ROOMATE FINDER

A MINI-PROJECT REPORT

Submitted by

KISHORE K 220701134

In partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

RAJALAKSHMI ENGINEERING COLLEGE

AUTONOMOUS, CHENNAI

NOV/DEC, 2024

BONAFIDE CERTIFICATE

Certified that this mini project **"Roomie Match"** is the bonafide work of **" KISHORE K (2116220701134)"** who carried out the project work under my supervision.

SIGNATURE

Mrs. JANANEE V,
Assistant Professor,
Computer Science and Engineering,
Rajalakshmi Engineering College,
Thandalam, Chennai - 602105

Submitted for the End Semester Practical examination to be held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I express my sincere thanks to my beloved and honorable chairman **Mr. S. MEGANATHAN** and the chairperson **Dr. M. THANGAM MEGANATHAN** for their timely support and encouragement.

I am greatly indebted to my respected and honourable principal **Dr. S.N. MURUGESAN** for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by my head of the department, **Dr. P. KUMAR,** and my Academic Head **Dr.R.SABITHA,** for being every supporting force during my project work.

I also extend my sincere and hearty thanks to my internal guide **Mrs. JANANEE V** for her valuable guidance and motivation during the completion of this project.

My sincere thanks to my family members, friends and other staff members of Computer Science and Engineering.

ABSTRACT

The **Roommate Finder** website is a web-based application designed to help individuals in Chennai find suitable roommates to share rental spaces. The platform addresses the common challenges of finding compatible roommates by allowing users to post their requirements and preferences, such as location, budget, lifestyle, and personal preferences. Users can search through existing roommate listings or post their own to find a match.

The website offers an intuitive user interface where users can easily browse listings, communicate with potential roommates, and make informed decisions. By facilitating this process, the platform aims to create a convenient and efficient way to connect people with shared housing needs. The system ensures privacy by allowing users to filter out irrelevant matches and control their contact information.

The project is built using modern web technologies, including HTML, CSS, JavaScript, and PHP for the frontend and backend, with a MySQL database to manage user data and listings. It aims to solve the problem of high rent costs by encouraging shared living, while also fostering a community-driven environment for safe and reliable roommate searching.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1INTRODUCTION

The rising cost of living, particularly in metropolitan areas like Chennai, has made shared housing an increasingly popular solution for many individuals, including students, working professionals, and expatriates. Finding a suitable roommate, however, can often be a challenging and time-consuming task. People seek compatibility in terms of lifestyle, location, budget, and personal preferences when choosing a roommate, and this is where a dedicated platform can be highly beneficial.

The **Roommate Finder** website is designed to simplify the process of finding compatible roommates in Chennai. It provides a user-friendly platform where users can either search for a roommate or post their own room or shared living space requirements. The system allows users to specify preferences such as rental budget, preferred location, and other important factors like cleanliness, habits, or work schedules. This reduces the effort and time typically required in the process of finding a roommate through conventional methods.

The platform is built with a focus on ease of use, security, and efficiency. By connecting individuals with similar housing needs, it not only helps in reducing rent expenses but also fosters a more harmonious living environment. This project leverages modern web technologies and provides an interactive

interface, making it convenient for users to find suitable roommates in a few simple steps.

1.2 SCOPE OF WORK

The **Roommate Finder** website is designed to streamline the process of finding compatible roommates in Chennai by providing a platform where users can search for or post available room listings based on preferences like budget, location, and lifestyle. The platform allows users to register, create profiles, and browse listings while filtering based on key criteria. It features a built-in messaging system for users to communicate directly, helping them evaluate compatibility before making commitments. With a focus on user experience, the site offers a responsive design for accessibility on various devices. While primarily focused on Chennai, the framework can be expanded to other locations, and future enhancements may include a rating system and map integration.

1.3 AIM AND OBJECTIVES OF THE PROJECT

The aim of the **Roommate Finder** website is to create a user-friendly platform that helps individuals in Chennai find compatible roommates by allowing users to search for or post listings based on preferences such as budget, location, and lifestyle. The platform's key objectives are to simplify the roommate search process, enable users to create profiles and listings, and provide a secure messaging system for communication. It also focuses on offering a mobile-friendly experience, ensuring privacy, and allowing for future expansion to other regions.

OBJECTIVES:

- **Simplify Roommate Search**: Provide users with an easy-to-use platform to search for compatible roommates based on filters like budget, location, and personal preferences.

- **Enable User Profiles and Listings**: Allow users to create profiles and post room listings with detailed information, such as rent, location, and roommate preferences.

- **Facilitate Secure Communication**: Integrate a messaging system that allows users to communicate directly and securely with potential roommates.

- **Ensure Responsive Design**: Build a mobile-friendly and responsive interface that ensures users can access the platform easily on any device.

- **Protect User Privacy**: Implement basic security measures to protect users' personal information and ensure safe browsing.

- **Support Future Expansion**: Design the platform to be scalable, allowing it to expand to other cities or regions in the future.

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATIONS

To run and maintain the **Roommate Finder** website efficiently, the following hardware specifications are recommended:

1. **Server Requirements**:
    - **Processor**: Intel Xeon or equivalent with at least 2.0 GHz or higher for efficient server performance.
    - **RAM**: Minimum 4 GB (8 GB or higher recommended for better performance under heavy traffic).
    - **Storage**: Minimum 100 GB SSD for fast read/write operations and to store user data, images, and listings.
    - **Network**: High-speed internet connection (1 Gbps or higher recommended) to handle user traffic smoothly.
    - **Operating System**: Linux (Ubuntu, CentOS) or Windows Server (based on preference).
2. **Client-Side Requirements** (for users):
    - **Device**: Desktop, laptop, tablet, or smartphone with basic internet access.
    - **Browser**: Modern browsers like Google Chrome, Firefox, Microsoft Edge, or Safari, supporting HTML5 and CSS3.
    - **Internet Speed**: Minimum of 1 Mbps internet speed for optimal performance.

These specifications ensure that both the server and client sides of the platform can handle smooth performance, scalability, and a good user experience.

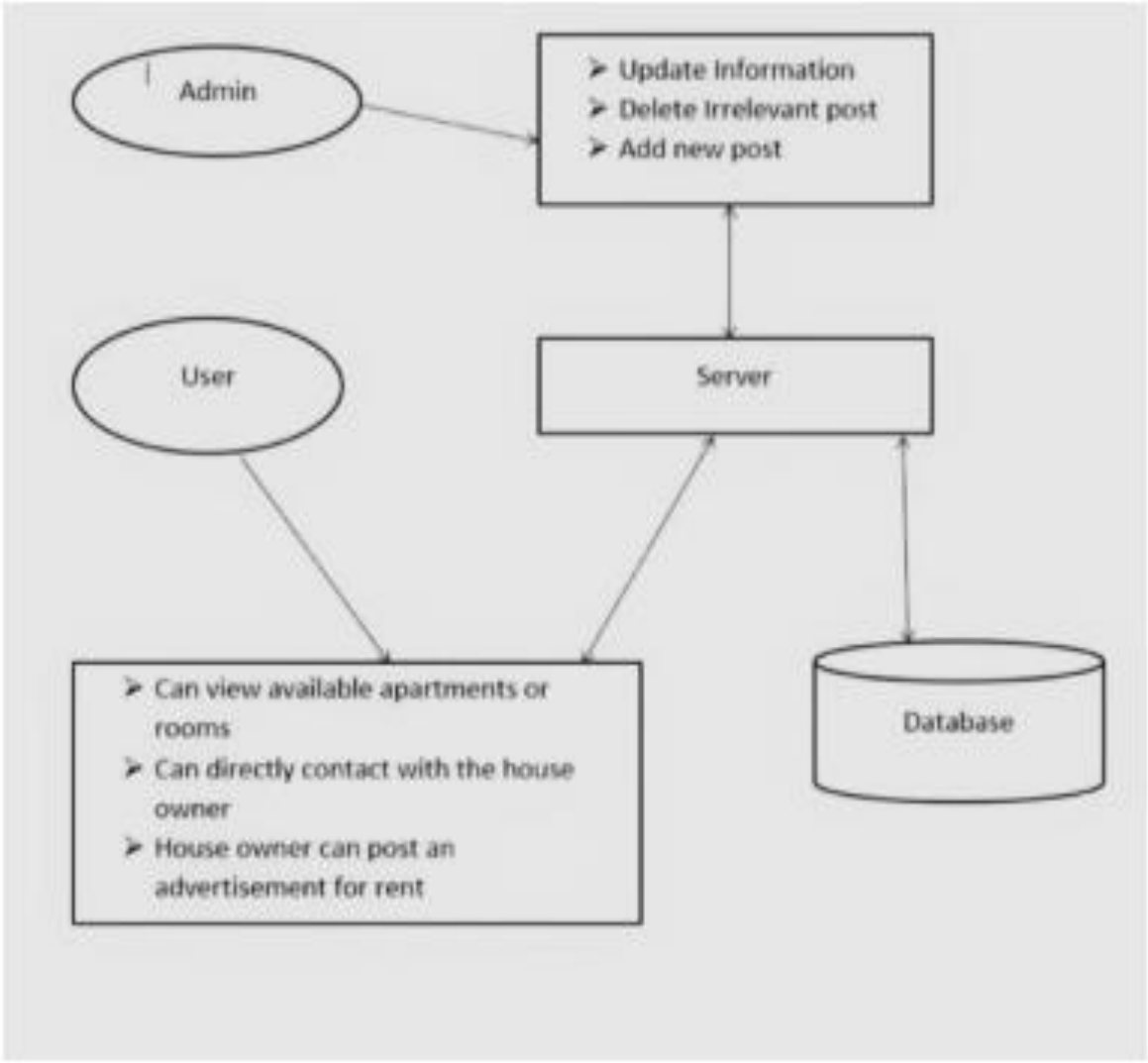
2.2 SOFTWARE SPECIFICATIONS

The project relies on a set of modern software tools and technologies to build, test, and deploy the system:

- **Front-End:** HTML, CSS, Bootstrap,Jquery and JavaScript
- **Framework:** PHP
- **Backend:** XAMPP,mysql

CHAPTER 3

ARCHITECTURE DIAGRAM



CHAPTER 4

MODULE DESCRIPTION

4.1Landing Page Module

The **Landing Page Module** serves as the first point of interaction for users visiting the **Roommate Finder** website. It is designed to create an engaging and informative experience, featuring a prominent header with navigation links to essential pages, a hero section with a catchy tagline and call-to-action buttons, and an overview of key features. This module aims to draw users in and facilitate their journey towards finding compatible roommates by providing easy access to the platform's functionalities.

4.2 User Registration and Authentication Module

The **User Registration and Authentication Module** allows new users to create accounts and existing users to log in securely. This module collects essential information such as usernames, passwords, and contact details during registration and verifies user credentials during login. Additionally, it implements password encryption and authentication protocols to ensure user data privacy and security. By streamlining the registration and login processes, this module enhances user experience and encourages active participation on the platform.

4.3 Roommate Search Module

The **Roommate Search Module** enables users to search for compatible roommates based on their specific criteria, such as location, budget, and lifestyle preferences. Users can input their requirements into a search form, which retrieves relevant listings from the database. This module may include filtering options to narrow down results and sorting features to organize listings by various parameters. By providing a straightforward and efficient way to find potential roommates, this module enhances user engagement and satisfaction.

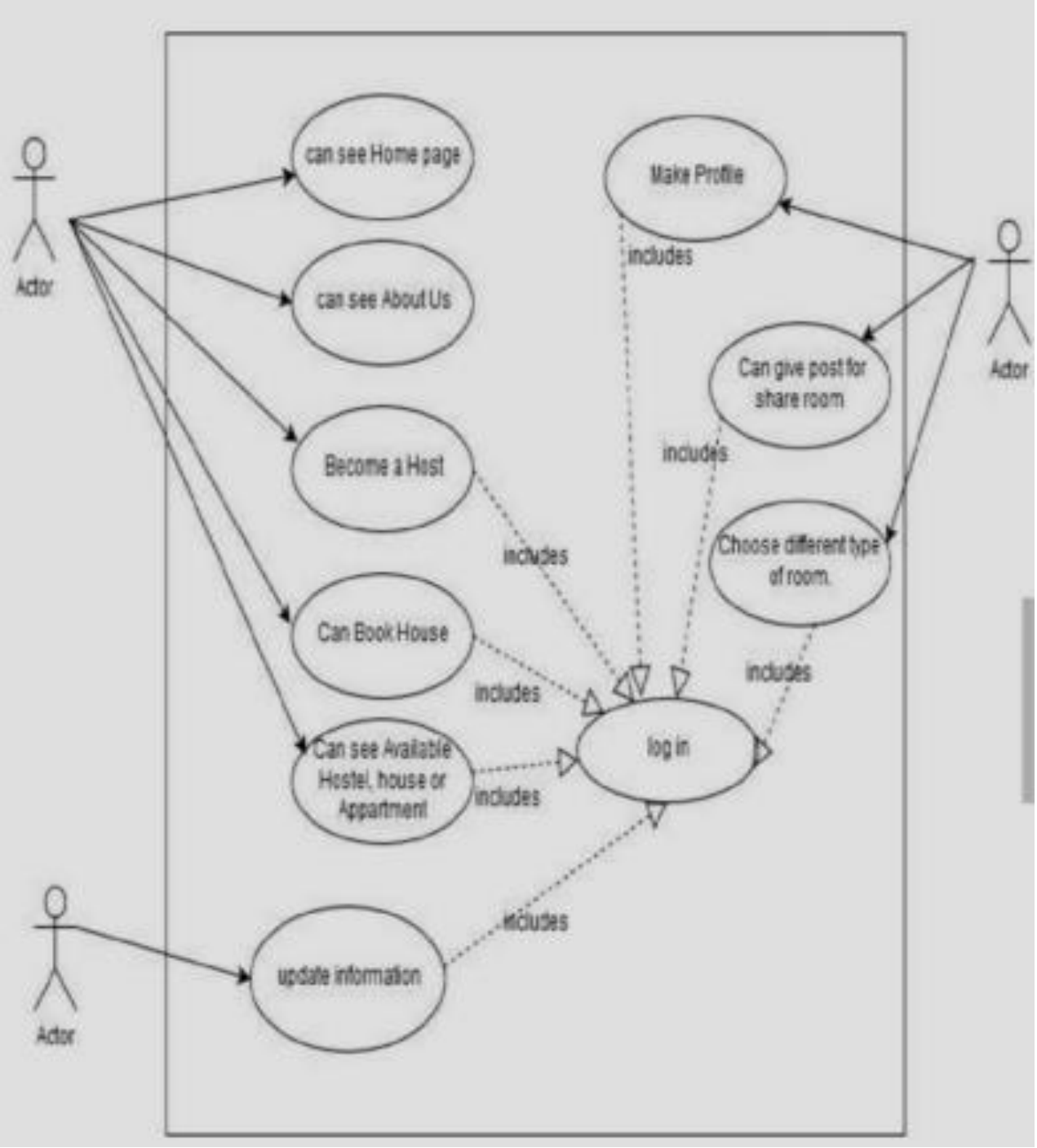
4.4 Messaging and Communication Module

The **Messaging and Communication Module** facilitates secure communication between users, allowing them to discuss living arrangements, preferences, and any other relevant information. This module includes a messaging interface where users can send and receive messages in real time, enhancing interaction and connection. Additionally, it may implement notifications to alert users of new messages, ensuring they remain engaged with potential roommates. By fostering open communication, this module plays a vital role in helping users evaluate compatibility and make informed decisions.

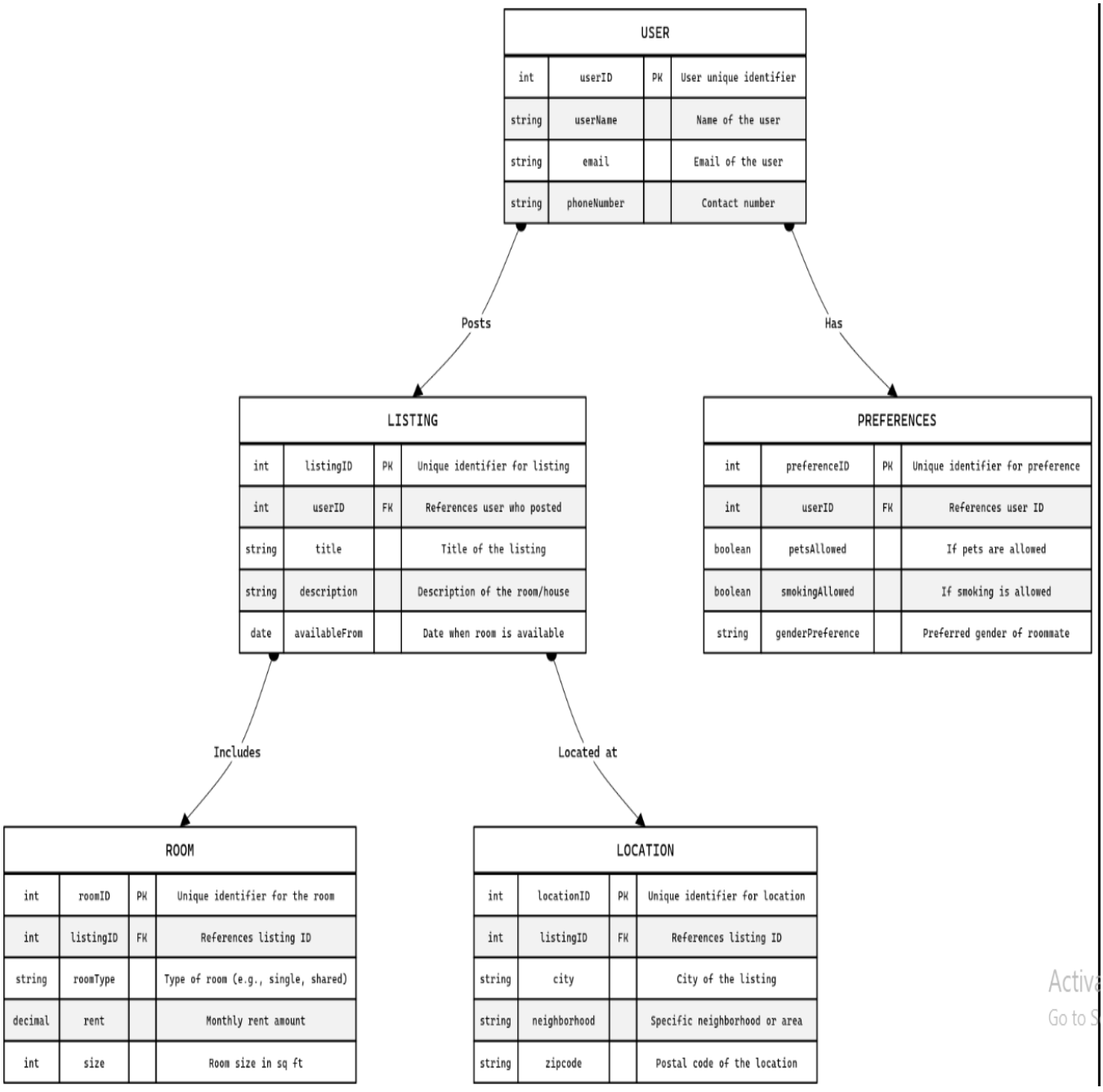
CHAPTER 5

SYSTEM DESIGN

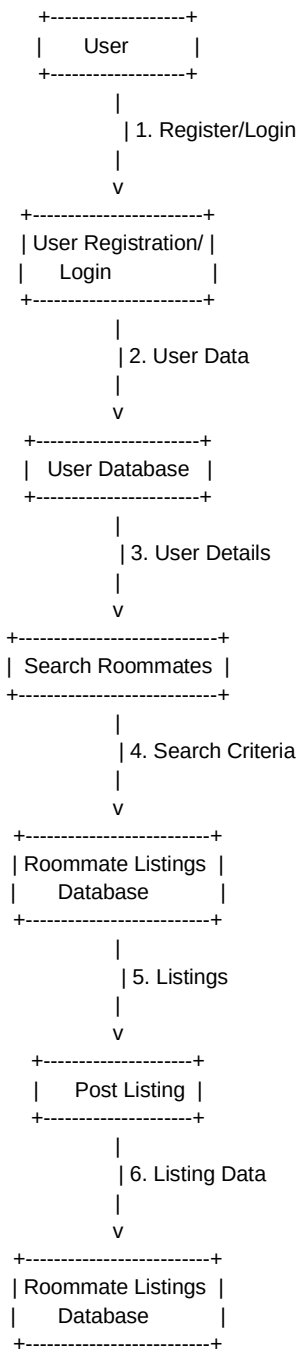
5.1 USE CASE DIAGRAM



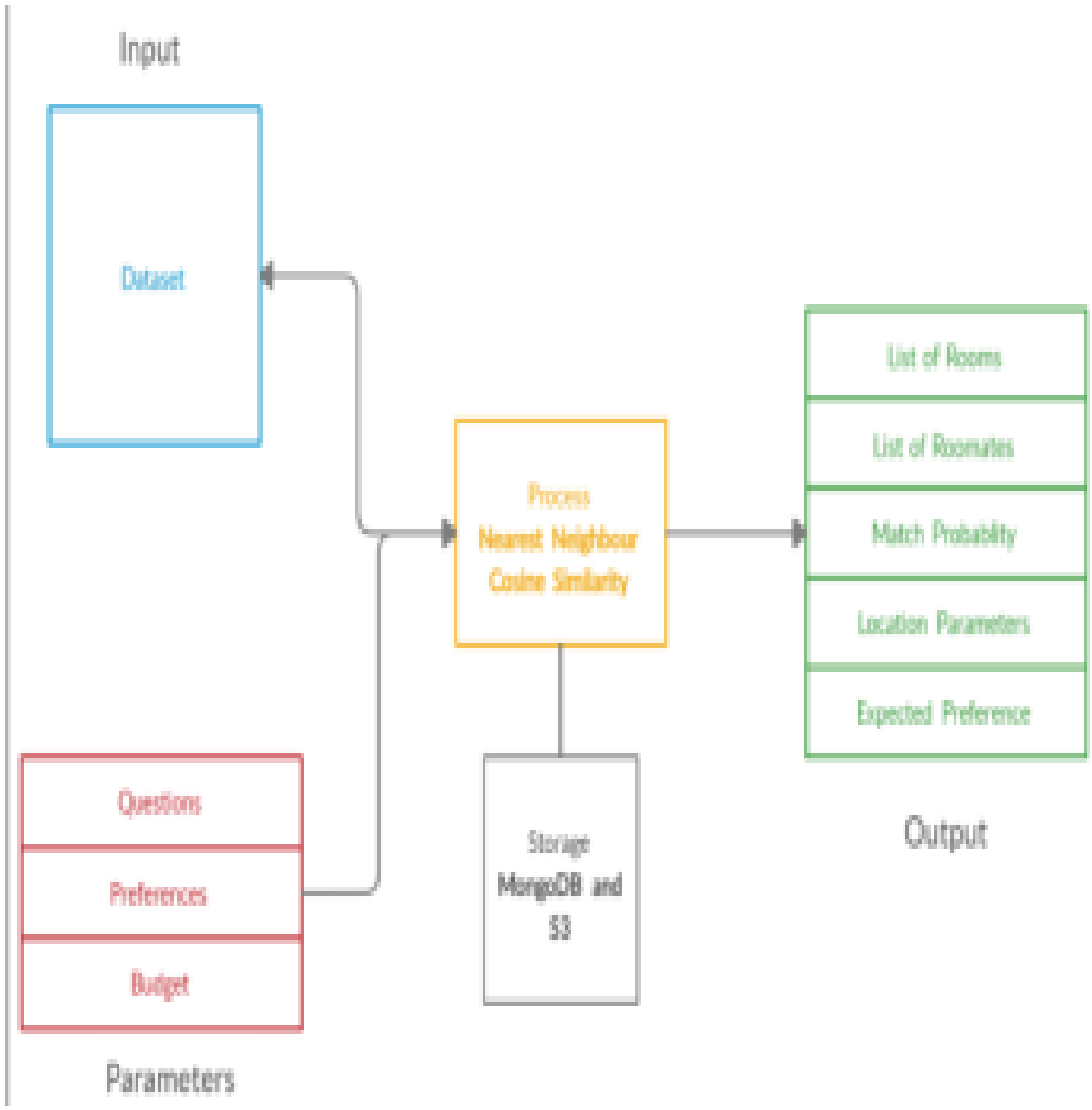
5.2 ER DIAGRAM



5.3 DATA FLOW DIAGRAM



5.4 ACTIVITY DIAGRAM



CHAPTER 6

SAMPLE CODING

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Roomie Match - Home</title>    <link rel="stylesheet" href="styles.css">
<link href="https://fonts.googleapis.com/icon?family=Material+Icons"
rel="stylesheet">
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

<!-- jQuery library -->
<script
src="https://cdn.jsdelivr.net/npm/jquery@3.7.1/dist/jquery.slim.min.js"></script>

<!-- Popper JS -->
<script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></scri
pt>
<script defer src="https://use.fontawesome.com/releases/v5.15.4/js/all.js"
integrity="sha384-
rOA1PnstxnOBLzCLMcre8ybwbTmemjzdNlILg8O7z1lUkLXozs4DHonlDtnE7fpc"
crossorigin="anonymous"></script>
<!-- Latest compiled JavaScript -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/js/bootstrap.bundle.min.js
"></script>
<script src="https://kit.fontawesome.com/a076d05399.js"
crossorigin="anonymous"></script>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

</head>
<style>
.modal {
display: none; /* Hidden by default */
position: fixed;
z-index: 10; /* Sit on top */
```

```
left: 0;
top: 0;
width: 100%;
height: 100%;
background-color: rgba(0, 0, 0, 0.6); /* Black background with opacity */
overflow: hidden; /* Disable scrolling */
}

/* Modal Content */
.modal-content {
position: relative;
background-color: #fff;
margin: 10% auto;
padding: 20px;
border: 1px solid #888;
width: 40%; /* Modal width */
border-radius: 10px;
box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
z-index: 100; /* Keep modal content above everything else */
max-height: 80vh; /* Limit modal height to fit within screen */
overflow: hidden; /* Disable internal scrolling */
}

.close {
color: #aaa;
position: absolute;
top: -15px;
right: -15px;
font-size: 28px;
font-weight: bold;
background-color: #fff;
border-radius: 50%;
padding: 5px 10px;
box-shadow: 0 2px 4px rgba(0, 0, 0, 0.2);
cursor: pointer;
z-index: 101;
opacity: 0.5; /* Semi-transparent, indicates disabled */
/* Disable click */
}

/* To enable the close button after login */

.close:hover,
.close:focus {
color: black;
```

```
text-decoration: none;
cursor: pointer;
}

/* Blur the background when the modal is open */
.background-blur {
filter: blur(5px);
}

/* Modal iframe styling */
.iframe-login {
width: 100%;
height: 400px; /* Adjust height according to your login page */
border: none;
}


</style>
<body>


<nav class="navbar">
<div class="navbar-logo">
<h1>Roomie Match</h1>
</div>
<div class="navbar-links">
<a href="index.html">Home</a>
<a href="post-room.html">Find Your Roomate</a>
<a href="login.html">Login</a>
<a href="register.html">Register</a>
</div>
</nav>

<div class="main-content d-flex flex-column">
<div>
<p class="ml-5">Find Male and Female roomies in Chennai<span><i class="material-
icons" style="font-size:15px;color:red">add_location</i></span> </p>
<div class="d-flex flex-row">
<form class="example" action="">
<input class="ml-5"type="text" id="myInput" placeholder=" Find Your Perfect
Roommate, Share Your Ideal Space in Chennai&#128205;"name="search">
</form>
<button class="sort" id="sort">
<i class="fa fa-sort"></i>
</button>
```

```
</div>
<div style="display:none;" id="div1">
<button class="but" style="margin-left:400px" id="showAll">
Show all

</button>
<button class="but" id="filterMale">
Male
</button>
<button class="but" id="filterFemale">
Female

</button>
<button class="but" id="filterOthers">
Others
</button>
</div>
<!-- Login Modal -->
<div id="loginModal" class="modal">
<div class="modal-content">
<span class="close">&times;</span>
<iframe src="login.html" class="iframe-login"></iframe>
</div>
</div>

<div class="product-list mt-5 ml-5" id="productList">
<!-- Cards will be dynamically generated here -->
</div>
<footer>
<div class="rounded-social-buttons mt-5">
<a class="social-button facebook" href="https://www.facebook.com/"
target="_blank"><i class="fab fa-facebook-f"></i></a>
<a class="social-button twitter" href="https://www.twitter.com/"
target="_blank"><i class="fab fa-twitter"></i></a>
<a class="social-button linkedin" href="https://www.linkedin.com/"
target="_blank"><i class="fab fa-linkedin"></i></a>
<a class="social-button tiktok" href="https://www.tiktok.com/" target="_blank"><i
class="fab fa-tiktok"></i></a>
<a class="social-button youtube" href="https://www.youtube.com/"
target="_blank"><i class="fab fa-youtube"></i></a>
<a class="social-button instagram" href="https://www.instagram.com/"
target="_blank"><i class="fab fa-instagram"></i></a>
</div>
</footer>
```

```
<script>
document.addEventListener('DOMContentLoaded', function() {
const modal = document.getElementById('loginModal');
const span = document.getElementsByClassName('close')[0];

// Check if user is not logged in (using localStorage)
if (!localStorage.getItem('isLoggedIn')) {
showLoginModal();
}

// Function to show the login modal and blur the background
function showLoginModal() {
modal.style.display = 'block';
document.body.classList.add('modal-open'); // Blur background
}

// Close modal when clicking on the 'x'
span.onclick = function() {
modal.style.display = 'none';
document.body.classList.remove('modal-open'); // Remove blur
};

// Close the modal if the user clicks outside the modal
window.onclick = function(event) {
if (event.target == modal) {
modal.style.display = 'none';
document.body.classList.remove('modal-open'); // Remove blur
}
};

// For testing - after login, set logged in status in localStorage
document.querySelector('.navbar').addEventListener('click', function() {
// Set login status (simulate login)
localStorage.setItem('isLoggedIn', 'true');
});
});

//start
$(document).ready(function(){
$('#sort').click(function(){
$('#div1').show(); // Show the filter buttons on clicking sort
});
});

// F// Function to create a room card
```

```
// Function to create a room card
// Function to create a room card
function filterRoomsByCity(searchTerm) {
const productList = document.getElementById('productList');
const rooms = JSON.parse(localStorage.getItem('rooms')) || [];

// Clear existing cards
productList.innerHTML = '';

// Filter rooms by the search term (location/city)
const filteredRooms = rooms.filter(room => {
// Combine area, city, landmark, street, and block into one search string
const roomDetails = `
${room.area}
${room.city}
${room.landmark}
${room.street}
${room.block}
${room.location}
`.toLowerCase();

// Check if the search term is found in the combined string
return roomDetails.includes(searchTerm.toLowerCase());
});

if (filteredRooms.length > 0) {
filteredRooms.forEach((room, index) => {
const card = createCard(room, index); // Pass index to createCard for unique key
productList.appendChild(card);
});
} else {
productList.innerHTML = '<p>No rooms found for the specified location.</p>';
}

}

// Add event listener to search input field (id="myInput")
document.getElementById('myInput').addEventListener('input', function() {
const searchTerm = this.value; // Get the value entered in the search input
filterRoomsByCity(searchTerm); // Call the filter function with the search term
});
function createCard(room, index) {
const card = document.createElement('div');
card.className = 'product-card';
```

```
card.setAttribute('data-index', index); // Add a unique identifier based on the
index

card.innerHTML = `
<img src="${room.photo}" alt="${room.name}" class="product-image">
<h3>${room.name}</h3>
<p>No.${room.block}, ${room.street}</p>
<p>${room.area}, ${room.city}</p>
<p>Landmark:${room.landmark}</p>
<p>${room.location}</p>
<p>Rent Price: ₹${room.rent}</p>
<p>Looking for: ${room.gender.charAt(0).toUpperCase() + room.gender.slice(1)}</p>
<button class="delete-button" data-index="${index}">Remove</button>
`;

const deleteButton = card.querySelector('.delete-button');
deleteButton.addEventListener('click', function() {
removeRoom(index); // Use the index instead of room id to remove the room
});

return card;
}

// Updated removeRoom function
function removeRoom(index) {
let rooms = JSON.parse(localStorage.getItem('rooms')) || [];

// Remove the room at the specific index
rooms.splice(index, 1);

// Update localStorage with the new array of rooms
localStorage.setItem('rooms', JSON.stringify(rooms));

// Reload the rooms to reflect the changes
loadRooms();
}

// Function to load all rooms from localStorage
function loadRooms() {
const productList = document.getElementById('productList');
const rooms = JSON.parse(localStorage.getItem('rooms')) || [];

// Clear existing cards
productList.innerHTML = '';
```

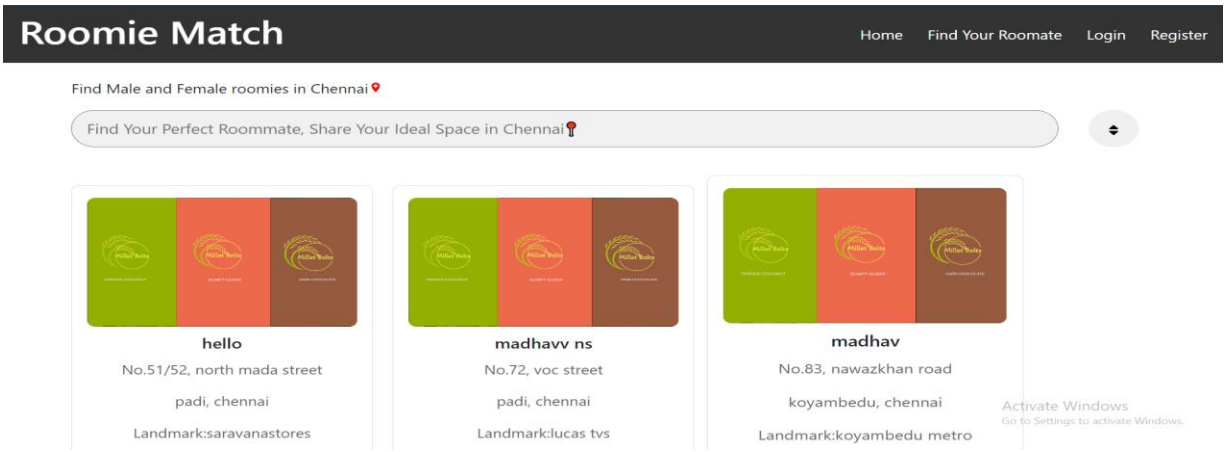
```
// Display all rooms
rooms.forEach((room, index) => {
const card = createCard(room, index); // Use index in createCard
productList.appendChild(card);
});
}

// Load rooms on page load
loadRooms();

// Load rooms on page load (if necessary)
// loadRooms();



// Filter rooms by gender
function filterByGender(gender) {
const productList = document.getElementById('productList');
const rooms = JSON.parse(localStorage.getItem('rooms')) || [];

// Clear existing cards
productList.innerHTML = '';

// Filter and display rooms by gender
const filteredRooms = rooms.filter(room => room.gender.toLowerCase() ===
gender.toLowerCase());
if (filteredRooms.length > 0) {
filteredRooms.forEach((room) => {
const card = createCard(room);
productList.appendChild(card);
});
} else {
productList.innerHTML = '<p>No rooms found for the selected gender.</p>';
}
}

// Add event listeners for filter buttons
document.getElementById('showAll').addEventListener('click', loadRooms); // Show
all rooms
document.getElementById('filterMale').addEventListener('click', () =>
filterByGender('male')); // Male rooms
document.getElementById('filterFemale').addEventListener('click', () =>
filterByGender('female')); // Female rooms
document.getElementById('filterOthers').addEventListener('click', () =>
filterByGender('others')); // Others
```

```
// Load all rooms on page load
loadRooms();
</script>




</body>
</html>
```

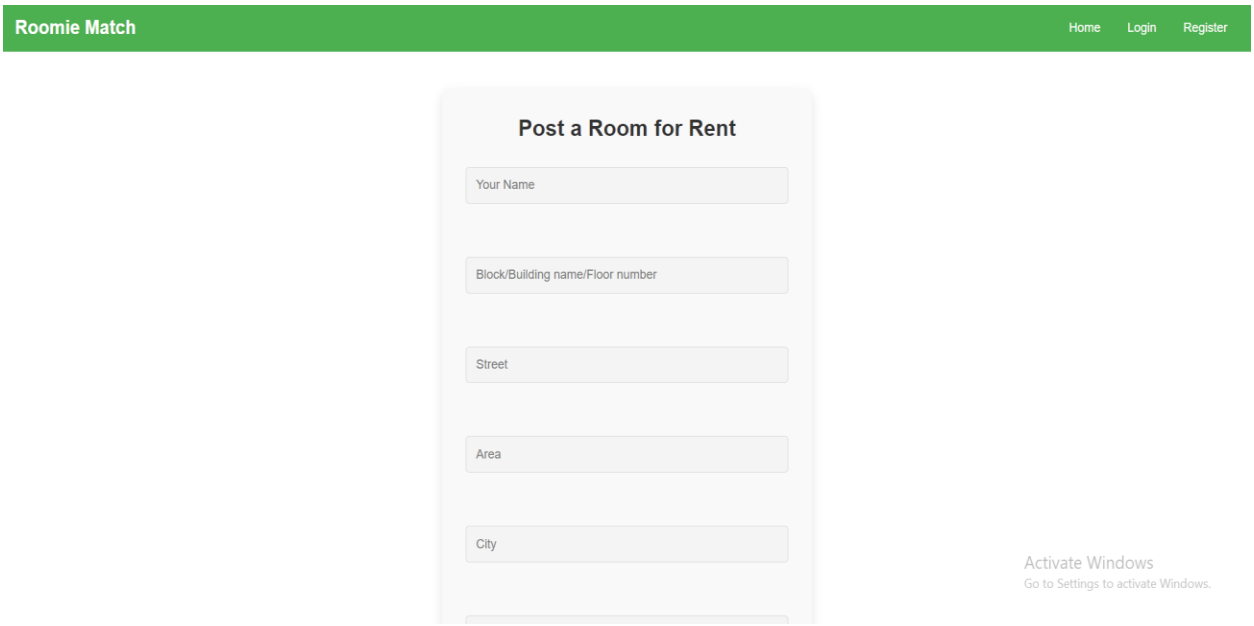
CHAPTER 7

SCREENSHOTS

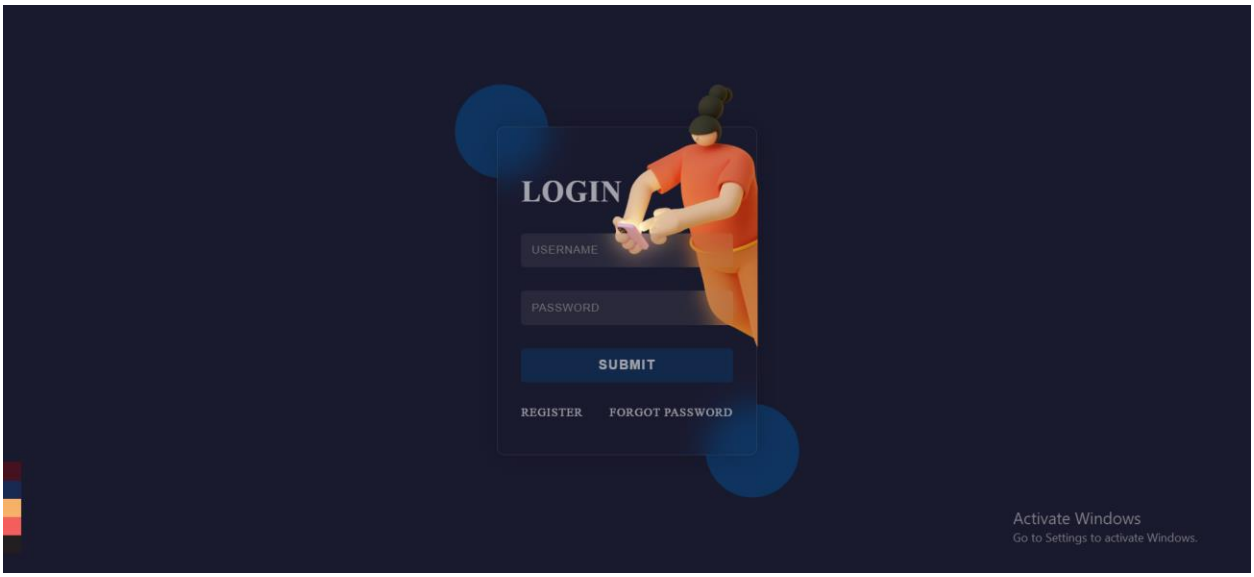
7.1 HOME PAGE



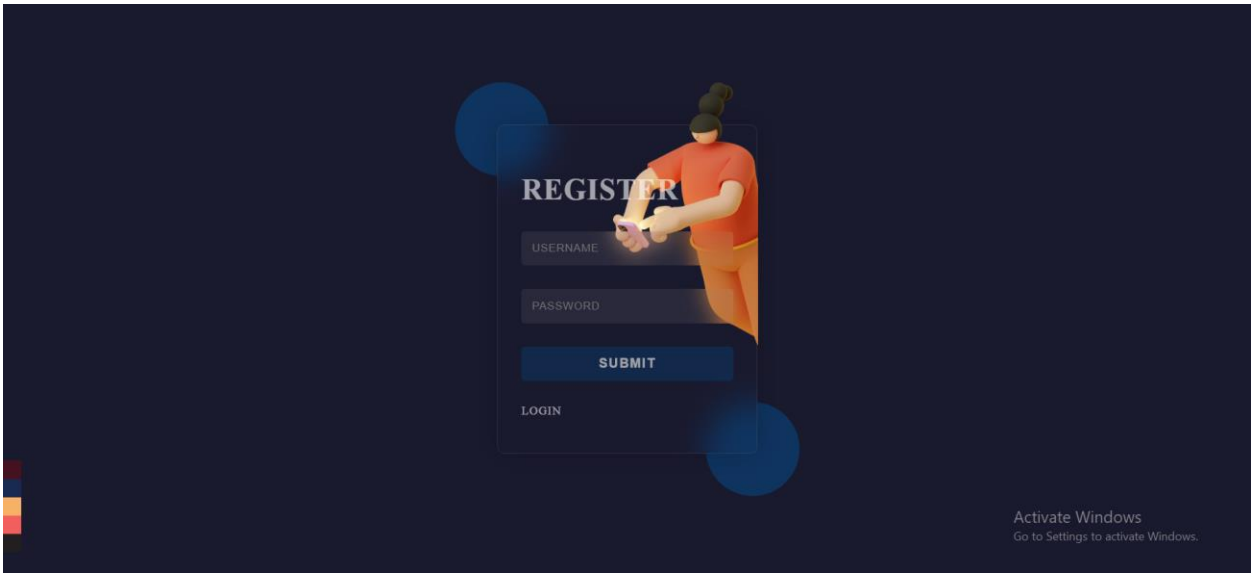
7.2 POST DETAIL PAGE



7.3 LOGIN PAGE



7.4 REGISTER PAGE



CHAPTER 8

CONCLUSION

In conclusion, the **Roommate Finder** website project presents a comprehensive solution for individuals seeking compatible roommates in Chennai. Through its well-structured modules, including the Landing Page, User Registration and Authentication, Roommate Search, and Messaging and Communication, the platform effectively addresses the challenges faced by users in finding suitable living arrangements. By prioritizing user experience and security, the project fosters a welcoming environment that encourages engagement and facilitates meaningful connections. As the demand for shared living continues to rise, this innovative platform is poised to meet the needs of a diverse user base, providing a reliable and user-friendly resource for roommate matching. Future enhancements may include the integration of advanced features and expansion to other cities, ensuring that the **Roommate Finder** remains a valuable tool in the evolving landscape of shared accommodations.

REFERENCES

- **Welling, L., & Thomson, L.** (2017). *PHP and MySQL Web Development*. Addison-Wesley.
- **Beighley, L., & Morrison, M.** (2009). *Head First PHP & MySQL*. O'Reilly Media.
- **Norman, D. A**. (2013). *The Design of Everyday Things*. Basic Books.
- **Garret, J. J.** (2011). *The Elements of User Experience: User-Centered Design for the Web and Beyond*. New Riders.
- **B. Boehm and P. Papalambros**, "Software Engineering," IEEE Software, vol. 22, no. 4, pp. 56-63, 2005.
- • **Boehm, B., & Papalambros, P.** (2005). "Software Engineering." *IEEE Software*, 22(4), 56-63. DOI: 10.1109/MS.2005.118.
- • **Salama, M. M. A., Elbakkali, M. Y., & Ghandour, M. F.** (2017). "A Review of User Experience in Web Applications." *International Journal of Computer Applications*, 175(12), 1-5. DOI: 10.5120/ijca2017915232.
- • **Shafique, S., & Chaudhary, A. R.** (2017). "Web-based Roommate Matching System: A Case Study of Online Roommate Finder." *Journal of Software Engineering and Applications*, 10(4), 123-134. DOI: 10.4236/jsea.2017.104012.
- • **Liu, J., Wong, C. L. D., & Huang, Z. H.** (2019). "A Study on User Interface Design of Web Applications." *Journal of Software Engineering Research and Development*, 7(1), 1-10. DOI: 10.1186/s40411-019-0080-4.
- • **Chen, Y. A.** (2016). "User-Centered Design for Web Applications: A Systematic Review." *Journal of Computer Information Systems*, 56(2), 1-10. DOI: 10.1080/08874417.2015.11646043.
- • **Nielsen, J., & Loranger, H.** (2006). "Prioritizing Web Usability." *New Riders Publishing*. (While this is a book, it often appears in academic references related to web usability and user experience research.)
-