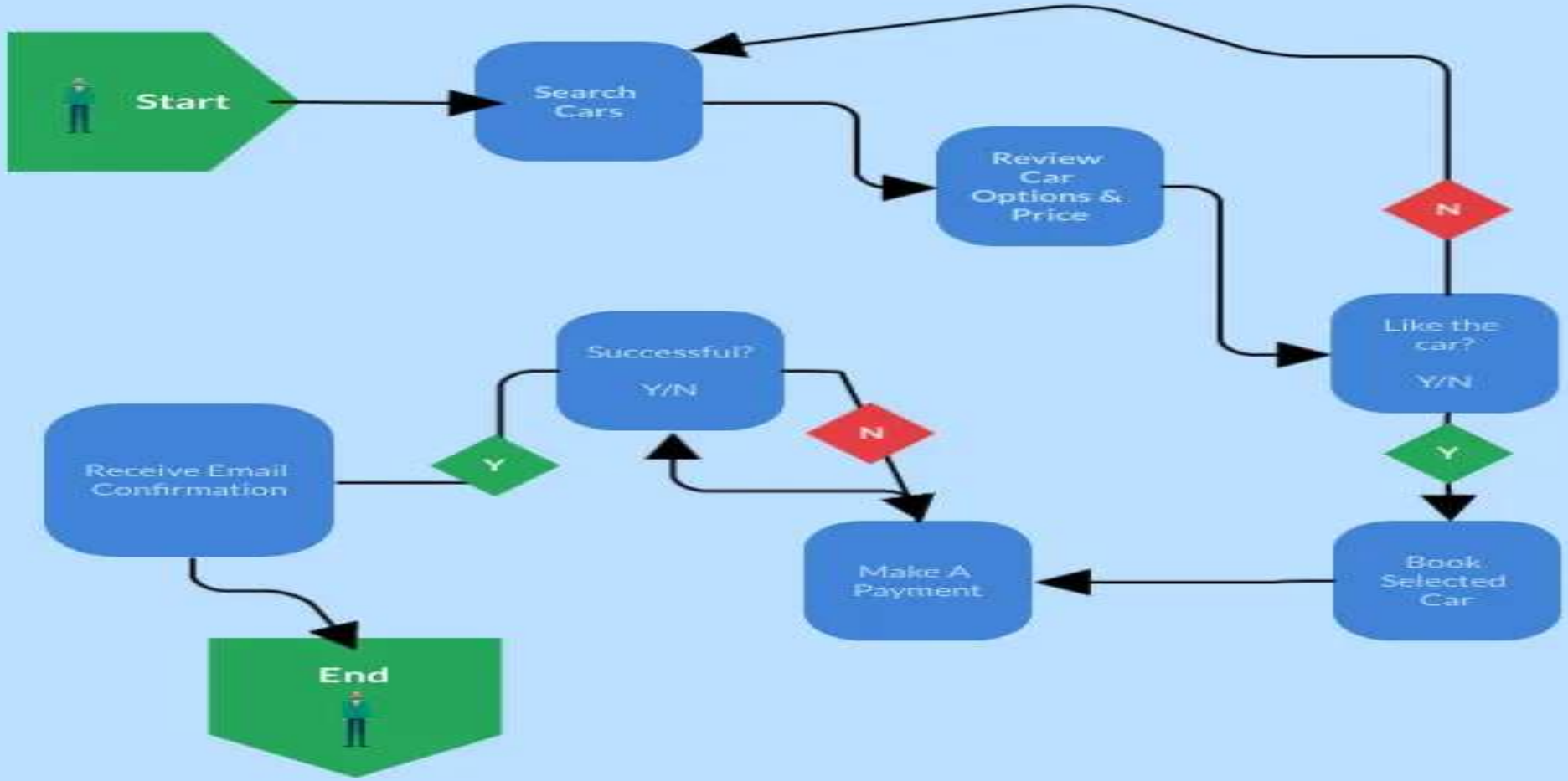
Introduction

- **This document outlines the key steps in the user journey and the technical flow for the Rental Car E-Commerce website.**

- **User Journey Overview**
- **Homepage**: Users can explore available rental cars by applying filters for car type, price, and rental duration.
- **Car Information**: Users can access detailed information about each car, including its name, type, rental price, available features (such as GPS or hybrid options), and its availability status.

- **Adding to Cart**: Once users are signed in or have registered, they can add selected cars to their shopping cart.
- **Checkout Process**: Users review their cart items, select rental durations, and proceed to payment.
- **Post-Booking**: After completing the booking, users receive a confirmation along with payment details and shipment tracking information.

Workflow: Rent A Car



Backend Workflow

Sanity CMS: Manages and stores information about cars, customer bookings, and user data.

Payment Gateway: Facilitates secure payment transactions and records transaction details for further reference.

Shipment Tracking: Updates the status of the car to "Booked" and provides real-time delivery status updates.

System Architecture

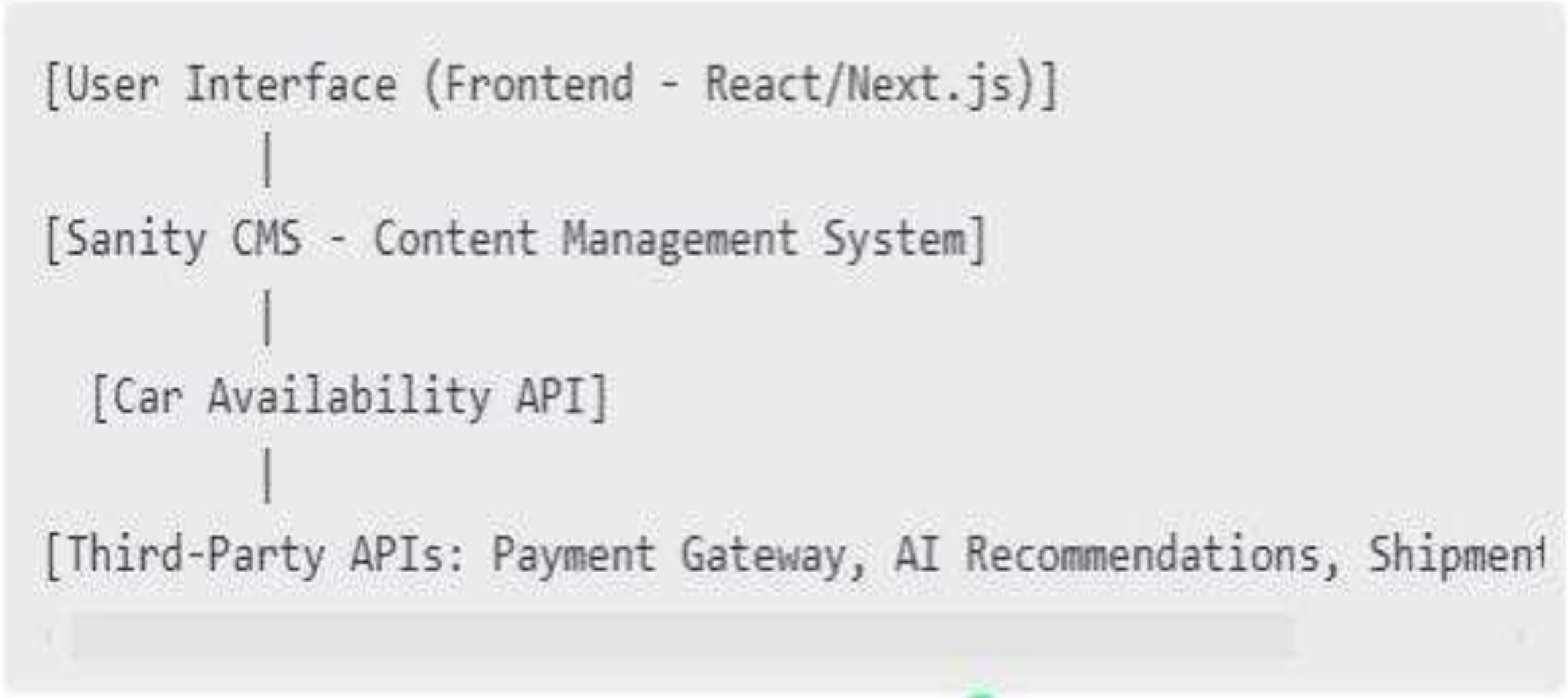
Overview:

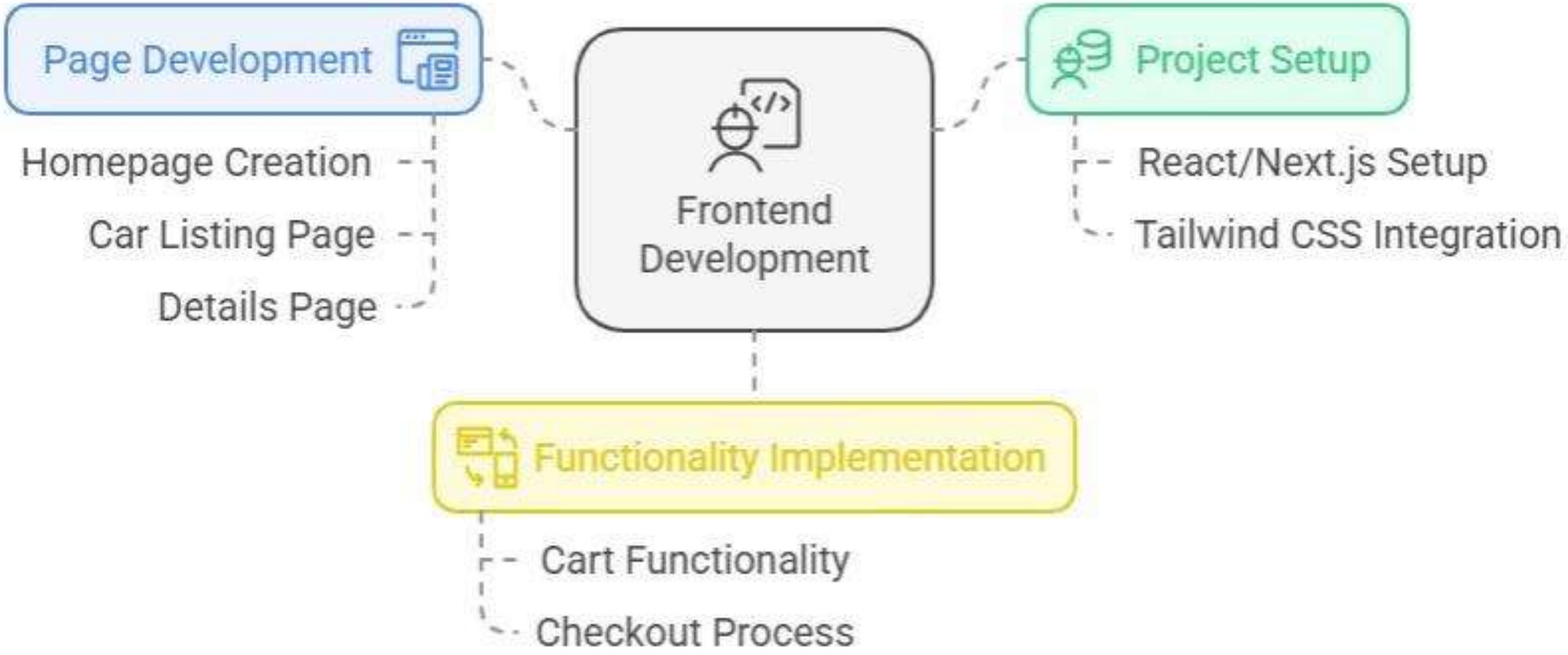
Frontend: The user interface is developed using React/Next.js, allowing users to browse, book, and interact with the website seamlessly.

Sanity CMS: Responsible for managing and storing all car-related information and customer data.

APIs: Handles external services such as payment processing, shipment tracking, and real-time car availability updates.

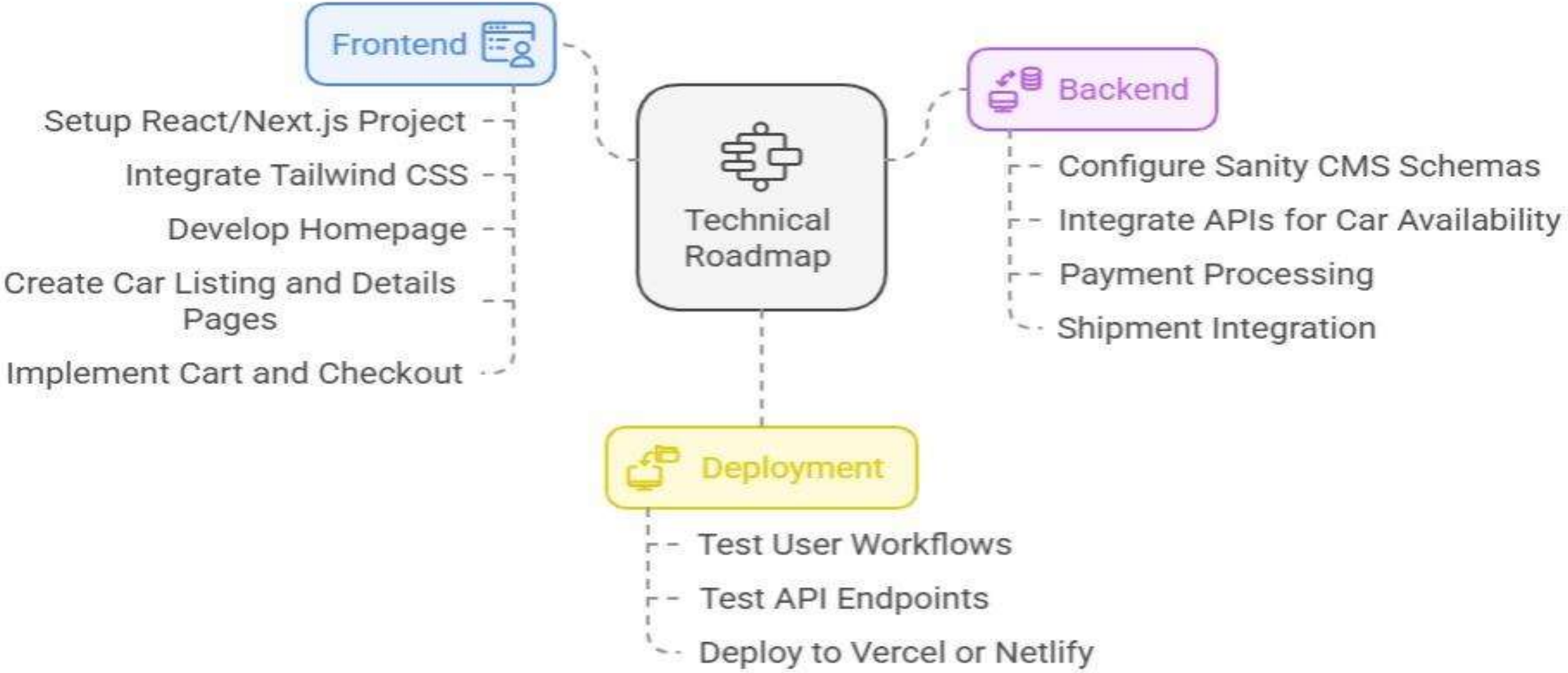
High-Level Architecture




Page Development
Homepage Creation
Car Listing Page
Details Page
Frontend Development
Project Setup
React/Next.js Setup
Tailwind CSS Integration
Functionality Implementation
Cart Functionality
Checkout Process

- Technical Roadmap
- Frontend
- Setup React/Next.js project with Tailwind CSS.
- Build homepage, car listing, and details pages.
- Create cart and checkout functionality.

- Backend
- Configure Sanity CMS schemas for cars and bookings.
- Integrate APIs for car availability, payments, and shipment.
- Deployment
- Test user workflows and API endpoints.
- Deploy to platforms like Vercel or Netlify.


Technical Roadmap
Frontend
Setup React/Next.js Project
Integrate Tailwind CSS
Develop Homepage
Create Car Listing and Details Pages
Implement Cart and Checkout
Backend
Configure Sanity CMS Schemas
Integrate APIs for Car Availability
Payment Processing
Shipment Integration
Deployment
Test User Workflows
Test API Endpoints
Deploy to Vercel or Netlify

Key API Endpoints

Endpoint	Method	Purpose
GET /cars	GET	Fetch all cars
GET /cars/:id	GET	Fetch specific car
POST /bookings	POST	Create a booking
POST /payments	POST	Process payment
GET /shipment/:id	GET	Fetch shipment details

Sanity Schema Examples

Cars Schema : export default {

```
  name: ‘car’,
  type: ‘document’,
  fields: [
    { name: ‘name’, type: ‘string’, title: ‘Car Name’ },
    { name: ‘price’, type: ‘number’, title: ‘Price’ },
    { name: ‘availability’, type: ‘boolean’, title: ‘Availability’ },
```

Customer Schema : export default {
  name: ‘customer’,
  type: ‘document’,
  fields: [
    { name: ‘name’, type: ‘string’, title: ‘Customer Name’ },
    { name: ‘email’, type: ‘string’, title: ‘Email Address’ },
    { name: ‘phone’, type: ‘string’, title: ‘Phone Number’ },
    { name: ‘address’, type: ‘string’, title: ‘Address’ },