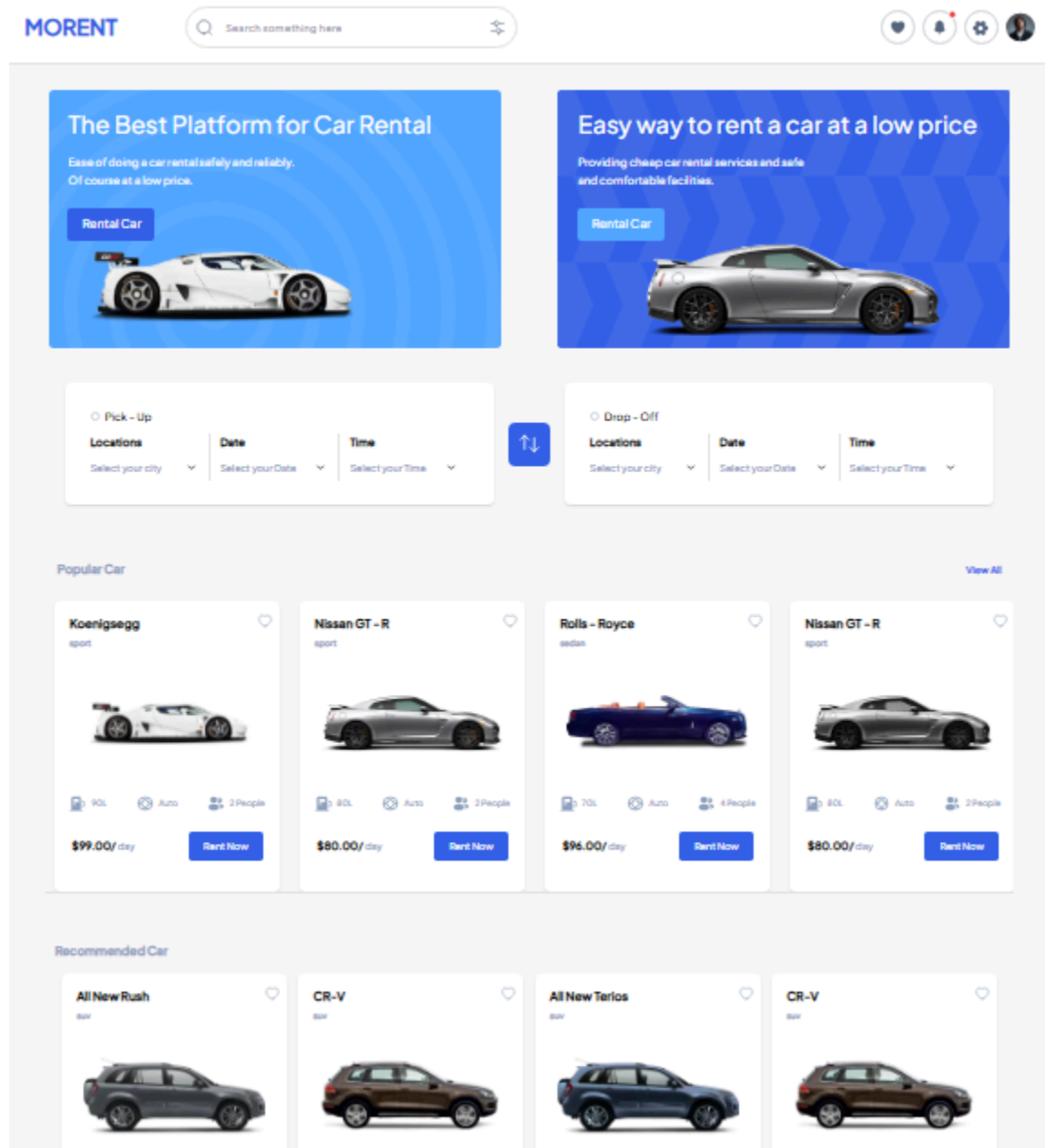


Day 3 - API Integration in Rent E-commerce website :

Platform Overview:

The platform you're creating will be a local rent-a-car service, primarily focused on providing immediate access to vehicles for local users. The service is intended to allow individuals in a neighborhood to rent their vehicles and provide users with quick access to cars as per their requirements.



Data Store in Sanity (CMS) Considered as a Database

Structure

Vision


Schedules


RecommendedCar


+ ...


New MG ZS


Q Search list


 All New Rush


 CR-V


 All New Terios

 CR-V

 MG ZX Excluisce

 New MG ZS

 MG ZX Excite

 New MG ZS

RecommendedCar

New MG ZS

Brand

...

New MG ZS

Slug


new-mg-zs2

Generate

Category

SUV

Image



Data Schema Structure

```
1  export const card = {
2    name: 'card',
3    title: 'Card',
4    type: 'document',
5    fields: [
6      {
7        name: 'brand',
8        title: 'Brand',
9        type: 'string',
10     },
11     {
12       name: 'slug',
13       title: 'Slug',
14       type: 'slug',
15       options: {
16         source: 'brand',
17         maxLength: 96,
18       },
19     },
20     {
21       name: 'category',
22       title: 'Category',
23       type: 'string',
24       options: {
25         list: [
26           { title: 'Sport', value: 'sport' },
27           { title: 'Sedan', value: 'sedan' },
28           { title: 'SUV', value: 'suv' },
29           { title: 'Hatchback', value: 'hatchback' },
30         ],
31       },
32     },
33     {
34       name: 'image',
35       title: 'Image',
36       type: 'image',
37     },
38     {
39       name: 'price',
40       title: 'Price',
41       type: 'number',
42     },
43     {
44       name: 'liters',
45       title: 'Liters',
46       type: 'number',
47     },
48     {
49       name: 'auto',
```

Groq Query for Data Fetching

```
src > sanity > lib > JS fetchData.js > [x] FetchCarsData
1  import { client } from './client';
2
3  //Popular Car Data
4  export const FetchCarsData = async () => {
5    try {
6      const query = `*[_type == "card"] | order(_createdAt desc) {
7        _id,
8        brand,
9        category,
10       price,
11       liters,
12       auto,
13       seats,
14       "slug": slug.current,
15       "imageUrl": image.asset->url
16     }`;
17
18     const data = await client.fetch(query, { cache: 'no-store' });
19     return data;
20   } catch (error) {
21     console.error('Error fetching from Sanity:', error);
22     throw error;
23   }
24 };
25
26 //single car data
27 export const fetchSingleCarData = async (slug) => {
28   const query = `*[_type == "card" && slug.current == ${slug}][0]{
29     _id,
30     brand,
31     category,
32     price,
33     liters,
34     auto,
35     seats,
36     "slug": slug.current,
37     "imageUrl": image.asset->url,
38     description
39   }`;
40
41   return client.fetch(query, { slug, cache: 'no-store' });
42 };
43
```