KARNATAK UNIVERSITY

DHARWAD

PROJECT REPORT ON

"REAL-ESTATE WEBSITE"

SUBMITTED TO KARNATAKA UNIVERSITY DHARWAD IN PARTIAL FULFILLMENT FOR THE DEGREE OF

BACHELOR OF COMPUTER APPLICATIONS

SUBMITTED BY

Mr. V Kashi Reddy U02CS22S0027

Mr. Channabasavana Gouda U02CS22S0049

UNDER THE GUIDANCE OF

Prof. Ashwini Hongal

KLE SOCIETY'S COLLEGE OF COMPUTER APPLICATIONS
DHARWAD

2024-25

KARNATAKA UNIVERSITY

DHARWAD

KLE’S BCA COLLEGE DHARWAD - 580008

DEPARTMENT OF BACHELOR OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that project work entitled

“Real-Estate Website”

Submitted in partial fulfilment of the requirement for the award of degree of Bachelor of Computer Applications is a result of bonafide work carried out by

Mr. V KASHI REDDY

Mr. CHANNABASAVANA GOUDA

During the academic year 2024-2025

Under the Guidance of Prof.Ashwini Hongal

Prof.Ashwini Hongal
Project Guide
KLE’S BCA College
Dharwad

Prof.P.D REVANKAR
KLE’s BCA-Coordinator,
Dharwad

Examination Centre: KLE’S BCA College Dharwad

Examiner Signature

Date:

1. ________________

2.________________

DECLARATION

WE **V KASHI REDDY, CHANNABASAVANA GOUDA** students of 6th SEM BCA, **KLE'S BCA College Dharwad** bearing **U02CS22S0027 , U02CS22S0049** hereby declare that the project entitled **"REAL-ESTATE WEBSITE"** has been carried out by us under the supervision of BCA Coordinator **Prof.P.D REVANKAR** and Guide **Prof.ASHWINI HONGAL** and submitted in partial fulfilment of the requirements for the award of the Degree of Bachelor of Computer Applications by the **Karnataka University, Dharwad during** the academic year 2024-2025**.** This report has not been submitted to any other Organization/University for any award of degree or certificate.

Name:	Name:
Signature:	Signature:

ACKNOWLEDGMENT

"Task successful" makes everyone happy. But the happiness will be gold without glitter if we didn't state the persons who have supported us to make it a success.

Success will be crowned to people who made it a reality but the people whose constant guidance and encouragement made it possible will be crowned first on the eve of success.

This acknowledgement transcends the reality of formality when we would like to express deep gratitude and respect to all those people behind the screen who guided, inspired and helped me for the completion of our project work.

I consider myself lucky enough to get such a good project. This project would add as an asset to my academic profile.

I would like to express my gratitude to **Prof.P.D REVANKAR** BCA Coordinator for his constant supervision, guidance and co-operation throughout the project.

I would like to express my thankfulness to my guide **Prof.ASHWINI HONGAL** for constant motivation and valuable help through the project work.

Finally, I would like to thanks my friends for their co-operation to complete this project.

V KASHI REDDY-[U02CS22S0027]

CHANNABASAVANA GOUDA-[U02CS22S0049]

CONTENT

FIGURES CONTENT

ABSTRACT

In the modern digital era, the real estate industry is rapidly transitioning from traditional, offline methods to more efficient, web-based platforms. This project, titled "Real Estate Website", aims to provide a comprehensive and user-friendly solution for listing, searching, and managing real estate properties online. The main objective of this system is to bridge the gap between property owners, buyers, and renters through a centralized and interactive web portal.

The website is developed using HTML for structuring the web pages, CSS for styling and layout design, JavaScript for client-side interactivity, and PHP for server-side scripting and database integration. The backend is supported by MySQL, which is used to store user data, property listings, and transaction records.

This platform enables users to register and log in securely, post their properties with complete details (such as price, location, type, and images), and search or filter listings based on various criteria like location, budget, and property type. The admin panel allows for the management of users and listings, ensuring that the website remains organized and free of spam or unauthorized content.

CHAPTER 1

INTRODUCTION

In today’s digital age, the real estate industry has seen a significant transformation with the help of technology. People are increasingly turning to online platforms to buy, sell, or rent properties instead of relying solely on traditional methods like newspaper ads or physical agents. This trend has highlighted the need for efficient, user-friendly, and secure online real estate platforms.

This project focuses on developing a **Real Estate Website** that simplifies the property search and management process for buyers, sellers, and real estate agents. The system is designed to offer a digital solution where users can explore property listings, filter based on their preferences, and communicate with property owners or agents—all within a single platform.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing system

In the current digital age, many commercial real estate platforms such as **99acres**, **MagicBricks**, **Housing.com**, and NoBroker are widely used. These platforms typically provide features such as:

- User registration and login
- Property search and filtering by location, type, price, etc.
- Contact options to get in touch with agents or property owners
- Premium listings and advertisements for increased visibility

Limitations of Existing Systems:

- ➢ **High Listing Fees**: Most platforms charge users or agents a fee for listing properties or for premium visibility, making it difficult for individuals or small businesses to use them regularly.
- ➢ **Limited Customization**: Users cannot customize the UI or backend functionalities according to their specific needs or branding.
- ➢ **Overloaded Interface**: Many commercial sites are cluttered with ads, premium promotions, and third-party content, which can hinder user experience.
- ➢ **Privacy Concerns**: Many users are concerned about how their contact data is handled and often receive unsolicited calls or messages.
- ➢ **Dependency on Third-Party Services**: These platforms often rely heavily on third-party services for payments, analytics, and communication, which can introduce reliability issues.

2.2 Proposed system

The proposed system is a **custom-built Real Estate Website** developed using **HTML, CSS, JavaScript, and PHP**. This system addresses the drawbacks of existing platforms by offering a simplified, ad-free, and easy-to-maintain solution for property listing and search.

Features of the Proposed System:

- **Property Listing**: Property owners or agents can list properties by submitting details such as location, type, price, images, and contact information.
- **User Interaction**: Users can view detailed property information and use a contact form to get in touch with the property lister.
- **Responsive UI**: Using HTML and CSS along with media queries, the website is optimized for use on desktops, tablets, and smartphones.
- **Dynamic Content with PHP**: PHP is used to interact with the database, enabling functionalities like user registration, login, property listing, and search.
- **Search and Filter**: Users can search for properties based on parameters like location, type (e.g., residential, commercial), and price range using JavaScript for client-side filtering and PHP for server-side filtering.
- **Admin Panel**: A secure admin interface can manage listings, user accounts, and perform content moderation.

CHAPTER 3

SYSTEM SPECIFICATION REQUIREMENTS

3.1 Software requirement

Component	Specification
Operating	Windows 7/8/10/11 or Linux
Backend Technology	PHP 7.x or above
Frontend	HTML, CSS, JavaScript
Database	MySQL
Web Server	Apache (via XAMPP/WAMP)
Scripting Support	PHP, JavaScript
Browser Compatibility	Google Chrome, Mozilla Firefox
Editor	VS Code

3.2 Hardware requirement

Component	Specification
Processor	Intel Pentium i3 or higher
RAM	Minimum 4 GB
Hard Disk	Minimum 100 GB
Monitor	15” LED/LCD
Keyboard and Mouse	Standard input devices
Internet Connection	Required for cloud hosting or APIs

These specifications ensure that the system can run smoothly in a lab setup, small business environment, or even be scaled for larger setups with appropriate hosting.

CHAPTER 4

SOFTWARE REQUIREMENTS SPECIFICATION

4.1 Introduction

This section defines the software requirements for the Real Estate Website project. It provides a comprehensive overview of the functionality and technical constraints of the system. The aim is to ensure all users and developers understand what the system is expected to do and how it will be implemented using only HTML, CSS, JavaScript, and PHP.

4.2 Purpose

The purpose of the Software Requirements Specification (SRS) is to document the functional and non-functional requirements of the Real Estate Website. It serves as a guideline for developers and stakeholders and helps maintain consistency throughout the development cycle. This document outlines how the website should behave and what technologies are utilized.

4.3 Scope

The Real Estate Website allows users to:

- Search for properties by location, price, and type.
- Post property listings (for sellers and agents).
- Contact property listers through inquiry forms.
- View property images and details.
- Admin panel for managing listings and users.

The system is built entirely with front-end (HTML, CSS, JavaScript) and back-end (PHP) technologies. The database interaction is handled through PHP and MySQL. The platform is accessible via modern web browsers

4.4 Specific requirements

4.4.1 Functional Requirements

- **User Registration and Login:** Users can sign up and log in using their email and password. PHP handles session management.
- **Property Listing:** Authenticated users can add property details including title, description, images, location, and price. Data is stored in a MySQL database.
- **Search and Filter Properties:** Users can search properties using filters like city, property type, price range, and more. JavaScript is used for real-time filtering; PHP fetches filtered results from the database.
- **Property Details Page:** Clicking on a property redirects to a detailed page showing full information and contact form.
- **Admin Panel:** Admin users can log in to manage listings and remove inappropriate content.

4.4.2 Non-Functional Requirements

- **Responsiveness:** The website layout adjusts to different screen sizes using CSS media queries.
- **Performance:** The website is optimized to load quickly by minimizing CSS and JS files.
- **Scalability:** Though simple, the code structure supports future enhancements such as adding chat support or map-based search.
- **Security:** Basic form validation is done via JavaScript; PHP sanitizes inputs to prevent SQL Injection and XSS attacks.
- **Maintainability:** Code is structured using reusable PHP components and modular CSS/JS files for easy updates.

CHAPTER 5

SYSTEM DESIGN

5.1 Architecture Diagram

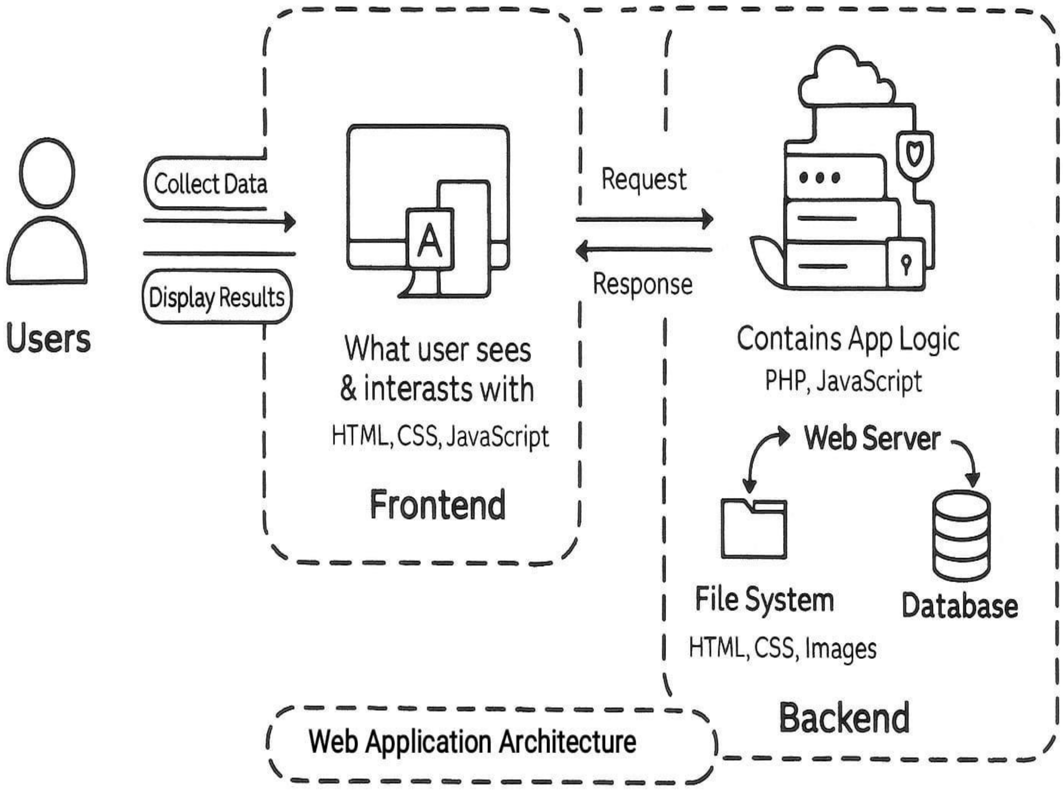


Fig 5.1 Architecture Diagram

5.2 Detailed design

5.2.1 Use case Diagram (User)

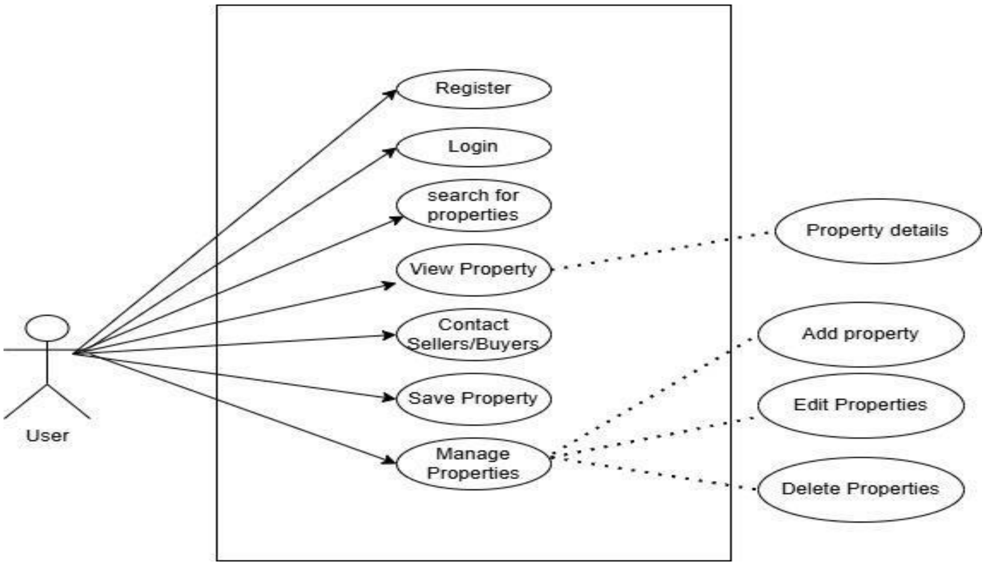


Fig 5.2.1 Use Case Diagram (User)

5.2.2 User Diagram (Admin)

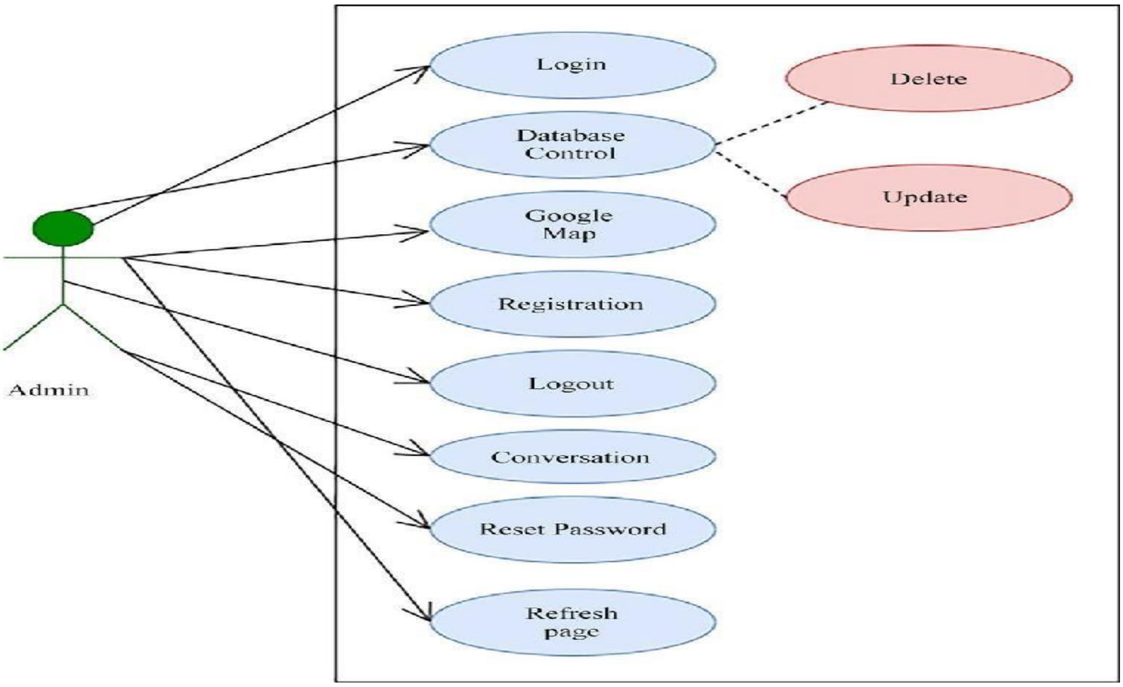


Fig 5.2.2 Use Case Diagram (Admin)

5.3 Data Flow Diagram

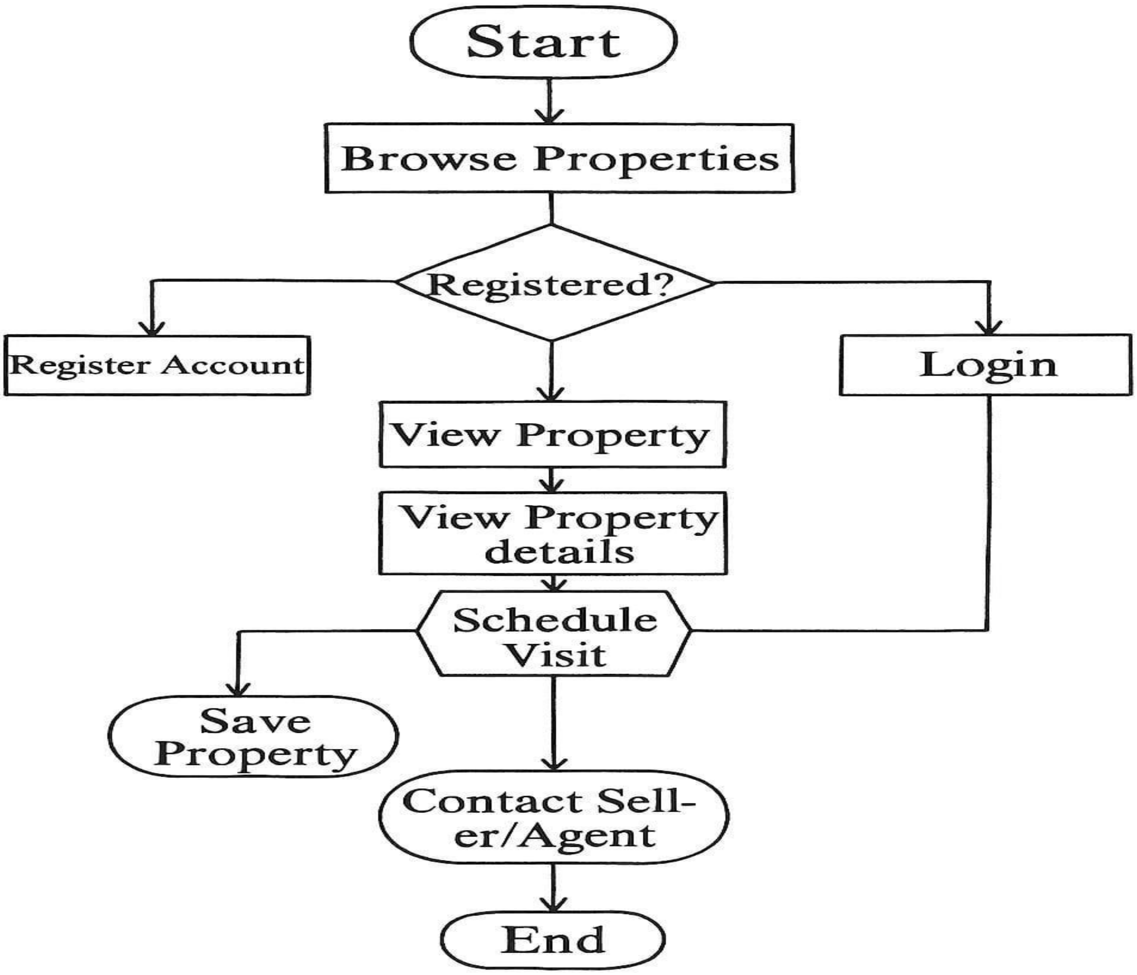


Fig 5.3 Data Flow Diagram

5.4 Sequence diagram

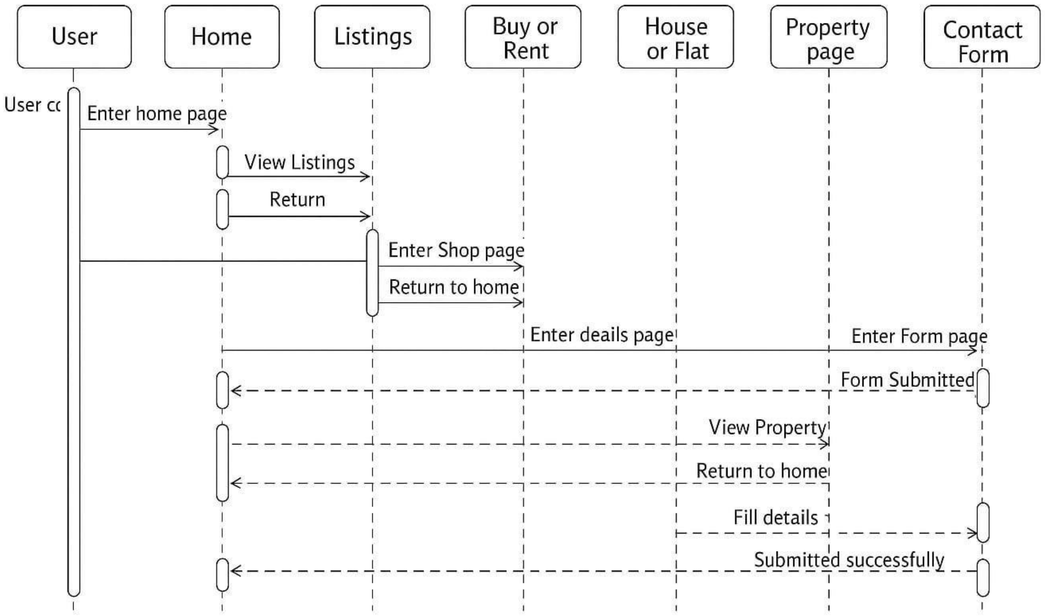


Fig 5.4 Sequence Diagram

CHAPTER 6

IMPLEMENTATION

6.1 Introduction PHP

PHP (Hypertext Preprocessor) is a widely-used open-source scripting language suited for web development and can be embedded into HTML. It executes on the server and is responsible for handling the backend logic of the **Real Estate Website**.

Key reasons for using PHP in this project include:

- Easy integration with HTML
- Wide community support and extensive libraries
- Seamless connection with MySQL
- Fast development and deployment

In this project, PHP is used for:

- Handling form submissions (e.g., property inquiries, registration)
- Validating and storing user input
- Authenticating users (Visitors, Registered Users, Agents)
- Managing dynamic content like property listings, saved favorites, and appointments

6.2 Introduction to MySQL Database integration

MySQL is a relational database management system that securely stores all system data. It manages data like:

- User credentials and profiles
- Property listings
- Agent and agency details
- Inquiries and contact requests
- Appointments for property visits
- Transaction

Each entity in the system corresponds to a table in the database, with relationships properly normalized to avoid redundancy.

This implementation provides a dynamic, reliable backend infrastructure essential for a web-based **Real Estate Website**.

6.3 Introduction HTML

HTML (Hypertext Markup Language) is the backbone of any web application, serving as the structure and content layer of the **Real Estate Website**. It defines how elements such as text, images, and links are displayed in the browser.

Key features of HTML in this project include:

- Structuring web pages with elements like headings, paragraphs, lists, and links
- Organizing property listings using containers, cards, and grids
- Embedding multimedia content such as property images and videos
- Creating forms for user interactions like search queries, registration, and appointment scheduling
- Supporting semantic elements (e.g., <header>, <section>, <article>, <footer>) for better accessibility and SEO

In this Real Estate Website project, HTML is used for:

- Building the home page, search pages, and property detail pages
- Creating forms for user registration, login, and contact inquiries
- Structuring the layout of property cards and agent profiles
- Embedding interactive elements (e.g., search filters, dropdowns, and call-to-action buttons)
- Providing the foundation for styling with CSS and dynamic interactions with JavaScript or PHP

6.4 Introduction CSS

CSS (Cascading Style Sheets) is used to style the HTML content. It enhances the visual presentation of the **Real Estate Website**, including:

- Layouts (grids, sections, containers for property listings and search results)
- Colour schemes and typography for a professional, inviting look
- Responsive design for seamless browsing on mobile and desktop devices
- User interaction feedback (hover effects on property cards, smooth transitions on forms

CSS ensures a clean and modern look for the application, improving the browsing and searching experience for all users of the **Real Estate Website**.

6.5 Introduction JavaScript

JavaScript is a versatile, high-level scripting language that brings interactivity and dynamic behaviour to web pages. It executes in the user's browser and is essential for enhancing the user experience in modern web applications, including real estate websites.

Key reasons for using JavaScript in this project include:

- Enabling dynamic updates without reloading the entire page. (e.g., using AJAX to fetch updated property listings or search results)
- Enhancing the interactivity of the website through animations, sliders, and image galleries. (e.g., property photo carousels)
- Validating user inputs on the client side to ensure data integrity before submission.
- Providing real-time feedback and alerts. (e.g., form validation errors, saved search notifications)
- Implementing advanced features like map integrations, interactive property markers, and real-time filtering.

In this project, JavaScript is used for:

- Implementing search filters and dynamically updating property listings based on user selections.
- Handling form submissions asynchronously to improve responsiveness.
- Enhancing the visual presentation of property details. (e.g., sliders for image galleries, expandable sections for property features)

CHAPTER 7

UI DESIGN AND OUTPUT

7.1 Landing Page [Login Page]

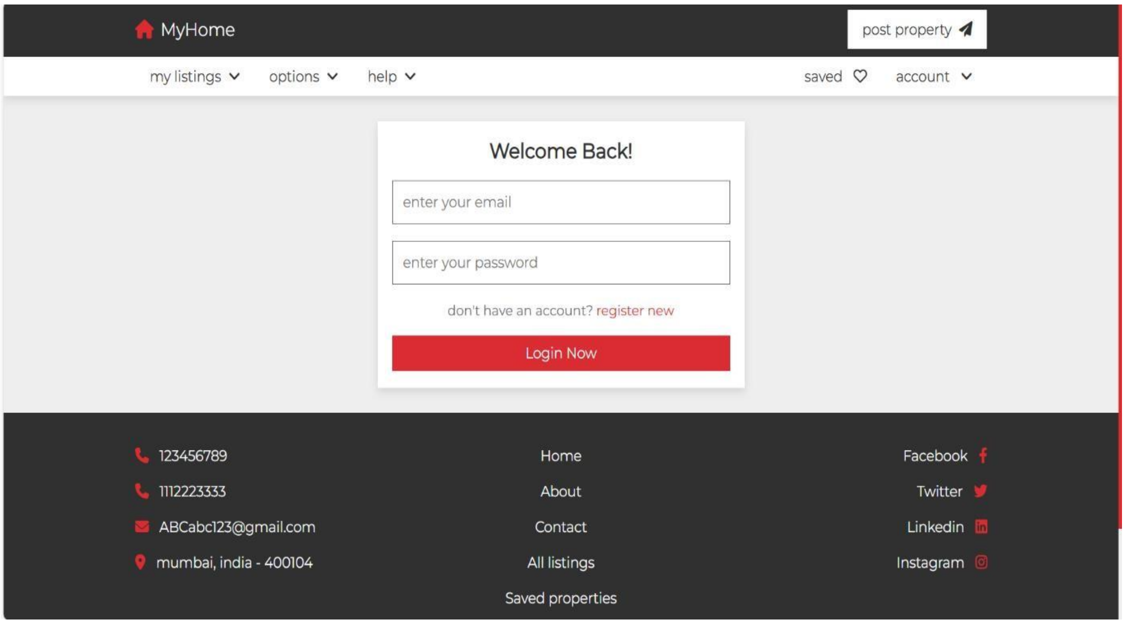


Fig 7.1 Landing Page [Login Page]

7.2 Main Page

7.2.1 Home Page

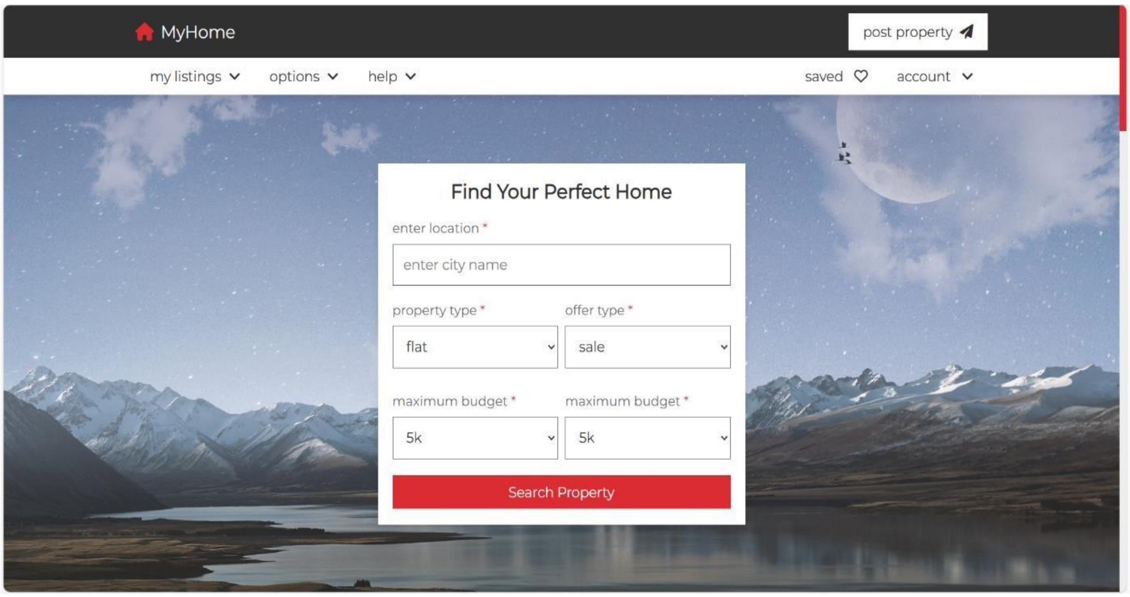
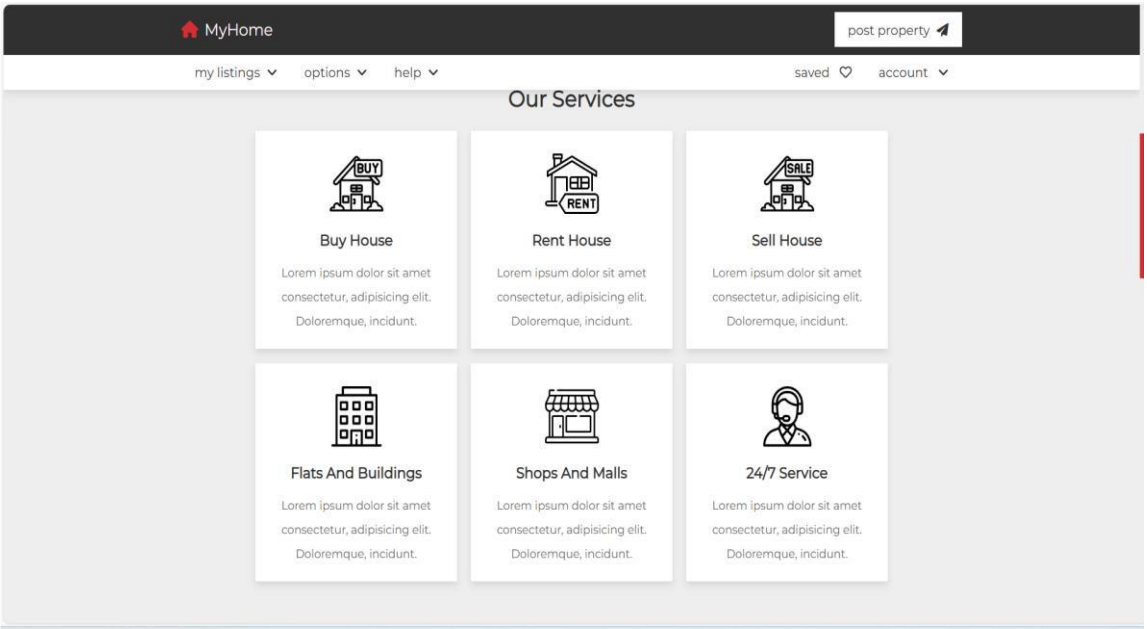


Fig 7.2.1 Home Page

7.2.2 Our Services



Our Services

Buy House

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Doloremque, incidunt.

Rent House

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Doloremque, incidunt.

Sell House

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Doloremque, incidunt.

Flats And Buildings

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Doloremque, incidunt.

Shops And Malls

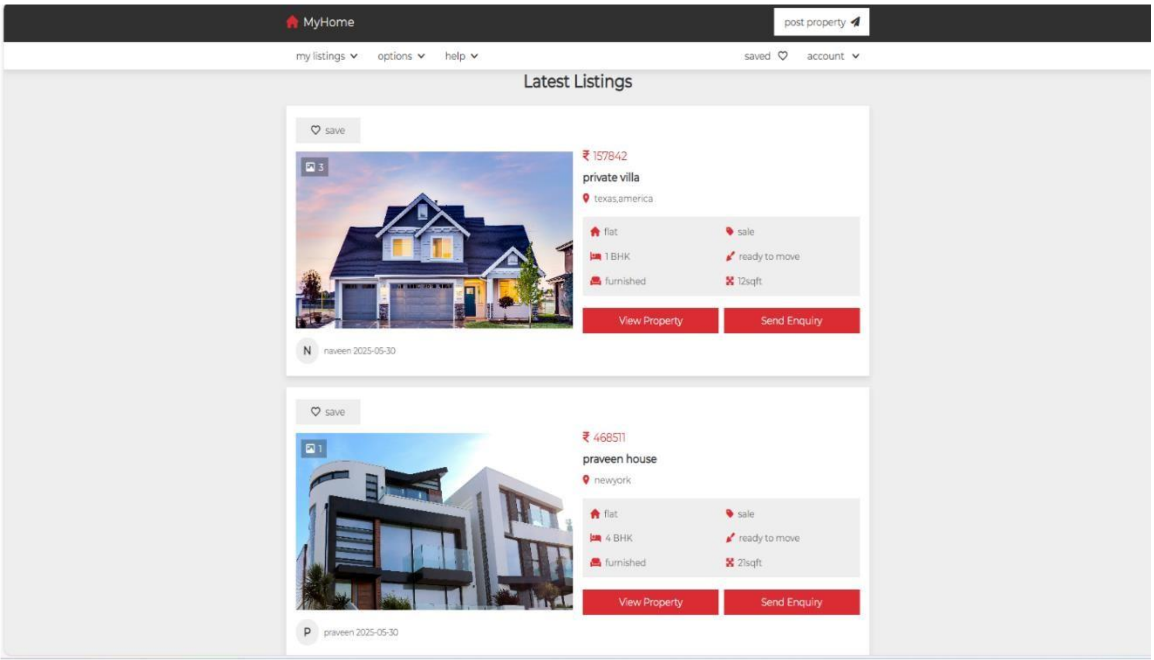
Lorem ipsum dolor sit amet consectetur, adipisicing elit. Doloremque, incidunt.

24/7 Service

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Doloremque, incidunt.

Fig 7.2.2 Our Services

7.2.3 Property Listings



Latest Listings

₹ 157842

private villa

texas,america

flat | sale
1 BHK | ready to move
furnished | 12sqft

N naveen 2025-05-30

₹ 468511

praveen house

newyork

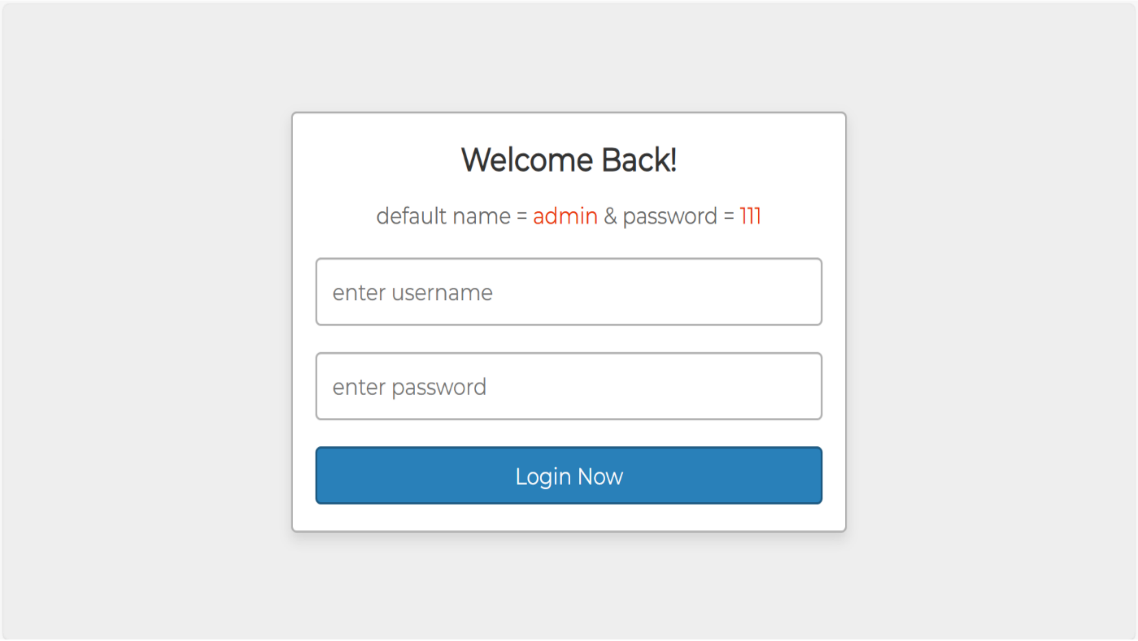
flat | sale
4 BHK | ready to move
furnished | 21sqft

P praveen 2025-05-30

Fig 7.2.3 Property Listings

7.3 Admin Pannel

7.3.1 Admin Pannel Login

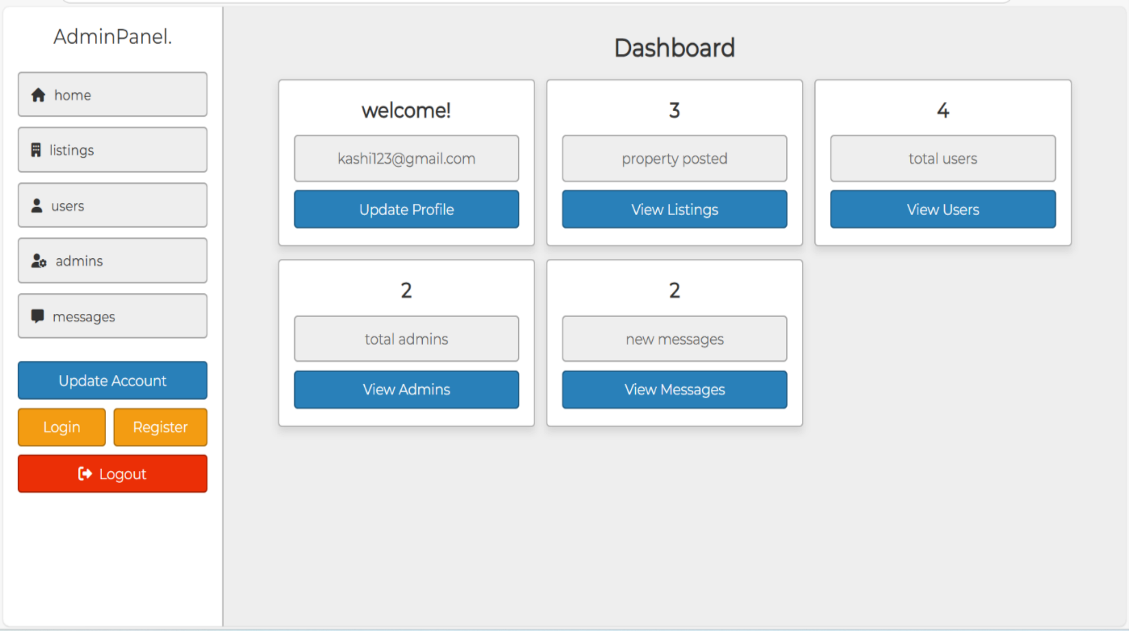


Welcome Back!

default name = admin & password = 111

7.3.1 Admin Pannel Login

7.3.2 Admin Dashboard



welcome!
kashi123@gmail.com

3
property posted

4
total users

2
total admins

2
new messages

Fig 7.3.2 Admin Dashboard

CHAPTER 8

Coding

the development of the Real-estate website involves creating modular and organized code for various functionalities. Below are some key code snippets and explanations.

8.1 User Registeration

```
<?php
include 'components/connect.php';
if(isset($_COOKIE['user_id'])){
   $user_id = $_COOKIE['user_id'];
}else{
   $user_id = '';
}
if(isset($_POST['submit'])){
   $id = create_unique_id();
   $name = $_POST['name'];
   $name = filter_var($name, FILTER_SANITIZE_STRING);
   $number = $_POST['number'];
   $number = filter_var($number, FILTER_SANITIZE_STRING);
   $email = $_POST['email'];
   $email = filter_var($email, FILTER_SANITIZE_STRING);
   $pass = sha1($_POST['pass']);
   $pass = filter_var($pass, FILTER_SANITIZE_STRING);
   $c_pass = sha1($_POST['c_pass']);
   $c_pass = filter_var($c_pass, FILTER_SANITIZE_STRING);
    $select_users = $conn->prepare("SELECT * FROM `users` WHERE
email = ?");
   $select_users->execute([$email]);
 if($select_users->rowCount() > 0){
 $warning_msg[] = 'email already taken!';
   }else{
      if($pass != $c_pass){
         $warning_msg[] = 'Password not matched!';
      }else{
         $insert_user = $conn->prepare("INSERT INTO `users`(id, name, 
number, email, password) VALUES(?,?,?,?,?)");
        $insert_user->execute([$id, $name, $number, $email, $c_pass]);
        if($insert_user){
            $verify_users = $conn->prepare("SELECT * FROM `users` 
WHERE email = ? AND password = ? LIMIT 1");
           $verify_users->execute([$email, $pass]);
           $row = $verify_users->fetch(PDO::FETCH_ASSOC);
           if($verify_users->rowCount() > 0){
              setcookie('user_id', $row['id'], time() + 60*60*24*30, '/');
              header('location:home.php');
           }else{
              $error_msg[] = 'something went wrong!';
           }
        }
?>
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
 <meta name="viewport" content="width=device-width, initial-
scale=1.0">
   <title>Register</title>
   <!-- font awesome cdn link  -->
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.2.0/css/all.min.css">
   <!-- custom css file link  -->
   <link rel="stylesheet" href="css/style.css">
</head>
<body>
<?php include 'components/user_header.php'; ?>
<!-- register section starts  -->
<section class="form-container">
   <form action="" method="post">
```

```
      <h3>create an account!</h3>
         <input type="tel" name="name" required maxlength="50"
placeholder="enter your name" class="box">
        <input type="email" name="email" required maxlength="50"
placeholder="enter your email" class="box">
         <input type="number" name="number" required min="0"
max="9999999999" maxlength="10" placeholder="enter your number"
class="box">
       <input type="password" name="pass" required maxlength="20"
placeholder="enter your password" class="box">
      <input type="password" name="c_pass" required maxlength="20"
placeholder="confirm your password" class="box">
         <p>already have an account? <a href="login.html">login
now</a></p>
         <input type="submit" value="register now" name="submit"
class="btn">
  </form>
</section>
<!-- register section ends -->
<script
src="https://cdnjs.cloudflare.com/ajax/libs/sweetalert/2.1.2/sweetalert.m
in.js"></script>
<?php include 'components/footer.php'; ?>
<!-- custom js file link  -->
<script src="js/script.js"></script>
<?php include 'components/message.php'; ?>
</body>
</html>
```

8.2 Dashboard Page

```
<?php
include 'components/connect.php';
if(isset($_COOKIE['user_id'])){
  $user_id = $_COOKIE['user_id'];
}else{
  $user_id = '';
  header('location:login.php');
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>dashboard</title>
  <!-- font awesome cdn link  -->
      <link      rel="stylesheet"      href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.2.0/css/all.min.css">
  <!-- custom css file link  -->
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
<?php include 'components/user_header.php'; ?>
<section class="dashboard">
  <h1 class="heading">dashboard</h1>
  <div class="box-container">
    <div class="box">
    <?php
       $select_profile = $conn->prepare("SELECT * FROM `users` WHERE id = ?
LIMIT 1");
      $select_profile->execute([$user_id]);
      $fetch_profile = $select_profile->fetch(PDO::FETCH_ASSOC);
    ?>
    <h3>welcome!</h3>
    <p><?= $fetch_profile['name']; ?></p>
    <a href="update.php" class="btn">update profile</a>
    </div>
    <div class="box">
      <h3>filter search</h3>
      <p>search your dream property</p>
      <a href="search.php" class="btn">search now</a>
    </div>
    <div class="box">
    <?php
       $count_properties = $conn->prepare("SELECT * FROM `property` WHERE
user_id = ?");
      $count_properties->execute([$user_id]);
      $total_properties = $count_properties->rowCount();
    ?>
```

```
      <h3><?= $total_properties; ?></h3>
      <p>properties listed</p>
      <a href="my_listings.php" class="btn">view all listings</a>
      </div>
      <div class="box">
      <?php
         $count_requests_received = $conn->prepare("SELECT * FROM `requests`
WHERE receiver = ?");
        $count_requests_received->execute([$user_id]);
        $total_requests_received = $count_requests_received->rowCount();
      ?>
      <h3><?= $total_requests_received; ?></h3>
      <p>requests received</p>
      <a href="requests.php" class="btn">view all requests</a>
      </div>
      <div class="box">
      <?php
$count_saved_properties = $conn->prepare("SELECT * FROM `saved` WHERE
user_id = ?");
        $count_saved_properties->execute([$user_id]);
        $total_saved_properties = $count_saved_properties->rowCount();
      ?>
      <h3><?= $total_saved_properties; ?></h3>
      <p>properties saved</p>
      <a href="saved.php" class="btn">view saved properties</a>
      </div>
   </div>
</section>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/sweetalert/2.1.2/sweetalert.min.js"></scri
pt>
<?php include 'components/footer.php'; ?>
<!-- custom js file link  -->
<script src="js/script.js"></script>
<?php include 'components/message.php'; ?>
</body>
</Html>
```

CHAPTER 9

TESTING

Testing:

Testing is a critical phase in software development that ensures the functionality, reliability, and performance of the application. The Real Estate Website underwent multiple levels of testing to validate its core features such as property listings, user registration, search and filter functionalities, and admin panel operations. Each module was tested individually (unit testing) and then integrated (integration testing) to ensure seamless interaction across components. System testing was conducted to verify the website's behaviour in a real-world environment. The results confirmed that the application meets its functional requirements and performs efficiently under expected load conditions.

9.1 Unit Testing

Each module was tested independently to ensure that all functions operated as expected in isolation.

Examples:

- User registration form validation
- Login authentication for different user roles (e.g., buyer, seller, admin)
- Property listing submission and data validation
- Test cases for invalid input, empty fields, and incorrect login credentials were also included and passed, ensuring robust input handling and system stability.

9.2 System Testing

System testing involved integrating all modules and testing them as a whole. The goal was to ensure that the interactions between modules worked flawlessly.

Scenarios Tested:

- Full flow: Register → Login → Browse Properties → Contact Seller/Agent
- Admin adding new property listings and managing user accounts

- Buyers and sellers accessing accurate property details and notifications

9.3 : Test cases

Customer Module:

Test Id	Test case	Input Description	Expected Output	Test Status
1	User Registration	Valid name, email,phone, and password	User account created and saved to database	Pass
2	Login Validation	Correct login credentials	Redirected to respective user dashboard	Pass
3	Property Inquiry Submission	User fills inquiry form for selected property	Inquiry saved, confirmation shown to user	Pass
4	Invalid Login	Incorrect password	Display error message	Pass
5	Empty Form Submission	Registration or inquiry form left blank	Show alert for missing/required fields	Pass

CONCLUSION

The Real Estate Website using PHP successfully addresses the key challenges faced by property buyers, sellers, and administrators in managing real estate transactions and listings online. The system automates and streamlines essential tasks such as user registration, property listing, inquiry management, and admin-level content control.

Key Achievements:

- **User-Friendly Interface:** Designed using HTML, CSS, and Bootstrap to ensure a smooth and intuitive experience for all users, including buyers, sellers, and agents.
- **Secure Backend:** PHP and MySQL integration ensures secure data handling and protects sensitive user information during transactions and communications.
- **Modular Structure:** Clearly defined modules for Admin, Buyer, Seller, and Agent ensure role-based access, better maintainability, and streamlined workflows.
- **Efficiency:** Reduces manual processes, enhances accuracy, and speeds up the real estate transaction cycle for all stakeholders.
- **Scalability:** The system is built with scalability in mind, allowing for easy expansion to support additional features, more users, or integration with third-party services in the future.

This project not only enhances the operational efficiency of real estate platforms but also delivers a modern and professional digital experience, making it a valuable tool for property management and marketing in today’s competitive market.

BIBLIOGRAPHY

REFERENCES

1. **PHP Manual**

   PHP Group. PHP: Hypertext Preprocessor Manual.

   https://www.php.net/manual/en/

2. **MySQL Documentation**

   Oracle Corporation. MySQL 8.0 Reference Manual.

   https://dev.mysql.com/doc/

3. **Bootstrap Framework**

   Twitter, Inc. Bootstrap Documentation.

   https://getbootstrap.com

4. **HTML and CSS Tutorials**

   W3Schools. HTML and CSS Guide.

   https://www.w3schools.com

5. **XAMPP/WAMP Server**

   Apache Friends. XAMPP for Local Web Development.

   https://www.apachefriends.org/

6. **Real Estate Web Development Resources**

   Mozilla Developer Network (MDN). Web Development Guides and Tutorials.

   https://developer.mozilla.org/en-US/docs/Learn

7. **JavaScript Documentation**

   Mozilla Developer Network (MDN). JavaScript Guide.

   https://developer.mozilla.org/en-US/docs/Web/JavaScript