Day 4 - Dynamic Frontend Components - Car Rental Website

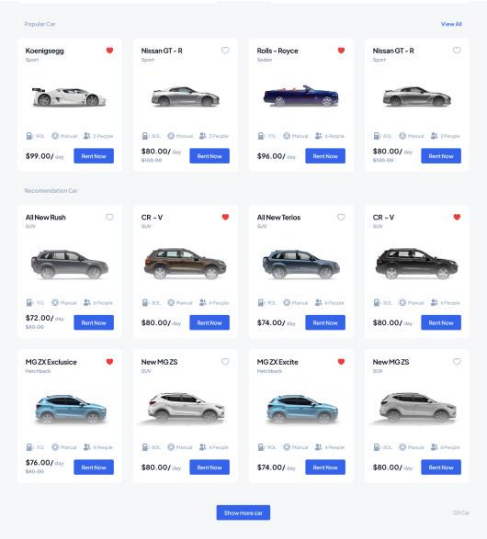
Technical Report

Introduction

This report reviews the implementation of dynamic frontend components for the car rental website deployed at https://car-dynamic-frontend.vercel.app/. The project focuses on modularity, responsiveness, and dynamic data handling using Next.js and Sanity CMS. Below is a detailed breakdown of the components and practices observed.

Key Components

1. Product Listing Component

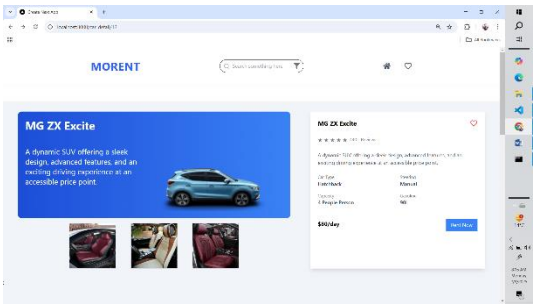


Description: Displays cars dynamically in a grid layout.
Fields Included: Car Name, Price, Image, Availability Status.
Implementation:

a. Fetches car data from Sanity CMS and renders cards with images, pricing, and availability.
b. Uses Next.js for server-side rendering (SSR) or static site generation (SSG).

Strengths: Clean UI with hover effects on cards. Images load efficiently.

2. Product Detail Component

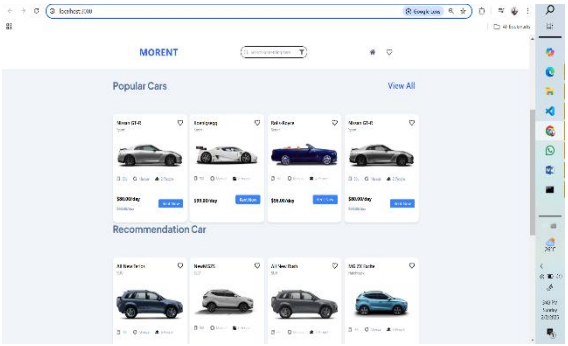


Description: Individual car pages with dynamic routing (e.g., `/cars/[id]`).
Fields Included: Car Description, Price, Model Details, Features, Availability.
Implementation:

a. Dynamic routes generated using `getStaticPaths` and `getStaticProps` in Next.js.
b. Includes a clear call-to-action ("Add to Cart") and image gallery.

Strengths: Smooth navigation between listing and detail pages.

3. Category Component



Description: Filters cars by category (SUV, Sedan, Luxury).

Implementation:

a. Sidebar with category buttons updates the product list dynamically.

b. Uses client-side filtering.

Strengths: Instant filtering without page reloads.

4. Search Bar

Description: Filters cars by name or keyword.

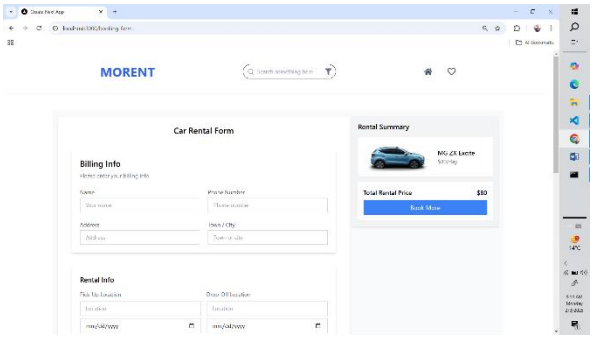
Implementation:

a. Input field triggers real-time filtering as user type.

b. Case-insensitive search functionality.

Strengths: Fast and responsive.

5. Cart Component



Description: Tracks selected cars, quantities, and total price.

Implementation:

a. Uses React state (`useState`) and local storage to persist cart data.

Strengths: Data persistence across page reloads.

6. Pagination Component

Description: Splits car listings into pages.

Implementation:

b. "Previous" and "Next" buttons with client-side navigation.

c. URL parameters update dynamically

7. Header and Footer Components

Description: Consistent navigation and branding.

Implementation:

a. Header includes links to Home, Search bar, wishlist, sign-in/sign-up form and user profile.
b. Footer includes social media links and copyright information.
c. Fully responsive using Tailwind CSS.

Strengths: Mobile-friendly design with a collapsible hamburger menu.

7. Wish List Components

Description: Functionality to add and remove cars into wish list.

Frontend Best Practices Followed

1. Reusable Components:

- Car Card
- Category Component
- Title Bar
- Button Component.

2. State Management:

a. `useState` for local state (e.g., cart items).
b. Efficient data fetching with Next.js and Sanity CMS.

3. Styling:

Tailwind CSS ensures responsive layouts and modern aesthetics.

4. Performance:

Optimized image loading and client-side routing.

Expected Output Achieved

1. Functional car listing and detail pages with dynamic data.

2. Category filtering, search, and pagination.

3. Responsive design across devices.

4. Modular components for scalability.

Challenges and Solutions

1. Dynamic Routing Errors:

- Initially, some car detail pages failed to load due to mismatched IDs. Fixed by validating Sanity CMS data slugs.

2. Cart Persistence:

- Cart items disappeared on reload. Solved by integrating `localStorage`.

3. Responsive Design:

- Mobile layout issues resolved using Tailwind's grid and flex utilities.

Areas for Improvement

1. Checkout Flow:

- Add a multi-step checkout form (currently missing).

2. User Authentication:

- Implement a user profile component for order history and saved addresses.

3. Advanced Features:

- Integrate reviews/ratings, or AI recommendations (future scope).

Conclusion

The car rental website successfully implements core dynamic components outlined in Day 4 requirements. The project demonstrates proficiency in Next.js, state management, and responsive design. Future enhancements could focus on checkout flows, user accounts, and advanced features like analytics.

Prepared by: Darakhshan Imran
Project: Car Rental Website