A Project Report

On

Rento

Submitted in Partial Fulfillment of the

Requirement of Project – VII (BIT) of

Bachelor of Information Technology

Submitted To:

Purbanchal University Biratnagar,

Nepal

Submitted By:

Diwakar Shrestha (371997)

Mani Maharjan (371999)

Anit Shrestha (371988)

KANTIPUR CITY COLLEGE

Putalisadak, Kathmandu

Date: Jan. 22 2021

A project report

On

Rento

Submitted in partial fulfillment of the requirement of

Project - VII (BIT) of

Bachelor of Information Technology

Submitted to:

Purbanchal University Biratnagar,
Nepal

Submitted By

Diwakar Shrestha (371997)

ManiMaharjan (371999)

Anit Shrestha (371988)

Project Supervisor

Ashim KC

Lecturer

KANTIPUR CITY COLLEGE

Putalisadak, Kathmandu

TOPIC APPROVAL SHEET

It is hereby informed that the topic selected by Mani Maharjan, Diwakar Shrestha and Anit Shrestha of BIT VII semester for their semester project has been found suitable and as per the credit assigned by Purbanchal University (PU), Biratnagar, Nepal.

The Project Committee has approved the following topic and supervisor for the above-mentioned students.

Topic Approved: **Rento**

_ _ _ _ _ _ _ _ _ _ _

Bikash Neupane

Program Coordinator, BCA & BIT

_ _ _ _ _ _ _ _

Ashim KC

Asst. Program Coordinator, BCA & BIT

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the project entitled **"Rento"** submitted by Mani Maharjan, Diwakar Shrestha and Anit Shrestha to the Department of Information Technology, School of Science and Technology at Kantipur City College, Kathmandu, Nepal towards the requirement for BIT378CO Project VII of is an original work carried out by them under my supervision and guidance.

Signature:

_ _ _ _ _ _ _ _ _ _ _

Ashim KC

Department of Information Technology

Kantipur City College

(Project Supervisor)

Place: Kantipur City College, Putalisadak, Kathmandu.

Date:

ACKNOWLEDGEMENT

We would like to express our gratitude to the **Purbanchal University**, **IT Department** of **Kantipur City College** (KCC) College for providing us the opportunity to ensure our knowledge in the field of web development through the assignment of this project.

In spite of our effort and hard work the project would not have been successfully completed with the heartfelt assistance and guidance of our friends, seniors and teachers. We would like to thank each and every one who directly and indirectly provide the helping hands for the successful completion of this project.

Finally, we thank the **KCC IT Department** as well who deserve a good appreciation from us for providing resources to complete our project. We even thank **Mr. AshimKC, Mr. Bikash Neupane** all for sincere guidance and supervision in every possible manner despite the lack of time.

Thanks

Group Member (Symbol No :)

Mani Maharjan (371999)

Diwakar Shrestha (371997)

Anit Shrestha (371988)

ABSTRACT

Nowadays for any business to compete in the market needs an online platform or a website. This project entitled “Rento”, is a website for providing a platform to rent rooms or search available rooms which will create an online platform for house owners of different places. The idea of “Rento" is to provide an online platform for room owners and room seekers.

The Project “Rento” is designed with HTML, CSS, Python, Bootstrap as front end and MONGO, python and Django as backend. There is use of different style sheets and boot straps for completing this project. Since this project is a project without the provision of any budget, the internet is used as a prime source of collection of data and information.

LIST OF FIGURES

LIST OF TABLES

Table of Contents

CHAPTER 1: INTRODUCTION

Rento is a website for providing a platform to rent rooms or search available rooms which will create an online platform for house owners of different places. To find a room in any place a person has to travel on foot house by house and also room providers are not gaining any room seekers attention for their empty rooms, so this website is being developed with the motive of making the work of room providers and room seekers easier.

1.1 PROBLEM STATEMENT

The problem occurred before having Rento includes:

- room seeker had to visit different places on foot house by house for searching rooms, even the middle man who shows rooms charges huge amount before showing rooms
- room owners were not being able to gain attention from any room seekers and rooms of house are left empty without making any incomes

1.2 OBJECTIVE OF PROJECT

- To create platform to rent room for room owners and find room for room seekers

1.3 FEATURES OF PROJECT

- Room owner gets to add rooms detail and manage rooms setting either to make private (if room is not empty) or public (if room is empty)
- Rooms seekers can search for rooms in the location they wish by using filter bar
- Room owners also are able to maintain their rent detail sheets to manage their rent
- Room seeker can give feedback for rooms and rooms detail provided by room owner
- Room owner can feature their rooms

1.4 ASSIGNMENT OF ROLES AND RESPONSIBILITIES

Member Name	Symbol No.	Role Assigned
Mani Maharajan	371999	Requirement Analysis, System design, Coding, Testing, documentation
Diwakar Shrestha	371997	Requirement Analysis, System Design, Coding, Testing, documentation
Anit Shrestha	371988	Requirement Analysis, System Design, Coding, Testing, documentation

1.5ORGANIZATION OF DOCUMENT

This documentation is categorized into 5 various chapters and each chapter contains various sub-chapter.

Chapter	Heading	Contents
Chapter 1	Introductions	1.1 Introduction 1.2 Problem Statement 1.3 Objectives 1.4 Features 1.5 Organization of document
Chapter 2	Literature Review	2.1 Existing System 2.2 Proposed system
Chapter 3	System Analysis	3.1 Interface Requirements 3.2 Functional Requirement 3.3 Non-Functional Requirement 3.4 Feasibility Study
Chapter 4	System Design	4.1 System Architecture 4.2 Context and Data flow diagram

		4.3 Use-Case diagram 4.4 Sequence diagram 4.5 Database design
Chapter 5	System development and implementation	5.1 Programming platform and implementation, Tools and technologies 5.2 Operating Environment 5.3 Testing and debugging 5.3 Implementation and Result Analysis
Chapter 6	Conclusion and Future Enhancement	6.1 Conclusion 6.2 Limitation 6.3 Future Enhancements

CHAPTER 2: EXISTING SYSTEM OVERVIEW

2.1 Existing System

2.1.1 Rental Nepal (www.rentalnepal.com)

Rental Nepal is real estate system which deals with the rental services such as buying and selling houses, office spaces, apartments, shutters and shops, rooms, and lands. User who register in this website has access to add the properties of different types with properties details, images and google map location and user who is seeking for room gets access to enquiry form for room they want.

PROS

- includes all types of properties to add or search and also has service to get featured
- provides features like video links, photos, sharing in social media, comment and more

CONS

- not so user focused and user are not given ability to search by location filters

2.1.2 Gharbheti (www.gharbheti.com)

Gharbeti is another real estate system similar to Rental Nepal here user can register as a investor to the properties or properties owner who wants to post their properties and the room seekers can find room, book, visit and move-in if they like.

PROS

- includes all types of properties and also has feature to advertise
- room seeker can book and get field visit

CONS

- website has functional issues
- owner cannot have direct contact with room seeker which may cause problem to visit the site

2.2 Proposed System

Rento is a real estate system which deals with the room renting services anddeveloped to facilitate the room owner and room seeker. User can either be room owner and room seeker,if usersareroom

owner then they can add the properties detail including videos links and photos, and if they are room seeker then they can enquire directly with room owner. User to post properties on must create account and login. They have also ability to manage properties and also set room to public (if room is available for rent) and private (if room is full), if they don't want to remove room that is pack. Also, if owner want to manage the account of their room rentals system provides them a default rent management system of website. If user is room seeker then they can also enquire in properties without having account by providing details and they has also ability to provide rating and report for fake room details.

CHAPTER 3: SYSTEM ANALYSIS

Before starting software development process, system analysis is important part as we need to be clear which system development model is to be followed to develop software then the requirement specification is to be collected following the steps of system development model, so this part gives all the information about system development model used, requirements analysis and specification.

3.1 SYSTEM DEVELOPMENT MODEL

As this is an academic project, we do not have proper requirement specification and the requirement might be changing and the project might have to be redesigned so we used prototyping model. Figure is prototyping model which is used when actual project is being developed.

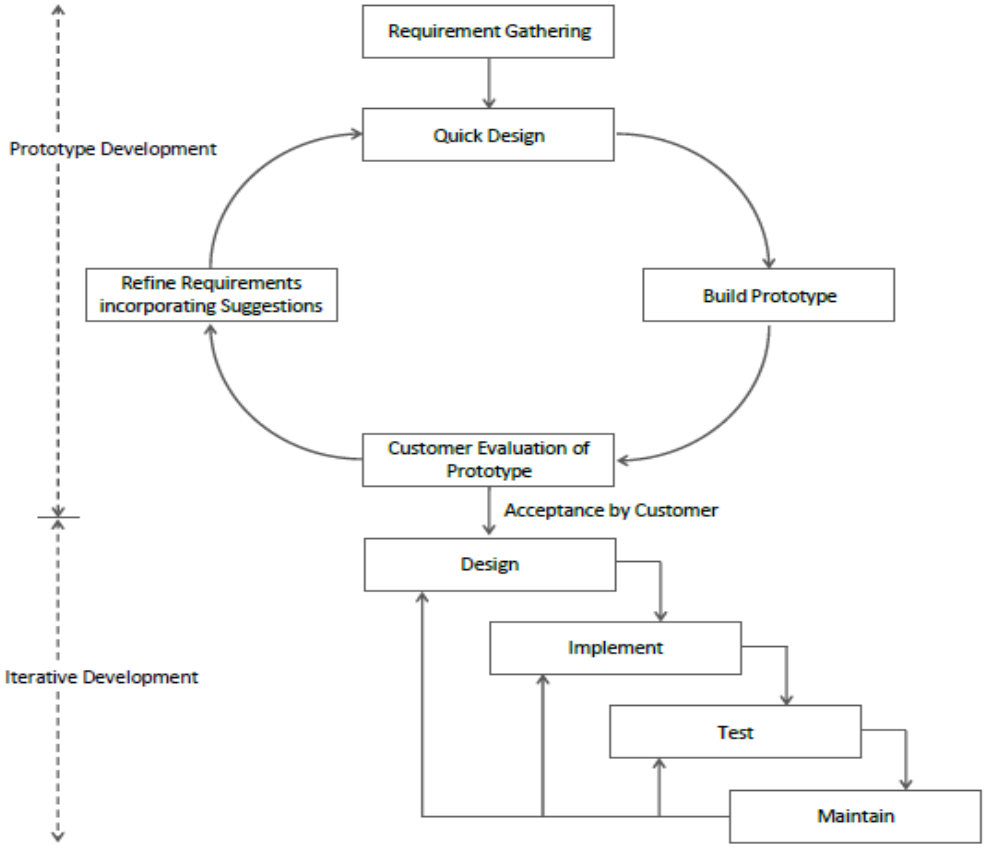


Fig:- System Development Model

3.2 REQUIREMENT SPECIFICATION

3.2.1 Functional Requirements

Requirement Name	Requirement Statement	Must/Want
Add room	New member shall be able to sign up and add their rooms	must
Show rooms	Website shall have the page “Rooms” to show the rooms added by users	must
Search room	Website should have search filter to search the room on required location	must
Enquire	User shall be able to enquire on the room detail page	must
Manage room	User who is registered shall be able to manage the properties added	must
Manage rent account	Website should provide a rent management system to registered user	must

3.2.2 Non-Functional Requirements

Requirement Name	Requirement Statement	Must/Want
Compatibility	The website shall support all browsers and also should be responsive on smart mobile devices	must
Maintainability	Website shall be able to be modified to adapt the changing environment	must
Usability	User shall be provided a guide to use the system	must
Security	User shall be able to hide or show their detail and their data must be protected	must

3.3 FEASIBILITY STUDY

As this is the academic project the feasibility study is performed being based on the academic requirements and we followed following feasibility studies for the projects.

3.3.1 Technical Feasibility

As to develop any projects hardware and software's are the basic things that we need as we are the bit students all three of our team member have laptops with windows operating system so there is no problem for develop of the software. We willbe able to install all the development tools required in our machine so this project is technically feasible.

3.3.2 Economic Feasibility

As this is our academic project we have our college to support us on any cost if required but the tools and technology we will be using for this project are all free of cost and the effort we provide in the project is evaluated by our academic score so we think this project is economically feasible

CHAPTER 4: SYSTEM DESIGN

4.1 CONTEXT DIAGRAM AND DATA FLOW DIAGRAM

4.1.1 CONTEXT LEVEL DFD

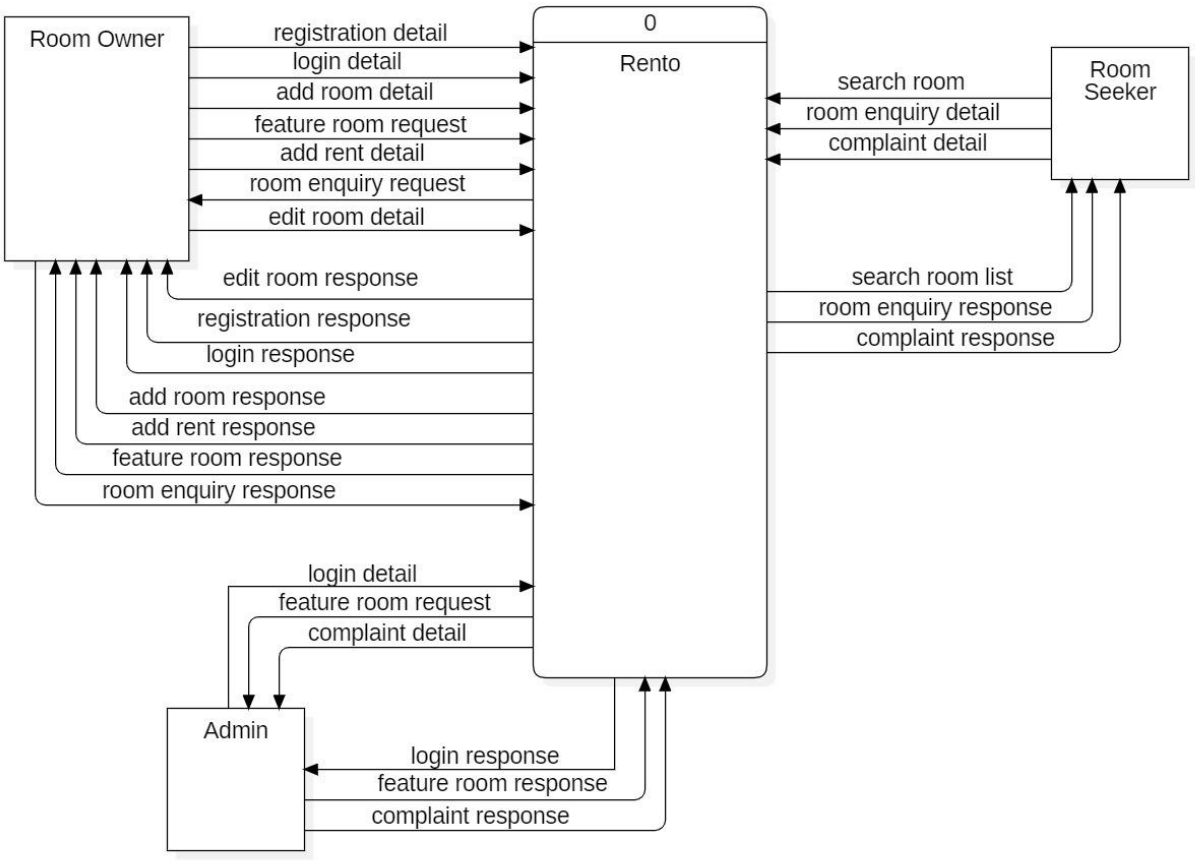


Fig:- Context Diagram

4.1.2 DFD LEVEL 0

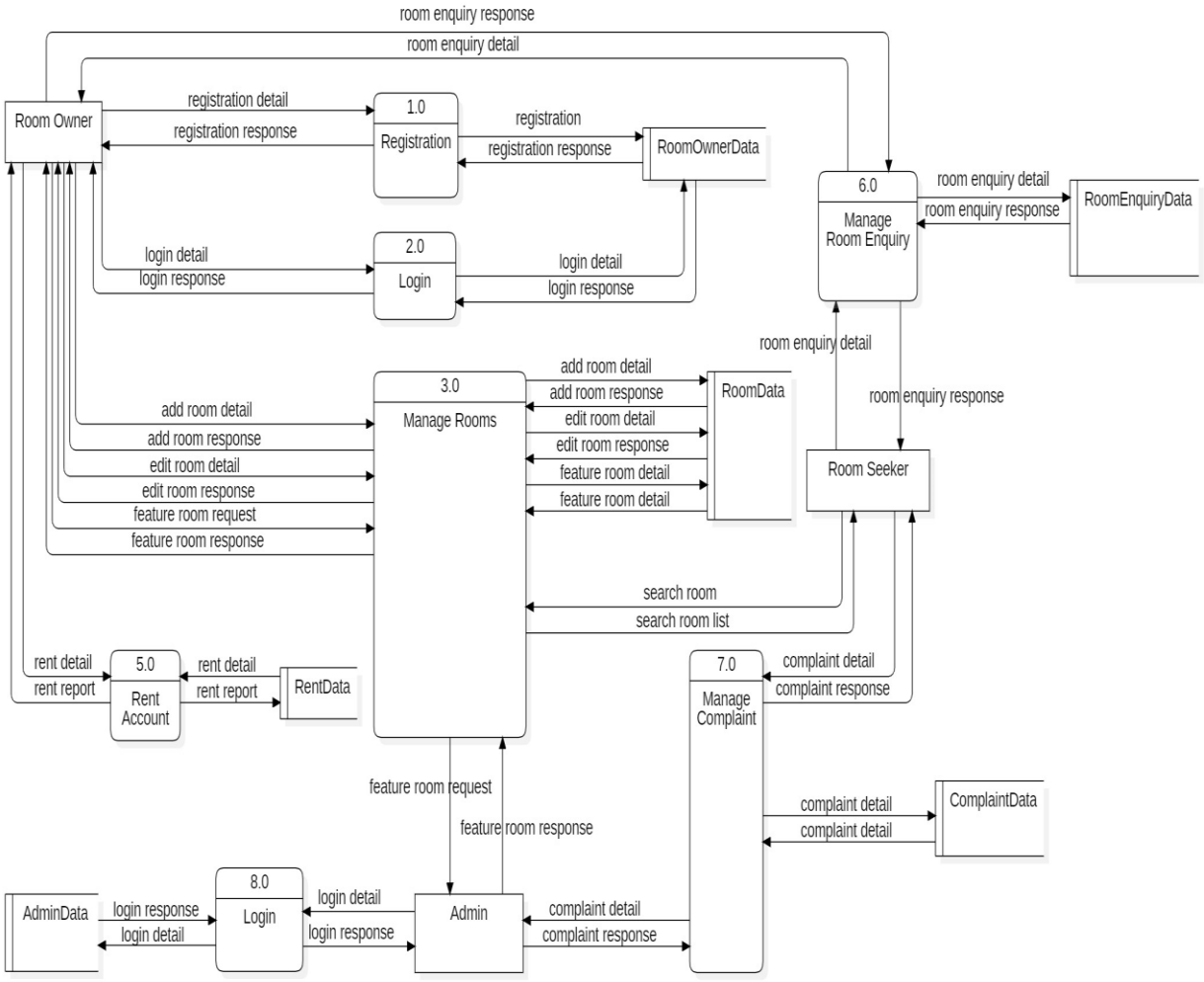


Fig:- Level 0 DFD

4.1.3 DFD LEVEL 1 (3.0)

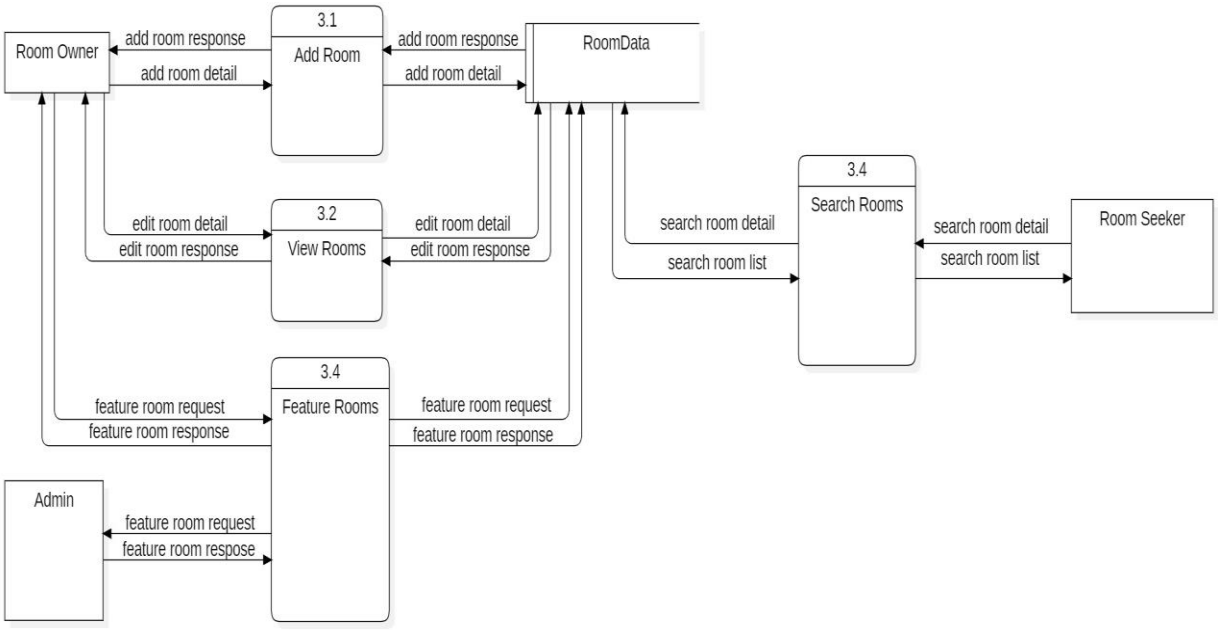


Fig:- Level 1 DFD

4.2 Entity Relationship Diagram (ER-Diagram)

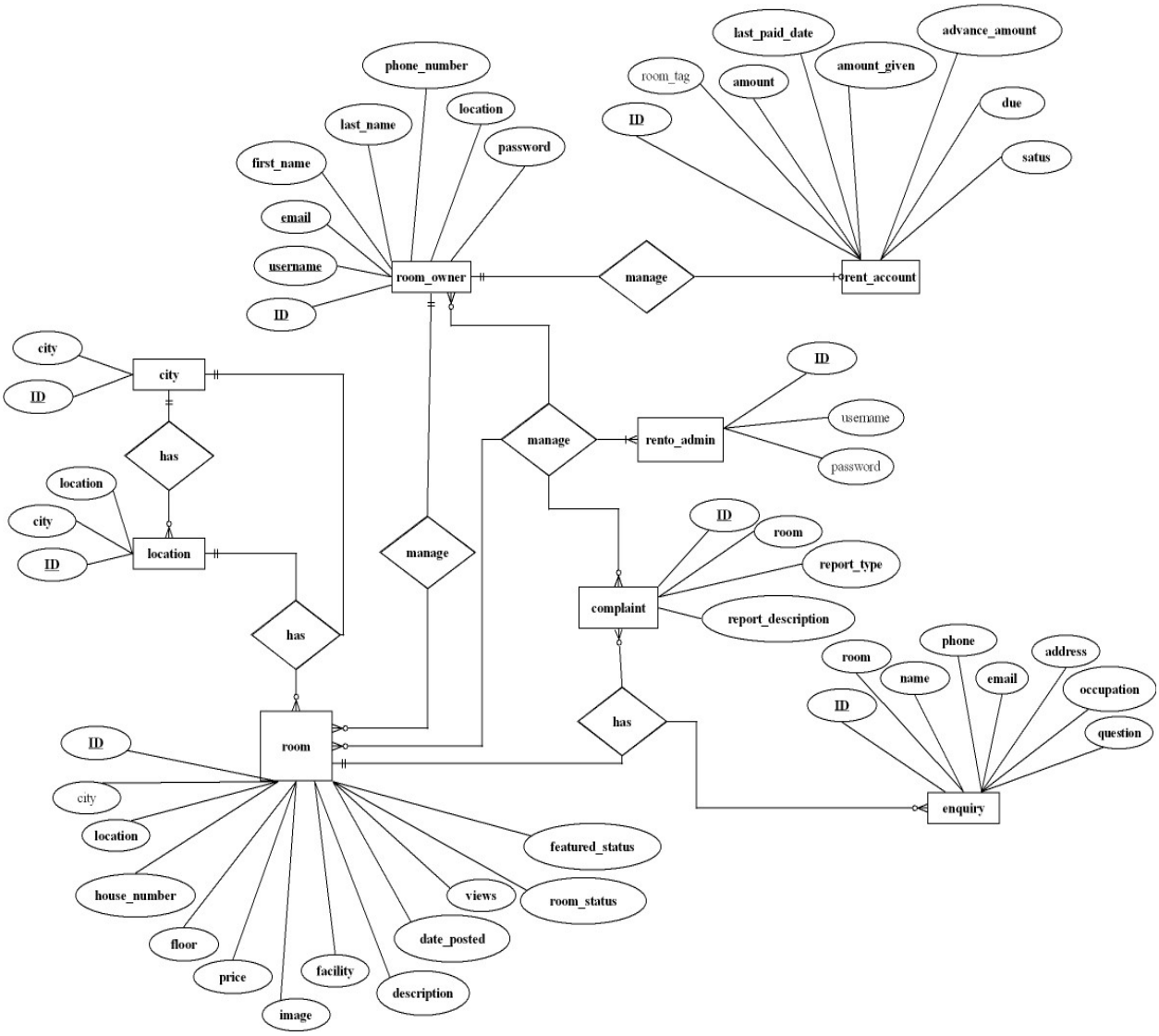


Fig :- ER Diagram

4.3 USE CASE DIAGRAM

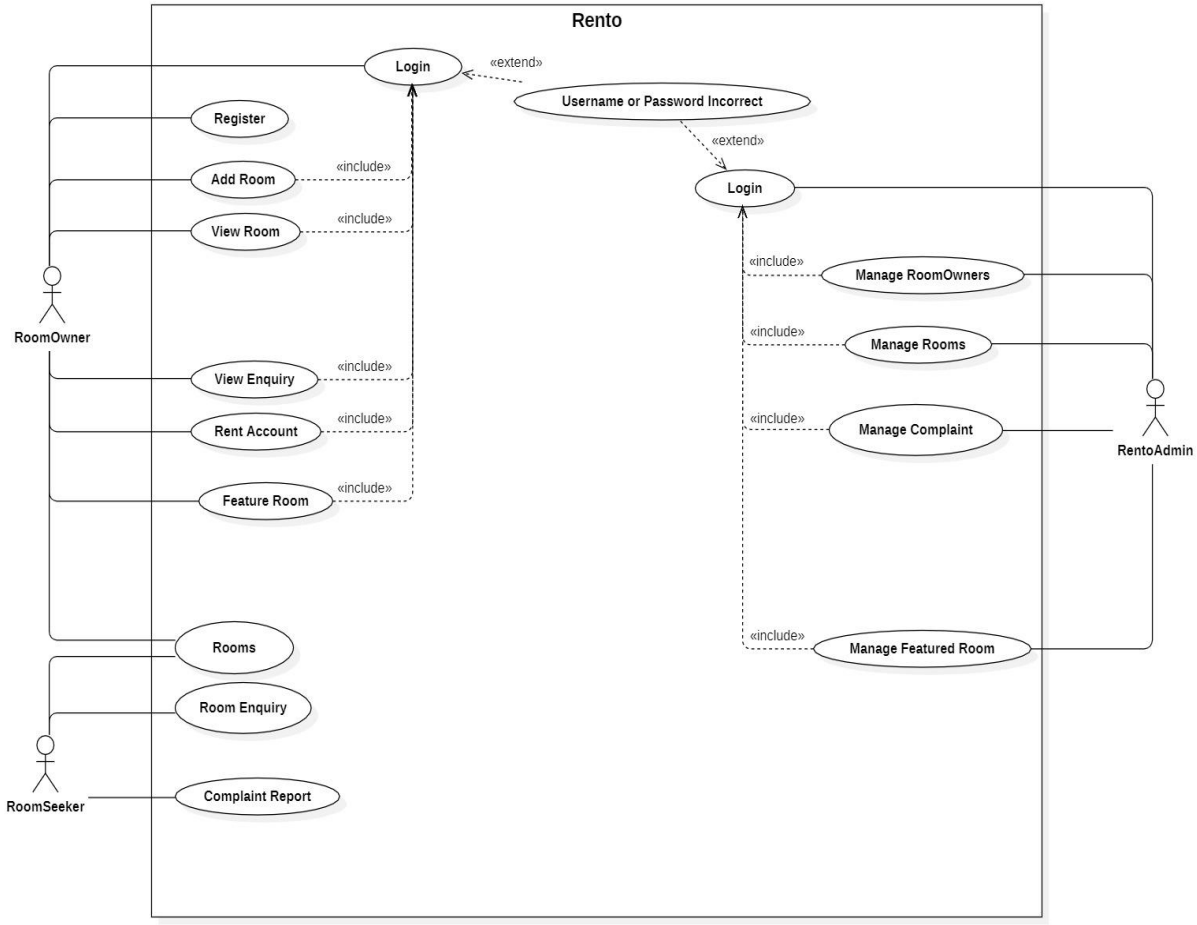


Fig :- Use Case Diagram

CHAPTER 5: SYSTEM DEVELOPMENT AND IMPLEMENTATION

5.1 Programming platform and implementation (Tools and technologies)

5.1.1 Frontend Tools:

- HTML, CSS, BOOTSTRAP
- PYTHON, DJANGO

5.1.2 Backend Tools:

- PYTHON, DJANGO
- SQLITE

5.1.3 Development tools:

- VISUAL STUDIO CODE

5.2 Operating Environment

Application was tested and developed in windows 8.1/10 with 4/8GB RAM and i5/i7 intel processor

5.3 Testing and Debugging

Test case	1
Test objective	To check whether Sign Up is successful
Test data	Entering required data for sign up
Expected result	User should be registered into the database
Test result	User registered to the database
Conclusion	Expected result matches actual result

Test case	2
Test objective	To check whether login is successful
Test data	Entering username and password
Expected result	User should be logged into the system
Test result	User is logged in to the system
Conclusion	Expected result matches actual result

Test case	3
Test objective	To check whether rooms is added or not
Test data	Entering required room data
Expected result	Rooms should be added to the database
Test result	Rooms added to the database successfully
Conclusion	Expected result matches actual result

Test case	4
Test objective	To check weather rooms are displayed or not
Test data	None
Expected result	Rooms from database should be displayed on the rooms tab
Test result	All the public and non-blocked rooms are successfully displayed
Conclusion	Expected result matches actual result

Test case	5
Test objective	To delete room
Test data	User should delete a room
Expected result	Selected room should be removed from the database
Test result	Selected room was successfully removed from the database
Conclusion	Expected result matches actual result

Test case	6
Test objective	To list a room as featured room
Test data	User should request for feature room and admin should approve it
Expected result	Rooms selected to be featured by the admin should be displayed in homepage
Test result	Selected room was successfully updated as featured room in the database and displayed in home screen
Conclusion	Expected result matches actual result

Test case	7
Test objective	To check whether the admin can block/delete user/room
Test data	Admin should select a user/room to be blocked or deleted
Expected result	Selected user/room should be removed from database if deleted or flagged as blocked if blocked
Test result	Selected user/room was successfully removed from database if deleted or was flagged as blocked if blocked
Conclusion	Expected result matches actual result

Test case	8
Test objective	To check if all rooms are in rent account and if user is able input rent information
Test data	User should select rent tab to create rent account and enter rent information
Expected result	Rent account should be created for user and rent information should be updated in database
Test result	Rent account was created successfully for user and rent information was updated successfully
Conclusion	Expected result matches actual result

CHAPTER 6: CONCLUSION AND FUTURE ENHANCEMENT

6.1 Conclusion

This project is a web-based application for room seekers to find a suitable room for themselves and room owners to advertise their availed room space to room seekers. This project will help owners to maintain and update their rooms posted and set them to private or public depending if the rooms are given in rent or not. This app also helps admin to monitor and maintain their users and rooms and allows them to delete any misleading or ill advanced rooms and users.

6.2 Limitations

Despite the hard works and efforts, this system still lacks some simple yet good features, such as:

- Room seeker are not able to book the room instead they can only enquire and wait for room owner's response
- Room seekers cannot see the exact location of room as there is not any pinned map location

6.3 Future Enhancement

Application can be updated in the future by adding following features

- Direct payment from the website itself
- Full chat system for the owner and room seeker
- Live notification system for the room owner

References

AKJASIM. (n.d.). *cb_dj_dependent_dropdown*. Retrieved 2021, from GitHub: https://github.com/akjasim/cb_dj_dependent_dropdown

DFSQ. (n.d.). *Confirm deletion in modal*. Retrieved 2021, from stackoverflow: https://stackoverflow.com/questions/8982295/confirm-deletion-in-modal-dialog-using-twitter-bootstrap

Freitas, V. (2021). *How to Paginate with Django*. Retrieved from https://simpleisbetterthancomplex.com/tutorial/2016/08/03/how-to-paginate-with-django.html

Gale, V. (n.d.). *Get user group in a template*. Retrieved 2021, from stackoverflow: https://stackoverflow.com/questions/1052531/get-user-group-in-a-template/17087532

Jobelle. (n.d.). *How get the src value from the img clicked*. Retrieved 2021, from stackoverflow: https://stackoverflow.com/questions/34992165/how-get-the-src-value-from-the-img-clicked?answertab=oldest#tab-top

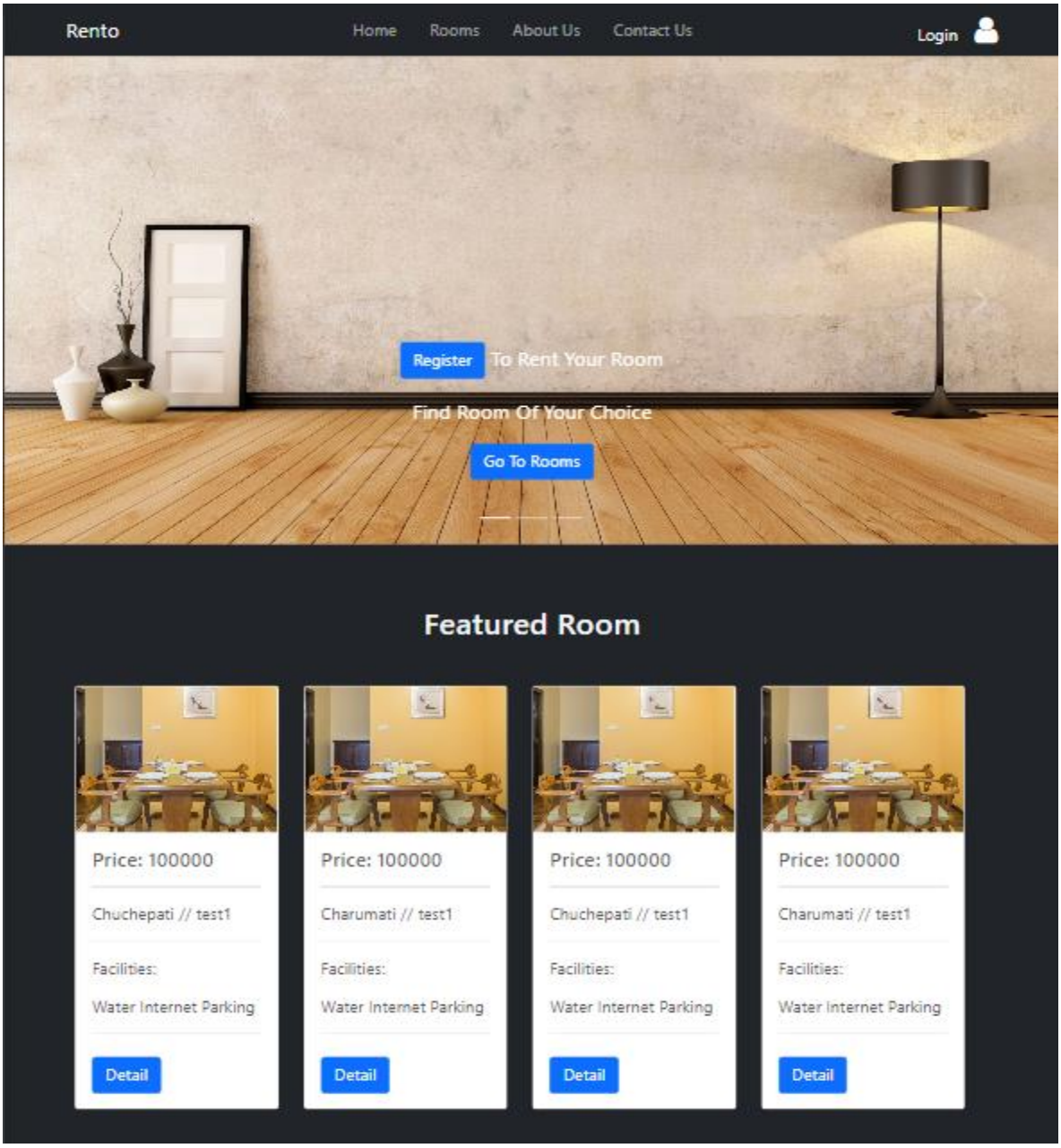
Lemayzeur. (n.d.). *pass value to bootstrap modal form with django*. Retrieved 2021, from stacoverflow: https://stackoverflow.com/questions/50196148/pass-value-to-bootstrap-modal-form-with-django

MAXG203. (n.d.). Retrieved 2021, from GITHUB: https://github.com/maxg203/Django-Tutorials/blob/master/home/views.py

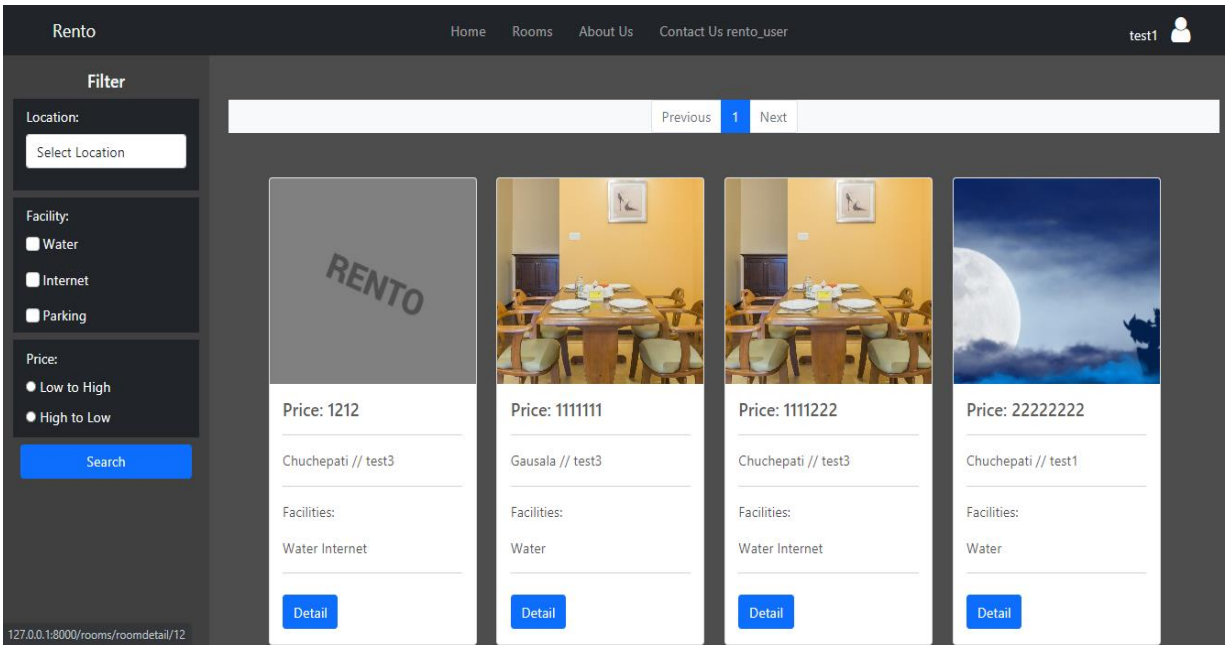
Olexiy, Z. (n.d.). *How to calculate diff between two dates in django*. Retrieved 2021, from stackoverflow: https://stackoverflow.com/questions/41229963/how-to-calculate-diff-between-two-dates-in-django

Appendixes

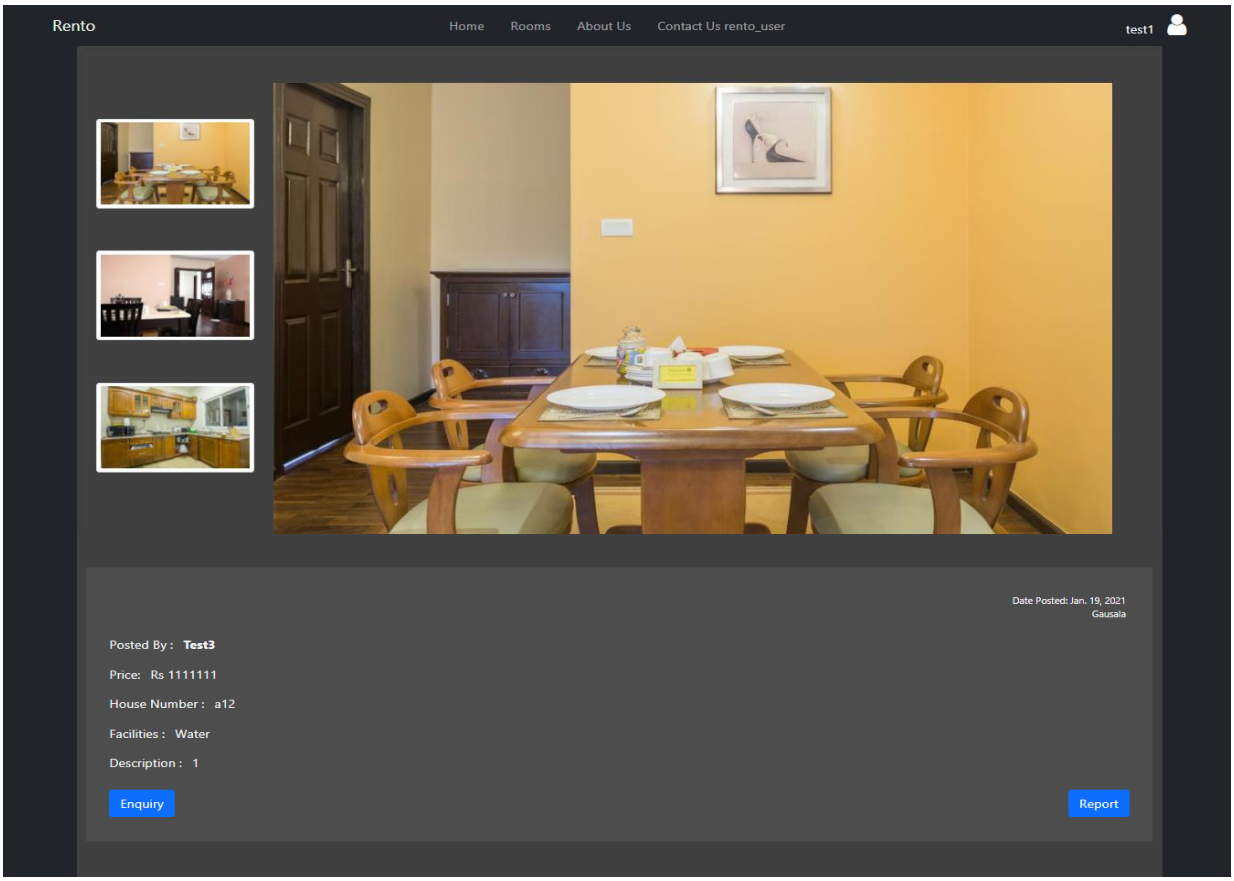
Apendix1: Screenshots



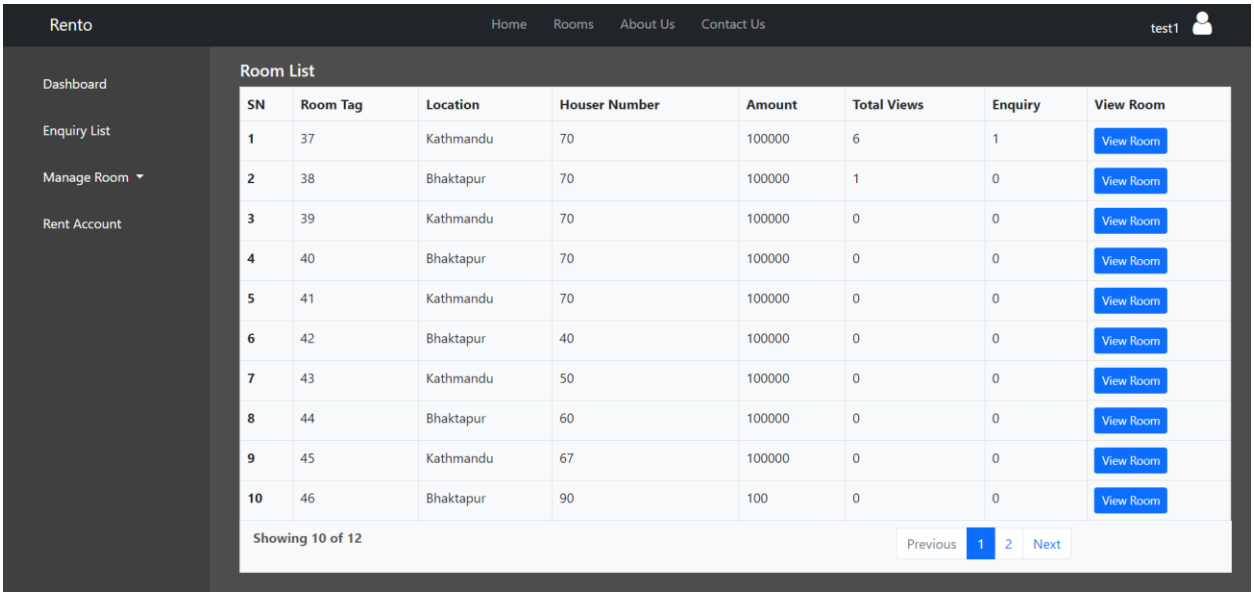
Home screen



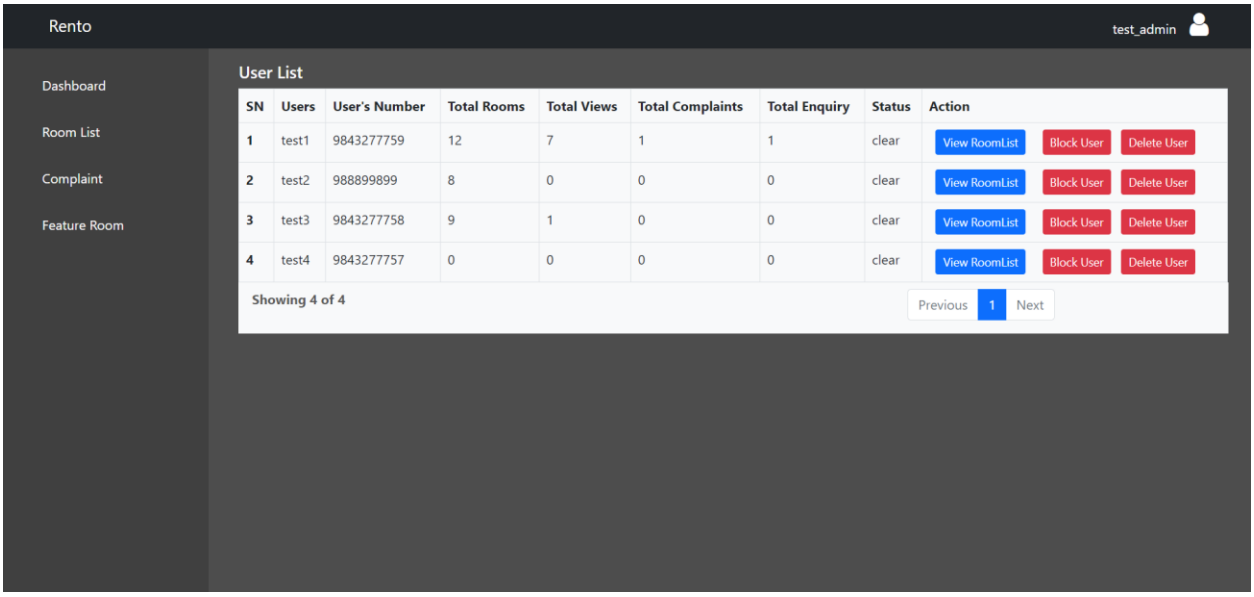
Room page



Room detail page



User dashboard



Admin dashboard

Apendix2: Gantt Chart