Hackathon 3

DAY 4 DYNAMIC FRONTEND COMPONENTS

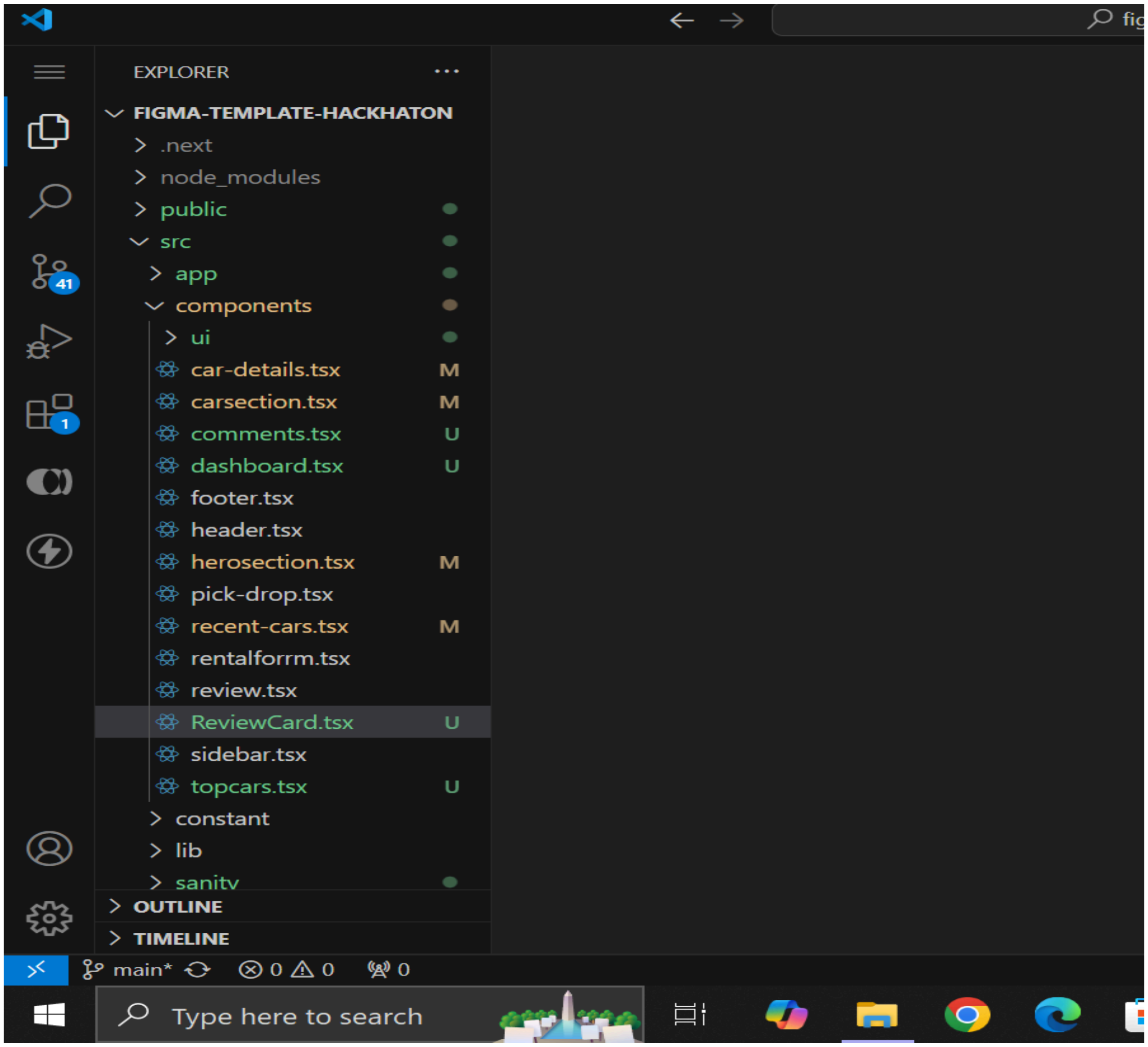
CHECKLIST OF DAY 4

1) Components Ddevelopment ✅
2)Styling and Responsive ✅
3)Code Quality ✅
4)Documentation And Submission

Objective:

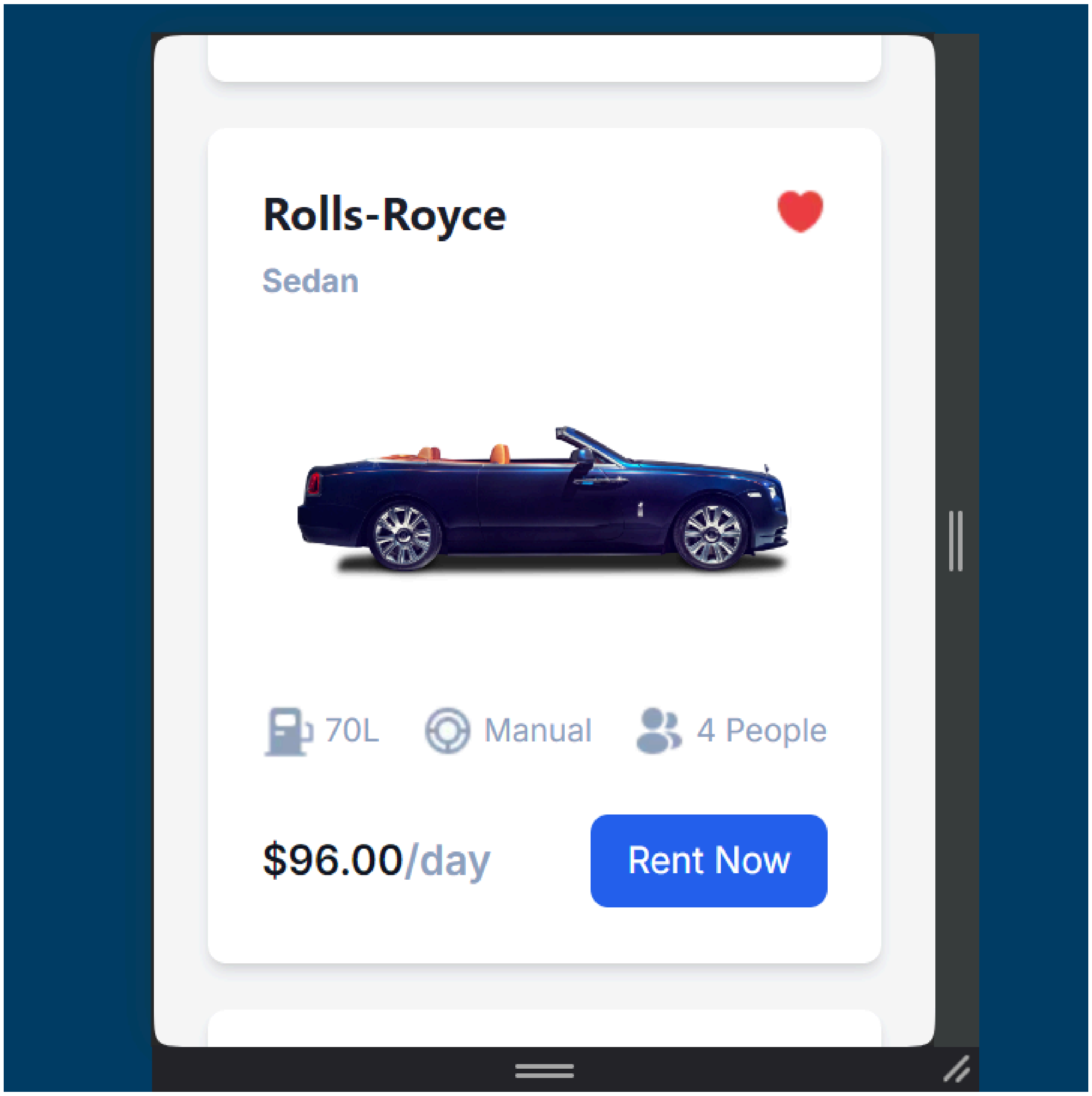
On Day 4, focus IS on designing and developing dynamic frontend components to display marketplace data fetched from Sanity CMS or APIs. This step emphasizes modular, reusable component design and real-world practices for building scalable and responsive web applications.

Reusable And modular Components



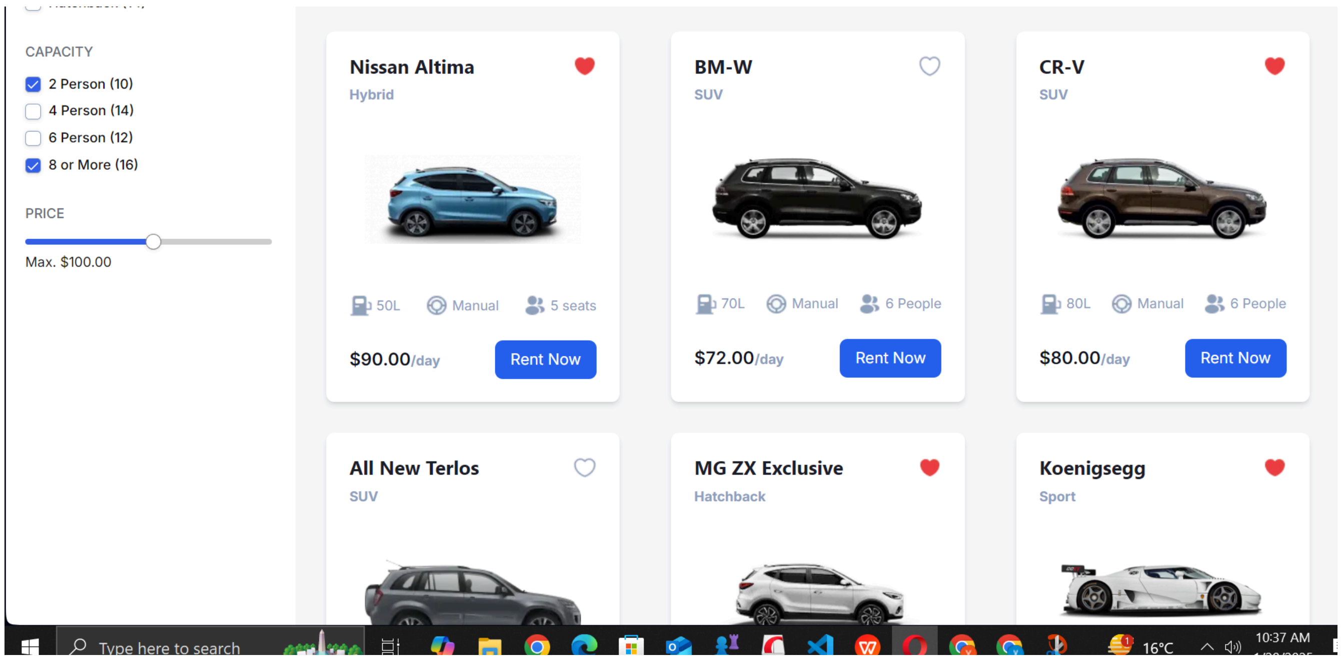
FULLY Responsive Screen

Through this process i ensure that my whole website is responsive for all devices,providing an optimal user experience for mobile and desktop screens.



Product Listing

The product listing feature dynamically displays data stored in Sanity CMS. By leveraging the flexibility of GROQ queries, this functionality ensures the product data is fetched efficiently and rendered in real-time on the frontend.



CAPACITY

- [✓] 2 Person (10)
- [ ] 4 Person (14)
- [ ] 6 Person (12)
- [✓] 8 or More (16)

PRICE

Max. $100.00

Nissan Altima
Hybrid
50L Manual 5 seats
$90.00/day Rent Now

BM-W
SUV
70L Manual 6 People
$72.00/day Rent Now

CR-V
SUV
80L Manual 6 People
$80.00/day Rent Now

All New Terlos
SUV

MG ZX Exclusive
Hatchback

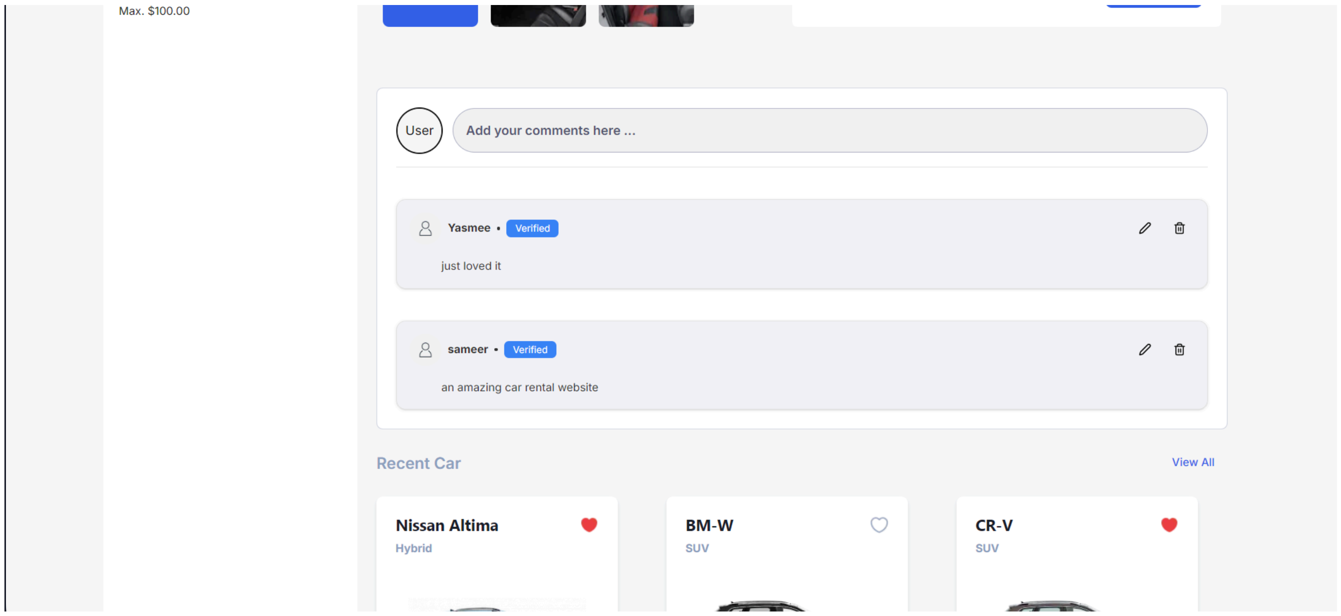
Koenigsegg
Sport

CUSTOMER'S Reviews

The comments section allows users to leave feedback, reviews, or suggestions about the rent-a-car service. This feature enhances user engagement, builds trust through shared experiences, and helps improve services based on user insights.

Comments are stored in a `data-base` or collection with fields such as:

- `id`
- `userName`
- `userEmail`
- `commentText`



Dynamic Routing

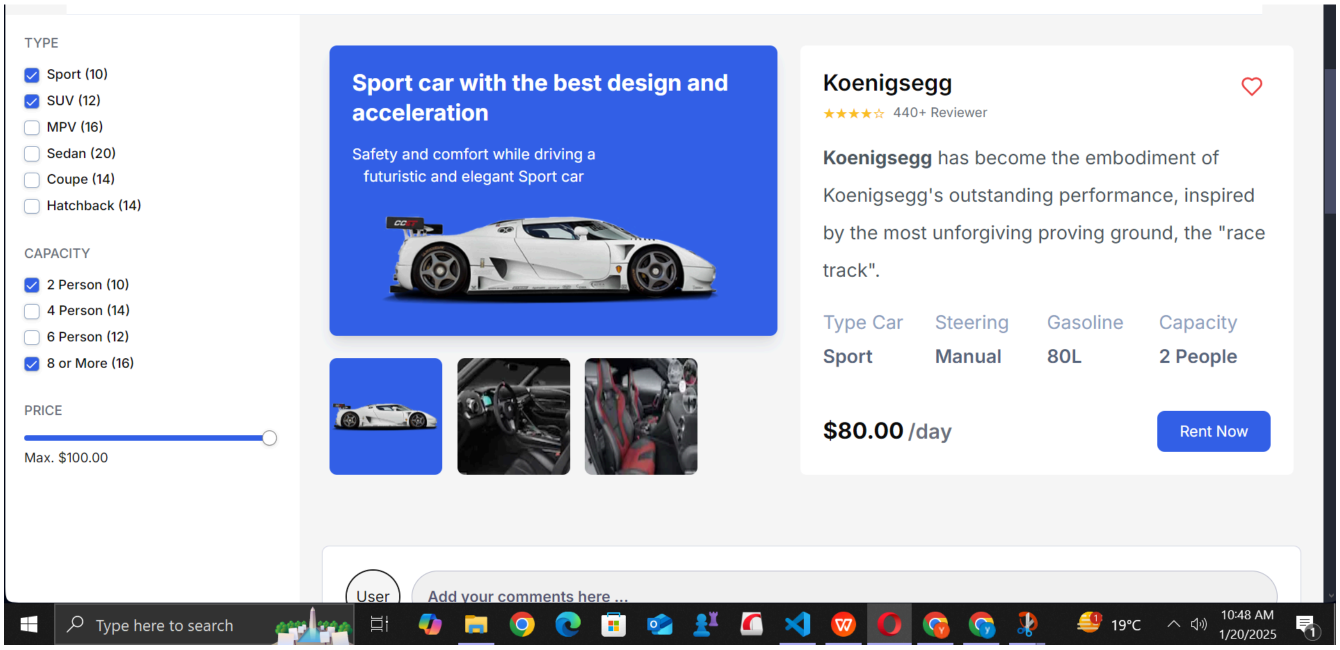
The dynamic page is designed to display content that adapts based on user interactions, parameters, or external data, providing a personalized and interactive experience. This enhances the website's usability by delivering relevant information dynamically without the need for hard-coded static pages.

Dynamic Content:

- **This page content changes based on query parameters**
- **For example, a car details page dynamically displays information based on the selected car**

Data Fetching:

- **Content is fetched from sanity using API calls.**



TYPE

- [✓] Sport (10)
- [✓] SUV (12)
- [ ] MPV (16)
- [ ] Sedan (20)
- [ ] Coupe (14)
- [ ] Hatchback (14)

CAPACITY

- [✓] 2 Person (10)
- [ ] 4 Person (14)
- [ ] 6 Person (12)
- [✓] 8 or More (16)

PRICE

Max. $100.00

Sport car with the best design and acceleration

Safety and comfort while driving a futuristic and elegant Sport car

Koenigsegg

★★★★☆ 440+ Reviewer

Koenigsegg has become the embodiment of Koenigsegg's outstanding performance, inspired by the most unforgiving proving ground, the "race track".

Type Car	Steering	Gasoline	Capacity
Sport	Manual	80L	2 People

$80.00 /day

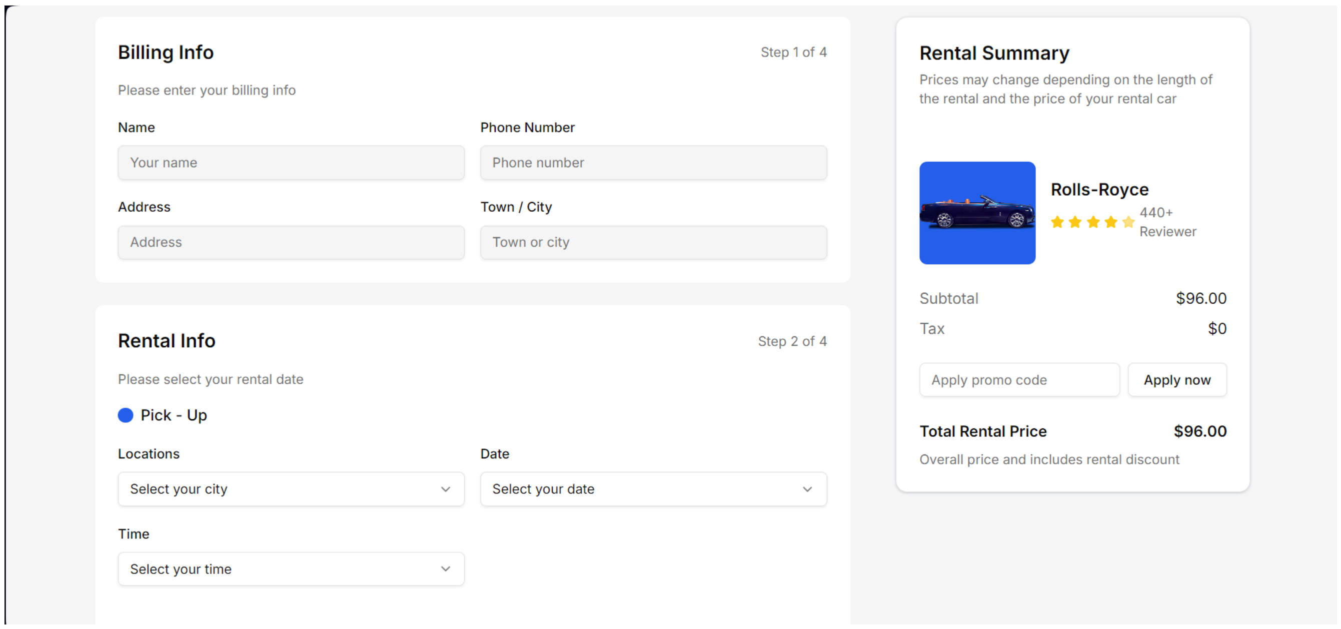
Rent Now

User

Add your comments here ...

Billing Information and Rental Summary

The Billing Information and Rental Summary sections provide users with a clear breakdown of their rental details, associated costs, and payment options. These sections ensure transparency and enhance the user experience by summarizing critical details before finalizing a transaction.



Billing Info

Step 1 of 4

Please enter your billing info

Name	Phone Number
Your name	Phone number
Address	Town / City
Address	Town or city

Rental Info

Step 2 of 4

Please select your rental date

● Pick - Up

Locations	Date
Select your city	Select your date
Time	
Select your time	

Rental Summary

Prices may change depending on the length of the rental and the price of your rental car

Rolls-Royce

440+ Reviewer

Subtotal	$96.00
Tax	$0

Apply promo code — Apply now

Total Rental Price	$96.00

Overall price and includes rental discount

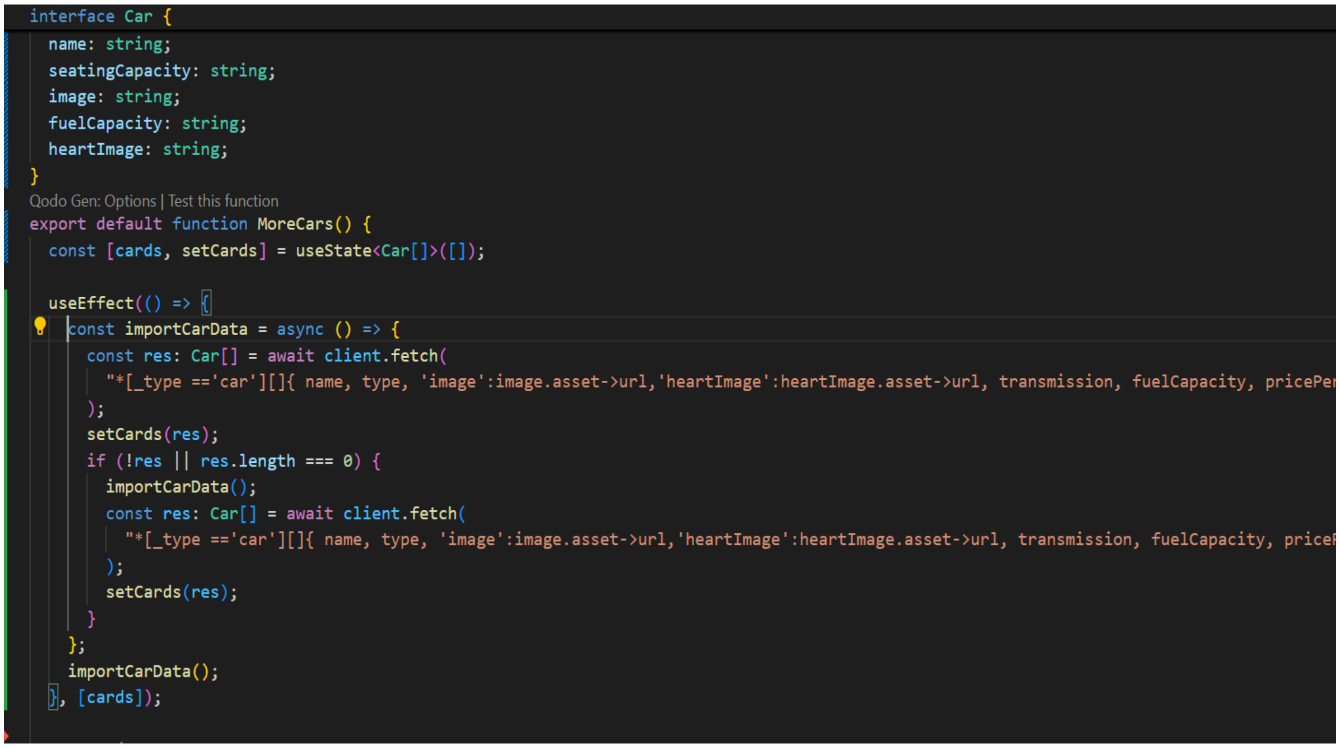
Fetching Car Data from Sanity using GROQ

The purpose of this implementation is to fetch car data stored in Sanity CMS using the GROQ query language and display it dynamically on the website. This ensures that the content is easily manageable through Sanity's CMS interface while dynamically rendering on the frontend.

Implementation Details

Setting Up State for the Data

- A React state, `cards`, is initialized to store the fetched car data.
- The `setCards` function updates this state when new data is fetched.



```
interface Car {
  name: string;
  seatingCapacity: string;
  image: string;
  fuelCapacity: string;
  heartImage: string;
}
Qodo Gen: Options | Test this function
export default function MoreCars() {
  const [cards, setCards] = useState<Car[]>([]);

  useEffect(() => {
    const importCarData = async () => {
      const res: Car[] = await client.fetch(
        "*[_type =='car'][]{ name, type, 'image':image.asset->url,'heartImage':heartImage.asset->url, transmission, fuelCapacity, pricePer
      );
      setCards(res);
      if (!res || res.length === 0) {
        importCarData();
        const res: Car[] = await client.fetch(
          "*[_type =='car'][]{ name, type, 'image':image.asset->url,'heartImage':heartImage.asset->url, transmission, fuelCapacity, priceP
        );
        setCards(res);
      }
    };
    importCarData();
  }, [cards]);
```

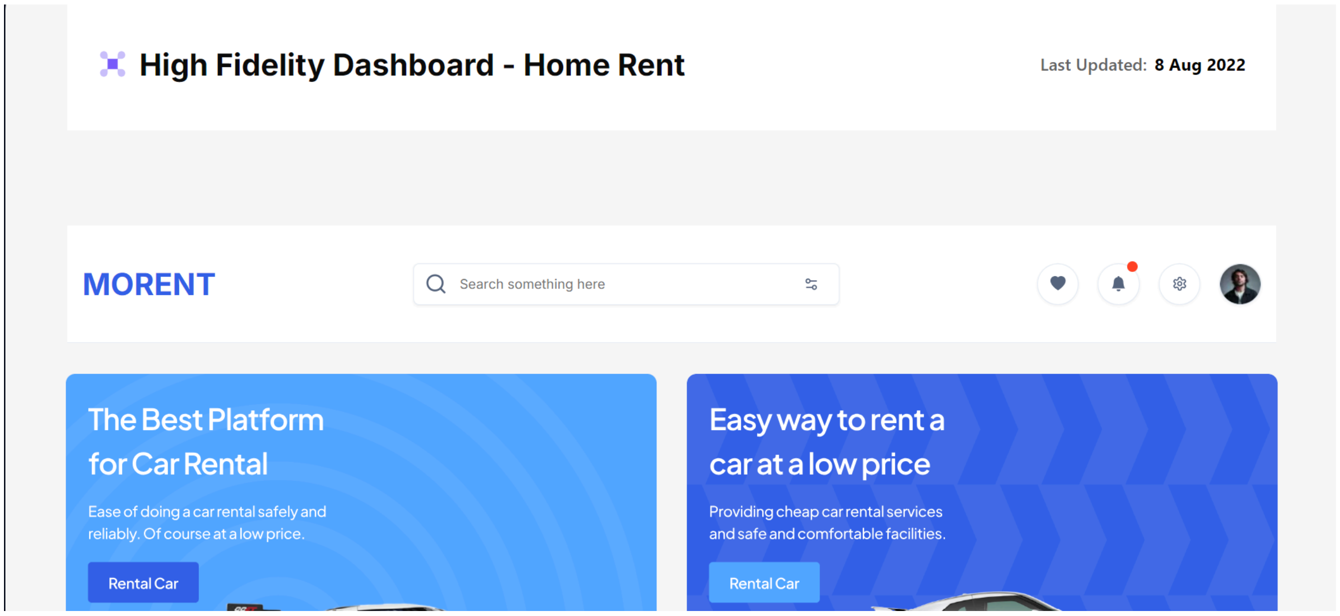
Key Concepts Used

- GROQ Query Languag: Used to fetch structured data from Sanity CMS.

- Sanity Client: The `client.fetch` method executes the GROQ query and returns the result.
- React State and Hooks:
  - `useState` manages the fetched data.
  - `useEffect` triggers the data fetching process when necessary.
- Asynchronous Programming: Ensures data fetching occurs without blocking the UI.

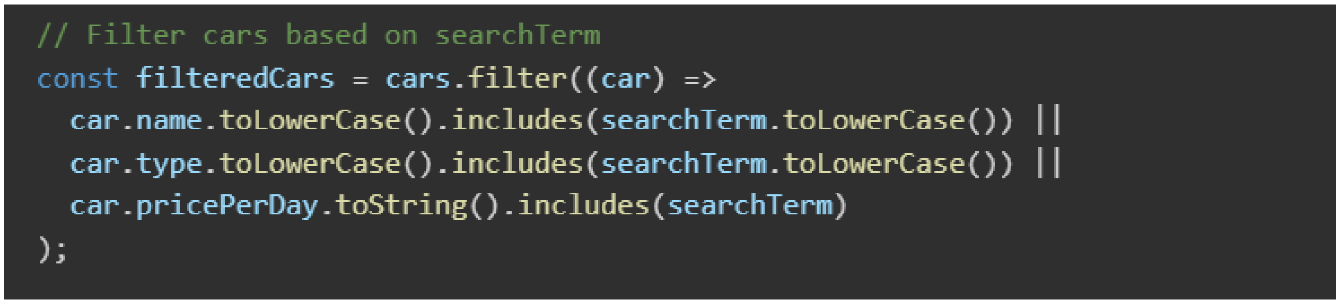
Searchbar

The search bar provides users with the ability to quickly find specific items, such as cars, within the application. It enhances user experience by offering a fast and intuitive way to filter and locate content.


High Fidelity Dashboard - Home Rent
Last Updated: 8 Aug 2022
MORENT
Search something here
The Best Platform for Car Rental
Ease of doing a car rental safely and reliably. Of course at a low price.
Rental Car
Easy way to rent a car at a low price
Providing cheap car rental services and safe and comfortable facilities.
Rental Car

Functionality

1. User Input:
   - Users can type keywords or phrases into the search bar to find matching items.
   - The search is case-insensitive to ensure a seamless experience.



```
// Filter cars based on searchTerm
const filteredCars = cars.filter((car) =>
  car.name.toLowerCase().includes(searchTerm.toLowerCase()) ||
  car.type.toLowerCase().includes(searchTerm.toLowerCase()) ||
  car.pricePerDay.toString().includes(searchTerm)
);
```

CONCLUSION:

I have successfully implemented key features that enhance the functionality, user experience, and scalability of the application.

Improved Technical Proficiency:

- Strengthened my expertise in modern web development tools and frameworks such as Next.js AND Sanity CMS.
- Gained a deeper understanding of API integration, dynamic routing, and state management.

Enhanced User-Centric Design:

- Focused on creating interactive and responsive features like the search bar and comments section to ensure a positive user experience.

This project not only enriched my technical knowledge but also reinforced the importance of user-centric design and modular development practices. It serves as a robust foundation for future enhancements and showcases my ability to build dynamic, scalable, and interactive web applications.

BY YASMEEN NAZEER