

(4)User Stories

User Story 1:

As a tenant, I want my profile to be verified through ID and email so that I can trust others and be trusted in return.

Acceptance Criteria

- The system must allow government ID and institutional email upload.
- Verified profiles should display a badge.

Failures

- Invalid ID: “The ID provided is not valid. Please upload a valid ID.”
- Duplicate Email: “Email already exists in the system. Please log in instead.”
- Invalid Email: “Invalid email. Please enter a valid email address.”
- System Failure: “Oops! We couldn’t verify your profile. Please try again later.”

INVEST Justifications

- Independent: Verification can be built independently of other modules like property listings or messaging.
- Negotiable: The order of verification and optional vs. mandatory verification steps can be adjusted.
- Valuable: Builds trust between tenants and property owners, reducing risks of fraud.
- Estimable: Clear scope—upload ID, verify email, update profile with status.
- Small: Focused only on verification, not on additional identity checks.
- Testable: Rules for valid ID, email confirmation, and badge display are easy to test.

Identified from:

- Functional requirements - Authentication & User Management
- Survey Ans :- Need trustworthy roommate + Safety and trust concerns (Verified ID)

User Story 2

As a tenant, I want to filter flats by location and budget so that I can quickly find options near my workplace/college within my price range.

Acceptance Criteria

- Search must include a location radius filter.
- The budget filter must support ranges (e.g., 5,000–15,000).
- Only flats within specified constraints should appear.

Failures

- Invalid Location: “We couldn’t find flats in this location. Try another area.”
- No Results for Budget: “No flats match your budget range. Try adjusting filters.”
- Incorrect Format: “Invalid input format. Please enter a valid location or numeric budget.”
- System Error: “Oops! Something went wrong while applying filters. Please try again later.”

INVEST Justifications

- Independent: Filtering can be implemented without dependency on recommendation or chat modules.
- Negotiable: Range options (exact amount vs. slider vs. categories) can be adjusted.
- Valuable: Saves tenants time by letting them find affordable, nearby flats quickly.
- Estimable: Scope is well-defined—filter by two parameters (location and budget).
- Small: Only covers filtering, not saving searches or notifications.
- Testable: Easy to test with specific inputs and expected filtered results.

Identified from:

- Functional requirement - Searching
- Survey Ans: Budget friendly flats

User Story 3

As a tenant, I want to specify my flatmate preferences (cleanliness, lifestyle, gender, diet) so that I am matched with compatible roommates.

Acceptance Criteria

- Users must be able to select hygiene, smoking, drinking, diet (Veg/Non-Veg), and gender preferences.
- Matches should prioritize compatibility scores.
- The system must allow optional hobbies/cultural compatibility fields.

Failures

- Missing Fields: “Please select at least one preference to continue.”
- Invalid Input: “Invalid preference value. Please choose from the available options.”
- No Matches: “No compatible flatmates found at the moment. Try adjusting your preferences.”
- Compatibility Score Calculation Error: “We couldn’t calculate compatibility scores at the moment. Please try again later.”

INVEST Justifications

- Independent: Can be implemented independently, interacting only with the user database.
- Negotiable: Categories (e.g., add pets, study habits) can be refined later.
- Valuable: Helps tenants avoid conflicts and improves their living experience.
- Estimable: Scope is clear—collect preferences, store them, and feed into the match algorithm.
- Small: Avoids complexity, keeping it limited to preference recording.
- Testable: Each preference can be tested for correct storage and retrieval in matches.

Identified from:

- Functional requirement – Matching
- Survey Ans – Matching lifestyle / habits with roommate

User Story 4

As a tenant, I want to choose whether I'm searching for a flat, a flatmate, or both so that the app aligns with my current needs.

Acceptance Criteria

- Users must select one of three modes at search start: flat only, flatmate only, flat + flatmate.
- Results must adapt to selected mode.

Failures

- No Mode Selected: "Please select a search mode (Flat, Flatmate, or Both) to continue."
- No Results for Selected Mode: "No results found for the chosen mode. Try again later."

INVEST Justifications

- Independent: Can be developed separately from features like recommendations or verification.
- Negotiable: The available modes (Flat, Flatmate, Both) can be expanded later to include advanced search types.
- Valuable: Ensures users don't waste time browsing irrelevant results, making the search more efficient.
- Estimable: Clearly scoped—requires mode selection and adapting results accordingly.
- Small: Narrow in focus, covering only mode choice and filtered results.
- Testable: Easy to verify by checking whether results match the selected mode.

Identified from:

- Functional Requirements – Searching

User Story 5

As a tenant, I want to read reviews of flats and flatmates so that I can avoid unreliable landlords and mismatched roommates.

Acceptance Criteria

- Users can leave ratings/reviews after tenancy.
- Reviews must be tied to verified stays (to avoid fake entries).
- Average ratings should be displayed on listings/profiles.

Failures

- Unverified Review Attempt: “Only verified tenants can leave a review. Your stay must be completed first.”
- Empty Review Text: “Review cannot be empty. Please provide your feedback.”
- Average Rating Calculation Error: “Unable to calculate average rating at the moment. Please refresh the page later.”
- No Reviews Yet: “No reviews available for this flat/flatmate yet. Be the first to leave feedback!”

INVEST Justifications

- Independent: Works as a separate module, connecting only to completed tenancy records.
- Negotiable: Optional fields (detailed comments, images) can be refined later.
- Valuable: Helps tenants make informed decisions, reducing the risk of mismatched roommates or unreliable flats.
- Estimable: Scope is clear—submit/view reviews, tie to verified stays, calculate average ratings.
- Small: Focuses only on the review submission and display; does not include advanced analytics or moderation.
- Testable: Can be tested with verified and unverified tenants leaving reviews, ensuring correct behavior.

Identified from:

- Functional Requirement - Feedback & Report Management
- Survey Report – Safety and trust concerns (reviews from past)

User Story 6

As a tenant, I want to chat with potential flatmates through secure in-app messaging so that I don't have to share my phone number right away.

Acceptance Criteria

- Messaging must be private and secure.
- Contact details should only be shared after both parties agree.
- Users must be able to block/report others.

Failures

- Message Not Sent: "Your message could not be delivered. Please try again later."
- Unauthorized Access: "You are not authorized to view this chat. Please log in and try again."
- Contact Sharing Without Consent: "Contact details cannot be shared until both parties agree."
- Blocking Failure: "Unable to block this user at the moment. Please try again."

INVEST Justifications

- Independent: Operates independently using user IDs; does not rely on search or recommendation engines.
- Negotiable: Optional features like emoji support, media sharing, or typing indicators can be implemented later.
- Valuable: Ensures users can communicate safely without exposing personal information and reducing risk.
- Estimable: Clear development scope—chat, privacy enforcement, block/report options.
- Small: Limited to one-to-one messaging; does not include group chats or full chat history export.
- Testable: Can be tested for sending, receiving, privacy, blocking, and reporting scenarios.

Identified from:

- Functional Requirement – Other → 2
- Survey Report – Safety and trust concerns (In App chat)

User Story 7

As a tenant, I want to type search queries like “quiet non-smoker near XYZ College” so that I can find matches more naturally.

Acceptance Criteria

- The app must support NLP-based search input.
- Results must match keywords with preference filters with 90% accuracy.
- Users must be able to refine NLP results with standard filters.

Failures

- NLP Parsing Failure: “We couldn’t understand your query. Please try rephrasing your search.”
- No Matching Results: “No flats or flatmates match your search criteria. Try adjusting your filters.”
- Incorrect Mapping of Keywords: “Some preferences from your search could not be applied. Check your filters and try again.”
- Filter Refinement Failure: “Unable to apply filters to NLP results at the moment. Please try again later.”

INVEST Justifications

- Independent: Works independently, consuming only the flat/flatmate database for search results.
- Negotiable: Accuracy threshold (90%) can be increased over time, and fall-back search options can be refined.
- Valuable: Improves user experience by letting tenants describe their preferences naturally instead of manually applying multiple filters.
- Estimable: Well-bounded development effort—NLP parser + keyword mapping + filter application.
- Small: Limited to NLP query parsing and basic filtering.
- Testable: Queries with known keywords can be tested for correct mapping and expected result sets.

Identified from:

- Functional Requirement – Searching → 1, Matching → 3
- Survey Report – 50% of tenants say NLP would be highly useful.

User Story 8

As a first-time tenant, I want guidance on renting basics (documents, agreements, safety checks) so that I avoid mistakes.

Acceptance Criteria

- Information of different advocates should be provided.

Failures

- Information Missing: “Guidance on this topic is currently unavailable. Please check back later.”
- Advocate Details Unavailable: “No advocate information available at the moment. Try again later.”

INVEST Justifications

- Independent: Guidance module can be implemented independently without relying on property listings or chat modules.
- Negotiable: The number of topics or advocates provided can be adjusted based on user feedback.
- Valuable: Helps first-time tenants avoid costly mistakes, increasing confidence and trust in the platform.
- Estimable: Scope is clear—provide articles, instructions, and advocate details.
- Small: Narrow focus on general guidance and details about advocates.
- Testable: Each guidance article and advocate link can be verified for accessibility and correctness.

Identified from:

- Functional Requirement – Other → 1

User Story 9

As a tenant, I want to know the features around a property so that I can choose the most convenient property.

Acceptance Criteria

- The places around the property must be listed by the property owner.
- User must be able to filter based on these features.

Failures

- No Features Listed: “No nearby feature information is available for this property yet.”
- No Results for Selected Features: “No properties match the selected features. Try adjusting your filter criteria.”

INVEST Justifications

- Independent: Feature listing and filtering can be implemented without depending on chat, verification, or review modules.
- Negotiable: Optional features like distance radius or feature rating can be added later.
- Valuable: Helps tenants select properties based on convenience, saving time and improving satisfaction.
- Estimable: Scope is clear—owners add features, tenants view and filter by them.
- Small: Focused only on listing and filtering; map integration can come later.
- Testable: Each feature listed can be verified, and filters can be tested for expected results.

Identified from:

- Functional Requirement – Searching → 5
- Interview Property Owner – Dharmesh Kachhia, Question 1

User Story 10

As a tenant, I want a roommate who has similar habits as mine so that we can be comfortable with each other.

Acceptance Criteria

- The default suggestions to the users without filters should be roommates with similar habits.
- Upon further filters, the filters given by the user would be active and the similarity would be given less priority but not zero.

Failures

- No Default Matches: “No roommates with similar habits found. Try adjusting your search criteria.”
- Filter Overrides Similarity: “Filters applied exclude all similar habit matches. Try broadening your filters.”
- Similarity Calculation Error: “Could not calculate habit similarity at the moment. Please try again later.”

INVEST Justifications

- Independent: Matching algorithm can work independently of messaging or verification modules.
- Negotiable: The weight of habit similarity vs. filters can be adjusted in future analysis.
- Valuable: Increases tenant satisfaction and reduces roommate conflicts by suggesting compatible matches.
- Estimable: Scope is clear—calculate similarity, apply filters, generate ranked suggestions.
- Small: Limited to default suggestions and filtered ranking; advanced recommendation engine enhancements can come later.
- Testable: Default and filtered suggestions can be validated using sample profiles with known habits.

Identified from:

- Functional Requirements – Matching → 1
- Survey – 50% people are concerned about cleanliness habits and 68% about lifestyle choices.

User Story 11

As a tenant, I want a roommate who comes from a similar city as mine so that we have similar culture.

Acceptance Criteria

- Mostly all cities should be listed in the city selection feature.

Failures

- City Not Listed: “The city you selected is not available. Please choose another city from the list.”
- No Matches in Selected City: “No roommates found from your selected city. Try adjusting your filters or city preference.”

INVEST Justifications

- Independent: Works as a standalone module using only tenant profile city data.
- Negotiable: Supported city list can be expanded or filtered based on user feedback.
- Valuable: Supports cultural compatibility, reducing potential conflicts among roommates.
- Estimable: Scope is clear—list cities, select tenant city, prioritize matches by city.
- Small: Limited to city-based matching; additional ranking features can be added later.
- Testable: City selection and resulting matches can be tested with sample city data and tenant profiles.

Identified from:

- Functional Requirements - Matching

User Story 12

As a tenant, I want a notification to be sent to the property owner/roommate when I want to connect with them so that I can quickly communicate with them.

Acceptance Criteria

- Notification email should be immediately sent to the person of interest.
- The user profile of the interested user should be communicated.

Failures

- Notification Not Sent: “Your connection request could not be sent. Please try again later.”
- Profile Details Not Communicated: “Tenant profile details could not be attached to the notification. Try again.”
- Email Delivery Failure: “Notification could not be delivered via email. Please check the email address or try again.”

INVEST Justifications

- Independent: Works independently using only user profile and contact information.
- Negotiable: Type of notification (email, push, SMS) can be adjusted or expanded later.
- Valuable: Enables faster communication, improving tenant engagement and reducing delays.
- Estimable: Scope is clear—trigger notification, attach profile details, send email/push.
- Small: Only handles immediate notification delivery.
- Testable: Tests can check that notifications are triggered correctly for valid users and that failures are handled gracefully.

Identified from:

- Functional requirement – Notification System

User Story 13

As a tenant, I want the admin to block malicious accounts so that I can safely get a roommate.

Acceptance Criteria

- A profile which has been reported must be conveyed to the admin.
- Admin must be able to ban the user and blacklist the email.

Failures

- Report Not Conveyed: “The reported profile could not be sent to admin. Please try again later.”
- Admin Action Failure: “Unable to ban this user at the moment. Please try again.”

INVEST Justifications

- Independent: Reporting and banning are independent features that interact only with user profiles and admin privileges.
- Negotiable: Additional actions (temporary suspension, warning) can be added in future iterations.
- Valuable: Reduces risk of tenant harassment or fraud, increasing platform credibility.
- Estimable: Effort is predictable—notification to admin, ban action, blacklist update.
- Small: Covers only malicious account handling; advanced analytics or automated detection can come later.
- Testable: Reporting, banning, and blacklisting can be verified using test accounts.

Identified from:

- Functional Requirements – Feedback and Report Management

All the upcoming user stories have been extracted from interviews of property owners

User Story 14: Nearby properties

As a property owner, I want a feature that showcases the nearby facilities so that my property becomes more attractive to tenants.

Acceptance Criteria

- System must allow owners to input nearby landmarks, transport, and facilities.
- Tenants must be able to see facilities listed on the property details page.

Failures

- Missing Facility Input: “Please provide at least one nearby facility before saving.”
- Display Error: “Facilities could not be displayed on property page. Please try again later.”
- System Failure: “Oops! We couldn’t save facility details. Please try again.”

INVEST Justification

- Independent: Facility listing can be implemented separately from tenant matching or chat modules.
- Negotiable: Facility types (landmarks, transport, amenities) can be adjusted or extended.
- Valuable: Makes properties more attractive, helping owners get tenants faster.
- Estimable: Scope is clear—input, save, and display nearby facilities.
- Small: Focused solely on facility listing and display.
- Testable: Can be tested by entering facilities and verifying display for tenants.

User Story 15: Property Visibility

As a property owner, I want my property to be suggested more so that I quickly get a tenant.

Acceptance Criteria

- Properties with complete profiles (photos, description, rent details) should be prioritized.
- Suggested properties must be shown on tenant's search results based on relevance (location, budget, filters).

Failures

- Incomplete Profile: "Property cannot be suggested because required details are missing."
- Search Ranking Error: "Property could not be displayed in suggested results. Please check details."
- System Failure: "Oops! Unable to prioritize property at this moment."

INVEST Justification

- Independent: Works independently from chat, tenant verification, or legal support modules.
- Negotiable: Prioritization criteria can be adjusted (e.g., premium listing, boosting options).
- Valuable: Increases chances of renting out properties quickly.
- Estimable: Well-defined scope—profile completeness check + prioritization algorithm.
- Small: Only involves visibility logic, not notifications or recommendations.
- Testable: Testable by creating complete/incomplete property profiles and checking suggestions.

User Story 16: Tenant Matching

As a property owner, I want to have tenants who are according to my preferences so that my relationship with the tenant is cordial.

Acceptance Criteria

- Owner can specify preferences (e.g., gender, occupation, lifestyle, diet).
- Matching algorithm must filter tenants based on these preferences.
- Owners should see a compatibility score for each tenant profile.

Failures

- Missing Preferences: “Please select at least one preference to continue.”
- No Matching Tenants: “No tenants match your preferences at the moment. Try broadening filters.”
- Compatibility Calculation Error: “Could not calculate tenant compatibility. Please try again later.”

INVEST Justification

- Independent: Matching algorithm can work independently using tenant profiles.
- Negotiable: Preference categories and weights can be adjusted in future iterations.
- Valuable: Improves tenant-owner relationships, reducing conflicts.
- Estimable: Scope is clear—collect preferences, filter tenants, calculate scores.
- Small: Focused on preference-based filtering; advanced AI recommendations can come later.
- Testable: Enter preferences and verify matched tenant list and compatibility scores.

User Story 17: Minimum Stay

As a property owner, I want the tenant to stay for a minimum time so that I don't need to search for tenants repeatedly.

Acceptance Criteria

- Owner must be able to set a minimum stay duration (e.g., 3 months, 6 months).
- System must filter out tenants who do not agree to the minimum stay.
- Contracts must clearly display the minimum stay condition before confirmation.

Failures

- Invalid Duration: "Please enter a valid minimum stay duration."
- Tenant Rejection: "Selected tenant does not agree to the minimum stay requirement."
- System Error: "Could not enforce minimum stay filter. Please try again."

INVEST Justification

- Independent: Can be implemented separately from tenant verification or messaging.
- Negotiable: Minimum stay rules can be adjusted by the owner.
- Valuable: Reduces turnover and effort in searching for tenants repeatedly.
- Estimable: Clear scope—set, filter, and enforce minimum stay.
- Small: Only involves minimum stay logic.
- Testable: Test by setting different durations and verifying tenant eligibility.

User Story 18: Tenant Ratings

As a property owner, I want the tenant to have a rating according to previous rentals so that I can get good tenants.

Acceptance Criteria

- Tenants must have visible ratings/reviews from past landlords.
- Ratings must only be allowed from verified rental agreements.
- System should calculate average rating and show it on tenant profiles.

Failures

- No Ratings: “This tenant has no prior ratings yet.”
- Invalid Review Submission: “Only verified landlords can submit ratings.”
- Calculation Error: “Average rating could not be computed. Please try later.”

INVEST Justification

- Independent: Rating module can work independently of property listing or chat.
- Negotiable: Display format (stars, score) can be adjusted.
- Valuable: Helps owners identify reliable tenants.
- Estimable: Scope is clear—submit, verify, and display ratings.
- Small: Focused on ratings only.
- Testable: Testable by submitting verified/unverified reviews and checking display.

User Story 19: Tenant Verification

As a property owner, I want the tenant to be verified so that I am secure and my house stays safe.

Acceptance Criteria

- Tenant must verify identity using government ID and institutional/work email.
- Verified tenants must display a “Verified” badge.
- Properties should allow filtering/search by verified tenants only.

Failures

- Invalid ID/Email: “Tenant verification failed. Invalid ID or email.”
- Unverified Tenant Filtering: “Could not apply verified tenant filter.”
- System Failure: “Tenant verification service is currently unavailable.”

INVEST Justification

- Independent: Verification works independently of property listing or ratings.
- Negotiable: Verification steps (ID/email) can be updated later.
- Valuable: Ensures property safety and builds trust.
- Estimable: Scope—verify identity, show badge, filter by verification.
- Small: Limited to tenant verification logic.
- Testable: Test ID/email verification, badge display, and filtering.

User Story 20: Legal Support

As a property owner, I want a list of advocates so that my rent agreement is made easily.

Acceptance Criteria

- Owners must be able to browse advocates/legal advisors within their city.
- Contact details of advocates must be available inside the app.

Failures

- No Advocates Found: “No advocates available in your city. Try later or expand search.”
- Contact Details Missing: “Advocate contact information could not be loaded. Try again.”
- System Failure: “Unable to fetch advocate list at this time.”

INVEST Justification

- Independent: Legal support can be implemented separately from property or tenant modules.
- Negotiable: Advocates list and details can be expanded or updated later.
- Valuable: Simplifies rent agreements and reduces legal hassles.
- Estimable: Scope—list advocates, display contact info.
- Small: Focused solely on legal support.
- Testable: Test by searching for advocates and verifying contact info.