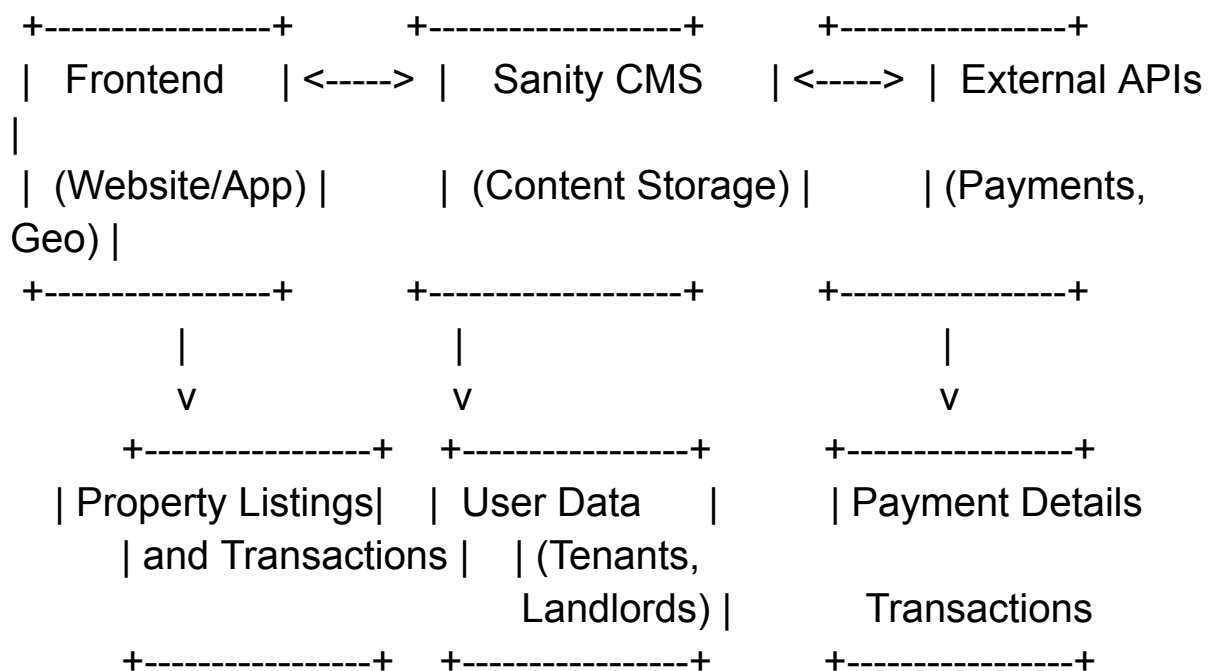


Day -2 Of MarketPlace Launch (Technical Foundation)



The diagram below shows the relation/link among the data schema I made for the last task..



A simple yet understandable foundation to be implemented initially. It can be modified over time with understanding.

What does it show?

- How the frontend fetches data from **Sanity CMS**.
- The roles of **Sanity** in managing dynamic content.
- The interaction with **third-party APIs** for payment processing and other functionalities.

2- WorkFlows (From a buyer's perspective)

1-User opens Website/App

2-A welcome page pops up

3-App Asks whether is the user buyer, seller, investor

4-App Asks for Current Location / Approximate Location

5-App Asks which category they are interested (House, Bungalow, Apartment, PG Room)

6- App asks about the number of family members or whether they are single

7-App asks the reason behind shifting

(Past experiences /issues with current location, owner, residency)

8-App asks the user's budget, desired location, rooms, size of house etc.

9-App asks about personal details

10-App shows photos/videos of enlisted residencies available at the specific or nearer location.

11-User selects their desired option by clicking on view details and may add on their wishlist.

12-User may click on the contact agent.

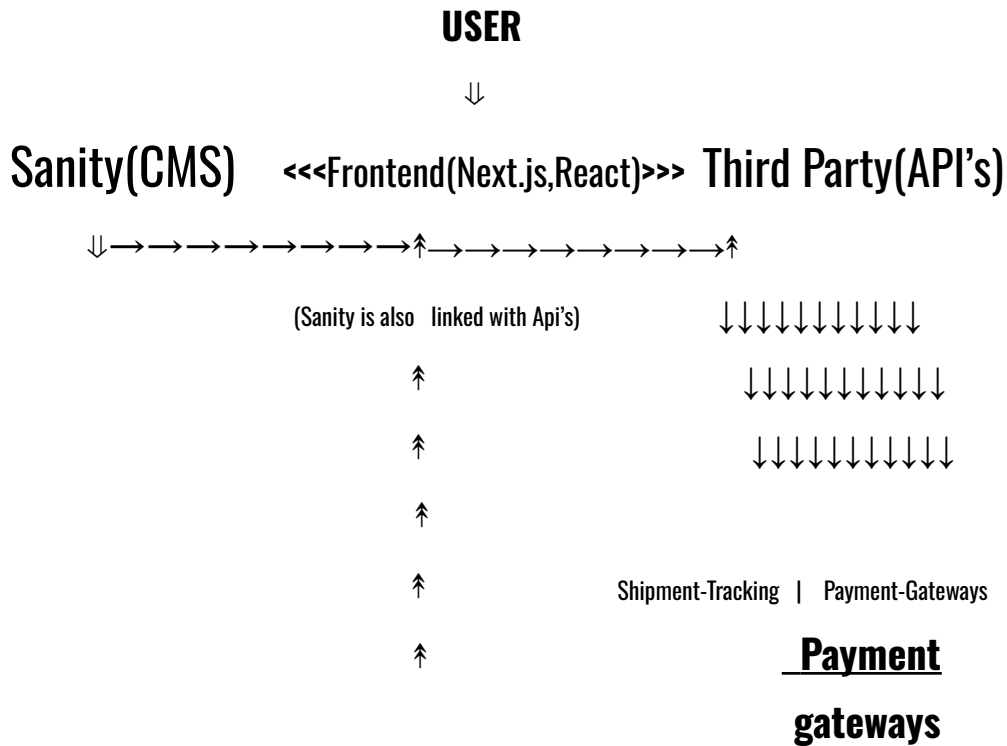
13-User sends a message .

14-A human/Ai agent will verify their request & send an automated message to wait.

15-Human or Ai agent will contact them and ensure availability of the property & the timings to visit

(Will continue adding more after deep planning)

Here's another brainstormed ER Diagram which is crucial for one's understanding and may follow it as a blueprint initially.



are also linked with the front-end for the user interactive interface..

Endpoint	Method	Description	Request Body (if applicable)	Response Body
/users	GET	Get all users	-	List of User objects
/users/{user_id}	GET	Get user by ID	-	User object
/users	POST	Create a new user	User object (full_name, email, phone_number, password, etc.)	User object
/users/{user_id}	PUT	Update user information	User object (partial update allowed)	Updated User object
/users/{user_id}	DELETE	Delete user account	-	Success/Failure message
/properties	GET	Get all properties	-	List of Property objects
/properties/{property_id}	GET	Get property by ID	-	Property object
/properties	POST	Create a new property listing	Property object	Property object
/properties/{property_id}	PUT	Update property information	Property object (partial update allowed)	Updated Property object
/properties/{property_id}	DELETE	Delete property listing	-	Success/Failure message
/transactions	GET	Get all transactions	-	List of Transaction objects
/transactions/{transaction_id}	GET	Get transaction by ID	-	Transaction object
/reviews	GET	Get all reviews	-	List of Review objects
/reviews/{review_id}	GET	Get review by ID	-	Review object
/reviews	POST	Create a new review	Review object	Review object
/wishlist	GET	Get user's wishlist	-	List of Property objects in wishlist
/wishlist/{wishlist_id}	POST	Add property to wishlist	Property ID	Success/Failure message
/wishlist/{wishlist_id}	DELETE	Remove property from wishlist	Property ID	Success/Failure message
/messages	GET	Get user's messages	-	List of Message objects
/messages/{message_id}	GET	Get message by ID	-	Message object
/messages	POST	Send a new message	Message object	Message object
/requests	GET	Get user's requests	-	List of Request objects
/requests/{request_id}	GET	Get request by ID	-	Request object
/requests	POST	Submit a new request	Request object	Request object
/notifications	GET	Get user's notifications	-	List of Notification objects
/notifications/{notification_id}	GET	Get notification by ID	-	Notification object

Sanity Schema for My MarketPlace:

Property.Schema	User-Schema	Transaction-Schema	Review Schema
<pre>export default { name: 'property', type: 'document', title: 'Property', fields: [{ name: 'title', type: 'string', title: 'Property Title', }, { name: 'description', type: 'text', title: 'Description', }, {</pre>	<pre>export default { name: 'user', type: 'document', title: 'User', fields: [{ name: 'name', type: 'string', title: 'Full Name', }, { name: 'email', type: 'string', title: 'Email Address', }, { name: 'phone',</pre>	<pre>export default { name: 'transaction', type: 'document', title: 'Transaction', fields: [{ name: 'property', type: 'reference', title: 'Property', to: [{ type: 'property' }], }, { name: 'tenant', type: 'reference', title: 'Tenant', to: [{ type: 'user' }], }, { name: 'amount', type: 'number',</pre>	<pre>export default { name: 'review', type: 'document', title: 'Review', fields: [{ name: 'property', type: 'reference', title: 'Property', to: [{ type: 'property' }], }, { name: 'reviewer', type: 'reference', title: 'Reviewer', to: [{ type: 'user' }], }, { name: 'rating',</pre>

name: 'location', type: 'geopoint', title: 'Location', }, { name: 'price', type: 'number', title: 'Monthly Rent Price (PKR)', }, { name: 'propertyType', type: 'string', title: 'Type', options: { list: ['Apartment', 'House', 'Shop', 'Office Space'], }, }, {	type: 'string', title: 'Phone Number', }, { name: 'role', type: 'string', title: 'User Role', options: { list: ['Tenant', 'Landlord', 'Broker'], }, { name: 'profilePicture', type: 'image', title: 'Profile Picture', },], };	title: 'Payment Amount (PKR)', }, { name: 'status', type: 'string', title: 'Transaction Status', options: { list: ['Pending', 'Completed', 'Failed'], }, }, { name: 'transactionDate', type: 'datetime', title: 'Date of Transaction', }, { name: 'rentalDuration', type: 'object', title: 'Rental Duration', fields: [type: 'number', title: 'Rating (1-5)', validation: Rule => Rule.min(1).max(5), }, { name: 'comment', type: 'text', title: 'Comment', }, { name: 'reviewDate', type: 'datetime', title: 'Date of Review', },], };
---	--	---	--

name: 'status', type: 'string', title: 'Status', options: { list: ['Available', 'Booked', 'Rented'], }, }, { name: 'amenities', type: 'array', title: 'Amenities', of: [{ type: 'string' }], }, { name: 'images', type: 'array', title: 'Property Images', of: [{ type: 'image' }],		name: 'startDate', type: 'datetime', title: 'Start Date', }, { name: 'endDate', type: 'datetime', title: 'End Date', }, },], },], };	
--	--	---	--

<pre> }, { name: 'owner', type: 'reference', title: 'Property Owner', to: [{ type: 'user' }], }, { name: 'createdAt', type: 'datetime', title: 'Date Added', },], }; </pre>			
Booking-Schema	Agent-Schema	Inquiry-Schema	Notification-Schema
<pre> export default { name: 'booking', type: </pre>	<pre> export default { name: 'agent', type: </pre>	<pre> export default { name: 'inquiry', type: 'document', </pre>	<pre> export default { name: 'notification', type: 'document', </pre>

'document', title: 'Booking', fields: [{ name: 'property', type: 'reference', title: 'Property', to: [{ type: 'property' }], }, { name: 'user', type: 'reference', title: 'User', to: [{ type: 'user' }], }, { name: 'bookingDate', type: 'datetime', title: 'Booking Date',	'document', title: 'Agent', fields: [{ name: 'name', type: 'string', title: 'Agent Name', }, { name: 'email', type: 'string', title: 'Email Address', }, { name: 'phone', type: 'string', title: 'Phone Number', }, { name: 'associatedProper ties', type: 'array', title:	title: 'Inquiry', fields: [{ name: 'property', type: 'reference', title: 'Property', to: [{ type: 'property' }], }, { name: 'user', type: 'reference', title: 'User', to: [{ type: 'user' }], }, { name: 'message', type: 'text', title: 'Message', }, { name: 'inquiryDate', type: 'datetime', title: 'Inquiry Date', }, { name: 'status',	title: 'Notification', fields: [{ name: 'user', type: 'reference', title: 'User', to: [{ type: 'user' }], }, { name: 'message', type: 'string', title: 'Notification Message', }, { name: 'read', type: 'boolean', title: 'Read Status', initialValue: false, }, { name: 'createdAt', type: 'datetime',
--	--	---	---

<pre> }, { name: 'status', type: 'string', title: 'Booking Status', options: { list: ['Pending', 'Confirmed', 'Cancelled'], }, }, { name: 'notes', type: 'text', title: 'Special Requests/Notes', },], }; </pre>	<pre> 'Managed Properties', of: [{ type: 'reference', to: [{ type: 'property' }] }], },], }; </pre>	<pre> type: 'string', title: 'Status', options: { list: ['New', 'Responded', 'Closed'], },], }; </pre>	<pre> title: 'Created At', },], }; </pre>
---	--	---	--

Collaboration Notes

As an independent creator, I relied entirely on my ideas, research, and creativity to design and plan my R-Commerce marketplace. My focus was on solving real problems in Pakistan's real estate industry by creating a platform that prioritizes transparency, affordability, and efficiency.

Here's how I approached this project:

- **Research-Driven Insights**

- Studied the challenges faced by renters and property owners, such as unprofessional brokers and communication gaps, to create solutions that directly address these pain points.

- **Self-Guided Learning**

- Explored tools like Sanity CMS and examples from global eCommerce platforms to build a solid technical foundation.
- Designed workflows, schemas, and API plans to ensure a smooth and reliable experience for users.

- **User-Centric Approach**

- Always thought from the perspective of the end user, ensuring the platform is easy to use, fast, and trustworthy.

- **Iterative Refinement**

- Continuously improved my technical plans and designs to align them with the business goals and real-world needs.

Every step of this journey reflects my dedication to creating something meaningful and impactful. I hope my work inspires confidence and demonstrates my ability to take on complex challenges independently.