

# Sistem Informasi Sewa Kendaraan Berbasis Website Menggunakan Metode Model View Controller

Nityasa Tustika<sup>\*1</sup>, Noora Qotrun Nada<sup>2</sup>

<sup>1,2</sup> Program Studi Informatika, Universitas PGRI Semarang, Kota Semarang

\*Email: [tustika.nityasa@gmail.com](mailto:tustika.nityasa@gmail.com)

## Abstract

As technology continues to rapidly advance, the use of websites in business has become a common practice. The application of information systems in vehicle rental businesses enables renters to easily and quickly make online reservations. This rental information system was developed using PHP programming language, Laravel framework, and MySQL as relational database management system. In this study, the website was built with using the Model View Controller (MVC) method. By applying the MVC approach, the development and maintenance of the website can be managed efficiently because there is a separation between business logic, presentation, and control. If changes are made to one component, it will not directly affect other components. The result of this study concludes that the rental information system using MVC method not only provides a technical solution to improve the efficiency of website development and maintenance but also enhances the user experience in the rental process.

Keywords: Vehicle Rental Information System; Website; Model View Controller; Waterfall

## Abstrak

Seiring dengan perkembangan teknologi yang pesat ini, penggunaan website dalam bidang bisnis menjadi hal yang sudah umum digunakan. Sistem informasi dalam bisnis persewaan kendaraan memungkinkan penyewa untuk melakukan pemesanan kendaraan secara online dengan mudah dan cepat. Sistem informasi sewa kendaraan ini dibuat menggunakan bahasa pemrograman PHP, framework Laravel, dan MySQL sebagai sistem pengelola basis data relasional. Dalam penelitian ini, website dibangun dengan menggunakan metode *Model View Controller* (MVC). Dengan menerapkan pendekatan MVC, pengembangan dan pemeliharaan website dapat dikelola secara efisien dan terstruktur karena terdapat pemisahan antara logika bisnis, tampilan, dan kontrol yang apabila terdapat perubahan pada suatu komponen tidak akan secara langsung mempengaruhi komponen lainnya. Hasil dari penelitian ini, dapat disimpulkan bahwa sistem informasi sewa kendaraan menggunakan metode MVC tidak hanya memberikan solusi teknis untuk meningkatkan efisiensi pengembangan dan pemeliharaan website namun juga meningkatkan pengalaman pengguna dalam proses penyewaan kendaraan.

Kata kunci: Sistem Informasi Rental Kendaraan; Website; Model View Controller; Waterfall

## 1. Pendahuluan

Alat transportasi menjadi salah satu kebutuhan utama dalam menjalani aktivitas sehari-hari di masyarakat. Di era yang semakin modern ini, sistem bisnis konvensional sudah mulai tersisihkan. Proses bisnis konvensional yang kurang efisien dan efektif mengurangi minat masyarakat terhadap produk yang ditawarkan termasuk dalam bisnis rental kendaraan[1]

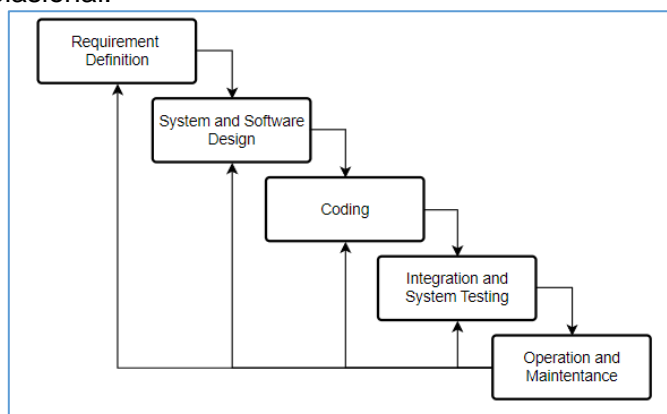
Untuk mengatasi tantangan tersebut, beberapa penelitian telah mulai mengembangkan sistem informasi berbasis web. Sistem informasi memberikan kemudahan bagi pemilik dalam memberikan informasi yang akurat dan mengecek pemesanan kendaraan. Selain itu, sistem ini juga memudahkan customer dalam melihat ketersediaan kendaraan dan melakukan proses

penyewaan tanpa harus datang ke lokasi rental [2]. Sistem informasi berbasis web mampu memudahkan admin untuk mengatur data-data perusahaan agar lebih mudah dan efisien [3]. MVC adalah sebuah konsep arsitektur yang mengelompokkan pembangunan aplikasi mejadi 3 bagian besar yaitu pemrosesan (*model*), proses manipulasi (*controller*), serta tampilan (*view*) untuk ditampilkan pada sebuah antarmuka pengguna [4]. Pendekatan MVC mampu memisahkan tampilan, logika bisnis, dan kontrol. Pada pengembangan MVC, modifikasi yang dilakukan terhadap suatu komponen tidak mempengaruhi komponen lain [5]. Pengembangan sistem informasi sewa kendaraan menggunakan model Waterfall mempermudah visualisasi interaksi antara aktor dan sistem [6].

Adapun tujuan dari penelitian ini adalah untuk meningkatkan efisiensi proses pemesanan dan manajemen inventaris kendaraan serta memastikan struktur sistem yang mudah dipelihara dan dikembangkan lebih lanjut. Pendekatan MVC dapat diimplementasikan secara efektif dalam sistem informasi sewa kendaraan, serta untuk meningkatkan pengalaman pengguna dalam proses penyewaan kendaraan.

## 2. Metode

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode *waterfall*. Model ini menggunakan pendekatan yang sistematis dan terstruktur, dimana setiap tahap harus diselesaikan sebelum melanjutkan ke tahap berikutnya. Pengembangan dengan model Waterfall dimulai dari tahap perencanaan, kemudian dilanjutkan dengan tahap-tahap berikutnya secara linear hingga mencapai tahap pemeliharaan [7]. Sistem ini dibuat menggunakan bahasa pemrograman PHP, framework Laravel, dan MySQL sebagai sistem pengelola basis data relasional.



Gambar 1 Metode Waterfall

Gambar 1 adalah gambar diagram alir metode *waterfall* yang digunakan dalam pengembangan sistem.

Metode waterfall terdiri dari tahapan-tahapan berikut:

1. Requirement Definition (analisis kebutuhan)  
Tahapan ini meliputi proses pengumpulan dan analisis data apa saja yang dibutuhkan dalam pengembangan website.
2. System and Software Design (Perencanaan dan pembuatan konsep)  
Hasil analisis kebutuhan dari tahap sebelumnya kemudian dirancang menjadi desain sistem informasi sewa kendaraan.
3. Coding (Penulisan kode program)  
Desain sistem yang telah dibuat kemudian diimplementasikan dalam bentuk kode program. Pada project ini bahasa pemrograman yang digunakan adalah PHP dengan framework Laravel, dan MySQL sebagai database.
4. Integration and System testing (Pengujian) Program yang telah dibangun kemudian di uji coba untuk diperiksa apakah program sudah dapat digunakan atau perlu perbaikan.
5. Operation and Maintenance (Pemeliharaan)  
Tahapan ini dilakukan apabila aplikasi perlu pemeliharaan dari kesalahan yang tidak ditemukan pada langkah pengujian sebelumnya.

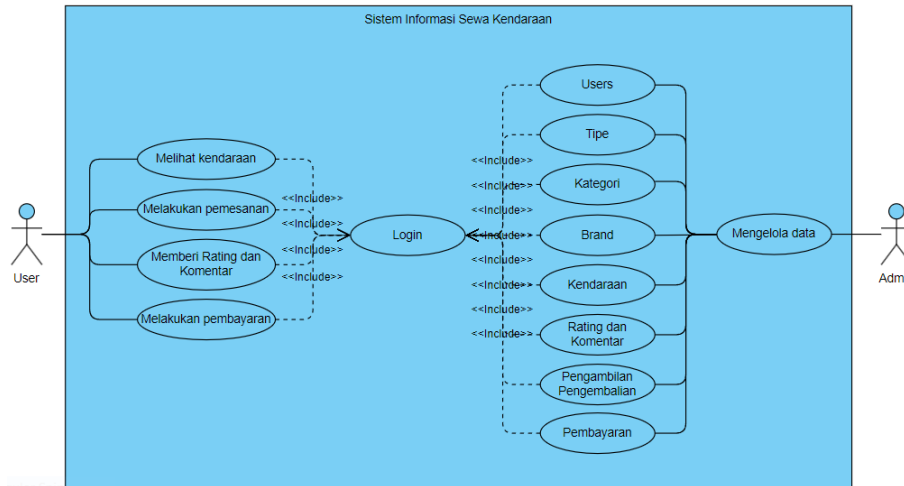
Dalam penelitian ini, tahapan yang dilakukan adalah tahapan requirement definition sampai implementasi and unit testing.

### 3. Hasil dan Pembahasan

#### 3.1. Penyajian Hasil

Pada bagian ini akan disajikan hasil penelitian yang telah dilakukan untuk memberika gambaran yang jelas terkait hasil penelitian.

##### a. Use Case Diagram

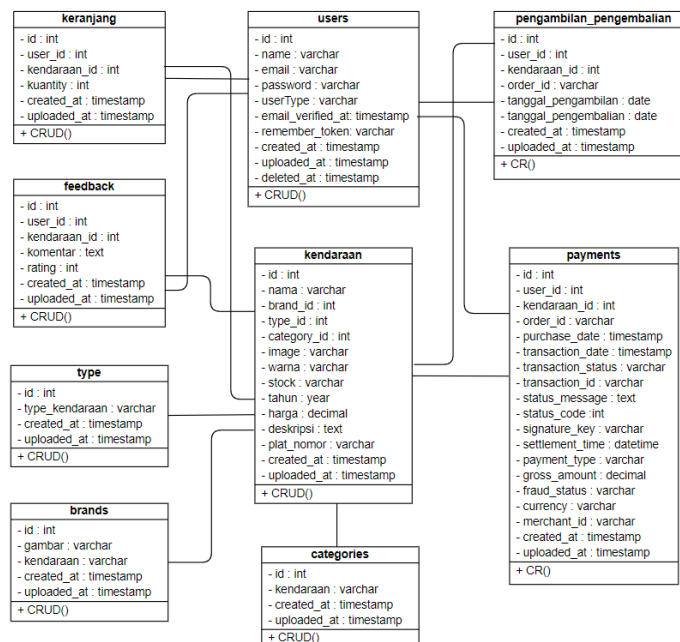


Gambar 2 Use Case Diagram

Gambar 2 adalah gambar *Use Case Diagram* aktor admin dan aktor user. Admin mendapatkan hak akses dengan akun *default* yang diberikan saat website selesai dibuat, admin dapat memperbarui *password login*. Fitur yang dapat diakses oleh admin yaitu, mengelola data pelanggan, mengelola data admin, pengelola data kendaraan, mengelola pesanan dari pelanggan.

Aktor user dapat melihat halaman *landing page* tanpa harus melakukan *register* dan *login* terlebih dahulu. Untuk mengakses halaman sewa, tentang kami, dan memasukkan produk ke keranjang, user diharuskan melakukan *login* terlebih dahulu.

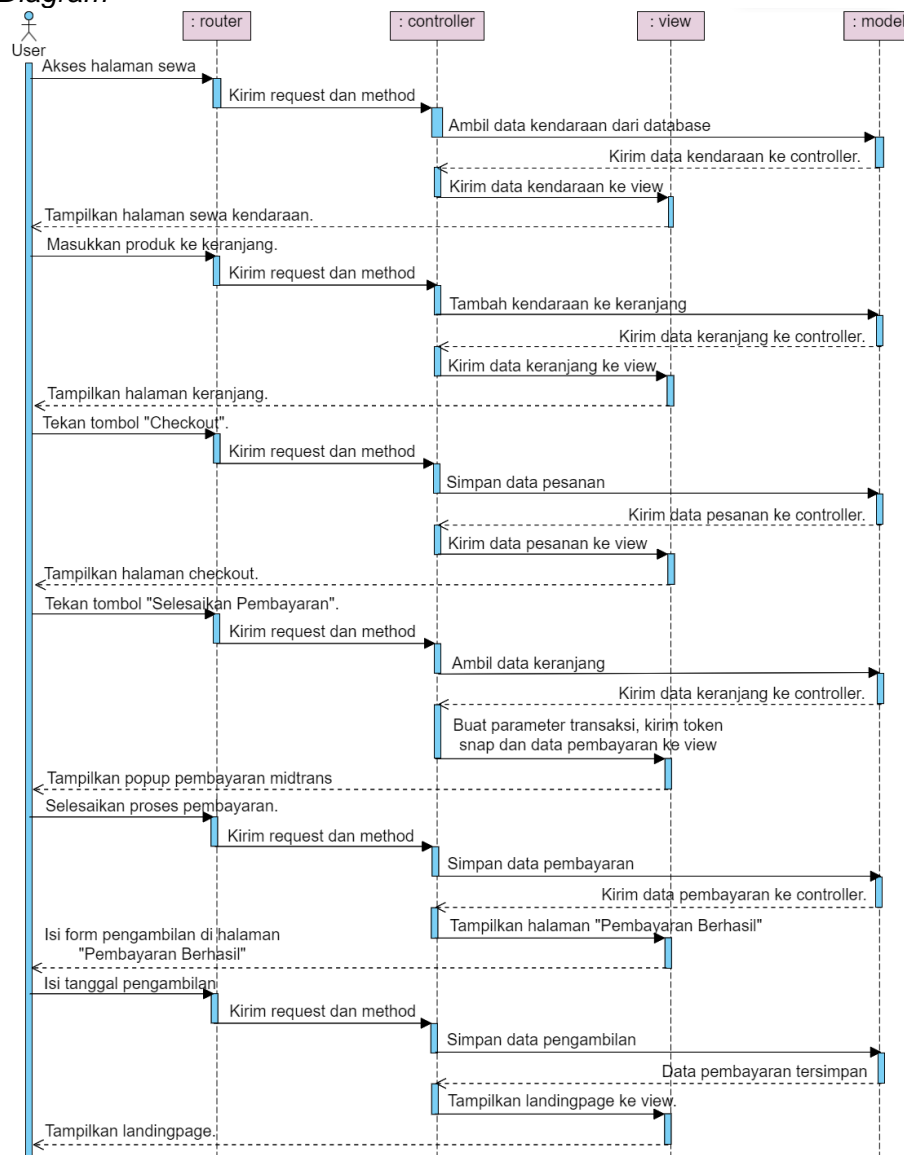
##### b. Class Diagram



Gambar 3 Class Diagram

Gambar 3 menunjukkan *class diagram* pada *database* sistem. *Class diagram* menggambarkan jenis-jenis atribut dan hubungan antar mereka. *Class diagram* menunjukkan batasan operasi yang dilakukan sebuah *class*. Pada kelas user terdapat atribut *userType* yang menunjukkan peran pengguna admin atau user. Kelas ini memiliki metode *create*, *read*, *update*, dan *delete*. Pada kelas kendaraan terdapat operasi *create*, *read*, *update*, dan *delete* yang dapat diakses oleh admin. User hanya dapat melakukan operasi *read* saja. Pada kelas *categories* terdapat operasi operasi *create*, *read*, *update*, dan *delete* yang dapat diakses oleh admin. User hanya dapat melakukan operasi *read* saja. Pada kelas *payments* terdapat operasi *create*, dan *read* yang dapat diakses user user. Admin hanya dapat melakukan operasi *read*. Pada kelas pengambilan-pengembalian terdapat operasi *create*, dan *read* yang dapat diakses user user. Admin hanya dapat melakukan operasi *read*. Pada kelas *brands* terdapat operasi *create*, *read*, *update*, dan *delete* yang dapat diakses oleh admin. User hanya dapat melakukan operasi *read* saja. Pada kelas *type* terdapat operasi *create*, *read*, *update*, dan *delete* yang dapat diakses oleh admin. User hanya dapat melakukan operasi *read* saja. Pada kelas *feedback* terdapat operasi *create*, *read*, dan *update* bagi user. Admin dapat melakukan operasi *delete* pada *feedback*. Pada kelas keranjang terdapat operasi *create*, *read*, *update*, dan *delete* bagi user saja.

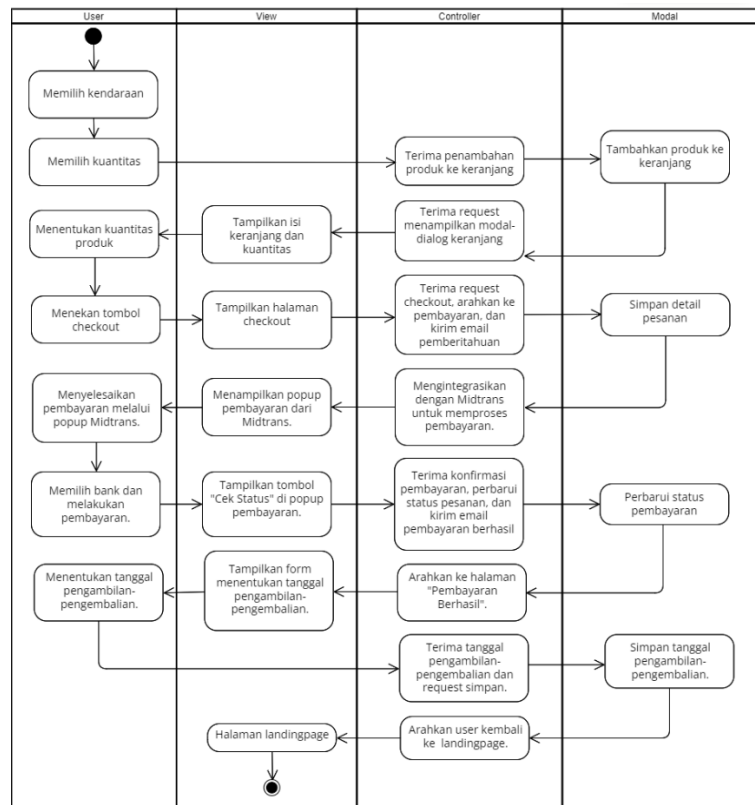
c. Sequence Diagram



Gambar 4 Sequence Diagram

Gambar 4 menunjukkan *sequence diagram user* melakukan pemesanan produk. *User* yang telah *login* dapat menambahkan kendaraan ke dalam keranjang, mengakses halaman keranjang belanja, mengelola isi keranjang, melanjutkan proses *checkout*, menyelesaikan proses pembayaran, menentukan pengambilan dan pengembalian produk.

d. *Activity Diagram*

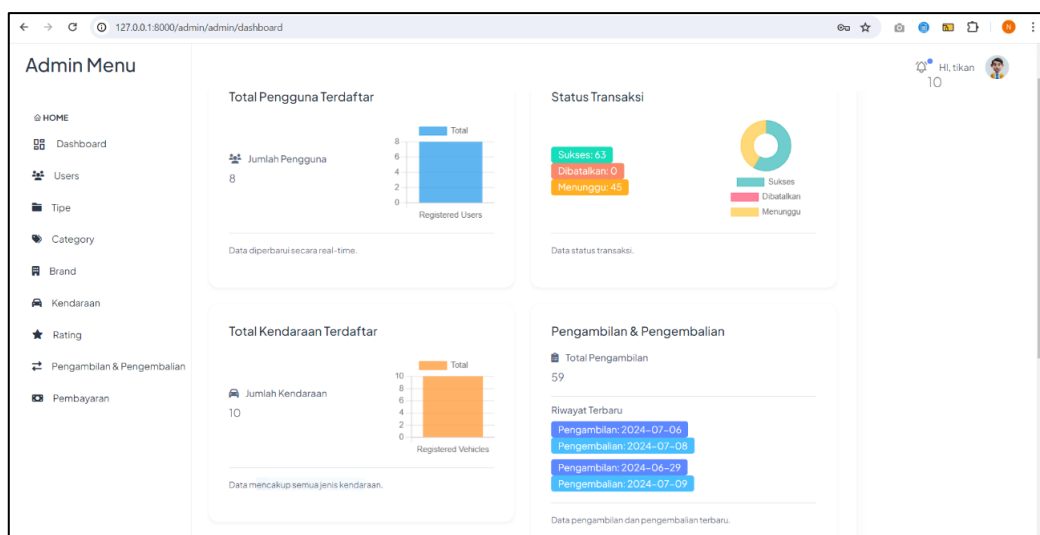


Gambar 5 Activity Diagram

Gambar 5 menunjukkan alur *activity diagram user* melakukan pemesanan produk. *Activity diagram* menunjukkan aktivitas yang terjadi di dalam proses user saat melakukan pemesanan produk

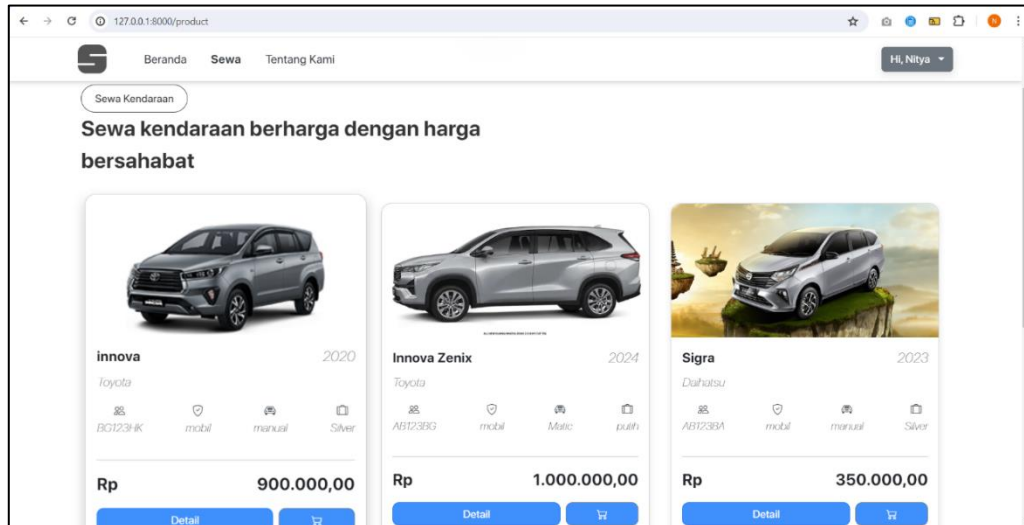
3.2. **Pembahasan**

Pada bagian ini hasil-hasil yang telah disajikan akan dibahas lebih mendalam.



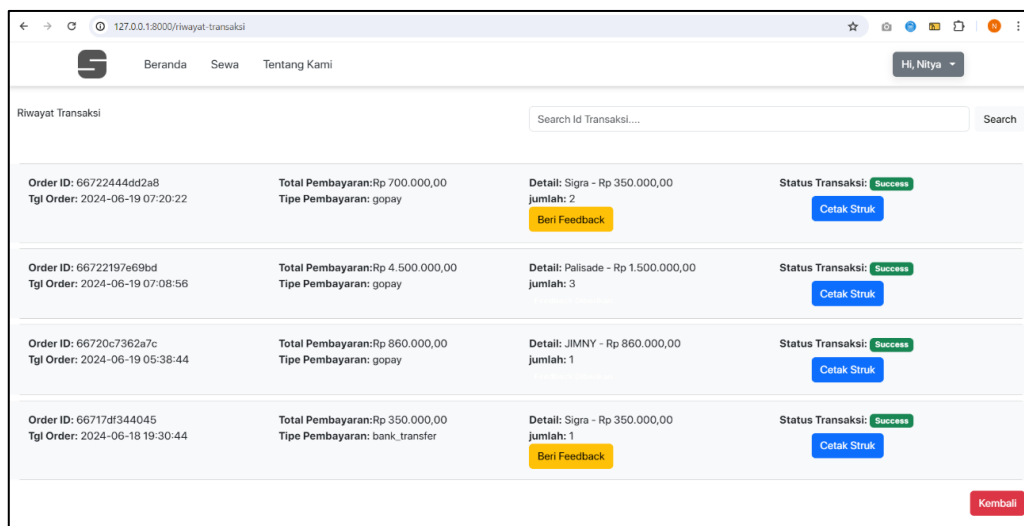
Gambar 6 Halaman Dashboard Admin

. Gambar 6 menunjukkan tampilan halaman admin, berisi berbagai fitur yang hanya dapat diakses oleh *admin*. Admin dapat memantau pembayaran dan pengambilan pengembalian kendaraan, melakukan penambahan, pembaruan, dan penghapusan data pada menu tipe, category, brand, dan kendaraan. Pada menu users dan rating admin dapat melihat dan melakukan penghapusan.



Gambar 7 Halaman Sewa Produk

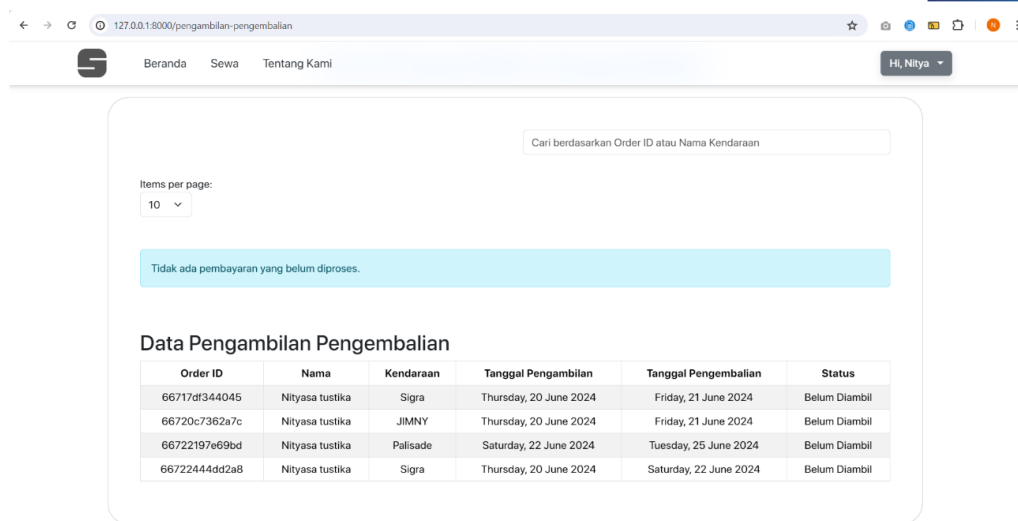
Gambar 8 menunjukkan halaman sewa produk. Halaman ini berisi semua produk website dan hanya dapat dikunjungi setelah melakukan *login*. User dapat memasukkan produk ke keranjang dan melihat detail produk.



Gambar 8 Halaman Riwayat Transaksi

Gambar 8 adalah tampilan halaman riwayat transaksi *user*. Halaman ini dapat diakses setelah *user* melakukan login. Halaman ini menampilkan riwayat sewa *user* selama melakukan pemesanan menggunakan website ini.





Gambar 9 Halaman Pengambilan

Gambar 9 menunjukkan tampilan halaman pengambilan pengembalian. Semua informasi terkait pengambilan dan pengembalian produk sewa tercatat di halaman ini.

### 3.2.1 Pengujian

Setelah tahap desain dan pengkodean selesai, selanjutnya dilakukan pengujian program untuk mengetahui apakah program sudah dapat berjalan sesuai dengan input dan output yang diharapkan.

Tabel 1 Tabel Pengujian Website

Skenario Pengujian	Data Sample Uji	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
Jumlah karakter lebih sedikit dari ketentuan minimal pengisian <i>password</i>	<i>Password</i> : abc	Input data gagal	Input data gagal	<b>Berhasil</b>
Melakukan operasi <i>read</i> pada <i>dashboard</i> admin	<i>Dashboard</i> admin	Data berhasil tampil	Data berhasil tampil	<b>Berhasil</b>
Melakukan operasi <i>create</i> pada <i>dashboard</i> admin	Tipe kendaraan: <i>manual</i>	Input data berhasil	Input data berhasil	<b>Berhasil</b>
Melakukan operasi <i>edit</i> pada <i>dashboard</i> admin	Tipe kendaraan: <i>matic</i>	Input data berhasil	Input data berhasil	<b>Berhasil</b>
Melakukan operasi <i>delete</i> pada <i>dashboard</i> admin	Hapus tipe kendaraan : <i>manual</i>	Hapus data berhasil	Hapus data berhasil	<b>Berhasil</b>
Mengakses halaman sewa tanpa <i>login</i>	Menekan menu sewa tanpa <i>login</i>	Akses ditolak dan diarahkan ke halman <i>login</i>	Akses ditolak dan diarahkan ke halman <i>login</i>	<b>Berhasil</b>
Memasukkan produk ke keranjang	Menekan tombol keranjang di sebuah produk	Muncul produk yang dipilih di keranjang	Muncul produk yang dipilih di keranjang	<b>Berhasil</b>

Tabel 1 menunjukkan hasil pengujian website untuk memastikan website terbebas dari *error* dari user maupun sistem saat dijalankan. Hasil dari pengujian tersebut, website telah berjalan sesuai harapan berdasarkan skenario dan data sample uji yang diujikan.

#### 4. Kesimpulan

Hasil penelitian menunjukkan bahwa sistem informasi sewa kendaraan berbasis website yang dikembangkan menggunakan metode MVC memungkinkan pengguna dapat melakukan penyewaan kendaraan dengan cepat dan mudah serta memudahkan admin dalam mengelola ketersediaan kendaraan. Pemisahan antara logika bisnis, kontrol, dan tampilan dalam metode MVC mengurangi resiko kesalahan dan meningkatkan fleksibilitas pengembangan sistem.

#### 5. Referensi

- [1] Osvaldo Silitonga and Novrini Hasti, "Sistem Informasi Rental Mobil Berbasis Website," *J. Teknol.*, vol. 10, no. 2, pp. 47–50, Nov. 2020, doi: 10.51132/teknologika.v10i2.93.
- [2] S. Panesa and A. Mirza, "Perancangan Sistem Informasi Penyewaan Rental Mobil Berbasis Website (Studi kasus: Riz Rent Cars)," vol. 2, no. 2, 2022.
- [3] H. Riyadli, A. Arliyana, and F. E. Saputra, "Rancang Bangun Sistem Informasi Keuangan Berbasis WEB," *J. Sains Komput. Dan Teknol. Inf.*, vol. 3, no. 1, pp. 98–103, Nov. 2020, doi: 10.33084/jsakti.v3i1.1770.
- [4] F. Fadhullah, "Sistem Informasi Geografis Pemetaan Industri Kecil Menengah di Kabupaten Pidie Berbasis Androd Dengan Metode MVC (Model View Controller)," *Future Acad. J. Multidiscip. Res. Sci. Adv.*, vol. 1, no. 1, pp. 46–53, Sep. 2023, doi: 10.61579/future.v1i1.5.
- [5] B. R. P. Surya, A. P. Kharisma, and N. Yudistira, "Perbandingan Kinerja Pola Perancangan MVC, MVP, dan MVVM Pada Aplikasi Berbasis Android (Studi kasus : Aplikasi Laporan Hasil Belajar Siswa SMA BSS)".
- [6] Y. J. Palis, P. W. Siswanto, and E. Alexander, "Perancangan Pembuatan Website Untuk Rental Mobil Pada Toko Cakrawala Trans," *SIMKOM*, vol. 9, no. 1, pp. 67–78, Jan. 2024, doi: 10.51717/simkom.v9i1.368.
- [7] A. A. Wahid, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi," 2020.