

Section 10

Section 1 Maham

52107

Hackathon Day 2

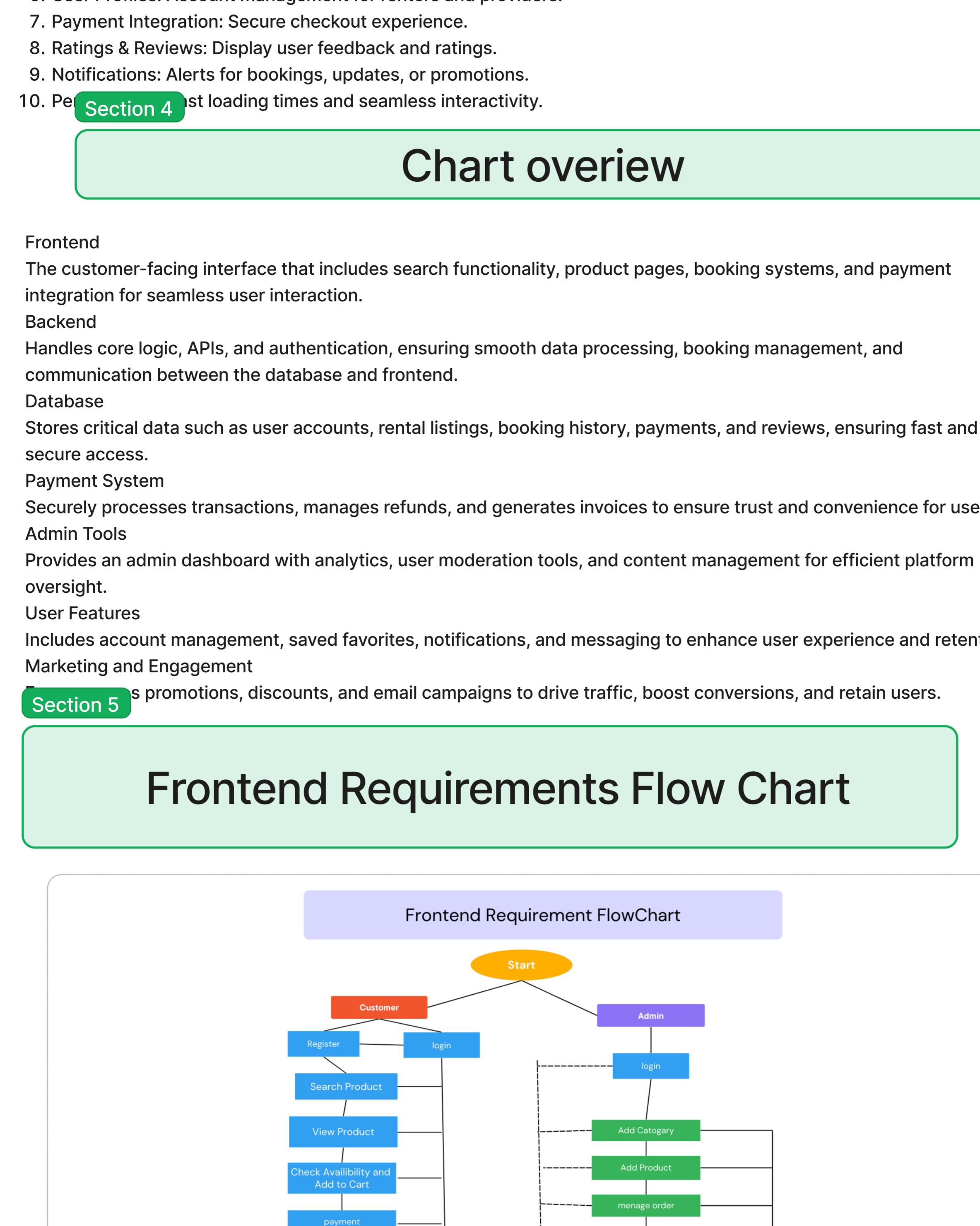
Marketplace Technical Foundation

(Rental E-Commerce)

This document outlines the end-to-end user experience for an online rental marketplace. This foundation ensures scalability, reliability, and user-friendly experiences tailored for rental businesses.

Section 2

Homepage Rental E-commerce



Section 3

Frontend Requirements

Frontend Requirements for Rental E-commerce:

- User Interface: Intuitive design for browsing, searching, and booking.
- Responsive Design: Optimized for mobile, tablet, and desktop.
- Search & Filters: Easy navigation with advanced filters (price, location, availability).
- Product Pages: Detailed listings with images, descriptions, and availability.
- Booking System: Interactive calendar for reservations.
- User Profiles: Account management for renters and providers.
- Payment Integration: Secure checkout experience.
- Ratings & Reviews: Display user feedback and ratings.
- Notifications: Alerts for bookings, updates, or promotions.
- Performance: Fast loading times and seamless interactivity.

Section 4

Chart overview

Frontend

The customer-facing interface that includes search functionality, product pages, booking systems, and payment integration for seamless user interaction.

Backend

Handles core logic, APIs, and authentication, ensuring smooth data processing, booking management, and communication between the database and frontend.

Database

Stores critical data such as user accounts, rental listings, booking history, payments, and reviews, ensuring fast and secure access.

Payment System

Securely processes transactions, manages refunds, and generates invoices to ensure trust and convenience for users.

Admin Tools

Provides an admin dashboard with analytics, user moderation tools, and content management for efficient platform oversight.

User Features

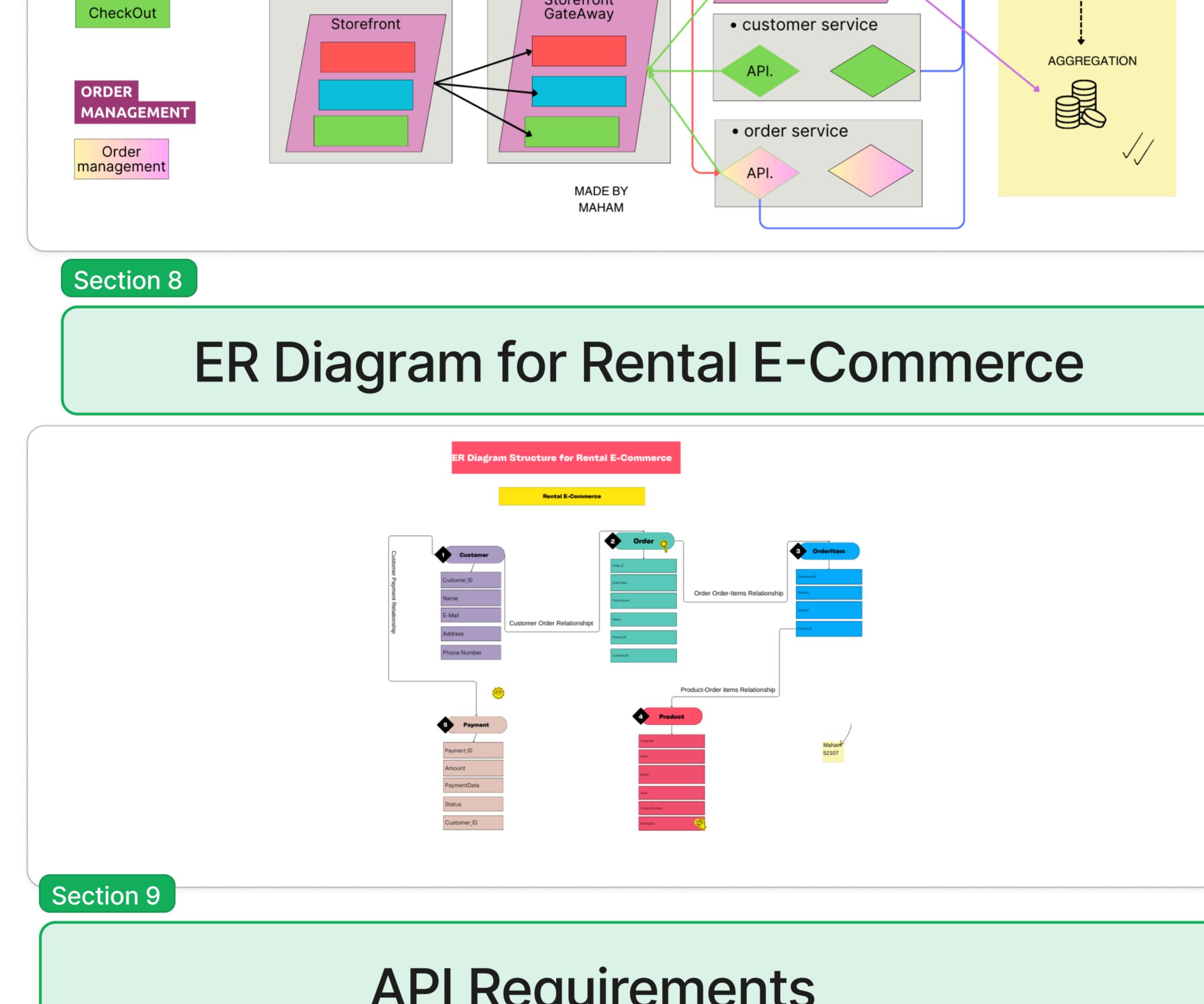
Includes account management, saved favorites, notifications, and messaging to enhance user experience and retention.

Marketing and Engagement

Manages promotions, discounts, and email campaigns to drive traffic, boost conversions, and retain users.

Section 5

Frontend Requirements Flow Chart



Section 6

System Architecture

Frontend:

- Built with Next.js for fast, responsive interfaces.
- Features: Search, filters, booking calendar, user profiles.
- Dynamic product data fetched from backend.
- Send API request to fetch product listing, manage orders and track shipment.

Sanity CMS:

- Manages dynamic content like car listings and marketing material.
- stores data for products(name, price, availability), Orders (Customer info, status) and rentals.
- provide end-points for fronted data retrievals.
- Integrates with third-party APIs for payment and shipment.

Third-party APIs:

- APIs: Sanity for content, Stripe/PayPal for payments, Google Maps for location services, and insurance APIs.
- Handles booking validation, availability checks, and pricing logic.
- process the secure payments for rental transactions.
- send real-time updates to the frontend via the backend.

DataFlow:

product Browsing:

- User requests product data via the frontend.
- Backend (sanity CMS) serves product data dynamically.

Order Placement:

- Frontend sends order details (rental duration, customer info) to the backend.
- backend saves the orders and process payment via the payment gateway.

Shipment Tracking:

Backend requests order shipment details from the shipment tracking API.

Real-time updates are displayed on the frontend.

Section 7

System Architecture Overview Diagram



Section 8

ER Diagram for Rental E-Commerce



Section 9

API Requirements

Rental duration and booking:

post: post/API/rental-duration.

{ "product ID": "343", "duration": "4 days", "deposit": 2000, "start Date": "2025-03-11", "end Date": "2025-03-15", "user ID": "user111" }

Response:

{ "confirmation ID": 766, "status": "Success", "rental Details": { "total Cost": 4500, "deposit Amount": 2000, "start Date": "2025-03-11", "end Date": "2025-03-15", "product Info": { "id": 333, "name": "Corolla" } } }

Rental Availability

GET /Api/rental-availability/{product ID}

Description: Check if a product is available for specific dates.

Query Parameters: start Date, end Date.

{

"is Available": true,

"available Dates": [

"start Date": "2025-03-11",

"end Date": "2025-04-15"

]

"base Price": 2000,

"deposit Required": 2500,

]

Extend Rental Duration

PATCH /Api/rental-duration/{rental ID}/extend

{

"extension Days": 3,

"newEndDate": "2025-03-21"

}

{

"status": "Success",

"updated Rental": {

"rental ID": 333,

"newEndDate": "2025-03-21",

"additional Cost": 1500

}

Rental Condition Report

POST /Api/rental/{rental ID}/condition

{

"condition": "Good",

"notes": "Minor dent on backside or slightly colour fade from dent",

"images": ["image1.jpg", "image2.jpg"],

"report Type": "pre-rental"

}

{"report ID": "report111",

"status": "Submitted",

"timestamp": "2025-03-11T12:00:00Z"

}

Get Rental History

GET /Api/rental/history/{user ID}

{

"rentals": [

"rental ID": 333,

"product Name": "Corolla",

"start Date": "2025-03-11",

"end Date": "2025-03-15",

"status": "Active",

"total Paid": 1500

]

"total Rentals": 5,

"active Rentals": 1

Cancel Rental

POST /Api/rental/{rental ID}/cancel

{

"cancellation Reason": "Change of plans",

"request Refund": true

}

{

"status": "Cancelled",

"refund Amount": 1000,

"refunded": "ref111",

"cancellation Fee": 500

}

Process Rental Return

POST /Api/rental/{rental ID}/return

{

"return Date": "2025-03-15",

"condition": "Good",

"damage Report": null,

"deposit Return": true

}

{

"status": "Returned",

"deposit Return Amount": 1500,

"processing Id": "proc123",

"additional charges": 0

}

Rental Details

GET /Api/rental/{rental ID}

{

"rental": {

"id": 333,

"product": {

"id": 111,

"name": "Corolla",

"category": "Cars"

"duration": "4 days",

"status": "Active",

"deposit": {

"amount": 1500,

"status": "Held"

}

"dates": {

"start": "2025-03-11",

"end": "2025-03-15",

}

"costs": {

"rentalFee": 4500,

"deposit": 1500,

"total": 3000

,

"condition Reports": {

"preRental": "report111",

"post Rental": null

}

}

}

</div