-tabel relasional yang digunakan dalam basis data.

E-commerce dan Marketplace

E-commerce mengacu pada aktivitas jual beli produk atau jasa melalui jaringan elektronik, terutama internet (Fatah dkk., 2023). *Marketplace* merupakan salah satu model *e-commerce* di mana sebuah platform menyediakan ruang bagi banyak penjual untuk menawarkan produk mereka kepada banyak pembeli. Website marketplace khusus suku cadang motor bisa secara nyata membantu bengkel menjadi lebih dikenal, menjangkau lebih banyak pelanggan, dan memudahkan konsumen dalam mencari serta membeli suku cadang secara online (Mahendra dkk., 2024). Sistem informasi persediaan digital (online inventory) memudahkan bengkel untuk mencatat stok barang secara langsung (real-time), merencanakan pemesanan ulang dengan lebih efisien, dan memastikan onderdil selalu tersedia, sehingga pelanggan pun lebih puas (Apriyansah dkk., 2025). Model ini bertujuan untuk meningkatkan kepercayaan dan efisiensi dalam transaksi onderdil motor, serta menciptakan ekosistem digital otomotif yang aktif dan terpercaya.

3. METODE PENELITIAN

Waterfall

Model Waterfall adalah model pengembangan perangkat lunak sekuensial yang sistematis yang dimulai dengan analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Setiap tahap dibutuhkan untuk diselesaikan sebelum dapat melanjutkan. Model ini masih relevan untuk pengembangan aplikasi web berstruktur, seperti sistem e-commerce MotoPart, karena ruang lingkup dan kebutuhannya telah dipahami sejak awal, sehingga mengurangi kemungkinan perubahan selama proses pengembangan. Sebagai contoh, dalam pengembangan aplikasi berbasis web, metode Waterfall dilaporkan dapat meningkatkan struktur dokumentasi, mempermudah manajemen proyek, dan mengurangi risiko fase tumpang tindih (Supriatiningsih et al., 2023). Selain itu, muatan adaptasi Waterfall ke dalam konteks pengembangan hybrid, seperti penggabungan metode Agile, menunjukkan bahwa model ini tetap menjadi dasar yang kuat untuk tahap awal yang jelas.

Analisis Kebutuhan

Tahap awal penting dalam siklus Waterfall adalah analisis kebutuhan, yang bertujuan untuk menentukan kebutuhan fungsional dan non-fungsional sistem. Kebutuhan fungsional seperti pencarian produk, manajemen stok, keranjang belanja, dan integrasi pembayaran

ditemukan saat mengembangkan MotoPart E-Commerce. Sementara itu, untuk memastikan bahwa apa yang diharapkan pengguna terpenuhi, persyaratan non-fungsional seperti keamanan, kinerja, dan kemudahan penggunaan juga dipertimbangkan. Studi tentang sistem CRM oleh Putrianto, Widyastuti, dan Oktiarso (2024) menemukan bahwa perencanaan dan analisis kebutuhan pengguna yang menyeluruh sangat memengaruhi kesuksesan sistem; metode SDLC yang terstruktur mengurangi kesalahan pada tahap implementasi.

Perancangan Sistem

Tujuan dari perancangan sistem adalah untuk mengubah hasil analisis kebutuhan menjadi model teknis yang jelas dan terstruktur. Pendekatan perancangan utama untuk E Commerce MotoPart adalah menggunakan Diagram Aliran Data (DFD) untuk memetakan aliran data antar entitas (pelanggan, manajer, sistem), serta antarmuka eksternal seperti modul pembayaran dan notifikasi. Desain dimulai dengan Level 0 DFD konteks, yang menunjukkan garis besar interaksi sistem dengan aktor eksternal. Level 1 DFD menggambarkan proses utama seperti pencarian produk, pemrosesan pesanan, manajemen stok, dan lap. Diagram ini membantu pengembang memahami alur data dan titik pemrosesan penting. Ini juga membantu pemangku kepentingan menyetujui struktur alur sistem sebelum implementasi dimulai. Sistem juga dirancang menggunakan Entity Relationship Diagram (ERD) untuk menunjukkan struktur basis data dan hubungan antar entitas seperti Produk, Pelanggan, Transaksi, dan Pembayaran. ERD ini berfungsi sebagai dasar untuk membangun skema database relasional yang konsisten dan efektif. Selain itu, perancangan antarmuka pengguna diselaraskan dengan alur DFD, sehingga tampilan dan interaksi pengguna antarmuka pengguna benar-benar mencerminkan proses bisnis yang sudah dimodelkan. Metode ini dirancang untuk memastikan bahwa implementasi berjalan sesuai dengan rancangan tanpa banyak perubahan. Sebuah penelitian oleh Fanisa Dia Ayu Kharisma, Rina Firliana, dan Arie Nugroho (2024) menunjukkan bahwa model DFD sangat efektif dalam memodelkan alur data dan membantu menjelaskan kebutuhan sistem sebelum coding. Studi ini menggunakan diagram DFD untuk memperjelas arus data dari permintaan stok hingga update database, dan terbukti mampu mencegah banyak kesalahan saat pengembangan. Dengan demikian, penerapan diagram DFD pada sistem E Commerce MotoPart diharapkan dapat meningkatkan kejelasan desain dan kualitas sistem secara keseluruhan.

Pengembangan

Selama tahap pengembangan, rancangan sistem ditransformasi menjadi aplikasi praktis. MotoPart adalah sistem e-commerce yang dikembangkan menggunakan bahasa

pemrograman PHP dengan framework Laravel dan basis data MySQL. Pemilihan Laravel berbasis arsitektur Model-View-Controller (MVC) membuat pengelolaan kode yang terstruktur dan modular lebih mudah. Fitur ini sangat bermanfaat untuk membangun sistem e-commerce yang kompleks (Defni & Lestari, 2020). Setiap modul dibangun berdasarkan desain sebelumnya (DFD, ERD). Proses ini mencakup pembuatan modul manajemen produk, transaksi, pengguna, dan laporan. Modul-modul ini diuji secara efektif selama proses pengkodean untuk memastikan bahwa mereka sesuai dengan kebutuhan. Menurut Pangastuti, Nugroho, dan Prami Swari (2021), implementasi Laravel pada aplikasi perpustakaan berbasis web telah terbukti dapat meningkatkan efisiensi pengembangan dan memudahkan perawatan karena struktur kodenya yang mudah dipahami. Dalam studi mereka tentang e-commerce berbasis Laravel, Defni dan Lestari (2020) menemukan bahwa rangka kerja ini memudahkan pembuatan modul transaksi dan memungkinkan integrasi pembayaran dengan MySQL dengan cepat dan aman. Ini adalah fitur penting untuk sistem MotoPart yang berfokus pada kelancaran pembayaran pelanggan.

Pengujian

Tahap pengujian adalah langkah penting dalam model Waterfall. Tujuannya adalah untuk memastikan bahwa sistem e-commerce MotoPart beroperasi sesuai dengan spesifikasi dan tidak mengalami kesalahan. Pada titik ini, teknik Black Box Testing digunakan, yang berfokus pada menguji kinerja elemen luar sistem tanpa melihat kode internal. Untuk memperluas cakupan pengujian, teknik analisis nilai batas juga digunakan untuk mengevaluasi sistem pada nilai-nilai batas input. Perencanaan kasus pengujian memulai proses pengujian, yang mencakup skenario input yang sah dan salah di setiap fitur utama, seperti login, registrasi, pencarian produk, proses checkout, dan manajemen stok. Pengujian dilakukan dengan membandingkan hasil yang diharapkan dari sistem dengan hasil yang diharapkan. Jalani test case tersebut secara menyeluruh untuk menemukan bug atau abnormalitas fungsi. Selanjutnya, hasil pengujian dicatat dalam tabel uji untuk analisis. Dalam penelitian oleh Jailani dan Yaqin (2024) pada aplikasi sistem informasi akademik, metode Black Box Testing dengan teknik Boundary Value Analysis menunjukkan keberhasilan deteksi bug hingga 60% dari seluruh percobaan, menunjukkan bahwa ada perlunya penyesuaian untuk membuat aplikasi lebih andal. Ini menunjukkan bahwa teknik ini efektif dalam mencari kesalahan fungsional pada aplikasi berbasis web. Selain itu, pendekatan serupa diterapkan pada aplikasi e-Promkes. Kombinasi teknik Partitioning Equivalent dan Analysis of Boundary Value telah terbukti memberikan cakupan pengujian yang luas dalam berbagai skenario input. Dengan menerapkan

metode ini pada sistem E-Commerce MotoPart, diharapkan menghasilkan pengujian yang sistematis dan efektif, terutama untuk memvalidasi komponen penting dan mencegah kesalahan sebelum sistem siap dirilis.

4. HASIL DAN PEMBAHASAN

Analisis Kebutuhan

Analisi Kebutuhan Fungsional

Dalam pengembangan sistem aplikasi *MotoPart*, telah diidentifikasi sejumlah kebutuhan fungsional yang menjadi dasar perancangan sistem. Aplikasi ini menyediakan fitur autentikasi pengguna, di mana pengguna dapat melakukan pendaftaran dan masuk ke dalam sistem sesuai perannya, baik sebagai pembeli, admin, maupun owner. Admin dan owner diberikan kemampuan untuk mengelola produk yang mereka tawarkan, termasuk menambahkan informasi produk baru, memperbarui data produk, dan menghapus produk yang sudah tidak tersedia. Di sisi pembeli, sistem menyediakan kategori permerk dan penyaringan produk yang memudahkan pengguna dalam menemukan suku cadang sesuai dengan kategori mesin, body, maupun aksesoris. Untuk proses transaksi, pengguna dapat menambahkan produk ke dalam keranjang belanja dan menyelesaikan pembelian melalui proses checkout yang terintegrasi dengan sistem pembayaran online midtrans sehingga transaksi dapat dilakukan secara cepat dan aman.

Setelah pembelian, Admin dapat mencatat informasi pengiriman seperti nomor resi, dan pembeli dapat memantau perkembangan pengiriman secara langsung melalui sistem. Untuk meningkatkan transparansi dan kepercayaan, pengguna juga dapat memberikan ulasan dan penilaian terhadap produk yang dibeli. Sistem secara otomatis mengirimkan notifikasi kepada pengguna terkait status pesanan, pembayaran, dan informasi penting lainnya. Selain itu, setiap aktivitas transaksi dicatat dan disimpan sebagai riwayat yang dapat diakses kembali oleh pengguna sesuai kebutuhan. Pihak admin memiliki akses penuh terhadap sistem melalui panel khusus yang digunakan untuk mengelola akun pengguna, memverifikasi produk yang ditawarkan, menangani pelaporan dari pengguna, serta menghasilkan laporan transaksi dan statistik penjualan. Seluruh fungsi ini dirancang untuk menciptakan platform digital suku cadang motor yang efisien, terorganisir, dan dapat dipercaya oleh seluruh pemangku kepentingan.

Analisis Kebutuhan Non Fungsional

Kebutuhan Perangkat Lunak

Dalam pengembangan aplikasi *MotoPart*, kebutuhan non-fungsional pada aspek perangkat lunak menjadi faktor penting guna memastikan sistem dapat dikembangkan, diuji, dan dioperasikan secara optimal. Sistem operasi yang digunakan oleh pengembang disarankan minimal Windows 10 versi terbaru agar kompatibel dengan perangkat lunak modern. Untuk lingkungan pengembangan lokal, aplikasi dapat dijalankan menggunakan alat bantu seperti Laragon atau XAMPP yang mendukung server lokal berbasis PHP. Pada sisi backend, aplikasi dibangun menggunakan framework Laravel versi terbaru dengan dukungan PHP 8.2 ke atas, Sementara pada bagian frontend, teknologi yang digunakan mencakup HTML, CSS, dan JavaScript.

Untuk manajemen basis data, sistem menggunakan MySQL atau MariaDB yang mendukung integrasi kuat dengan Laravel serta mampu menangani transaksi skala besar secara efisien. Selain itu, aplikasi harus mampu berjalan secara optimal pada berbagai browser modern seperti Google Chrome, Mozilla Firefox, dan Safari versi terbaru guna memastikan kenyamanan akses bagi seluruh pengguna. Kebutuhan perangkat lunak ini disusun dengan mempertimbangkan efisiensi kerja pengembang, kompatibilitas antar sistem, serta kemudahan dalam pemeliharaan dan pengembangan berkelanjutan.

Kebutuhan Perangkat Keras

Dalam pembangunan dan pengoperasian sistem aplikasi *MotoPart*, kebutuhan perangkat keras menjadi aspek penting untuk memastikan proses pengembangan dan penggunaan berjalan dengan lancar. Dari sisi pengembang (developer), diperlukan perangkat komputer dengan spesifikasi minimal prosesor setara Intel Core i5, RAM sebesar 8 GB, media penyimpanan tipe SSD berkapasitas 256 GB atau lebih, serta layar dengan resolusi Full HD guna mendukung aktivitas pengkodean, pengujian, dan pemantauan sistem secara efektif. Spesifikasi ini dirancang untuk mendukung software pengembangan modern seperti server lokal, editor kode, dan browser uji coba.

Analisis Kebutuhan Operasional

Dalam pengoperasian aplikasi *MotoPart*, kebutuhan non-fungsional dari sisi operasional mencakup kesiapan sumber daya manusia dan infrastruktur pendukung yang sesuai dengan struktur peran pengguna dalam sistem. Sistem ini dirancang dengan dua jenis peran utama, yaitu admin (juga merangkap sebagai owner) dan pembeli. Admin/owner memiliki akses penuh terhadap sistem, termasuk pengelolaan data produk, pengguna, transaksi, pelaporan penjualan, serta validasi proses pembayaran. Oleh karena itu, individu yang

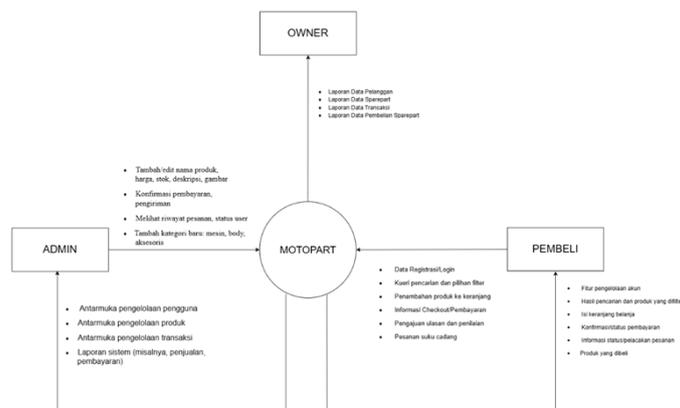
menjalankan peran ini harus memiliki pemahaman teknis yang baik tentang manajemen e-commerce dan kemampuan dalam mengelola data secara sistematis.

Sementara itu, pengguna dengan peran pembeli hanya memiliki akses terbatas pada fitur pemesanan produk. Mereka dapat melakukan pencarian, memilih produk, menambahkannya ke keranjang, melakukan pembayaran, serta memantau status pesanan. Untuk mendukung operasional ini, diperlukan ketersediaan infrastruktur seperti jaringan internet yang stabil, serta pelatihan dasar bagi admin untuk mengoperasikan dashboard sistem. Sistem juga dirancang agar antarmukanya ramah pengguna sehingga pembeli dapat menggunakan aplikasi tanpa memerlukan keahlian teknis. Dengan struktur peran yang sederhana namun terfokus, kebutuhan operasional aplikasi MotoPart dapat dikelola dengan efisien dan tetap mampu memberikan layanan yang optimal.

Perancangan Sistem

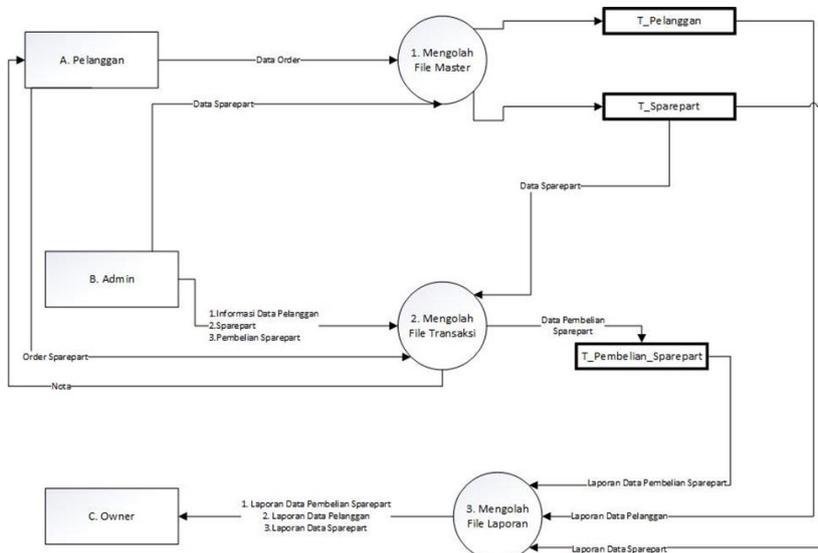
Untuk memahami bagaimana informasi diproses dalam sistem, diperlukan gambaran yang jelas mengenai alur data. Salah satu cara untuk merepresentasikan hal tersebut adalah melalui Data Flow Diagram (DFD). DFD merupakan representasi visual yang menggambarkan bagaimana data masuk ke dalam sistem, bagaimana data diproses, disimpan, dan kemudian menghasilkan keluaran. Dalam konteks aplikasi *MotoPart*, DFD digunakan untuk menjelaskan hubungan antara pengguna dengan sistem, serta menggambarkan proses-proses utama seperti registrasi, pengelolaan produk, pemesanan barang, pembayaran, hingga pelacakan pengiriman. Berikut ini adalah gambaran umum alur data dalam sistem *MotoPart* melalui DFD.

Diagram Konteks



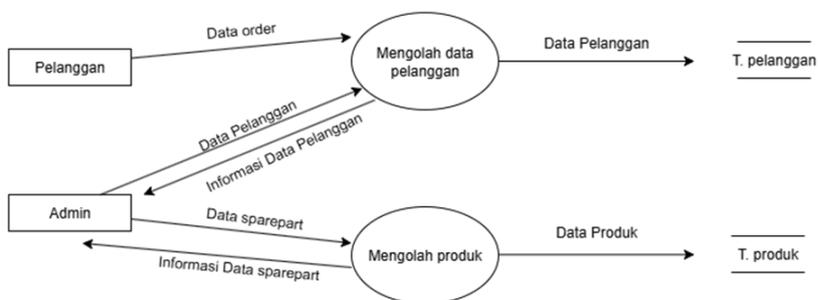
Gambar 1. Diagram Konteks

DFD Level 0



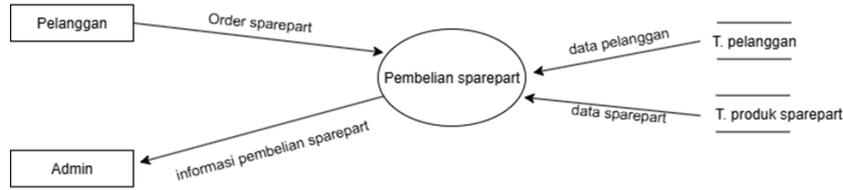
Gambar 2. DFD Level 0

Diagram Level 1 (Mengelola file master)



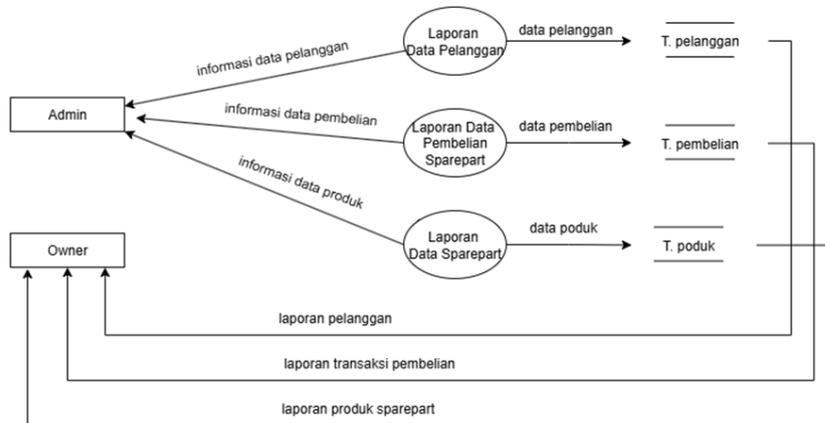
Gambar 2. Diagram Level 1 (Mengelola file master)

Diagram Level 1 (Mengelola file transaksi)



Gambar 3. Diagram Level 1 (Mengelola file transaksi)

Diagram Level (Mengelola file laporan)



Gambar 4. Diagram Level (Mengelola file laporan)

Implementasi Sistem

Arsitektur Sistem

Tahap implementasi merupakan bagian akhir dari proses pengembangan, di mana hasil analisis dan perancangan sistem diwujudkan dalam bentuk aplikasi yang dapat digunakan. Aplikasi *MotoPart* dikembangkan sebagai sistem berbasis web menggunakan framework Laravel, dengan struktur client-server yang menghubungkan pengguna (admin dan pembeli) melalui antarmuka berbasis browser.

Dalam sistem ini, admin/owner memiliki akses penuh untuk mengelola produk, pengguna, transaksi, dan laporan. Sementara itu, pembeli hanya dapat melakukan pencarian, pemesanan produk, serta pelacakan pesanan. Seluruh data disimpan dalam satu basis data MySQL yang terintegrasi langsung dengan backend Laravel. Untuk tampilan antarmuka (frontend), digunakan HTML5, CSS3, dan JavaScript agar aplikasi responsif di berbagai perangkat.

Sistem ini juga dilengkapi integrasi payment gateway guna mendukung transaksi online secara aman. Arsitektur ini dirancang agar efisien, mudah dikembangkan, serta mampu mendukung aktivitas jual beli suku cadang motor secara digital.

Konfigurasi Database

Dalam proses pengembangan aplikasi *MotoPart*, pengaturan koneksi ke database dilakukan melalui file konfigurasi `.env` yang disediakan oleh Laravel sebagai bagian dari manajemen lingkungan aplikasi. Sistem ini dibangun dengan menggunakan framework Laravel. Basis data tersebut berfungsi sebagai pusat penyimpanan seluruh informasi penting dalam sistem, seperti data akun pengguna, daftar produk suku cadang, detail transaksi, serta catatan riwayat pembelian. Dengan penggunaan satu database terintegrasi, sistem dapat memastikan konsistensi data serta memudahkan pengelolaan informasi secara menyeluruh dalam proses bisnis yang berjalan.

Konfigurasi database

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=motopart_6
DB_USERNAME=root
DB_PASSWORD=
```

Gambar 5. Konfigurasi database

Konfigurasi tersebut menunjukkan bahwa aplikasi *MotoPart* menggunakan database MySQL yang terhubung secara lokal melalui alamat host 127.0.0.1 dan port default 3306 yang disesuaikan dengan database kita. Dengan konfigurasi ini, sistem dapat menjalankan proses backend seperti login pengguna, penyimpanan data produk, serta manajemen transaksi secara efisien.

Konfigurasi pembayaran midtrans

```
MIDTRANS_SERVER_KEY=SB-Mid-server-
MIDTRANS_CLIENT_KEY=SB-Mid-client-
MIDTRANS_IS_PRODUCTION=false
```

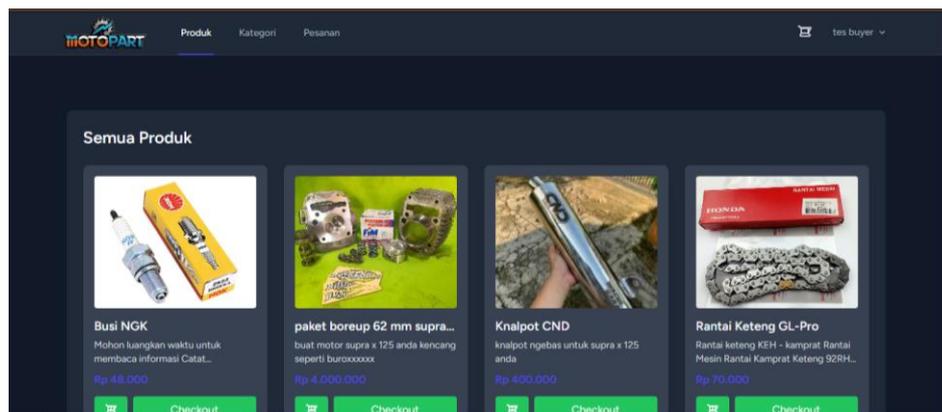
Gambar 6. Konfigurasi pembayaran midtrans

Konfigurasi ini berfungsi untuk menghubungkan aplikasi dengan layanan Midtrans sebagai penyedia payment gateway. Kunci server (`SERVER_KEY`) dan kunci klien (`CLIENT_KEY`) digunakan untuk autentikasi transaksi, sedangkan parameter

IS_PRODUCTION=false menandakan bahwa sistem masih berjalan dalam mode *sandbox* atau pengujian. Penggunaan Midtrans memungkinkan proses pembayaran berjalan secara otomatis, aman, dan terintegrasi langsung dengan sistem aplikasi.

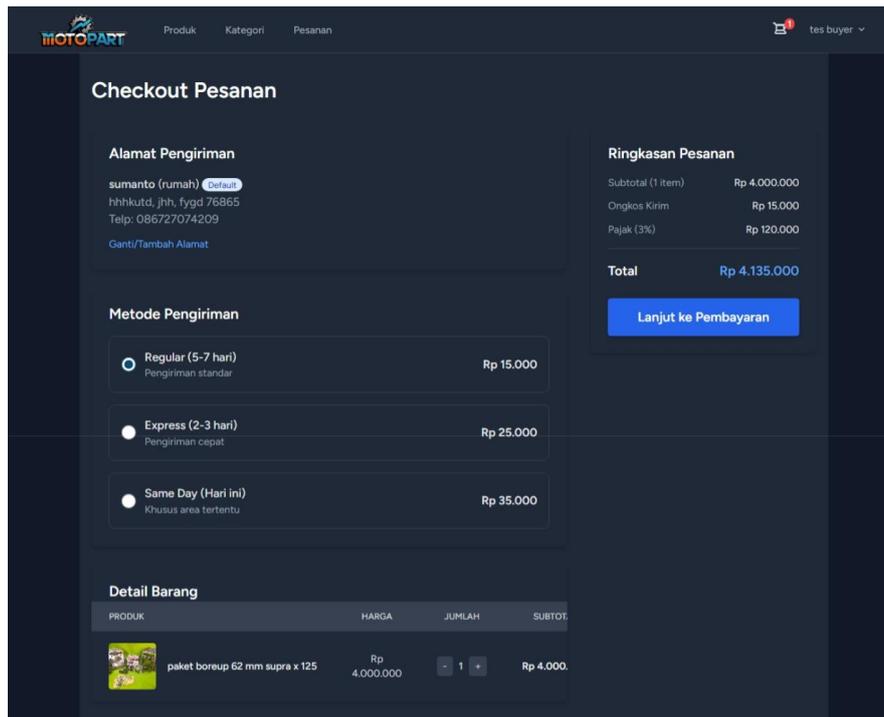
Pengembangan Sistem

Tampilan interface pengguna



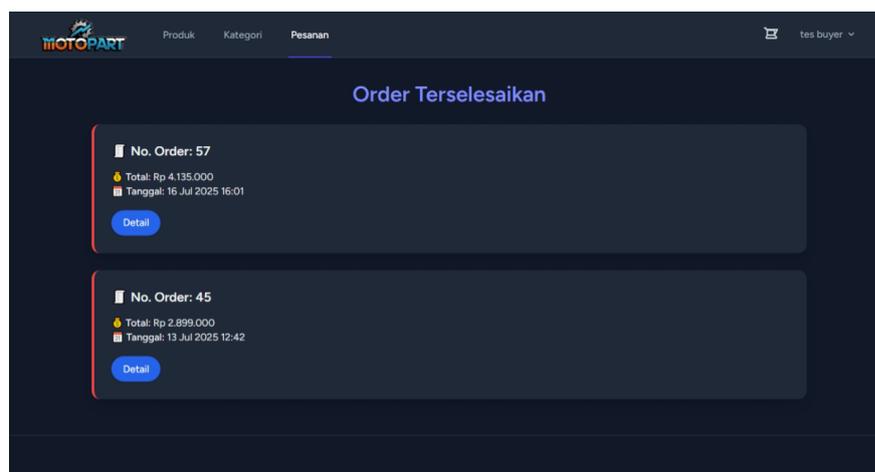
Gambar 7. Tampilan interface pengguna

Antarmuka pengguna didesain agar pembeli dapat berbelanja secara mudah dan efisien. Halaman beranda menyajikan beragam produk yang dikelompokkan berdasarkan kategori, serta menawarkan rekomendasi yang disesuaikan dengan preferensi pengguna. Informasi detail seperti harga dan deskripsi singkat tersedia untuk setiap produk, mempermudah proses penambahan ke keranjang belanja.



Gambar 8. Halaman penyelesaian pesanan

Halaman penyelesaian pesanan bertujuan untuk memfasilitasi pembeli dalam menuntaskan transaksi secara transparan dan terjamin. Pada halaman ini, ringkasan pesanan ditampilkan, meliputi rincian barang, total sementara, biaya kirim, dan total akhir yang perlu dibayarkan. Selain itu, pembeli diwajibkan untuk memverifikasi detail alamat pengiriman serta menentukan opsi pengiriman, sebelum beranjak ke tahapan pembayaran



Gambar 9. Halaman 'Pesanan Selesai'

Halaman 'Pesanan Selesai', sebagai bagian tak terpisahkan dari alur pasca-pembelian, menandai berakhirnya proses transaksi online. Halaman ini secara langsung mengonfirmasi keberhasilan transaksi kepada pembeli, sekaligus menyajikan detail penting seperti ID dan

tanggal pesanan yang dapat menjadi acuan di kemudian hari. Desain ini dirancang untuk menciptakan pengalaman penutup yang memuaskan pada setiap transaksi

Tampilan interface admin

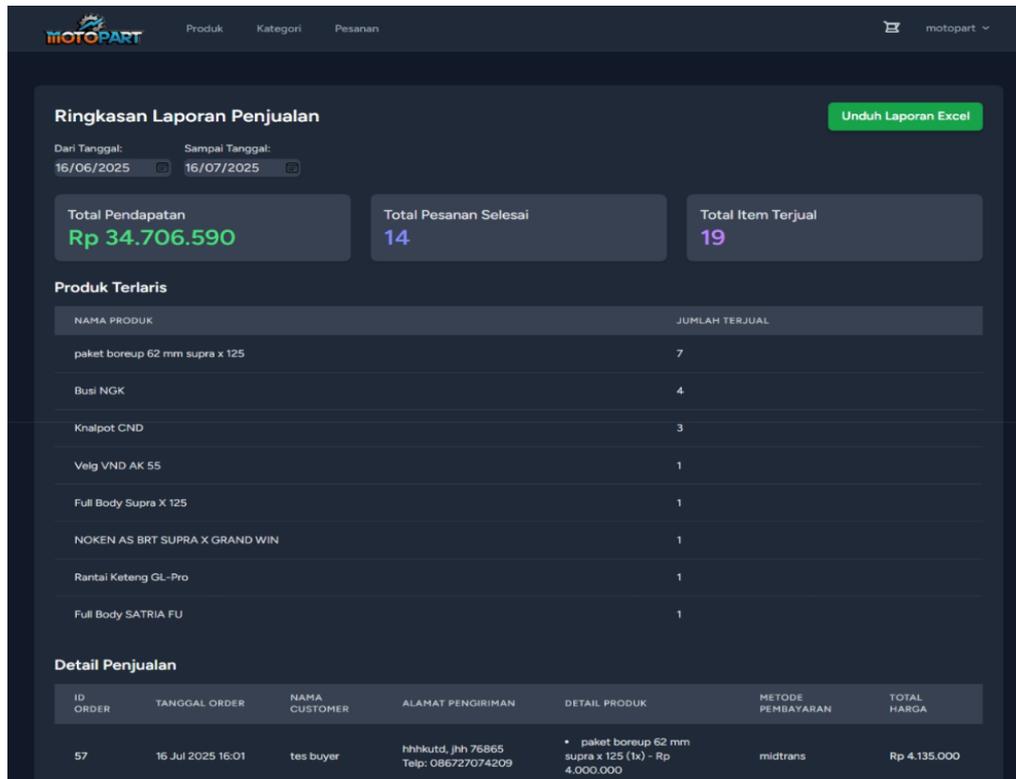
Gambar 10. Tampilan interface admin

Tampilan antarmuka admin untuk create product (Buat Produk) dirancang untuk mempermudah administrator dalam mengelola katalog produk. Halaman ini menyediakan formulir lengkap untuk memasukkan detail produk, seperti Nama Produk, Deskripsi Lengkap, Harga, hingga mengunggah gambar produk. Desainnya yang terstruktur memastikan bahwa semua informasi penting dapat diinput dengan efisien.

GAMBAR	NAMA PRODUK	KATEGORI	BRAND	HARGA	AKSI
	Busi NGK	Mesin	universal	Rp 48.000	Edit Hapus
	paket boreup 62 mm supra x 125	Mesin	honda	Rp 4.000.000	Edit Hapus
	Knalpot CND	Mesin	honda	Rp 400.000	Edit Hapus
	Rantai Keteng GL-Pro	Mesin	honda	Rp 70.000	Edit Hapus
	Full Body Supra X 125	Body	honda	Rp 800.000	Edit Hapus
	Velg VND AK 55	Body	honda	Rp 2.000.000	Edit Hapus

Gambar 11. Antarmuka Admin Daftar Produk

Antarmuka admin daftar produk bertujuan untuk menampilkan ikhtisar menyeluruh dari seluruh produk yang tersedia. Tampilan ini menyuguhkan tabel berisi kolom-kolom krusial seperti nama produk, kategori, stok, dan harga, dilengkapi dengan pilihan edit atau hapus untuk setiap produk. Fungsi ini sangat penting bagi administrator untuk memantau inventaris dan memperbarui data secara efisien.



Gambar 12. Antarmuka admin

Antarmuka admin menampilkan ringkasan laporan penjualan yang menyajikan data penjualan utama selama periode yang dipilih. Halaman ini menampilkan data krusial seperti total pendapatan, jumlah pesanan selesai, dan total item terjual. Terdapat juga daftar produk terlaris yang mengidentifikasi produk dengan penjualan terbanyak, serta detail penjualan yang merinci setiap transaksi. Kemampuan untuk mengunduh laporan turut disediakan untuk analisis data lebih lanjut.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil perancangan dan implementasi sistem, dapat disimpulkan bahwa aplikasi *MotoPart* mampu menjawab kebutuhan digitalisasi proses jual beli suku cadang motor dengan menyediakan fitur yang terintegrasi antara pengguna, produk, transaksi, serta pembayaran. Penggunaan framework Laravel dan basis data MySQL memberikan struktur

sistem yang stabil, sementara integrasi dengan Midtrans mendukung proses pembayaran secara aman dan efisien. Aplikasi ini berhasil memenuhi tujuan awal pengembangan, yaitu membangun platform marketplace suku cadang motor yang dapat diakses oleh admin dan pembeli dengan peran masing-masing. Meskipun demikian, sistem masih memiliki keterbatasan, seperti belum tersedianya sistem pelacakan otomatis dari pihak logistik dan minimnya fitur keamanan lanjutan pada sisi pengguna. Oleh karena itu, pengembangan selanjutnya disarankan untuk menambahkan lapisan keamanan seperti autentikasi dua faktor, enkripsi data sensitif, serta mendukung versi aplikasi mobile untuk memperluas jangkauan pengguna. Penelitian mendatang juga dapat mengeksplorasi integrasi sistem dengan penyedia logistik dan pengembangan kecerdasan buatan untuk rekomendasi produk, guna meningkatkan pengalaman pengguna secara menyeluruh.

DAFTAR REFERENSI

- Al-Amin, M., & Mariana, R. (2022). Pengaruh Sistem Informasi Penjualan Suku Cadang Motor Berbasis Web terhadap Kepuasan Pelanggan. *Jurnal Teknologi Informasi*, 10(2), 123-134.
- Apriyansah, F., Ramadhan, I., Fawaz, N., & Saefudin. (2025). Meningkatkan manajemen persediaan suku cadang (sparepart) motor pada Bengkel Tiga Putra di Bojonegara Banten. *Jurnal Manuhara*, 3(1), 355-361. <https://doi.org/10.61132/manuhara.v3i1.1569>
- Fatah, D. A., Rochman, E. M. S., & Soesilo, B. (2023). E-business dan e-commerce: Konsep, infrastruktur, dan penerapan. *Perkumpulan Rumah Cemerlang Indonesia*.
- Harahap, I., & Wijaya, R. (2021). Analisis Pengaruh Digitalisasi terhadap Kinerja Bisnis Otomotif di Era E-Commerce. *Jurnal Ekonomi Digital*, 14(2), 200-213.
- Hendra, B., & Setiawan, D. (2023). Transformasi Digital dalam Bisnis Otomotif dan Peran E-Commerce. *Jurnal Teknologi Bisnis*, 19(3), 56-67.
- Mahendra, V. Y., Nastiti, F. E., & Yuliana, M. E. (2024). Pemodelan sistem penjualan suku cadang motor dan layanan cepat berbasis website di Bengkel Meca dengan metode waterfall. *JEKIN - Jurnal Teknik Informatika*, 4(3), 709-717. <https://doi.org/10.58794/jekin.v4i3.930>
- Mirwansyah, D., Zahro, K. A., & Irfan, M. (2023). Perancangan sistem informasi monitoring akademik dengan menggunakan data flow diagram. *Jurnal Locus: Penelitian dan Pengabdian*, 2(12), 1201-1207. <https://doi.org/10.58344/locus.v2i12.2352>
- Nugroho, A., & Yuliazmi, R. (2021). Optimasi SEO dan Business Model Canvas untuk E-Commerce Suku Cadang Motor. *Jurnal E-Commerce Indonesia*, 7(1), 75-88.
- Saputra, D. G. E., Agustino, D. P., & Suryasa, I. W. (2024). Rancang bangun web e-commerce pada Dewa Bike berbasis framework Laravel. *SPINTER*, 1(3), 557-562.
- Sari, L., & Pratama, F. (2020). Implementasi Sistem Penjualan Online pada Bisnis Sparepart Motor di Indonesia. *Jurnal Pengembangan Teknologi*, 8(1), 112-125.

- Thoib, I., & Candra, B. P. (2025). Perancangan basis data relasional sistem informasi penerimaan mahasiswa baru di Institut Teknologi Mojosari. *Tepi*, 1(1), 28-36. <https://doi.org/10.14710/tepi.v39n1>.
- Valentino, N., Nastiti, S., & Yuliana, M. (2022). Desain Sistem Penjualan dan Layanan Cepat Bengkel Motor Berbasis Web dengan Metode Waterfall. *Jurnal Sistem Informasi dan Komputer*, 15(4), 89-102.
- Wathani, M. R., Wijaya, E. S., Zaenuddin, Z., & Abidin, M. Z. (2025). Efektivitas indeks dalam meningkatkan performa query join di sistem basis data relasional. *Technologia: Jurnal Ilmiah*, 16(2), 361. <https://doi.org/10.31602/tji.v16i2.18624>
- Yanuardi, Y., Destriana, R., & Faridi, F. (2020). Perancangan sistem informasi penjualan online gas dalam strategi e-business menggunakan analisis SWOT. *JIKA (Jurnal Informatika)*, 4(1), 1. <https://doi.org/10.31000/jika.v4i1.2451>
- Yuliani, D., & Wahyuni, S. (2022). Analisis implementasi konsep basis data relasional pada sistem informasi penjualan di PT Mayora. *Jurnal Ilmiah Teknologi Informasi*, 6(2), 120-130.