**Mongo Db\_Data\_Modelling**

**Data Collections:**

**1. amenities\_data.json**

**Purpose**: Stores a list of amenities linked to each property.

**Sample Fields**:

* Property Id: ID referencing the property
* amenities: List of features like *Power Backup*, *Gym*, *Park*, *Swimming Pool*, etc.

**Detailed Description**:  
This collection enhances the tenant experience by allowing users to filter/search based on lifestyle preferences. Amenities listed here directly affect the perceived value

**2. maintenance\_requests\_data.json**

**Purpose**: Tracks maintenance issues raised by tenants.

**Sample Fields**:

* tenant\_id, property\_id: References to who raised the request and where
* description: Problem detail
* status: Current state — *Pending*, *In Progress*, or *Resolved*
* photo: Visual proof of the issue
* createdAt: Date of submission

**Detailed Description**:  
A vital collection that ensures tenant satisfaction and timely issue resolution. It helps prioritize resources and track recurring problems. Property managers use this data to monitor performance and maintain standards.

**3. owners\_data.json**

**Purpose**: Holds data about all property owners in the community.

**Sample Fields**:

* name, email, phone: Contact details
* house Number, floor, tower: Location owned
* Bank Details: Bank info for rent deposits (includes IFSC)

**Detailed Description**:  
This collection connects properties to landlords. It helps automate payments, notifications, and communications with owners. It’s useful for generating rent reports and auditing owner-specific issues or history.

**4. properties\_data.json**

**Purpose**: Core catalog of all rental properties.

**Sample Fields**:

* property\_type: *Studio*, *Affordable*, etc.
* Rent Amount, Advance Amount: Pricing info
* furnishing: *Furnished*, *Semi-Furnished*
* Car Parking: Boolean + Parking number
* bedrooms, bathrooms, facing, pet allowed

**Detailed Description**:  
Central to the platform’s operation, this collection powers listings and search filters. It also helps tenants understand the pricing structure and property specifications. Enables categorization and recommendation logic.

**5. rental\_agreements\_data.json**

**Purpose**: Tracks the lease agreements between tenants and landlords.

**Sample Fields**:

* tenantId, propertyId: Links to related entities
* startDate, endDate: Contract duration
* status: *Active*, *Expiring Soon*
* eSignature: Whether digitally signed

**Detailed Description**:  
This legally significant collection governs tenancy. It is used for alerts (renewals, expiries), conflict resolution, and payment scheduling. It also provides insights into occupancy trends and leasing patterns.

**6. tenants\_data.json**

**Purpose**: Profiles all tenants in the system.

**Sample Fields**:

* name, email, phone: Contact info
* house Number, floor, tower: Where they reside
* pets: Pet ownership and details
* rent Paid: Monthly rent paid

**Detailed Description**:  
Used for communication, personalization, and analytics. Helps in issuing reminders, tailoring policies (e.g., pet-friendly zones), and generating rent collection reports. Can also feed into loyalty/reward or feedback systems.

**Project Description:**

The **Rental Management & Community Housing Hub** is a full-stack web application designed as a comprehensive platform for managing rental properties and enhancing co-living experiences. It enables **property owners**, **tenants**, and **community members** to seamlessly coordinate day-to-day operations, share responsibilities, and foster a sense of connected living.

This all-in-one solution promotes collaboration, transparent communication, and streamlined rental processes—from property listing and lease management to maintenance and payments—while building a vibrant residential community.