Day 4 - Dynamic Frontend Components
Car Rental E-Commerce

Overview

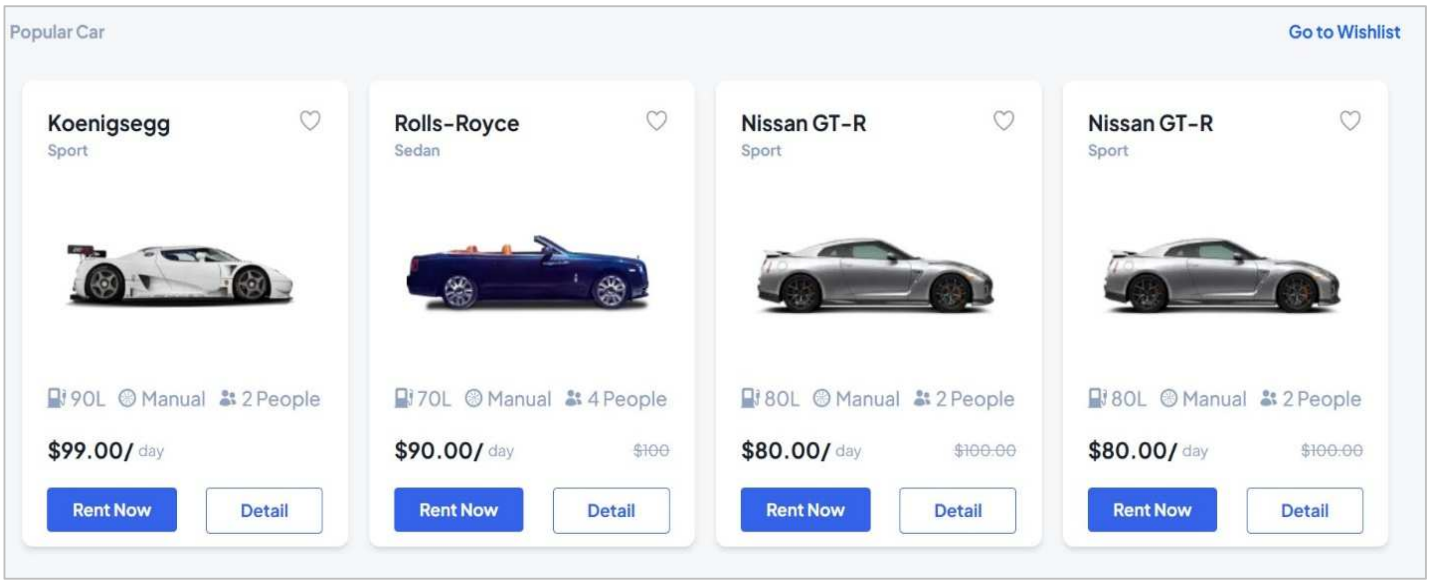
This document outlines the development process, features, and components implemented in the marketplace project, which dynamically displays cars data using Sanity CMS or APIs. The project incorporates professional, scalable, and responsive design principles, ensuring an enhanced user experience across devices.

Features Implemented

1. Car Listing Page

- **Description**: A fully functional product listing page displays dynamic data fetched from Sanity CMS or APIs.
- **Features**:
    - Products rendered dynamically in a grid layout.

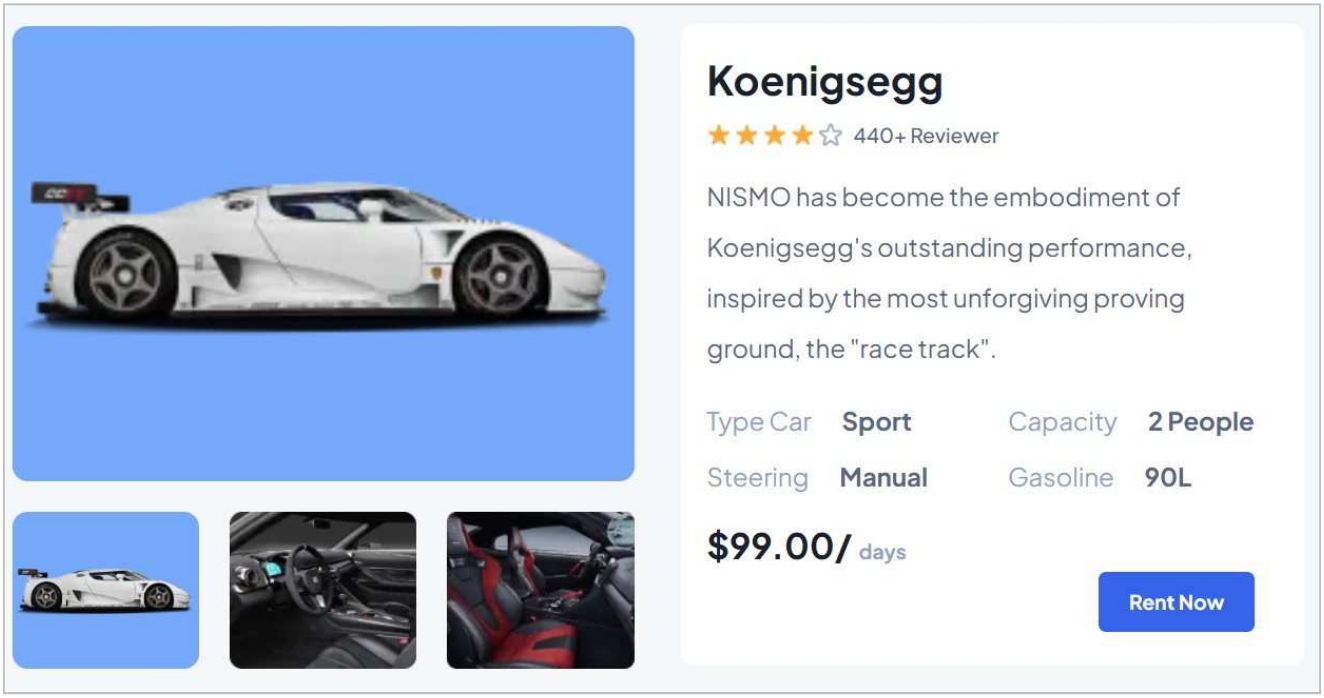
- Fields displayed include:
  - Car Name
  - Type
  - Fuel Capacity
  - Transmission
  - Seating Capacity
  - Price Per Day
  - Image
- Responsive design ensures a seamless experience across devices.



2. Car Detail Pages

- **Description**: Individual product detail pages implemented using dynamic routing in Next.js.

- **Features**:
    - Detailed fields such as:
        - Car Name
        - Description
        - Type
        - Seating Capacity
        - Transmission
        - Fuel Capacity
        - Price Per Day
        - Image’s
    - Dynamic routing for unique car detail pages.
    - Related Cars displayed based on tags or categories.

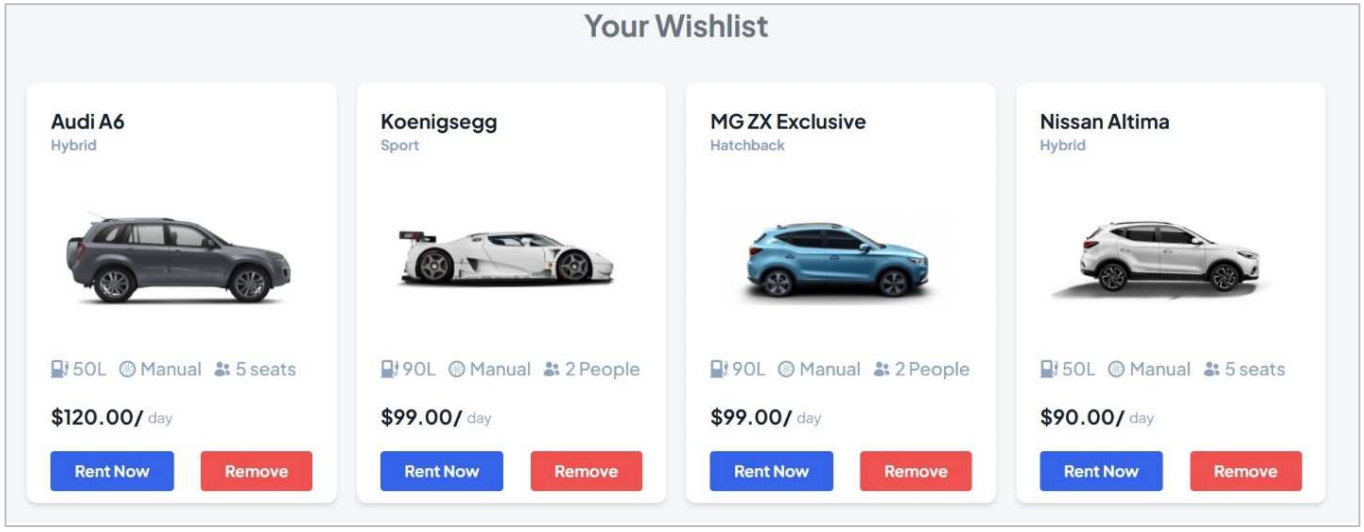


3. Wishlist Feature

• **Description:** A Wishlist feature implemented for cars in the application.

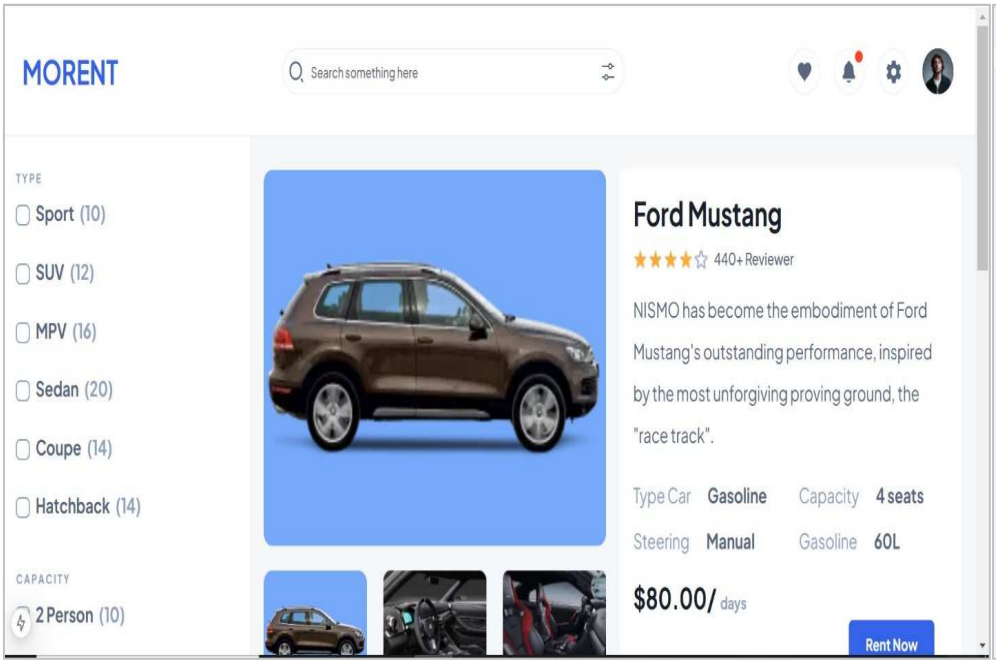
• **Features:**

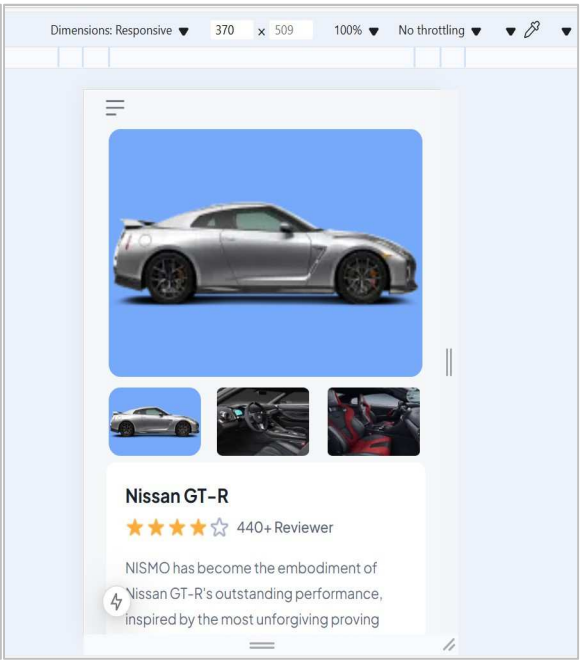
- **Car Wishlist:**
  - Users can add cars to their Wishlist by clicking on the heart icon on each car card.
  - Once added, the car appears in the Wishlist section.
- **Actions in Wishlist:**
  - Users can either rent the car directly from the Wishlist or remove it from the Wishlist.
  - These actions are performed through buttons provided on each wishlist item.
- **Dynamic Functionality:**
  - The Wishlist is dynamic, fetching data from Sanity and using local storage to maintain the list across sessions.
  - The Wishlist updates automatically as users interact with it, ensuring a seamless experience.



4. Responsive and Professional Design

- **Description**: All components styled to ensure responsiveness and a professional look across devices.
- **Features**:
    - Modular CSS for consistency and scalability.
    - Tested on multiple screen sizes to guarantee a seamless user experience.





Development Process

Step 1: Setting Up the Environment

- **Tools Used**:
  - Next.js for server-side rendering and dynamic routing.
  - Sanity CMS for content management.
  - CSS for responsive styling.
- **Process**:
  - Connected the Next.js project to Sanity CMS.
  - Verified data fetching using test API endpoints.

Step 2: Building Components

- **Components Developed**:
  - Product Listing Component: Displays car data in a grid layout.
  - Product Detail Component: Provides detailed information for each car dynamically.
  - Wishlist Component: Displays a list of favorite cars with the option to rent or remove cars from the wishlist.

Step 3: Styling and Responsiveness

- Applied modular and reusable CSS to ensure consistency.
- Tested on different devices for responsiveness and accessibility.

Step 4: Testing and Debugging

- Verified data fetching and rendering using console logs and developer tools.
- Ensured all routes and components function as expected.