Rent A Space

Developed by:

Taimoor-Ul-Hassan
4303-FBAS/BSSE/F21

Shahid Ramzan
4346-FBAS/BSSE/F21

M. Suleman Faiz
4349-FBAS/BSSE/F21

Supervised By:

Mr. Sana Ullah Shahbaz
Lecturer

Department of Software Engineering

Faculty of Computing and Information Technology

International Islamic University, Islamabad

2025

Final Approval

Date: ______________________________

It is certified that we have read the project report of Rent a Space submitted by M. Suleman Faiz 4349-FBAS/BSSE/F21, Taimoor-Ul-Hassan 4303-FBAS/BSSE/F21 Shahid Ramzan 4346-FBAS/BSSE/F21 under the Supervision of Mr. Sana Ullah Shahbaz. We believe this project meets the requirements for the Bachelor's Degree in Software Engineering set by the Department of Software Engineering at International Islamic University, Islamabad.

Committee:

External Examiner:

Dr. Islam Zada ________________________
Assistant Professor
DSE, FCIT, IIUI

Internal Examiner:

Dr. Shaukat Hayat ________________________
Assistant Professor
DSE, FCIT, IIUI

Supervisor:

Mr. Sana Ullah Shahbaz ________________________
Lecturer

Abstract

Islamabad’s fast-growing rental market still relies on fragmented listings, manual negotiations and minimal price transparency. Rent A Space addresses these gaps by delivering a dedicated, web-based rental platform built with the MERN stack that connects tenants, landlords and property dealers on a single, intuitive interface. The system streamlines every stage of the rental journey: landlords publish high-quality listings with flexible pricing options (including seasonal discounts), while renters access a powerful search engine that filters properties by location, budget, size and location.

A key contribution is an AI-powered rent-price-prediction module implemented in Python. Using a linear-regression model trained on local market data, the module estimates fair rental values These data-driven insights empower users to make informed decisions and foster price consistency across the platform.

The project scope encompasses user authentication, role-based dashboards, property CRUD operations, advanced search with server-side pagination, and the price-prediction micro service. Out of scope are property sales, payment-gateway integration and legal services beyond downloadable contract templates

By focusing exclusively on the Islamabad rental ecosystem and augmenting traditional listings with AI analytics, *Rent A Space* is expected to reduce search effort, improve pricing transparency and ultimately contribute to a more efficient, trustworthy rental marketplace.

Project in Brief

Project Title:	Rent A Space
Undertaken By:	M. Suleman Faiz Taimoor-ul-Hassan Shahid Ramzan
Supervised By:	Mr. Sana Ullah Department of Software Engineering International Islamic University Islamabad
Date Started:	March 2025
Date Completed:	June 2025
Tools, Technology and Language Used:	MERN Stack (MonogDB, Express, React.js, Node.js) Visual Studio Code , GitHub

Acknowledgement

We extend our deepest gratitude to Allah Almighty, whose countless blessings have guided us throughout the completion of this project. His divine support helped us overcome every challenge and remained a constant source of strength, even during our most difficult times.

We are profoundly thankful to our parents for their unwavering support, endless prayers, and unconditional love. They laid the foundation for our education and personal growth, often making great sacrifices to ensure our success. Their dedication and belief in us are truly invaluable.

We would also like to express our sincere appreciation to our esteemed teachers for their continuous guidance, encouragement, and efforts in shaping our ability to think critically and grow intellectually. Their support has played a vital role in our academic journey.

Taimoor-ul-hassan
4303/FBAS/BSSE/F21

Shahid Ramzan
4346/FBAS/BSSE/F21

M. Suleman Faiz
4349/FBAS/BSSE/F21

Declaration

We hereby declare that this software application, in whole or in part, is a product of our original work and has not been copied from any source. It is with great pride that we affirm the development of this software was carried out entirely through our own efforts, under the careful guidance and supervision of our instructors and project supervisor.

We further certify that the content of this report has not been submitted, either in part or in full, in support of any other academic degree or certification at this or any other institution. This software, along with its accompanying documentation, reports, and related materials, is submitted as a partial fulfillment of the requirements for the Bachelor’s degree in Software Engineering.

Taimoor-Ul-Hassan
4303/FBAS/BSSE/F21

Shahid Ramzan
4346/FBAS/BSSE/F21

M. Suleman Faiz
4349/FBAS/BSSE/F21

Dedication

We dedicate this work to our parents and other family members who have always been very supportive throughout this endeavor, as well as throughout our entire degree, and to our respected project supervisor whose guidance made us enable to complete this project with our full effort.

Taimoor-Ul-Hassan
4303/FBAS/BSSE/F21

Shahid Ramzan
4346/FBAS/BSSE/F21

M. Suleman Faiz
4349/FBAS/BSSE/F21

Table of Contents

Contents

Chapter 1 Introduction **10**
1.1 Overview 1
1.2 Scope 1
1.3 Problem statement 2

Chapter 2 System Analysis **3**
2.1 Use Case Diagram 5
2.2 Use case Description in Detail 6
2.3 Class Diagram 18
2.4 System Sequence Diagram 19

Chapter 3 System Design **26**
3.1 Data Flow Diagram 32
3.1 Entity Relationship Diagram 34
3.2 Activity Diagram 35

Chapter 4 System Testing **36**
4.1 Black Box Testing 39
4.2 White Box Testing 39
4.3 Test Cases 39

Chapter 5 Conclusion **46**
5.1 Limitations 47
5.2 Future of the System 48
5.3 Conclusion 49

Chapter 6 User Interface **50**
6.1 Signup Page 51
6.2 Home Page 52
6.3 Add Property 52
6.4 Price Prediction 53

References **54**

List of Figures

Figure 2.1: Use Case Diagram..6
Figure 2.2: Sequence Diagram Registration..30
Figure 2.3: Sequence Diagram Login..31
Figure 2.4: Sequence Diagram Search property and Add favorite...32
Figure 2.5: Sequence Diagram Predict Price...33
Figure 2.6: Sequence Diagram Add Property..34
Figure 2.7: Sequence Diagram Edit Property..35
Figure 2.8: Sequence Diagram Manage Property..36
Figure 3.1: Entity Relationship Diagram..45
Figure 3.2: Data Flow Diagram...42
Figure 6.1: :Signup Page 45
Figure 6.2: Login...42
Figure 6.3: Home Page..45
Figure 6.4: Price Prediction 42

List of Tables

Table 2.1: Use Case for Registration 17
Table 2.2: Use Case for Login 18
Table 2.3: Use Case for Property Search 19
Table 2.4: Use Case for View Property Details 20
Table 2.5: Use Case for Contact 21
Table 2.6: Use Case for View Map 22
Table 2.7: Use Case for Price prediction 23
Table 2.8: Use Case for Add Property 24
Table 2.9: Use Case for Manage listing 25
Table 2.10: Use Case for Seasonal Discount 26
Table 2.11: Use Case for Edit Property 27
Table 2.12: Use Case for Listing Status 28
Table 4.1: Test Case for Registration 51
Table 4.2: Test Case for Login 51
Table 4.3: Test Case for Search Property 52
Table 4.4: Test Case for View Property 52
Table 4.5: Test Case for Add to favorite 53
Table 4.6: Test Case for Price Prediction 53
Table 4.7: Test Case for Contact 54
Table 4.8: Test Case for Add New Property 54
Table 4.9: Test Case for Edit Property 55
Table 4.10: Test Case for Manage Listing 55
Table 4.11: Test Case for Seasonal Discount 56
Table 4.12: Result of Test Cases 56

Chapter 1

Introduction

Rent A Space is a modern web-based rental management platform specifically designed to improve the rental process within Islamabad. It integrates property listings, tenant-landlord interactions, and smart rent predictions using AI. By combining the power of the MERN stack with a Python-based machine learning module, the system simplifies the search, comparison, and decision-making process for both landlords and tenants.

The platform offers advanced search filters, real-time listings, and intelligent rent estimation using a linear regression model, trained on local property data. This ensures transparency in pricing, reduces manual errors, and minimizes time spent in property discovery. Overall, Rent A Space automates and digitizes the property rental ecosystem making it faster, more reliable, and data-driven.

1.1 Overview

Rent A Space aims to transform the traditional rental process by offering an all-in-one platform for property rental management. The system allows landlords to post property listings and define details such as location, size, facilities, and rent. Tenants can explore available spaces using advanced filters and view detailed property profiles. A major feature is the integration of AI-powered rent prediction, which estimates fair rental values based on selected property features, improving trust and decision-making for both parties

1.2 Scope

The scope of Rent A Space covers the end-to-end development and deployment of a property rental platform focused on Islamabad. It includes Creation and management of property listings, Landlord and tenant authentication with role-based dashboards, Integration of AI-based rent price prediction using machine learning, Centralized database for storing user and property data, A clean, intuitive user interface for easy access and interaction.

Although initially targeted for Islamabad’s market, the system is scalable and can be extended to other cities or regions with minimal modifications.

1.3 Problem statement

The current rental system faces several challenges:

Manual listings on social media and outdated portals result in scattered and unverified property information. Lack of price transparency makes it difficult for tenants to evaluate if the asking rent is fair. No data-driven tools exist for predicting rent based on location, size, and facilities.

Landlords and tenants must rely on multiple third-party agents, increasing time, cost, and risk of miscommunication.

Solution

Legalese seamlessly incorporates these functionalities into existing systems, ensuring a user-friendly experience for all stakeholders.

- Introduce a virtual consultation platform, allowing clients to connect with lawyers online, eliminating the need for physical visits.
- Develop a secure online repository for case documents, reducing the necessity for lawyers to be present at the office.
- Implement a digital case management system to eliminate the risk of paper loss and enhance overall document organization.
- Create an efficient online scheduling system, reducing the need for hiring assistants and streamlining administrative tasks.
- Improve accessibility and save time for both clients and lawyers by enabling virtual collaboration, ultimately enhancing the efficiency of legal processes.

Chapter 2

System Analysis

System analysis plays a vital role in understanding the purpose, goals, and scope of the proposed system. It helps identify the flaws in the existing process, define the project’s objectives clearly, and plan a structured path to implementation. For Rent A Space, system analysis forms the foundation for developing a user-centric and efficient rental management platform tailored to the real estate needs of Islamabad.

The first phase of analysis involves a comprehensive investigation of the current rental system, which is largely manual, fragmented, and inefficient. This is followed by clearly identifying the requirements and the scope of the proposed system. A detailed evaluation helps determine whether existing tools can support future goals or whether a custom-built platform is necessary.

Since system understanding requires both logical and creative thinking, unfamiliarity with domain-specific operations can introduce complexity. Therefore, special care is taken to analyze user roles, functional flows, and real-world rental processes before proceeding with development.

The system aims to improve transparency, save time, and offer data-driven insights to all users.

In For this project, we follow the Waterfall Model — a linear and sequential development methodology. Each stage of the project (Requirement Gathering, Design, Implementation, Testing, Deployment, and Maintenance) is completed before moving on to the next.

This approach provides a clear structure and is ideal for projects with well-understood requirements, such as Rent A Space. It ensures that each phase is thoroughly reviewed and validated, minimizing the chance of failure due to overlooked functionality.

The analysis model includes:

1. Use Case Diagram

2. Use Cases with Description

3. System Sequence Diagram

2.1 Use Case Diagram

This A Use Case Diagram is a UML diagram that shows how different users (actors) interact with a system through various functions (use cases) like login or search. It defines the system's scope and uses relationships such as association to represent functionality. Its main purpose is to provide a high-level view of system behavior, helping stakeholders understand and communicate requirements clearly.

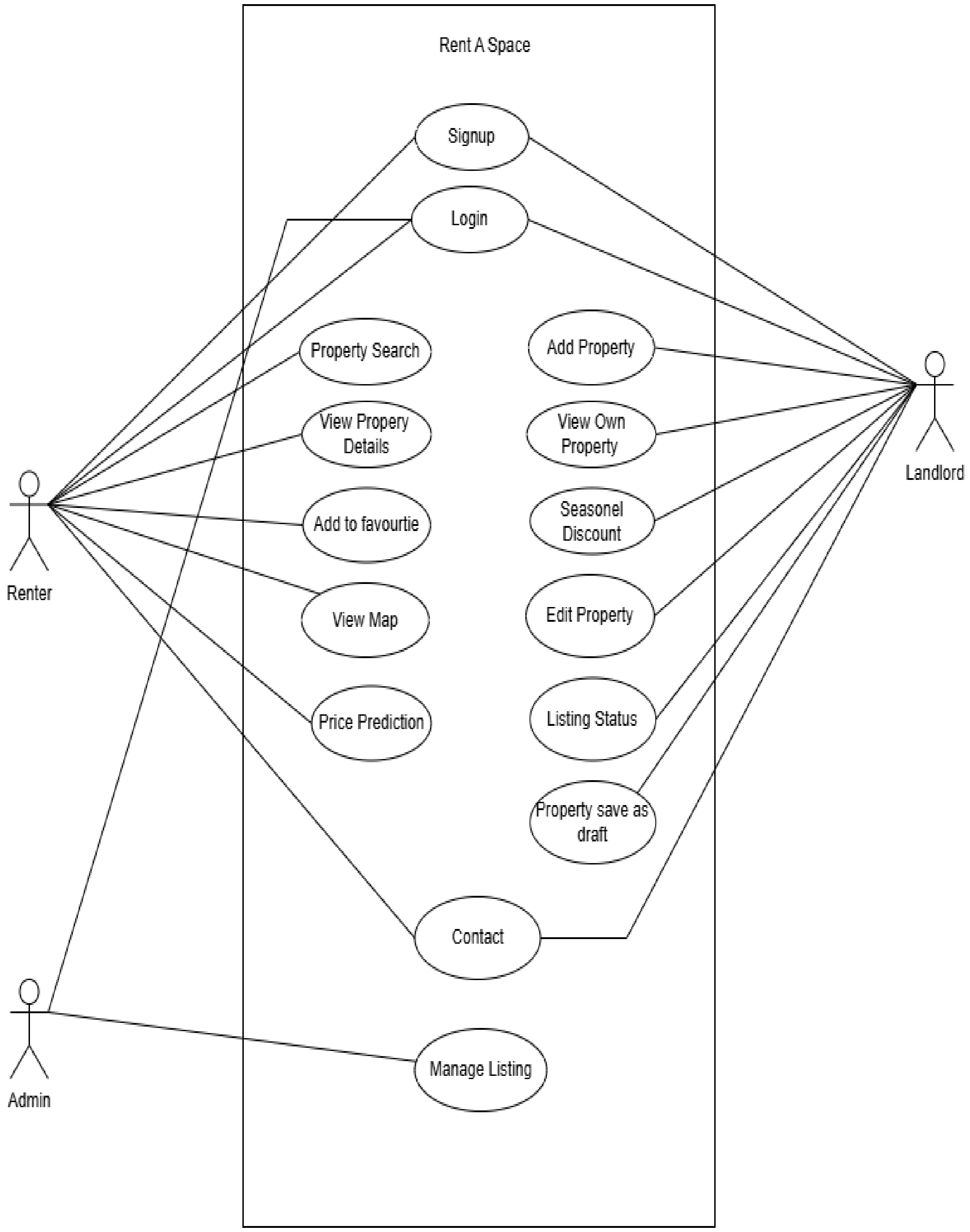


Figure 2.1: Use Case Diagram

2.2 Use case Description in Detail

This section provides a comprehensive explanation of the primary functionalities and Interactions within the system. Each use case outlines the specific roles of users such as Renters, landlords, and admins and describes how they interact with the platform to achieve their goals. It includes step-by-step processes, system responses, and the flow of data for each use case, ensuring a clear understanding of the system's behavior in real-world scenarios.

2.2.1 Registration

This scenario illustrates the detailed procedure of the user registration process within the Rent A Space platform. It outlines how new users, including tenants and landlords, provide their information to create an account. The system validates the input, stores the data securely, and grants access based on the assigned user role.

Table 2.1: Use Case for Registration

Use Case ID:	01
Actor:	Landlords or Renters
Description:	A new user can create an account on the platform.
Pre-Condition:	User must not be already registered.
Post-Condition:	User account is created and stored.
Success-Scenario:	User clicks "Signup". Enters email, password, and other required info. Submits the form. Account is created.
Alternate:	Error in validation (email already used).

2.2.2 Login

This scenario illustrates the detailed procedure of the login process within the Rent A Space platform. It describes how registered users enter their credentials to access the system. The system verifies the provided information and grants access according to the user's assigned role and permissions.

Table 2.2: Use Case for Login

Use Case ID:	02
Actor:	Landlords , Renters, Admin
Description:	Enables a user to access their account.
Pre-Condition:	User must have a valid account.
Post-Condition:	User is logged in and can access features.
Success-Scenario:	User enters email and password. Credentials are verified. User is redirected to dashboard.
Alternate:	An admin may enter incorrect or invalid login details. An admin is not currently registered within our database. Internet Problem/Server Issue.

2.2.3 Property Search

This scenario describes how the system enables users to search for properties using various filters. Users can narrow down listings based on criteria such as location, price range, property type, and size. The search functionality enhances user experience by providing quick and relevant results.

Table 2.3: Use Case for Property Search

Use Case ID:	03
Actor:	**Renter**
Description:	Allows users to search properties based on filters.
Pre-Condition:	User is logged in.
Post-Condition:	The result shown to user

2.2.4 View Property Details

This scenario explains how the system displays complete information about a selected property. When a user clicks on a listing, detailed data such as images, price, location, size, features, and landlord contact information are shown. This helps users make informed rental decisions based on comprehensive property details.

Table 2.4: Use Case for View Property Details

Use Case ID:	04
Actor:	Renter
Description:	User can see full details of a selected property
Pre-Condition:	Search must be performed
Post-Condition:	Property detail page is shown.
Success-Scenario:	The system displays full details of the selected property, including price, location, images, and features.
Alternate:	If no properties match the renter's search filters, the system displays a message like "No properties found." The renter can adjust the filters and search again

2.2.5 Contact

This scenario describes how the system allows users to send messages to landlords or the admin. Users can initiate communication regarding property inquiries, availability, or support issues. The messaging feature ensures seamless and direct interaction within the platform.

Table 2.5: Use Case for Contact

Use Case ID:	14
Actor:	Renter, Landlord
Description:	Renter can contact to landlord through WhatsApp
Pre-Condition:	The property being contacted about must be listed and available.
Post-Condition:	A message or inquiry is delivered to the recipient (property owner)
Success Scenario:	Renter contacts with Landlord for rent purposes

2.2.6 View Map

This scenario explains how the system shows the location of the property on an interactive map. When viewing a property listing, users can see its exact position using an embedded map interface. This helps users assess the property's surroundings and accessibility more effectively.

Table 2.6: Use Case for View Map

Use Case ID:	06
Actor:	Renter
Description:	Shows location of property on map
Pre-Condition:	A property must be selected
Post-Condition:	Map view is displayed
Success-Scenario:	The renter selects a property and clicks the "View Map" option. The system displays the property's exact location on an interactive map successfully.

2.2.7 Price Prediction

This scenario describes how the system predicts property prices using an AI-based model. By analyzing input features such as location, size, and number of rooms, the system estimates a fair rental price. This AI-powered prediction helps users make informed and competitive pricing decisions.

Table 2.7: Use Case for Price Prediction

Use Case ID:	07
Actor:	**Renter**
Description:	Predicts expected rent for a property using AI
Pre-Condition:	Property details must be entered
Post-Condition:	Suggested price is shown
Success-Scenario:	The landlord enters property details such as location, size, and features, then clicks "Predict Price." The system uses the AI model to calculate and display the estimated rent amount successfully.
Alternate:	If the landlord enters incomplete or invalid property details, it will give error message

2.2.8 Add Property

This scenario illustrates the detailed procedure of the Add Property process, which allows landlords to post new rental listings on the platform. Landlords fill out a form with property details such as title, location, rent amount, size, number of rooms, and upload images. The system validates the information and saves the listing to the database. Once approved, the property becomes visible to potential tenants through the search interface. This feature streamlines the listing process and ensures high-quality property entries.

Table 2.8: Use Case for Add Property

Use Case ID:	08
Actor:	**Landlord**
Description:	Landlord can post a new property.
Pre-Condition:	Landlord must be logged in.
Post-Condition:	Property is added and saved (or as draft)
Success-Scenario:	Landlord clicks "Add Property". Fills out property form. Submits property.
Alternate:	Landlord can also view property

2.2.9 Manage listing

This scenario describes how the system allows admins to accept or reject property listings submitted by landlords. After reviewing the details of a listing, the admin can either approve it for public visibility or reject it if it doesn't meet platform guidelines. This moderation process helps maintain the quality and reliability of listings shown to users.

Table 2.9: Use Case for Manage Listing

Use Case ID:	15
Actor:	**Admin**
Description:	Admin can review , accept and reject the properties
Pre-Condition:	User has at least one listed property.
Post-Condition:	Listing changes (accepted, rejected) are saved in the system.
Success-Scenario:	Admin navigates to "Manage Listings" section. System displays a list of properties added by the user. User selects a property and chooses an action (view, accept, reject). System updates the listing as requested. Confirmation message is shown to the user.

2.2.10 Seasonal Discount

This scenario illustrates how the system enables landlords to apply temporary price discounts to their rental listings. Landlords can specify the discount amount and set a start and end date for the promotional period. Once submitted, the system automatically updates the property's displayed rent to reflect the discounted price. This feature helps landlords attract more tenants during off-peak seasons or special promotions, increasing visibility and competitiveness.

Table 2.10: Use Case for Seasonal Discount

Use Case ID:	10
Actor:	Landlord
Description:	Apply discounts to attract more renters in off-season
Pre-Condition:	Property is active
Post-Condition:	Discount is applied.
Success-Scenario:	Owner selects a property and applies discount. System updates pricing.
Alternate:	Invalid date or range: system prompts correction.

2.2.11 Edit Property

This scenario describes how the system allows landlords to edit the details of properties they have already listed on the platform. After logging in, landlords can access their dashboard and view all their active listings. By selecting a specific property, they can update information such as the title, description, rent amount, number of rooms, location, and images. The system validates the new data before saving the changes to ensure consistency and accuracy. Once updated, the revised property details are immediately reflected in search results. This feature ensures that listings remain current, relevant, and informative for potential tenants.

Table 2.11: Use Case for Edit Property

Use Case ID:	11
Actor:	**Landlord**
Description:	Modify property details
Pre-Condition:	Property exists and belongs to user
Post-Condition:	Updated data is saved.
Success-Scenario:	Owner edits listing and saves changes. System confirms update.
Alternate:	Unauthorized edit attempt: system blocks action

2.2.12 Listing Status

This scenario outlines how the system enables landlords to toggle a property listing between active and inactive status. Through their dashboard, landlords can view all their listed properties and change the visibility status with a simple toggle switch or button. When a property is marked as inactive, it is hidden from public search results, preventing tenants from viewing or inquiring about it. This is useful when a property is temporarily unavailable or undergoing maintenance. Conversely, marking a property as active makes it visible again to all users. The system immediately updates the listing status in the database. This functionality gives landlords full control over the availability and visibility of their properties without needing to delete and re-upload listings.

Table 2.12: Use Case for Listing Status

Use Case ID:	12
Actor:	**Landlord**
Description:	Change the status (active, inactive) of a listing.
Pre-Condition:	Property exists
Post-Condition:	Status is updated
Success-Scenario:	Owner toggles listing status. System updates the listing.
Alternate:	Invalid status: system shows error.

2.3 Class Diagram

A Class Diagram shows the system’s classes, their attributes, methods, and relationships. It represents the static structure and serves as a blueprint for system design.

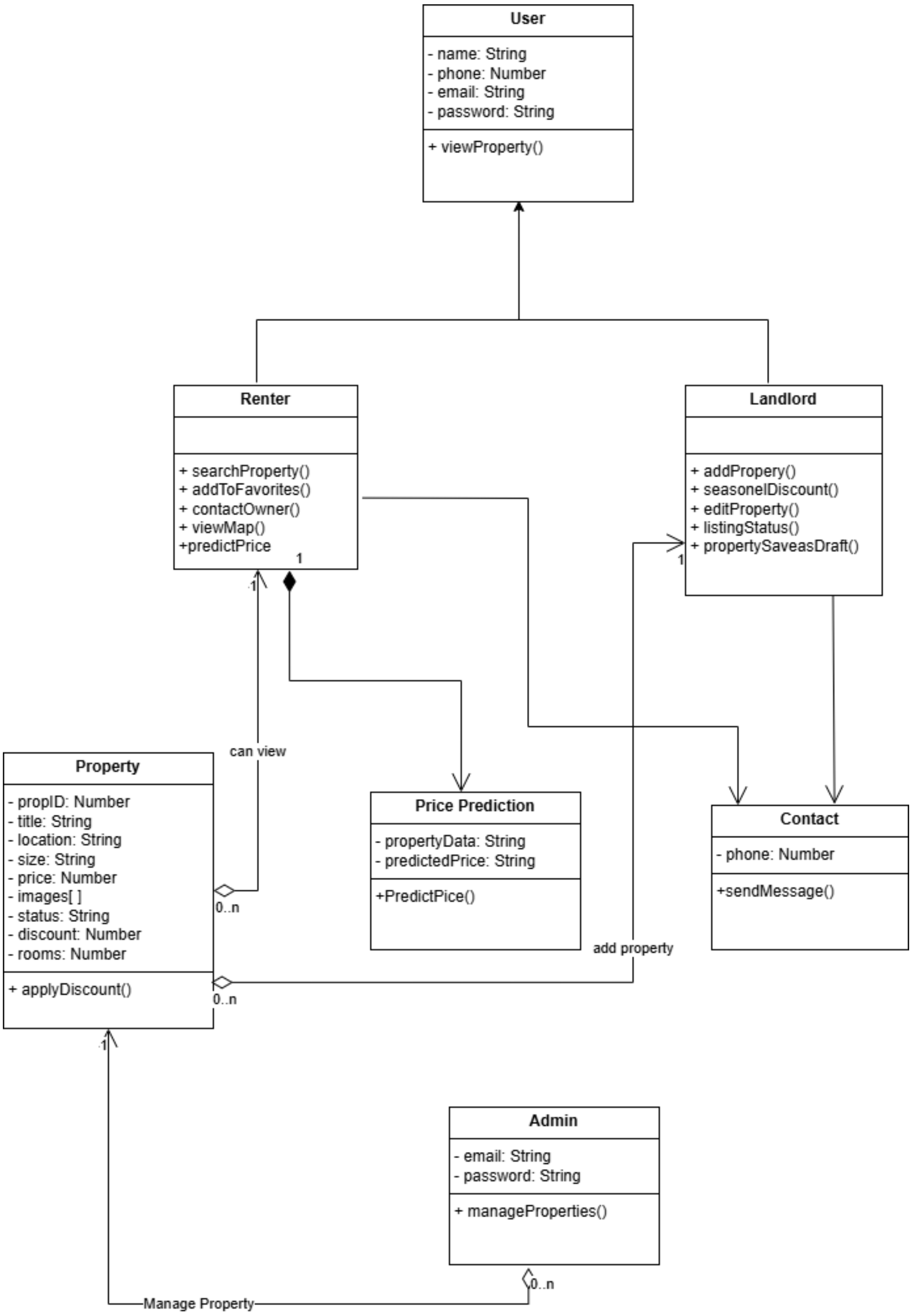


Figure 2.2: Class Diagram

2.4 System Sequence Diagram

A System Sequence Diagram (SSD) visually represents the interactions between an external actor and a system to achieve a specific functionality. It outlines the sequence of messages exchanged between the actor and the system in a concise and structured manner, providing a high-level overview of the system's behavior.

2.4.1 Registration

The registration use case outlines the process by which an actor such as a renter, landlord, or admin signs up on the Rent A Space platform. The user provides necessary details like name, email, password, and role through a registration form. The system validates the input, stores the data securely in the database, and creates a new account. Upon successful registration, the user is redirected to the login page to access their dashboard. This use case is illustrated in Figure 2.3.

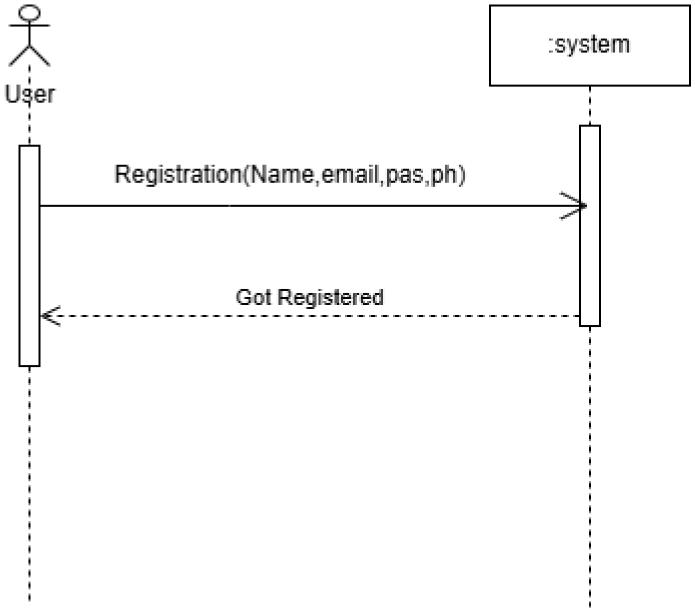


Figure 2.3: Sequence Diagram Registration

2.4.2 Login

The login use case thoroughly details the steps involved when a registered user accesses the system. The user enters their email and password, which the system verifies against stored credentials in the database. Upon successful authentication, the user is redirected to their role-specific dashboard renter, landlord, or admin. If the login attempt fails, the system displays an appropriate error message. This process is illustrated in Figure 2.4, providing a clear overview of the login workflow.

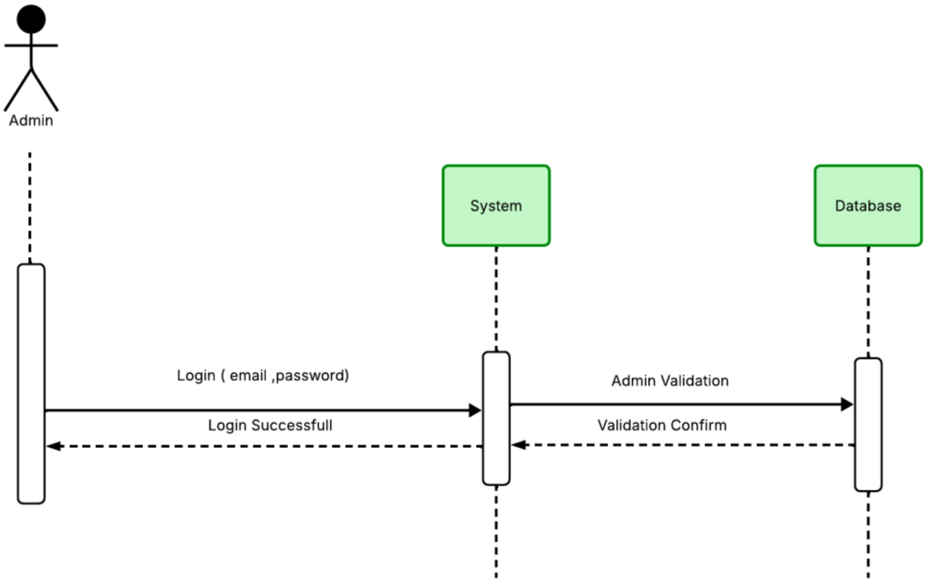


Figure 2.4: Sequence Diagram Login

2.4.3 Search Property and Add to Favorite

The Search Property and Add to Favorite use case describes the steps a user takes to find and save preferred rental listings. The user begins by entering search criteria such as location, price range, and property type. The system processes this input, retrieves matching listings from the database, and displays them on the interface. The user can then view property details and choose to add a specific listing to their favorites. The system saves this selection to the user's favorites list in the database for easy access later. This entire sequence is visually represented in Figure 2.5, offering a clear view of the process flow.

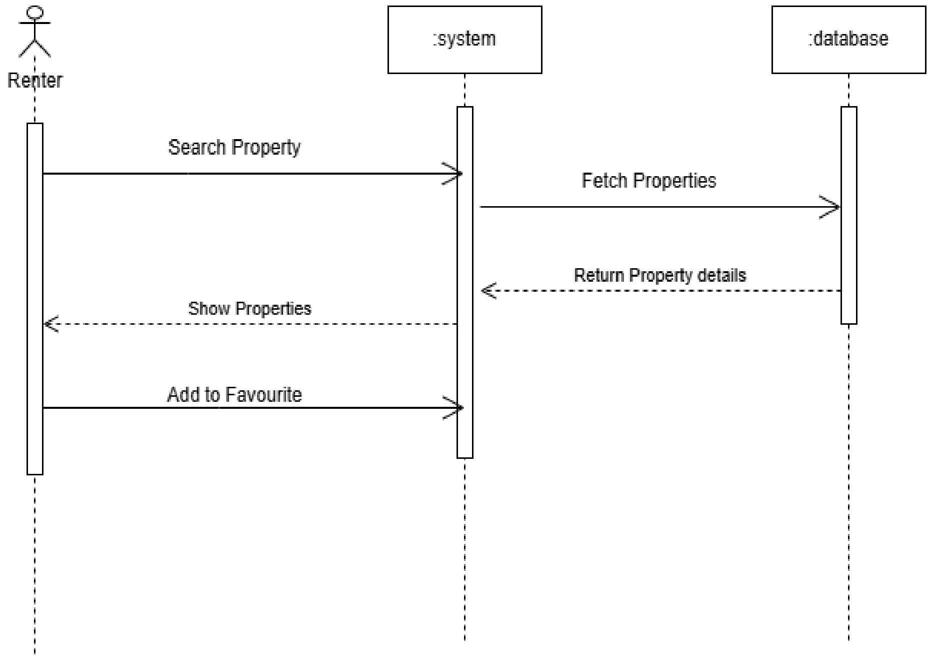


Figure 2.5: Sequence Diagram Add Property and Add to favorite

2.4.4 Predict Price

In the Rent A Space system, the Predict Price use case outlines the sequence of estimating a fair rental price using AI. The user begins by entering property details such as location, size, number of rooms, and other relevant features. These inputs are sent to the AI-powered prediction module, which is built using a linear regression model trained on local rental market data. The system processes the input and generates a predicted rental price based on historical trends and market patterns. This estimated price is then displayed on the user interface, helping landlords and tenants make informed decisions. The complete process is illustrated in figure 2.6, providing a detailed view of the interaction between the user and the AI model.

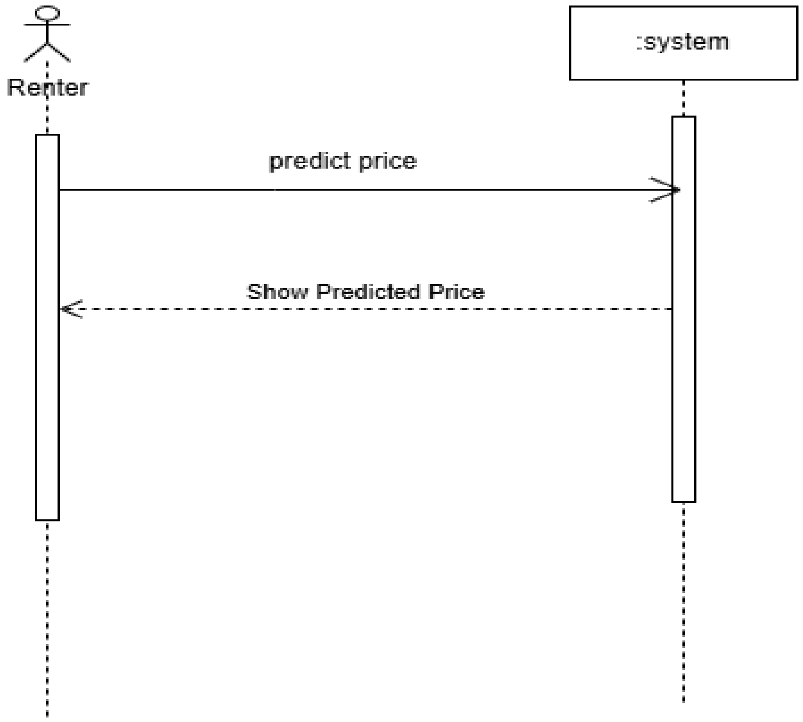


Figure 2.6: Sequence Diagram Predict Price

2.4.5 Add Property

The Add Property use case describes the process by which a landlord posts a new rental listing on the platform. The landlord fills in property details such as title, location, rent amount, size, number of rooms, and uploads images. These inputs are sent to the backend, where the system validates the information for completeness and accuracy. Once validated, the property data is stored in the database and made available for tenants to view and search. This entire sequence is illustrated in figure 2.7, showing the flow from user input to successful listing.

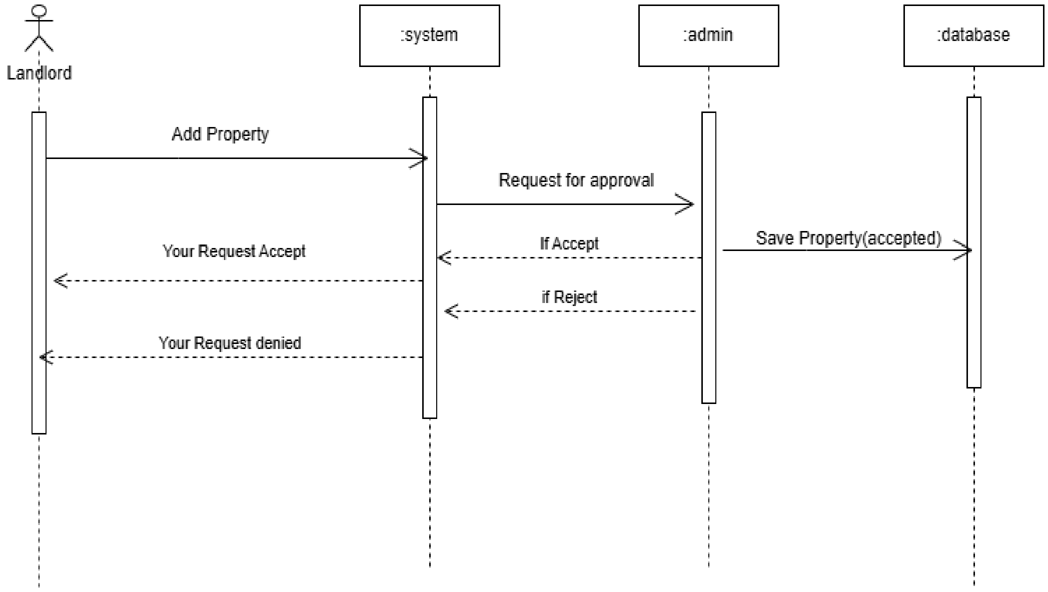


Figure 2.7: Sequence Diagram Add Property

2.4.6 Edit Property

The Edit Property use case outlines the steps a landlord follows to update an existing rental listing on the platform. The landlord selects a property from their dashboard and modifies the desired fields, such as rent amount, description, or images. The system then validates the updated information to ensure accuracy and completeness. Once validated, the changes are saved to the database, and the listing is updated accordingly on the platform. This process is depicted in figure 2.8, illustrating the flow of editing and updating property details.

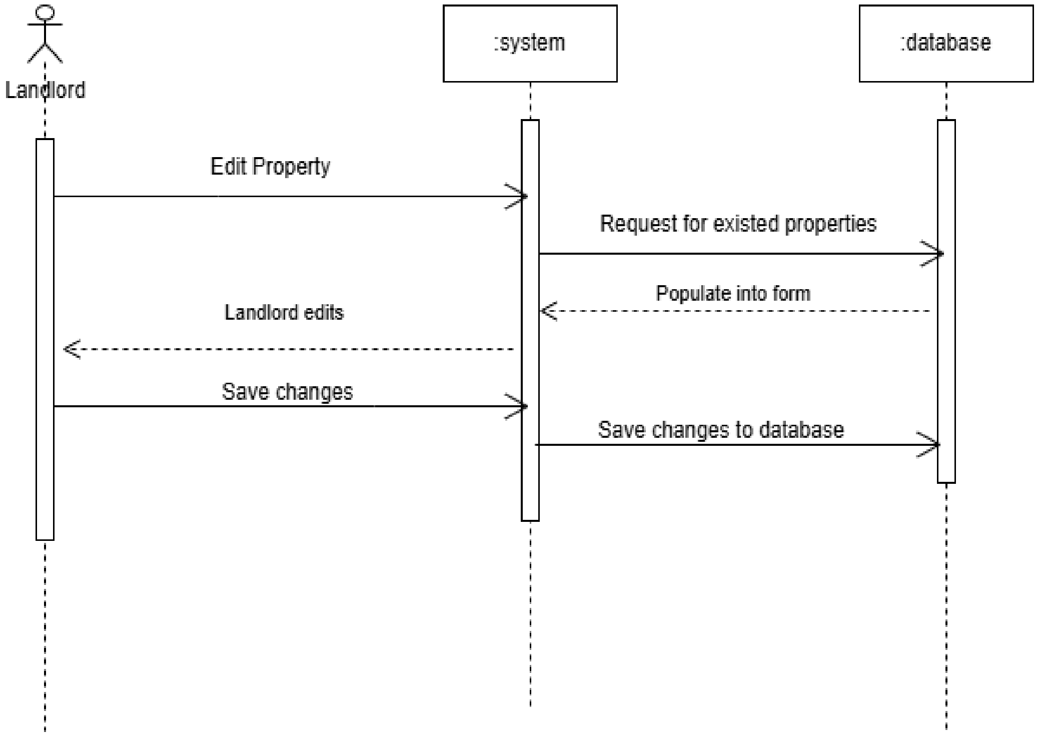


Figure 2.8: Sequence Diagram Edit Property

2.4.7 Manage Property

The Manage Property use case describes how landlords interact with their existing listings on the platform. The landlord accesses their dashboard, where the system retrieves all their listed properties from the database. From this interface, they can choose to edit property details, delete a listing, or toggle its status between active and inactive. This functionality gives landlords full control over the visibility and accuracy of their rental offerings. The entire sequence is illustrated in figure 2.9, showing how property management operations are handled within the system.

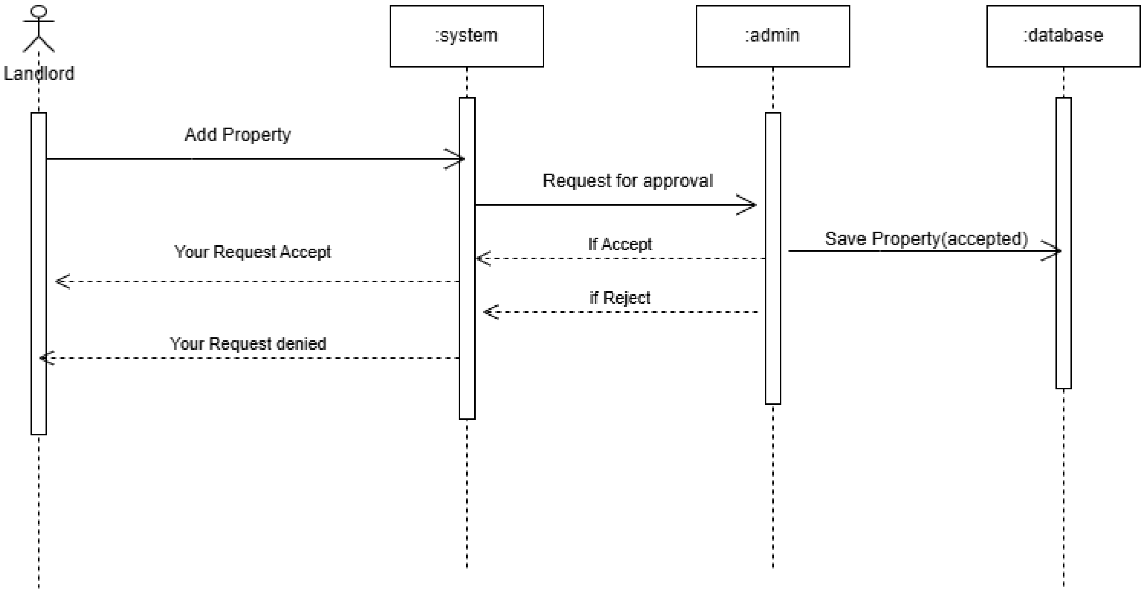


Figure 2.9: Sequence Diagram of Manage Property

Chapter 3

System Design

This explanation breaks down how different types of software components work together and react to specific situations.

The Class Diagram visually represents the various software components, showcasing their characteristics, abstraction, and encapsulation. It also illustrates their methods and how they embody object-oriented concepts. Meanwhile, a sequence diagram provides a visual representation of how these components interact and communicate with each other.

In simpler terms, an Entity-Relationship Diagram (ERD) enhances the understanding of a system by demonstrating the creation and connections between tables. It serves as a visual aid for comprehending the structure of the data.

System design acts as a guide for programmers, helping them determine where to start building software components and how to connect them effectively. It's like a blueprint that assists in the organized construction of a software system.

3.1 Data Flow Diagram

A Data Flow Diagram (DFD) visually represents how data moves through a system, focusing on the processes, data stores, and external entities involved. In the Rent A Space system, the DFD illustrates how renters and landlords interact with processes like login, property search, adding listings, and contacting others. It shows how user inputs are processed, stored in databases, and how results (like property details or predicted prices) are returned. The diagram helps in understanding the system’s logical flow and how data is handled at various stages, making it a valuable tool for analyzing and designing system functionality.

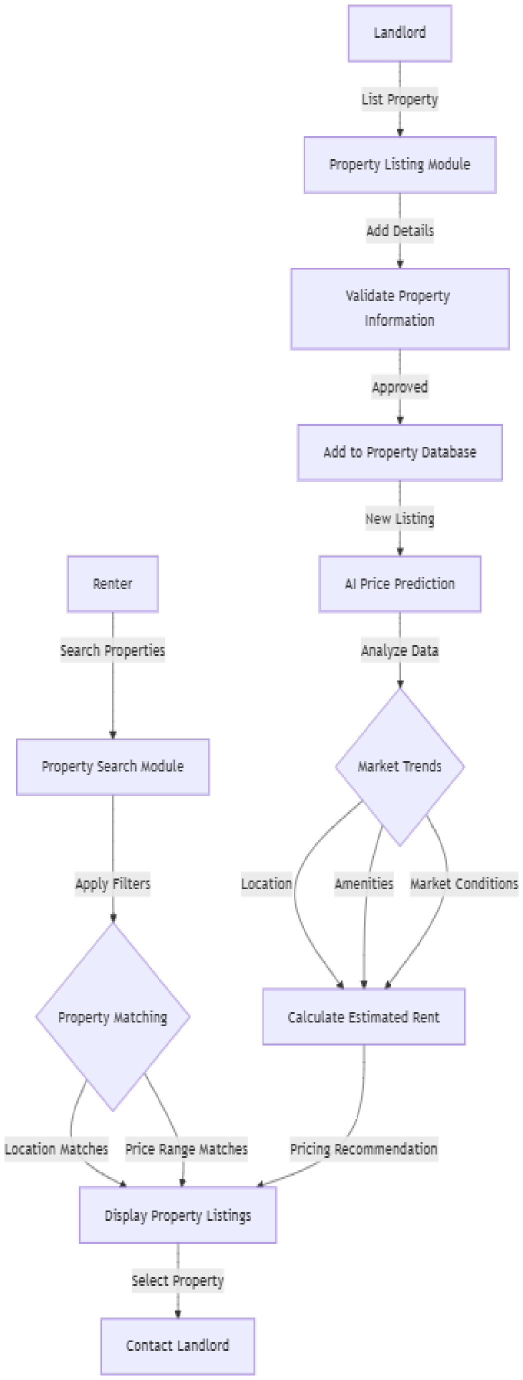
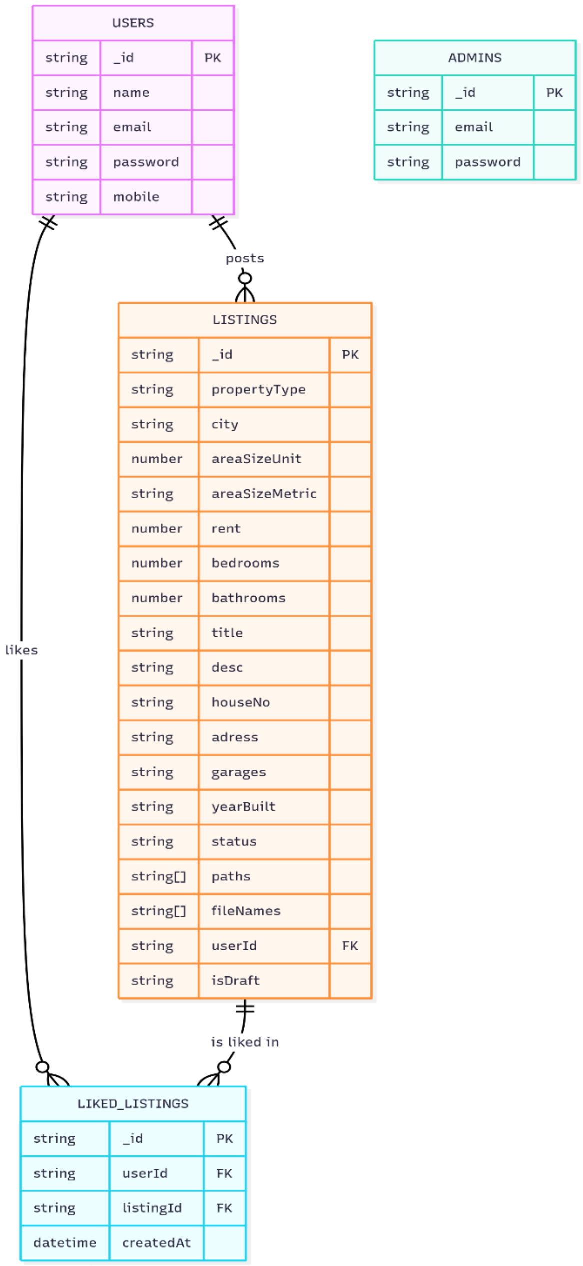


Figure 3.1: DFD

3.1 Entity Relationship Diagram

Entity Relationship Diagram Figure 3.3, is like a map that visually illustrates how things, people, and processes within a system interact. It guides the design of the database by depicting the connections between tables and identifying primary and secondary keys, offering insights into the overall structure and relationships within the system.



USERS		
string	_id	PK
string	name	
string	email	
string	password	
string	mobile	

ADMINS		
string	_id	PK
string	email	
string	password	

LISTINGS		
string	_id	PK
string	propertyType	
string	city	
number	areaSizeUnit	
string	areaSizeMetric	
number	rent	
number	bedrooms	
number	bathrooms	
string	title	
string	desc	
string	houseNo	
string	adress	
string	garages	
string	yearBuilt	
string	status	
string[]	paths	
string[]	fileNames	
string	userId	FK
string	isDraft	

LIKED_LISTINGS		
string	_id	PK
string	userId	FK
string	listingId	FK
datetime	createdAt	

Figure 3.3: Entity Relationship Diagram

3.2 Activity Diagram

Activity diagrams showcase the structured steps of operations at different abstraction levels. They reveal the necessary procedures for completing an action, especially when the process involves multiple goals or interconnected events.

In Rent A Space, it shows how users and landlords interact through processes like signup, search, add listing, and manage property

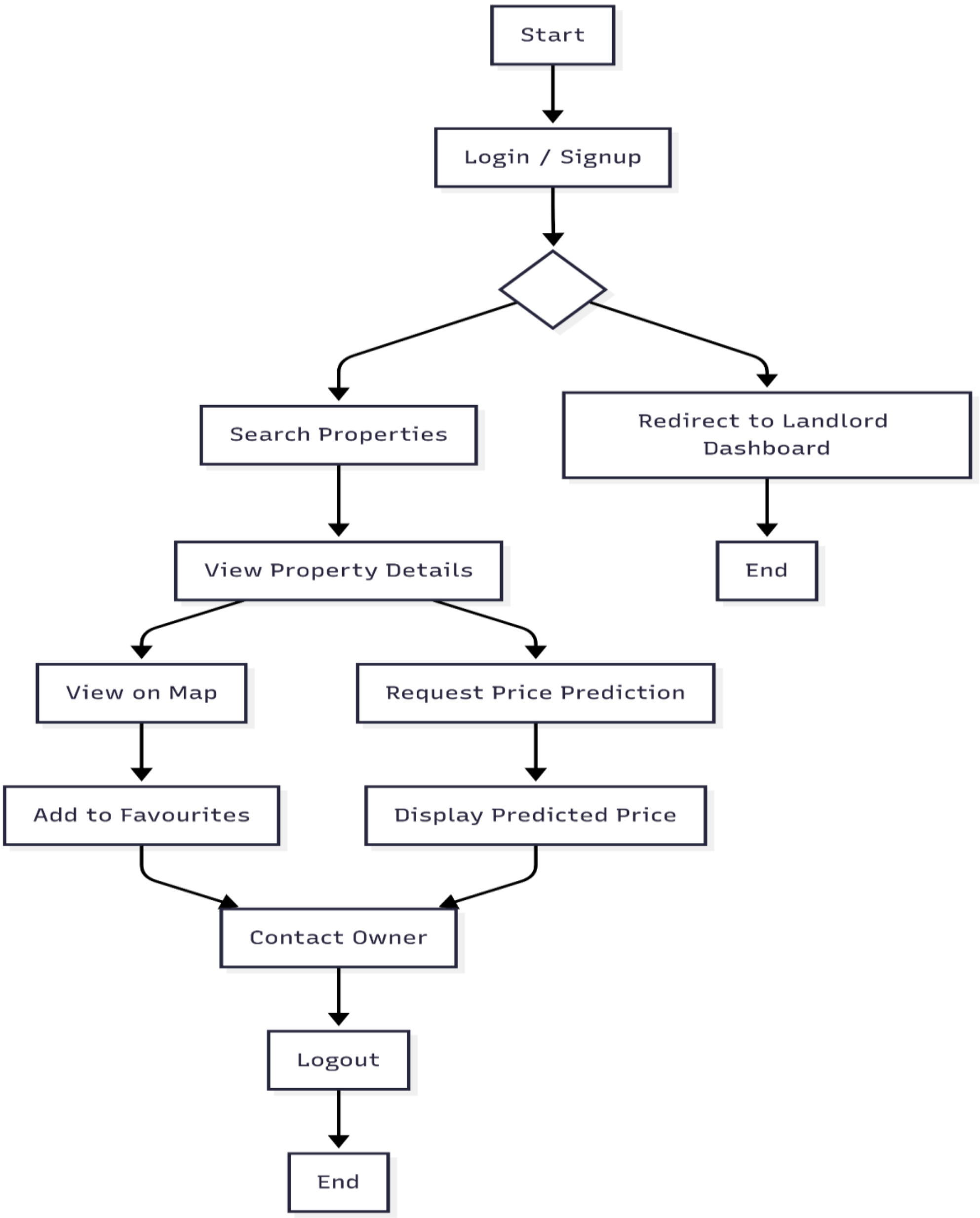


Figure 3.4: Activity Diagram

Chapter 4

System Testing

Testing software is like making sure the end product does what it's supposed to do. It involves checking different parts of the project, like features and tools, to find any mistakes or things that are missing. The goal is to make sure everything meets the requirements set before it goes live.

4.1 Black Box Testing

It is a sort Black box testing is a method where the tester doesn't know how the system works internally but understands its external functions. This type of test checks the overall performance of the system from the outside. The idea is to test the system without knowing its internal workings.

Every time a user logs in, the system produces an output that is then tested. A test case is created by giving the system input and checking the output it gives. The result can be either passing or failing the test.

4.2 White Box Testing

White box testing, also referred to as glass box testing, structural testing, clear box testing, open box testing, and transparent box testing, examines the internal coding and infrastructure of a software. The main emphasis is on verifying predefined inputs against expected and desired outputs.

4.3 Test Cases

System testing involves subjecting an application to specific settings to evaluate its performance. This ensures that all functionalities within the application operate correctly under various scenarios.

4.3.1 Case for Registration

This test case for registration verifies whether a new user can successfully create an account by entering valid information. It checks that all required fields are correctly filled, input is validated, and data is stored in the database. It ensures the user is redirected appropriately after successful registration.

Table 4.1: Test Case for Registration

C-ID:	**01**
Feature	Registration
Purpose	To verify that individuals successfully sign up and gain access to the system.
Action to Perform	Admins are expected to select the Sign-Up button and proceed to register by providing the necessary information.
Prerequisites	The system is functioning and has an active internet connection.
Environment	Rent A Space System
Expected Result	Admins complete the registration process successfully.
Comment	The registration process was successful.

4.3.2 Case for Login

This login test case checks if users can successfully log in with valid credentials. It also verifies that incorrect inputs trigger appropriate error messages.

Table 4.2: Test Case for Login

C-ID	**02**
Feature	Login
Purpose	To verify that registered users can log in with valid credentials.
Action to Perform	Enter a registered email and correct password on the login page.
Prerequisites	The user must already be registered in the system.
Environment	Rent A Space System
Expected Result	User is successfully logged in and redirected to the dashboard.
Comment	Login functionality is working correctly.

4.3.3 Case for Search Property

This test case for search property verifies that users can filter and retrieve listings based on criteria like location, price, and size. It ensures accurate results are displayed according to the entered search parameters.

Table 4.3: Test Case for Search Property

Feature	Property Search
Purpose	To verify that users can search for properties using filters.
Action to Perform	Enter search criteria and submit.
Prerequisites	User must be logged in.
Environment	Rent A Space System
Expected Result	System displays relevant property listings.
Comment	Search results displayed as expected.
Feature	Property Search

4.3.4 Case for View Property Details

This test case for viewing a property verifies that users can access and view complete details of a selected listing. It ensures that information such as images, price, location, and features is correctly displayed.

Table 4.4: Test Case for View Property Details

C-ID	**04**
Feature	View Property
Purpose	To check if users can view full details of a selected property.
Action to Perform	Click on a property from search results.
Prerequisites	Valid search must have been performed.
Environment	Rent A Space System
Expected Result	Property details (price, location, images, etc.) are shown.
Comment	Details viewed successfully.

4.3.5 Case for Add to Favorite

This test case verifies that a user can successfully add a property to their favorites list and that it is stored correctly in the database. It also checks that the property appears in the user's favorites section for future access.

Table 4.5: Test Case for Add to favorite

C-ID	**05**
Feature	Add to Favorite
Purpose	To ensure users can save properties for later viewing.
Action to Perform	Click the favorite icon on a property.
Prerequisites	User must be logged in.
Environment	Rent A Space System
Expected Result	Property is added to the user's favorites list.
Comment	Favorite feature working properly.

4.3.6 Case for Price Prediction

The subsequent test case predicts rent price for renter.

Table 4.6: Test Case for Price Prediction

C-ID	**06**
Feature	Price Prediction
Purpose	To test if AI predicts rental prices based on inputs.
Action to Perform	Fill prediction form and submit.
Prerequisites	AI model should be trained and available.
Environment	Rent A Space System
Expected Result	Predicted price is displayed based on inputs.
Comment	Prediction was generated accurately.

4.3.7 Case for Contact

This test case verifies that a renter can successfully send a message to the landlord through the platform's messaging feature. It ensures the message is delivered, stored properly, and visible to both parties in their respective dashboards.

Table 4.7: Test Case for Contact

C-ID	**07**
Feature	Contact
Purpose	To verify users can send messages to landlords/admin.
Action to Perform	Fill contact form and send.
Prerequisites	User must be logged in.
Environment	Rent A Space System
Expected Result	Message is sent and saved.
Comment	Contact message sent successfully.

4.3.8 Case for Add New Property

This test case is for Add New Property.

Table 4.8: Test Case for Add New Property

C-ID	**08**
Feature	Add Property
Purpose	To check if landlords can post new property listings.
Action to Perform	Enter property details and submit listing.
Prerequisites	User must be a logged-in landlord.
Environment	Rent A Space System
Expected Result	Property is added to the platform.
Comment	Property added without issues.

4.3.9 Case for Edit Property

This test scenario verifies that a landlord can successfully edit the details of an existing property listing. It ensures that the updated information is validated and correctly saved in the database, reflecting changes on the platform.

Table 4.9: Test Case for Edit Property

C-ID	**09**
Feature	Edit Property
Purpose	To confirm landlords can update existing listings.
Action to Perform	Edit fields in listing and save.
Prerequisites	Property must exist.
Environment	Rent A Space System
Expected Result	Changes are saved successfully.
Comment	Edits reflected in the listing.

4.3.10 Case for Manage Listing

This test case is for manage listing.

Table 4.10: Test Case for Manage Listing

C-ID	**10**
Feature	Manage Listing
Purpose	To ensure landlords can view and manage their properties.
Action to Perform	Access the "My Listings" page.
Prerequisites	User must be a landlord.
Environment	Rent A Space System
Expected Result	All listings are displayed with edit/delete options.
Comment	Manage listing works as expected.

4.3.11 Case for Seasonal discount

This test case verifies that a landlord can apply a seasonal discount to a property listing for a specified time period. It ensures the discounted price is correctly calculated, displayed to users, and automatically removed after the discount period ends.

Table 4.11: Test Case for Seasonal discount

C-ID	**11**
Feature	Seasonal Discount
Purpose	To test application of temporary discounts to properties.
Action to Perform	Set discount percentage and time period.
Prerequisites	Property must be listed by landlord.
Environment	Rent A Space System
Expected Result	Discounted price appears during active dates.
Comment	Discount applied successfully.

4.3.12 Results

The table displays the results of tests conducted in the predefined modules.

Table 4.12: Result of Test Cases

TC-01	**Pass**
TC-02	**Pass**
TC-03	**Pass**
TC-04	**Pass**
TC-05	**Pass**
TC-07	**Pass**
TC-08	**Pass**
TC-09	**Pass**
TC-10	**Pass**
TC-11	**Pass**

Chapter 5

Conclusion

5.1 Limitations

Every software project faces certain challenges and limitations, and Rent A Space is no exception. Despite its usefulness and innovative features, some constraints exist that may affect its performance in specific scenarios.

- **Dependence on Internet Connectivity:** As a web-based platform, Rent A Space requires a stable internet connection. In areas with poor connectivity or server issues, the user experience can be interrupted, affecting property browsing, listing updates, or prediction results.

- **Accuracy of AI-Based Price Prediction**: The price prediction module relies on data quality and the accuracy of the AI model. Limited or outdated data may lead to inaccurate predictions, which could mislead renters or property owners during decision-making.

- **Security Risks:** Although basic authentication (e.g., JWT) is implemented, the system remains vulnerable to common cybersecurity threats such as unauthorized access, data leaks, or token hijacking. Without regular updates and audits, sensitive user information may be at risk.

- **Lack of Payment Integration:** Currently, the system does not support built-in payment processing. This limits the ability to manage rent transactions within the platform, requiring users to handle payments externally.

- **Limited Scope to Islamabad Only:** As of now, the platform is designed exclusively for the Islamabad rental market. Users or landlords outside this geographic region cannot benefit from its features unless future expansion is implemented.

5.2 Future of the System

Rent A Space has strong potential for growth and improvement, with several future enhancements that can make it more robust, secure, and user-friendly.

- Enhanced User Experience: Future versions can include real-time notifications, chat systems between renters and landlords, and personalized dashboards. Improving the design for mobile responsiveness and adding support for regional languages will make the platform more accessible.

- **Expansion Beyond Islamabad:** The system can be scaled to include other cities in Pakistan, enabling landlords and renters from different regions to benefit from the platform's features.

- **Integration of Secure Payment Gateway:** Integrating popular payment solutions such as Easy paisa, Jazz Cash, or bank transfers will allow tenants to pay rent and landlords to manage transactions within the system.

- Improved AI & Analytics: Enhancing the AI price prediction algorithm with more comprehensive datasets and machine learning models can lead to higher accuracy. Predictive analytics can also suggest market trends and recommend ideal rent rates.

- Stronger Security Features: Implementing multi-factor authentication (MFA), data encryption, and automated security audits will help protect sensitive user and listing information from cyber threats.

5.3 Conclusion

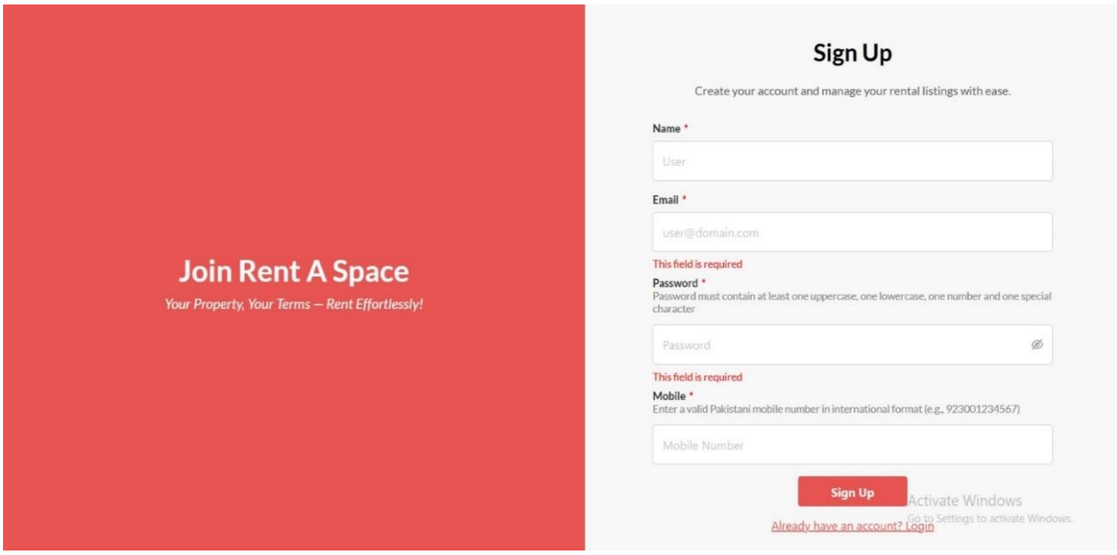
Rent A Space is a modern solution tailored to the needs of renters and landlords in Islamabad. By simplifying the rental process, providing a platform for direct listing and searching, and offering AI-powered price predictions, the system streamlines property management and discovery. The project not only addresses the limitations of broad real estate platforms like Zameen.com but also introduces focused, local innovation using the MERN stack and machine learning technologies.

The system successfully combines user-friendly design, property management tools, and intelligent predictions to create a better rental experience. While limitations exist, Rent A Space sets the foundation for a scalable, secure, and smart rental marketplace, ready for future enhancements and expansion.

Chapter 6

User Interface

6.1 Signup Page



Join Rent A Space

Your Property, Your Terms – Rent Effortlessly!

Sign Up

Create your account and manage your rental listings with ease.

Name *

User

Email *

user@domain.com

This field is required

Password *

Password must contain at least one uppercase, one lowercase, one number and one special character

Password

This field is required

Mobile *

Enter a valid Pakistani mobile number in international format (e.g., 923001234567)

Mobile Number

Sign Up

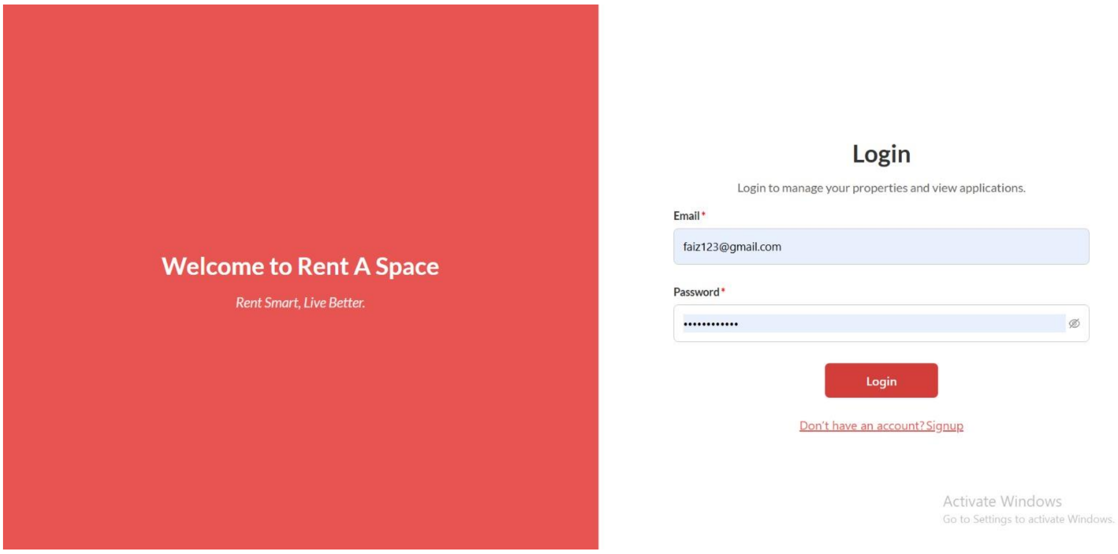
Already have an account? Login

Activate Windows

Go to Settings to activate Windows.

Figure 6.1: Signup Page

Figure 6.1 is the Signup Page



Welcome to Rent A Space

Rent Smart, Live Better.

Login

Login to manage your properties and view applications.

Email *

faiz123@gmail.com

Password *

••••••••••••

Login

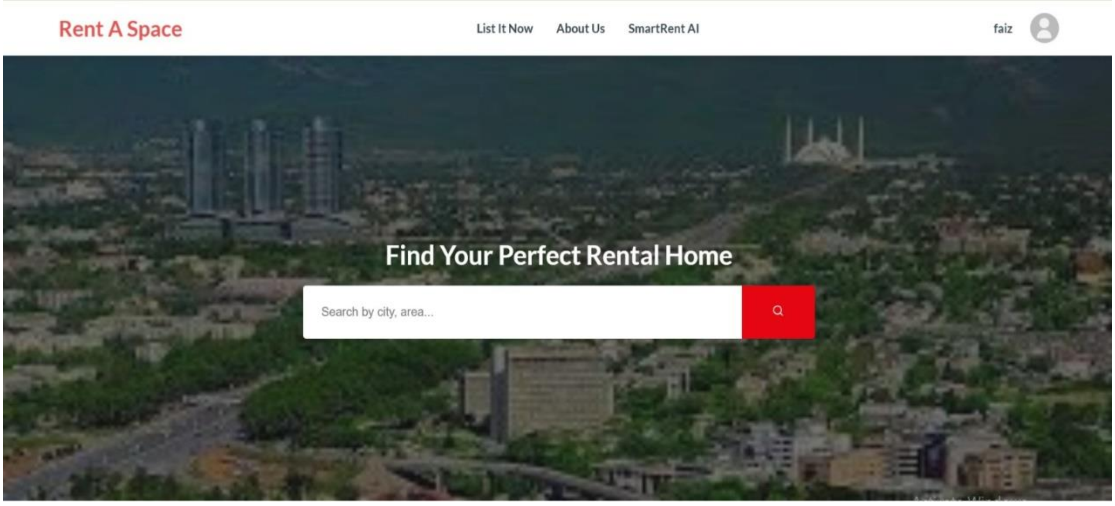
Don't have an account? Signup

Activate Windows

Go to Settings to activate Windows.

Figure 6.2: Login page

6.2 Home Page

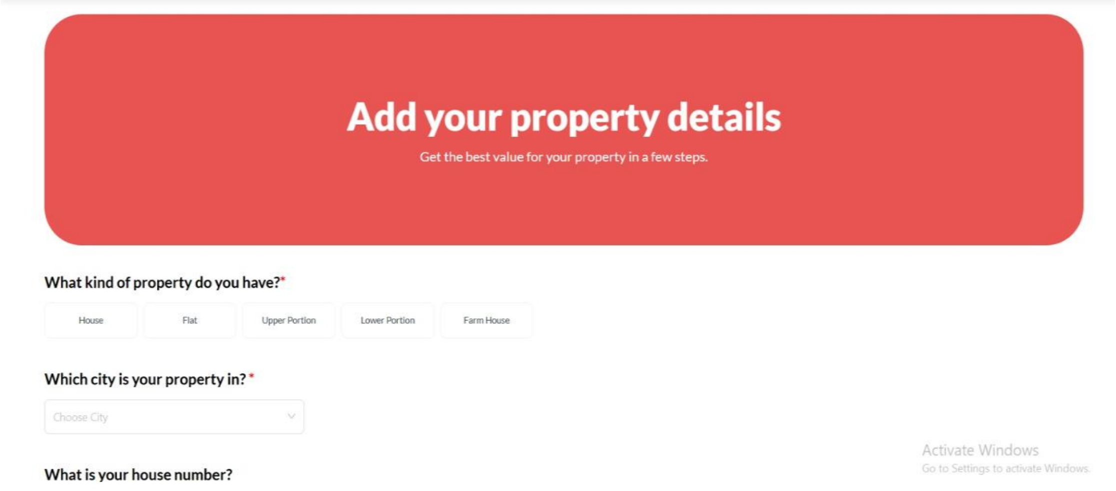


Find Your Perfect Rental Home

Search by city, area...

Figure 6.3: Home Page

6.3 Add Property



Add your property details

Get the best value for your property in a few steps.

What kind of property do you have?*

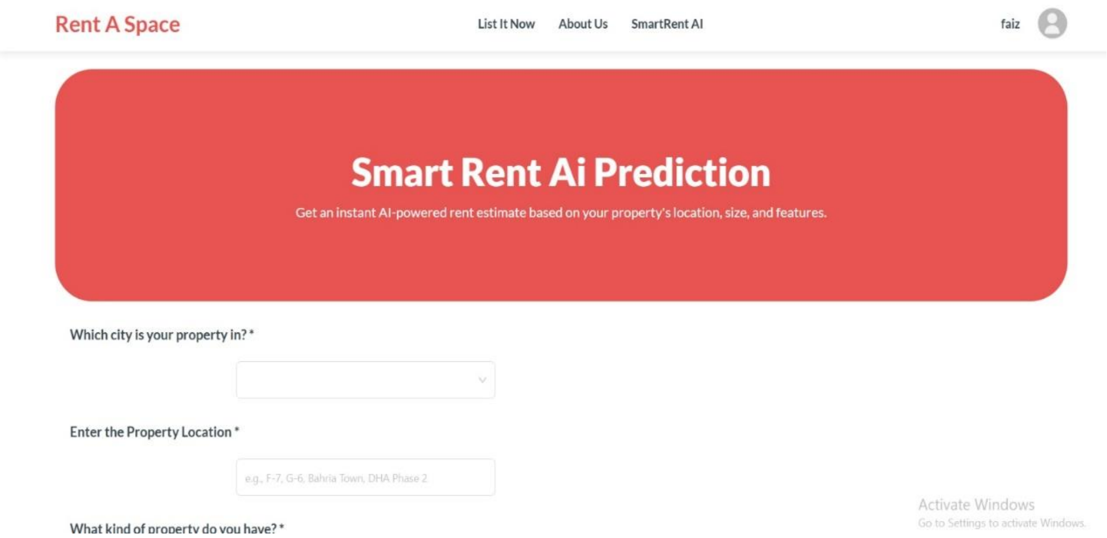
Which city is your property in? *

Choose City

What is your house number?

Figure 6.4: Add Property

6.4 Price Prediction



Smart Rent Ai Prediction

Get an instant AI-powered rent estimate based on your property's location, size, and features.

Which city is your property in? *

Enter the Property Location *

e.g., F-7, G-6, Bahria Town, DHA Phase 2

What kind of property do you have? *

Figure 6.5: Price Prediction

References

[1] https://www.w3schools.com/

[2] https://stackoverflow.com/questions

[3] https://www.kaggle.com

[4] https://app.diagrams.net/draw.io

[5] https://nodejs.org/en

[6] https://mongodb.com