ARTIFICIAL INTELLIGENCE SYSTEMS

Intelligent Renting Recommendation System

Group 14
Chen Sigen A0326351L
Qi Yilin A0328843W
Ru Yanjie A0295237W
Yao Yiyang A0294873L

Project Report

Table of Contents

Executive Summary

1. Project Background & Problem Statement

Singapore, a global education hub, attracts a significant number of international students. However, securing suitable rental accommodation presents a major challenge for these students. The process is often time-consuming, fragmented, and characterized by information asymmetry. Prospective tenants must navigate complex trade-offs between multiple, often competing, objectives such as proximity to their educational institution, commute time, rental budget, housing type, and availability of nearby amenities. Existing rental platforms lack the intelligent, personalized capabilities required to address the specific needs of this demographic effectively.

2. Project Objective & Proposed Solution

To address this gap, our team has developed the Intelligent Recommendation System, an AI-powered rental platform specifically designed for international students in Singapore. The core objective of this system is to streamline the property search process by intelligently identifying accommodation that best matches a user's unique and multi-faceted preferences. It transforms a traditionally manual and frustrating search into an efficient, data-driven decision-making process. A key differentiator of our system is its dual-mode interface, catering to both structured and natural user interactions.

3. System Methodology & Core Workflow

Our solution operates through a robust, user-centric workflow:

- Dual-Mode Input Interface: The system accepts user requirements through both a structured form for precise filtering and a conversational Natural Language Processing (NLP) interface, allowing users to express their needs freely and intuitively (e.g., "I need a cheap room under 1200$ near NUS, maybe 30 minutes away, and close to an MRT").
- Intelligent Filtering & Multi-Objective Optimization: The system processes the input (either structured or parsed from natural language) to filter the property database. It then employs a sophisticated weighted scoring model to rank properties based on three critical dimensions: Rent, Location, and Facility convenience, weighted by user-defined importance.
- Explainable AI Recommendations: A pivotal feature of our system is its ability to generate personalized explanation for each recommendation. For every listed property, the system provides a clear, data-driven rationale (e.g., "This apartment is recommended because it offers the shortest commute to your school (15 mins) and is within your budget, despite being slightly further from the MRT"), thereby building user trust and transparency.

4. Key Outcomes & Value Delivered

The successful development of the Intelligent Recommendation System demonstrates:

- Enhanced Usability & Accessibility: The conversational NLP interface makes the system accessible to users regardless of their technical proficiency, accommodating a wide range of query styles.
- Decision Intelligence & Transparency: The combination of multi-objective optimization and explainable recommendations not only identifies optimal listings but also justifies *why* they are optimal, empowering users to make confident, informed decisions.
- High Efficiency: The system processes complex user queries and returns highly relevant, explained recommendations within seconds, dramatically improving the entire search and evaluation workflow.

5. Conclusion

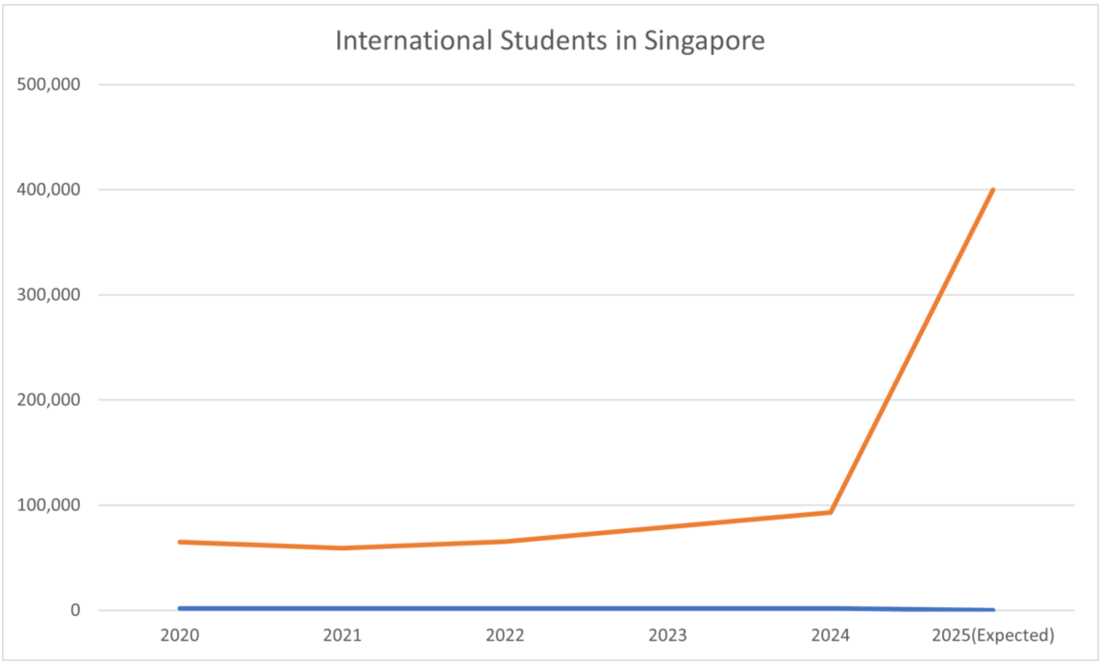
In conclusion, the Intelligent Recommendation System provides a highly efficient, intelligent, and user-adaptive rental solution for international students in Singapore. It transcends traditional filtering tools by integrating natural language understanding and explainable AI, creating a comprehensive decision-support system poised to significantly enhance the student accommodation search experience and facilitate a smoother transition to academic life in Singapore.

Business Case & Market Research

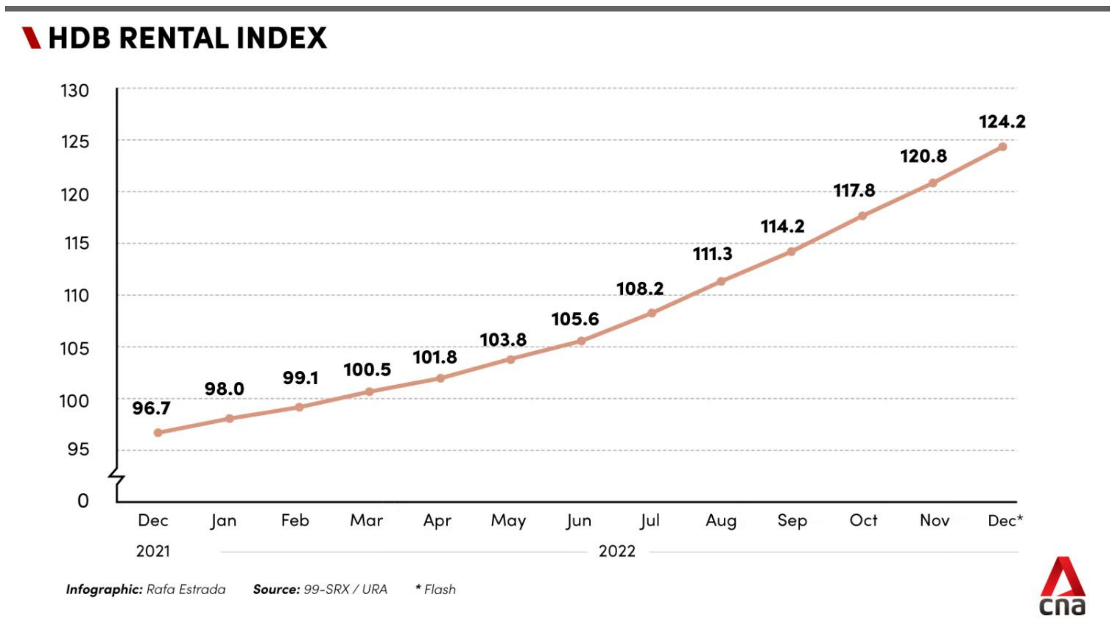
1. Market Overview & Problem Significance

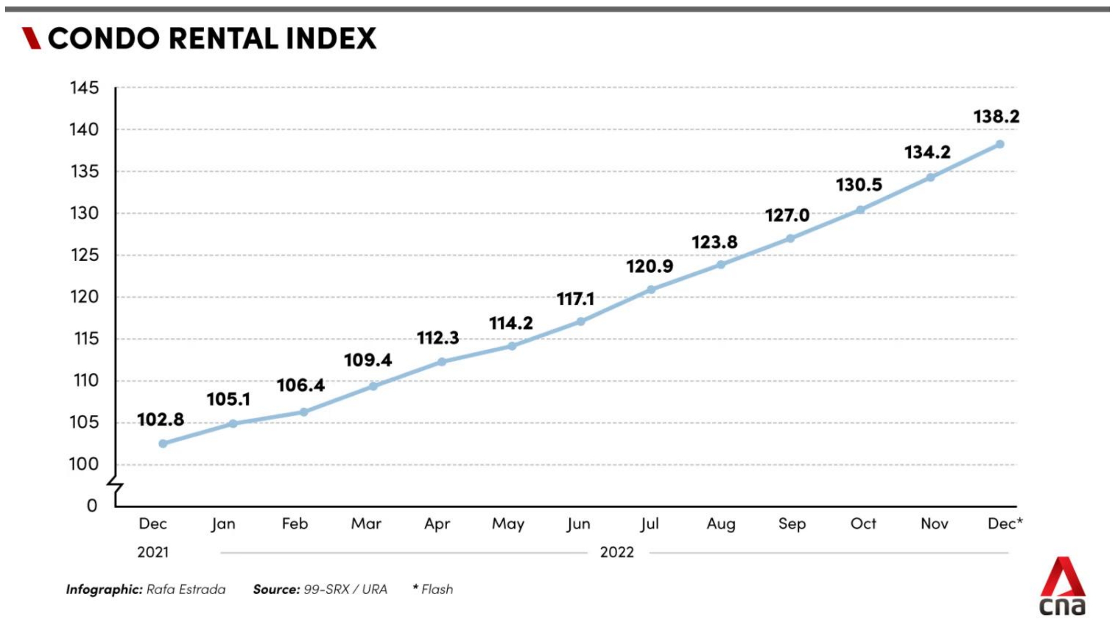
The rental market for international students in Singapore is substantial, growing, and characterized by inherent inefficiencies.

- Substantial and Growing Market Size: The international student population in Singapore is both sizable and expanding rapidly. As of 2023, Singapore hosted approximately 79,300 to 80,000 international students. This figure represents a remarkable year-on-year increase of approximately 25% from 2022, translating to an influx of over 10,000 new international students within a single year, and a huge increase that leads to an expectation of 400,000 in 2025. This creates a continuous and renewable demand for rental accommodation.



- The Core Problem: As outlined in the introduction, this growing student body faces a rental search process that is highly fragmented. Students are forced to navigate multiple platforms, filter through irrelevant listings, and manually reconcile conflicting priorities (e.g., cost vs. location). This process is not only time-consuming but also leads to decision fatigue, suboptimal choices, and heightened stress during what is already a major life transition.





- Economic Impact: The inefficiency in the market represents a tangible economic cost in terms of time wasted and potential financial loss from choosing unsuitable accommodation.

2. Analysis of Existing Solutions & Gap Identification

Currently, students primarily rely on a few types of solutions, all of which have significant limitations. The intense market demand is exacerbating these limitations, as seen in rising rental costs and increased competition for housing.

- General Property Portals (e.g., PropertyGuru, 99.co):
  - Strengths: Large inventory of listings.
  - Weaknesses/Gap: They are designed for the general public, not students. They lack specific filters for "commute time to a specific university" and cannot intelligently

balance multiple, weighted preferences. The search results are often not personalized. Furthermore, students are confronted with a challenging market where apartment rents can average S$3,500 per month and HDB rents in popular areas near universities have seen significant increases, putting pressure on student budgets.

- Social Media & Community Groups (e.g., Facebook, Telegram):
    - Strengths: May offer direct contact with landlords or current tenants.
    - Weaknesses/Gap: Highly unstructured, unverified information, prone to scams, and incredibly time-intensive to monitor. No systematic filtering or ranking capabilities. The need to secure housing months in advance, as is the case near NUS where students may need to lock in a rental 3 months ahead, adds to the pressure.
- University Noticeboards & Housing Offices:
    - Strengths: Often provide verified and trusted listings.
    - Weaknesses/Gap: Inventory is typically very limited and cannot meet the full demand of the growing student population, and the search and discovery process is often manual and not integrated with broader market data.
- Emerging Co-living Operators:
    - Strengths: Offer all-inclusive, flexible packages.
    - Weaknesses/Gap: This segment, while growing, still only represents a fraction of the total market. It also tends to be a premium option, with all-inclusive monthly rents typically ranging from S$3,000 to S$4,000, which may be above the budget for many students.

Our system directly addresses these gaps by providing a centralized, intelligent, and student-centric platform that actively solves the multi-objective optimization problem at the heart of the rental search, helping students navigate a competitive and expensive market efficiently.

3. Value Proposition of the Intelligent Recommendation System

Our system delivers unique value to primary users, international students by directly addressing the market challenges:

- Efficiency & Time-Saving: Drastically reduces apartment search time from days/hours to minutes by intelligently navigating the vast and fragmented rental market.
- Cost-Effectiveness: Helps students identify housing that maximizes their budget in a market where rental costs are a primary concern, potentially preventing costly and unsuitable rental decisions.
- Informed Decision-Making: The multi-objective optimization model and explainable AI provide transparent, data-backed reasons for each recommendation, leading to more

confident and satisfactory choices amidst complex trade-offs.

- Reduced Stress & Improved Experience: The conversational NLP interface and personalized results create a smooth, user-friendly experience, lowering the barrier to entry and anxiety associated with finding a home in a new country.

4. Market Potential & Business Model Feasibility

The project demonstrates strong potential for sustainability and growth, backed by concrete market dynamics.

- Target Market: The primary target market is the entire population of over 79,000 international students in Singapore, a sizable and annually replenishing segment that has shown a capacity for rapid growth (25% YoY).
- Validated Demand: The booming co-living sector, which attributes 25% to 40% of its occupancy to international students and has attracted over S$1.4 billion in investments since 2022, clearly signals strong, unmet demand for tailored rental solutions that our system can capture and serve.
- Potential Business Models: The system can be monetized through several viable channels:
    - Freemium Model: Basic recommendations are free for students. Premium features (e.g., direct contact with top landlords, advanced market insights, priority support) could be offered via a subscription.
    - B2B Commission Model: Charge property agents or landlords a small commission for successful leases facilitated through the platform.
    - Featured Listings: Offer agents/landlords the option to promote their listings for higher visibility in the recommendation feed for a fee.
- Scalability: The core AI architecture is scalable. It can be expanded to include more schools, incorporate more data points (e.g., roommate matching), or even be adapted to other cities with a significant international student population.

5. Conclusion of Business Case

There is a clear, documented, and significantly growing market need for an intelligent rental recommendation system tailored to international students in Singapore. The market has swelled to approximately 80,000 students, with impressive year-on-year growth of 25%. Existing solutions are inadequate in addressing the core challenges of multi-criteria decision-making and personalized search in a competitive rental environment. The Intelligent Recommendation System is uniquely positioned to fill this gap, offering compelling value to both tenants and listers, with a clear path to becoming a sustainable and valuable service in the educational ecosystem of Singapore.

System Design

1. Project Scope & Objectives

1.1 Project Scopes

This project encompasses the design and development of an end-to-end, AI-powered rental recommendation system. The in-scope deliverables are defined as follows:

- Frontend Interaction: A user interface that supports dual-mode input, including a structured form and a natural language conversation interface.
- Backend Services: Core server-side logic responsible for request handling, property filtering, multi-objective optimization ranking, and the generation of explanatory recommendations.
- Data Layer: A structured database housing property listings with key attributes such as geographic location, monthly rent, facility information, and flat type, etc.
- Core Algorithms: The implementation of a weighted scoring model for multi-objective optimization and a fundamental Natural Language Understanding (NLU) module for parsing user intents and entities.

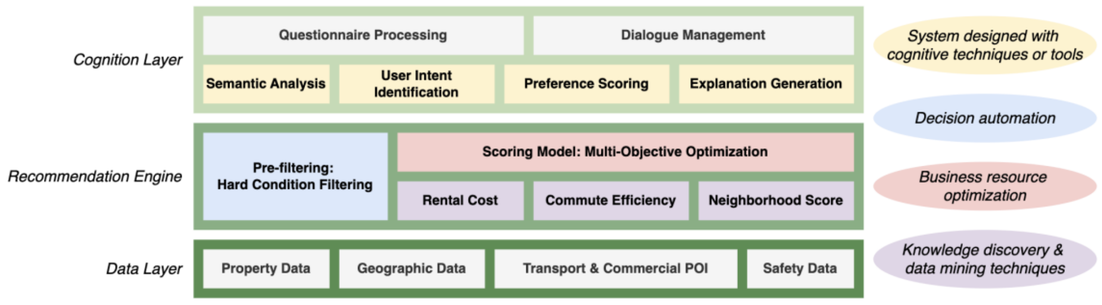
The project explicitly excludes facilitating actual property transactions, payment processing, facilitating direct communication between tenants and landlords/agents, or implementing a full-scale user identity verification system.

1.2 Core Objectives

The system is designed to fulfill the following core objectives:

- Functionality: To accurately capture and process users' multi-faceted rental preferences provided through both structured and unstructured inputs.
- Accuracy: To generate and return a list of recommended properties that demonstrates a high degree of relevance to the user's comprehensive and weighted criteria.
- Intelligence: To intelligently comprehend and trade off user-assigned priorities across different factors (rent, location, facilities) using a multi-objective optimization algorithm.
- User-Friendliness: To provide an intuitive user experience through conversational interaction and to foster trust by delivering transparent, explainable recommendations.
- Performance: To ensure system responsiveness, delivering the top 10 ranked results within seconds for an efficient user experience.

2. System Architecture


Cognition Layer
Questionnaire Processing
Dialogue Management
Semantic Analysis
User Intent Identification
Preference Scoring
Explanation Generation
System designed with cognitive techniques or tools
Recommendation Engine
Pre-filtering: Hard Condition Filtering
Scoring Model: Multi-Objective Optimization
Rental Cost
Commute Efficiency
Neighborhood Score
Decision automation
Business resource optimization
Data Layer
Property Data
Geographic Data
Transport & Commercial POI
Safety Data
Knowledge discovery & data mining techniques

2.1 Cognitive Layer: Natural language input/output

- Natural language interface powered by LLMs.
- Explainable reasoning based on knowledge.
- Transform user queries into structured optimization constraints.
- Provide explainable recommendations.

2.2 Recommendation Engine: Decision Automation

- Multi-objective optimization: combine user-defined preferences (e.g., price vs. distance vs. safety) to rank options.
- Recommendation algorithms: weighted scoring, Pareto optimization.

2.3 Data Layer: Knowledge Discovery & Data Mining

- Collect and clean historical rental data and neighborhood knowledge.
- Feature extraction: rental price trends, safety scores, amenity proximity, commuting times.
- Build structured user profiles for personalized recommendations.

System Implementation

1. System Components

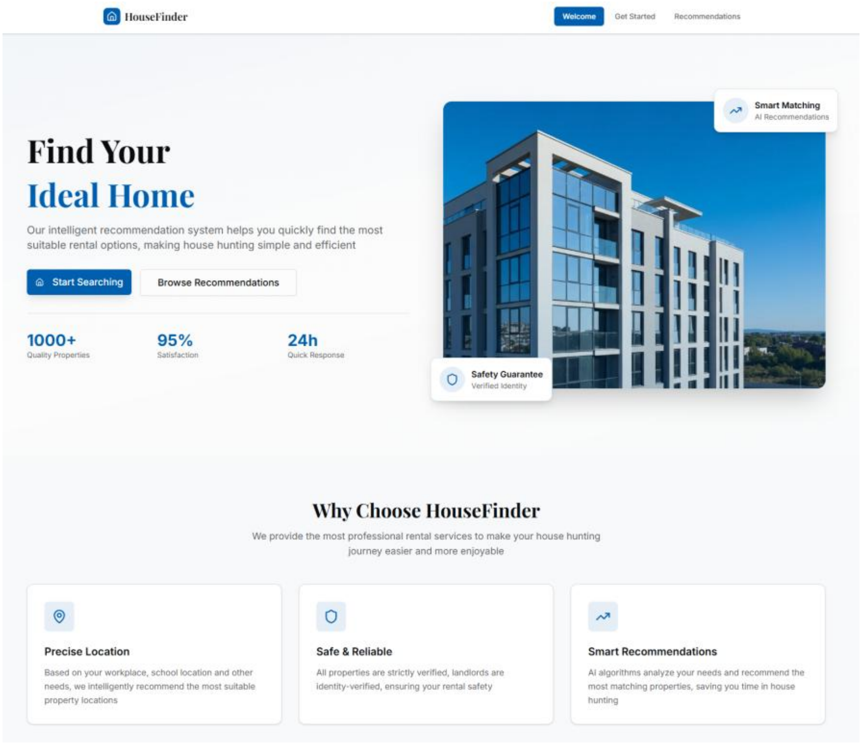
1.1 User Interface

User Interface Description for the Rental Recommendation System

Our platform is structured around five key pages that guide users from initial setup to exploring personalized rental listings. The interface is designed for clarity and ease of use.

1.1.1 Welcome Page

The Welcome page serves as the entry point. It features a clean layout with a brief introductory tagline and a few bullet points summarizing the system's core features, such as budget filtering and commute-aware recommendations. A prominent "Get Started" call-to-action button is the primary focus, which navigates users to the Preferences page. A persistent top navigation bar is present, providing direct links to the main sections of the site.



Find Your
Ideal Home

Our intelligent recommendation system helps you quickly find the most suitable rental options, making house hunting simple and efficient

Why Choose HouseFinder

We provide the most professional rental services to make your house hunting journey easier and more enjoyable

Precise Location

Based on your workplace, school location and other needs, we intelligently recommend the most suitable property locations

Safe & Reliable

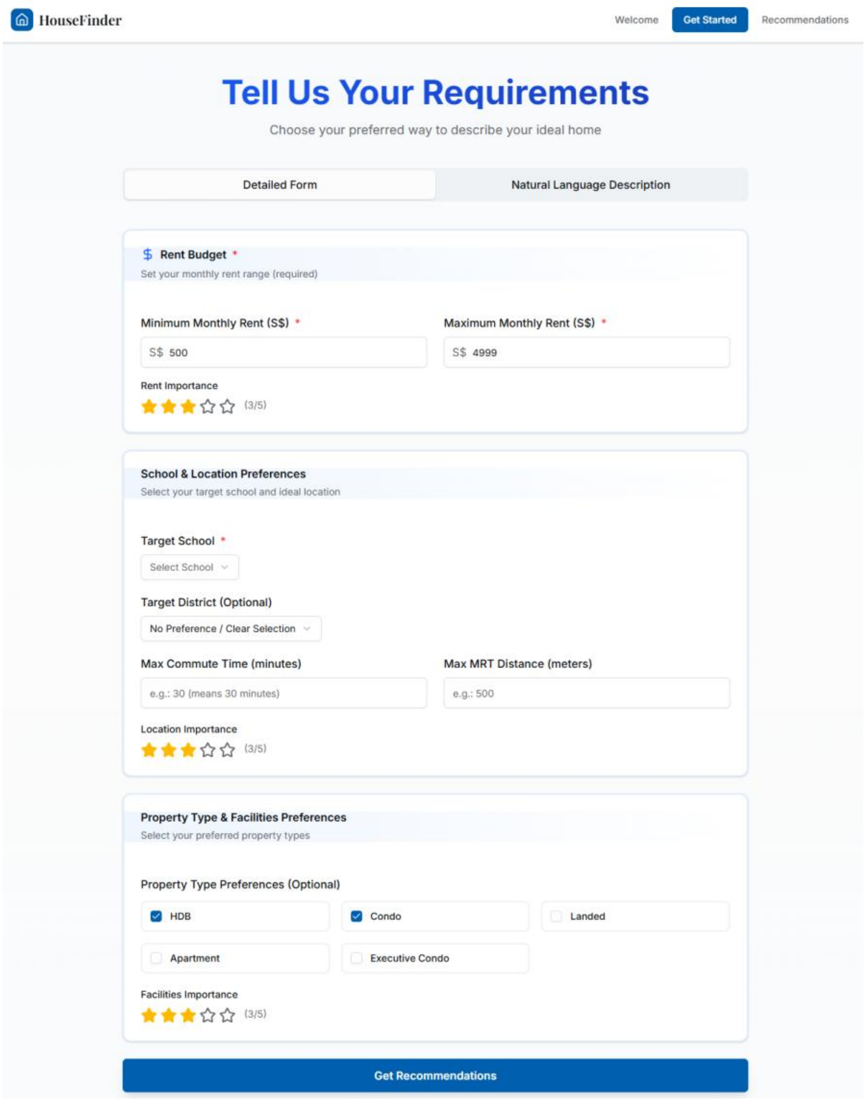
All properties are strictly verified, landlords are identity-verified, ensuring your rental safety

Smart Recommendations

AI algorithms analyze your needs and recommend the most matching properties, saving you time in house hunting

1.1.2 User Input / Preferences Page

This page is used to collect users' rental preferences, and adopts two input methods of "detailed form" and "natural language description" and switches with tabs to meet different user habits. In the form mode, the user needs to fill in the minimum and maximum monthly rent (numerical input with currency prefix), select the target school (required) and optional target district (clear), fill in the maximum commute time (minutes) and the maximum subway walking distance (meters), and select the housing type preference (such as HDB, Condo, Landed, etc.). At the same time, the importance of "rent", "location" and "facility" is set respectively through the star control. The form includes front-end validation (e.g., rental range and required school), displays loading status on submission and sends the form content to the back-end along with the unique identifier of the device, saves the returned recommendation data and source to local storage and skips to the recommendation page, and gives explicit warnings in case of network issues or missing fields returned by the back-end in case of errors. The natural language mode provides a text box of up to 500 characters with input length count and prompt information. It is recommended that the user include the price range and target school in the description, which can describe the preferred facilities, ideal location and transportation requirements. If the backend indicates that some information is missing, the interface will prompt the user with an alert and a pop-up window. The whole page focuses on usability: mandatory items are marked in red, optional support is clear, check and drop-down selection is clear, errors and prompts are displayed through the alert area or popup window, and user input is cached locally for easy recovery, so as to ensure that the submission process is clear, friendly and personalized housing recommendations can be obtained quickly.



Tell Us Your Requirements

Choose your preferred way to describe your ideal home

Rent Budget *

Set your monthly rent range (required)

Minimum Monthly Rent (S$) *: S$ 500

Maximum Monthly Rent (S$) *: S$ 4999

Rent Importance ★★★☆☆ (3/5)

School & Location Preferences

Select your target school and ideal location

Target School *: Select School

Target District (Optional): No Preference / Clear Selection

Max Commute Time (minutes): e.g.: 30 (means 30 minutes)

Max MRT Distance (meters): e.g.: 500

Location Importance ★★★☆☆ (3/5)

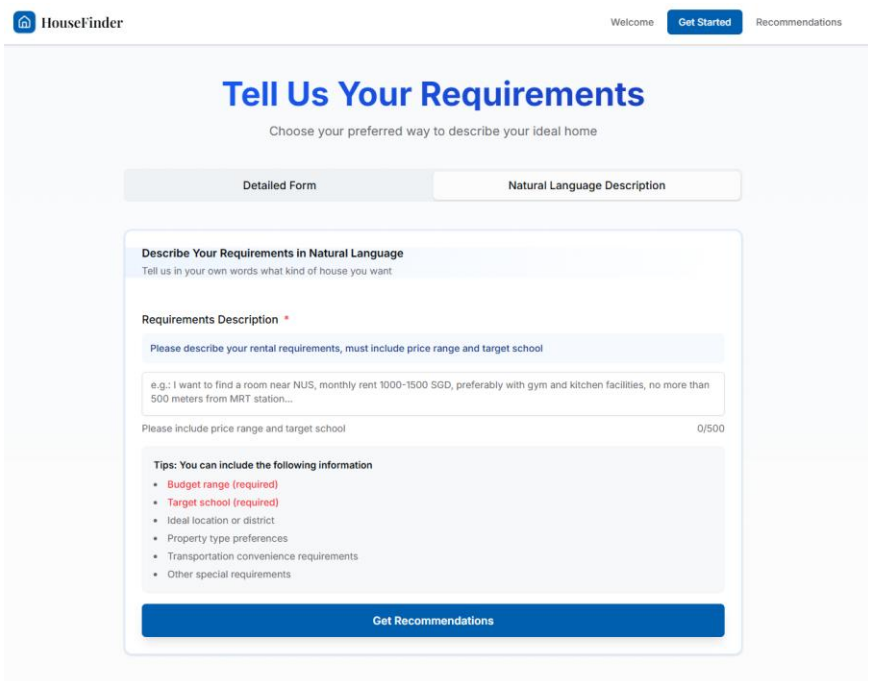
Property Type & Facilities Preferences

Select your preferred property types

Property Type Preferences (Optional)

[✓ HDB] [✓ Condo] Landed Apartment Executive Condo

Facilities Importance ★★★☆☆ (3/5)


HouseFinder

Welcome | Get Started | Recommendations

Tell Us Your Requirements

Choose your preferred way to describe your ideal home

Detailed Form | Natural Language Description

Describe Your Requirements in Natural Language

Tell us in your own words what kind of house you want

Requirements Description *

Please describe your rental requirements, must include price range and target school

e.g.: I want to find a room near NUS, monthly rent 1000-1500 SGD, preferably with gym and kitchen facilities, no more than 500 meters from MRT station...

Please include price range and target school 0/500

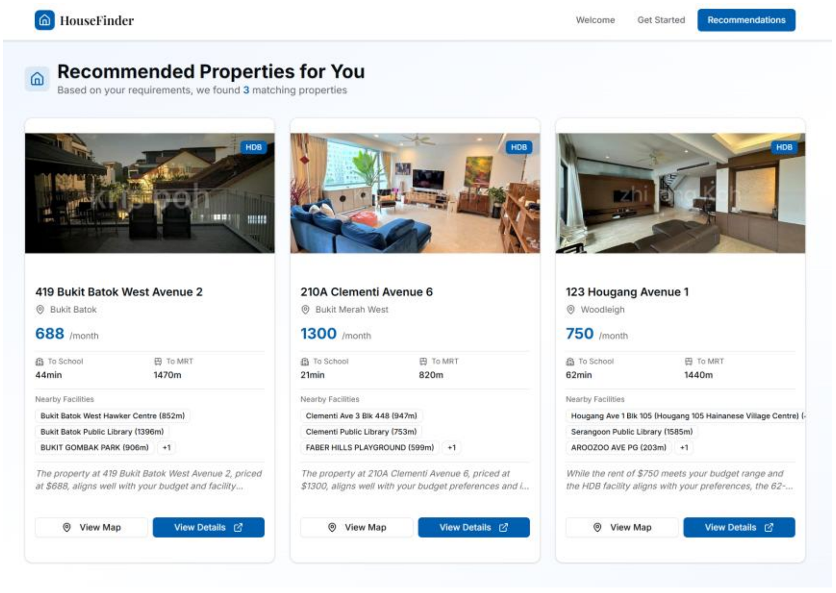
Tips: You can include the following information

- Budget range (required)
- Target school (required)
- Ideal location or district
- Property type preferences
- Transportation convenience requirements
- Other special requirements

Get Recommendations

1.1.3 Recommendations Page

The Recommendations page presents the results returned after users submit their preferences in a clean, responsive grid layout and prioritizes clarity, error handling, and quick actions: while loading it shows a centered spinner with a short “Finding your ideal home…” message, and if an error occurs the page displays a destructive alert with details and a prominent button to return to the home page; when results are available the header summarizes the total match count and the main content renders responsive RentBlock cards (one column on small screens, two on medium, three on large), each card offering a large thumbnail image with a facility-type badge, the property name and district, the monthly price clearly emphasized, and key metrics including commute time to the chosen school (minutes) and distance to the nearest MRT (meters); cards also list up to three nearby public facilities as small badges with distances and collapse extras behind a “+N” badge, and include a short recommendation reason; interactive affordances let users click the card or “View Details” to open the property page and use “View Map” to open an external map tab centered on the property’s coordinates, with smooth hover and transition effects to aid scanning; the page gracefully handles empty results by showing an inviting illustration and a helpful message prompting users to adjust filters; under the hood the UI first attempts to load cached recommendations from local storage (when coming from a form submit) and otherwise fetches default recommendations from the API, keeping the user informed of state and errors so they can quickly explore, map, or inspect each matched property.



Recommended Properties for You

Based on your requirements, we found 3 matching properties

419 Bukit Batok West Avenue 2

Bukit Batok

688 /month

To School 44min

To MRT 1470m

Nearby Facilities

Bukit Batok West Hawker Centre (852m)

Bukit Batok Public Library (1396m)

BUKIT GOMBAK PARK (906m) +1

The property at 419 Bukit Batok West Avenue 2, priced at $688, aligns well with your budget and facility...

210A Clementi Avenue 6

Bukit Merah West

1300 /month

To School 21min

To MRT 820m

Nearby Facilities

Clementi Ave 3 Blk 448 (947m)

Clementi Public Library (753m)

FABER HILLS PLAYGROUND (599m) +1

The property at 210A Clementi Avenue 6, priced at $1300, aligns well with your budget preferences and i...

123 Hougang Avenue 1

Woodleigh

750 /month

To School 62min

To MRT 1440m

Nearby Facilities

Hougang Ave 1 Blk 105 (Hougang 105 Hainanese Village Centre) (-

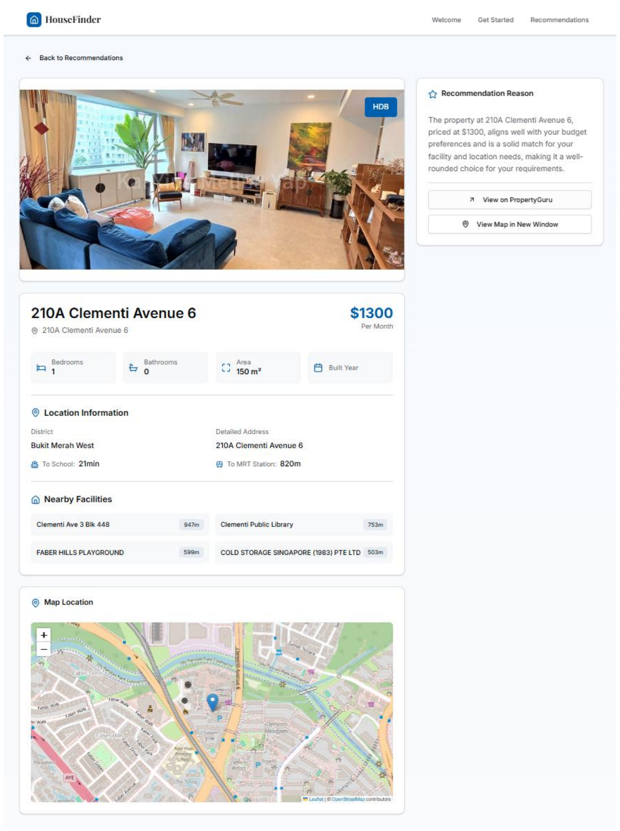
Serangoon Public Library (1585m)

AROOZOO AVE PG (203m) +1

While the rent of $750 meets your budget range and the HDB facility aligns with your preferences, the 62-...

1.1.4 Property Details Page

The property detail page gives a focused, action-oriented view for a single rental listing. The layout surfaces core facts in a compact metric grid (bedrooms, bathrooms, floor area, and built year) followed by a location section that shows district, detailed address, estimated commute time to the selected school and distance to the nearest MRT, and a short recommendation reason that explains why this property was suggested; nearby public facilities are listed with distances in an easy-to-scan card grid, and the lower content includes an embedded, sandboxed map iframe fetched from the server with a fallback map when the map service is unavailable so users always see a visual context for the listing; interactions are designed for discovery and safety — clicking images or buttons opens details or new windows, map scripts run isolated inside the iframe, and hover/transition effects improve scanability — together delivering a dependable, informative, and actionable single-listing experience that helps renters inspect, map, and act on a property without leaving the application.


HouseFinder

Welcome Get Started Recommendations

← Back to Recommendations

HDB

Recommendation Reason

The property at 210A Clementi Avenue 6, priced at $1300, aligns well with your budget preferences and is a solid match for your facility and location needs, making it a well-rounded choice for your requirements.

View on PropertyGuru

View Map in New Window

210A Clementi Avenue 6

210A Clementi Avenue 6

$1300
Per Month

Bedrooms	Bathrooms	Area	Built Year
1	0	150 m²	

Location Information

District	Detailed Address
Bukit Merah West	210A Clementi Avenue 6
To School: 21min	To MRT Station: 820m

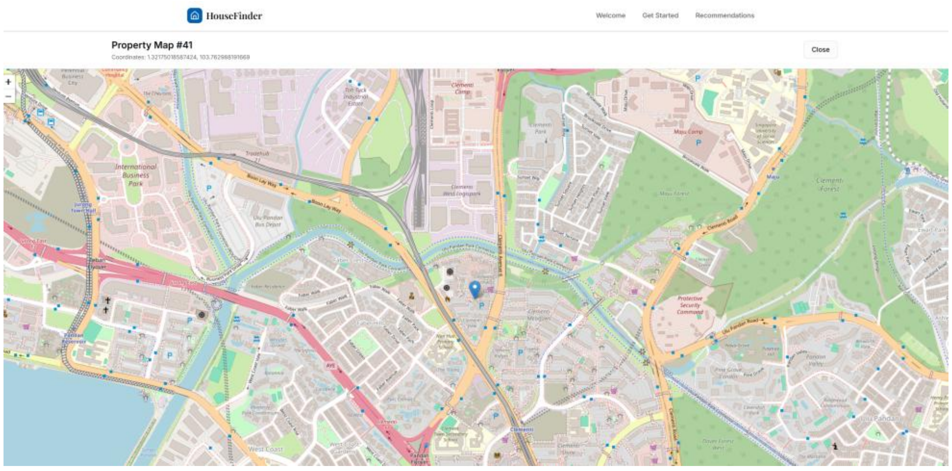
Nearby Facilities

Clementi Ave 3 Blk 448	947m	Clementi Public Library	753m
FABER HILLS PLAYGROUND	599m	COLD STORAGE SINGAPORE (1983) PTE LTD	503m

Map Location

1.1.5 Map View Page

The Map View page loads an embeddable map for a specific property by requesting server-generated HTML which is rendered inside a full-screen, sandboxed iframe; the header prominently shows the property identifier and the provided coordinates, warns when the fallback map is used, and provides a single "Close" action that closes the window, while the map itself is supplied by the server-side HTML and therefore carries the interactive behaviors implemented by that embed; users get a clear loading state, explicit error messaging, coordinate visibility, and a dependable map experience whether the server-provided map is available or the fallback is used.


HouseFinder

Welcome Get Started Recommendations

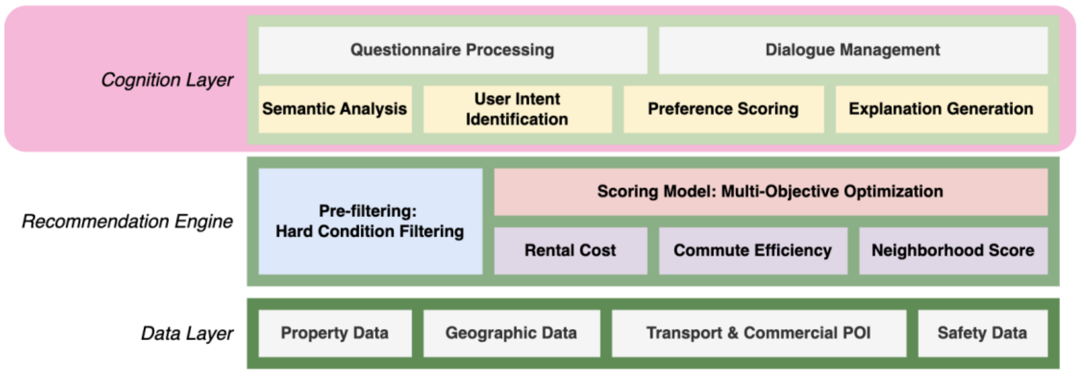
Property Map #41

Coordinates: 1.32175018587424, 103.762988191669

Close

1.2 Natural Language Module

The Natural Language Module is central to the Intelligent Renting Recommendation System's (IRRS) Cognition Layer, bridging user conversational inputs with the system's structured data processing.



Utilizing Large Language Models (LLMs) via the OpenAI API, it enables bidirectional intelligent interaction: understanding user requests and explaining system recommendations in natural language. This significantly enhances usability, intelligence, and user experience.

LLM application within this module focuses on two key stages: user requirement input ("downstream") and system recommendation output ("upstream").

1.2.1 Semantic Analysis & User Intent Identification

- Objective: Understanding and Structuring User Natural Language Requirements. To accurately convert unstructured user rental needs (e.g., "I need a room near NUS, budget around $1000.") into the structured query format required by the recommendation engine.
- Process: The language processing function receives user input, constructs the full prompt (instructions, knowledge, user text), requests structured output via the LLM API, parses the JSON response, and returns a dictionary for validation and instantiation into the internal query object.
- Implementation Techniques:
    - LLM: Utilizes OpenAI API (GPT-4 Turbo) for advanced natural language understanding.
    - Prompt Engineering: A detailed system prompt defines the LLM's role, task (preference extraction), and operational rules (ID mapping, inference, importance assessment).
    - Structured Output Generation (Tool Calling): Employs LLM's tool-calling feature with a predefined data schema (akin to a Pydantic model) specifying required fields (e.g., rent range, school ID, importance ratings). Forcing the LLM to use this schema ensures structured JSON output.
    - Context-based Knowledge Injection: Predefined entity mappings (school, district, etc.,

in JSON) are injected into the prompt as a "knowledge base," guiding the LLM to correctly link user mentions (e.g., "NUS") to system identifiers (e.g., `1`).

1.2.2 Explanation Generation

- Objective: Generation of Personalized Recommendation Rationales. To generate concise, personalized natural language explanations for each of the Top-K ranked properties, clarifying why it's a good match for the user.
- Implementation Techniques:
  - LLM: Utilizes OpenAI API (GPT-4 Turbo) for text generation.
  - RAG-style Prompt Engineering: Applies RAG principles without a vector database.
    - Retrieval: Gathers user's structured query (preferences, importance scores) and the specific property's details (price, commute, scores).
    - Augmentation: Dynamically inserts this retrieved information into a prompt template, augmenting the LLM's context.
    - Generation: The LLM, given the augmented prompt, performs a comparative analysis based on defined rules (prioritize importance, be specific, brief, diverse, honest about mismatches) to generate the explanation.
  - Asynchronous Parallel Processing: concurrently generate explanations for the Top-K list, reducing overall latency.

Through these applications of LLMs and Prompt Engineering, the Natural Language Module significantly enhances the IRRS system's intelligent interaction and the interpretability of its recommendations.

1.3 Backend Database

1.3.1 Housing-related Module

The system's intelligence is underpinned by a structured Knowledge Base (implemented as a PostgreSQL database), which stores and enriches all property, geographic, and facility data. Our implementation focuses on data collection, cleaning, enrichment, and efficient spatial modeling.

Data Acquisition and Preprocessing

- Property Listing Data: A corpus of approximately 1,000 rental listings was scraped from major Singaporean property platforms, including PropertyGuru and 99.co. This raw dataset underwent a rigorous cleaning process where listings with incomplete critical information (e.g., missing price or location) or containing obvious errors were filtered out to ensure data quality.
- Spatial and Contextual Data Enrichment:
    - Geocoding: The address of each cleaned property listing was processed using the OpenMap API to obtain precise geographic coordinates (latitude and longitude).
    - District Mapping & Safety Scoring: Singapore was logically segmented into 36 neighbourhoods based on the Neighbourhood Police Centre (NPC) boundaries. A safety score for each district was calculated based on the average total number of crimes reported over the past five years (2020-2024), providing a quantifiable safety metric for the ranking algorithm.
    - Facility Data: Public facility data—including parks, hawker centres, supermarkets, and libraries—were sourced from OpenMap and data.gov.sg APIs. The geographical coordinates of these facilities were stored to enable proximity-based scoring.
    - University Data: The geographical locations of six major Singaporean universities were collected and stored to serve as primary commute destinations.

Precomputation for Performance Optimization

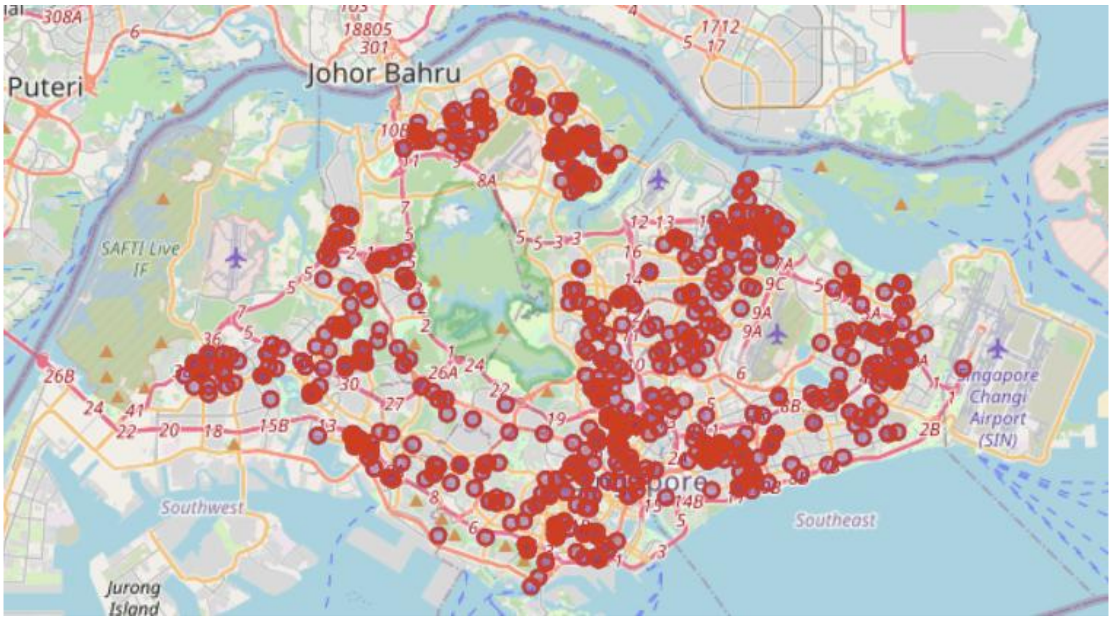
To ensure the system's responsiveness, several computationally intensive spatial calculations were precomputed and stored in the database:

- Commute Times: For each property, the estimated commute time to each of the six universities was precomputed using the OpenMap API's routing service and stored in a dedicated table.
- Facility Proximity: For each property, the distances to the nearest facilities (e.g., the closest park, hawker centre, library, supermarket with 3 ranks) were calculated and stored. This precomputation eliminates the need for real-time geometric calculations during the recommendation process.

Database Schema and Data Models

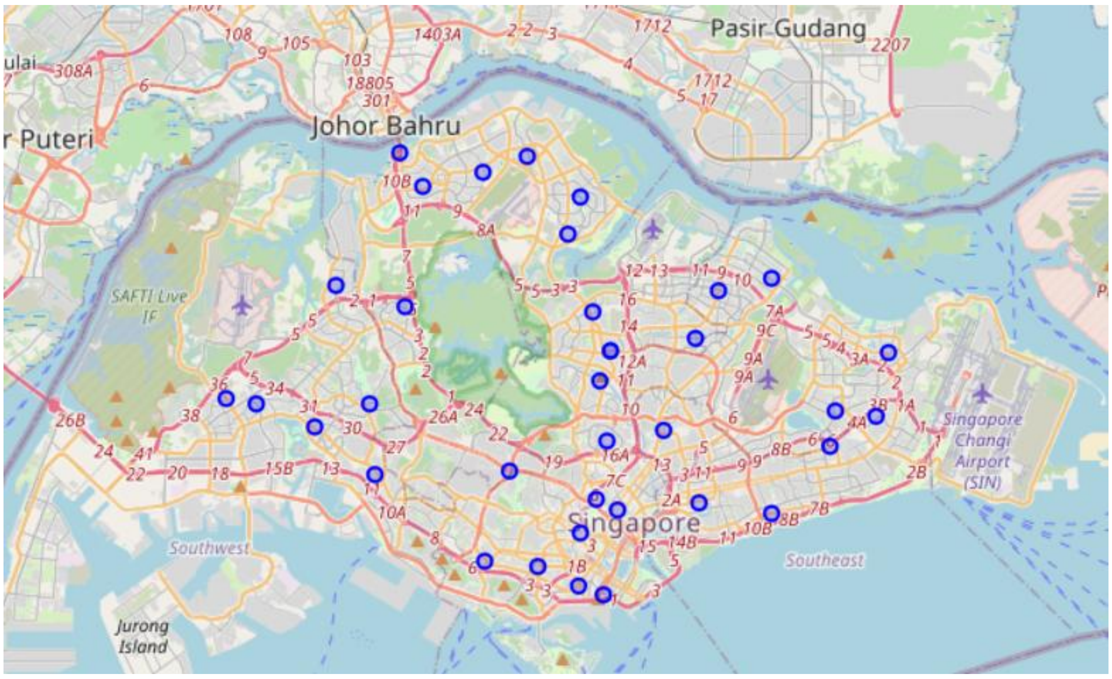
The database was implemented using a relational model with strong support for geospatial queries (PostGIS). The core entities and their relationships are defined as follows:

- HousingData: The central table containing all property listings, serving as the main data source for rental properties.



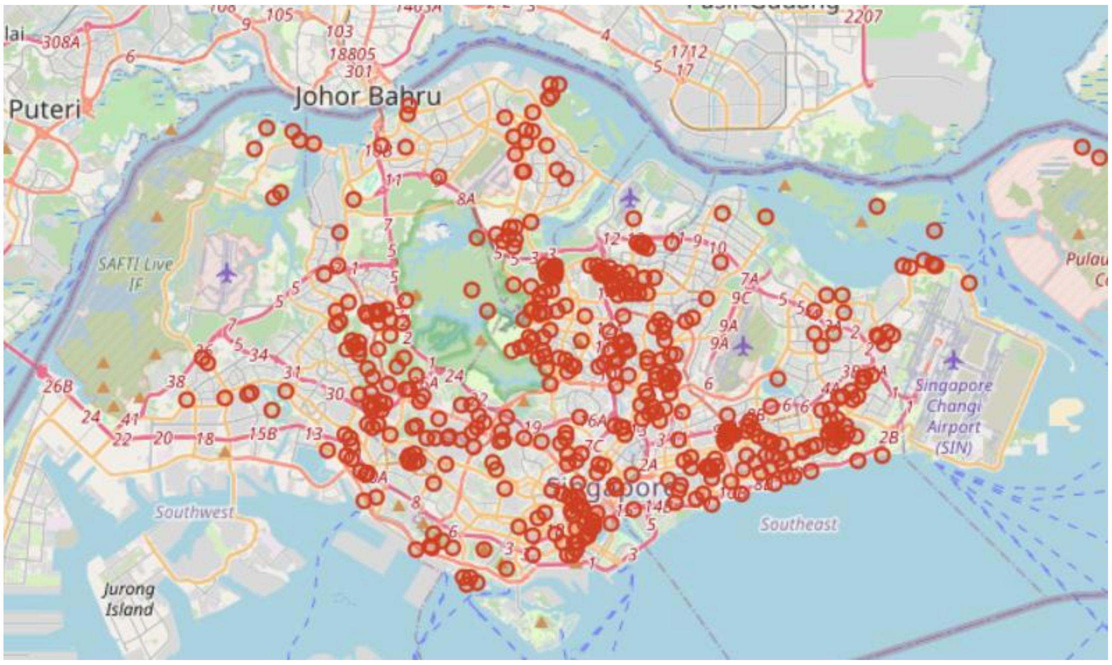
Collected housing data distribution

- District: Stores information for the 36 neighbourhoods, providing a safety dimension for the ranking algorithm and allows filtering by region.

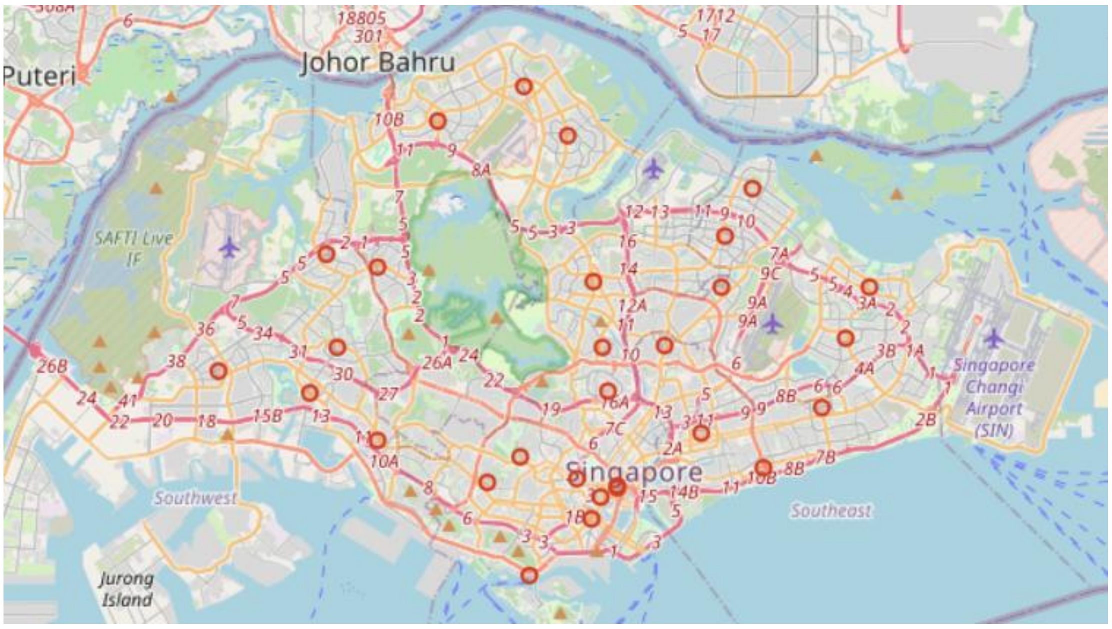


District distribution

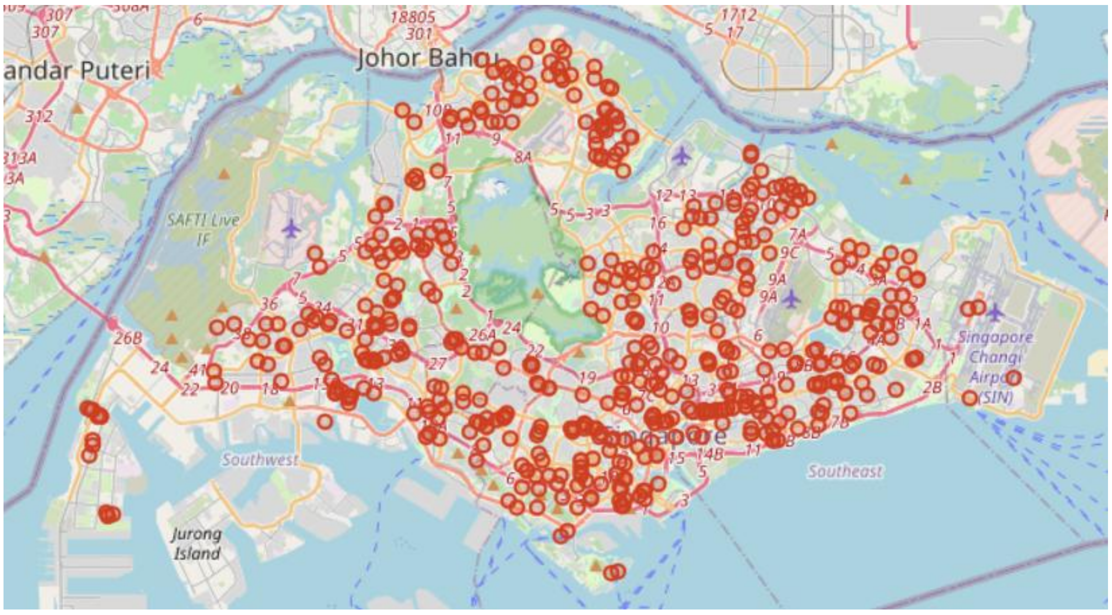
- University: Contains the locations of the six target universities defining the primary commute destinations for students.
- CommuteTime: A junction table that pre-stores commute times between properties and universities enabling efficient filtering and ranking based on the max_school_limit user input.
- Facility Tables (Park, HawkerCenter, Supermarket, Library): These tables store the locations of various public amenities, serving as reference data for calculating lifestyle convenience.



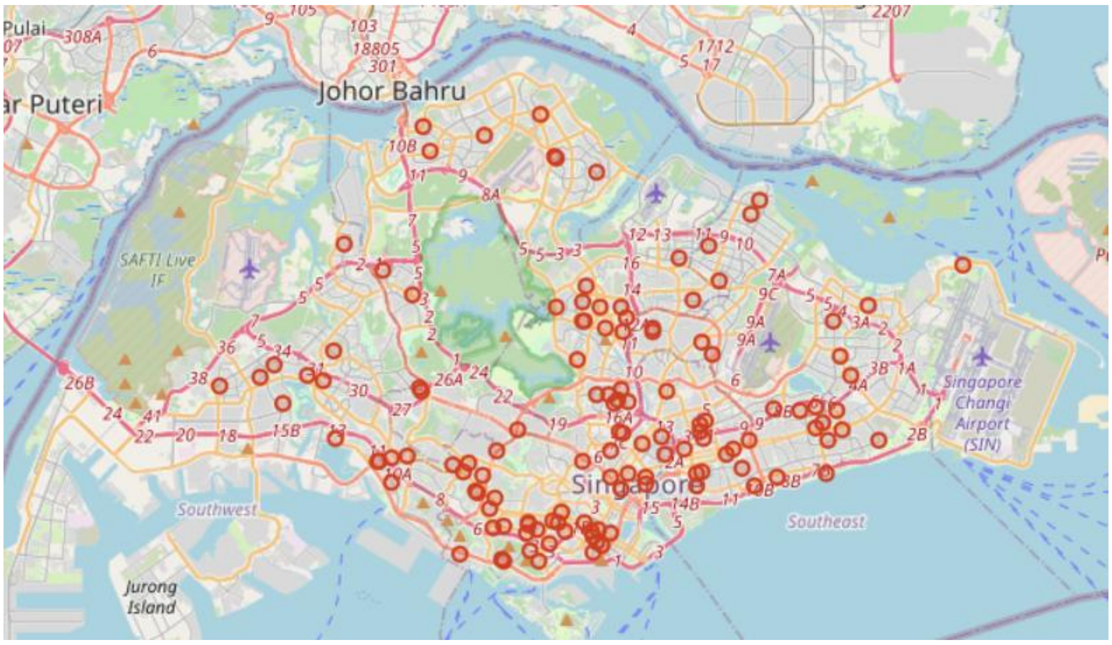
Park distribution



Library distribution

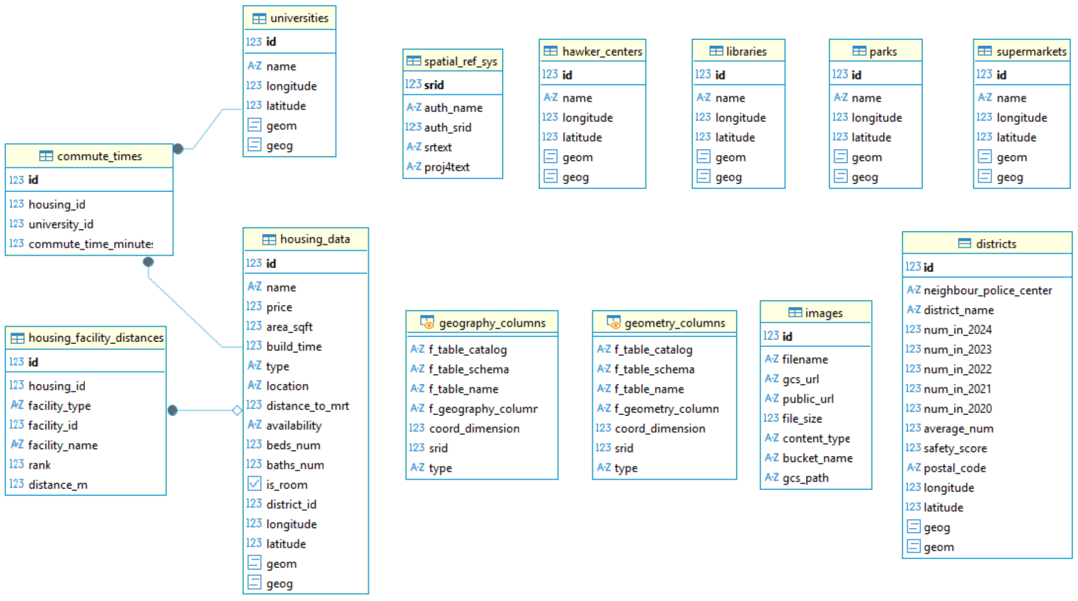


Supermarket distribution



Hawker center distribution

- HousingFacilityDistance: A critical table that materializes the proximity of each property to nearby facilities, allows the system to efficiently compute a "facility convenience score" for each property based on the number and proximity of surrounding amenities.



Overview of schemas

The final curated dataset, after the cleaning and enrichment pipeline, comprises the following volumes:

- Housing Data (HousingData): 730 records

- Facility Data:
  - Libraries: 30 records
  - Parks: 441 records
  - Supermarkets: 526 records
  - Hawker Centres: 129 records

This robust and precomputed data architecture ensures that the core recommendation engine can perform complex, multi-criteria queries with low latency, delivering a seamless user experience.

Cloud Deployment & Infrastructure

To ensure high availability, reliability, and ease of access for the application's backend services, the designed database and its associated spatial extensions have been deployed on the Google Cloud Platform (GCP).

The PostgreSQL database, with the PostGIS extension enabled for geospatial operations, is hosted on Google Cloud SQL.The deployment of the database on GCP provides a robust, scalable, and secure foundation for the recommendation system, aligning with production-level standards and facilitating future integration with other cloud-native services.

1.3.2 User Data Module

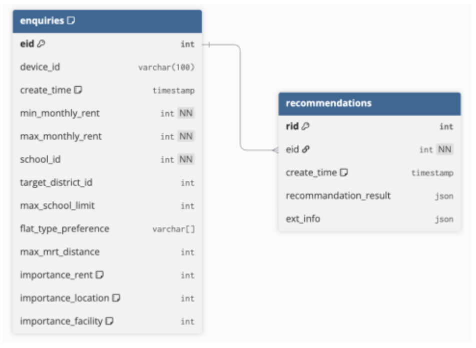
To enable future personalization and continuous optimization of the recommendation algorithm, the system implements a dedicated module for logging user interactions. This module captures the complete lifecycle of a user's query, from the initial requirements to the final recommendations presented.

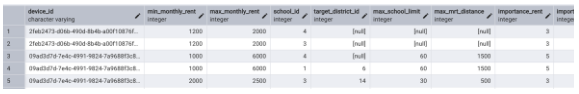
Schema Design

The module is built around two core tables: *enquiries* and *recommendations*.

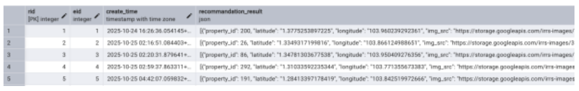
enquiries: This table serves as the root of each user session, storing the original input parameters submitted by the user.

recommendations: This table records every property that was recommended in response to an enquiry, establishing a clear many-to-one relationship with the enquiries table.





	device_id character varying	min_monthly_rent integer	max_monthly_rent integer	school_id integer	target_district_id integer	max_school_limit integer	max_mrt_distance integer	importance_rent integer	import intege
1	2feb2473-d06b-490d-8b4b-a00f10876f...	1200	2000	4	[null]	[null]	[null]	3	
2	2feb2473-d06b-490d-8b4b-a00f10876f...	1200	2000	3	[null]	[null]	[null]	3	
3	09ad3d7d-7e4c-4991-9824-7a9688f3c8...	1000	6000	4	[null]	60	1500	5	
4	09ad3d7d-7e4c-4991-9824-7a9688f3c8...	1000	6000	1	6	60	1500	5	
5	09ad3d7d-7e4c-4991-9824-7a9688f3c8...	2000	2500	3	14	30	500	3	



	rid [PK] integer	eid integer	create_time timestamp with time zone	recommandation_result json
1	1	1	2025-10-24 16:26:36.054145+...	[{"property_id": 200, "latitude": "1.3775253897225", "longitude": "103.960239292361", "img_src": "https://storage.googleapis.com/irrs-images/
2	2	2	2025-10-25 02:16:51.084403+...	[{"property_id": 26, "latitude": "1.3349317199816", "longitude": "103.866124988651", "img_src": "https://storage.googleapis.com/irrs-images/3
3	3	3	2025-10-25 02:20:31.879641+...	[{"property_id": 86, "latitude": "1.34781303677538", "longitude": "103.950409276356", "img_src": "https://storage.googleapis.com/irrs-images/
4	4	4	2025-10-25 02:59:37.863311+...	[{"property_id": 292, "latitude": "1.31033592235344", "longitude": "103.771355673383", "img_src": "https://storage.googleapis.com/irrs-images
5	5	5	2025-10-25 04:42:07.059832+...	[{"property_id": 191, "latitude": "1.28413397178419", "longitude": "103.842519972666", "img_src": "https://storage.googleapis.com/irrs-images

Purpose and Value for Future Improvement

This data collection strategy serves as a critical asset for the system's evolution:

- Algorithm Auditing and Tuning: By analyzing the relationship between user inputs (enquiries) and the resulting rankings (recommendations), we can validate the effectiveness of the scoring model. For instance, we can check if properties with higher ranking are truly more likely to be of interest to users with those specific preferences.

- Training Data for Personalization: This log forms a rich dataset for training machine learning models. Future models can move beyond static weights to predict personalized preference weights based on historical user behavior.

In summary, the User Data Module transforms the system from a one-time recommendation tool into a learning and adaptive platform, laying the groundwork for all data-driven improvements outlined in the Future Work section.

1.4 Basic Query and Filtering Module

This module forms the core of the recommendation engine's first stage. It is responsible for translating user preferences into a structured database query, performing an intelligent initial filtration, and generating a preliminary ranked list of properties based on a weighted scoring model.

1.4.1 Input Model and Request Handling

The module accepts a *RequestInfo* object from the frontend, which encapsulates the user's rental preferences. This object is defined by the following parameters (one or more may be present):

```
min_monthly_rent: int # Minimum monthly rent (S$)
max_monthly_rent: int # Maximum monthly rent (S$)
school_id: int # Target school ID (1-6)
target_district_id: int # Target district ID (1-36)
max_school_limit: int # Maximum acceptable commute time to school
(minutes)
flat_type_preference: string[] # Preferred housing types
max_mrt_distance: int # Maximum acceptable distance to the nearest
MRT station (meters)
importance_rent: int # Importance of rent (1-5)
importance_location: int # Importance of location (1-5)
importance_facility: int # Importance of facility convenience (1-
5)
```

1.4.2 Rule-based filtering

The module first constructs a database query using the provided constraints as hard filters. If the initial query returns an insufficient number of properties (e.g., less than 20), the system iteratively relaxes the least critical constraints until a viable candidate pool is established.This ensures that the system provides meaningful results even when user criteria are overly restrictive, enhancing the overall user experience.

1.4.3 Multi-Dimensional Scoring and Ranking

For each property in the filtered candidate list, the module calculates a normalized score across three distinct dimensions, which are then aggregated into a final weighted score.

1. Rent Score (*S_rent*):
   - Objective: Lower rent is better.
   - Calculation: The score is normalized inversely within the candidate pool. The property with the lowest rent receives a score of 1.0, and the highest rent receives a

score of 0. The formula shows as follows:

$$S_{rent} = \frac{max(rent) - rent_i}{max(rent) - min(rent)}$$

2. Commute Score(*S_commute*):

- Objective: Shorter commute time is better.
- Calculation: This score is derived from the precomputed *commute_time_minutes* from the *CommuteTime* table for the user's specified *school_id*. It is normalized similarly to the rent score, where the shortest commute gets a score of 1.0.

3. Neighbourhood Score(*S_neighbourhood*):

- Objective: A safer district with better facility access is better.
- Calculation: This is a composite score.
  - Facility Score: A normalized facility score 0-1 is calculated based on the summary number of facilities within a predefined radius (e.g., 2km). The data is fetched from the *HousingFacilityDistance* table.
  - Safety Score: The safety score is from the *District* table and normalized 0-1.
  - The final *S_neighbourhood* is a weighted average of the normalized facility score and the safety score, emphasizing the user's preference for convenience and safety.

1.4.4 Weighted Aggregation and Final Ranking

The final composite score for each property is calculated using the user-provided importance weights:

$$FinalScore_i = w_{rent} \times S_{rent_i} + w_{commute} \times S_{commute_i} + w_{facility} \times S_{neighbourhood_i}$$

The candidate properties are then sorted in descending order by their final score. The top 50 properties (or all properties if the list contains fewer than 50) are passed to the next stage of the system for further refinement and explanation generation. This ensures a manageable and high-quality set of recommendations is presented to the user.

1.5 Multi-objective Optimization Module

1.5.1 Purpose

The core challenge for the Intelligent Renting Recommendation System (IRRS) lies in handling users' diverse and potentially conflicting requirements (e.g., low rent vs. short commute time vs. comprehensive neighborhood amenities). To effectively address this, the project designed and implemented a Multi-objective Optimization (MOO) module. The goal of this module is not to find a single "best" solution, but rather to identify and rank a set of recommended properties that represent various excellent trade-offs.

1.5.2 Methodology

The ranking strategy employed by this module draws inspiration from the core concepts of the classic NSGA-II (Non-dominated Sorting Genetic Algorithm II) algorithm. It primarily involves data preprocessing, non-dominated sorting (Pareto front layering), crowding distance calculation, and a final ranking based on user preferences.

1.5.3 Implementation Steps

1. Data Validation and Filtering:

   Initially, the input property list is validated to ensure each entry contains the required score fields (costScore, commuteScore, neighborhoodScore) within the valid range (0, 1]. Incomplete or invalid properties are filtered out.

2. Score Normalization:

   To ensure comparability across objectives, scores are scaled to the [0, 1] interval using the following Min-Max Normalization formula:

$$x_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

3. Pareto Front Layering:

   Properties are layered based on Pareto Dominance, determined by the _dominates function:

   A dominates B if $A \succ B$

   This process iteratively identifies and removes layers of non-dominated solutions (Pareto fronts), starting with Layer 0 representing the best trade-offs.

4. Crowding Distance Calculation:

   To promote diversity within each Pareto layer, Crowding Distance is calculated. Boundary solutions receive infinite distance(∞). For others, the distance is the sum of normalized distances to adjacent neighbors across sorted objectives, favouring less crowded (more unique) solutions.

5. Final Ranking:

Properties are sorted primarily by Pareto Layer index (ascending), secondarily by Crowding Distance (descending), and finally tie-broken using a Weighted Score. This score directly uses the user's importance ratings (1-5) from EnquiryForm as weights, prioritizing properties that match user preferences in tie situations.

2. Workflow and Module Integration

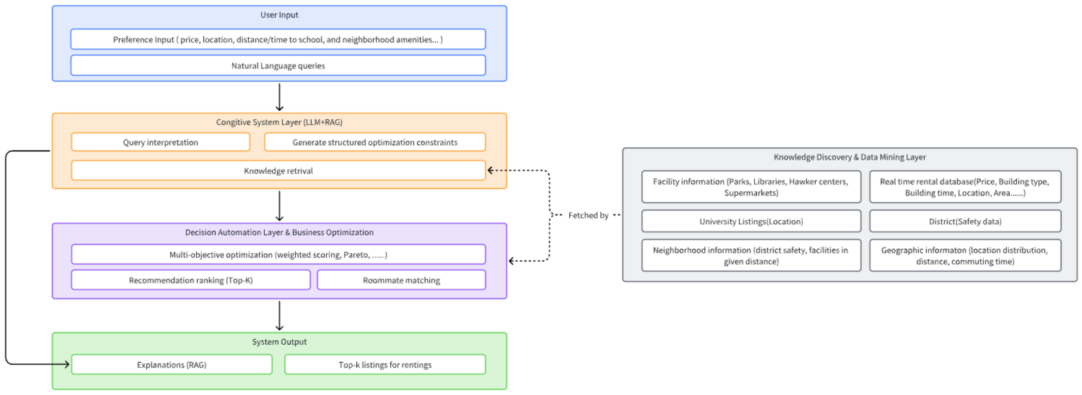
This section outlines the end-to-end interaction between the system's core components as follows:

1. Input Reception & Parsing: User requests are submitted via the interface. Structured input is parsed and passed to the Basic Query and Filtering Module. Natural language input is first processed by the Natural Language Module for intent recognition and entity extraction, converting it into structured data for the filtering module.

2. Candidate Retrieval & Ranking: The Basic Query and Filtering Module retrieves candidate properties from the Knowledge Base (Housing Database) based on the structured request, applying an initial scoring and ranking.

3. Result Optimization & Explanation: The preliminary ranked list is refined by the Multi-Objective Optimization Module for final sorting. Concurrently, this list is sent to the Natural Language Module, which generates concise natural language explanations for each property.

4. Result Presentation: The final list of recommendations, complete with explanations, is returned to the user interface for display.


User Input
Preference Input (price, location, distance/time to school, and neighborhood amenities...)
Natural Language queries
Congitive System Layer (LLM+RAG)
Query interpretation
Generate structured optimization constraints
Knowledge retrival
Decision Automation Layer & Business Optimization
Multi-objective optimization (weighted scoring, Pareto, ...)
Recommendation ranking (Top-K)
Roommate matching
System Output
Explanations (RAG)
Top-k listings for rentings
Fetched by
Knowledge Discovery & Data Mining Layer
Facility information (Parks, Libraries, Hawker centers, Supermarkets)
Real time rental database(Price, Building type, Building time, Location, Area......)
University Listings(Location)
District(Safety data)
Neighborhood information (district safety, facilities in given distance)
Geographic informaton (location distribution, distance, commuting time)

Overview of data flow

Future Improvement

While the current Intelligent Recommendation System provides a robust foundation, several key enhancements are planned to transition it from a functional prototype to a fully-fledged, adaptive, and production-ready service.

1. Advanced Personalization through Implicit Feedback

The current system relies solely on explicit user input. A primary direction for future work is to incorporate implicit feedback mechanisms to capture user preferences more subtly and accurately. This would involve:

- Tracking user interactions with the recommendation results, such as click-through rates, time spent viewing a listing, and save-to-favorites actions.
- Utilizing this behavioral data to train a machine learning model that can identify latent user preferences.
- Dynamically adjusting future recommendations to align with these inferred preferences, thereby creating a truly personalized experience that improves with use.

2. Automated Data Pipeline for Live Accuracy

To ensure the long-term relevance and accuracy of the recommendations, the static database will be evolved into a dynamic automated data pipeline. This improvement entails:

- Deploying the existing data acquisition and processing scripts as scheduled tasks (e.g., using cron jobs or Apache Airflow).
- Implementing incremental updates to the knowledge base, ensuring that new listings are added, outdated ones are removed, and spatial calculations are refreshed, all without manual intervention.

These planned improvements will significantly enhance the system's intelligence, responsiveness, and practical utility.

References

Statistics of international students in Singapore:

- https://www.population.gov.sg/files/media-centre/publications/pib-2020-final.pdf
- https://www.sgtimes.com/2024/09/20/singapores-2023-population-statistics-international-students-account-for-4-of-non-resident-population/
- https://www.businesstimes.com.sg/property/foreign-students-fuelling-demand-singapores-maturing-co-living-market-jll-report

Facility Data source:

- https://data.gov.sg/
- https://www.onemap.gov.sg/apidocs/themes/#getThemesInfo

Housing Data source:

- https://www.propertyguru.com.sg/
- https://www.99.co/

Appendix A: Project Proposal

PROJECT PROPOSAL

Date of proposal: 13 Sep 2025
Project Title: Intelligent Renting Recommendation System
Group ID (As Enrolled in Canvas Class Groups): Group 14 **Group Members (name , Student ID):** Chen Sigen A0326351L Qi Yilin A0328843W Ru Yanjie A0295237W Yao Yiyang A0294873L
Sponsor/Client: *(Company Name, Address and Contact Name, Email, if any)*

Background/Aims/Objectives:

1. Project Overview

International students in Singapore often face complex trade-offs when choosing rental accommodations, balancing price, location, safety, and convenience. This project aims to develop an **intelligent rental recommendation system** that leverages **multi-objective optimization** and knowledge-driven data to provide personalized, explainable recommendations.

2. Core Objectives

2.1 Multi-criteria rental recommendation

- Users can search based on multiple indicators: price, location, distance/time to school, crime rate, and neighborhood amenities such as food court and market, etc.
- Users can evaluate the price based on historical average price/system predicted price in the area.
- The system builds a model to evaluate the living environment using rental data and a knowledge base of regional information (e.g., rental trends, crime rates, nearby facilities), in order to generates **Top-K recommended options.**

2.2 Roommate matching

- Based on budget, lifestyle preferences, and habits, the system can suggest potential roommates with high compatibility scores.

2.3 Intelligent, interactive interface

- LLM-driven natural language interface: users can describe their needs conversationally.
- System interprets queries, applies multi-objective optimization, and provides explainable recommendations.

Project Descriptions:

1. Project Description

1.1 Product Form

- Web-based or mobile prototype where students can input preferences (budget, location, amenities, ...) and receive Top-K rental recommendations.
- Interactive interface supports natural language queries via LLM, displays recommended listings with key details, and optionally shows compatible roommate suggestions.
- Users can explore neighborhood insights (rent trends, safety, facilities) and adjust preferences to refine recommendations.

1.2 Business Use Case Design

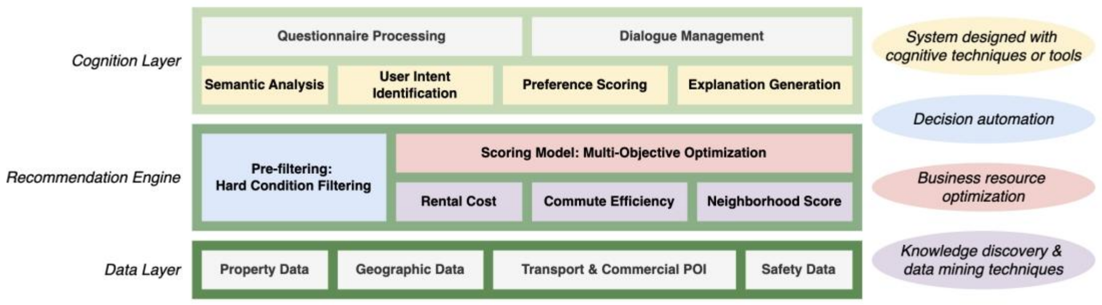
No.	Use Case	Functionality	User Input	System Output
1	**Rental Recommendation**	The system uses multi-objective optimization and a knowledge base (historical rental data, safety scores, commute time, amenities, etc.) to recommend top-K rental options that best match the user's preferences. The system supports natural language input and provides explainable recommendations.	1. Users fill out the questionnaire, which includes budget, expected commuting time, requirements for the surrounding environment, etc. 2. User input a complete sentence, for example:"I need a room under $1000, within 30 minutes to NTU, and near MRT.""Show me safe areas with supermarkets nearby."	A ranked list of rental recommendations with detailed attributes (price location, commute time, safety score, amenities).Each recommendation include a brief explanation (e.g., "Recommended because under budget, 25 min to NTU, 300m to MRT").
2	**Roommate Matching**	The system suggests compatible roommates based on recommended house, budget, location preference, lifestyle habits (e.g., sleep schedule), and gender. It uses similarity scoring and historical user profiles to recommend potential matches.	1. Simply click the "Recommended Roommates" button. Choose whether to fill out the roommate requirements questionnaire, which includes information such as gender and daily schedule. 2. User input a complete sentence, for example: "Find me	A list of potential roommates with compatibility scores and contact info (e.g., email).Each suggestion includes a reason (e.g., "90% match: same budget, sleep after 1am, female").

			a roommate near NUS, budget $800–$1000, female, night owl.”“I’m looking for a quiet roommate in Clementi.”	

2. System Design

2.1 System Architecture

Overall Architecture Diagram:



Cognition Layer
Questionnaire Processing
Dialogue Management
Semantic Analysis
User Intent Identification
Preference Scoring
Explanation Generation
Recommendation Engine
Pre-filtering: Hard Condition Filtering
Scoring Model: Multi-Objective Optimization
Rental Cost
Commute Efficiency
Neighborhood Score
Data Layer
Property Data
Geographic Data
Transport & Commercial POI
Safety Data
System designed with cognitive techniques or tools
Decision automation
Business resource optimization
Knowledge discovery & data mining techniques

2.1.1 Cognitive Layer: Natural language input/output

- Natural language interface powered by LLMs.
- Explainable reasoning based on knowledge using RAG.
- Transform user queries into structured optimization constraints.
- Provide explainable recommendations and support iterative user feedback.

2.1.2 Recommendation Engine: Decision Automation

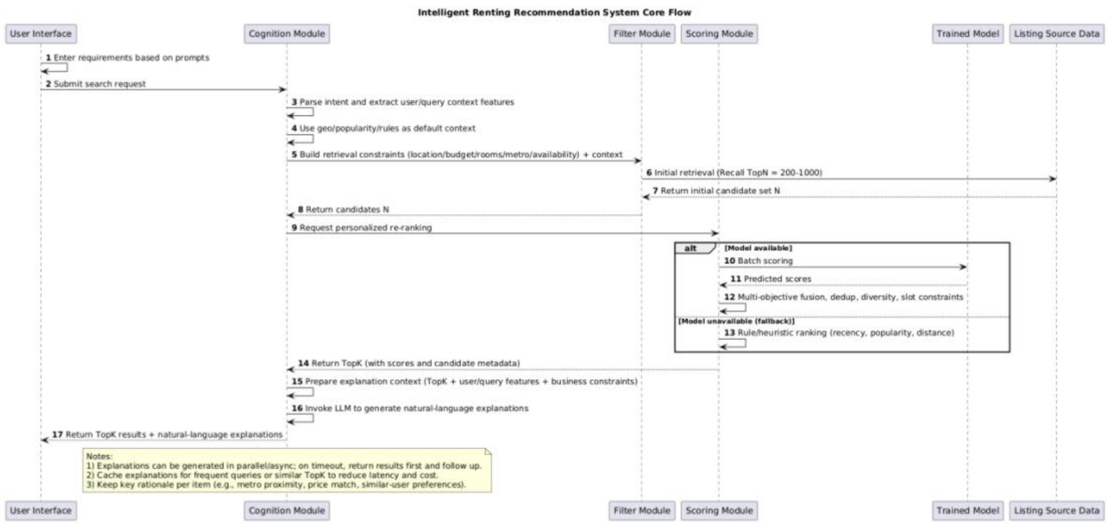
- Multi-objective optimization: combine user-defined preferences (e.g., price vs. distance vs. safety) to rank options.
- Recommendation algorithms: weighted scoring, Pareto optimization, or learning-to-rank models.
- Roommate matching using similarity measures and preference alignment.

2.1.3 Data Layer: Knowledge Discovery & Data Mining

- Collect and clean historical rental data and neighborhood knowledge.
- Feature extraction: rental price trends, safety scores, amenity proximity, commuting times.
- Build structured user profiles for personalized recommendations.

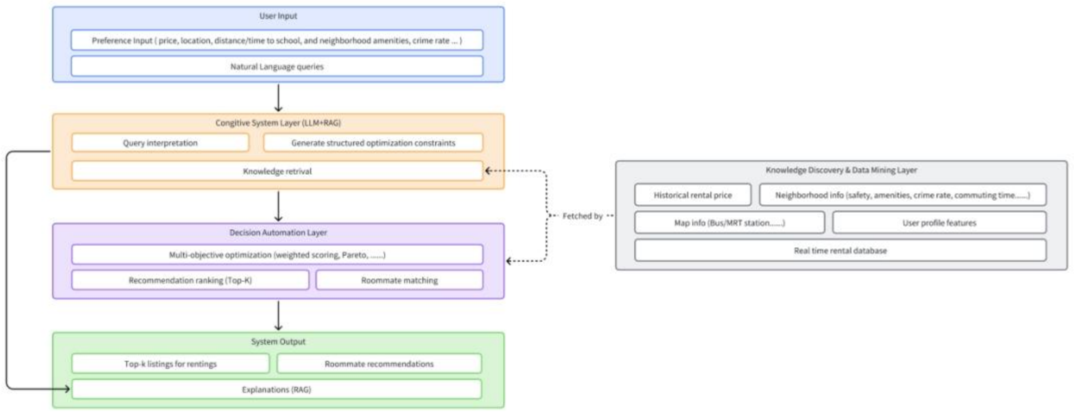
2.2 Component Interaction Sequence

Sequence Diagram:


Intelligent Renting Recommendation System Core Flow
User Interface
Cognition Module
Filter Module
Scoring Module
Trained Model
Listing Source Data
1 Enter requirements based on prompts
2 Submit search request
3 Parse intent and extract user/query context features
4 Use geo/popularity/rules as default context
5 Build retrieval constraints (location/budget/rooms/metro/availability) + context
6 Initial retrieval (Recall TopN = 200-1000)
7 Return initial candidate set N
8 Return candidates N
9 Request personalized re-ranking
alt
[Model available]
10 Batch scoring
11 Predicted scores
12 Multi-objective fusion, dedup, diversity, slot constraints
[Model unavailable (fallback)]
13 Rule/heuristic ranking (recency, popularity, distance)
14 Return TopK (with scores and candidate metadata)
15 Prepare explanation context (TopK + user/query features + business constraints)
16 Invoke LLM to generate natural-language explanations
17 Return TopK results + natural-language explanations
Notes:
1) Explanations can be generated in parallel/async; on timeout, return results first and follow up.
2) Cache explanations for frequent queries or similar TopK to reduce latency and cost.
3) Keep key rationale per item (e.g., metro proximity, price match, similar-user preferences).

2.3 Data description

2.3.1 Data Flow


User Input
Preference Input (price, location, distance/time to school, and neighborhood amenities, crime rate ...)
Natural Language queries
Congitive System Layer (LLM+RAG)
Query interpretation
Generate structured optimization constraints
Knowledge retrival
Decision Automation Layer
Multi-objective optimization (weighted scoring, Pareto, ...)
Recommendation ranking (Top-K)
Roommate matching
System Output
Top-k listings for rentings
Roommate recommendations
Explanations (RAG)
Fetched by
Knowledge Discovery & Data Mining Layer
Historical rental price
Neighborhood info (safety, amenities, crime rate, commuting time......)
Map info (Bus/MRT station......)
User profile features
Real time rental database

2.3.2 Data Sources

- Historical renting data (200~500 entries) fetched with the help of GPT from PropertyGuru (https://www.propertyguru.com.sg/). We will use it to build a mock database, so that we can demonstrate the function of real time recommendation.
  - Data Samples

#	Name	Price	Beds	Baths	Area	Build Time	Type	Location	To Mrt St	Welcome people
1	Hillview Garden	S$40,000/mo	7 Beds	10 Baths	15,000 sqft	—	Detached House	Hillview Garden	5 min (450 m) from DT3 Hillview MRT	Everyone welcome (propertygur

	Estate							Estate	Station	u.com.sg)
2	Villa Delle Rose	S$10,500/mo	3 Beds	4 Baths	2,800 sqft	1982	Condominium	Taman Nakhoda / Farrer Road area	14 min (1.21 km) from CC20 Farrer Road MRT Station	Everyone welcome (propertyguru.com.sg)
3	Newly Built Modern Family Home Near Namly Avenue	S$48,000/mo	5 Beds	7 Baths	9,000 sqft	—	Detached House	Near Namly Avenue	18 min (1.51 km) from DT7 Sixth Avenue MRT Station	— (propertyguru.com.sg)

- Neighborhood and regional information from government open data portals (https://data.gov.sg/datasets?agencies=HDB&page=1&query=Renting+condo&resultId=d_c9f57187485a850908655db0e8cfe651).
    - Data Samples:

Rent Approval DateMonth (YYYY-MM)	**TownText**	**BlockText**	**Street NameText**	**Flat TypeText**	**Monthly RentNumeric**
(Null)0.0%2021-032.3%2025-072.2%2021-052.1%2025-032.1%51 more values	(Null)0.0%2021-032.3%2025-072.2%2021-052.1%2025-032.1%51 more values	(Null)0.0%10.5%20.6%40.5%80.4%2775 more values	(Null)0.0%ANG MO KIO AVE 31.3%ANG MO KIO AVE 101.3%YISHUN RING RD1.2%BEDOK RESERVOIR RD1.0%595 more values	(Null)0.0%4-ROOM36.1%3-ROOM32.7%5-ROOM23.6%EXECUTIVE5.5%2 more values	(Null)0.0%20005.7%25005.1%28004.8%30007.2%648 more values

- Real-time geographic information and commuting data will come from the **Google Maps platform**.
    - Distance Matrix API: It can calculate the travel time and distance between different residential options and key destinations such as schools, food courts and shopping centers.
    - Location API: It can identify various nearby facilities (such as food courts, shopping centers) and other relevant information such as their ratings and business hours.
    - Geocoding API: It converts addresses into geographic coordinates for spatial analysis and visualization operations.

These APIs serve as the basic data for building spatial features related to user preferences, such as "at least 3 food courts within a 10-minute walk" or "commuting time no more than 30 minutes". Based on the data obtained from the Google Maps API, we will build a **customized model** to evaluate the geographical convenience of each housing candidate location: Using the user-defined commuting threshold and the output results of the distance matrix API to calculate the commuting cost score. Utilizing location API data (such as the number and ratings of nearby food courts and shopping centers) to generate a score for lifestyle convenience.

2.3.3 Assumptions

The system is **assumed** to have access to real-time rental APIs for demonstration purposes, so that user can receive recommendations based on up-to-date rental listings.

2.3.4 Feasibility & System Objective

- The main focus is to demonstrate the methodology—**multi-objective optimization, knowledge-base integration, and recommendation algorithms**—rather than require live API integration.
- Historical rental data will be used as the **experimental dataset to validate the effectiveness of the algorithms** in generating Top-K recommendations.
- For prototype demonstration, the system can assume real-time API access to **showcase interactive recommendations**.
- Once connected to live APIs, **the same optimization and recommendation logic** can be applied to operate in real market conditions.

2.4 Technology Stack

Category	Technology / Tool	Purpose / Note
Backend	Python, FastAPI	High performance, excellent asynchronous support
NLP	spaCy, Hugging Face Transformers	For NER, relationship extraction, summarization, sentiment analysis
AI Framework	PyTorch, PyTorch Geometric, Stable-Baselines3	For GNN and RL algorithm implementation
LLM	OpenAI API, Llama.cpp	For the generative part of RAG (locally deployed)
Database	Neo4j	Graph database

Task Queue & Cache	Celery, Redis	Distributed task queue; In-memory data store / cache
Frontend	Vue.js, ECharts	Progressive framework with rich ecosystem; Powerful visualization library
Deployment	Docker	Containerization for environment consistency and deployment

Appendix B: Mapped System Functionalities to Courses

Decision automation	• **Rule-Based Pre-Filtering (Business Rules & Process):** The recommendation engine initially applies Hard Condition Filtering using defined business rules. It automatically excludes properties failing to meet the user's explicit constraints (e.g., budget, commute time, room type) specified via questionnaire or natural language, streamlining the initial selection. • **Conversion from Natural Language to Structured Constraints (Business Rules & Process):** The Cognition Layer employs LLM Tool Calling to process natural language input. Prompt Engineering defines rules that enable the LLM to convert ambiguous user descriptions (e.g., "near MRT") into precise, structured query constraints. • **Recommendation Rationale Generation (Knowledge-Based Reasoning):** Adopting a RAG-like approach, the system generates explanations by providing the LLM with retrieved context (user query features and property data). The LLM reasons over this information to produce personalized rationales justifying the recommendation based on the user's needs.
Business resource optimization	• **Multi-Objective Optimization (Evolutionary Computing):** The recommendation engine employs a Multi-Objective Optimization (MOO) model to balance conflicting user objectives (cost, commute, neighborhood score). Using NSGA-II inspired methods like non-dominated sorting and crowding distance, it identifies Pareto optimal properties. This offers users diverse recommendations representing optimal trade-offs, applying evolutionary computing to find best-matching properties.

Knowledge discovery & data mining techniques	• **Recommendation System Construction (Recommendation):** The project directly applies recommendation technology by building an intelligent system that generates a Top-K list of rental properties based on user preferences. • **Data Collection and Feature Extraction (Data Mining):** Integrating diverse data sources (property, geographic, POI, safety), the system cleans and processes raw data. Key features like commute time, safety scores, and facility convenience are extracted through feature engineering, forming the basis for scoring and recommendation models via data mining. • **Scoring Model (Data Mining):** Established through analysis of historical data and regional knowledge, the recommendation engine's scoring model quantitatively evaluates properties against user needs across dimensions like rent, commute, and neighborhood environment.
System designed with cognitive techniques or tools	• **Natural Language Interface:** Powered by an LLM, the system allows users to describe needs conversationally. The cognitive layer performs Semantic Analysis and User Intent Identification, converting unstructured input into structured data (slot filling) for the system. • **Knowledge Base Application:** While lacking an explicit knowledge graph, the system uses implicit knowledge components: ◦ **Structured Mapping:** Predefined JSON mappings (school ID, district, etc.) serve as contextual knowledge, guiding the LLM in accurate entity linking during natural language understanding. ◦ **RAG (Retrieval-Augmented Generation):** For explanation generation, retrieved user queries and property details augment the LLM's input context, enabling personalized, fact-based rationales, reflecting the core RAG concept. • **Explanation Generation:** The system provides natural language explanations for recommendations, illustrating the reasoning. This cognitive system feature aims to enhance transparency and user trust.

Appendix C: Installation and User Guide

1. Installation Guide

1.1 Resource Acquisition

- Clone this repository: https://github.com/csgen/IRS-PM-2025-10-01-ISY5001FT-GRP14-IntelligentRentingRecommendationSystem

```
Plain Text
git clone https://github.com/csgen/IRS-PM-2025-10-01-ISY5001FT-
GRP14-IntelligentRentingRecommendationSystem.git
```

- Open `config_secret.zip`, which contains two files:
  - `irrs-sql-key.json`
  - `secret.txt`

1.2 Establishing a Secure Cloud Connection

- Download and authorize the Cloud SQL Auth Proxy:

  Refer to the official documentation for instructions: https://docs.cloud.google.com/sql/docs/mysql/sql-proxy?hl=zh-cn#install

- Launch the Cloud SQL Auth Proxy:

  Execute the following command in the terminal. Ensure it is run in the directory where the Cloud SQL Auth Proxy has been installed and authorized.

  Note: Replace `/xxx/xxx/irrs-sql-key.json` with the absolute path to the `irrs-sql-key.json` file.

```
Plain Text
// Windows
./cloud-sql-proxy.exe --credentials-file "/xxx/xxx/irrs-sql-
key.json" --port 5433 compact-harbor-475309-h1:asia-
southeast1:irrs-db


// macOS / Linux
./cloud-sql-proxy --credentials-file "/xxx/xxx/irrs-sql-key.json"
--port 5433 compact-harbor-475309-h1:asia-southeast1:irrs-db
```

1.3 Starting the Backend

- Open a new terminal window and navigate to the `SystemCode/backend` directory:

```
Plain Text
cd SystemCode/backend
```

- Configure environment variables
    - Create the `.env` file:

    ```
    Plain Text
    // Windows
    Copy-Item .env.example .env

    // macOS / Linux
    cp .env.example .env
    ```

    - Edit the `.env` file:

      Open `secret.txt` and replace the corresponding content in `.env` with reference to its contents.

- Launch:

  Ensure you are in the `backend` directory and that Docker Desktop is running, then execute the following command:

```
Plain Text
docker-compose up --build -d
```

1.4 Starting the Frontend

- Open a new terminal window and navigate to the `frontend` directory:

```
Plain Text
cd SystemCode/frontend
```

- Install dependencies and launch:

```
Plain Text
pnpm install && pnpm build && pnpm dev
```

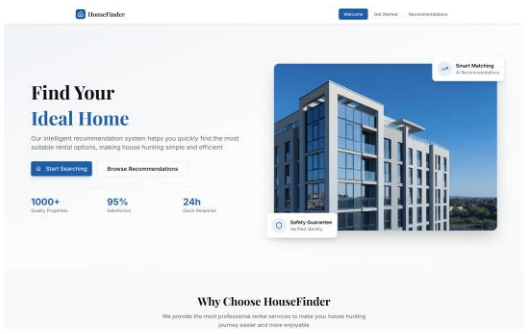
1.5 Getting Started

- Ensure the cloud proxy, backend, and frontend are all running (as per steps 2, 3, and

4). Open a browser and visit `http://localhost:3000` or `http://127.0.0.1:3000` to start using the system!

2. User Guide

2.1 System Homepage



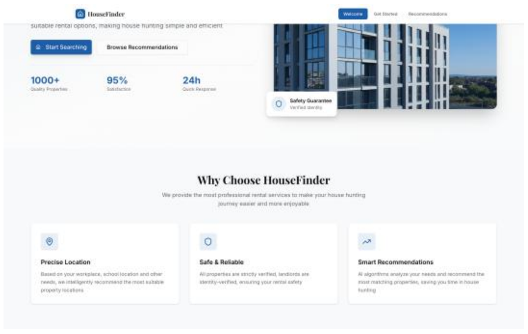
Find Your
Ideal Home

Our intelligent recommendation system helps you quickly find the most suitable rental options, making house hunting simple and efficient

1000+ Quality Properties
95% Satisfaction
24h Quick Response

Why Choose HouseFinder

We provide the most professional rental services to make your house hunting journey easier and more enjoyable



suitable rental options, making house hunting simple and efficient

1000+ Quality Properties
95% Satisfaction
24h Quick Response

Why Choose HouseFinder

We provide the most professional rental services to make your house hunting journey easier and more enjoyable

Precise Location
Based on your workplace, school location and other needs, we intelligently recommend the most suitable property locations

Safe & Reliable
All properties are strictly verified, landlords are identity-verified, ensuring your rental safety

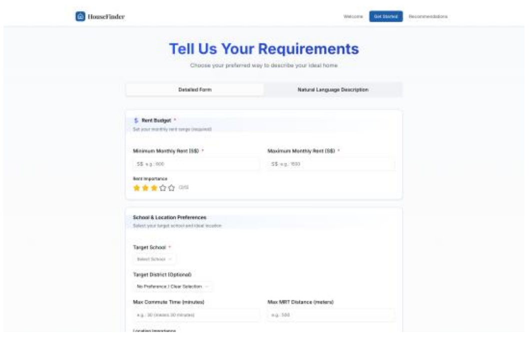
Smart Recommendations
AI algorithms analyze your needs and recommend the most matching properties, saving you time in house hunting

Welcome to the HouseFinder Intelligent Rental Recommendation System. We are committed to making the rental process more efficient and hassle-free for international students in Singapore. Please let us know your rental requirements, and HouseFinder wishes you to quickly find your ideal accommodation!

2.2 Preference Collection

You can share your rental requirements with us in two ways: filling out a questionnaire or describing them in natural language.

2.2.1 Questionnaire



Tell Us Your Requirements

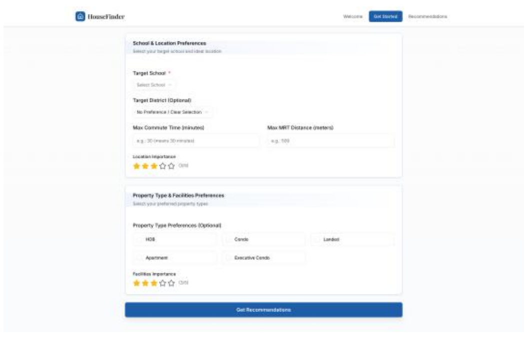
Choose your preferred way to describe your ideal home

Rent Budget *
Set your monthly rent range (required)

Minimum Monthly Rent (S$) *: S$ e.g.: 800
Maximum Monthly Rent (S$) *: S$ e.g.: 1500
Rent Importance: ★★★☆☆ (3/5)

School & Location Preferences
Select your target school and ideal location

Target School *: Select School
Target District (Optional): No Preference / Clear Selection
Max Commute Time (minutes): e.g.: 30 (means 30 minutes)
Max MRT Distance (meters): e.g.: 500
Location Importance



School & Location Preferences
Select your target school and ideal location

Target School *: Select School
Target District (Optional): No Preference / Clear Selection
Max Commute Time (minutes): e.g.: 30 (means 30 minutes)
Max MRT Distance (meters): e.g.: 500
Location Importance: ★★★☆☆ (3/5)

Property Type & Facilities Preferences
Select your preferred property types

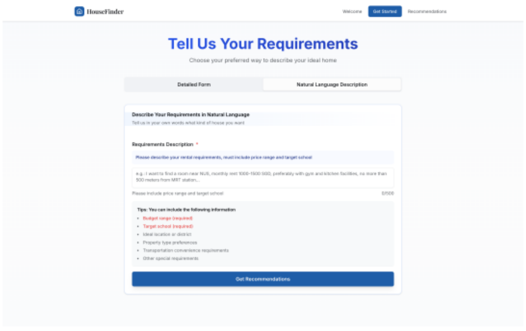
Property Type Preferences (Optional): HDB, Condo, Landed, Apartment, Executive Condo
Facilities Importance: ★★★☆☆ (3/5)

- Our questionnaire gathers your requirements from three dimensions: price range, location & commute, and amenities. You may fill in the information selectively.
- Please note that some information is mandatory, such as your expected price range

and your school.

- Additionally, you can indicate the importance of each of these three dimensions to you (rated on a 1–5 star scale). This enables us to recommend suitable rental properties more accurately and efficiently.

2.2.2 Natural Language Description


HouseFinder — Welcome | Get Started | Recommendations

Tell Us Your Requirements

Choose your preferred way to describe your ideal home

Detailed Form | Natural Language Description

Describe Your Requirements in Natural Language

Tell us in your own words what kind of house you want

Requirements Description *

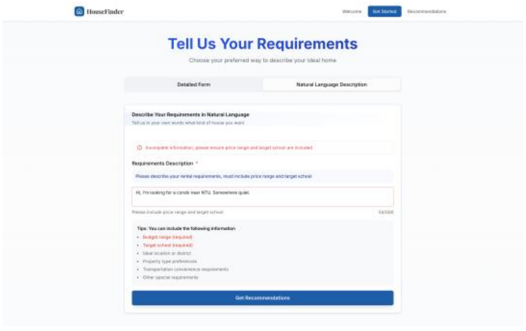
Please describe your rental requirements, must include price range and target school

e.g.: I want to find a room near NUS, monthly rent 1000-1500 SGD, preferably with gym and kitchen facilities, no more than 500 meters from MRT station...

Please include price range and target school 0/500

Tips: You can include the following information
- Budget range (required)
- Target school (required)
- Ideal location or district
- Property type preferences
- Transportation convenience requirements
- Other special requirements

Get Recommendations


HouseFinder — Welcome | Get Started | Recommendations

Tell Us Your Requirements

Choose your preferred way to describe your ideal home

Detailed Form | Natural Language Description

Describe Your Requirements in Natural Language

Tell us in your own words what kind of house you want

Incomplete information, please ensure price range and target school are included

Requirements Description *

Please describe your rental requirements, must include price range and target school

Hi, I'm looking for a condo near NTU. Somewhere quiet.

Please include price range and target school 54/500

Tips: You can include the following information
- Budget range (required)
- Target school (required)
- Ideal location or district
- Property type preferences
- Transportation convenience requirements
- Other special requirements

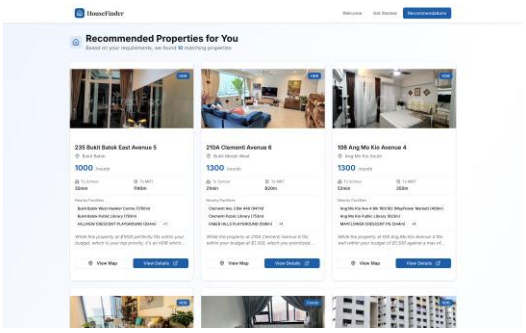
Get Recommendations

- You may also enter a text paragraph to share your requirements with us.

- Please note that, consistent with the questionnaire requirements, your description must include at least your expected price range and your school information.

- If not, the system will prompt you to supplement the details—this ensures we can leverage the system's language understanding capability to recommend rental properties for you accurately and efficiently.

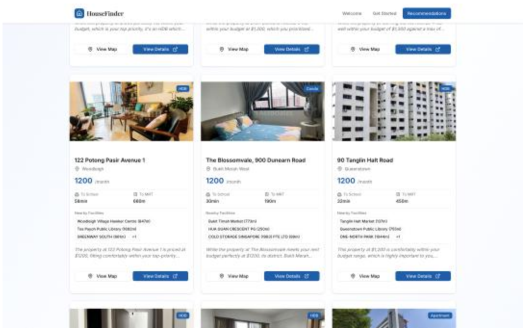
2.3 Recommended Rental Listings


HouseFinder — Welcome | Get Started | Recommendations

Recommended Properties for You

Based on your requirements, we found 10 matching properties

HDB	HDB	HDB
235 Bukit Batok East Avenue 5	210A Clementi Avenue 6	108 Ang Mo Kio Avenue 4
Bukit Batok	Bukit Merah West	Ang Mo Kio South
1000 /month	1300 /month	1300 /month
To School 35min · To MRT 1548m	To School 21min · To MRT 820m	To School 53min · To MRT 358m
Nearby Facilities: Bukit Batok West Hawker Centre (780m); Bukit Batok Public Library (730m); HILLVIEW CRESCENT PLAYGROUND (834m) +1	Nearby Facilities: Clementi Ave 3 Blk 448 (347m); Clementi Public Library (753m); FABER HILLS PLAYGROUND (598m) +1	Nearby Facilities: Ang Mo Kio Ave 4 Blk 160/162 (Mayflower Market) (409m); Ang Mo Kio Public Library (822m); MAYFLOWER CRESCENT PG (344m) +1
While this property at $1000 perfectly fits within your budget, which is your top priority, it's an HDB which...	While the property at 210A Clementi Avenue 6 fits within your budget at $1,300, which you prioritized...	While the property at 108 Ang Mo Kio Avenue 4 fits well within your budget of $1,300 against a max of...
View Map · View Details	View Map · View Details	View Map · View Details
HDB	Condo	HDB


HouseFinder — Welcome | Get Started | Recommendations

budget, which is your top priority, it's an HDB which...	within your budget at $1,300, which you prioritized...	well within your budget of $1,300 against a max of...
View Map · View Details	View Map · View Details	View Map · View Details
HDB	Condo	HDB
122 Potong Pasir Avenue 1	The Blossomvale, 900 Dunearn Road	90 Tanglin Halt Road
Woodleigh	Bukit Merah West	Queenstown
1200 /month	1200 /month	1200 /month
To School 58min · To MRT 660m	To School 30min · To MRT 190m	To School 22min · To MRT 458m
Nearby Facilities: Woodleigh Village Hawker Centre (847m); Toa Payoh Public Library (908m); GREENWAY SOUTH (861m) +1	Nearby Facilities: Bukit Timah Market (773m); HUA GUAN CRESCENT PG (250m); COLD STORAGE SINGAPORE (1983) PTE LTD (88m)	Nearby Facilities: Tanglin Halt Market (107m); Queenstown Public Library (765m); ONE-NORTH PARK (1644m) +1
The property at 122 Potong Pasir Avenue 1 is priced at $1200, fitting comfortably within your top-priority...	While the property at The Blossomvale meets your rent budget perfectly at $1200, its district, Bukit Merah...	This property at $1,200 is comfortably within your budget range, which is highly important to you...
View Map · View Details	View Map · View Details	View Map · View Details
HDB	HDB	Apartment

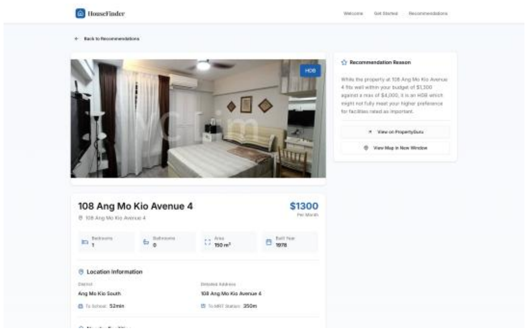
- After you submit your requirements, we will intelligently filter and present a list of the most matching rental properties based on your preferences, allowing you to quickly browse and compare them.

- In the recommendation list, each property clearly displays key information, such as price, address, property configuration, distance to the nearest station, surrounding amenities, and more.

- Most importantly, for each property recommended to you, we will provide corresponding recommendation reasons. This helps you understand how well the property matches your requirements in the shortest time.

- You can also click the "View Map" button to check the property’s location on the map, or click "View Details" to access more detailed information about the property.

2.4 Property Details



108 Ang Mo Kio Avenue 4 — $1300 Per Month

108 Ang Mo Kio Avenue 4

Bedrooms 1 · Bathrooms 0 · Area 150 m² · Built Year 1978

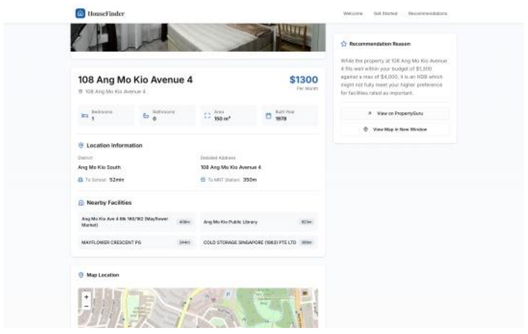
Location Information

District: Ang Mo Kio South · To School: 52min

Detailed Address: 108 Ang Mo Kio Avenue 4 · To MRT Station: 350m

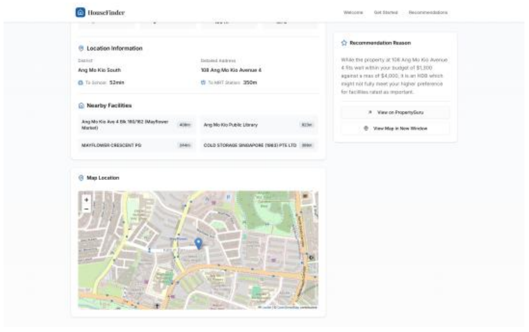
Recommendation Reason

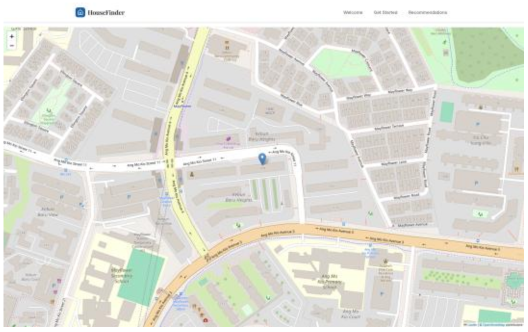
While the property at 108 Ang Mo Kio Avenue 4 fits well within your budget of $1,300 against a max of $4,000, it is an HDB which might not fully meet your higher preference for facilities rated as important.



Nearby Facilities

Ang Mo Kio Ave 4 Blk 160/162 (Mayflower Market) 408m	Ang Mo Kio Public Library 823m
MAYFLOWER CRESCENT PG 244m	COLD STORAGE SINGAPORE (1983) PTE LTD 369m





- On the recommended property details page, we display comprehensive information for each rental property, as shown in the figure.

- The left side presents the property’s detailed information, including real photos of the property, configuration details, location information, estimated commute time, surrounding amenities, and more.

- The right side shows the complete reasons for recommending the current property. This allows you to quickly assess how well it matches your requirements.

- You can click the "View Map in New Window" button to open a new page and check the property’s location on the map in greater detail.

- Additionally, we provide a one-click access to PropertyGuru. You can click "View on PropertyGuru" to navigate to the platform and communicate further with the property agent.