

Project Overview: Rental Property Discovery App

Project Description

The *Rental Property Discovery App* is a cross-platform mobile and web application designed to simplify the process of finding, listing, and managing rental properties. It connects **property owners/agents** with **tenants/renters** in real time, offering features like location-based search, advanced filtering, in-app messaging, and secure booking.

The app aims to be an all-in-one rental marketplace similar to platforms like No Broker, 99acres, or Zillow — but built leaner, faster, and more accessible for users across Tier 2 and Tier 3 cities in India and emerging markets.

1. Purpose

The app will enable users to search verified rental properties with filters, view properties on a map or list, and contact landlords directly. Property owners can post listings, boost them for visibility, and manage tenant inquiries. Monetization includes ads, subscriptions, affiliate offers, and premium listings.

2. Scope

The system will consist of:

- A **mobile-first app** (React Native + Expo)
- A **web portal** for renters and landlords
- Backend (Supabase) with auth, database, chat, and storage
- Admin interface for moderation and analytics
- Monetization modules: AdMob, Stripe/Razorpay, Affiliate links

Project Goals

1. **Bridge the gap** between property owners and renters with a seamless, easy-to-use interface.
2. Enable **verified listings, owner uploads, and map-based property discovery**.
3. Offer a **free tier** for renters and a **premium listing/analytics dashboard** for property owners.
4. Integrate **deep linking, passwordless auth, and mobile-first design** for maximum reach.
5. Monetize via ads, promoted listings, and subscription plans.

What Problems Does It Solve?

Problem	Solution
<input checked="" type="checkbox"/> Renters struggle to find verified, real-time listings.	<input checked="" type="checkbox"/> Provide updated listings with verified owner uploads.
<input checked="" type="checkbox"/> Listing portals are cluttered with brokers or fake properties.	<input checked="" type="checkbox"/> Filter out brokers with owner-only listings and verification.
<input checked="" type="checkbox"/> Property owners don't have a fast way to list/manage rentals on mobile.	<input checked="" type="checkbox"/> Mobile-first owner dashboard with image/document upload, analytics, and chat.
<input checked="" type="checkbox"/> No real-time booking, communication, or availability updates.	<input checked="" type="checkbox"/> In-app chat, calendar scheduling, and instant booking system.

Vision

To become the most trusted, lightweight, and mobile-first rental discovery platform for local communities, small towns, and independent landlords — empowering millions to find or list homes with just a few taps.

"We envision a future where **middlemen are minimized, trust is built-in, and both renters and landlords have full control** through technology — affordable, intuitive, and reliable."

2. System Overview

- Platform will support **renters** (search, chat, save) and **owners** (list, promote, respond).
- Map and list-based property discovery
- In-app messaging and notification
- Verified owner badges and ratings (future scope)
- Monetization through ads, listing upgrades, and optional subscriptions.

3. Functional Requirements.

3.1 Renter Features

ID	Feature	Description
FR-1	Search Listings	Filter by location, price, BHK, furnished
FR-2	Map + List View	Show rentals on a map and as a scrollable list

<i>FR-3</i>	Property Detail View	Show photos, amenities, location, and contact options
<i>FR-4</i>	Save Favorites	Ability to save and revisit properties
<i>FR-5</i>	In-app Chat	Chat with landlords within the app
<i>FR-6</i>	Notifications	Push alerts for new listings or messages
<i>FR-7</i>	Subscription Upgrade	Unlock early access, AI search, or verified-only listings

3.2 Owner Features.

<i>ID</i>	<i>Feature</i>	<i>Description</i>
<i>FR-8</i>	Post Property	Add title, rent, address, photos
<i>FR-9</i>	Edit/Delete Listings	Manage property listings from dashboard
<i>FR-10</i>	Listing Insights	View number of views and inquiries
<i>FR-11</i>	Chat with Renters	Reply to tenant inquiries in-app
<i>FR-12</i>	Boost Listing	Upgrade visibility via Stripe/Razorpay checkout

3.3 Admin Features.

<i>ID</i>	<i>Feature</i>	<i>Description</i>
<i>FR-13</i>	Content Moderation	Flag, hide, or review suspicious listings
<i>FR-14</i>	Metrics Dashboard	DAU, listing views, revenue, conversions
<i>FR-15</i>	Support Chat	Admin-level communication channel

4. Non-Functional Requirements.

<i>ID</i>	<i>Requirement</i>	<i>Description</i>
<i>NFR-1</i>	Scalability	Support 1M+ users and city-level expansion
<i>NFR-2</i>	Security	Secure auth (Supabase), image uploads, rate-limiting
<i>NFR-3</i>	Performance	<2s average page load, 99.9% uptime
<i>NFR-4</i>	Cross-platform	Responsive on both mobile and web (Expo + Next.js)

NFR-5	Localization	Support INR and English/Hindi UX
NFR-6	Analytics	Firebase or PostHog instrumentation for user events
NFR-7	Privacy & Compliance	GDPR and Indian data privacy laws adherence



User Persona Table.

Persona Name	Demographics	Goals	Pain Points	How App Helps
Ravi (Tenant)	24, Software Engineer, Mumbai	Find affordable rental in good locality with verified listings.	Fake listings, agent scams, hard to compare options.	Verified listings, filters by location/price, chat with landlord.
Anjali (Student)	20, College Student, Pune	Rent a PG near college for 1 year, preferably girls-only accommodation.	Unclear PG conditions, no pictures, too many brokers.	PG-specific filters, images, reviews, direct landlord contact.
Karan (Landlord)	38, Property Owner, Bangalore	Easily post and manage multiple properties without broker involvement.	Reaching the right audience, manual work for each post.	Post in minutes, manage from dashboard, featured listing options.
Meera (Broker)	42, Real Estate Agent, Delhi	Reach more clients and close deals faster.	No digital presence, hard to track inquiries.	Broker account, inquiry dashboard, premium promotion option.
Suresh (NRIs Owner)	45, Lives in US, Owns flat in India	Rent out his property safely while abroad.	Trust issues, no time to manage listings or verify tenants.	Verified tenant profiles, remote property management features.



6. Product Roadmap

Phase 1: MVP (0–8 Weeks)

- Supabase Auth
- Listing CRUD (owner)
- Filters, List + Map view (renter)

- Property Detail Page
- In-app Chat
- AdMob Integration

Phase 2: Engagement (9–14 Weeks)

- Push Notifications
- Favorites, Alerts
- Verified badges
- Listing Metrics (views, chats)

Phase 3: Monetization (15–18 Weeks)

- Premium Listings
- Stripe / Razorpay integration
- Affiliate hooks (rental docs, movers)

Phase 4: Delight + Scale (19–24 Weeks)

- AI Search Bot
- Virtual Tours
- Auto Lease Generator & Subscription Tiers

₹ 7. Budget & Cost Breakdown.

Item	Estimated Cost
<i>Dev Team (1 PM, 2 Devs, 1 Designer, 1 QA - 6 months)</i>	₹25–30 Lakh
<i>Infra (Supabase, Cloudinary, Mapbox, Firebase, Stripe fees)</i>	₹2 Lakh
<i>Design (UI/UX + assets)</i>	₹1–2 Lakh
<i>Marketing (City launch, influencers, ads)</i>	₹5 Lakh
<i>Contingency + Legal + Docs</i>	₹1 Lakh
Total (6–9 Months)	₹35–40 Lakh (~\$40K–\$48K USD)

8. Risks & Mitigation.

Risk	Impact	Mitigation
<i>Low owner adoption</i>	Medium	Incentivize listings with referral program
<i>Chat abuse or spam</i>	High	Moderation tools + rate limits
<i>Ad fatigue</i>	Medium	Keep AdMob unobtrusive; use interstitials only in transition screens
<i>Payment issues (failures/fraud)</i>	Medium	Use verified gateways (Razorpay/Stripe), offer COD for docs
<i>Regulatory changes</i>	Low	Comply with local housing and data laws from day one

9. Open Requirements (TBD for Scaling)

- Admin CMS (moderation, analytics)
- Auto KYC for verified badge
- Rent receipt generation / payment integration
- SMS fallback (for low-data users)
- Tenant history / reviews (from owners)
- Support for tiered pricing based on city/region.

10. Appendices.

Tech Stack Summary

Layer	Tool
<i>Mobile</i>	React Native + Expo
<i>Web</i>	Next.js + Tailwind
<i>Backend</i>	Supabase (DB, Auth, Storage, Realtime)
<i>Maps</i>	Google Maps or Mapbox
<i>Payment</i>	Stripe (global), Razorpay (India)
<i>Chat</i>	Supabase Realtime or third-party like Stream
<i>Ads</i>	AdMob (via Expo SDK)
<i>Analytics</i>	PostHog, Firebase

JIRA-Ready User Stories: Rental Property Discovery App

Epic 1: Authentication & Onboarding (Week 1–2)

Story: User signup and login via Email OTP

- **User:** All
- **As a user, I want** to sign up or log in using my email so I can securely access the app.
-  **Acceptance Criteria:**
 - Email-based OTP login flow
 - Error states for wrong OTP or email
 - Store user session securely
-  **Tasks:**
 - Implement Supabase Auth setup
 - Build login/signup screen
 - Handle OTP verification and state
 - Session persistence on app restart

Story: First-time onboarding screens

- **User:** All
- **As a new user, I want** to see onboarding so I understand what the app does.
-  **Acceptance Criteria:**
 - 2–3 intro screens with skip button
 - Call to action for sign-up
-  **Tasks:**
 - Design onboarding assets
 - Implement screen swiper (e.g. React Native PagerView)
 - Store flag in local storage: onboarded = true

Epic 2: Property Listings (Week 3–4)

Story: Add new property listing (owner)

- **User:** Owner
- **As a property owner, I want** to post my property details so renters can find them.
-  **Acceptance Criteria:**
 - Fields: Title, Rent, Address (with map), BHK, Type, Photos
 - Successful submission shows listing
-  **Tasks:**
 - Build Add Listing form
 - Use Supabase Storage for image uploads
 - Geolocation API for map picker
 - Write to listings table

Story: Search + filter properties (renter)

- **User:** Renter
- **As a renter, I want** to search by location and filter by price/type so I find relevant listings.
-  **Acceptance Criteria:**
 - Search input with debounce
 - Filters: Price, Type (1BHK, etc.), Furnished
 - Instant result updates
-  **Tasks:**
 - Search input + filtering UI
 - Backend filtering query
 - Pagination or infinite scroll

Story: Map + list toggle view

- **User:** Renter
- **As a renter, I want** to toggle between map and list views so I can browse visually or in detail.
-  **Acceptance Criteria:**
 - Toggle button: Map <-> List
 - Map pins show property markers
 - Tap marker → preview → detail
-  **Tasks:**
 - Integrate Google Maps or Mapbox
 - Display pins from listing data
 - Build toggleable component state

Story: Property detail view

- **User:** Renter
- **As a renter, I want** to view full property info so I can decide if I'm interested.

- **Acceptance Criteria:**
 - Carousel for images
 - Display rent, amenities, location, owner contact
- **Tasks:**
 - Detail screen layout
 - Carousel implementation
 - Owner info + "Contact" CTA

Epic 3: Communication & Chat (Week 5)

Story: In-app messaging between renter and owner

- **User:** All
- **As a renter/owner, I want** to chat in-app so I can ask questions and schedule visits.
- **Acceptance Criteria:**
 - Chat thread UI (owner ↔ renter)
 - Notifications for new messages
- **Tasks:**
 - Supabase Realtime or Stream setup
 - Build chat UI (list, thread)
 - Write to messages table

Story: Save/favorite a property

- **User:** Renter
- **As a renter, I want** to save properties I like so I can revisit them later.
- **Acceptance Criteria:**
 - Save button (toggle heart icon)
 - View saved list in profile

-  **Tasks:**
 - Toggle save function in favorites table
 - Build Saved tab UI
 - Sync saved state to UI

Epic 4: Notifications & Alerts (Week 6)

Story: New listing notification for saved searches

- **User:** Renter
- **As a renter, I want** to get notified when new listings match my saved search so I don't miss out.
-  **Acceptance Criteria:**
 - Create saved search query
 - Push notification on match
-  **Tasks:**
 - Save search criteria in DB
 - Background job to check for new matches
 - Expo Push Notifications integration

Epic 5: Monetization – AdMob + Premium (Week 6–7)

Story: Show AdMob ads to free users

- **User:** Renter
- **As a user, I want** to see ads occasionally so the app can stay free.
-  **Acceptance Criteria:**
 - Banner on list page
 - Interstitial ad on navigation

-  **Tasks:**
 - Integrate AdMob (Expo SDK)
 - Configure placement logic
 - Test live ads with AdMob test units

Story: Boost listing via Stripe/Razorpay

- **User:** Owner
- **As an owner, I want** to pay and boost my listing so it gets more visibility.
-  **Acceptance Criteria:**
 - Upgrade button in listing card
 - Checkout via Razorpay or Stripe
 - Flag "isPremium" in DB
-  **Tasks:**
 - Razorpay/Stripe setup
 - Payment button + flow
 - Update listing UI with "Premium" badge

Epic 6: Analytics + Admin (Week 8)

Story: View listing performance (views, inquiries)

- **User:** Owner
- **As an owner, I want** to see how many views and chats my property has so I can gauge interest.
-  **Acceptance Criteria:**
 - Show views counter
 - Show number of chats
-  **Tasks:**
 - Track views with Supabase counter

- Count messages from messages table

Story: Admin listing moderation tool (Optional Phase 2+)

- **User:** Admin
- **As an admin, I want** to review and flag listings so that I can maintain quality.
-  **Acceptance Criteria:**
 - Admin-only access panel
 - Flag/approve listings
-  **Tasks:**
 - Create admin role in Supabase
 - Build admin panel with filters
 - Add moderation flags in DB

Epic 7: Future Growth Features (Phase 3+)

Story: AI property search

- **User:** Renter
- **As a user, I want** to search using natural language so I can find listings faster.
-  **Acceptance Criteria:**
 - Input: “2BHK under 20k near Bandra”
 - Parse into structured query
-  **Tasks:**
 - Use OpenAI or custom NLP to parse input
 - Map to search filters
 - Return results + fallback if nothing found

Story: Virtual tour support

- **User:** Owner
- **As an owner, I want to upload video tours so my property stands out.**
-  **Acceptance Criteria:**
 - Upload MP4 or YouTube link
 - Render in detail page
-  **Tasks:**
 - Update listing schema: videoUrl
 - Video upload + playback logic

JIRA Setup Suggestion

Epics:

1. Auth & Onboarding
2. Listings (CRUD + Discovery)
3. Chat & Communication
4. Notifications
5. Monetization
6. Analytics + Admin
7. Future Growth

Labels to Use:

- frontend, backend, design, infra, mobile, web, critical, nice-to-have

Role Based Authentication and Access

1. User Roles & Definitions.

Role	Description
 Renter	End-user looking to search, save, and contact properties for rent
 Landlord/Owner	Individual property owner listing a flat/house
 Broker (Optional)	Agent listing and managing multiple properties on behalf of others
 Admin	Internal app admin responsible for moderating listings, verifying users, and managing analytics

 **Note:** Landlord and Owner are used interchangeably in your current model. Brokers are optional but can be added later with slightly different permissions.

2. Login Requirements (All User Types)

User Type	Login Method	Requirements	Notes
Renter	Email + OTP (via Supabase)	Valid email address	No password; OTP only for smoother onboarding
Landlord/Owner	Email + OTP (via Supabase)	Valid email + Listing Profile	Optional phone verification in future
Broker (Optional Role)	Email + OTP	Business/KYC info for verification	May require additional fields like GST/license
Admin	Email + Password or invite-only	Internal access	Should not use OTP; use admin panel login securely

 OTP-based login = Fast, secure, no password fatigue

 Admins should use password + 2FA to avoid risks

3. Feature Access & Role-Based Permissions.

Feature Access Matrix

Feature	Renter	Owner	Broker	Admin

 <i>Search Properties</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Map/List View</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Save Favorites</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>In-App Chat</i>	<input checked="" type="checkbox"/> (with owner)	<input checked="" type="checkbox"/> (with renters)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Add Listing</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Edit/Delete Listing</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>View Insights (views/chats)</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Promote Listing (Boost)</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Receive Notifications</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Access Admin Panel</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Flag/Moderate Listings</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Access Dashboard (analytics)</i>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 <i>Upload Docs (KYC, Lease)</i>	<input checked="" type="checkbox"/>	Optional	Optional	<input checked="" type="checkbox"/>

Feature Control Logic

- Renter** → Focused on browsing, saving, and contacting. Can't create or edit listings.
- Owner** → Can post, manage, and promote listings. Can chat with interested renters.
- Broker** → Like an owner, but may manage multiple properties. Optional verification/KYC required.
- Admin** → Can do everything. Includes moderation, analytics, and user management.

4. Additional Login or Access Considerations

Area	Need	Description
<input checked="" type="checkbox"/> <i>Regular Users (Renter/Owner)</i>	OTP Only	Supabase Email OTP Login
<input checked="" type="checkbox"/> <i>Admin Users</i>	Password Login + Role-based Access	Prefer invite-only admin panel with 2FA

? Future: Super Admin	Optional	For managing affiliate links, revenue, and referral logic
? Optional: Support Staff	Optional	Limited access to resolve tickets or disputes

Suggestions to Enhance Control & Security:

- Use Supabase roles/claims to tag users (renter, owner, broker, admin)
- Create a role field in the users table
- Restrict API endpoints using RLS (Row-Level Security) policies based on role
- Consider **owner verification (KYC upload)** to unlock "Verified Badge" and premium access.