Name: **ABDULLAH**

Roll Number: **00433137**

Class : **Friday (7 pm to 10 pm)**

Hakathon 3 day 3 : Car Rental-Commerce Platform Report

Overview:

The primary focus of Day 3 was on integrating APIs, specifically with Sanity CMS, using GROQ queries, and updating the schema to support dynamic data requirements. Here's a summary of the tasks accomplished:

1. API Integration Process

Process Overview:

1. Identify the required APIs for the platform functionality (e.g., car data).
2. Configure API keys and endpoints in the application.
3. Set up HTTP requests using Fetch for seamless data transfer between the backend and APIs.
4. Implement error handling mechanisms for API calls.

Steps Taken:

- Integrated APIs for **car listings**.
- Ensured secure API requests with proper authentication tokens.
- Debugged and tested API calls in Postman for reliability.

2. Adjustments Made to Schemas

Schema Adjustments for Integration:

Car Schema:

The **Car Schema** was developed with the following fields to match project requirements and enable API integration.

Car Schema Details:

```
export default {
  name: 'car',
  type: 'document',
  title: 'Car',
  fields: [
    {
      name: 'name',
      type: 'string',
      title: 'Car Name',
    },
    {
      name: 'brand',
      type: 'string',
      title: 'Brand',
      description: 'Brand of the car (e.g., Nissan, Tesla, etc.)',
    },
    {
      name: 'type',
      type: 'string',
      title: 'Car Type',
      description: 'Type of the car (e.g., Sport, Sedan, SUV, etc.)',
    },
    {
      name: 'fuelCapacity',
      type: 'string',
      title: 'Fuel Capacity',
      description: 'Fuel capacity or battery capacity (e.g., 90L, 100kWh)',
    },
    {
      name: 'transmission',
      type: 'string',
      title: 'Transmission',
      description: 'Type of transmission (e.g., Manual, Automatic)',
    },
    {
      name: 'seatingCapacity',
      type: 'string',
      title: 'Seating Capacity',
      description: 'Number of seats (e.g., 2 People, 4 seats)',
    },
    {
      name: 'pricePerDay',
      type: 'string',
      title: 'Price Per Day',
      description: 'Rental price per day',
    },
    {
      name: 'originalPrice',
      type: 'string',
      title: 'Original Price',
      description: 'Original price before discount (if applicable)',
    },
    {
      name: 'tags',
      type: 'array',
      title: 'Tags',
      of: [{ type: 'string' }],
      options: {
        layout: 'tags',
      },
      description: 'Tags for categorization (e.g., popular, recommended)',
    },
    {
      name: 'image',
      type: 'image',
      title: 'Car Image',
      options: {
        hotspot: true
      }
    }
  ],
};
```

Old Schema:

The old schema contained the following fields:

1. **Car ID**: A unique identifier for each car.
2. **Brand**: The brand of the car.
3. **Model**: The model name or number.
4. **Year**: The year of manufacture.
5. **Category**: Classification of the car (e.g., SUV, Sedan).
6. **Pricing**: The cost or rental price.
7. **Availability**: Whether the car is available for rent.
8. **Features**: Additional features of the car (e.g., GPS, Bluetooth).

New Schema:

The new schema expands on the old one, with these fields:

1. **Car Name** (name): Combines brand and model for easier identification.
2. **Brand** (brand): Retained for brand-specific filtering.
3. **Car Type** (type): Similar to category but uses more user-friendly labels like "Sport" or "Luxury."
4. **Fuel Capacity** (fuelCapacity): Added for specifying the car’s fuel or battery capacity.
5. **Transmission** (transmission): Indicates the type of gearbox (Manual/Automatic).
6. **Seating Capacity** (seatingCapacity): Number of seats in the car.
7. **Price Per Day** (pricePerDay): Added for rental calculations based on a daily rate.

8. **Tags** (tags): Helps categorize cars (e.g., "popular" or "luxury").
9. **Image** (image): Allows attaching an image for better frontend visualization.

3. Data Migration:

Sanity CMS Setup:

- Installed the `@sanity/client` library to enable communication with Sanity's API.

Configured Sanity Client:

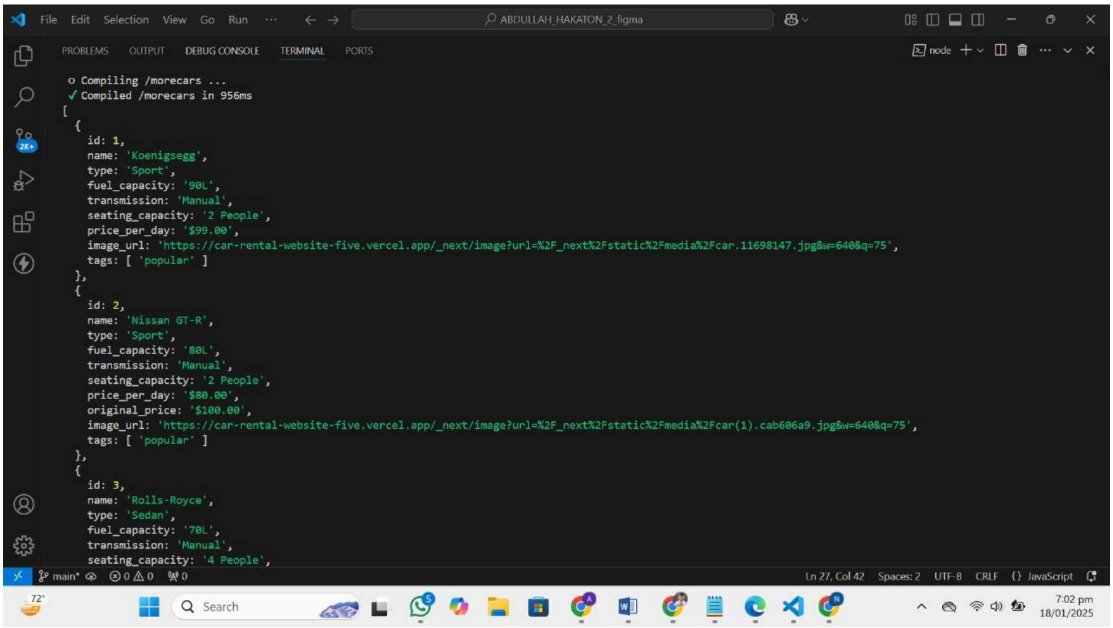
- Configured the Sanity client using essential project details like **Project ID**, **Dataset**, and **API Version**, ensuring a secure and direct connection to the Sanity CMS backend.

Custom Schemas:

- **Car Details Schema**: Designed to store car-related data, including fields for car name, type, price per day, availability, and an image URL.

Data Fetching and Rendering:

- **API Fetch**: Data stored in Sanity CMS is fetched through its API using the configured client.
- **Real-Time Updates**: The platform dynamically fetches and renders updated data whenever there are changes in Sanity CMS. This ensures that car details, customer information, and booking records on the platform always reflect the latest state.
- **Frontend Integration**: The fetched data is rendered on the platform’s frontend, ensuring a seamless and updated user experience for browsing cars, viewing customer records, or managing bookings.



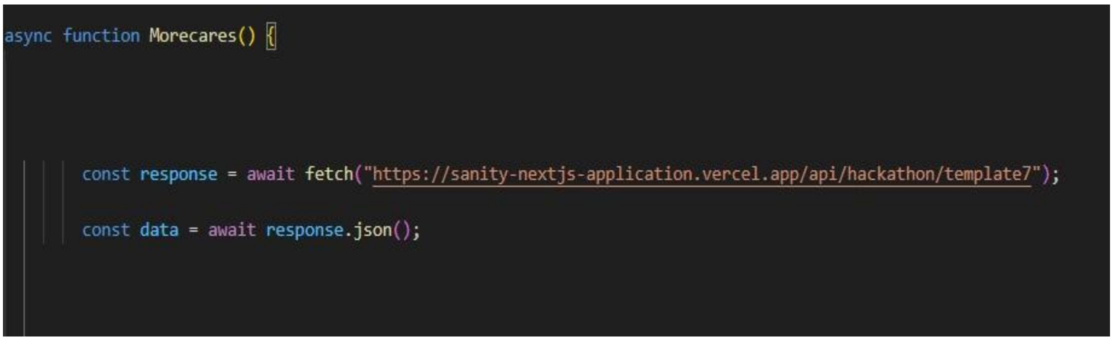
```
 o Compiling /morecars ...
 ✓ Compiled /morecars in 956ms
[
  {
    id: 1,
    name: 'Koenigsegg',
    type: 'Sport',
    fuel_capacity: '90L',
    transmission: 'Manual',
    seating_capacity: '2 People',
    price_per_day: '$99.00',
    image_url: 'https://car-rental-website-five.vercel.app/_next/image?url=%2F_next%2Fstatic%2Fmedia%2Fcar.11698147.jpg&w=640&q=75',
    tags: [ 'popular' ]
  },
  {
    id: 2,
    name: 'Nissan GT-R',
    type: 'Sport',
    fuel_capacity: '80L',
    transmission: 'Manual',
    seating_capacity: '2 People',
    price_per_day: '$80.00',
    original_price: '$100.00',
    image_url: 'https://car-rental-website-five.vercel.app/_next/image?url=%2F_next%2Fstatic%2Fmedia%2Fcar(1).cab606a9.jpg&w=640&q=75',
    tags: [ 'popular' ]
  },
  {
    id: 3,
    name: 'Rolls-Royce',
    type: 'Sedan',
    fuel_capacity: '70L',
    transmission: 'Manual',
    seating_capacity: '4 People',
```

Why Sanity CMS is Ideal for This Project:

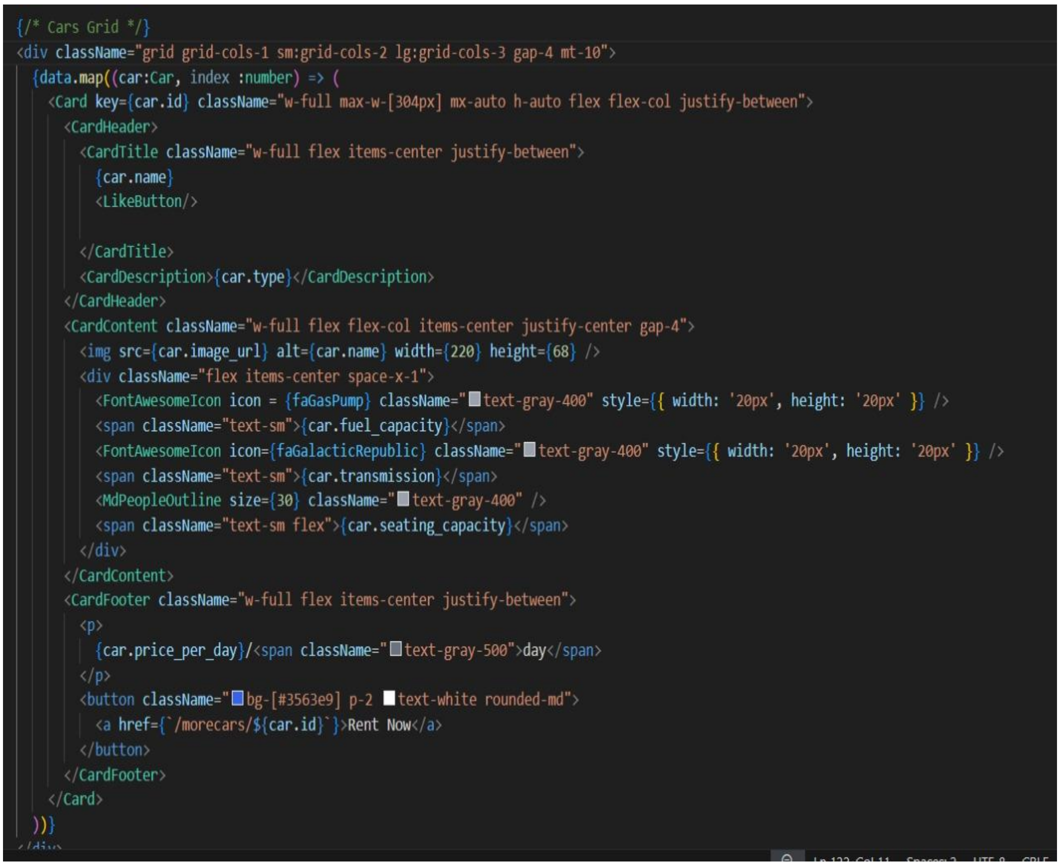
- **Centralized Data Management**: All information is managed and updated in one place, streamlining data handling.
- **Real-Time Sync**: Any changes in car availability, pricing, or bookings in Sanity CMS are instantly reflected on the platform via API calls.
- **Customizable Content Models**: Created tailored schemas to meet the unique requirements of a car rental platform.
- **Optimized Performance**: Sanity's built-in features, like image optimization, ensure faster loading times for car images.
- **Scalability**: The platform is built to handle growth, accommodating increasing data and user interactions efficiently.

Screenshots

1. Api calling :

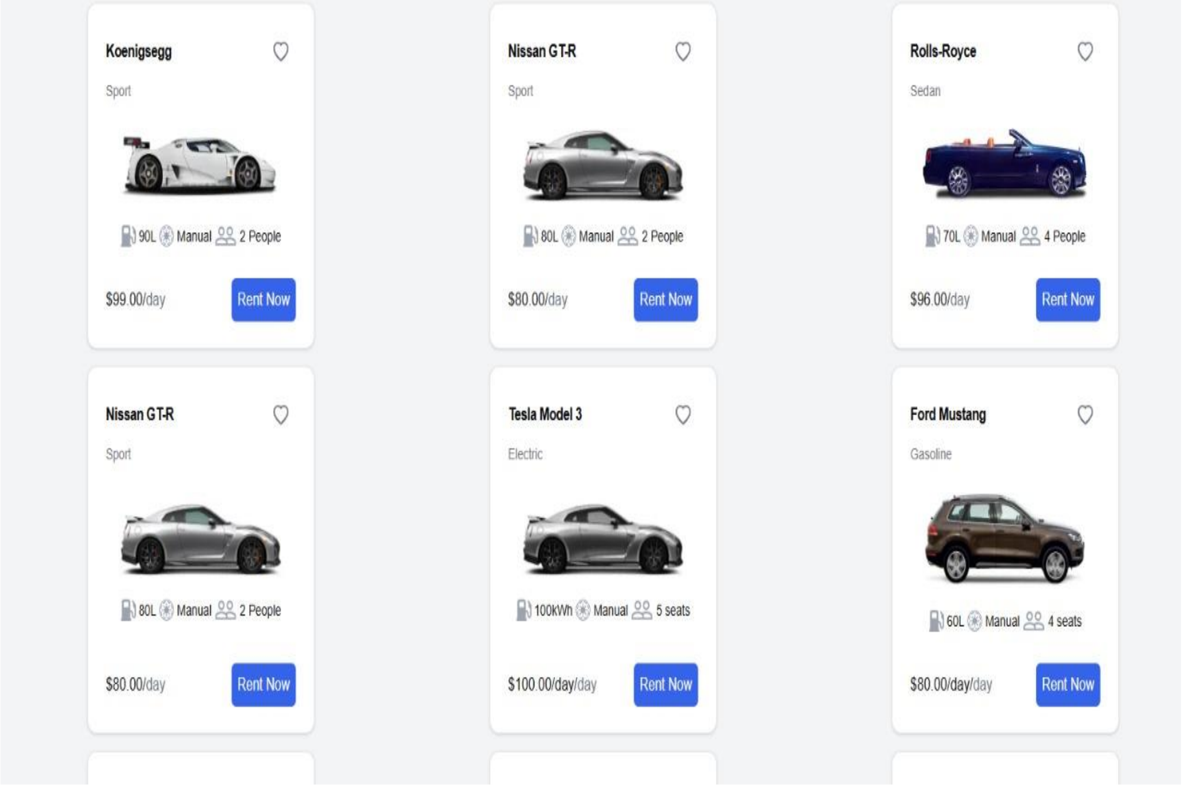


```
async function Morecares() {



    const response = await fetch("https://sanity-nextjs-application.vercel.app/api/hackathon/template7");

    const data = await response.json();
```



```
{/* Cars Grid */}
<div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-4 mt-10">
  {data.map((car:Car, index :number) => (
    <Card key={car.id} className="w-full max-w-[304px] mx-auto h-auto flex flex-col justify-between">
      <CardHeader>
        <CardTitle className="w-full flex items-center justify-between">
          {car.name}
          <LikeButton/>

        </CardTitle>
        <CardDescription>{car.type}</CardDescription>
      </CardHeader>
      <CardContent className="w-full flex flex-col items-center justify-center gap-4">
        <img src={car.image_url} alt={car.name} width={220} height={68} />
        <div className="flex items-center space-x-1">
          <FontAwesomeIcon icon = {faGasPump} className="text-gray-400" style={{ width: '20px', height: '20px' }} />
          <span className="text-sm">{car.fuel_capacity}</span>
          <FontAwesomeIcon icon={faGalacticRepublic} className="text-gray-400" style={{ width: '20px', height: '20px' }} />
          <span className="text-sm">{car.transmission}</span>
          <MdPeopleOutline size={30} className="text-gray-400" />
          <span className="text-sm flex">{car.seating_capacity}</span>
        </div>
      </CardContent>
      <CardFooter className="w-full flex items-center justify-between">
        <p>
          {car.price_per_day}/<span className="text-gray-500">day</span>
        </p>
        <button className="bg-[#3563e9] p-2 text-white rounded-md">
          <a href={`/morecars/${car.id}`}>Rent Now</a>
        </button>
      </CardFooter>
    </Card>
  ))}
</div>
```

2. Data successfully displayed in the frontend :



Koenigsegg
Sport
90L Manual 2 People
$99.00/day
Rent Now

Nissan GT-R
Sport
80L Manual 2 People
$80.00/day
Rent Now

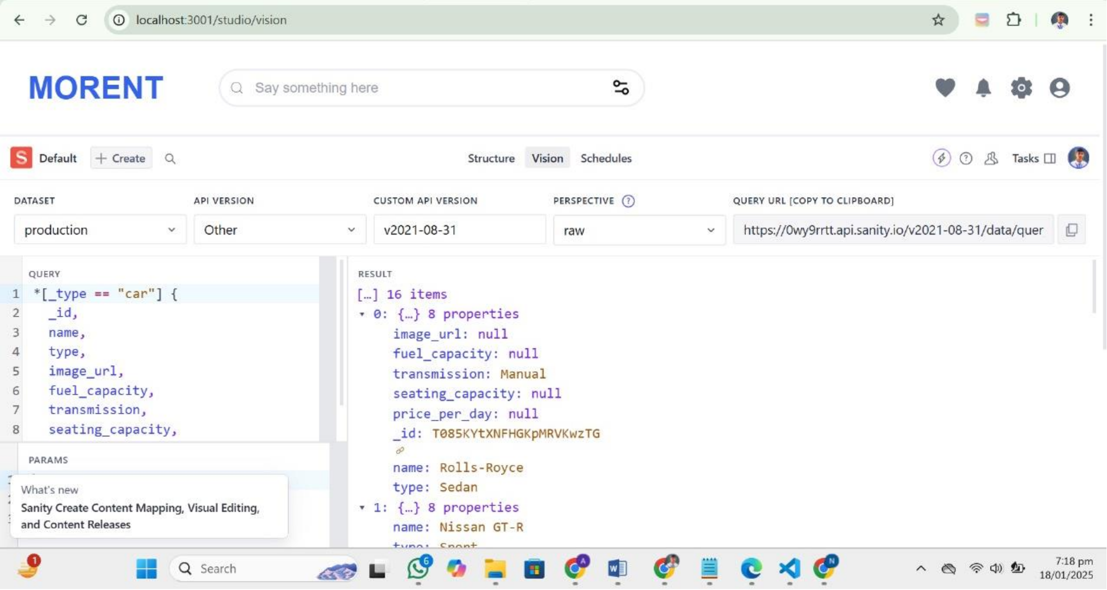
Rolls-Royce
Sedan
70L Manual 4 People
$96.00/day
Rent Now

Nissan GT-R
Sport
80L Manual 2 People
$80.00/day
Rent Now

Tesla Model 3
Electric
100kWh Manual 5 seats
$100.00/day/day
Rent Now

Ford Mustang
Gasoline
60L Manual 4 seats
$80.00/day/day
Rent Now

Populated Sanity CMS fields :



localhost:3001/studio/vision

MORENT

Say something here

Default · Create · Structure · Vision · Schedules · Tasks

DATASET: production
API VERSION: Other
CUSTOM API VERSION: v2021-08-31
PERSPECTIVE: raw
QUERY URL [COPY TO CLIPBOARD]: https://0wy9rrtt.api.sanity.io/v2021-08-31/data/quer

QUERY

```
*[_type == "car"] {
  _id,
  name,
  type,
  image_url,
  fuel_capacity,
  transmission,
  seating_capacity,
```

PARAMS

What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases

RESULT

```
[…] 16 items
▾ 0: {…} 8 properties
    image_url: null
    fuel_capacity: null
    transmission: Manual
    seating_capacity: null
    price_per_day: null
    _id: T085KYtXNFHGKpMRVKwzTG
    name: Rolls-Royce
    type: Sedan
▾ 1: {…} 8 properties
    name: Nissan GT-R
    type: Sport
```

7:18 pm
18/01/2025

Day 3 Checklist:

- Self-Validation Checklist: API Understanding: □ ✓
- Schema Validation: □ ✓
- Data Migration: □ ✓
- API Integration in Next.js: □ ✓
- Submission Preparation : ✓