

# PURBANCHAL UNIVERSITY



## DEPARTMENT OF COMPUTER ENGINEERING KHWOPA ENGINEERING COLLEGE LIBALI-08, BHAKTAPUR

### A FINAL REPORT ON **RentEra**

*Project work submitted in partial fulfillment of the requirements for the degree of  
Bachelor of Engineering in Computer Engineering(Fifth Semester)*

Submitted by:

Anisha Kumpakha	(790303)
Binu Prajapati	(790308)
Krishala Shrestha	(790316)
Shreeyukta Lohani	(790340)

**Under the Supervision of**  
Er. Sunil Datheputhe

**Khwopa Engineering College**  
Libali-08, Bhaktapur  
December 2, 2025

# BONAFIDE CERTIFICATE

This is to certify that the project entitled “**RentEra-A new era of renting begins**” submitted by **Anisha Kumpakha (790303)**, **Binu Prajapati (790308)**, **Krishala Shrestha (790316)**, **Shreeyukta Lohani (790340)** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Engineering in Computer Engineering** of Purbanchal University, is a bonafide work to the best of our knowledge and may be placed before the examination board for their consideration.

**Panel Of Examiners:**

**Signature**

**1. External Examiner**

Er. Ganesh Ram Suwal  
Billing Division, NEA  
2082-08-16

**2. Supervisor**

Er. Sunil Datheputhe  
Assistant Lecturer  
Khwopa College of Engineering  
2082-08-16

**3. Head of Department**

Bikash Chawal  
Head of Department  
Khwopa Engineering College  
2082-08-16

# Acknowledgement

First and foremost, we would like to express our heartfelt gratitude to the **Purbanchal University, Khwopa Engineering College**, and the Department of Computer Engineering especially **HOD Er. Bikash Chawal** for providing us this profound opportunity to analyze the problems around us and come up with our project ‘RentEra’ to solve our day-to-day problems while assisting us with all the possible help during the development, implementation and presentation work. Their valuable insight has greatly contributed to our understanding and approach toward problem-solving. .

We would like to extend our heartfelt appreciation to **Er. Sunil Datheputhe** for his constant guidance, suggestions, and support throughout the project.

Anisha Kumpakha (790303)

Binu Prajapati (790308)

Krishala Shrestha (790316)

Shreeyukta Lohani (790340)

# Abstract

RentEra is a dynamic, user-centric web application built using the Django framework, designed to revolutionize the way rental property transactions are managed. In an age where convenience and digital solutions are paramount, RentEra offers an intelligent platform that connects property owners (proprietors) with potential tenants (renters) in a seamless, organized, and secure environment. It eliminates the traditional challenges of property renting such as lack of transparency, scattered communication, and time-consuming manual processes by offering an all-in-one solution that is efficient, scalable, and easy to use. The system is divided into three well-structured modules: ‘proprietor’, ‘renters’, and ‘admin’. Through the Proprietor module, landlords can effortlessly create accounts, list their properties with detailed descriptions, images, pricing, and available amenities, and maintain complete control over their listings. They can easily manage, update, delete, and restore properties at any time, while also tracking the entire rental or sales process in a streamlined and organized interface. On the other hand, the Renters module empowers users with a smart recommendation system that enables them to search for properties using advanced filters such as location, price range, and number of rooms. Renters can explore detailed property pages and directly contact or chat with property owners for quick communication. The Users module ensures smooth authentication, profile management, and role-based access control, making the platform secure and personalized. What sets RentEra apart is its clean design, intuitive interface, and strong focus on enhancing the experience for both property owners and renters, supported by a robust property verification system that authenticates every listing, making the platform exceptionally safe, trustworthy, and reliable for all users. Whether you are a homeowner looking to list a space or a tenant in search of the perfect place to live, RentEra provides the digital infrastructure to make the process smooth, fast, and transparent. As urbanization and digital adoption continue to grow, RentEra aims to be at the forefront of smart property management making renting easier, more reliable, and just a few clicks away.

**Keywords:** *Django Framework, proprietor, renter, tenant*

# Contents

BONAFIDE CERTIFICATE . . . . .	i
Acknowledgement . . . . .	ii
Abstract . . . . .	iii
Contents . . . . .	v
List of Tables . . . . .	vi
List of Figures . . . . .	vii
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Motivation . . . . .	1
1.3 Objectives . . . . .	2
1.4 Scope and Applications . . . . .	2
<b>2 Literature Review</b>	<b>3</b>
<b>3 Requirement Analysis</b>	<b>4</b>
3.1 Software Requirement . . . . .	4
3.2 Hardware Requirement . . . . .	4
3.3 Functional Requirement . . . . .	4
3.4 Non-Functional Requirement . . . . .	5
3.4.1 User Experience . . . . .	5
3.4.2 System Scalability . . . . .	5
3.4.3 Data Security . . . . .	5
3.4.4 System Responsiveness . . . . .	5
3.4.5 Availability and Reliability . . . . .	6
<b>4 Feasibility Study</b>	<b>7</b>
4.1 Technical Feasibility . . . . .	7
4.2 Economic Feasibility . . . . .	7
4.3 Operational Feasibility . . . . .	7
4.4 Schedule Feasibility . . . . .	7
<b>5 Methodology</b>	<b>8</b>
5.1 Incremental Model as Software Development Model . . . . .	8
5.2 Phases Completed So Far . . . . .	8
5.2.1 Requirement Analysis . . . . .	9
5.2.2 System Design . . . . .	9
5.2.3 Increments . . . . .	9
5.3 Work Breakdown of Completed Works . . . . .	9
<b>6 System Design and Architecture</b>	<b>10</b>
6.1 Use Case Diagram . . . . .	10
6.2 Entity Relationship Diagram . . . . .	11

<b>7 Block Diagram and Description of the System</b>	<b>12</b>
7.1 Block Diagram . . . . .	12
7.2 Description . . . . .	13
<b>8 Work Done</b>	<b>14</b>
8.1 User Authentication . . . . .	14
8.2 Property and their details . . . . .	15
8.3 AI Chatbot . . . . .	17
8.4 Dashboards . . . . .	18
8.5 Adding Property . . . . .	22
8.6 Payment Process . . . . .	22
<b>9 Future Enhancements</b>	<b>26</b>
9.1 Review and Rating System . . . . .	26
9.2 Image Processing for Verification . . . . .	26
9.3 Location and environment analysis . . . . .	26
9.4 Map Navigation . . . . .	26
9.5 Codebase Optimization for differently abled users . . . . .	26
<b>Bibliography</b>	<b>27</b>

# List of Tables

5.1 Work Breakdown of Completed Tasks . . . . .	9
---	---

# List of Figures

6.1	Use Case diagram of RentEra . . . . .	10
6.2	Entity Relationship Diagram of RentEra . . . . .	11
7.1	System Block Diagram . . . . .	12
8.1	User Sign-Up . . . . .	14
8.2	Mail for account activation . . . . .	15
8.3	User Login . . . . .	15
8.4	Properties Recommendation and Filter Page . . . . .	16
8.5	Properties Page . . . . .	16
8.6	Property Detail and Similar Properties . . . . .	17
8.7	AI Chatbot . . . . .	18
8.8	Renter Dashboard . . . . .	18
8.9	Proprietor Dashboard . . . . .	19
8.10	Request section(Proprietor Dashboard) . . . . .	19
8.11	Notifications(Proprietor Dashboard) . . . . .	20
8.12	Chat between Proprietor and Renter(Proprietor Dashboard) . . . . .	20
8.13	Delete properties(Proprietor Dashboard) . . . . .	21
8.14	Restore properties(Proprietor Dashboard) . . . . .	21
8.15	Email notification regarding property expiry and restoration . . . . .	21
8.16	Add Property . . . . .	22
8.17	Payment confirmation . . . . .	23
8.18	Login into PayPal . . . . .	23
8.19	Payment through PayPal . . . . .	24
8.20	Login into eSewa . . . . .	24
8.21	Payment through eSewa . . . . .	25
8.22	Django Admin . . . . .	25

# **Chapter 1**

## **Introduction**

### **1.1 Background**

Over the years, the rising demand for rental properties has been attributed to several reasons such as urbanization, population growth, increased mobility, and changing lifestyle preferences. People relocate for work, career, education, better living standards, etc., and thus there persists the demand for temporary as well as long-term accommodations. Traditional renting methods like relying heavily on physical visitations to rental spaces or searching through newspaper advertisements or word-of-mouth have time and again proved to be tedious, inefficient, and limited in reach [1].

The property owners undergo a nightmare publicizing their rental space to prospective tenants, while renters find it challenging to locate appropriate accommodation that matches their diverse needs and budgets. Without such an easy-to-use, straightforward platform, communication, negotiation, and finalization of agreements would occur with relative difficulty.

Digitalization has created many ways forward that are solutions to such challenges. And this is what an online room rental website can do: connect property owners and renters through a click to make property browsing, advertising, and transactions easily accessible. This kind of system is much simpler and enhances accessibility, efficiency, and satisfaction in the rental process.

The advent of the internet has paved the way for developing online platforms that streamline the rental process by offering a range of features aimed at enhancing user experience. Such platforms provide functionalities like property listing, advanced search filters, online payment systems, messaging capabilities, and user verification [2].

### **1.2 Motivation**

The motivation behind RentEra stems from the challenges faced by both property owners and renters in the traditional rental market, which is often fragmented, slow and lacks transparency. With the rapid growth of urbanization and digital adoption, there is a clear need for a centralized, efficient, and user-friendly platform that simplifies property listing, searching, and communication. RentEra aims to address these challenges by providing an intelligent web application that empowers landlords to efficiently manage their properties, while enabling renters to quickly find suitable homes through advanced search, smart filtering, and a smart browsing experience. By leveraging modern web technologies, RentEra seeks to transform the rental process into a seamless, reliable, and transparent experience for all users.

## 1.3 Objectives

The main objective of our project is:

- The objective of the rental web application “RentEra” is to systemize and optimize the process of finding and renting accommodations, while helping proprietors find suitable tenants based on their uploaded details and renters explore accommodations considering their needs and desires.

## 1.4 Scope and Applications

### 1. Property and Resource Management

Rental system centralizes property's information and providing real-time availability updates. It reduces the struggle to find the optimal use of resources, minimizing empty spaces and ensuring efficient allocation. Automated reports further aid in maximizing property and resource utilization

### 2. Communication Management

Rental system manages customer inquiries, automatic notifications (updates requests), feedback collection, comment section, and chat system. Good communication makes the user experience more enjoyable, operations smoother, and customer satisfaction enhanced.

### 3. Time Management

Users can check real-time availability, send requests, receive notifications, manage their rentals through self-service options, and access a centralized platform for efficient renting services. This streamlines the rental process and helps users save time while easily accessing their desired accommodations

### 4. Remote Work

As part of the system's scope, it overcomes the traditional renting process providing an online platform where residences can be filtered based on user preferences. With 24/7 accessibility, renters can connect with proprietors, allowing users to find and secure accommodations conveniently, without physical visits, making the process more efficient and accessible.

### 5. Education and Collaboration

Rental system as a web application facilitates the educational sector by providing hostel services to students for convenient accommodation near colleges and universities. It also assists collaboration with real estate agents to make finding suitable properties easier, ensuring a seamless experience for students and enhancing access to educational resources.

### 6. Tourism and Travelling

Tourists can easily search for rooms, check real-time availability, and book accommodations based on location, price, and amenities. For long-term stays, tourists can prior contact proprietors to negotiate terms and make deals in advance, streamlining the renting process and ensuring more personalized arrangements for extended stays.

# Chapter 2

## Literature Review

The rise of digital platforms has transformed the traditional rental market, providing streamlined processes for both landlords and tenants. Rental room service systems are integral in facilitating these changes, offering various features such as property listings, tenant management, and payment processing. This literature review explores the evolution, functionalities, and advancements in this field.

The initial phase of rental systems relied on offline methods such as classified ads in newspapers and word-of-mouth referrals. With the advent of the internet, the industry saw a shift toward online listings through websites like Craigslist and Zillow [3]. These platforms allowed for broader reach and easier rental property management.

The rental property market in Nepal has witnessed the emergence of several online platforms aimed at simplifying the process for tenants and property owners. Websites such as NPROOM provide a dedicated space for renting rooms and flats, offering a user-friendly interface where tenants can filter properties by their preferences, and property owners can easily list their spaces [4]. NPROOM aims to make the rental process more accessible and secure by ensuring that all listings are reliable and well-detailed. Similarly, Hamrobazar, a well-known online marketplace in Nepal, includes a real estate section where users can find rental properties such as flat-mates or paying guests. While its focus extends beyond real estate, it remains an essential platform for individuals looking for rental spaces in the Kathmandu Valley and other cities [5].

Another popular platform, Rooms Finder Nepal, provides a simplified approach to finding rental properties, offering a detailed search function that helps tenants find suitable accommodations based on various criteria such as price, location, and property type [6]. Rental Nepal focuses on both renting and buying properties, aiming to facilitate the entire process of real estate transactions within Nepal. Their platform emphasizes legal documentation and the security of property deals, making it a trusted choice for tenants and property owners [7]. Additionally, Gharbheti specializes in helping users find rental spaces, offering a detailed listing of rooms, flats, and houses with specific information such as price, location, and amenities [8].

# Chapter 3

## Requirement Analysis

### 3.1 Software Requirement

Software requirement for our prepared system includes:

- a. Django
- b. Python
- c. SQLite
- d. HTML, CSS, and JavaScript
- e. Visual Studio Code
- f. Git and GitHub
- g. Hunter.io API
- h. PayPal API

### 3.2 Hardware Requirement

To develop and operate the RentEra system efficiently, the following hardware resources are recommended:

- a A personal computer or laptop equipped with a multi-core processor (Intel i5 or equivalent and above) and at least 8 GB of RAM to ensure smooth execution of the development environment.
- b A stable and high-speed internet connection for running the web application, accessing remote databases, and deploying the system to hosting services.

### 3.3 Functional Requirement

The core functionalities of RentEra are designed to meet the needs of three primary roles: Renter, Proprietor, and Administrator.

- a. **Account Management:** New users can register as either Renters or Proprietors, verify their email addresses along with their citizenship, and securely log in to the system. They can also securely retrieve their accounts using an OTP-based verification process if they forget their password.

- b. **Property Exploration:** Renters can search and filter listings based on property type, location, price, and available facilities assisted with the help of a recommendation system and AI chatbot.
- c. **Property Listing Management:** Proprietors can create, update, and delete property listings, upload images, and provide detailed descriptions along with the exact map location while the system verifies the property. As the listing's expiry time approaches, both an in-app notification and an email are sent to the proprietor, providing them with the option to restore the listing for a minimal fee.
- d. **Request and Approval Workflow:** Renters can send inquiry requests to proprietors, who can accept or reject them through their dashboard then further decide to connect through the chatting system.
- e. **Verification and Payments:** Proprietors can pay a verification fee to have their property listings validated by the admin.
- f. **Administrative Controls:** Admin can oversee user accounts, manage property verifications, handle payment records, and maintain the integrity of the platform.
- g. **Profile Customization:** Both Renters and Proprietors can edit their personal details, change passwords, and manage uploaded profile pictures.

## 3.4 Non-Functional Requirement

For optimal performance and user satisfaction, the following non-functional aspects are prioritized in RentEra's design:

### 3.4.1 User Experience

The interface must be visually appealing, intuitive to navigate, and consistent across all pages.

### 3.4.2 System Scalability

The architecture should support growth in terms of concurrent users, property listings, and database size without service disruption.

### 3.4.3 Data Security

All sensitive information, including login credentials and payment data, must be encrypted and protected against unauthorized access.

### 3.4.4 System Responsiveness

Search results, page navigation, and property details should load quickly, ensuring minimal waiting time for users.

### **3.4.5 Availability and Reliability**

The system should be available with minimal downtime and ensure data accuracy during simultaneous transactions.

# **Chapter 4**

## **Feasibility Study**

Before initiating full-scale development, the RentEra project was analyzed to determine whether it can be implemented effectively within the available resources, budget, and timeframe. This assessment covers technical, financial, operational, and scheduling perspectives.

### **4.1 Technical Feasibility**

RentEra is being developed using modern yet widely accessible web technologies. The backend will be built with Django framework (Python) and SQLite for database management, while the frontend will be implemented with HTML, CSS and JavaScript for a clean, responsive design. These technologies are open-source, ensuring no licensing costs, and can run smoothly on standard college or personal computer setups.

### **4.2 Economic Feasibility**

From a financial standpoint, RentEra requires only a modest investment. The main recurring expenses will be domain registration and hosting, both of which are affordable. The use of free development software such as VS Code, Git, and GitHub further reduces costs. Even optional features like payment integration or map-based property search can be tested with free sandbox services before moving to a production environment, allowing budget control throughout the development phase.

### **4.3 Operational Feasibility**

The system is designed to address practical challenges faced by property owners and renters by providing a digital platform for listings, inquiries, and communication. It enhances efficiency by reducing the need for physical visits during the initial rental process. The interface will be user-friendly, ensuring minimal training for end users. Because it complements existing property rental practices rather than replacing them, the adoption process is expected to be smooth and well-accepted.

### **4.4 Schedule Feasibility**

A carefully planned timeline ensures that RentEra can be completed within the allocated period. The schedule includes phases for requirement analysis, UI/UX design, backend development, integration, testing, and deployment.

# **Chapter 5**

## **Methodology**

### **5.1 Incremental Model as Software Development Model**

This rental management system follows the Incremental Development Model, where the core components are developed and delivered in a phased manner, allowing continuous integration and feedback. This model was selected because it enables faster deployment of essential features like property listings and user authentication, while later increments can add complex features such as payment gateways, messaging, and verification.

Initially, the core modules such as renter and proprietor signup/login were implemented. Subsequent increments introduced the property listing module, advanced search and filtering features, and dedicated dashboards for renters and proprietors. As development progressed, additional functionalities such as the request management system, real-time chat between renters and proprietors, an AI-assisted chatbot, backup and restore capabilities, and a personalized recommendation system were integrated. This incremental development approach provided flexibility to adapt to user needs, minimized implementation risks, and allowed each component to be tested thoroughly without waiting for the full system to be completed.

### **5.2 Phases Completed So Far**

The overall project has been completed in three main phase which are:

- a. Problem Identification
- b. Requirement Analysis
- c. System Design
- d. Increment 1: User Authentication ,login & Citizenship verification
- e. Increment 2: Property Listing ,land verification & Detail Page
- f. Increment 3: Payment Integration & Verification
- g. Increment 4: Search, Filter & home page setup
- h. Increment 5: Renter Dashboard & Proprietor Dashboard
- i. Increment 6: Request & Message system
- j. Increment 7: Restore Backup , AI chat & Recommendation system

### **5.2.1 Requirement Analysis**

Essential requirements were gathered, focusing on the dual roles of renters and proprietors. Key features include: Secure signup/login for both user types, Property listing and verification, Search and filter functionality, Communication between users, Dashboard with CRUD functionalities for both user types.

### **5.2.2 System Design**

The system architecture was laid out using the Django framework with Dbsqlite for database support. Wireframes and data models were designed for: User Profiles, Property Listings, Requests and Messaging, Payment & Verification Records.

### **5.2.3 Increments**

The development process began with setting up the Django environment, defining user models, and implementing an authentication system to differentiate between renters and proprietors, along with citizenship verification. Following this, property listing functionality was introduced, allowing proprietors to upload properties with images and verify land ownership, accompanied by detailed property pages for users to browse and explore. Next, a payment integration and verification system was added to ensure the authenticity of listings and streamline transactions. To enhance usability, a search and filtering system was implemented along with the home page setup, enabling users to find properties based on location, price range, and available facilities. Subsequently, dedicated dashboards were created for both renters and proprietors: renters could manage their profiles, save favorites, and update preferences, while proprietors could handle rental requests, update property details, and interact with potential tenants. Further development introduced a request and messaging system for smooth communication between renters and proprietors. Finally, advanced features such as backup restoration, AI-assisted chat, and a recommendation system were integrated, providing a more robust, interactive, and personalized user experience.

## **5.3 Work Breakdown of Completed Works**

A work breakdown planning of work done is given below:

Task Description	1st Week	2nd Week	3rd Week	4th Week
Problem Identification				
Design and Analysis				
Coding				
Implementation & testing				
Overall Code Optimization				
Documentation				

Table 5.1: Work Breakdown of Completed Tasks

# Chapter 6

# System Design and Architecture

## 6.1 Use Case Diagram

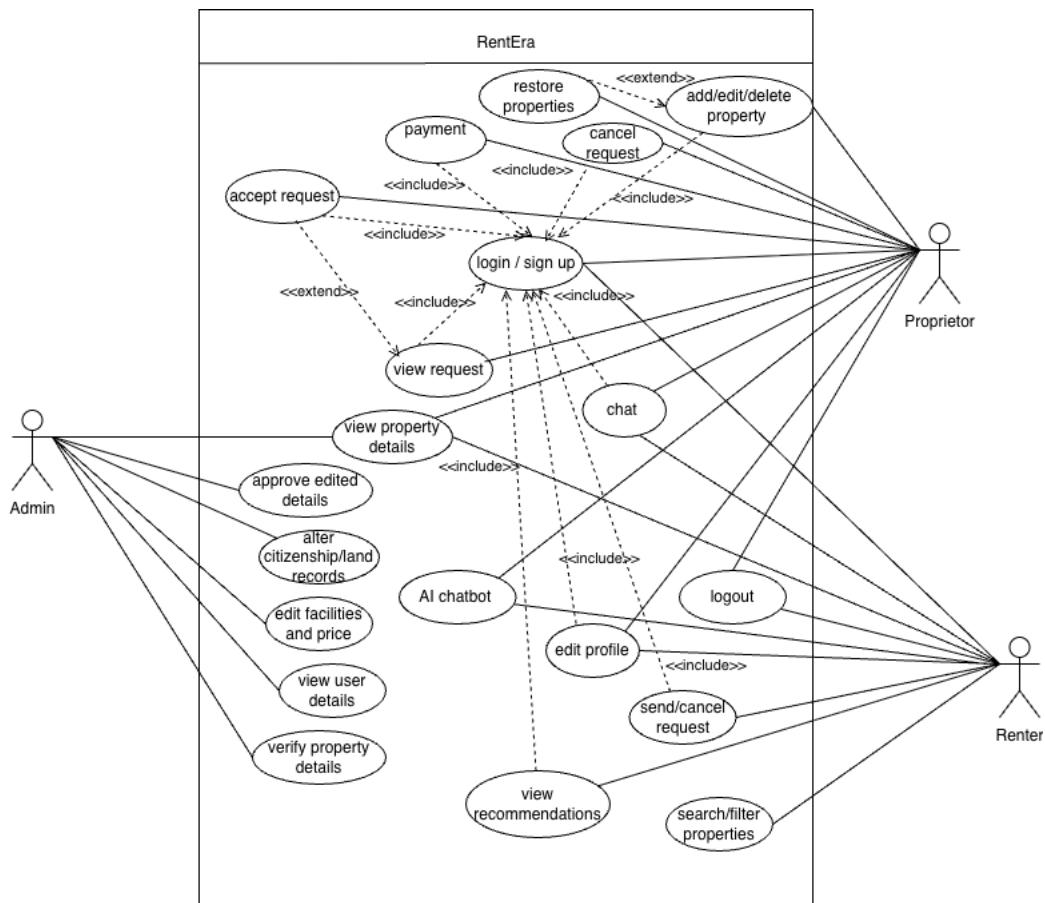


Figure 6.1: Use Case diagram of RentEra

The use case diagram for the RentEra web application presents three actors—Renter, Proprietor, and Admin—and the key functionalities they perform. Renters can authenticate, search and view properties, manage requests, chat, edit profiles and view recommendations. Proprietors can manage property listings, handle renter requests, restore deleted properties, update profiles, and use chat and payment features. Admins oversee verification, approvals, record updates, and user or property details. Shared actions such as login or sign up, viewing property details, viewing requests, chat, and the AI chatbot use include relationships to show essential reused processes, while extend relationships represent optional or conditional actions like canceling or restoring requests. This structure supports modularity and consistent interactions across all RentEra user roles.

## 6.2 Entity Relationship Diagram

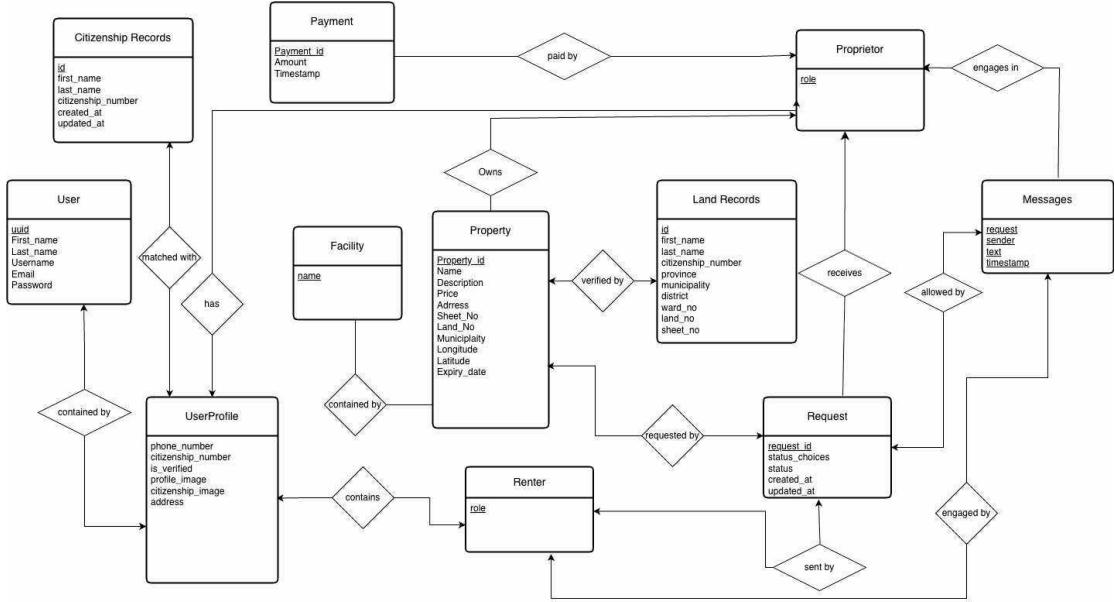


Figure 6.2: Entity Relationship Diagram of RentEra

The ER diagram for the RentEra application outlines the main entities and the relationships that shape the system's data structure. At the center is the **User** entity, which specializes into **Renter** and **Proprietor**, each with distinct roles. Every **User** is linked to a **UserProfile**, and their identity is verified through associated **Citizenship Records**. **Renters** can send **Requests**, with each request being “requested by” a renter and “received” and “allowed by” a proprietor. **Properties** are owned by **Proprietors** and are verified through corresponding **Land Records** to ensure authenticity. Each **Property** contains multiple **Facilities**, showing a contained-by relationship. Communication between renters and proprietors is represented through **Chat**, which both parties can engage in. The diagram also includes **Payment**, which is made by the proprietor. Overall, the ER model captures how users, requests, properties, facilities, verification records, communication, and payments interact cohesively within the RentEra ecosystem.

# Chapter 7

## Block Diagram and Description of the System

### 7.1 Block Diagram

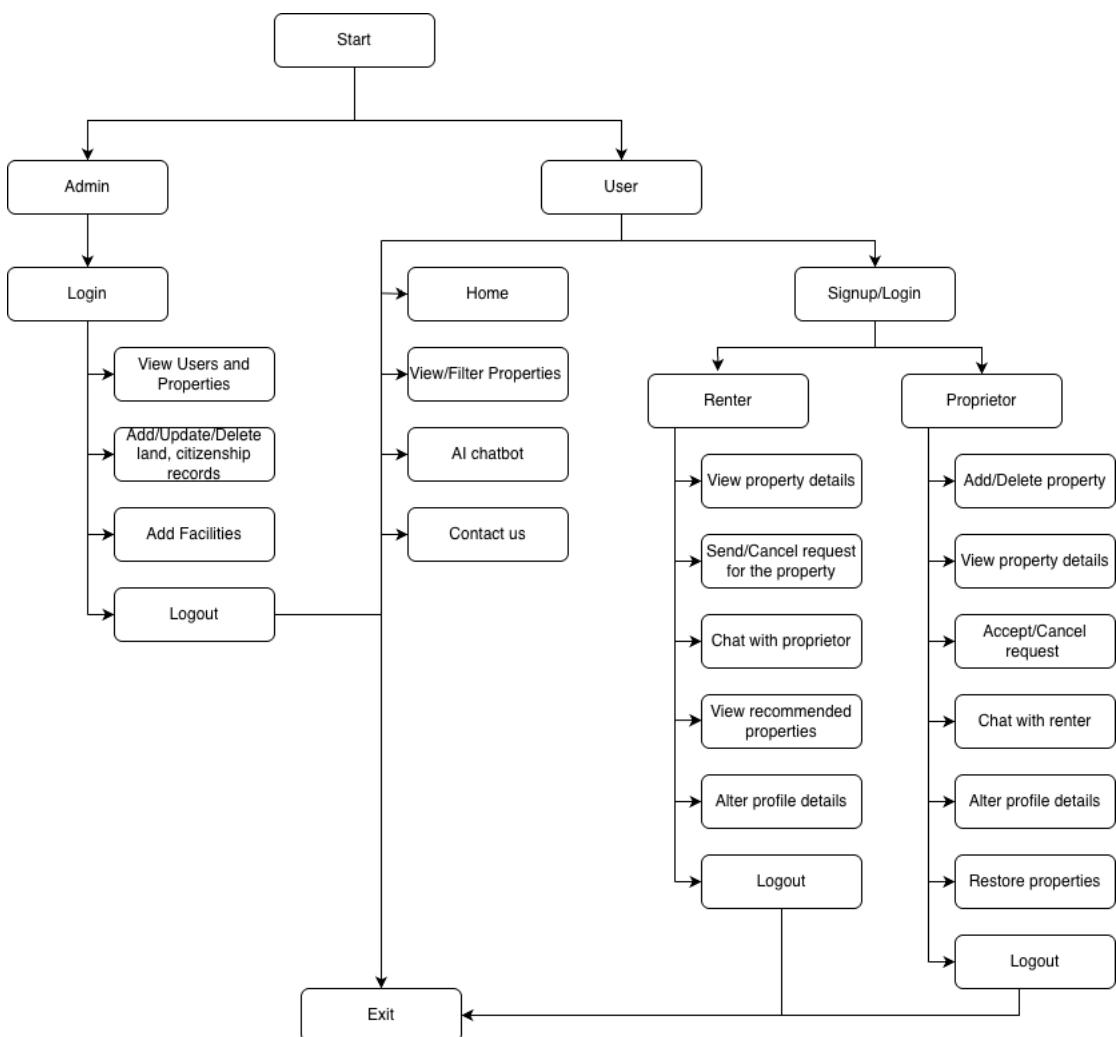


Figure 7.1: System Block Diagram

The block diagram illustrates the workflow of a property rental platform-RentEra, showing the interactions for Admin, Renter, and Proprietor roles. The process begins with the Start node and splits into two main paths: Admin and User. Admin can log in to the system and perform tasks such as viewing users and properties, managing land and citizenship records, adding facilities, and logging out. On the user side, Users can access the Home section to view or filter properties, use AI chatbot and contact support.

Users can also sign up or log in to proceed as either a Renter or a Proprietor. Renters can send or cancel property requests, view property details, view recommended properties, chat with proprietors, update their profile details, and log out. Proprietors, on the other hand, can view property details, add or delete properties, restore the properties after they expire, chat with renters, accept or cancel requests, update their profiles, and log out. Finally, the process concludes at the Exit point, representing the end of user interaction with the system.

## 7.2 Description

This rental management web app lets users browse and filter rental properties like rooms, apartments, offices, and houses. It displays top listings with details stored in the database, including pricing and services. Users register as Renters or Proprietors with email verification. Proprietors manage property listings uploading details, images, paying verification fees, and completing land verification via the database while Renters can view properties, send requests to proprietors, and chat once requests are accepted. Renters also send requests and manage profile settings. The system securely handles authentication, data storage, communication, log-in checks, log-out, an AI chatbot, and live chat to ensure a smooth and consistent user experience.

# Chapter 8

## Work Done

The development of RentEra ( web application) has been carried out in multiple incremental stages. Following are the work done:

### 8.1 User Authentication

The Signup page allows users to register as either a Renter or a Proprietor by selecting their role and submitting required details such as name, username, email, phone number, password, and citizenship number. Renters also provide preferences like location and services. The system checks if the email exists and validates the citizenship number for authenticity. A verification email is sent, and users can only Sign in after successful email verification. The Signin page requires valid credentials, and only verified users can access their respective dashboards. All data is securely stored and managed through the backend database.

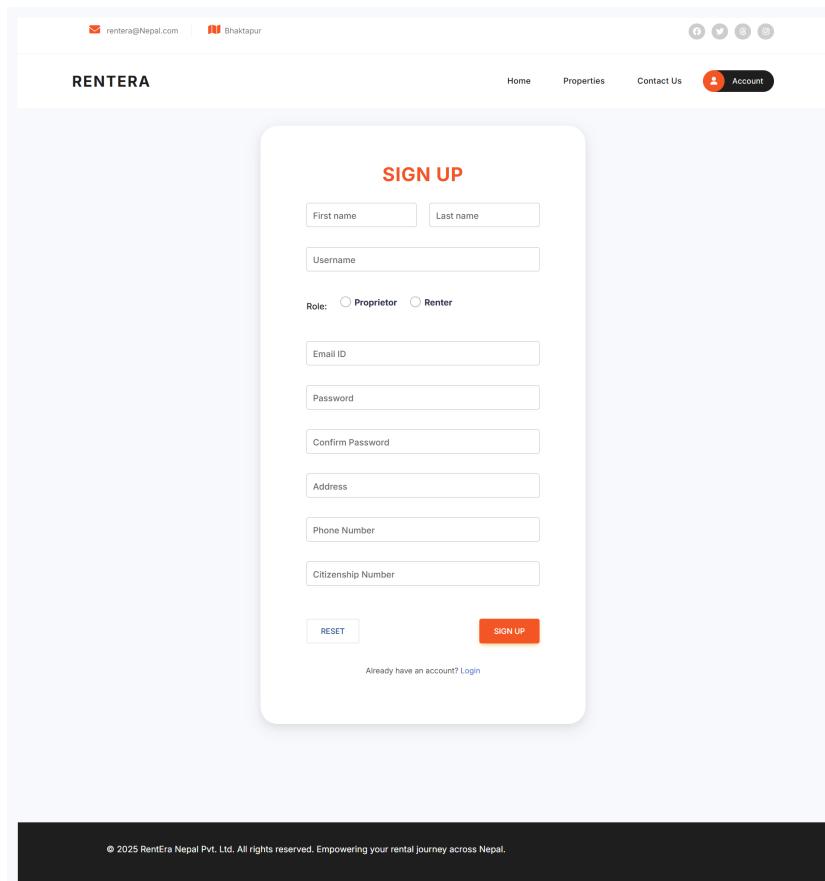


Figure 8.1: User Sign-Up

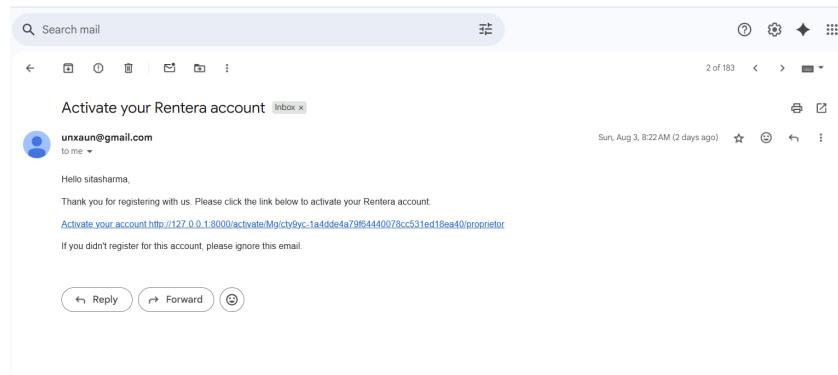


Figure 8.2: Mail for account activation

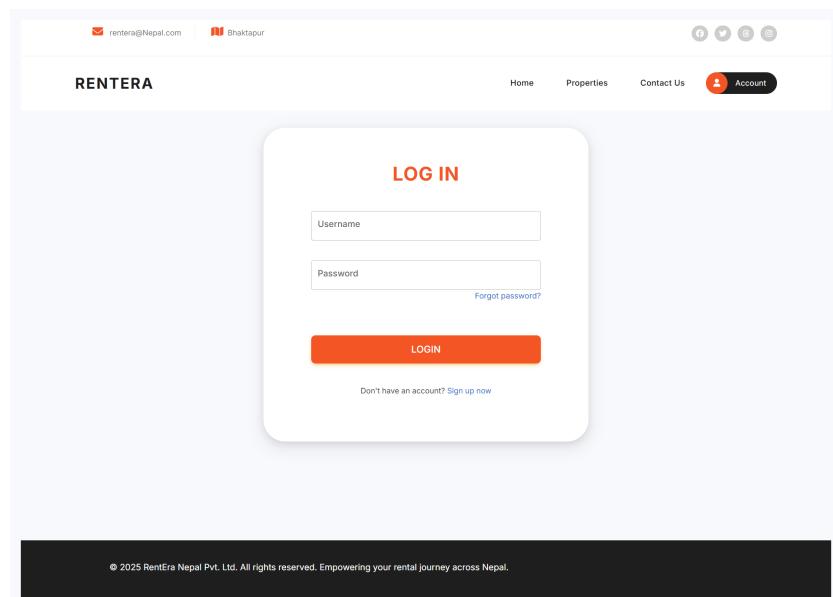


Figure 8.3: User Login

## 8.2 Property and their details

The users can view the different properties and their details. User can select from recommended properties altered based on their interactions. Users can also filter and search the properties according to different criteria like max price, facilities, province, etc. in the 'property page'. The following is the model used for building this feature:

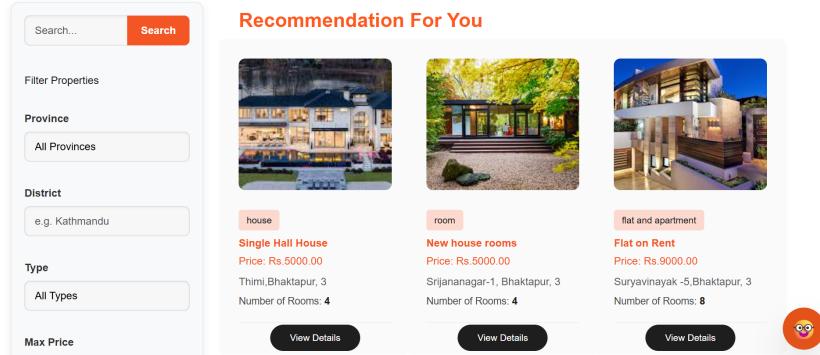


Figure 8.4: Properties Recommendation and Filter Page

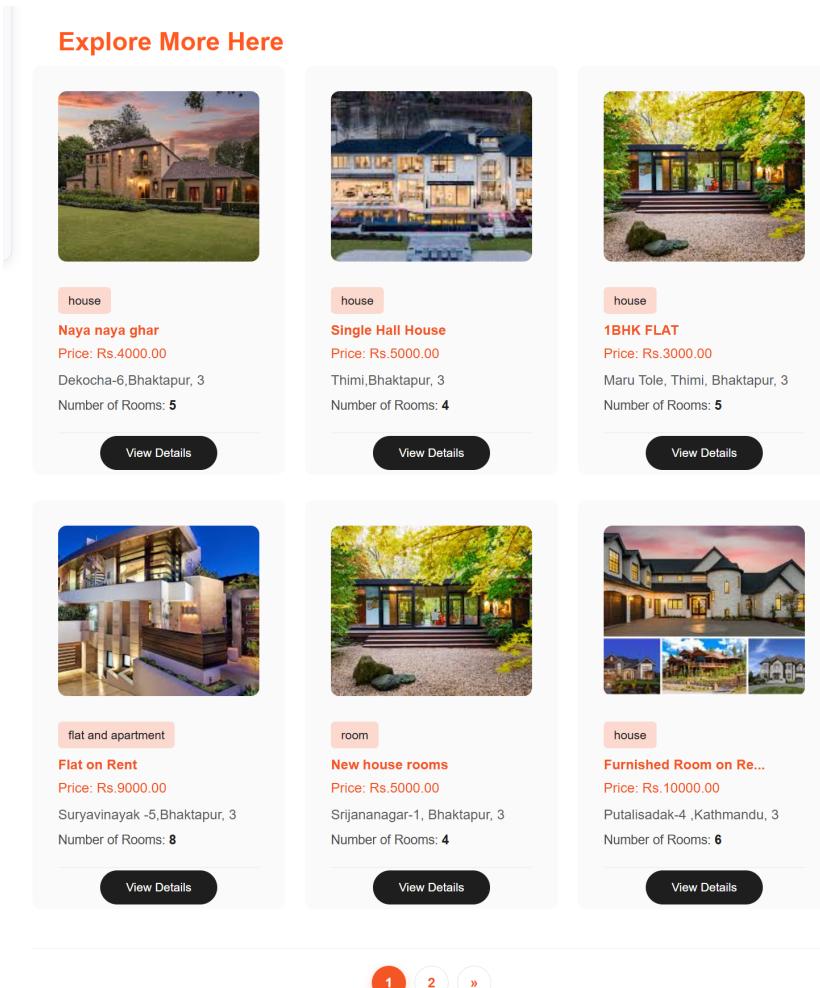


Figure 8.5: Properties Page

Similarly, users can view the details of the properties with their address, price, number of rooms, map, etc. Renters can only send the request to the proprietors of the desired property to confirm more details. Also properties similar to the one they clicked are displayed below property detail to find other similar properties.

[rentera@Nepal.com](#)

**RENTERA**

Home Properties Contact Us Dashboard

Naya naya ghar

Location: Dekocha-6, Bhaktapur, Bhaktapur, 3  
Municipality: Bhaktapur  
Ward No: 6  
Property Type: house

Description:  
Peaceful, ready-to-move 2nd-floor rooms ideal for a single, couple, or small family. 1 Bedroom | Dining | Kitchen | Bathroom | Large East-Facing Balcony Garden, terrace & rooftop seating with natural views Parking  
No specific facilities listed.

Location on Map:

[Send Request](#)

### SIMILAR PROPERTIES

room

**New house rooms**  
Price: Rs. 5000.00  
Srijananagar-1, Bhaktapur, 3  
Number of Rooms: 4

[View Details](#)

house

**Single Hall House**  
Price: Rs. 5000.00  
Thimi, Bhaktapur, 3  
Number of Rooms: 4

[View Details](#)

flat and apartment

**Flat on Rent**  
Price: Rs. 9000.00  
Suryavinyay -5, Bhaktapur, 3  
Number of Rooms: 8

[View Details](#)

© 2025 RentEra Nepal Pvt. Ltd. All rights reserved. Empowering your rental journey across Nepal.

Figure 8.6: Property Detail and Similar Properties

## 8.3 AI Chatbot

Through the AI Chatbot in RentEra, users can receive instant assistance while navigating the platform. Users can ask questions, view property-related guidance, send connection requests, manage their profiles, and understand the payment workflow with ease.

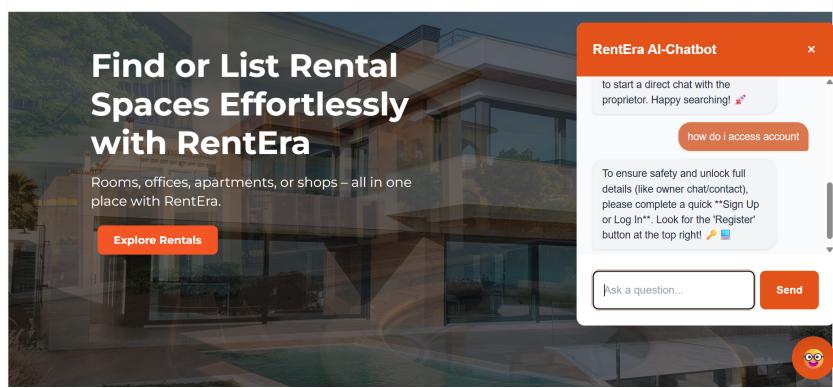


Figure 8.7: AI Chatbot

## 8.4 Dashboards

Both the users renter and proprietor have their respective dashboards.

### a. Renter

Renter can view their personal details, edit the details through their dashboard. Similarly, they can check their notification to be notified of the state of their requests. They can also check their requests and see their list of favorites properties.

Figure 8.8: Renter Dashboard

### b. Proprietor

Proprietor can also view their personal details, edit the details and access their notifications through their dashboard. They can also view the properties they added

in the dashboard. The requests they received and the renter lists are also shown in the dashboard.

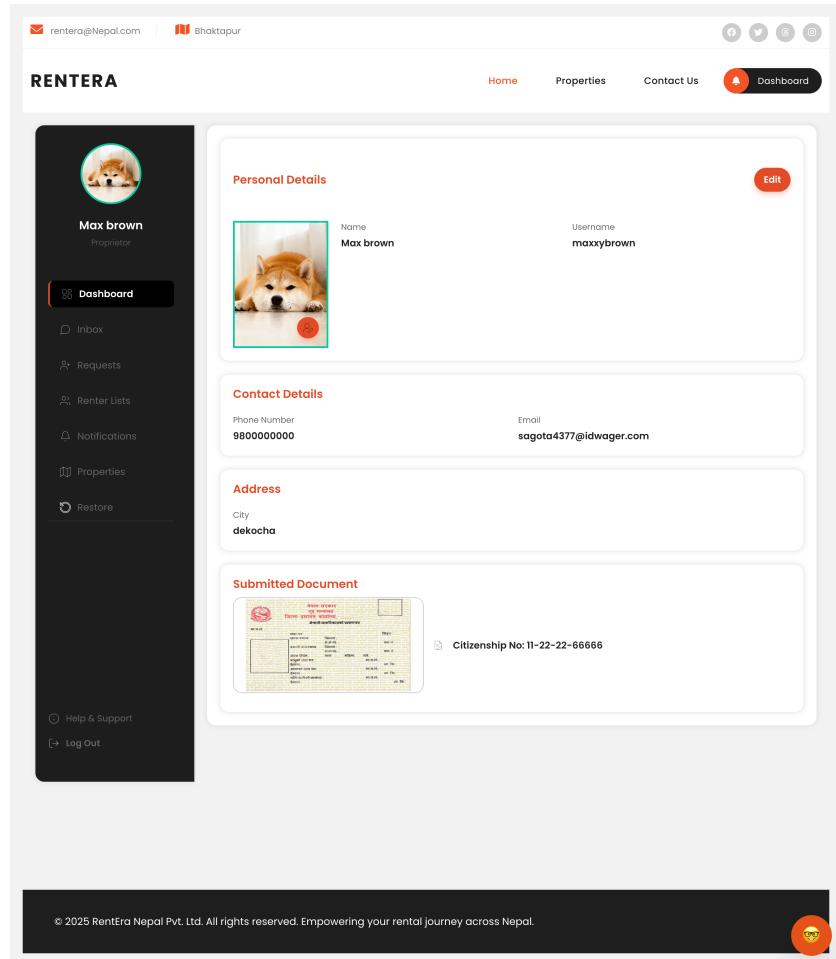


Figure 8.9: Proprietor Dashboard

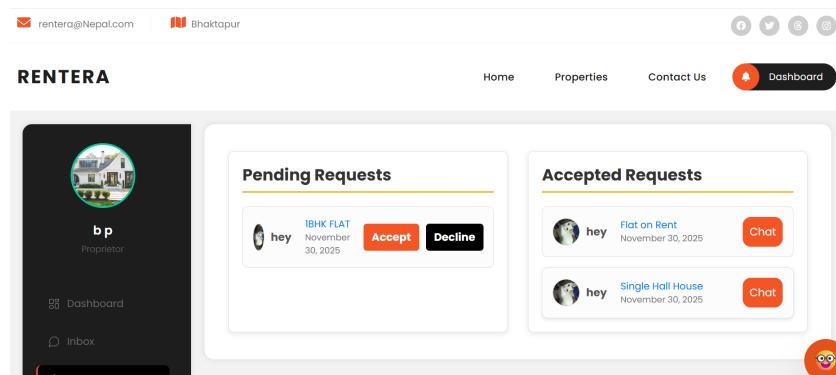


Figure 8.10: Request section(Proprietor Dashboard)

Proprietor can view the list of pending and accepted requests.

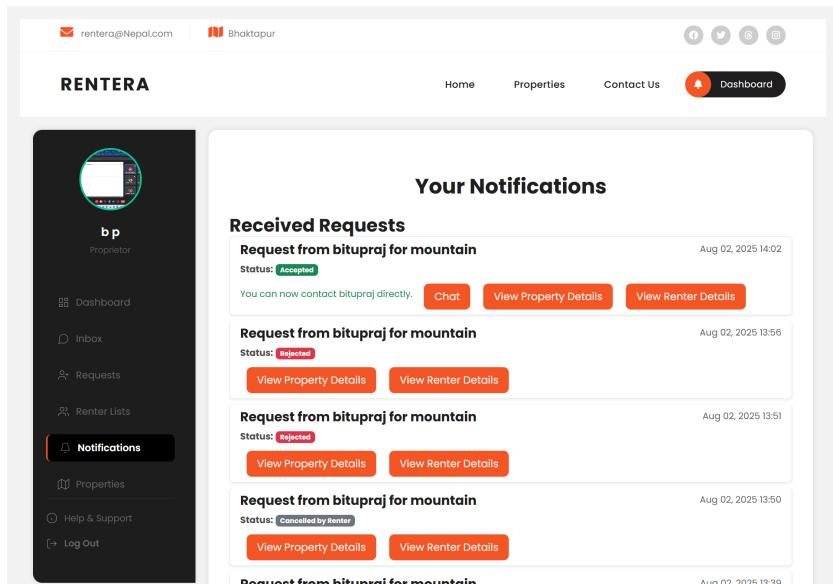


Figure 8.11: Notifications(Proprietor Dashboard)

After the Proprietor accepts the request of the renter on one of their property, the option of chatting is enabled then they can exchange messages within our system.

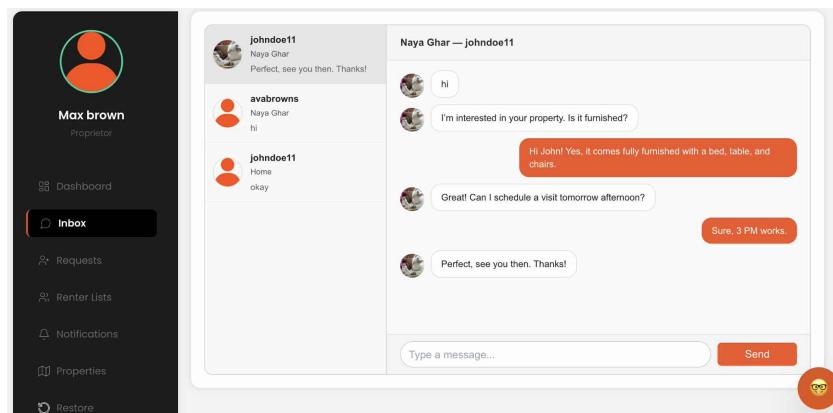


Figure 8.12: Chat between Proprietor and Renter(Proprietor Dashboard)

Proprietors can view their listed properties in the dashboard and if required they can delete them as well. They can add new properties through the dashboard. On deleting the property, it will remain on restore for certain time (15 days).

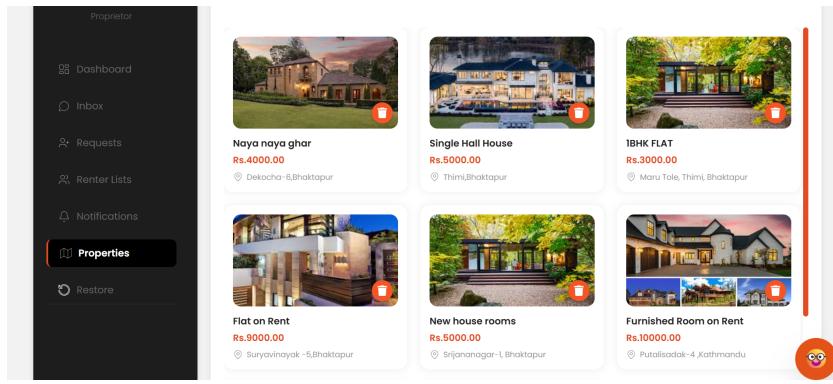


Figure 8.13: Delete properties(Proprietor Dashboard)

Proprietors can restore previously removed or inactive properties by making the required payment again. This allows them to reactivate their listings and make them visible to renters without needing to recreate the property details.

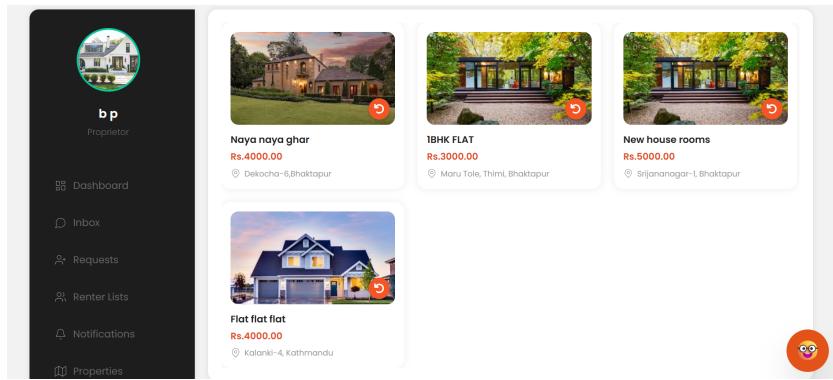


Figure 8.14: Restore properties(Proprietor Dashboard)

After three months, a property is automatically deleted from the platform. Before the expiry date, an email notification is sent to the proprietor, allowing them to restore the property by making the required payment if they wish to keep it active.

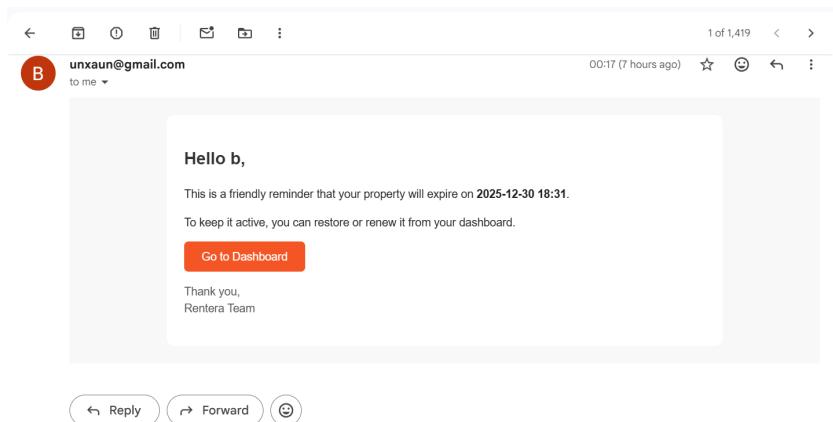


Figure 8.15: Email notification regarding property expiry and restoration

## 8.5 Adding Property

Proprietor can add their property in the website only if it is valid in the database. They should fill in the form with the correct information to verify the property.

The screenshot shows the 'Add A New Property' page on the RentEra website. The top navigation bar includes links for rentera@Nepal.com, Bhaktapur, Home, Properties, Contact Us, and Dashboard. The main form fields are as follows:

- Name\***: @BHK
- Address\***: Suryavinayak
- Municipality**: Bhaktapur
- Ward Number\***: 4
- Sheet.No**: 10
- Land.No**: 5
- Province\***: Province No. 3 (Bagmati)
- District\***: Bhaktapur
- Number of rooms**: 5
- Price\***: 100000
- Property Type\***: Flat and Apartment
- Description\***: Just recently emptied apartment, with exclusive furniture already included here with all the basic facilities
- Facilities\***:
  - Drinking Water
  - Hospital
  - Parking
- Locate on Map**: A map showing the location of the property at Suryavinayak, Bhaktapur. The map includes labels for Korean Academy, Sodhali Secondary School, Adarsh Jai Jagriti Secondary School, Balaji House, and Synergy Consulting LLC.
- Property Image\***: Choose File (No file chosen)
- Submit** button

At the bottom of the page, a footer bar contains the text: © 2025 RentEra Nepal Pvt. Ltd. All rights reserved. Empowering your rental journey across Nepal.

Figure 8.16: Add Property

## 8.6 Payment Process

After the datas are filled correctly, they are redirected to the payment page. Each room is charged Rs.100 and calculated likewise. Only after the payment, the property is fully verified and listed on the website to be rented.

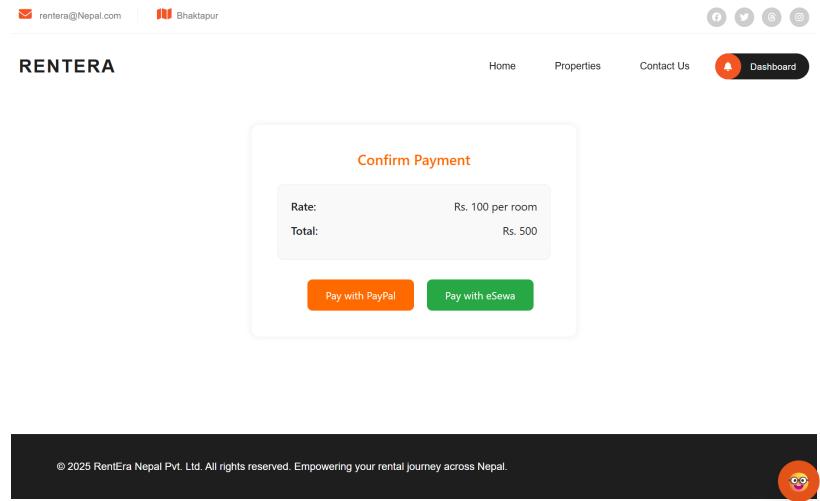


Figure 8.17: Payment confirmation

Proprietor can choose the method of payment either eSewa or Paypal.

a. **Paypal**

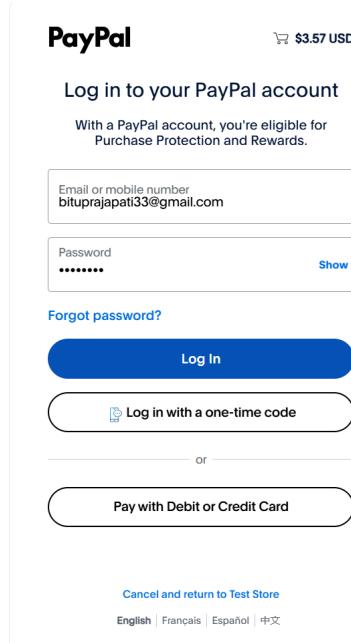


Figure 8.18: Login into PayPal

Test Store



You paid \$3.57 USD  
to Test Store

[Details](#)

**Paid with**

CREDIT UNION 1 (AK)-1231

\$3.57 USD

**Purchase details**

Receipt number: 5DS762530N5531931

We'll send confirmation to: bituprajapati33@gmail.co  
m

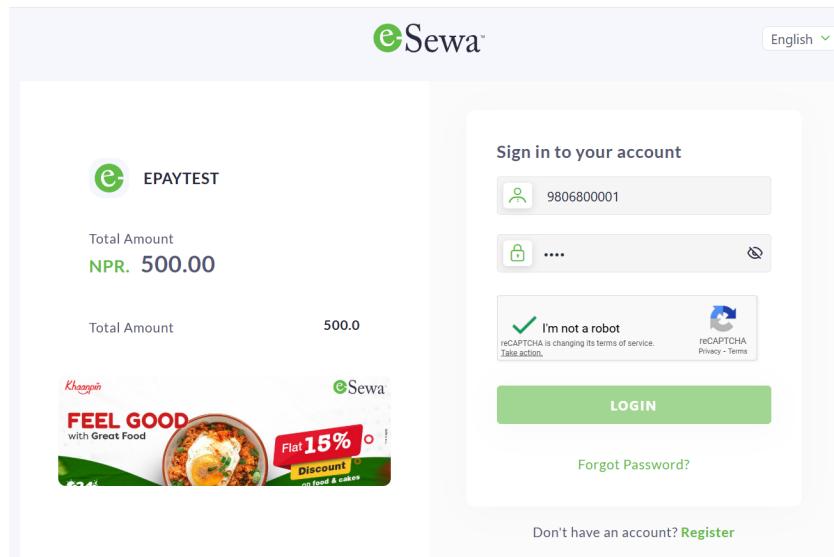
**Merchant details**

Test Store

[Return to Merchant](#)

Figure 8.19: Payment through PayPal

b. eSewa



The screenshot shows the eSewa login page. At the top right, there is a language selection dropdown set to "English". The main header "eSewa" is in green. On the left, there is a sidebar with the "EPAYTEST" logo and a "Total Amount" section showing "NPR. 500.00". Below this is a promotional banner for "Khao Phat" with the text "FEEL GOOD with Great Food" and "Flat 15% Discount on food & cakes". On the right, the main login form has fields for "User ID" (containing "9806800001") and "Password" (containing "\*\*\*\*\*"). There is a "reCAPTCHA" checkbox checked with the message "I'm not a robot". Below the login button, there are links for "Forgot Password?" and "Don't have an account? Register".

Figure 8.20: Login into eSewa

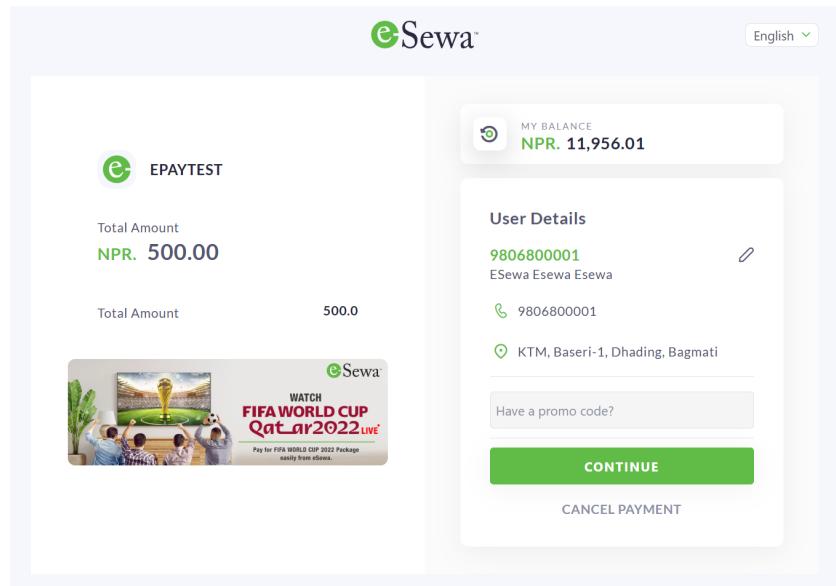


Figure 8.21: Payment through eSewa

The screenshot shows the Django Admin dashboard for the "Rentera" application. The left sidebar has a dark theme with the "admin" user icon. It lists "Authentication and Authorization", "Proprietor", and "Rentals". Under "Authentication and Authorization", there are sections for "Groups" and "Users", each with "Add" and "Change" buttons. Under "Proprietor", there are sections for "Facilities", "Land Records", and "Properties", each with "Add" and "Change" buttons. Under "Rentals", there are sections for "Citizenship Records", "Messages", "Requests", and "User profiles", each with "Add" and "Change" buttons. On the right side, there's a "Recent actions" log showing several entries from users like "k s", "111", and "Sita Sharma". The top navigation bar shows "Dashboard" and "Home > Dashboard".

Figure 8.22: Django Admin

# **Chapter 9**

## **Future Enhancements**

Following are the works to be done:

### **9.1 Review and Rating System**

Allow renters to rate properties and proprietors to improve trust and transparency.

### **9.2 Image Processing for Verification**

Allow the citizenship number and profile picture to be scanned using image processing.

### **9.3 Location and environment analysis**

Allow the user to understand and analyse the distance from main areas : market , hospitals ,colleges etc

### **9.4 Map Navigation**

Allow the user to navigate from the current location to the destination i.e to rental property.

### **9.5 Codebase Optimization for differently abled users**

Enhance both back-end and front-end code for improved performance, maintainability, and scalability while allowing differently-abled user to easily access the web application

# Bibliography

- [1] S. Anand, “Craigslist case study,” *Stuti Anand’s Design Portfolio*, 2025, this article is a fictionalized example based on search results and is not a real publication. [Online]. Available: <https://www.designbystuti.com/work/craigslist-case-study>
- [2] A. Dirie, H. Osman, O. Adan, and F. Ali, “Investigating the impact of property characteristics, cost of living, and environmental factors on rental prices in baidoa’s climate-affected real estate market: a hybrid approach using hedonic regression and neural networks,” *Frontiers in Sustainable Cities*, 2025, this article is a fictionalized example based on search results and is not a real publication. [Online]. Available: <https://www.frontiersin.org/journals/sustainable-cities/articles/10.3389/frsc.2025.1511761/full>
- [3] Wikipedia contributors. (2024, Jun.) Craigslist. [Accessed: Apr. 2, 2025]. [Online]. Available: <https://en.wikipedia.org/wiki/Craigslist>
- [4] NPROOM. Find rental rooms and flats in nepal. [Accessed: Apr. 2, 2025]. [Online]. Available: <https://www.nproom.com>
- [5] Hamrobazar. Online marketplace for rental properties in nepal. [Accessed: Apr. 2, 2025]. [Online]. Available: <https://www.hamrobazar.com>
- [6] Rooms Finder Nepal. Simplified rental property search. [Accessed: Apr. 2, 2025]. [Online]. Available: <https://www.roomsfindernepal.com>
- [7] Rental Nepal. Real estate rental and buying platform in nepal. [Accessed: Apr. 2, 2025]. [Online]. Available: <https://www.rentalnepal.com>
- [8] Gharbheti. Find rental spaces in nepal. [Accessed: Apr. 2, 2025]. [Online]. Available: <https://www.gharbheti.com>