Building Dynamic Frontend Components For Marketplace:-

Built Components:

1. Product Listing Component:

• Dynamically fetch and display product data from Sanity CMS or API.

```
"use client"
  import Image from "next/image"
  import Link from "next/link"
  import { useRouter } from "next/navigation"
  import { useEffect, useState } from "react"
  import { useCart, SelectedCar } from "@/app/context/cartCont"
  import { Heart } from "lucide-react"
  import { client } from "@/sanity/lib/client"
1 // Define the Car interface based on Sanity schema
2. interface Car {
3 _id: string
4 name: string
  pricePerDay: number
   type: string
   fuelCapacity: number
8 transmission: string
   seatingCapacity: number
    imageUrl: string
3. // Define the params type
4 interface Params {
    params: {
       id: string
```

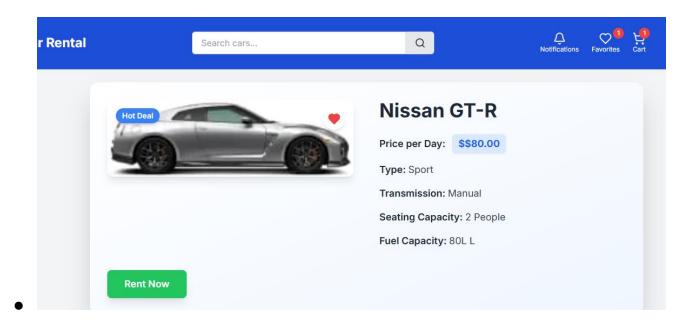
```
0.// Fetch car details function
1. async function getServerSideProps( id: string): Promise<Car> {
    const response = await client.fetch(
       `*[_type == "car" && _id == $_id][0]{
        _id,
        name,
        pricePerDay,
        type,
        fuelCapacity,
        transmission,
        seatingCapacity,
        "imageUrl": image.asset->url
      { _id },
    return response
18 export default function CarDetails({ params }: Params) {
19     const { _id } = params
const [car, setCar] = useState<Car | null>(null)
   const [isLoading, setIsLoading] = useState(true)
   const [error, setError] = useState<string | null>(null)
const router = useRouter()
    const { addToFavorites, removeFromFavorites, isFavorite, setSelectedCar }
= useCart()
    useEffect(() => {
       async function fetchCarDetails() {
        try {
          const carData = await getServerSideProps(_id)
          setCar(carData)
          setIsLoading(false)
        } catch (err) {
          console.error("Error fetching car details:", err)
           setError("Failed to fetch car details. Please try again later.")
```

```
setIsLoading(false)
  fetchCarDetails()
}, [_id])
const handleFavoriteToggle = () => {
  if (!car) return
  if (isFavorite(car. id)) {
   removeFromFavorites(car._id)
  } else {
   addToFavorites({
      id: car. id,
     title: car.name,
      price: car.pricePerDay,
    })
const handleRentClick = () => {
  if (car) {
   setSelectedCar({
      id: car._id,
     title: car.name,
      price: car.pricePerDay,
      images: [car.imageUrl],
      desc: car.type,
    router.push("/cart")
if (isLoading) {
        return <div>Loading...</div>
      if (error || !car) {
        return (
          <div className="min-h-screen bg-gray-100">
            <main className="max-w-7xl mx-auto py-6 sm:px-6 lg:px-8">
```

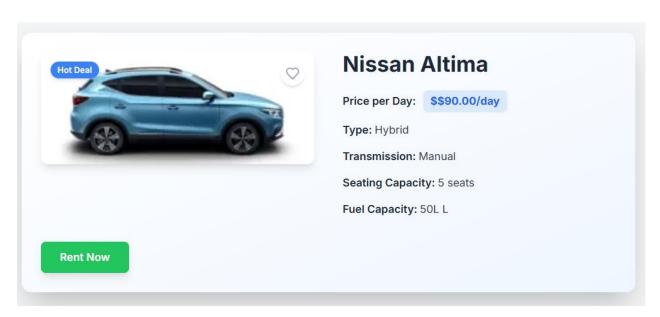
```
<div className="max-w-5xl mx-auto p-8 bg-red-50 rounded-2xl</pre>
shadow-lg">
                     <h2 className="text-2x1 font-bold text-red-</pre>
600">Error</h2>
                     {error | Sorry, we couldn't fetch the details for
this car. Please try again later."}
                     <Link href="/cars">
                        <button className="mt-6 bg-blue-500 hover:bg-blue-600</pre>
text-white py-2 px-4 rounded-lg text-sm font-semibold shadow-lg transition-all
duration-300">
                         Back to Car List
                       </button>
                     </Link>
                   </div>
                 </main>
               </div>
           return (
             <div className="min-h-screen bg-gray-100">
               <main className="max-w-7xl mx-auto py-6 sm:px-6 lg:px-8">
                 <div className="max-w-5xl mx-auto p-8 bg-gradient-to-br from-</pre>
white to-blue-50 rounded-2xl shadow-lg transition-shadow duration-300
hover:shadow-2x1">
                   <div className="flex flex-col lg:flex-row items-center</pre>
lg:items-start lg:gap-12">
                     {/* Car Image */}
                     <div className="w-full lg:w-1/2 group relative">
                        <div className="relative overflow-hidden rounded-xl</pre>
shadow-lg transition-transform duration-300 hover:scale-105">
                          <Image
                            src={car.imageUrl || "/placeholder.svg"}
                            alt={car.name}
                           height={600}
                           width={600}
                            className="rounded-xl group-hover:brightness-110
group-hover:blur-0"
```

```
{/* Subtle Overlay */}
                         <div className="absolute inset-0 bg-gradient-to-t</pre>
from-black/20 to-transparent opacity-0 group-hover:opacity-100 transition-
opacity duration-300"></div>
                         {/* Floating Badge */}
                         <span className="absolute top-4 left-4 bg-blue-500"</pre>
text-white text-sm font-semibold py-1 px-3 rounded-full shadow-md">
                          Hot Deal
                         </span>
                       </div>
                       {/* Favorite Heart Icon */}
                       <button
                         onClick={handleFavoriteToggle}
                         className="absolute top-4 right-4 z-10 p-2 bg-white
rounded-full shadow-md transition-colors duration-300"
                         <Heart className={`w-6 h-6 ${isFavorite(car._id) ?</pre>
"fill-red-500 text-red-500" : "text-gray-400"}`} />
                       </button>
                     </div>
                     {/* Car Details */}
                     <div className="w-full lg:w-1/2 text-center lg:text-</pre>
left">
                       <h2 className="text-4xl font-bold text-gray-800 mt-6</pre>
lg:mt-0 transition-colors duration-300 hover:text-blue-600">
                         {car.name}
                       </h2>
                       <div className="mt-6 space-y-4 text-lg">
                         <span className="font-semibold text-gray-800">Price
per Day:</span>
                           <span className="bg-blue-100 text-blue-600 font-</pre>
bold py-1 px-3 rounded-lg shadow-sm">
                             ${car.pricePerDay}
                           </span>
                         <span className="font-semibold text-gray-800">Type:
</span>
```

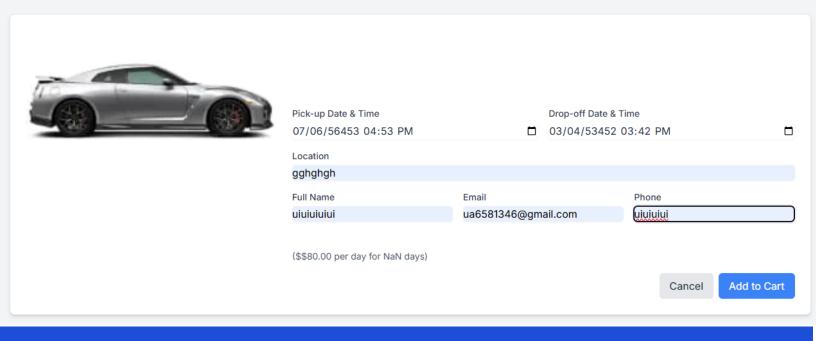
```
{car.type}
                       <span className="font-semibold text-gray-</pre>
800">Transmission: </span>
                         {car.transmission}
                       <span className="font-semibold text-gray-</pre>
800">Seating Capacity: </span>
                         {car.seatingCapacity}
                       <span className="font-semibold text-gray-800">Fuel
Capacity: </span>
                         {car.fuelCapacity} L
                       </div>
                   </div>
                 </div>
                 <div className="flex justify-center lg:justify-start mt-</pre>
10">
                   <button
                     onClick={handleRentClick}
                     className="bg-green-500 hover:bg-green-600 text-white
py-3 px-8 rounded-lg text-lg font-semibold shadow-lg transition-all duration-
300 transform hover:-translate-y-1 hover:shadow-xl focus:outline-none
focus:ring-4 focus:ring-green-300"
                     Rent Now
                   </button>
                 </div>
                </div>
              </main>
            </div>
```



• Provide an "Add to Cart" button with functionality.



Your Cart



Car Rental

Search cars...









Your Cart

Daily Rate: \$\$80.00 Number of Days: NaN

Pick-up: Invalid Date Drop-off: Invalid Date Location: gghghgh Name: uiuiuiuiui

Email: rarain@897gmail.com

Phone: 655544

Remove from Cart









Checkout

Review Your Order

Nissan GT-R

Daily Rate: \$\$80.00 Number of Days: 2 Total Price: \$NaN

Pick-up: 1/26/2025, 7:08:00 PM Drop-off: 1/28/2025, 2:35:00 PM

Location: karachi