

RANCANG BANGUN APLIKASI PARKIR OTOMATIS DENGAN PENDEKATAN PEMROGRAMAN BERORIENTASI OBJEK

Vandamme Muhammad Rahardjo¹, Apriade Voutama², Danu Firmansyah³, Salma Salsabila⁴, Wilda Aprilia Mustika Sari⁵

^{1,2,3,4,5}Universitas Singaperbangsa Karawang, Jl. H.S. Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat, 081218669229

Keywords:

Parking Management System;
Object-Oriented
Programming;
Waterfall Methodology;
Automated Parking System.

Correspondent Email:

xxxxxxxxx@xxxx.xxx



JITET is licensed under
a Creative Commons
Attribution-NonCommercial
4.0 International License

Abstrak. Meningkatnya jumlah kendaraan dan ketidakefisienan sistem parkir manual mendorong kebutuhan akan sistem manajemen parkir otomatis. Penelitian ini bertujuan untuk mengembangkan aplikasi manajemen parkir berbasis pemrograman berorientasi objek guna meningkatkan efisiensi dan akurasi pencatatan kendaraan serta transaksi parkir. Penelitian ini menggunakan metode Waterfall yang mencakup analisis kebutuhan, perancangan sistem, implementasi, pengujian, dan pemeliharaan. Sistem ini diimplementasikan menggunakan Java dan MySQL untuk penyimpanan data. Pengujian dilakukan dengan metode Black Box Testing yang menunjukkan bahwa semua fitur berfungsi dengan baik, termasuk pencatatan kendaraan otomatis, perhitungan biaya parkir, dan autentikasi pengguna. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan mampu mengurangi waktu pemrosesan dan meningkatkan keamanan data dibandingkan dengan metode manual. Studi ini berkontribusi dalam meningkatkan pengelolaan fasilitas parkir melalui solusi berbasis teknologi.

Abstract. The increasing number of vehicles and the inefficiency of manual parking systems have led to the need for an automated parking management system. This research aims to develop an object-oriented programming-based parking management application to enhance efficiency and accuracy in vehicle registration and transaction recording. The research adopts the Waterfall methodology, consisting of requirement analysis, system design, implementation, testing, and maintenance. The system was implemented using Java and MySQL for data storage. The testing phase, conducted using Black Box Testing, confirms that all features function correctly, including automatic vehicle registration, parking fee calculation, and user authentication. The findings indicate that the developed system significantly reduces processing time and enhances data security compared to manual methods. This study contributes to improving parking facility management through technology-driven solutions.

1. PENDAHULUAN

Dalam era digital saat ini, teknologi informasi telah menjadi bagian integral dalam berbagai aspek kehidupan, termasuk dalam pengelolaan fasilitas publik seperti area parkir.

Sistem manajemen parkir yang masih menggunakan metode manual sering kali menghadapi berbagai kendala, seperti pencatatan kendaraan yang tidak akurat, antrian panjang akibat proses administrasi yang lambat,

serta risiko kehilangan data akibat human error [1]. Masalah ini semakin kompleks seiring dengan meningkatnya jumlah kendaraan dan keterbatasan lahan parkir yang tersedia. Oleh karena itu, diperlukan sebuah solusi berbasis teknologi yang dapat meningkatkan efisiensi dan akurasi dalam pengelolaan parkir.

Seiring dengan perkembangan ilmu komputer, pendekatan Pemrograman Berorientasi Objek (PBO) menjadi salah satu metode yang efektif dalam pengembangan aplikasi manajemen parkir. Dengan konsep seperti enkapsulasi, pewarisan, dan polimorfisme, pengembangan sistem dapat lebih terstruktur, mudah dipelihara, dan diperluas di masa depan [2]. Implementasi sistem manajemen parkir berbasis PBO memungkinkan pencatatan kendaraan yang masuk dan keluar dilakukan secara otomatis, serta menyediakan fitur autentikasi pengguna yang aman.

Urgensi penelitian ini terletak pada kebutuhan akan sistem parkir yang lebih modern dan terotomatisasi guna mengurangi permasalahan yang ada pada metode konvensional. Dalam beberapa penelitian terdahulu, sistem manajemen parkir berbasis teknologi telah menunjukkan efektivitas dalam meningkatkan efisiensi operasional dan kenyamanan pengguna [3].

Tujuan dari penelitian ini adalah untuk merancang dan mengembangkan aplikasi manajemen parkir berbasis PBO yang memiliki fitur utama seperti sistem login yang aman, pencatatan kendaraan masuk dan keluar secara otomatis, serta penyimpanan data yang terorganisir. Selain itu, penelitian ini juga bertujuan untuk mengevaluasi efektivitas sistem yang dikembangkan dalam meningkatkan efisiensi dan akurasi pengelolaan parkir [4].

Dalam menyusun solusi terhadap permasalahan ini, metode pengembangan perangkat lunak berbasis PBO diterapkan dengan tahapan analisis kebutuhan, perancangan sistem menggunakan diagram UML, implementasi menggunakan bahasa pemrograman Java, serta pengujian sistem guna memastikan bahwa fitur yang dikembangkan

berfungsi sesuai dengan kebutuhan pengguna [5]. Diagram UML yang digunakan dalam penelitian ini meliputi Use Case Diagram, Activity Diagram, dan Sequence Diagram untuk memodelkan skenario pengguna, alur kerja sistem, serta komunikasi antara komponen dalam sistem [6]. NetBeans dipilih sebagai Integrated Development Environment (IDE) untuk mendukung efisiensi pengembangan, sedangkan MySQL digunakan sebagai sistem manajemen basis data guna memastikan penyimpanan data kendaraan yang aman dan terstruktur [7].

Berdasarkan tinjauan pustaka yang ada, penelitian ini mengacu pada berbagai studi tentang sistem manajemen parkir berbasis teknologi, konsep Pemrograman Berorientasi Objek, serta implementasi sistem autentikasi dalam aplikasi berbasis web atau desktop. Pengembangan hipotesis dalam penelitian ini berfokus pada asumsi bahwa penerapan sistem berbasis PBO akan meningkatkan efisiensi, keamanan, dan akurasi dalam pencatatan data parkir dibandingkan dengan metode manual [8]. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan solusi teknologi untuk sistem parkir yang lebih canggih dan efisien. Selain itu, hasil dari penelitian ini dapat menjadi dasar bagi pengelola fasilitas parkir dalam mengadopsi teknologi digital guna meningkatkan kualitas layanan dan operasional mereka.

2. TINJAUAN PUSTAKA

Sistem merupakan kumpulan dari berbagai elemen yang terdapat proses input dan output sehingga saling berkaitan. Sistem yaitu rangkaian dari satu maupun lebih komponen yang berhubungan dan dapat berinteraksi guna mencapai tujuan yang diterapkan. Sistem merupakan jaringan kerja dari berbagai prosedur yang berkaitan guna melakukan suatu aktivitas sesuai dengan tujuan[9].

Metode Waterfall merupakan salah satu metode klasik dalam pengembangan perangkat lunak yang mengadopsi pendekatan sistematis dan terstruktur melalui lima tahapan utama yaitu analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Metode ini cocok digunakan ketika kebutuhan sistem sudah terdefinisi dengan jelas sejak awal

dan jarang mengalami perubahan selama proses pengembangan [5][10].

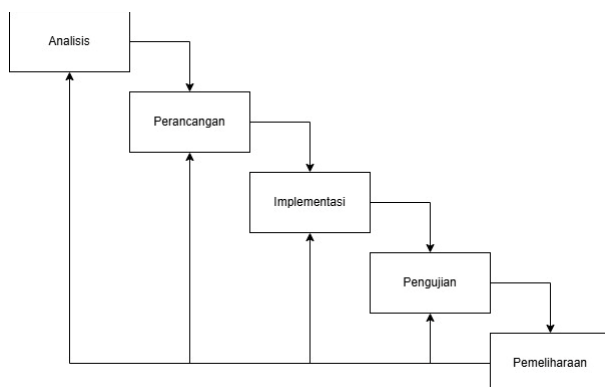
Pemrograman Berorientasi Objek (PBO) digunakan untuk mengembangkan sistem dengan pendekatan modular dan reusable melalui konsep seperti enkapsulasi, pewarisan, dan polimorfisme. Pendekatan ini memudahkan pengembang dalam membangun sistem yang kompleks namun tetap terstruktur dengan baik [2].

Sistem parkir otomatis merupakan solusi teknologi yang dapat menggantikan pencatatan manual untuk meningkatkan efisiensi dan akurasi dalam pengelolaan parkir. Sistem ini biasanya mencakup fitur pencatatan kendaraan masuk dan keluar, perhitungan biaya parkir otomatis, serta penyimpanan data transaksi secara terstruktur menggunakan basis data relasional seperti MySQL [7].

Untuk memastikan fungsionalitas sistem berjalan sesuai spesifikasi, dilakukan pengujian menggunakan metode Black Box Testing, yaitu metode yang menguji input dan output dari sistem tanpa memperhatikan struktur internal kode [8]. Pengujian ini dinilai efektif dalam memastikan fungsi utama sistem berjalan sesuai harapan.

3. METODE PENELITIAN

Penelitian ini menggunakan metode Waterfall, yaitu pendekatan rekayasa perangkat lunak yang dilakukan secara sistematis dan bertahap mulai dari analisis, perancangan, implementasi, pengujian, hingga pemeliharaan sistem [10].



Gambar 1. Metode Waterfall

3.1. Analisis

Tahapan pertama adalah analisis kebutuhan, di mana penelitian ini mengidentifikasi masalah utama dalam sistem parkir manual dan menentukan fitur utama yang harus dimiliki oleh sistem baru. Data dikumpulkan melalui observasi langsung terhadap sistem parkir manual untuk mendapatkan gambaran yang jelas mengenai kendala yang ada [7].

3.2. Perancangan

Setelah analisis kebutuhan selesai, tahapan selanjutnya adalah perancangan sistem. Pada tahap ini, dibuat berbagai diagram UML seperti Use Case Diagram, Activity Diagram, dan Sequence Diagram untuk memodelkan sistem yang akan dikembangkan [2]. Selain itu, struktur database dirancang menggunakan MySQL agar sistem dapat menyimpan data kendaraan masuk dan keluar secara terstruktur [3]. Perancangan ini juga mencakup pembuatan alur kerja sistem dengan skenario penggunaan yang jelas.

3.3. Implementasi

Tahapan implementasi dilakukan dengan mengembangkan aplikasi menggunakan bahasa pemrograman Java pada lingkungan NetBeans IDE, serta memanfaatkan MySQL sebagai sistem manajemen basis data [11]. Fitur-fitur utama seperti autentikasi pengguna, pencatatan kendaraan masuk dan keluar diimplementasikan sesuai dengan rancangan yang telah dibuat sebelumnya.

3.4. Pengujian

Setelah implementasi selesai, sistem diuji untuk memastikan semua fitur berfungsi dengan baik. Pengujian dilakukan dalam beberapa tahap, yaitu pengujian unit untuk memastikan setiap komponen berjalan sesuai dengan spesifikasi, serta pengujian integrasi untuk memastikan seluruh komponen dalam sistem bekerja secara harmonis. Pengujian dilakukan menggunakan metode Black Box Testing, yang berfokus pada pengujian fungsionalitas. Pengujian ini berfungsi untuk mengevaluasi peningkatan efisiensi dan keakuratan dari sistem.[12]

3.5. Pemeliharaan

Tahap terakhir dalam metode Waterfall adalah pemeliharaan sistem. Evaluasi dilakukan berdasarkan hasil pengujian sistem dan analisis performa dalam lingkungan simulasi. Selain itu, optimalisasi sistem terus dilakukan untuk memastikan performa tetap stabil dan handal dalam jangka panjang [4]. Evaluasi dilakukan melalui observasi terhadap sistem parkir manual sebelum penerapan sistem baru serta eksperimen dengan melakukan uji coba terhadap sistem yang telah dikembangkan. Definisi operasional variabel dalam penelitian ini meliputi keamanan sistem yang diukur berdasarkan keberhasilan autentikasi pengguna dalam sistem login, efisiensi sistem yang diukur berdasarkan waktu pemrosesan data kendaraan masuk dan keluar, serta akurasi data yang diukur berdasarkan perbandingan antara data yang tercatat di sistem dan data kendaraan yang sebenarnya.

Dengan menerapkan metode Waterfall, penelitian ini diharapkan dapat menghasilkan sistem manajemen parkir yang lebih aman, efisien, dan dapat diimplementasikan secara luas di berbagai lingkungan parkir.

4. HASIL DAN PEMBAHASAN

Penelitian ini dilakukan menggunakan metode pengembangan perangkat lunak Waterfall, yang terdiri dari lima tahapan utama yaitu analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Berikut adalah hasil dari setiap tahapan berdasarkan metode tersebut.

4.1. Analisis

Tahap analisis dilakukan untuk mengidentifikasi permasalahan dalam sistem parkir manual serta menentukan kebutuhan sistem baru. Berdasarkan hasil observasi ditemukan bahwa sistem parkir manual memiliki beberapa kendala utama, seperti pencatatan kendaraan yang lambat, rentan terhadap human error, dan kurangnya keamanan data transaksi. Oleh karena itu, sistem parkir otomatis dikembangkan dengan fitur utama sistem login untuk memastikan hanya petugas yang berwenang dapat mengakses sistem, pencatatan kendaraan masuk dan keluar secara otomatis, dan penyimpanan data secara terstruktur dalam database.

Tabel 1. Hasil observasi parkir manual

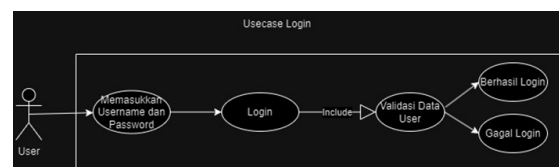
Parameter	Hasil Observasi
Pencatatan kendaraan	Ditulis di kertas, rawan kesalahan
Waktu pemrosesan data	10 sampai 15 detik per kendaraan
Keamanan data	Rentan hilang atau rusak
Akurasi Perhitungan tarif	Berpotensi kesalahan dalam perhitungan dan pencatatan

4.2. Perancangan

Pada tahap perancangan, sistem dimodelkan menggunakan Unified Modeling Language (UML), yang meliputi Use Case Diagram, Activity Diagram, dan Sequence Diagram.

4.2.1. Halaman Login

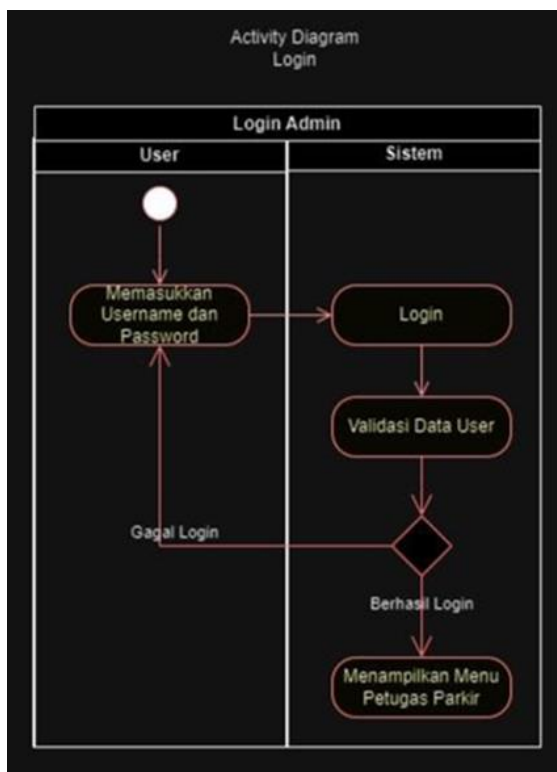
Dalam Use Case Diagram untuk proses login, terdapat aktor utama yaitu "User" yang berinteraksi dengan sistem untuk melakukan autentikasi. Proses diawali ketika user memasukkan username dan password yang kemudian dikirimkan ke sistem. Setelah itu, sistem melakukan validasi data user dengan mencocokkan kredensial yang dimasukkan dengan data yang tersimpan dalam database. Jika informasi yang dimasukkan sesuai, maka sistem memberikan akses kepada user dan proses login dinyatakan berhasil. Namun, jika data yang dimasukkan tidak valid, maka sistem akan menolak akses. Dalam diagram ini juga terdapat hubungan "include" antara use case "Login" dan "Validasi Data User", yang menunjukkan bahwa setiap kali proses login dilakukan, sistem selalu melakukan validasi data terlebih dahulu sebelum menentukan hasil akhirnya.



Gambar 2. Use Case Diagram Login

Dalam Activity Diagram untuk proses login menggambarkan alur aktivitas dari user dan sistem saat proses login berlangsung. Diagram ini terbagi menjadi dua swimlane, yaitu "User" dan "Sistem", yang menunjukkan peran masing-masing dalam proses login. Proses dimulai dari user yang memasukkan

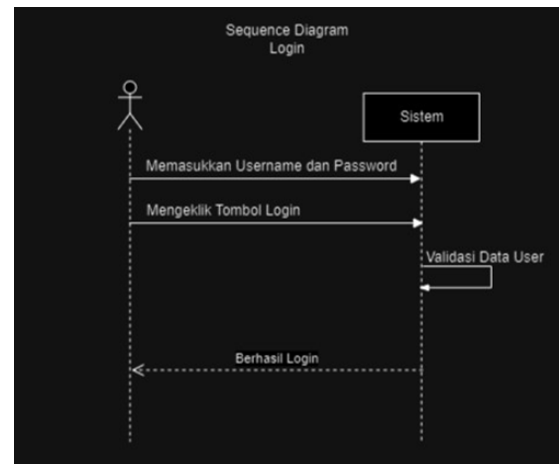
username dan password ke dalam sistem. Setelah input diberikan, sistem memproses permintaan login dengan memvalidasi data yang dimasukkan. Jika validasi berhasil, sistem akan memberikan akses dan menampilkan menu utama untuk petugas parkir. Sebaliknya, jika validasi gagal karena username atau password salah, maka sistem akan meminta user untuk menginput ulang kredensial yang benar. Pada diagram ini, terdapat decision node yang menentukan apakah login berhasil atau gagal. Jika data sesuai dengan yang ada di database, user akan diarahkan ke menu utama petugas parkir. Jika tidak, proses dikembalikan ke user untuk mencoba login kembali.



Gambar 3. Activity Diagram Login

Dalam Sequence Diagram untuk proses login menunjukkan urutan interaksi antara user dan sistem saat user melakukan login. Proses dimulai ketika user memasukkan username dan password ke dalam sistem. Setelah itu, user menekan tombol login, yang mengirimkan permintaan autentikasi ke sistem. Sistem kemudian melakukan validasi data user, yaitu mengecek apakah username dan password yang dimasukkan cocok dengan data yang tersimpan dalam database. Jika validasi berhasil, sistem mengirimkan respon berupa akses login yang

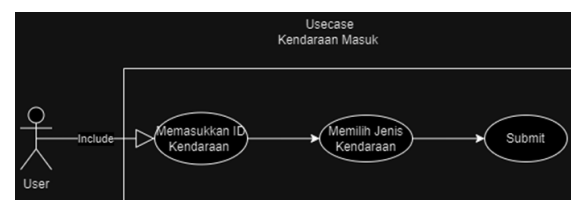
berhasil, sehingga user dapat masuk ke dalam sistem dan menggunakan fitur yang tersedia. Jika validasi gagal, dalam pengembangan lebih lanjut, sistem dapat mengembalikan notifikasi bahwa login gagal dan meminta user untuk mencoba kembali.



Gambar 4. Sequence Diagram Login

4.2.2. Halaman Kendaraan Masuk

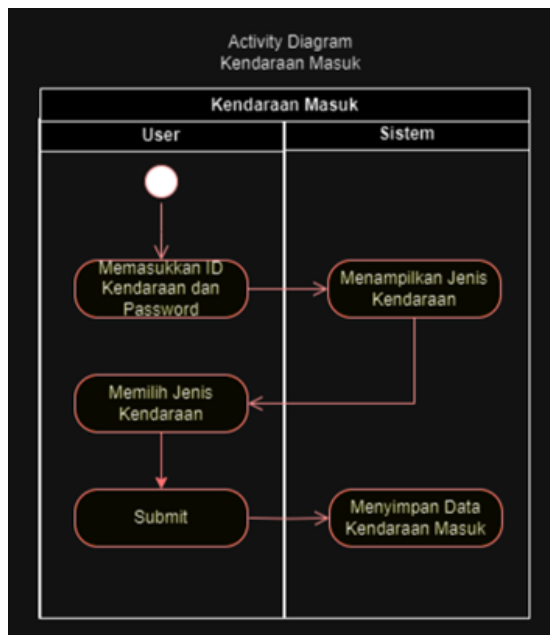
Dalam Use Case Diagram untuk proses kendaraan masuk menunjukkan bagaimana user berinteraksi dengan sistem saat mencatat kendaraan yang masuk. Proses dimulai ketika user, dalam hal ini petugas parkir, memasukkan ID kendaraan ke dalam sistem. ID kendaraan ini dapat berupa nomor polisi kendaraan yang digunakan untuk identifikasi. Setelah itu, user harus memilih jenis kendaraan, seperti roda dua atau roda empat, karena jenis kendaraan akan berpengaruh pada tarif parkir yang dikenakan. Setelah semua data diisi, user menekan tombol submit, yang berarti data kendaraan telah berhasil dicatat dalam sistem.



Gambar 5. Use Case Diagram Kendaraan Masuk

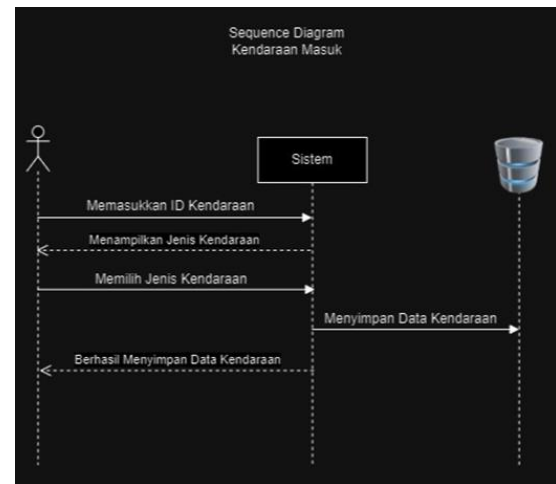
Dalam Activity Diagram untuk proses kendaraan masuk menggambarkan alur aktivitas yang dilakukan oleh user dan sistem dalam mencatat kendaraan yang masuk. Proses dimulai ketika user memasukkan ID kendaraan

ke dalam sistem. ID kendaraan ini bisa berupa nomor polisi kendaraan. Setelah itu, sistem akan menampilkan jenis kendaraan yang tersedia, yang kemudian harus dipilih oleh user sesuai dengan kendaraan yang akan masuk. Setelah memilih jenis kendaraan, user menekan tombol submit, yang mengindikasikan bahwa data sudah siap untuk disimpan. Sistem kemudian akan menyimpan data kendaraan masuk ke dalam database sebagai referensi untuk penghitungan biaya parkir saat kendaraan keluar.



Gambar 6. Activity Diagram Kendaraan Masuk

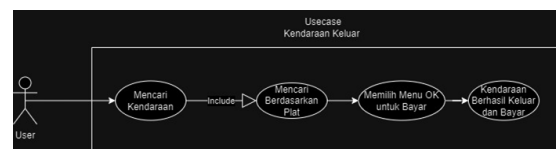
Dalam Sequence Diagram untuk proses kendaraan masuk menggambarkan interaksi antara user, sistem, dan database secara berurutan dalam mencatat data kendaraan yang masuk. Proses dimulai ketika user memasukkan ID kendaraan ke dalam sistem. Sistem kemudian akan menampilkan jenis kendaraan yang sesuai dengan ID yang dimasukkan. Setelah itu, user akan memilih jenis kendaraan, dan sistem akan melanjutkan proses dengan menyimpan data kendaraan ke dalam database. Setelah data kendaraan berhasil disimpan.



Gambar 7. Sequence Diagram Kendaraan Masuk

4.2.3. Halaman Kendaraan Keluar

Dalam Use Case Diagram untuk proses Kendaraan Keluar menggambarkan interaksi antara user dan sistem saat kendaraan akan keluar dari area parkir. Proses dimulai dengan user mencari kendaraan, yang kemudian mencakup langkah mencari berdasarkan plat nomor untuk memastikan kendaraan yang benar teridentifikasi dalam sistem. Setelah kendaraan ditemukan, user harus memilih menu OK untuk membayar biaya parkir yang telah dihitung oleh sistem. Setelah pembayaran berhasil dilakukan, kendaraan dapat berhasil keluar dari area parkir.



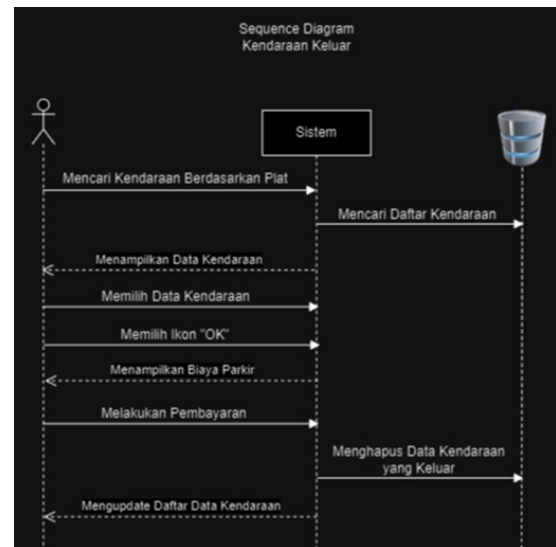
Gambar 8. Use Case Diagram Kendaraan Keluar

Dalam Activity Diagram untuk proses Kendaraan Keluar menunjukkan langkah-langkah yang dilakukan oleh user dan sistem saat kendaraan akan keluar dari area parkir. Proses dimulai dengan user mencari kendaraan berdasarkan plat nomor. Setelah itu, sistem akan menampilkan daftar kendaraan yang sesuai dengan pencarian. Selanjutnya, user memilih kendaraan yang ingin dikeluarkan, lalu sistem akan menampilkan biaya parkir yang harus dibayarkan. Setelah melihat biaya parkir, user harus memilih ikon "OK". Setelah pembayaran berhasil, sistem akan mengeluarkan kendaraan.



Gambar 9. Activity Diagram Kendaraan Keluar

Dalam Sequence Diagram untuk proses Kendaraan Keluar menggambarkan alur komunikasi antara User, Sistem, dan Database dalam proses kendaraan keluar dari area parkir. User memulai dengan mencari kendaraan berdasarkan plat nomor. Sistem kemudian mencari daftar kendaraan yang tersedia dan menampilkan data kendaraan yang sesuai kepada user. Setelah data kendaraan ditampilkan, user memilih kendaraan yang ingin dikeluarkan. User kemudian menekan ikon "OK", yang membuat sistem menampilkan biaya parkir yang harus dibayar. Sistem kemudian menghapus data kendaraan yang sudah keluar dari database. Setelah itu, sistem memperbarui daftar kendaraan yang masih berada di area parkir untuk memastikan data tetap akurat.



Gambar 10. Activity Diagram Kendaraan Keluar

Selain perancangan UML, struktur database juga dirancang menggunakan MySQL. Database ini menyimpan informasi kendaraan, transaksi parkir, dan data pengguna.

Tabel Kendaraan digunakan untuk menyimpan informasi kendaraan yang masuk dan keluar dari area parkir. Data ini diperlukan untuk mencatat durasi parkir kendaraan dan menghitung biaya parkir. Tabel ini memiliki beberapa kolom, yaitu `id_kendaraan` yang berfungsi sebagai Primary Key (PK) dan digunakan sebagai ID unik untuk setiap kendaraan yang masuk ke sistem. `plat_nomor` menyimpan nomor polisi kendaraan untuk identifikasi, sedangkan `jenis` menunjukkan jenis kendaraan, misalnya motor atau mobil, yang digunakan untuk menentukan tarif parkir. `waktu_masuk` mencatat waktu kendaraan masuk ke area parkir, sementara `waktu_keluar` mencatat waktu kendaraan keluar dari area parkir.

Tabel 2. Database Informasi Kendaraan

Nama Field	Tipe Data
<code>id_kendaraan</code>	INT (PK)
<code>plat_nomor</code>	VARCHAR(15)
<code>jenis_kendaraan</code>	VARCHAR(10)
<code>waktu_masuk</code>	TIME
<code>Waktu_keluar</code>	TIME

Tabel Transaksi digunakan untuk mencatat semua transaksi pembayaran parkir. Setiap transaksi memiliki informasi mengenai

jumlah biaya parkir yang harus dibayarkan. Tabel ini memiliki id_transaksi sebagai Primary Key (PK) yang menjadi ID unik untuk setiap transaksi pembayaran. id_kendaraan berperan sebagai Foreign Key (FK) yang menghubungkan transaksi dengan kendaraan yang dikenakan biaya parkir. Kolom biaya menyimpan jumlah uang yang dibayarkan berdasarkan durasi parkir dan jenis kendaraan.

Tabel 3. Database Transaksi

Nama Field	Tipe Data
id_transaksi	INT (PK)
id_kendaraan	INT (FK)
biaya	INT

Tabel Pengguna menyimpan informasi mengenai pengguna sistem yang terdiri dari petugas parkir dan admin. Tabel ini memastikan hanya pengguna yang terdaftar yang dapat mengakses sistem. Beberapa kolom yang terdapat dalam tabel ini adalah id_user sebagai Primary Key (PK) yang berfungsi sebagai ID unik untuk setiap pengguna sistem. Username digunakan sebagai nama pengguna yang berfungsi untuk login ke sistem, sementara password menyimpan kata sandi yang telah dienkripsi guna menjaga keamanan akses. Dengan adanya tabel ini, hanya pengguna yang memiliki hak akses yang dapat mengelola data kendaraan dan transaksi dalam sistem parkir.

Tabel 4. Database Pengguna

Nama Field	Tipe Data
id_user	INT (PK)
username	VARCHAR(25)
password	VARCHAR(15)

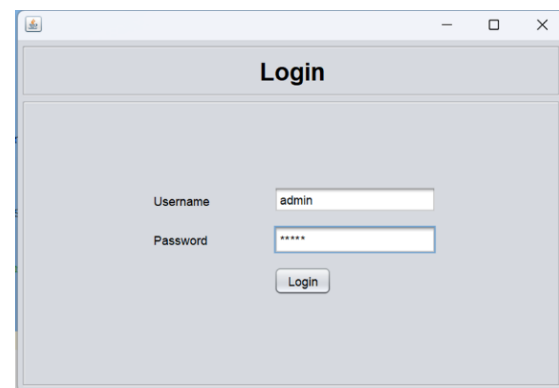
4.3. Implementasi

Tahap implementasi dilakukan dengan mengembangkan sistem menggunakan bahasa pemrograman Java dengan NetBeans sebagai Integrated Development Environment (IDE). Berikut hasilnya:

4.3.1. Halaman Login

Dalam tampilan antarmuka halaman login memiliki elemen-elemen utama yang terdiri dari judul "Login" yang terletak di bagian atas jendela untuk menunjukkan bahwa pengguna harus memasukkan kredensial mereka sebelum dapat mengakses sistem. Di bawah judul terdapat dua bidang input, yaitu Username dan Password, yang memungkinkan

pengguna untuk memasukkan nama pengguna serta kata sandi mereka. Tombol "Login" terletak di bawah bidang input dan berfungsi sebagai pemicu proses autentikasi. Jika pengguna memasukkan kredensial yang benar, mereka akan diarahkan ke halaman utama sistem parkir. Namun, jika terjadi kesalahan, sistem akan menampilkan pesan bahwa username atau password yang dimasukkan tidak valid.



Gambar 11. Halaman Login

4.3.2. Halaman Kendaraan Masuk

Dalam antarmuka Kendaraan Masuk digunakan untuk mencatat informasi kendaraan yang baru saja tiba di lokasi parkir. Formulir yang ditampilkan terdiri dari beberapa elemen penting. ID Kendaraan digunakan untuk memasukkan nomor polisi kendaraan yang masuk. Jenis Kendaraan tersedia dalam bentuk dropdown menu, yang memungkinkan petugas memilih jenis kendaraan. Tanggal Masuk dan Jam Masuk otomatis menampilkan waktu saat kendaraan dicatat ke dalam sistem. Di bagian bawah, terdapat tombol "Submit", yang berfungsi untuk menyimpan data kendaraan ke dalam database setelah petugas memastikan informasi yang dimasukkan sudah benar. Selain itu, terdapat tombol "Exit" di sudut kanan bawah yang memungkinkan petugas keluar dari menu ini atau menutup aplikasi.

Gambar 12. Halaman Kendaraan Masuk

4.3.3. Halaman Kendaraan Keluar

Dalam antarmuka Kendaraan Keluar digunakan untuk mencatat kendaraan yang akan meninggalkan area parkir dan menghitung biaya parkir yang harus dibayarkan oleh pengguna. Di bawahnya, terdapat kolom "Search", yang memungkinkan petugas mencari kendaraan berdasarkan nomor plat untuk memverifikasi informasi sebelum melakukan proses keluar. Tabel di tengah layar menampilkan daftar kendaraan yang telah masuk, termasuk informasi Plat Nomor, Jenis Kendaraan, Jam Masuk, dan Tanggal Masuk. Di bagian bawah, terdapat kolom "Bayar", di mana sistem akan menampilkan biaya parkir yang harus dibayar oleh pengguna berdasarkan durasi parkir dan tarif yang berlaku. Petugas dapat mengonfirmasi pembayaran dengan menekan tombol "OK" setelah pengguna membayar. Sebagai tambahan, di sisi kanan bawah, sistem juga menampilkan Tanggal dan Jam saat ini, yang menunjukkan waktu real-time saat transaksi kendaraan keluar dilakukan. Tombol "Exit" di sudut kanan bawah memungkinkan petugas keluar dari menu atau menutup aplikasi jika diperlukan.

Gambar 13. Halaman Kendaraan Keluar

4.4. Pengujian

Pengujian sistem dilakukan menggunakan metode Black Box Testing untuk memastikan setiap fitur berjalan sesuai dengan spesifikasi yang dirancang. Beberapa pengujian yang dilakukan meliputi:

Tabel 5. Hasil Black Box Testing

Test	Hasil yang Diharapkan	Status
Login	Input username dan password benar → Akses diterima	Valid
Login	Input username atau password salah → Akses ditolak	Valid
Pencatatan Kendaraan Masuk	Input data kendaraan → Data tersimpan di database	Valid
Pencatatan Kendaraan Keluar	Menekan kendaraan yang keluar → Sistem menampilkan detail kendaraan	Valid
Perhitungan Biaya Parkir	Sistem menghitung tarif berdasarkan durasi parkir	Valid

4.5. Pemeliharaan

Tahap terakhir dalam metode Waterfall adalah pemeliharaan sistem. Setelah implementasi dan pengujian, sistem terus dievaluasi untuk memastikan performa yang optimal. Pemeliharaan dilakukan dengan pemantauan sistem secara berkala untuk mengevaluasi kinerja dan memastikan kestabilannya. Jika ditemukan masalah teknis atau bug dalam sistem, maka perbaikan akan segera dilakukan agar sistem tetap berjalan dengan baik. Dengan adanya pemeliharaan yang berkelanjutan, sistem parkir otomatis ini dapat terus memberikan layanan yang efisien dan andal bagi penggunaannya.

5. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem manajemen parkir berbasis pemrograman berorientasi objek yang mampu meningkatkan efisiensi pencatatan kendaraan dan transaksi parkir. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan memiliki akurasi tinggi dalam pencatatan kendaraan, mempercepat proses administrasi, dan meningkatkan keamanan data dengan menggunakan database terstruktur. Dibandingkan dengan sistem manual, sistem ini menawarkan keunggulan dalam hal kecepatan

pemrosesan dan minimnya potensi kesalahan pencatatan. Dengan menerapkan metode Waterfall dalam pengembangan perangkat lunak, penelitian ini membuktikan bahwa sistem parkir otomatis dapat menjadi solusi efektif dalam mengatasi permasalahan yang dihadapi dalam pengelolaan parkir konvensional.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Singaperbangsa Karawang atas dukungan akademik yang diberikan dalam penelitian ini. Penghargaan juga disampaikan kepada semua pihak yang telah berkontribusi dalam pengumpulan data, pengembangan sistem, serta pengujian aplikasi ini. Selain itu, terima kasih kepada keluarga dan teman-teman yang telah memberikan motivasi serta dukungan moral selama proses penelitian dan penyusunan laporan ini.

DAFTAR PUSTAKA

- [1] P. Pt, T. Alfa, S. Mandiri, N. D. Arizona, and M. Rajagukguk, "Impelementasi Sistem Parkir Otomatis Berbasis Java Desktop Menggunakan Metode Pengembangan Waterfall," vol. 5, no. 2, pp. 145–152, 2024.
- [2] R. S. Eka Saudur Sihombing, "1,2 1 , 涂海丽 2," *Eka Saudur Sihombing , Rindi Sitepu*, vol. 15, no. 11, pp. 102–106, 2019.
- [3] V. D. Puspa and Nuryuliani, "Analisis Dan Perancangan Basis Data Pada Aplikasi," no. 11107722, 2002.
- [4] M. Usnaini, V. Yasin, and A. Z. Sianipar, "Perancangan sistem informasi inventarisasi aset berbasis web menggunakan metode waterfall," *J. Manajamen Inform. Jayakarta*, vol. 1, no. 1, p. 36, 2021, doi: 10.52362/jmijayakarta.v1i1.415.
- [5] N. Hidayati, "Penggunaan Metode Waterfall Dalam Rancang Bangun Sistem Informasi Penjualan," *Gener. J.*, vol. 3, no. 1, pp. 1–10, 2019, [Online]. Available: <https://ojs.unpkediri.ac.id/index.php/gj/article/view/12642>
- [6] M. Siddik and S. Samsir, "Rancang Bangun Sistem Informasi Pos (Point of Sale) Untuk Kasir Menggunakan Konsep Bahasa Pemrograman Orientasi Objek," *JOISIE (Journal Inf. Syst. Informatics Eng.)*, vol. 4, no. 1, p. 43, 2020, doi: 10.35145/joisie.v4i1.607.
- [7] G. Science and E. Outlook, "何霄嘉 1 , 郑大玮 2 , 许吟隆 3 (1.," vol. 32, no. 2, pp. 58–65, 2020.
- [8] A. M. Bayu Aji and B. Rudianto, "Sistem Informasi Pembayaran Parkir Berbasis Web Dengan Menggunakan Model Waterfall," *INTI Nusa Mandiri*, vol. 15, no. 1, pp. 9–16, 2020, doi: 10.33480/inti.v15i1.1367.
- [9] D. A. Safitri, "Rancang Bangun Aplikasi Penginputan Data Sensus Penduduk Berbasis Dekstop," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3821.
- [10] R. Hidayat, A. Satriansyah, and M. S. Nurhayati, "Penggunaan Metode Waterfall untuk Rancangan Bangun Aplikasi Penyewaan Lapangan Olahraga," *BIOS J. Teknol. Inf. dan Rekayasa Komput.*, vol. 3, no. 1, pp. 9–16, 2022, doi: 10.37148/bios.v3i1.35.
- [11] S. P. Pratama and B. Sujatmiko, "Pengembangan Aplikasi Goop Untuk Meningkatkan Kompetensi Object Oriented Programming Menggunakan Project Based Learning Pada Mata Pelajaran Pemrograman Berorientasi Objek," *IT-Edu J. Inf. Technol. Educ.*, vol. 9, no. 2, pp. 1–8, 2024, doi: 10.26740/it-edu.v9i2.61262.
- [12] M. A. Taufan, D. S. Rusdianto, and M. T. Ananta, "Pengembangan Sistem Otomatisasi Use Case Diagram berdasarkan Skenario Sistem menggunakan Metode POS Tagger Stanford NLP," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 8, pp. 3733–3740, 2022, [Online]. Available: <http://j-ptiik.ub.ac.id>