



Implementasi Quick Sort untuk Optimasi Pengurutan Data pada Sistem Pencatatan Transaksi Berbasis Web

Yoga Koswara, Dede Rizal Nursamsi

Sistem Informasi, Sains Teknologi, Universitas Cipasung Tasikmalaya

yogakoswara235@gmail.com, dederizalnursamsi@uncip.ac.id

Abstrak

Perkembangan teknologi informasi mendorong perusahaan untuk mengimplementasikan sistem informasi yang mampu meningkatkan efektivitas dan efisiensi dalam pengelolaan data transaksi. Perusahaan Jelita Farsel masih menghadapi kendala dalam proses pencatatan transaksi, khususnya pada pengurutan data yang dilakukan secara manual sehingga membutuhkan waktu lebih lama dan berpotensi menimbulkan kesalahan. Penelitian ini bertujuan untuk mengimplementasikan algoritma Quick Sort pada sistem pencatatan transaksi berbasis web guna meningkatkan kecepatan dan efisiensi proses pengurutan data transaksi. Metode penelitian yang digunakan adalah Research and Development (R&D) dengan model pengembangan Waterfall yang meliputi tahapan analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Sistem dikembangkan menggunakan bahasa pemrograman PHP dengan basis data MySQL, sedangkan pengujian dilakukan menggunakan metode Black Box Testing untuk memastikan seluruh fungsi berjalan sesuai kebutuhan pengguna. Hasil penelitian menunjukkan bahwa implementasi algoritma Quick Sort mampu mempercepat proses pengurutan data transaksi berdasarkan tanggal, nama pelanggan, dan nominal transaksi dengan tingkat akurasi yang baik. Sistem yang dikembangkan juga mendukung pencatatan transaksi yang lebih terstruktur, memudahkan pencarian data, meningkatkan efektivitas monitoring pembayaran, serta mendukung penyusunan informasi transaksi untuk kebutuhan operasional perusahaan. Hasil evaluasi fungsional menunjukkan bahwa integrasi sistem berbasis web dengan algoritma Quick Sort memberikan dukungan yang lebih konsisten terhadap proses pengelolaan transaksi. Dengan demikian, sistem ini dapat menjadi solusi yang efektif dalam mendukung pengelolaan data transaksi secara cepat, akurat, dan efisien.

Kata Kunci: Algoritma Quick Sort, Sistem Pencatatan Transaksi, Sistem Berbasis Web, Pengurutan Data, Waterfall.

Abstract

The development of information technology encourages companies to implement information systems that can improve the effectiveness and efficiency of transaction data management. Jelita Farsel Company still faces several obstacles in transaction recording, particularly in the process of sorting data manually, which requires more time and may increase the risk of errors. This study aims to implement the Quick Sort algorithm in a web-based transaction recording system to improve the speed and efficiency of transaction data sorting. The research method used was Research and Development (R&D) with the Waterfall development model, consisting of requirements analysis, system design, implementation, testing, and maintenance stages. The system was developed using PHP programming language and MySQL database, while system testing was carried out using the Black Box Testing method to ensure that all functions operated according to user requirements. The results show that the implementation of the Quick Sort algorithm can accelerate the process of sorting transaction data based on date, customer name, and transaction amount with good accuracy. The developed system also supports more structured transaction recording, facilitates data searching, improves payment monitoring, and increases the efficiency of transaction information management. Functional evaluation further indicates that the integration of the web-based system with the Quick Sort algorithm provides more consistent support for recording, searching, monitoring, and preparing transaction information for operational needs. Therefore, the system can serve as an effective solution for managing transaction data quickly, accurately, and efficiently.

Keywords: Quick Sort Algorithm, Transaction Recording System, Web-Based System, Data Sorting, Waterfall

1. Pendahuluan

Perkembangan teknologi informasi telah mendorong berbagai perusahaan untuk mengadopsi sistem informasi berbasis web sebagai upaya meningkatkan efektivitas dan efisiensi dalam pengelolaan data. Salah satu aspek penting dalam sistem informasi adalah kemampuan mengolah dan menyajikan data secara cepat, khususnya pada proses *sorting* data transaksi yang terus bertambah setiap harinya. Sistem pencatatan transaksi yang masih dilakukan secara manual atau belum menerapkan algoritma pengurutan yang efisien berpotensi menyebabkan

keterlambatan dalam proses pencarian data, menurunkan produktivitas kerja, serta meningkatkan kemungkinan terjadinya kesalahan dalam pengolahan informasi. Oleh karena itu, penerapan algoritma pengurutan yang tepat menjadi salah satu solusi untuk meningkatkan kinerja sistem informasi transaksi (Nasution et al., 2024).

Salah satu algoritma pengurutan yang banyak digunakan adalah *Quick Sort*. Algoritma ini memiliki kompleksitas waktu rata-rata sebesar $O(n \log n)$ sehingga mampu mengurutkan data secara lebih cepat dibandingkan beberapa algoritma pengurutan lainnya. Selain memiliki efisiensi waktu yang tinggi, *Quick Sort* juga menggunakan memori yang relatif lebih kecil sehingga sesuai diterapkan pada sistem informasi yang mengelola data dalam jumlah besar. Penelitian yang dilakukan oleh Pujiono et al. (2024) menunjukkan bahwa *Quick Sort* memiliki performa yang baik dari sisi waktu komputasi maupun efisiensi memori. Hasil tersebut diperkuat oleh Dorman (2025) yang menyatakan bahwa *Quick Sort* merupakan salah satu algoritma yang efektif untuk meningkatkan efisiensi pemrosesan data pada sistem informasi.

Penerapan algoritma *Quick Sort* telah banyak dilakukan pada berbagai sistem informasi. Rosdiana (2024) mengimplementasikan algoritma *Quick Sort* pada sistem pemesanan makanan dan menunjukkan bahwa algoritma tersebut mampu mempercepat proses pengurutan data pesanan. Penelitian lain yang dilakukan oleh Wijoyo et al. (2024) juga membuktikan bahwa implementasi *Quick Sort* pada sistem akademik dapat meningkatkan efisiensi pengolahan data. Selain itu, Yusuf dan Naimah (2025) mengembangkan aplikasi berbasis Python dengan menerapkan algoritma *Quick Sort* sehingga proses pengurutan data menjadi lebih cepat dan optimal. Meskipun demikian, penelitian-penelitian tersebut masih berfokus pada implementasi algoritma pada sistem yang telah tersedia, sedangkan penerapan *Quick Sort* pada sistem pencatatan transaksi perusahaan berbasis *web* masih relatif terbatas.

Perusahaan Jelita Farsel merupakan perusahaan yang melakukan pencatatan transaksi penjualan setiap hari sehingga membutuhkan sistem informasi yang mampu mengelola data transaksi secara cepat dan akurat. Berdasarkan hasil observasi, proses pencatatan transaksi masih mengalami kendala pada proses pengurutan data sehingga pencarian informasi membutuhkan waktu yang relatif lama ketika jumlah transaksi semakin meningkat. Kondisi tersebut menunjukkan perlunya pengembangan sistem informasi yang tidak hanya mampu melakukan pencatatan transaksi secara digital, tetapi juga mampu mengoptimalkan proses pengurutan data menggunakan algoritma yang memiliki tingkat efisiensi tinggi.

Dalam penelitian ini, pengembangan sistem dilakukan menggunakan metode *Waterfall* karena memiliki tahapan yang sistematis mulai dari analisis kebutuhan, perancangan, implementasi, pengujian, hingga pemeliharaan sistem (Pricillia & Zulfachmi, 2021; Sopian & Ernawati, 2022). Perancangan sistem menggunakan *Unified Modeling Language (UML)* sebagai alat pemodelan untuk menggambarkan kebutuhan sistem secara terstruktur (Hidayati et al., 2023). Selanjutnya, pengujian sistem dilakukan menggunakan metode *Black Box Testing* untuk memastikan seluruh fungsi aplikasi berjalan sesuai dengan kebutuhan pengguna (Ichsanudin & Yusuf, 2022; Asrin, 2024).

Research gap pada penelitian ini terletak pada masih terbatasnya penelitian yang mengintegrasikan implementasi algoritma *Quick Sort* pada sistem pencatatan transaksi perusahaan berbasis *web* dengan proses pengembangan sistem menggunakan metode *Waterfall*, pemodelan *Unified Modeling Language (UML)*, serta pengujian menggunakan *Black Box Testing*. Adapun *novelty* penelitian ini adalah pengembangan sistem pencatatan transaksi berbasis *web* yang mengimplementasikan algoritma *Quick Sort* untuk meningkatkan efisiensi proses pengurutan data transaksi pada Perusahaan Jelita Farsel. Oleh karena itu, penelitian ini bertujuan untuk mengimplementasikan algoritma *Quick Sort* pada sistem pencatatan transaksi berbasis *web* sehingga mampu mendukung proses pengelolaan data transaksi yang lebih cepat, akurat, dan efisien.

2. Metode Penelitian

Penelitian ini menggunakan metode *Research and Development (R&D)* yang bertujuan untuk mengembangkan sistem pencatatan transaksi berbasis *web* dengan mengimplementasikan algoritma *Quick Sort* sebagai metode pengurutan data transaksi. Pendekatan ini dipilih karena tidak hanya menghasilkan produk berupa perangkat lunak, tetapi juga menguji fungsionalitas sistem yang dikembangkan sehingga dapat digunakan sesuai kebutuhan pengguna.

Metode pengembangan perangkat lunak yang digunakan adalah *Waterfall*. Metode ini dipilih karena memiliki tahapan yang sistematis dan terstruktur, sehingga sesuai untuk pengembangan sistem yang kebutuhan pengguna telah teridentifikasi dengan jelas (Pricillia & Zulfachmi, 2021; Sopian & Ernawati, 2022). Tahapan penelitian meliputi analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan.

Tahap analisis kebutuhan dilakukan melalui observasi dan wawancara di Perusahaan Jelita Farsel untuk mengidentifikasi permasalahan pada proses pencatatan transaksi. Hasil analisis menunjukkan bahwa proses pengurutan data transaksi masih dilakukan secara manual sehingga pencarian data memerlukan waktu yang relatif lama ketika jumlah transaksi meningkat. Berdasarkan hasil analisis tersebut, dirancang sistem pencatatan transaksi berbasis *web* yang mampu melakukan proses pengurutan data secara otomatis menggunakan algoritma *Quick Sort*.

Tahap perancangan sistem dilakukan menggunakan *Unified Modeling Language (UML)* yang terdiri atas *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram* untuk menggambarkan kebutuhan fungsional sistem secara terstruktur (Hidayati et al., 2023). Sistem kemudian diimplementasikan menggunakan bahasa pemrograman PHP dengan *database MySQL* sehingga mampu mendukung proses pengelolaan data transaksi secara terintegrasi.

Implementasi algoritma *Quick Sort* dilakukan pada fitur pengurutan data transaksi berdasarkan beberapa kriteria, seperti tanggal transaksi, nama pelanggan, dan nominal transaksi. Algoritma ini dipilih karena memiliki kompleksitas waktu rata-rata sebesar $O(n \log n)$ sehingga mampu meningkatkan efisiensi proses pengurutan data dibandingkan metode pengurutan sederhana (Nasution et al., 2024; Pujiono et al., 2024).

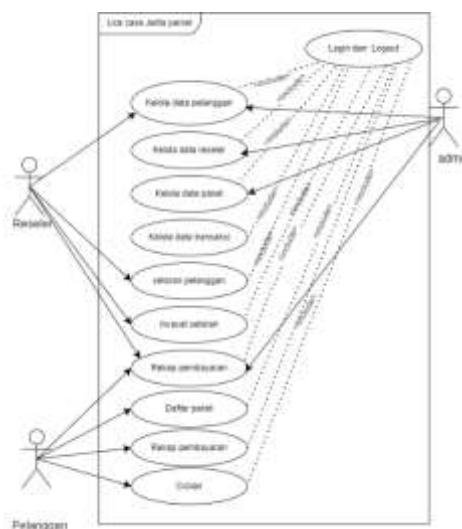
Pengujian sistem dilakukan menggunakan metode *Black Box Testing* untuk mengevaluasi kesesuaian fungsi-fungsi sistem dengan kebutuhan pengguna tanpa memperhatikan struktur kode program (Ichsanudin & Yusuf, 2022). Selain itu, dilakukan *User Acceptance Test (UAT)* untuk mengetahui tingkat penerimaan pengguna terhadap sistem yang dikembangkan (Asrin, 2024). Hasil pengujian digunakan sebagai dasar dalam mengevaluasi kinerja sistem serta memastikan bahwa seluruh fitur dapat berfungsi dengan baik sesuai kebutuhan Perusahaan Jelita Farsel.

3. Hasil dan Pembahasan

Penelitian ini menghasilkan sebuah sistem pencatatan transaksi berbasis *web* yang mengimplementasikan algoritma *Quick Sort* sebagai metode pengurutan data transaksi pada Perusahaan Jelita Farsel. Sistem dikembangkan menggunakan metode *Waterfall* yang terdiri atas tahap analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Berdasarkan hasil observasi dan wawancara, proses pencatatan transaksi sebelumnya masih dilakukan menggunakan buku pencatatan dan *Microsoft Excel*. Kondisi tersebut menyebabkan proses pencarian data, pengurutan transaksi, serta penyusunan laporan menjadi kurang efektif ketika jumlah transaksi meningkat. Oleh karena itu, sistem yang dikembangkan dirancang untuk membantu proses pengelolaan data transaksi secara lebih cepat, akurat, dan terstruktur.

Analisis dan Perancangan Sistem

Tahap awal pengembangan dilakukan dengan menganalisis kebutuhan pengguna melalui observasi dan wawancara. Berdasarkan hasil analisis, diperoleh tiga aktor utama yang berinteraksi dengan sistem, yaitu *admin*, *reseller*, dan *pelanggan*. Selanjutnya dilakukan perancangan sistem menggunakan *Unified Modeling Language (UML)* sebagai dasar implementasi aplikasi.



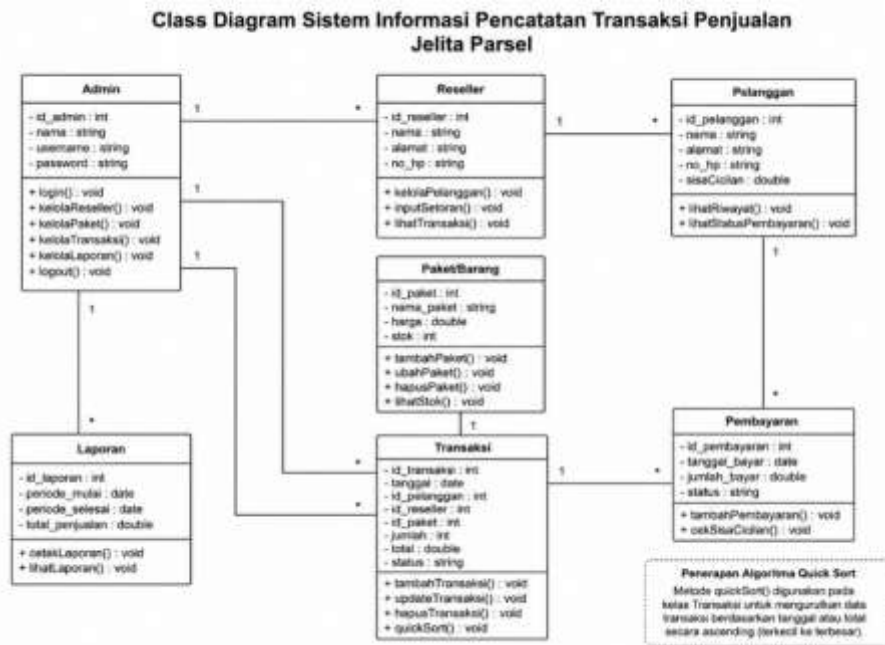
Gambar 1. Use Case Diagram Sistem Pencatatan Transaksi

DOI: <https://doi.org/10.69693/ijmst.v4i3.13262>

Lisensi: Creative Commons Attribution 4.0 International (CC BY 4.0)

Berdasarkan *use case diagram*, *admin* memiliki hak akses penuh terhadap pengelolaan data pelanggan, *reseller*, paket, transaksi, dan laporan. Selain itu, *admin* juga bertanggung jawab terhadap proses autentikasi sistem. *Reseller* dapat mengelola pelanggan, melakukan pencatatan setoran, dan melihat riwayat transaksi. Sementara itu, *pelanggan* dapat melihat daftar paket, status pembayaran, cicilan, serta riwayat pembayaran. Rancangan tersebut menjadi dasar dalam pembangunan sistem pencatatan transaksi berbasis *web*.

Selanjutnya dilakukan perancangan struktur sistem menggunakan *class diagram* untuk menggambarkan hubungan antarentitas yang terdapat pada sistem.



Gambar 2. Class Diagram Sistem Pencatatan Transaksi

Class diagram menunjukkan bahwa sistem terdiri atas tujuh entitas utama, yaitu *Admin*, *Reseller*, *Pelanggan*, *Paket*, *Transaksi*, *Pembayaran*, dan *Laporan*. Implementasi algoritma *Quick Sort* ditempatkan pada kelas *Transaksi* sehingga proses pengurutan data dapat dilakukan berdasarkan tanggal transaksi, nama pelanggan, maupun nominal transaksi secara otomatis.

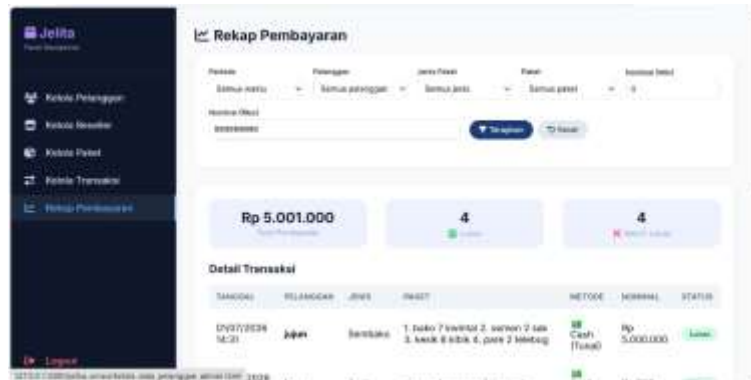
Implementasi Sistem

Tahap implementasi dilakukan menggunakan bahasa pemrograman PHP dengan *database* MySQL. Sistem yang dikembangkan menyediakan berbagai fitur yang mendukung proses pengelolaan transaksi, mulai dari pengelolaan data pelanggan, pengelolaan paket, transaksi, hingga rekapitulasi pembayaran.



Gambar 3. Halaman Kelola Pelanggan

Halaman kelola pelanggan digunakan oleh *admin* untuk menambahkan, mengubah, menghapus, dan mencari data pelanggan. Fitur pencarian memudahkan pengguna memperoleh informasi pelanggan secara cepat sehingga proses administrasi menjadi lebih efisien.



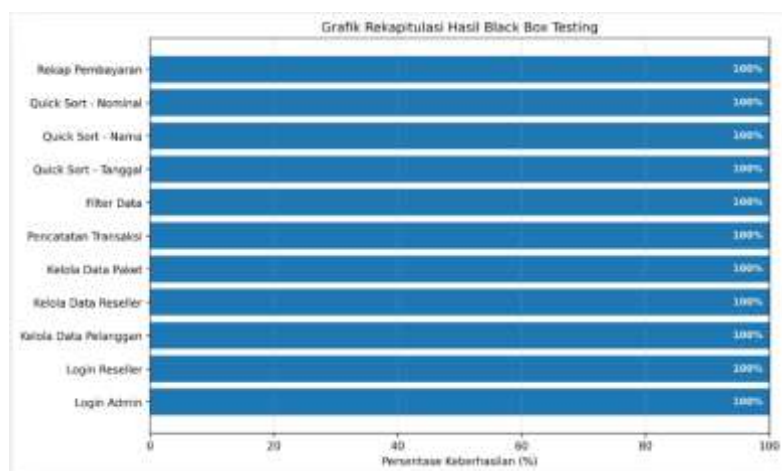
Gambar 4. Halaman Rekap Pembayaran

Halaman rekap pembayaran menampilkan informasi transaksi berdasarkan periode tertentu yang dilengkapi dengan fitur penyaringan (*filter*) berdasarkan pelanggan, jenis paket, maupun nominal transaksi. Selain itu, sistem menampilkan total pendapatan, jumlah transaksi lunas, dan transaksi yang belum lunas sehingga memudahkan pengguna dalam melakukan monitoring transaksi.

Implementasi algoritma *Quick Sort* diterapkan pada proses pengurutan data transaksi berdasarkan tanggal transaksi, nama pelanggan, dan nominal transaksi. Penggunaan algoritma tersebut memungkinkan sistem menyusun data secara otomatis sehingga proses pencarian informasi menjadi lebih cepat dan efisien dibandingkan pengurutan secara manual.

Pengujian Sistem

Setelah proses implementasi selesai, sistem diuji menggunakan metode *Black Box Testing* untuk memastikan seluruh fungsi berjalan sesuai dengan kebutuhan pengguna. Pengujian dilakukan terhadap fitur *login*, pengelolaan data pelanggan, pengelolaan *reseller*, pencatatan transaksi, proses *Quick Sort*, serta rekap pembayaran.



Gambar 5. Grafik Rekapitulasi Hasil Black Box Testing

Berdasarkan hasil pengujian, seluruh fitur memperoleh tingkat keberhasilan sebesar 100%. Hasil tersebut menunjukkan bahwa setiap fungsi pada sistem berjalan sesuai dengan rancangan dan tidak ditemukan kesalahan fungsional selama proses pengujian. Dengan demikian, sistem yang dikembangkan telah memenuhi kebutuhan pengguna dalam melakukan pencatatan transaksi serta pengelolaan data secara terintegrasi.

Pembahasan

Hasil penelitian menunjukkan bahwa implementasi algoritma *Quick Sort* pada sistem pencatatan transaksi berbasis *web* mampu meningkatkan efisiensi proses pengelolaan data transaksi di Perusahaan Jelita Farsel. Algoritma ini memungkinkan proses pengurutan data dilakukan secara otomatis berdasarkan tanggal transaksi, nama pelanggan, maupun nominal transaksi sehingga waktu yang dibutuhkan untuk memperoleh informasi menjadi lebih singkat dibandingkan proses pengurutan manual.

Selain meningkatkan efisiensi pengurutan data, sistem juga mempermudah proses pencarian transaksi, pengelolaan data pelanggan, penyusunan laporan, serta monitoring pembayaran. Seluruh fitur yang dikembangkan berhasil berfungsi dengan baik berdasarkan hasil *Black Box Testing* yang menunjukkan tingkat keberhasilan sebesar 100%. Hal ini menunjukkan bahwa sistem telah memenuhi kebutuhan pengguna serta mampu mendukung aktivitas operasional perusahaan secara lebih efektif.

Hasil penelitian ini sejalan dengan penelitian Nasution et al. (2024) yang menyatakan bahwa algoritma *Quick Sort* mampu meningkatkan efisiensi proses pengurutan data pada sistem informasi. Temuan ini juga mendukung penelitian Pujiono et al. (2024) yang menunjukkan bahwa *Quick Sort* memiliki performa yang baik dari sisi waktu komputasi dan efisiensi memori. Selain itu, penelitian Yusuf dan Naimah (2025) menjelaskan bahwa penerapan algoritma *Quick Sort* pada aplikasi berbasis *web* mampu mempercepat proses pengolahan data sehingga sistem dapat memberikan informasi secara lebih cepat dan akurat.

Analisis Efisiensi Pengurutan Data dengan Quick Sort

Penerapan *Quick Sort* pada sistem pencatatan transaksi perlu dipahami tidak hanya sebagai penambahan algoritma di dalam aplikasi, tetapi sebagai bagian dari strategi pengelolaan data yang menentukan seberapa cepat informasi dapat ditampilkan kembali kepada pengguna. Pada sistem transaksi, data terus bertambah mengikuti aktivitas penjualan, pembayaran, dan pencatatan pelanggan. Ketika jumlah data meningkat, proses pengurutan manual menjadi semakin tidak praktis karena pengguna harus melakukan pemeriksaan berulang terhadap data berdasarkan tanggal, nama pelanggan, atau nominal transaksi. *Quick Sort* relevan untuk kebutuhan ini karena secara umum memiliki kompleksitas waktu rata-rata $O(n \log n)$, sehingga mampu menangani pertumbuhan data lebih baik dibandingkan algoritma sederhana yang cenderung memiliki kompleksitas $O(n^2)$. Hasil perbandingan algoritma sorting oleh Pujiono et al. (2024) menunjukkan bahwa pemilihan algoritma berpengaruh terhadap waktu komputasi dan penggunaan memori, sedangkan Goel dan Dwivedi (2023) menegaskan bahwa performa sorting juga dipengaruhi oleh ukuran data, pola input, dan bahasa pemrograman yang digunakan. Dengan demikian, penggunaan *Quick Sort* pada penelitian ini memiliki dasar algoritmik yang sesuai untuk membantu proses pengurutan data transaksi yang bersifat dinamis.

Dalam konteks sistem Jelita Farsel, pengurutan tidak hanya berfungsi untuk menghasilkan urutan data yang rapi, tetapi juga menjadi bagian dari proses pencarian dan pengambilan keputusan operasional. Data yang telah terurut menurut tanggal membantu pengguna menelusuri riwayat transaksi secara kronologis, pengurutan berdasarkan nama pelanggan memudahkan pencarian hubungan transaksi dengan pelanggan tertentu, sedangkan pengurutan berdasarkan nominal membantu identifikasi transaksi dengan nilai terbesar atau terkecil. Fungsi tersebut memperlihatkan bahwa sorting memiliki hubungan langsung dengan keterbacaan informasi. Namun, penelitian ini belum menyajikan pengukuran waktu eksekusi sebelum dan sesudah implementasi *Quick Sort* pada beberapa ukuran dataset. Oleh karena itu, klaim peningkatan kecepatan pada penelitian ini terutama didukung oleh karakteristik algoritma dan hasil pengujian fungsional, belum berupa benchmark kuantitatif waktu respons. Pada pengembangan berikutnya, pengukuran dapat dilakukan menggunakan dataset bertingkat, misalnya 100, 1.000, 10.000, dan 100.000 transaksi, sehingga dapat diketahui pola peningkatan waktu eksekusi secara empiris.

Aspek lain yang perlu diperhatikan adalah strategi pemilihan pivot. *Quick Sort* dapat memberikan performa rata-rata yang baik, tetapi pada kondisi tertentu dapat mengalami penurunan kinerja apabila pembagian data sangat tidak seimbang. Karena naskah penelitian belum menjelaskan teknik pemilihan pivot secara rinci, aspek ini dapat menjadi ruang pengembangan. Penerapan pivot median, randomized pivot, atau pendekatan hybrid dapat dipertimbangkan apabila jumlah data menjadi jauh lebih besar. Studi komparatif modern menunjukkan bahwa variasi implementasi dan karakteristik input dapat memengaruhi performa aktual algoritma sorting, sehingga kompleksitas teoretis perlu dibaca bersama konteks implementasi (Goel & Dwivedi, 2023). Dengan demikian,

Quick Sort pada sistem ini sudah relevan untuk kebutuhan operasional saat ini, tetapi evaluasi teknis yang lebih rinci akan memperkuat bukti bahwa algoritma tersebut tetap efisien ketika volume transaksi bertambah secara signifikan.

Kesesuaian Implementasi Quick Sort pada Sistem Transaksi Berbasis Web

Keberhasilan implementasi algoritma juga dipengaruhi oleh arsitektur sistem tempat algoritma dijalankan. Sistem yang dikembangkan menggunakan PHP dan MySQL, sehingga proses pengurutan berada dalam ekosistem aplikasi web yang melibatkan antarmuka pengguna, logika aplikasi, dan basis data. Penelitian-penelitian terbaru mengenai sistem transaksi berbasis web menunjukkan bahwa digitalisasi pencatatan dapat mengurangi ketergantungan pada dokumen fisik, mempermudah penyimpanan, mempercepat penelusuran informasi, dan meningkatkan konsistensi data. William, Sharyanto, dan Whendasmoro (2024) menunjukkan bahwa sistem transaksi berbasis web dengan PHP, MySQL, dan Waterfall digunakan untuk mengatasi proses transaksi manual yang lambat dan rawan kehilangan data. Temuan serupa ditunjukkan Chrislie dan Birowo (2024) pada sistem Point of Sales berbasis web yang membantu pencatatan penjualan dan persediaan secara terkomputerisasi. Kesamaan konteks tersebut memperkuat bahwa pengembangan sistem di Jelita Farsel memiliki relevansi praktis, terutama karena masalah awalnya juga terkait pencatatan dan pencarian transaksi yang belum efisien.

Integrasi Quick Sort di dalam sistem web perlu mempertimbangkan lokasi pemrosesan data. Jika pengurutan dilakukan pada sisi aplikasi setelah data diambil dari basis data, maka aplikasi memperoleh fleksibilitas untuk menerapkan aturan sorting khusus. Sebaliknya, jika volume data sangat besar, pengurutan langsung pada basis data dapat menjadi alternatif karena DBMS memiliki mekanisme optimasi query dan indeks. Penelitian ini menempatkan Quick Sort sebagai komponen logika pada proses transaksi, sehingga kontribusi utamanya adalah menunjukkan penerapan algoritma eksplisit di dalam alur aplikasi. Pendekatan ini bermanfaat secara akademik karena hubungan antara konsep algoritma dan fungsi sistem dapat diamati secara langsung. Akan tetapi, untuk penggunaan produksi berskala besar, pengembang perlu membandingkan efisiensi sorting pada level aplikasi dengan perintah pengurutan basis data agar tidak terjadi pemindahan data berlebihan dari MySQL ke aplikasi PHP.

Pemanfaatan PHP dan MySQL juga konsisten dengan berbagai penelitian sistem transaksi dan penjualan berbasis web. Ningsi, Hendryadi, dan Iryani (2023) mengembangkan sistem administrasi berbasis web menggunakan PHP, MySQL, Waterfall, dan Black Box Testing untuk meningkatkan efisiensi pengelolaan data. Andy dan Widiono (2024) menunjukkan bahwa aplikasi Point of Sales berbasis web dapat mengintegrasikan pengelolaan stok dan transaksi serta mendukung akurasi pencatatan. Ahadi dan Gustina (2023) juga menggunakan PHP, MySQL, dan Waterfall pada sistem Point of Sales berbasis web untuk mengatasi masalah pencatatan transaksi manual. Dalam kaitannya dengan penelitian Jelita Farsel, literatur tersebut memperlihatkan bahwa pemilihan platform teknologi telah sesuai dengan kebutuhan aplikasi transaksi yang menuntut penyimpanan terstruktur, akses data cepat, dan pengelolaan data secara terpusat.

Dampak Implementasi terhadap Pengelolaan Transaksi dan Kualitas Informasi

Hasil penelitian menunjukkan bahwa manfaat sistem tidak berhenti pada fungsi pengurutan, tetapi meluas pada pengelolaan pelanggan, pencatatan transaksi, rekapitulasi pembayaran, dan penyusunan laporan. Integrasi fungsi-fungsi tersebut penting karena efisiensi operasional tidak hanya ditentukan oleh kecepatan satu algoritma, melainkan juga oleh kelancaran aliran informasi dari input sampai laporan. Pada sistem manual, pencarian kembali transaksi membutuhkan pemeriksaan catatan atau file secara berulang, sedangkan pada sistem berbasis web data tersimpan di dalam basis data dan dapat ditampilkan kembali menggunakan parameter tertentu. Pengurutan otomatis kemudian memperbaiki cara data tersebut disajikan kepada pengguna. Dengan demikian, Quick Sort berperan sebagai komponen pendukung yang memperkuat manfaat digitalisasi pencatatan, bukan sebagai satu-satunya sumber peningkatan efisiensi sistem.

Peningkatan kualitas informasi dapat dilihat dari empat aspek, yaitu keteraturan, aksesibilitas, konsistensi, dan ketepatan penyajian. Keteraturan diperoleh karena data transaksi memiliki struktur yang seragam dan dapat diurutkan berdasarkan atribut yang dibutuhkan. Aksesibilitas meningkat karena pengguna tidak harus mencari catatan fisik atau membuka banyak lembar kerja. Konsistensi didukung oleh penyimpanan data dalam basis data yang sama, sedangkan ketepatan penyajian diperkuat melalui proses sorting dan filtering. Sistem informasi transaksi berbasis web dalam penelitian William et al. (2024) dan sistem POS pada Chrislie dan Birowo (2024) sama-sama menekankan manfaat komputerisasi terhadap efektivitas pengarsipan dan transaksi. Hal tersebut

mendukung interpretasi bahwa sistem Jelita Farsel dapat membantu mengurangi beban administratif dalam pengelolaan transaksi sehari-hari.

Meskipun demikian, tingkat keberhasilan 100% pada Black Box Testing tidak dapat langsung diartikan sebagai bukti bahwa sistem memiliki performa maksimum pada seluruh kondisi. Black Box Testing berfokus pada kesesuaian input, proses, dan output terhadap kebutuhan fungsional. Pengujian ini tepat untuk memastikan fitur login, pengelolaan pelanggan, transaksi, Quick Sort, dan rekap pembayaran berjalan sesuai rancangan, tetapi tidak mengukur keamanan secara menyeluruh, ketahanan sistem pada banyak pengguna, waktu respons ketika jumlah data meningkat, atau kualitas pengalaman pengguna. Syahroni et al. (2024) menunjukkan bahwa pengujian black box pada aplikasi web juga dapat diarahkan pada aspek autentikasi dan otorisasi untuk menemukan kelemahan fungsi keamanan. Oleh karena itu, hasil pengujian pada penelitian ini dapat dipandang sebagai bukti kesiapan fungsional awal, sedangkan pengujian kinerja, keamanan, dan beban tetap diperlukan untuk tahap pengembangan berikutnya.

Evaluasi Metode Pengembangan dan Pengujian Sistem

Penggunaan model Waterfall pada penelitian ini memberikan alur pengembangan yang jelas, mulai dari analisis kebutuhan, perancangan, implementasi, pengujian, hingga pemeliharaan. Model ini sesuai ketika kebutuhan utama pengguna telah dapat diidentifikasi sejak awal, seperti kebutuhan pencatatan pelanggan, transaksi, pembayaran, laporan, dan pengurutan data. Sejumlah penelitian web dalam lima tahun terakhir masih menggunakan Waterfall karena tahapan yang terstruktur memudahkan dokumentasi dan pengendalian pekerjaan. Asqia (2023) menggunakan Waterfall pada pengembangan sistem korespondensi berbasis web dan melaporkan bahwa fitur yang diuji dengan Black Box Testing berjalan sesuai tujuan. Utami et al. (2023) juga menggunakan Waterfall untuk sistem reservasi berbasis web dengan tahapan analisis kebutuhan, desain, implementasi, pengujian, dan pemeliharaan. Pada konteks Jelita Farsel, penggunaan Waterfall mendukung hubungan yang runtut antara kebutuhan awal perusahaan dan fitur yang akhirnya dibangun.

Pemodelan menggunakan UML memperkuat tahap perancangan karena aktor, alur interaksi, serta hubungan antarentitas dapat divisualisasikan sebelum proses pengkodean. Use case diagram pada penelitian ini menunjukkan perbedaan hak akses admin, reseller, dan pelanggan, sedangkan class diagram memperlihatkan keterkaitan entitas utama yang digunakan oleh sistem. Struktur ini penting agar implementasi Quick Sort tidak berdiri sendiri, melainkan terhubung dengan kelas transaksi dan data lain yang diperlukan. Hidayati et al. (2023) menunjukkan penggunaan UML untuk menyusun struktur sistem informasi berbasis web secara lebih terarah. Pemodelan sebelum implementasi juga membantu mengurangi risiko ketidaksesuaian fungsi karena hubungan antarbagian sistem telah ditentukan sejak tahap desain.

Selain Black Box Testing, penelitian menyebutkan User Acceptance Test sebagai bagian evaluasi penerimaan sistem. Penggunaan UAT penting karena keberhasilan sistem informasi tidak hanya ditentukan oleh benar atau salahnya fungsi, tetapi juga oleh kesesuaian sistem dengan kebutuhan pengguna akhir. Pada aplikasi transaksi, aspek yang relevan meliputi kemudahan navigasi, kejelasan menu, kecepatan memperoleh informasi, dan kesesuaian laporan dengan kebutuhan operasional. Studi sistem Point of Sales berbasis web menunjukkan bahwa UAT dapat digunakan untuk mengukur tingkat kesesuaian aplikasi dengan kebutuhan pengguna setelah pengujian fungsional selesai. Karena hasil UAT pada naskah ini belum disajikan secara kuantitatif, penelitian berikutnya dapat menambahkan instrumen penilaian dengan skala Likert dan menyajikan persentase penerimaan pengguna untuk melengkapi hasil Black Box Testing.

Perbandingan dengan Penelitian Terdahulu

Dibandingkan penelitian terdahulu, kontribusi penelitian ini berada pada penggabungan pengembangan sistem transaksi berbasis web dengan penerapan Quick Sort secara eksplisit sebagai fitur pengurutan. Banyak penelitian sistem transaksi lebih berfokus pada digitalisasi proses pencatatan, sedangkan penelitian algoritma sorting sering hanya membandingkan performa pada dataset atau bahasa pemrograman tertentu. Pujiono et al. (2024) membandingkan efisiensi memori dan waktu komputasi beberapa algoritma sorting, sedangkan Goel dan Dwivedi (2023) melakukan analisis performa berbagai algoritma pada beberapa bahasa pemrograman. Di sisi lain, William et al. (2024), Chrislie dan Birowo (2024), serta Andy dan Widiono (2024) berfokus pada pengembangan sistem transaksi atau Point of Sales berbasis web. Penelitian Jelita Farsel menempatkan kedua perspektif tersebut dalam satu implementasi, yaitu menggunakan algoritma sorting sebagai bagian dari fungsi sistem informasi transaksi.

Kesamaan dengan penelitian Nasution et al. (2024) terlihat pada penggunaan Quick Sort untuk meningkatkan efisiensi pengurutan data. Namun, penelitian ini memiliki konteks penerapan yang lebih spesifik pada transaksi perusahaan dengan beberapa atribut pengurutan. Penelitian Rosdiana (2024) dan Wijoyo et al. (2024) juga menunjukkan bahwa Quick Sort dapat diterapkan pada sistem aplikasi untuk membantu pengolahan data. Sementara itu, Yusuf dan Naimah (2025) menunjukkan implementasi Quick Sort dalam aplikasi yang terhubung dengan pengelolaan data. Konsistensi hasil pada berbagai konteks tersebut menunjukkan bahwa Quick Sort cukup fleksibel untuk diintegrasikan ke dalam aplikasi yang memerlukan pengurutan data secara berulang.

Dari sisi sistem, penelitian Ningsi et al. (2023), Asqia (2023), dan Mustapa et al. (2023) menunjukkan bahwa sistem berbasis web dengan tahapan pengembangan terstruktur dapat memperbaiki proses administrasi dan akses informasi. Perbedaan utama penelitian ini adalah adanya kebutuhan pengurutan transaksi sebagai masalah inti yang secara khusus ditangani melalui algoritma Quick Sort. Hal tersebut memperkuat novelty yang telah dinyatakan pada bagian pendahuluan, yaitu integrasi antara pengembangan sistem berbasis web, Waterfall, UML, pengujian fungsional, dan algoritma pengurutan untuk mendukung pengelolaan transaksi pada satu objek perusahaan.

Keterbatasan Penelitian dan Implikasi Pengembangan

Penelitian ini masih memiliki beberapa keterbatasan yang penting untuk diperhatikan. Pertama, pengujian belum menyajikan benchmark waktu eksekusi Quick Sort pada ukuran data yang berbeda, sehingga besarnya peningkatan performa belum dapat dinyatakan secara kuantitatif. Kedua, belum dilakukan perbandingan langsung dengan algoritma lain seperti Merge Sort, Heap Sort, atau algoritma bawaan bahasa pemrograman. Padahal, studi komparatif menunjukkan bahwa kinerja aktual algoritma sangat dipengaruhi jumlah data, pola data, implementasi bahasa, dan lingkungan komputasi (Goel & Dwivedi, 2023; Pujiono et al., 2024). Ketiga, pengujian sistem masih dominan pada aspek fungsional, sehingga pengujian beban, keamanan, dan penerimaan pengguna perlu diperluas.

Implikasi praktis dari penelitian ini adalah bahwa perusahaan dapat menggunakan sistem sebagai dasar digitalisasi transaksi yang lebih terstruktur sekaligus sebagai landasan pengembangan fitur lanjutan. Pada tahap berikutnya, sistem dapat ditambah dengan indeks basis data, pagination untuk data berjumlah besar, pencarian asinkron, export laporan, audit trail, dashboard analitik, serta integrasi pembayaran digital. Fitur keamanan juga perlu diperkuat melalui pengelolaan hak akses, validasi input, pencatatan aktivitas pengguna, dan pengujian autentikasi serta otorisasi sebagaimana ditekankan Syahrani et al. (2024). Pengembangan tersebut akan membuat sistem tidak hanya mampu melakukan pencatatan dan pengurutan data, tetapi juga mendukung kontrol internal dan pengambilan keputusan berbasis informasi.

Secara akademik, penelitian selanjutnya dapat melakukan eksperimen komparatif dengan skenario data terurut, terbalik, acak, dan memiliki banyak nilai duplikat. Setiap skenario dapat diuji pada beberapa ukuran dataset untuk memperoleh waktu rata-rata eksekusi, penggunaan memori, dan kestabilan hasil. Selain itu, perbandingan antara Quick Sort di sisi aplikasi dan fungsi ORDER BY yang dioptimasi basis data dapat memberikan gambaran lebih lengkap mengenai lokasi pemrosesan yang paling efisien untuk sistem transaksi berbasis web. Dengan desain pengujian tersebut, kontribusi penelitian tidak hanya berupa keberhasilan implementasi sistem, tetapi juga bukti empiris mengenai batas dan kondisi ketika Quick Sort memberikan keuntungan performa paling besar. Pendekatan ini akan memperkuat validitas teknis sekaligus membantu pengembang menentukan strategi pengurutan yang sesuai dengan pertumbuhan data transaksi di masa mendatang.

4. Kesimpulan

Penelitian ini berhasil mengembangkan sistem pencatatan transaksi berbasis web pada Perusahaan Jelita Farsel dengan mengimplementasikan algoritma *Quick Sort* sebagai metode pengurutan data transaksi. Sistem yang dikembangkan mampu mendukung pengelolaan data pelanggan, data paket, transaksi, serta rekap pembayaran secara terintegrasi. Implementasi algoritma *Quick Sort* memungkinkan proses pengurutan data berdasarkan tanggal transaksi, nama pelanggan, dan nominal transaksi dilakukan secara lebih cepat, akurat, dan efisien dibandingkan proses manual. Hasil pengujian menggunakan metode *Black Box Testing* menunjukkan bahwa seluruh fungsi sistem berjalan dengan baik dan memperoleh tingkat keberhasilan sebesar 100%, sehingga sistem dinyatakan telah memenuhi kebutuhan pengguna. Dengan demikian, sistem yang dikembangkan dapat meningkatkan efektivitas proses pencatatan transaksi serta mempermudah pengelolaan informasi di Perusahaan Jelita Farsel. Penelitian selanjutnya diharapkan dapat mengembangkan sistem dengan menambahkan fitur analisis data, integrasi pembayaran digital, serta membandingkan performa *Quick Sort* dengan algoritma pengurutan

lainnya untuk memperoleh hasil yang lebih optimal.

Reference

1. Asrin, F. (2024). Analisis pengujian menggunakan *User Acceptance Test* (UAT) pada aplikasi manajemen notulensi. 8(1), 34–41.
2. Dorman, A. (2025). *Comparative performance analysis of sorting algorithms for data processing efficiency in sales information systems using Python* [Preprint]. Zenodo. <https://doi.org/10.5281/zenodo.17610539>
3. Hidayati, A. T., Widyantoro, A. E., & Ramadhani, H. J. (2023). Perancangan sistem informasi wirausaha mahasiswa (SIWIRMA) berbasis *web* dengan *Unified Modeling Language* (UML). 2(4).
4. Ichsanudin, M. N., & Yusuf, M. (2022). Perpustakaan dengan metode *Black Box Testing* bagi pemula. 1(2), 1–8.
5. Masturoh, M. R. Y., & Sastika, S. (2026). Perancangan sistem informasi *e-commerce* berbasis *website* pada PT Bimantara Sakti Persada menggunakan metode *Waterfall*. 20(2), 195–203.
6. Nasution, A. N., Singgam, P., Al-Hafiz, A. Y., Alvansyah, O., & Tampubolon, J. D. (2024). Penerapan algoritma *Quick Sort* untuk menyortir *array* berdasarkan kriteria tertentu untuk meningkatkan efisiensi komputasi. *VISA: Journal of Vision and Ideas*, 4(3), 1197–1204. <https://doi.org/10.47467/visa.v4i3.5409>
7. Noviana, R. (2022). Pembuatan aplikasi penjualan berbasis *web* Monja Store menggunakan PHP dan MySQL. 1(2), 112–124.
8. Oswari, A. T. H. T. N. W. I. Z. I. T. F. S. E. H. T. (2025). Tinjauan terhadap peran *chatbot* AI sebagai asisten virtual dalam pembelajaran digital. 8, 5721–5727.
9. Petrus, R., et al. (2026). Pengembangan sistem informasi persediaan barang berbasis *web* sebagai upaya meningkatkan efektivitas distribusi produk Tupperware. 6(1), 9–21.
10. Pricillia, T., & Zulfachmi. (2021). *Survey paper*: Perbandingan metode pengembangan perangkat lunak. 10(1), 6–12.
11. Pujiono, I. P., Trianto, R. B., & Hana, F. M. (2024). Perbandingan efisiensi memori dan waktu komputasi pada tujuh algoritma *sorting* menggunakan bahasa pemrograman Java. *Jurnal Sistem Informasi dan Sistem Komputer*, 9(2), 218–230. <https://doi.org/10.51717/simkom.v9i2.481>
12. Rosdiana, D. (2024). Implementasi algoritma *Quick Sort* dalam sistem pemesanan makanan di rumah makan. 1(2), 60–65.
13. Santoni, A., & Mulyono, H. (2022). Sistem informasi perpustakaan berbasis *website* pada SMKN 1 Pancung Soal. 2(2), 103–107.
14. Sopian, A., & Ernawati, S. (2022). Implementasi model *Waterfall* dalam rancang bangun *e-commerce*. 2(2), 70–75.
15. Tim Jurnal Algoritme. (2023). Perbandingan penempatan *pivot* pada *Quick Sort* berdasarkan ukuran pemusatan data. 4(1), 13–20. <https://doi.org/10.35957/algoritme.xxxx>
16. Waisen Awan, Hendra, Benny, & Yudi. (2026). Pelatihan pembuatan *website* sederhana dengan HTML, CSS, dan VSCode untuk siswa SMA Bodhicita. 6.
17. Wijoyo, A., Marsiano, J., Ikhwani, M., & Rianto, R. (2024). Implementasi algoritma *Quick Sort* pada sistem akademik. *JRIIN: Jurnal Riset Informatika dan Inovasi*, 1(11), 1216–1218.
18. Yusuf, M., & Naimah, F. (2025). Implementasi algoritma *Quick Sort* pada aplikasi pemrograman berorientasi objek berbasis Python dan MySQL Workbench. *FORMAT: Jurnal Ilmiah Teknik Informatika*, 14(1), 90–110. <https://doi.org/10.22441/format.2025.v14.i1.009>
19. Zurna, H. P. B., Rini, F., & Pratama, A. (2022). Sistem informasi perpustakaan berbasis *web*. 2(1), 5–10.
20. Goel, K., & Dwivedi, P. (2023). Performance Analysis of Various Sorting Algorithms: Comparison and Optimization. 2023 11th International Conference on Intelligent Systems and Embedded Design (ISED). <https://doi.org/10.1109/ISED59382.2023.10444609>
21. William, D., Sharyanto, S., & Whendasmoro, R. G. (2024). Rancang Bangun Sistem Informasi Transaksi Berbasis Web Studi Kasus Kopi Bar Depok. *Journal of Engineering, Technology and Computing (JETCom)*, 3(2), 20–29. <https://doi.org/10.63893/jetcom.v3i2.234>
22. Chrislie, E., & Birowo, S. (2024). Sistem Informasi Point of Sales Berbasis Web di Perusahaan Bangunan Andalas Jaya. *Jurnal Informatika dan Bisnis*, 13(1), 60–71. <https://doi.org/10.46806/jib.v13i1.1181>
23. Ningsi, S. H., Hendryadi, D., & Iryani, J. (2023). Web-Based Administrative Management Information System Using the Waterfall Method at the Bina Adinata Institute of Technology and Business. *Journal System Information and Computer*, 1(2), 68–79. <https://doi.org/10.63989/ammatao.v1i2.8>
24. Asqia, M. (2023). Development of a Web-Based Correspondence Information System to Enhance Administrative Services in Higher Education. *The Indonesian Journal of Computer Science*, 12(6). <https://doi.org/10.33022/ijcs.v12i6.3521>
25. Syahroni, A. W., Dewi, N. P., Ramadhani, N., Ubaidi, & Said, B. (2024). Uji Keamanan Back End Aplikasi Berbasis Website Menggunakan Metode Black Box Testing. *Jurnal PROCESSOR*, 19(2). <https://doi.org/10.33998/processor.2024.19.2.1752>
26. Andy, F. A. M., & Widiono, S. (2024). Inovasi Teknologi dalam Manajemen Penjualan: Aplikasi Point of Sales Berbasis Web untuk UMKM. *Infomatek*, 26(2), 161–174. <https://doi.org/10.23969/infomatek.v26i2.19007>
27. Ahadi, A. H., & Gustina. (2023). Perancangan Sistem Informasi Point of Sales (POS) Berbasis Web (Studi Kasus: Minimarket SMJ Paal 11). *FORTECH (Journal of Information Technology)*, 7(2). <https://doi.org/10.53564/fortech.v7i2.1126>
28. Utami, A. F. D., Ridwan, M., Khakim, A., & Grafvera, E. D. (2023). Pengembangan Sistem Informasi Reservasi Berbasis Website Menggunakan Metode Waterfall. *The Indonesian Journal of Computer Science Research*, 2(2), 1–7. <https://doi.org/10.59095/ijcsr.v2i2.61>
29. Mustapa, M., Amiruddin, E. G., Chaniago, E. M., & Rahmah, U. (2023). Web-Based Student Payment Administration Information System Using the Waterfall Method. *Ceddi Journal of Education*, 2(2), 11–23. <https://doi.org/10.56134/cje.v2i2.52>