



Department of Digital Business

Journal of Artificial Intelligence and Digital Business (RIGGS)

Homepage: <https://journal.ilmudata.co.id/index.php/RIGGS>

Vol. 5 No. 2 (2026) pp: 12695-12705

P-ISSN: 2963-9298, e-ISSN: 2963-914X

Implementasi Sistem Invoice dan Rekapitulasi Bulanan Metode Agile di Greyclean Cireundeu

Danang Wijanarko¹, Husnul Khotimah², Kayla Nabilla³, Joko Suwarno⁴

¹Ilmu Komputer, Teknik Informatika, Universitas Pamulang

¹hiimdanangwjianarko09@gmail.com, ²husnulkh498@gmail.com, ³kaylanabillaaaa@gmail.com, ⁴dosen02522@unpam.ac.id

Abstrak

Greyclean Cireundeu menghadapi kendala operasional dalam pencatatan transaksi, pembuatan invoice, dan penyusunan rekapitulasi transaksi bulanan manual yang memicu risiko kesalahan input, kehilangan data, serta keterlambatan pelaporan. Penelitian ini bertujuan membangun sistem invoice dan rekapitulasi transaksi bulanan berbasis website guna meningkatkan akurasi dan efisiensi manajemen data internal. Metode pengembangan menggunakan metode Agile yang bersifat iteratif dan fleksibel terhadap kebutuhan pengguna, dengan teknik pengumpulan data berupa observasi dan wawancara bersama pemilik usaha. Sistem dikembangkan menggunakan bahasa pemrograman PHP dengan framework Laravel dan database MySQL. Fitur utama yang diimplementasikan meliputi manajemen data pelanggan, pencatatan transaksi, pengelolaan invoice digital, kelola layanan, dan statistik laporan bulanan. Sistem ini dirancang untuk membantu proses administrasi agar seluruh data transaksi tersimpan dalam satu basis yang terintegrasi sehingga lebih mudah dikelola dan diakses ketika diperlukan. Dengan adanya sistem berbasis website, proses pembuatan invoice tidak lagi secara manual, melainkan dapat dibuat secara lebih cepat dan konsisten sesuai data transaksi yang telah tersimpan. Selain itu, rekapitulasi transaksi bulanan dapat disusun secara otomatis sehingga mengurangi pekerjaan yang berulang dan meminimalkan kemungkinan terjadinya kesalahan saat penyusunan laporan. Tampilan sistem juga dibuat sederhana agar mudah dipahami dan digunakan oleh pemilik usaha maupun pihak yang bertanggung jawab dalam pengelolaan administrasi. Hasil penelitian menunjukkan bahwa implementasi sistem invoice dan rekapitulasi transaksi bulanan metode Agile ini berhasil meminimalkan kesalahan pencatatan, mempercepat pencarian data secara real-time, dan mengotomatisasi penyusunan laporan keuangan bulanan secara akurat bagi pemilik usaha.

Kata Kunci: Sistem Invoice, Rekapitulasi Transaksi Bulanan, Metode Agile

1. Latar Belakang

Kemajuan pesat dalam teknologi informasi mendorong berbagai bidang ekonomi untuk mengimplementasikan sistem digital demi mengoptimalkan operasional bisnis, khususnya dalam hal manajemen transaksi dan pelaporan keuangan. Pada sektor usaha jasa UMKM seperti Greyclean Cireundeu (CV. Tartila Group) yang bergerak di bidang perawatan sepatu, akurasi pencatatan data transaksi menjadi pilar utama dalam menjaga keberlangsungan usaha. Berdasarkan observasi langsung, operasional Greyclean Cireundeu saat ini telah menggunakan aplikasi kasir komersial, namun sistem tersebut belum mampu memenuhi seluruh kebutuhan spesifik pengelolaan internal mereka secara optimal. Pemilik usaha masih perlu diperiksa lagi data transaksi satu per satu untuk memastikan kesesuaian. Kendala utama yang dihadapi meliputi proses pembuatan laporan rekapitulasi transaksi bulanan yang masih bersifat manual dan memakan waktu lama, tidak efisiennya pencarian riwayat transaksi pelanggan secara cepat, serta tingginya risiko miskomunikasi atau kesalahan pencatatan data ketika operasional toko dialihkan dari pemilik ke karyawan[1].

Beberapa penelitian yang telah dilakukan sebelumnya mengutamakan digitalisasi dokumen tagihan. Pembuatan sistem manajemen penjualan berbasis web yang saling terhubung telah terbukti meningkatkan efisiensi dalam proses pengolahan faktur. [2]. Selain itu, perancangan sistem berbasis website lainnya juga diterapkan untuk mengoptimalkan pengelolaan data sekaligus meminimalkan risiko kehilangan data operasional [3]. Di sisi lain, implementasi kerangka kerja untuk manajemen persyaratan dalam metode Agile terbukti memberikan efektivitas yang tinggi terhadap fleksibilitas kebutuhan perangkat lunak [4]. Meskipun sistem informasi invoice dan rekapitulasi telah banyak diteliti, sebagian besar solusi yang ditawarkan bersifat umum atau menggunakan platform desktop lokal yang membatasi fleksibilitas akses data[5].

Gap analysis (analisis kesenjangan) dalam penelitian ini terletak pada keterbatasan fungsionalitas sistem kasir yang ada di lapangan, di mana sistem komersial standar tidak menyediakan kustomisasi rekapitulasi performa bulanan dan monitoring status kategori multi-layanan perawatan secara terintegrasi dengan hak akses *multi-user* (*owner* dan *staff*). Kebaruan (*novelty*) yang ditawarkan dalam naskah ini adalah penerapan siklus *sprint* pada metode Agile untuk membangun sistem invoice dan rekapitulasi bulanan berbasis website yang secara spesifik diadaptasikan dengan model operasional internal toko laundry sepatu fisik. Sistem ini dibuat dengan memanfaatkan Bahasa pemrograman PHP serta framework Laravel untuk menciptakan struktur data internal yang terpusat, aman, adaptif, dan mudah disesuaikan.[6].

Alasan pokok pelaksanaan penelitian ini adalah untuk mengatasi masalah administrasi di Greyclean Cireundeu dan menghindari kerugian yang mungkin timbul akibat kesalahan dalam pembuatan ringkasan atau kehilangan informasi. Dengan demikian, tujuan dari penelitian ini adalah untuk merancang, mengembangkan, dan menerapkan sistem informasi berbasis web yang menangani faktur serta ringkasan transaksi bulanan dengan menggunakan metodologi agile. Diharapkan sistem ini dapat mengotomatisasi proses pembuatan laporan keuangan bulanan.[7].

2. Metode Penelitian

Penelitian ini menggunakan metode agile dengan kerangka kerja yang berfokus pada kerjasama yang erat, penyesuaian yang cepat, dan pengembangan sistem secara bertahap[8]. Proses pengumpulan data primer dilakukan selama periode Maret hingga Juni 2026 di Greyclean Cireundeu. Teknik pengumpulan data menggunakan metode observasi terstruktur terhadap alur pencatatan nota fisik dan metode wawancara mendalam bersama pemilik usaha (*owner*) serta staf operasional untuk memetakan seluruh persyaratan fungsional sistem.

Pembangunan sistem dilakukan dengan memanfaatkan metode pengembangan perangkat lunak yang bersifat adaptif, di mana proses pengembangan dibagi ke dalam sejumlah sprint sesuai dengan urutan kebutuhan pengguna. Setiap sprint dirancang untuk menciptakan fitur yang dapat diuji dan dievaluasi secara bertahap, sehingga mendukung pengembangan sistem selanjutnya. Tahapan yang dilakukan dalam setiap sprint terdiri dari Sprint Planning, Sprint Execution, Sprint Review, Sprint Retrospective. Pada tahap Sprint Planning dilakukan identifikasi kebutuhan sistem serta penyusunan *product backlog* berdasarkan hasil observasi dan wawancara. Selanjutnya, Sprint Execution berfokus pada proses pengembangan fitur sesuai daftar pekerjaan yang telah ditetapkan. Hasil pengembangan kemudian dievaluasi pada tahap Sprint Review bersama pengguna untuk memastikan kesesuaiannya dengan kebutuhan. Selanjutnya, diadakan tinjauan kembali sprint untuk menemukan berbagai kendala yang dihadapi selama proses pengembangan serta untuk merumuskan langkah-langkah untuk sprint yang akan datang.

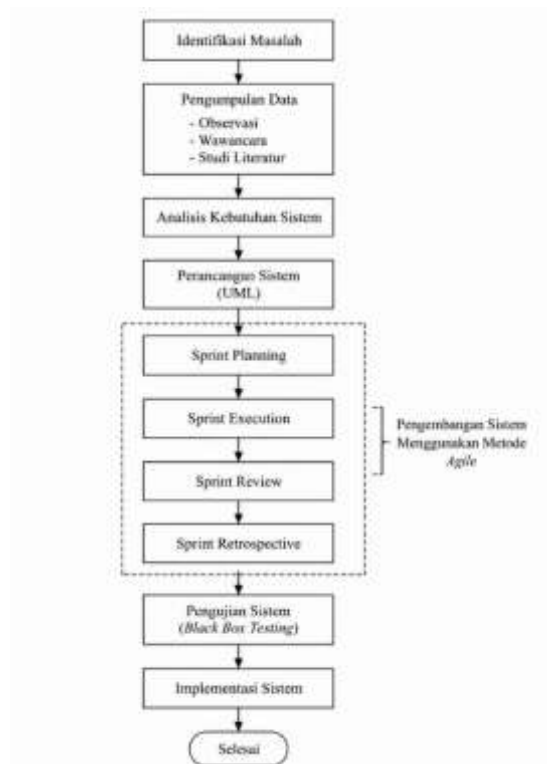
Selama masa sprint, sistem dirancang dan dibuat sesuai dengan spesifikasi yang telah ditentukan sebelumnya. Perancangan sistem dilakukan dengan menggunakan UML dan meliputi diagram penggunaan, diagram aktivitas, diagram urutan, serta diagram kelas untuk menggambarkan proses dan interaksi di antara elemen-elemen dalam sistem. Hasil perancangan tersebut kemudian diterapkan ke dalam bentuk program yang dikembangkan secara terstruktur. Selanjutnya, Setiap elemen yang sudah selesai dibuat akan melalui tahap penilaian di akhir periode sprint, termasuk pengujian *black-box* untuk memastikan semuanya berfungsi dengan baik. Pengguna juga terlibat dalam pengujian sebagai bagian dari pengujian penerimaan pengguna (UAT) untuk mengevaluasi apakah sistem memenuhi syarat operasional yang ditetapkan. Setelah itu dilakukan Sprint Retrospective sebagai tahap evaluasi untuk mengidentifikasi kekurangan dan hambatan yang ditemukan selama proses pengembangan, sehingga dapat menjadi bahan perbaikan pada sprint berikutnya.

Arsitektur basis data pada sistem ini dibangun sebagai sistem yang menyatu dengan memanfaatkan platform MySQL. Data disimpan dalam beberapa tabel utama yang berkaitan satu sama lain, termasuk tabel 'user' yang berfungsi untuk menyimpan informasi akun, tabel 'customers' untuk data mengenai pelanggan, tabel 'services' untuk harga serta jenis layanan yang ditawarkan, tabel 'invoices' untuk informasi transaksi utama, tabel 'invoice_items' untuk rincian item layanan dalam setiap transaksi, dan tabel 'payments' untuk catatan konfirmasi pembayaran. Secara teknis, pengaturan pengembangan sistem dilakukan di lingkungan lokal dengan menggunakan Laragon sebagai server web yang berfungsi secara lokal. Sistem ini dikembangkan menggunakan bahasa pemrograman PHP dan framework Laravel, yang memanfaatkan pola Model-View-Controller (MVC) untuk menyederhanakan replikasi, pengujian kode, dan pemeliharaan sistem di masa mendatang.

2.1. Tahapan Penelitian

Berdasarkan informasi yang didapat, kriteria serta rencana sistem ditelaah dengan menggunakan Unified Modeling Language (UML) sebagai metode pemodelan sistem untuk menggambarkan kebutuhan dan proses bisnis

secara visual[9]. Prosedur pengembangan sistem dilakukan dengan menggunakan pendekatan Agile, yang mencakup perencanaan sprint, pelaksanaan sprint, evaluasi sprint, dan tinjauan kembali sprint[8]. Dalam tahap terakhir dari studi ini, sistem akan diterapkan dan pengujian akan dilaksanakan dengan metode black-box untuk memastikan bahwa seluruh fungsi sistem sesuai dengan kebutuhan pengguna[10].



Gambar 1. Tahapan Penelitian

2.2. Teknik Pengumpulan Data

Pengumpulan informasi bertujuan untuk mendapatkan data yang diperlukan guna mendukung proses pengembangan sistem. Dalam penelitian ini, berbagai metode seperti pengamatan, interview, dan studi literatur diterapkan.[11].

Observasi dilakukan secara langsung di Greyclean Cireundeu, dengan memeriksa pelayanan kepada pelanggan, pencatatan transaksi, pengeluaran faktur, serta penyusunan laporan bulanan. Temuan menunjukkan bahwa prosedur administrasi masih dikerjakan secara manual, yang berpotensi menimbulkan kesalahan dalam penginputan data dan hilangnya informasi.

Wawancara diadakan dengan pemilik usaha guna mengumpulkan data mengenai kebutuhan sistem, kendala operasional, dan fungsi-fungsi penting dari sistem yang akan dibangun. Data yang dikumpulkan menjadi acuan untuk menyusun persyaratan fungsional sistem.

Tinjauan pustaka dilakukan dengan menganalisis sejumlah jurnal akademis, buku, serta studi sebelumnya yang berkaitan dengan tema sistem informasi, pembuatan faktur, ringkasan transaksi, Laravel, dan pendekatan Agile, yang berperan sebagai landasan teori untuk penelitian ini.

Tabel 1. Teknik Pengumpulan Data

Teknik	Tujuan
Observasi	Mengidentifikasi proses bisnis yang berjalan
Wawancara	Menggalai kebutuhan pengguna
Studi Literatur	Mendukung teori dan metode penelitian

2.3 Metode Pengembangan Sistem

Metode yang diterapkan dalam studi ini untuk membangun sistem adalah pemanfaatan perangkat lunak dengan metode agile. Metode agile dipilih karena mendukung pengembangan secara bertahap dan memberikan kesempatan untuk mengadaptasi sistem sesuai dengan kebutuhan pengguna selama proses pengembangan[8]

Fase-fase dalam metode Agile yang diterapkan dalam penelitian ini mencakup perencanaan sprint, pelaksanaan sprint, tinjauan sprint, dan retrospektif sprint.

Dalam tahap perencanaan, spesifikasi sistem ditetapkan, dan daftar produk dibuat untuk dikerjakan di setiap sprint.

Tahap pelaksanaan berlandaskan pada daftar yang telah dibuat sebelumnya.

Pada tahap evaluasi, dipastikan, bersama pengguna, bahwa fitur yang dikembangkan sesuai dengan kebutuhan mereka.

Tahap evaluasi juga bertujuan untuk mengenali kendala dan memperoleh masukan untuk perbaikan di sprint mendatang.

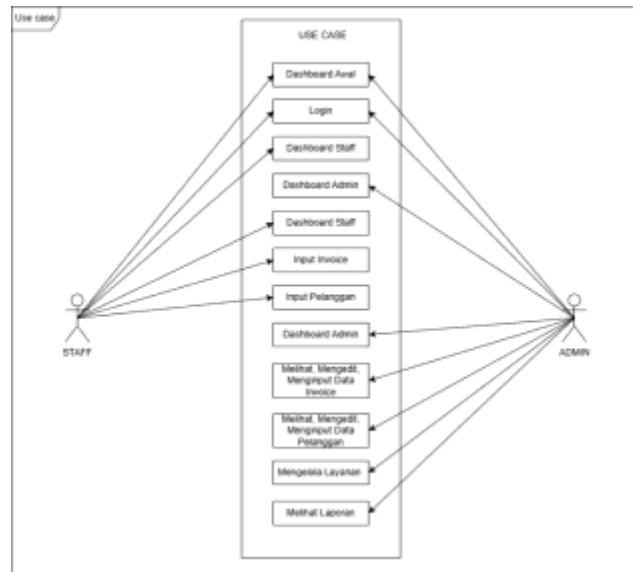
Tabel 2. Pembagian Sprint Pengembangan

Sprint	Fitur yang Dikembangkan
Sprint 1	Login, Dashboard, dan Kelola Pelanggan
Sprint 2	Input Invoice dan Kelola Layanan
Sprint 3	Pembayaran dan Penyimpanan Invoice
Sprint 4	Rekapitulasi Bulanan dan Laporan
Sprint 5	Pengujian dan Penyempurnaan Sistem

2.4 Perancangan Sistem

Proses pengembangan sistem ditangani dengan memanfaatkan Unified Modeling Language (UML) untuk menggambarkan secara visual dan terstruktur kebutuhan bisnis serta proses sistem[9]. Diagram yang dipakai mencakup Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram.

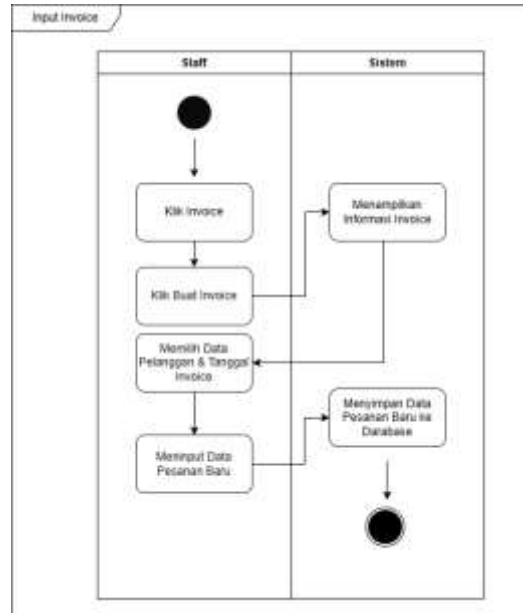
Use Case Diagram berfungsi untuk menjelaskan interaksi antar pengguna dengan sistem serta kebutuhan fungsional aplikasi[12].



Gambar 2. Use Case

Karyawan memiliki akses yang lebih sedikit dan bisa berkonsentrasi pada pekerjaan harian mereka. Setelah login berhasil, mereka akan dibawa ke halaman utama karyawan, di mana mereka dapat menjalankan aktivitas utama seperti memasukkan faktur dan informasi pelanggan. Di sisi lain, aktor Admin memegang kendali penuh atas sistem dengan hak akses yang jauh lebih luas. Selain masuk ke Dashboard Admin, seorang Admin tidak hanya bisa menginput data, melainkan juga memiliki wewenang untuk melihat dan mengedit Data Invoice serta Data Pelanggan. Admin juga memegang fungsi manajerial eksklusif yang tidak bisa diakses oleh Staff, yaitu fitur untuk Mengelola Layanan dan Melihat Laporan sistem secara keseluruhan.

Activity Diagram digunakan untuk menunjukkan rangkaian tindakan dalam sebuah sistem.

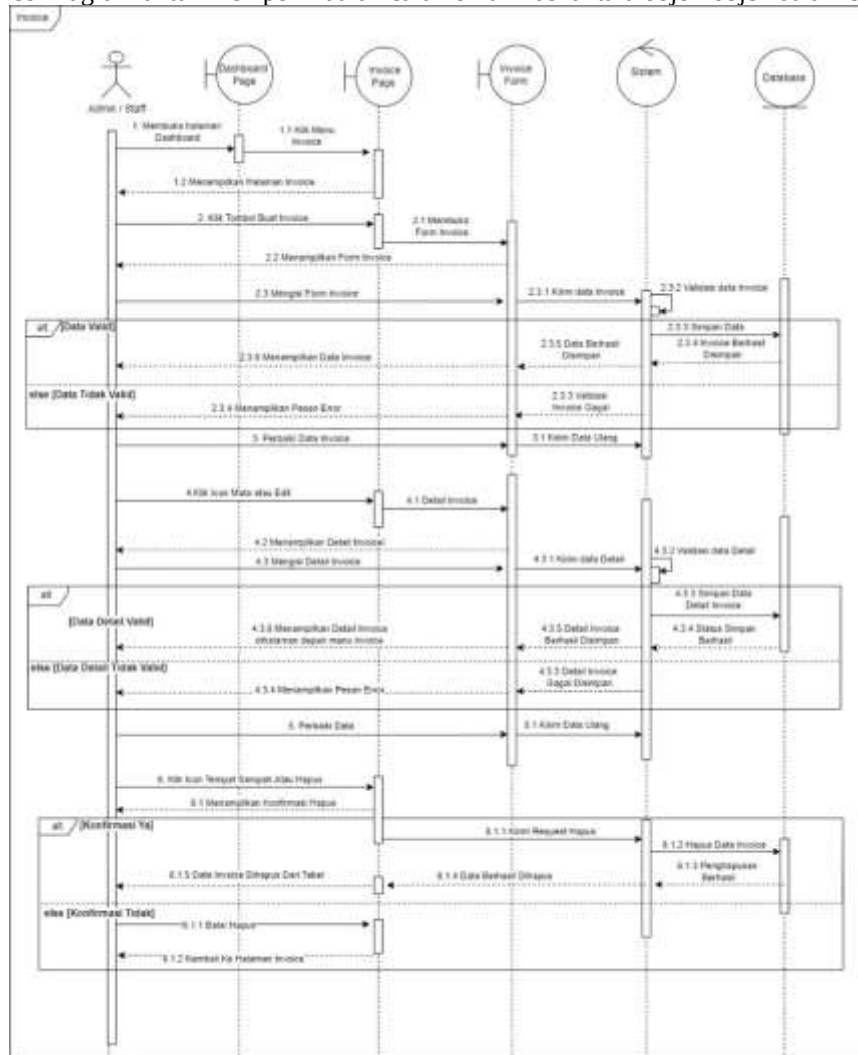


Gambar 3. Activity Diagram

Activity Diagram ini menunjukkan langkah-langkah teratur dalam proses pengisian faktur, yang mencakup kolaborasi langsung antara pegawai dan elemen sistem. Proses ini dimulai saat pegawai membuka menu dengan mengklik pilihan Invoice dan kemudian menekan tombol untuk menghasilkan faktur yang baru. Merespons tindakan tersebut, Sistem kemudian bekerja untuk menampilkan halaman informasi atau formulir invoice baru ke layar pengguna.

Setelah formulir tersebut muncul, alur kembali ke sisi Staff yang bertugas untuk memilih data pelanggan serta menentukan tanggal invoice yang sesuai. Para karyawan kemudian memasukkan detail pesanan baru ke dalam sistem. Setelah semua data dianggap lengkap dan telah dikirimkan oleh pengguna, sistem mengola dan menyimpan data pesanan baru di dalam database, sehingga menyelesaikan seluruh proses penagihan.

Sequence Diagram untuk memperlihatkan cara komunikasi antara objek-objek dalam sebuah sistem.

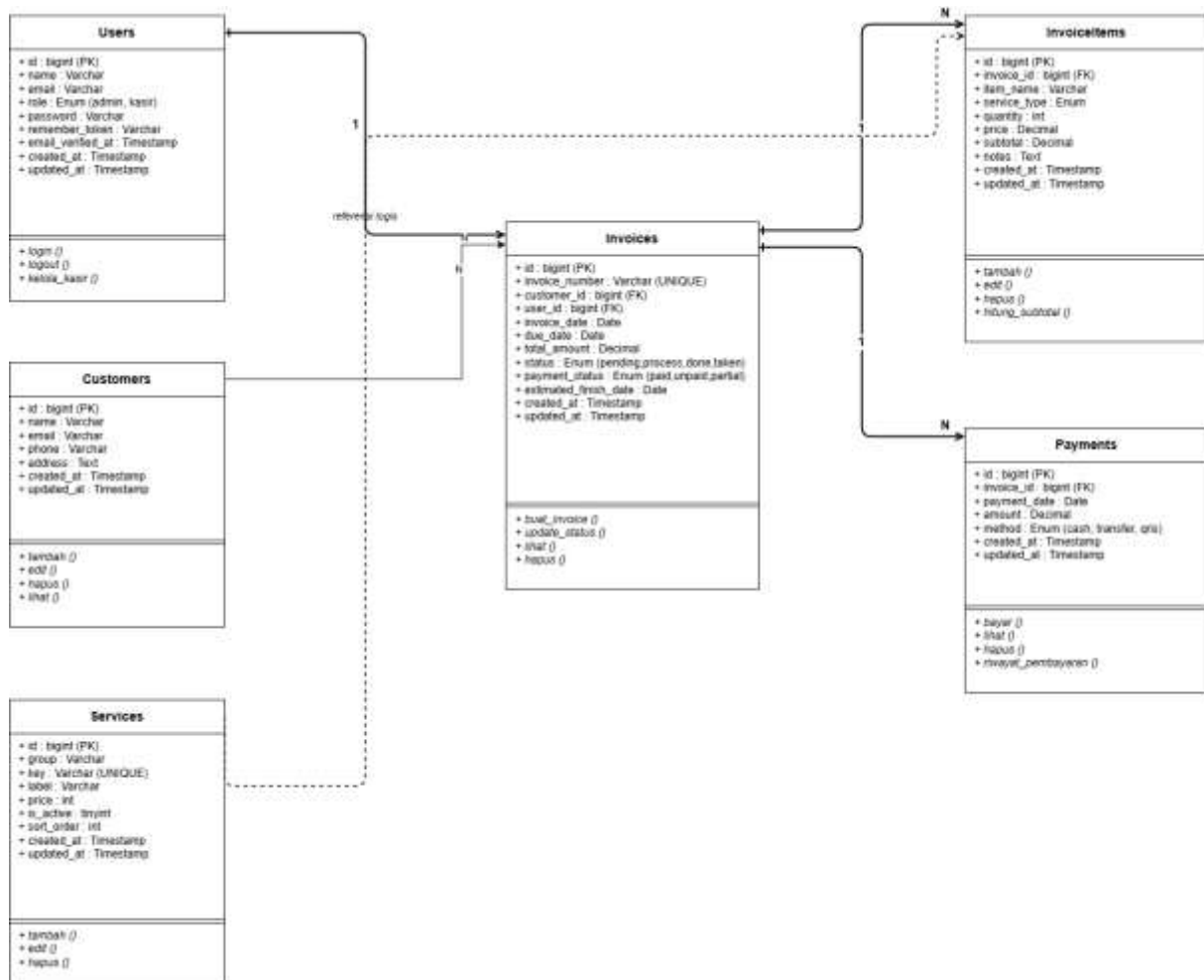


Gambar 4. Sequence Diagram

Diagram ini menggambarkan proses kerja dalam pengelolaan faktur serta menunjukkan hubungan antara pengguna (administrator/karyawan), sistem, dan database saat menangani informasi faktur. Proses kerja ini menjelaskan aktivitas utama: membuat, mengevaluasi, mengubah, dan menghapus informasi faktur.

Proses dimulai ketika pengguna mengakses menu faktur dan mengisi detail melalui formulir yang tersedia. Sistem akan memverifikasi informasi yang telah diinput sebelum menyimpan pembaruan ke dalam database. Jika data tersebut benar, proses akan diteruskan dan informasi akan disimpan atau diperbarui dengan berhasil. Sebaliknya, jika data tidak sesuai, sistem akan menampilkan pesan kesalahan agar pengguna dapat melakukan perbaikan. Skema ini juga menggambarkan langkah-langkah untuk menghapus data faktur, yang dimulai dengan konfirmasi dari pengguna sebelum sistem menghapus informasi dari database.

Sedangkan Class Diagram digunakan untuk menunjukkan susunan data serta keterkaitan antara entitas dalam sistem[13].



Gambar 5. Class Diagram

Class Diagram yang merancang struktur kode dan relasi antarkomponen objek dalam sistem manajemen invoice dan transaksi. Di dalam diagram ini terdapat enam kelas utama yang saling terhubung, yaitu Users, Customers, Invoices, InvoiceItems, Payments, dan Services, di mana masing-masing kelas telah dilengkapi dengan atribut data (seperti tipe data *Primary Key* dan *Foreign Key*) serta fungsi/method operasional untuk menjalankan fungsi CRUD (*Create, Read, Update, Delete*).

Hubungan antar-objek berpusat pada kelas Invoices sebagai dokumen transaksi utama. Satu pengguna (Users) dapat mengelola banyak invoice (relasi *1 to N* melalui *user_id*), dan satu pelanggan (Customers) juga dapat memiliki banyak data invoice (relasi *1 to N* melalui *customer_id*). Untuk detail isi dari transaksi itu sendiri, satu data invoice dapat dipecah menjadi banyak item pesanan pada kelas InvoiceItems (relasi *1 to N* melalui *invoice_id*), di mana setiap item tersebut memiliki referensi logis ke jenis layanan yang tersedia pada kelas Services. Terakhir, untuk mencatat aspek finansial, satu dokumen invoice dapat menerima beberapa kali riwayat pembayaran yang dicatat pada kelas Payments (relasi *1 to N* melalui *invoice_id*), guna mengakomodasi status pembayaran yang bersifat cicilan (*partial*) maupun lunas (*paid*).

2.5 Implementasi dan Pengujian Sistem

Sistem ini dirancang dengan menggunakan framework Laravel, yang memanfaatkan bahasa pemrograman PHP dan basis data MySQL. Laravel merupakan framework yang menerapkan model Model-View-Controller (MVC), yang mendukung pengembangan aplikasi web yang tertata rapi dan mudah untuk dikelola. [14]. Sistem yang dibangun memiliki fitur utama berupa pengelolaan pelanggan, pengelolaan invoice, pencatatan pembayaran, serta rekapitulasi transaksi bulanan.

Alur penggunaan sistem dimulai dari penginputan data pelanggan oleh kasir, dilanjutkan dengan penginputan data invoice dan pembayaran. Data invoice yang telah tersimpan akan digunakan sebagai dasar dalam penyusunan laporan rekapitulasi bulanan[15].

Sistem diujicobakan dengan teknik black box testing. Pendekatan ini diterapkan untuk memastikan bahwa sistem beroperasi dengan baik tanpa melakukan analisis terhadap struktur kode pemrograman.[10]. Fungsi masuk, pengelolaan pelanggan, pembuatan faktur, pembayaran, dan pelaporan diuji agar semua fungsi tersebut dapat memenuhi kebutuhan pengguna.

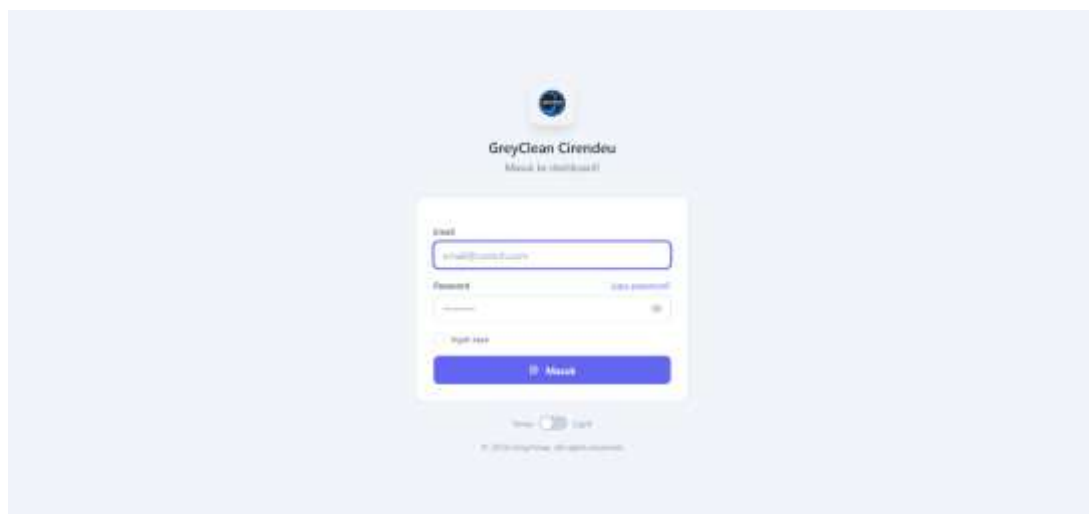
Tabel 3. Skenario Pengujian Black Box

Fitur	Skenario Pengujian	Hasil yang Diharapkan
Login	Input username dan password valid	Berhasil masuk ke sistem
Pelanggan	Menambah data pelanggan	Data tersimpan
Invoice	Menambah data invoice	Data tersimpan
Pembayaran	Menambah data pembayaran	Data tersimpan
Laporan	Menampilkan rekap bulanan	Data berhasil ditampilkan

3. Hasil dan Diskusi

3.1 Implementasi Halaman Login

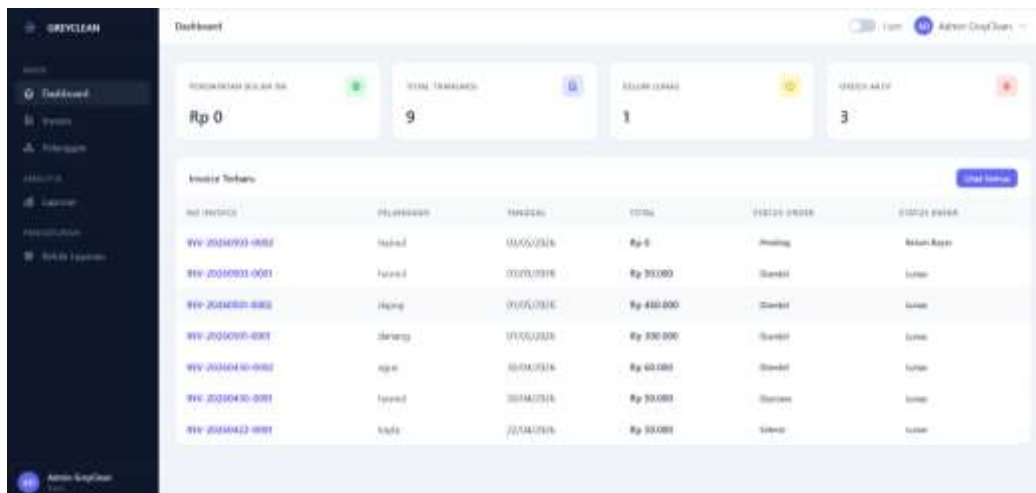
Halaman masuk merupakan gerbang utama untuk sistem pembayaran dan ringkasan bulanan. Pada halaman ini, pengguna perlu memasukkan nama pengguna dan kata sandi mereka. Setelah proses verifikasi selesai, mereka akan diarahkan ke halaman dasbor yang relevan.



Gambar 6. Halaman Login

3.2 Implementasi Dashboard

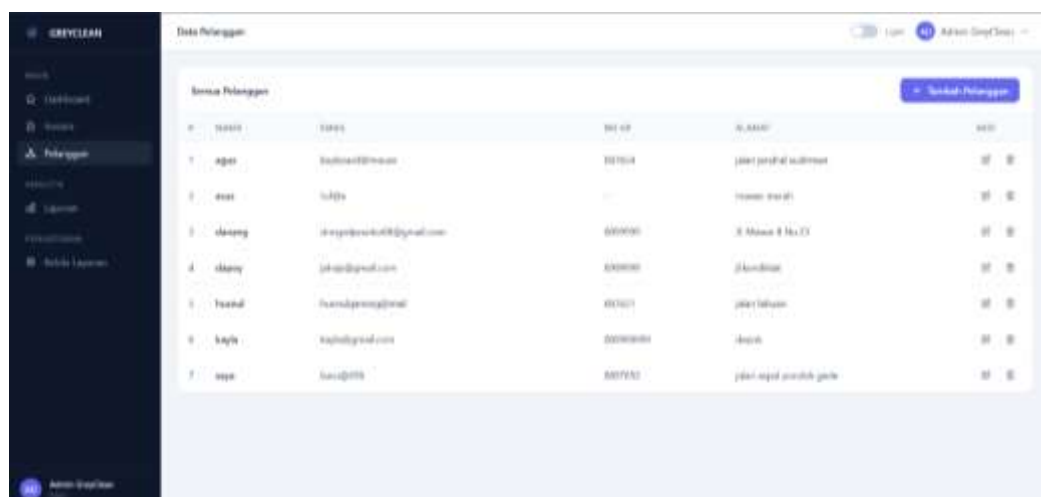
Dashboard ini menyajikan ringkasan aktivitas sistem, termasuk jumlah pelanggan, tagihan, pembayaran, dan data lain yang penting bagi pengguna. Halaman ini bertindak sebagai pusat utama untuk seluruh fungsi sistem.



Gambar 7. Halaman Dashboard

3.3 Implementasi Data Pelanggan

Halaman data pelanggan berfungsi untuk mengatur data pelanggan yang bertransaksi dengan Greyclean Laundry. Pengguna memiliki kemampuan untuk menyisipkan, memodifikasi, memeriksa, dan menghapus informasi pelanggan sesuai dengan kebutuhan operasional.

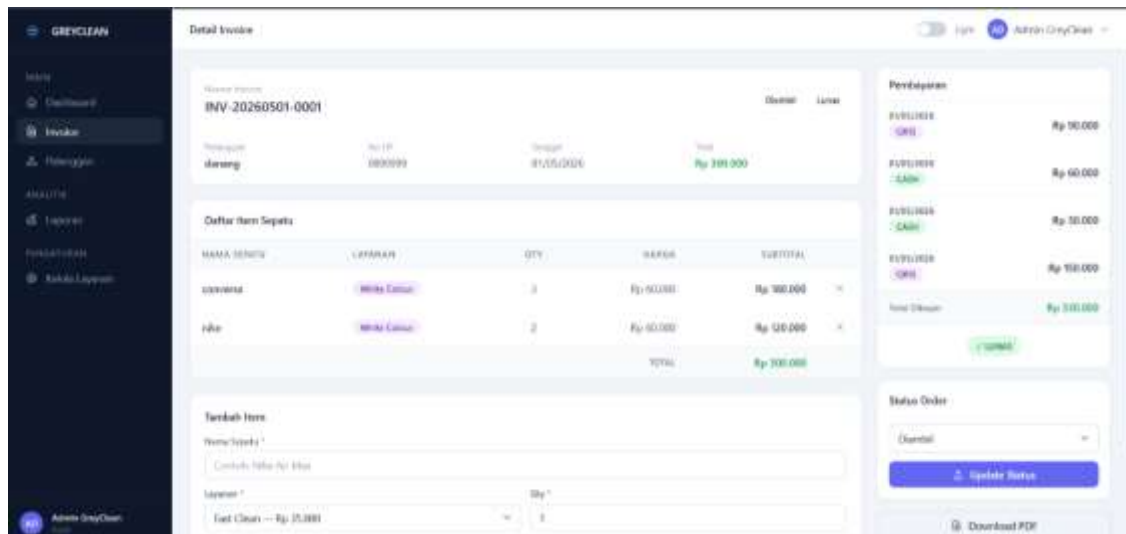


Gambar 8. Halaman Data Pelanggan

3.4 Implementasi Invoice dan Pembayaran

Halaman invoice dan pembayaran digunakan untuk mengelola transaksi layanan yang dilakukan pelanggan. Pengguna dapat membuat invoice baru dengan memilih data pelanggan dan menginput detail layanan yang digunakan. Setelah invoice dibuat, sistem juga menyediakan fitur pencatatan pembayaran yang terhubung

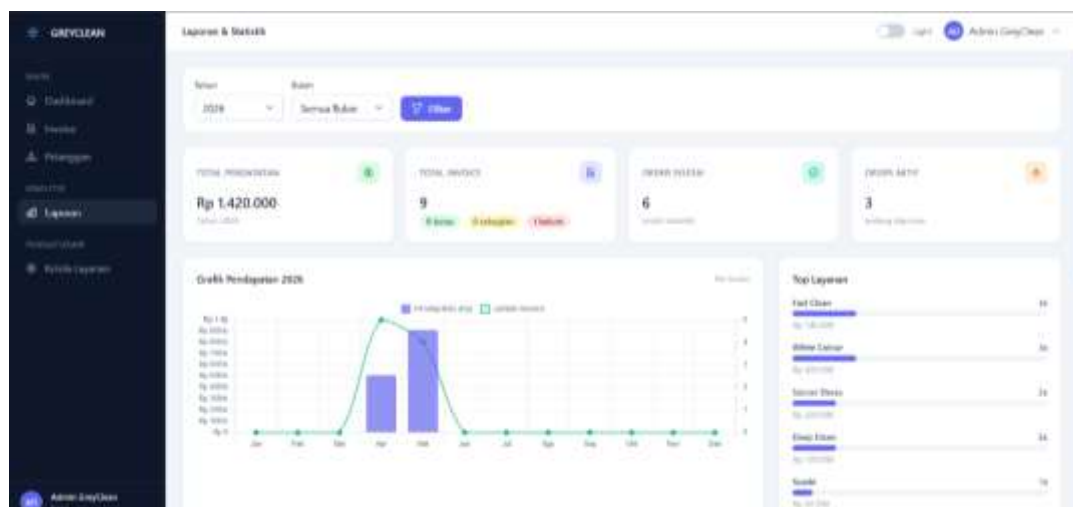
langsung dengan data invoice sehingga status pembayaran dapat dipantau secara terstruktur. Seluruh data transaksi yang tersimpan akan digunakan sebagai dasar dalam penyusunan laporan rekapitulasi bulanan.



Gambar 9. Halaman Invoice dan Pembayaran

3.5 Implementasi Laporan Rekapitulasi Bulanan

Halaman laporan menunjukkan ikhtisar dari transaksi yang tersimpan di dalam sistem. Data yang ditampilkan bisa dimanfaatkan untuk penilaian dan pengawasan aktivitas operasional Greyclean Laundry secara bulanan..



Gambar 10. Laporan Rekapitulasi Bulanan

3.6 Hasil Pengujian Black Box

Pengujian sistem dilaksanakan dengan metode pengujian kotak hitam untuk memastikan bahwa semua fitur aiatem sesuai dengan pengguna. Uji coba ini menitikberatkan pada fungsi registrasi, pengguna, pembuatan tagihan, transaksi pembayaran, dan pelaporan.

Tabel 4. Pengujian Black Box

No	Fitur yang Diuji	Skenario	Hasil yang Diharapkan	Status
1	Login	Memasukan data valid	Berhasil masuk sistem	Berhasil
2	Pelanggan	Menambah data Pelanggan	Data tersimpan	Berhasil
3	Invoice	Menambah invoice baru	Data tersimpan	Berhasil
4	Pembayaran	Menambah Pembayaran	Data tersimpan	Berhasil
5	Laporan	Menampilkan laporan bulanan	Data tampil	Berhasil

4. Kesimpulan

Proyek riset ini telah berhasil merancang dan menerapkan sistem informasi berbasis web untuk merangkum faktur serta transaksi bulanan di Greyclean Cireundeu, dengan memanfaatkan pendekatan pengembangan perangkat lunak yang agile. Sistem ini telah diuji dengan metode pengujian black box untuk memastikan bahwa seluruh fitur sistem sesuai dengan kebutuhan pengguna. Uji coba ini menitikberatkan pada fungsi registrasi, pengguna, pembuatan tagihan, transaksi pembayaran, dan pelaporan. Penerapan sistem ini memberikan kontribusi terhadap manajemen transaksi yang lebih terorganisir, mengurangi kemungkinan kesalahan dalam entri data, serta mempermudah proses pembuatan laporan bulanan di Greyclean Cireundeu.

Referensi

- [1] Y. Harjoseputro and Thomas Adi Purnomo Sidhi, "Pemanfaatan Sistem Informasi Pada Usaha Kecil Menengah Untuk Pencatatan dan Pelaporan Transaksi Penjualan," *Din. J. Pengabd. Kpd. Masy.*, vol. 5, no. 5, pp. 1305–1317, 2021. <https://doi.org/10.31849/dinamisia.v5i5.4209>
- [2] C. D. R. Pakpahan, "Pengembangan Sistem Administrasi Pengelolaan Penjualan Dan Produksi Secara Terintegrasi Berbasis Website," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 4, pp. 3279–3292, 2022. <https://doi.org/10.35957/jatisi.v9i4.2837>
- [3] A. A. A. K. Annas Khasbulloh, "Didik Baru Berbasis Web Menggunakan Framework Laravel," vol. 8, no. 1, 2023. <https://doi.org/10.51876/simtek.v8i1.165>
- [4] K. Asad, M. F. Farooqui, and M. Muqem, "An AHP-Based Framework for Effective Requirement Management in Agile Software Development (ASD)," *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 11, no. 10, 2023. <https://doi.org/10.17762/ijritcc.v11i10.8509>
- [5] K. Divisi and P. Pt, "Jurnal JTik (Jurnal Teknologi Informasi dan Komunikasi) Perancangan Sistem Informasi Perencanaan Produksi (Studi)," vol. 9, no. September, pp. 1070–1080, 2025. <https://doi.org/10.35870/jtik.v9i3.3820>
- [6] L. A. T. Nguyen, T. S. Huynh, D. T. Tran, and Q. H. Vu, "Design and Implementation of Web Application Based on MVC Laravel Architecture," *Eur. J. Electr. Eng. Comput. Sci.*, vol. 6, no. 4, pp. 23–29, 2022. <https://doi.org/10.24018/ejece.2022.6.4.448>
- [7] R. L. Z. F. Al Laiti, "Making Web-Based Product And Inventory Applications Using The Laravel Framework (Case Study : Cv. Global Best Ls)," *J. Sains dan Inform.*, vol. 8, no. 2, pp. 144–150, 2022. <https://doi.org/10.22216/jsi.v8i2.1451>
- [8] W. M. Kansha, Saherih, and Muchlis, "Analisis Perbandingan Framework Codeigniter Dan Laravel Dalam Pengembangan Web Application," *J. Tek. Inform. STMIK Antar Bangsa*, vol. 9, no. 1, pp. 27–33, 2023. <https://doi.org/10.51998/jti.v9i1.511>
- [9] B. Alturas, "Connection between UML use case diagrams and UML class diagrams: a matrix proposal," *Int. J. Comput. Appl. Technol.*, vol. 72, no. 3, pp. 161–168, 2023. <https://doi.org/10.1504/IJCAT.2023.133294>
- [10] A. C. Praniffa, A. Syahri, F. Sandes, U. Fariha, and Q. A. Giansyah, "Pengujian Sistem Informasi Parkir Berbasis Web Pada UIN SUSKA RIAU Menggunakan White Box dan Black Box Testing," *J. Test. dan Implementasi Sist. Inf.*, vol. 1, no. 1, pp. 1–16, 2023. <https://doi.org/10.55583/jtisi.v1i1.321>
- [11] M. Ahmadar, P. Perwito, and C. Taufik, "PERANCANGAN SISTEM INFORMASI PENJUALAN BERBASIS WEB PADA RAHAYU PHOTO COPY DENGAN DATABASE MySQL," *Dharmakarya*, vol. 10, no. 4, p. 284, 2021. <https://doi.org/10.24198/dharmakarya.v10i4.35873>
- [12] T. A. Rospricilia and M. N. P. Ma'ady, "Pemodelan Integration Use Case (IUC): Perancangan Use Case Diagram (UML) untuk Sistem-sistem yang Terintegrasi," *INTEGER J. Inf. Technol.*, vol. 9, no. 2, pp. 165–172, 2024. <https://doi.org/10.31284/j.integer.2024.v9i2.6345>
- [13] B. J. M. Putra, A. Fu'adi, and D. A. F. Yuniarti, "Analisa dan Rancangan Sistem Informasi Pariwisata Pacitan dengan UML dan ERD," *Inf. Syst. Educ. Prof. J. Inf. Syst.*, vol. 7, no. 1, p. 63, 2022. <https://doi.org/10.51211/isbi.v7i1.1920>
- [14] R. A. A. Pratama, M. Mahmud, Y. Aprizal, M. J. Syaftriandi, E. Setiawan, and N. E. Rieni, "Penerapan Metode Black Box dalam Pengujian Aplikasi Informasi Stok Barang pada PT. Trimega Jaya Medika Berbasis Web," *J. Teknol. Inform. dan Komput.*, vol. 9, no. 1, pp. 174–183, 2023. <https://doi.org/10.37012/jtik.v9i1.1411>
- [15] Zulrahmadi, Muhamad Amin, and Khairul Ihwan, "SISTEM INFORMASI PENJUALAN PAKET BERBASIS WEB (Studi Kasus : ChariNET)," *Juti Unisi*, vol. 6, no. 2, pp. 24–29, 2022. <https://doi.org/10.32520/juti.v6i2.2458>