A WEB BASED REAL ESTATE AND HOUSING ACCOMMODATIONS SYSTEM

A Project presented to the National University in partial fulfillment of the requirement for the degree of Bachelor of Science (Hon's) in Computer Science & Engineering.

Submitted By

Nusrat Zahan Tarin

Registration: 17502005064

Session: 2017-2018

Supervised By

Safrun Nesa Saira

Lecturer, Department of

Computer Science & Engineering,

Daffodil Institute of IT (DIIT)

Department of Computer Science & Engineering
Daffodil Institute of IT, Dhaka
Under National University, Bangladesh
Submission Date: August, 2023

APPROVAL

The Project “**A Web Based Real Estate And Housing Accommodation System**” is submitted to the Department of Computer Science & Engineering, Daffodil Institute of IT (DIIT), under National University of Bangladesh in partial fulfillment of the requirements for the degree of Bachelor of Science (Hon’s) in Computer Science and Engineering and approved as to its style and content.

Examiner

Examiner

Safrun Nesa Saira
Project Supervisor
Lecturer, Department of CSE,
Daffodil Institute of IT (DIIT)

Md. Imran Hossain
Head
Department of CSE,
Daffodil Institute of IT (DIIT)

DECLARATION

We affirm that the project work titled "**A Web Based Real Estate And Housing Accommodation System**" being submitted in partial fulfillment for the degree of B.Sc. (Hon's) in Computer Science & Engineering is the original work carried out by me. It has not formed the part of any other project work submitted for any degree or diploma, either in this or any other University.

Submitted By

Nusrat Zahan Tarin
Registration no: 17502005064
Session: 2017-2018

ACKNOWLEDGEMENTS

Despite our efforts, the success of this project depends largely on the encouragement and guidance of our mentors. We would like to take this opportunity to express our gratitude to the people who are playing vital role in the successful completion of this project.

Our sincere thanks to **Prof. Dr. Mohammed Shakhawat Hossain, Principal, DIIT** who has allowed us to work on this project and showed encouragement.

Our cordial & heartiest thanks to our Project Supervisor **Safrun Nesa Saira,** Lecturer, Department of Computer Science & Engineering, DIIT for her valuable guidance and support to meet the successful completion of our project and also his patronage and giving us an opportunity to undertake this Project.

We express our gratitude to **Md. Imran Hossain,** Head of Department, Computer Science & Engineering, DIIT for providing us the facilities to do the project successfully.

We express our gratitude to **Poly Bhoumik,** Senior Lecturer, Department of Computer Science & Engineering, DIIT for appreciating our goal.

We express our gratitude to our **Md. Saidur Rahman,** Senior lecturer & Co-ordinator, Computer Science & Engineering, DIIT for providing us the facilities to do the project successfully.

We extend our sincere thanks to our family & classmates for their constant support throughout this project.

Finally, we would be grateful to National University, Bangladesh and coordinators of the Bachelor of Science in Computer Science and Engineering degree program for giving us this opportunity to apply the knowledge that we have gained throughout the study of degree program.

ABSTRACT

The abstract of a web-based real estate and housing accommodation system encapsulates a dynamic online platform that revolutionizes property transactions and housing searches. This system offers a seamless digital environment connecting buyers, sellers, landlords, and tenants in the real estate sector. Through advanced search mechanisms, comprehensive property listings, virtual tours, and transparent information, the platform aims to enhance accessibility, trust, and efficiency in property transactions. By integrating secure payment gateways, interactive property visualization, and user-friendly interfaces, this abstract underscores the platform's potential to reshape the real estate industry, catering to diverse user needs and streamlining property management processes.

TABLE OF CONTENTS

Chapter 4: Requirement Analysis…………………………….(24-29)

Chapter 5: System Design………………………………………(30-41)

Chapter 6: Implementation……….……………………………(41- 46)

Lists of Figure

Chapter: 01
Introduction

1.1 Introduction

The project Real Estate and Housing Accommodations System which is based on the Real Estate Management System includes registration of customers, buyers, and users, storing their details in the system, and also a computerized billing system. The software has the facility to give a unique ID for every customer and stores the details of every customer and the user automatically. It includes a search facility to know the current status of each property. Users can search for the availability of properties and the details of a property using the id. The Real Estate Management System can be entered using a username and password. It is accessible either by an administrator or receptionist. Only they can add data to the database. The data can be retrieved easily. The interface is very user-friendly. The data is well protected for personal use and makes the data processing very fast.

The Real Estate Management System is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to the Real Estate business.

Real Estate Management System is designed for Real Estate businesses, to cover a wide range of properties and management processes. It is an integrated end-to-end Real Estate Management System that provides relevant information across the properties to support effective decision-making for users, buyers, and property agents, in a seamless flow.

Real Estate Management System is a software product suite designed to improve the quality and management of Real Estate Management in the areas of property analysis and activity-based costing. Real Estate Management System enables you to develop your organization and improve its effectiveness and quality of work. Managing the key processes efficiently is critical to the success of the Real Estate business helps you manage your processes

1.2 System Aims

The aim of this project is to create a web-based real estate and housing accommodation system that revolutionizes property transactions and housing searches. The system seeks to provide a seamless digital platform connecting buyers, sellers, landlords, and tenants in the real estate sector. Through advanced search mechanisms, comprehensive property listings, virtual tours, and transparent information, the project aims to enhance accessibility, trust, and efficiency in property transactions. The integration of secure payment gateways, interactive property visualization, and user-friendly interfaces underscores the project's goal of reshaping the real estate industry, catering to diverse user needs, and streamlining property management processes.

1.3 System Objectives

The objectives of the web-based real estate and housing accommodation system include:

1. **Efficient Property Transactions:** Create a platform that facilitates smooth and secure property transactions, benefiting both buyers and sellers.
2. **Comprehensive Property Listings:** Develop a system that offers detailed property information, aiding potential buyers and tenants in making informed decisions.
3. **Enhanced User Experience:** Design user-friendly interfaces, incorporating features like virtual property tours for an immersive experience.
4. **Transparent Information:** Ensure transparency by providing accurate property details, pricing, and legal information, fostering trust among users.
5. **Streamlined Property Management:** Offer tools for landlords and property managers to efficiently manage listings, inquiries, and tenant interactions.
6. **Secure Payment Processing:** Integrate secure payment gateways to enable safe and reliable financial transactions within the system.
7. **Diverse Property Search:** Implement advanced search options, allowing users to find properties based on specific criteria such as location, size, and amenities.
8. **Accessibility:** Create a platform accessible from various devices, enabling users to search for properties conveniently.
9. **User Interaction:** Foster user engagement through features like reviews, ratings, and feedback mechanisms, enhancing the community aspect of the system.
10. **Marketplace Efficiency:** Develop a platform that contributes to the efficiency of the real estate market by connecting buyers, sellers, landlords, and tenants seamlessly.

11. **Data Insights:** Utilize data analytics to provide insights into property trends, user preferences, and market demands.
12. **Innovation:** Incorporate innovative technologies such as virtual reality for property tours, setting a benchmark for modern real estate interactions.

1.4 Proposed System

The proposed system is a web-based real estate and housing accommodation platform designed to revolutionize property transactions and housing searches. This system aims to connect buyers, sellers, landlords, and tenants in a seamless digital environment. Key features of the proposed system include:

1. **Advanced Property Search:** A user-friendly interface with advanced search filters to help users find properties based on specific criteria such as location, size, price range, and amenities.
2. **Comprehensive Property Listings:** Detailed property listings with highquality images, property descriptions, pricing, and legal information for informed decision-making.
3. **User Profiles:** User accounts for buyers, sellers, landlords, and tenants, enabling them to manage listings, inquiries, and transactions.
4. **Property Management Tools:** Tools for landlords and property managers to easily manage their listings, schedule property tours, and communicate with potential tenants or buyers.
5. **Secure and Reliable:** Implementation of security measures to safeguard user data, privacy, and financial information.
6. **Community Engagement:** Features for users to interact, ask questions, and share their experiences, creating a community-driven environment.
7. **Innovation:** Integration of innovative technologies like virtual reality to enhance the property browsing experience.

1.5 System Feature

The system features of the proposed web-based real estate and housing accommodation platform include:

1. **Advanced Search:** Users can perform detailed property searches based on location, size, price, and amenities, ensuring precise results.
2. **Comprehensive Listings:** Detailed property listings with high-quality images, descriptions, pricing, and legal information for informed decision-making.
3. **Virtual Tours:** Interactive virtual property tours offer a realistic and immersive property viewing experience without physical visits.
4. **Secure Transactions:** Integration of secure payment gateways ensures safe financial transactions for property purchases and rentals.
5. **User Profiles:** Individual accounts for buyers, sellers, landlords, and tenants, enabling personalized management of listings, inquiries, and interactions.
6. **Reviews and Ratings:** Transparent review and rating system that builds trust by showcasing user experiences and property reliability.
7. **Property Management Tools:** Landlords and property managers can easily manage listings, schedule property tours, and communicate with potential clients.
8. **Data Insights:** Utilization of data analytics to provide trends, preferences, and market insights for users and stakeholders.
9. **Security Measures:** Implementation of robust security protocols to protect user data and financial information.
10. **Community Engagement:** Interactive features allow users to interact, ask questions, and share experiences, fostering a sense of community.
11. **Innovative Technologies:** Integration of modern tools like virtual reality for immersive property exploration.

1.6 Why We Select this System

This project, centered around a web-based real estate and housing accommodation system, has been chosen for its immense potential to address critical challenges in the real estate industry. The market demand for streamlined property transactions and the ongoing digital transformation of the sector underscore the relevance of this platform. By offering advanced search features, virtual property tours, secure payment gateways, and transparent property information, the project aims to significantly enhance user experience and efficiency in the real estate market. Additionally, the system's focus on security, data insights, and community engagement positions it as a forward-looking solution in an industry ripe for innovation. The unique combination of convenience, transparency, and technological innovation makes this project a strategic choice with the potential to reshape the way real estate transactions are conducted.

1.7 Advantages of the Proposed System

The first advantage of the proposed system is that its front end provides easy and précised information to the user to interact with the system and hence it is faster to complete the work. The next advantage of the present system is its faster capability with the database. Even the database also provides its inbuilt features to maintain the records. It also provides security to the records from the system itself and also from the database itself. Hence it prevents the database conjunction. The present system provides better screening of the record, another advantage of the proposed system is its faster capability to search the records from the database.

Chapter: 02

BACKGROUND STUDY

2.1 Background Study

The background study for this project involves a thorough examination of the current real estate and housing sector. This analysis includes identifying challenges like lack of transparency and time-consuming processes in property transactions. It also considers trends such as the shift towards digital solutions and changing user expectations for more efficient and accessible property searches. The study evaluates existing online real estate platforms and explores emerging technologies like virtual reality and secure payment gateways. User feedback and regulatory considerations are incorporated to assess the project's feasibility. Overall, the background study informs the project's development by addressing industry challenges and aligning with user demands.

2.2 Limitation of existing System

The limitations of this system encompass potential challenges and constraints that could impact its functionality and effectiveness:

1. **Limited Data Accuracy:** Reliability of property information heavily relies on data input by users, which could lead to inaccuracies and outdated listings.
2. **Geographic Coverage:** The system's effectiveness could vary in regions with limited technology adoption or incomplete property data coverage.
3. **User Adoption:** Convincing traditional users to shift to an online platform might face resistance, affecting the platform's initial adoption rate.
4. **Technical Complexity:** Integrating advanced features like virtual property tours or secure payment gateways might introduce technical complexities and potential glitches.
5. **Privacy Concerns:** Handling user data and financial transactions requires stringent data protection measures to address privacy concerns.
6. **Trust Building:** Establishing trust among users for online property transactions and reviews can be challenging due to security and reliability concerns.
7. **Regulatory Compliance:** Adhering to diverse real estate regulations across regions requires careful consideration and legal expertise.
8. **Initial Investment:** Developing and launching a feature-rich platform demands substantial upfront investment in terms of time, resources, and technology.

9. **Competitive Landscape:** The real estate tech market can be competitive, necessitating innovative strategies to stand out and attract users.

2.3 Solution of the Problem

To mitigate the challenges associated with the Multi-Vendor Service Marketplace and Service Finder project, a set of solutions can be implemented to ensure a successful and sustainable platform:

- User Trust: Implement a robust review and rating system with clear guidelines for users to leave authentic and constructive feedback. Introduce trust badges or verification markers for providers who undergo a thorough verification process.
- Security and Privacy: Implement strong encryption protocols to safeguard user data and financial transactions. Regularly audit and update security measures to stay ahead of potential threats.
- Competitive Landscape: Focus on a niche or underserved market segment to establish a stronghold before expanding to broader markets.
- Local vs. Global Dynamics: Consider a phased approach where the platform starts in a specific region and gradually expands to other countries, leveraging initial success as a springboard. Offer localized content, customer support, and user interfaces to cater to specific markets.
- Payment Integration: Partner with reputable payment gateway providers that offer multiple payment options, currencies, and secure processing. Ensure compliance with regional financial regulations and standards.
- Localization and Language: Offer multilingual support for the platform, allowing users to interact in their preferred language.
- Regulatory Compliance: Consult legal experts to navigate complex regulatory requirements related to financial transactions, data privacy, and industryspecific standards.
- Market Saturation: Focus on a specific niche or underserved category of services to establish a strong foothold and build a loyal user base.

2.4 Scope of the project

The Online Real Estate is a Website designed to handle the basic tasks of the Real Estate. Hence the main function of this system or company is to convert the manual trading service of properties / flats into the better quality service that customers want. The functionality of this system is like handling the properties or property management and handles the main function that mentioned above. The functionality of this system is described as follows:

- Allows the customers to view or get details of various properties/flats available.
- Add the properties/flats to the stock.
- Allow admin to check the availability of properties/flats.
- Customers can view all properties/flats online. · Displays the details like customers details, rates of flats, locations, remarks details etc. This is all the functionality of the Online Real Estate Website.

2.5 Feasibility Study

Feasibility study is an important step in the software development process. It involves assessing whether a proposed project is technically, economically, and operationally feasible.

2.5.1 Technical Feasibility Study

Here's an overview of the technical feasibility:

- **Accessible Technologies:** Utilization of established web development technologies, such as HTML, CSS, JavaScript, and PHP, for creating the platform.
- **Database Management:** Implementation of database systems like MySQL for efficient storage and retrieval of property information.
- **Responsive Design:** Development of a responsive user interface to ensure optimal user experience across various devices.
- **Secure Payment Integration:** Integration of secure payment gateways, ensuring safe financial transactions for property purchases and rentals.

- **Virtual Property Tours:** Incorporation of virtual reality technology to provide immersive property tours, enhancing user engagement.
- **User Authentication:** Implementation of robust user authentication and authorization mechanisms to safeguard user data.
- **Scalability Planning:** Designing the system architecture to handle increasing user traffic and property listings without compromising performance.
- **Data Accuracy:** Ensuring data accuracy through validation mechanisms to provide reliable property details to users.
- **Technical Expertise:** The availability of skilled developers proficient in web development and the technologies required for system implementation.

2.5.2 Operational Feasibility Study

The operational feasibility of implementing a web-based real estate and housing accommodation system examines whether the proposed solution aligns with the organization's resources, capabilities, and objectives. It evaluates the practicality of introducing the system within the current operational framework and considers factors such as user acceptance, training requirements, and potential resistance to change. This assessment involves collaboration with stakeholders to understand their needs, expectations, and concerns. Additionally, it examines how the system would fit into the existing workflow and if any modifications are needed to integrate it seamlessly. Operational feasibility also considers the availability of skilled personnel to manage and maintain the system and identifies any potential challenges that could arise during its deployment. Ultimately, this assessment ensures that the proposed system is not only technically viable but also a practical and beneficial addition to the organization's operational landscape.

Chapter: 03
PROPOSED SYSTEM

3.1 Methodology

Methodology is a set of rules, methods and principles that make a guideline to the developers to develop in a systematic way. In software development, there are methods. Every methodology is different by steps and systematic way. There are some methodologies

They are given in the below:

a) Agile Software Development

b) V Shape Model

c) Feature Driven Development (FDD)

d) Rapid Application Development (RAD)

e) Rational Unified Process (RUP)

f) System Development Life Cycle (SDLC)

g) Waterfall Model

Though there are many methodologies in system development, we use Methodologies for our flexibility and apply a level of pragmatism.

3.2 Why have We Chosen Agile?

The agile method anticipates change and allows for much more flexibility than traditional methods. Clients can make small objective changes without huge amendments to the budget or schedule. This Method saves money and time because the tests and approves the product at each step of development.

3.3 Agile Method

Agile software development describes a set of principles for software development under which requirements and solutions evolve through the collaborative effort of selforganizing cross-functional teams. It advocates adaptive planning, evolutionary development, early delivery, and continuous improvement, and it encourages rapid and flexible response to change. These principles support the definition and continuing evolution of many software development methods.

If we relate with the term 'agile' we can easily understand where it should be used. It can be implemented in all types of projects but its essence can only be extracted if we use it on bigger and complex projects. We can use agile when we can actually implement it for the success of the project or the nature of the project requires it. Agile should not be used just as a run away from a waterfall.

We can use agile where collaboration is highly important for the success of the project, where we have long term goals and no bound on the requirements, and agile can be well utilized where we have great power of either utilizing time as much as we want or resource as much as we want.

3.4 When to Use Agile Method

Agile development model is also a type of Incremental model. Software is developed in incremental, rapid cycles. This results in small incremental releases with each release building on previous functionality. Each release is thoroughly tested to ensure software quality is maintained. It is used for time-critical applications. Extreme Programming is currently one of the most well-known agile development life cycle models.

3.5 Advantages of Agile Method

Some advantages of using the Agile methodology for software development include:

- Flexibility: Agile methodology is highly flexible and can accommodate changing requirements, even late in the development process.
- Customer satisfaction: The Agile methodology emphasizes customer collaboration and continuous feedback, which helps ensure the product meets the customer's needs and expectations.
- Rapid delivery: The Agile methodology promotes incremental development, allowing developers to deliver working software quickly and frequently.

- Improved quality: Agile methodology focuses on continuous testing and feedback, which can help identify and fix issues early on in the development process.

- Efficient use of resources: Agile methodology allows for prioritization of features and functionality, ensuring that resources are allocated to the most important areas of the project.

- Collaboration and teamwork: Agile methodology emphasizes regular communication and collaboration between team members, promoting a cohesive and productive team environment.

3.6 Disadvantages of Agile Method

Some potential disadvantages of the Agile methodology are:

- Lack of predictability: Due to the flexible and adaptive nature of Agile, it can be difficult to predict exactly when a project will be completed or what the final outcome will be.

- Limited documentation: Agile focuses more on working software over comprehensive documentation, which can make it difficult to track changes or provide detailed documentation.

- Requires experienced team members: Agile relies heavily on collaboration and communication between team members, which can be challenging if team members are not experienced in working in this type of environment.

- Scope creep: Agile methodology is based on constantly evolving requirements, which can lead to scope creep or an increase in project scope without proper management.
- Lack of emphasis on design: Agile methodology often places more emphasis on working software than on design and architecture, which can lead to technical debt and poor code quality if not managed properly.

Chapter: 04
REQUIREMENT ANALYSIS

4.1 Requirements Introduction

To be used efficiently, all computer software needs certain hardware components or the other software resources to be present on a computer. These prerequisites are known as(computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

4.2 Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatibility and sometimes incompatible hardware devices for a particular operating system or application. The following subsections discuss the various aspects of hardware requirements.

4.3 Hardware Requirements for this project

The hardware listed by no means a minimum requirement to run the system, but rather a base limit for running the system smoothly and comfortably. This is also considering the potential amount of traffic that may go through the server.

- Intel core i3 2.8 GHz processor
- 4 GB of RAM
- 120 GB of disk space
- Keyboard
- Mouse
- Monitor
- Other basic computer components

4.4 Software Requirements for our project

Real Estate Management System (REMS) that includes property listings, tenant management, financial tracking, and reporting. It should have a user-friendly interface, strong security, scalability, integration capabilities, compliance with regulations, rigorous testing, and consistent maintenance. This REMS will enhance real estate operations and transactions for efficient management.

- HTML
- CSS
- JAVASCRIPT
- PHP
- LARAVEL
- MYSQL
- DATABASE

4.5 Frontend Language

The choice of frontend programming languages for a multi-vendor service marketplace and service finder platform depends on factors such as your team's expertise, the technology stack you're using, and the specific requirements of your project.

4.5.1 HTML

HTML (Hyper Text Markup Language) is a markup language used to create and structure content on the web. It is the standard language used to build websites and web applications. HTML consists of a series of elements, which can be used to define the structure and content of a web page. These elements can include headings, paragraphs, images, links, forms, and more. HTML is based on a syntax of tags and attributes, which describe how the content should be displayed and interacted with by users.

Creating the entire HTML structure for a multi-vendor service marketplace and service finder is quite extensive and beyond the scope of a single response. However, I can provide you with a basic example of the HTML structure for a simplified landing page of such a platform. Keep in mind that a real platform would require multiple pages, dynamic components, and more advanced features.

4.5.2 CSS

CSS (Cascading Style Sheets) is a styling language used to define the visual appearance and layout of web pages written in HTML. It provides a way to separate the presentation of a document from its content, allowing developers to define the styles for all the elements of a website in one place. CSS works by defining styles for HTML elements using selectors and properties. Selectors are used to target specific HTML elements, while properties are used to define the style or layout of those elements. CSS can be used to specify a wide range of styles, including font styles, colors, margins, padding, borders, backgrounds, and more.

Creating a complete CSS (Cascading Style Sheets) for a multi-vendor service marketplace and service finder platform is quite an extensive task due to the complexity and diversity of design elements. However, I can provide you with a basic example of CSS styles that you might use as a starting point for your platform. Keep in mind that this is a very simplified example, and in a real-world scenario, you'd need to design a more comprehensive and visually appealing stylesheet.

4.5.3 JavaScript

JavaScript is a programming language that is primarily used for creating dynamic web pages and applications. JavaScript is a high-level, interpreted language that is often referred to as a scripting language. It is primarily used to add interactive functionality to web pages and is executed by a web browser. JavaScript can also be used on the server-side. JavaScript code can be embedded directly into HTML documents, or it can be stored in separate js files and linked to the HTML file. It supports a wide variety of programming paradigms, including object-oriented, functional, and procedural programming.

4.6 Backend Language

The choice of backend programming language for a multi-vendor service marketplace and service finder platform depends on various factors such as your team's expertise, project requirements, scalability needs, and the technology ecosystem you plan to use.

4.6.1 PHP

PHP is a server-side scripting language that is used to create dynamic web pages and web applications. PHP is an open-source language, which means that it is free to use and modify. It is often used in combination with a database such as MySQL to create dynamic web pages that can store and retrieve data. PHP code is executed on the server before the web page is sent to the client's web browser. This allows for dynamic content to be generated based on user input or other variables.

4.6.2 LARAVEL

Laravel is an open-source PHP web application framework that's known for its elegant syntax, developer-friendly features, and robust set of tools for building web applications and APIs. It was created by Taylor Otwell and was first released in 2011. Laravel follows the Model-View-Controller (MVC) architectural pattern and offers a wide range of features that streamline the development process and promote clean, maintainable code.

4.6.3 MYSQL

MySQL is an open-source relational database management system that allows you to efficiently store, manage, and retrieve structured data. It provides a structured way to organize data into tables with rows and columns, and it uses a query language called SQL for interacting with the data. MySQL is widely used for various applications, from small personal projects to large-scale enterprise systems, due to its reliability, speed, and scalability. Creating a complete MySQL database schema for a multi-vendor service marketplace and service finder is quite comprehensive and beyond the scope of a single response. However, I can provide you with an outline of how you might structure the MySQL database for such a platform. Please note that this is a simplified example, and in a real-world scenario, you would need to consider many more details and relationships.

4.6.4 Database

A database is a structured collection of data that is organized and stored in a way that allows for efficient access, retrieval, and manipulation. The database structure for a multi-vendor service marketplace and service finder platform can be quite complex, as it needs to manage various aspects of user profiles, services, bookings, reviews, and more. It’s a simplified example of how you might structure the database using a relational database management system like MySQL. Keep in mind that this is a basic illustration, and a real-world application would likely have additional tables and relationships.

CHAPTER 5

SYSTEM DESIGN

5.1 Use case Diagram

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases Use case diagrams are formally included in two modeling languages defined by the OMG the unified modeling language (UML) and the systems modeling language.

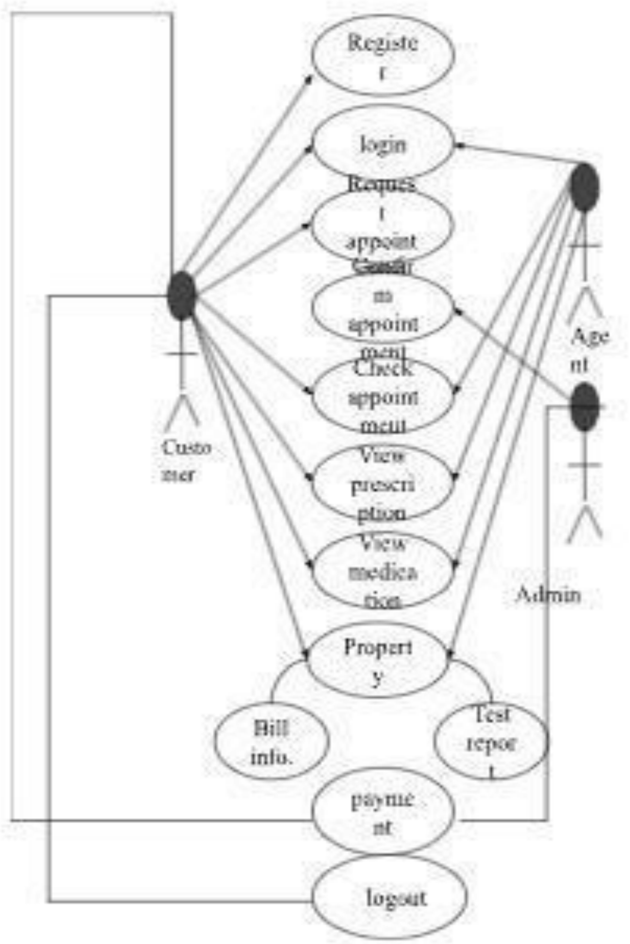


Figure 5.1: Use Case Diagram

5.2 Flow Chart

Flowchart is a graphical representation of an algorithm. Programmers often use it as a programplanning tool to solve a problem. It makes use of symbols that are connected among them to indicate the flow of information and processing. The process of drawing a flowchart for an algorithm is known as “flowcharting”. Basic Symbols used in Flowchart Designs:

Terminal: The oval symbol indicates Start, Stop and Halt in a program’s logic flow. A pause or halt is generally used in a program logic under some error conditions. Terminal is the first and last symbols in the flowchart.

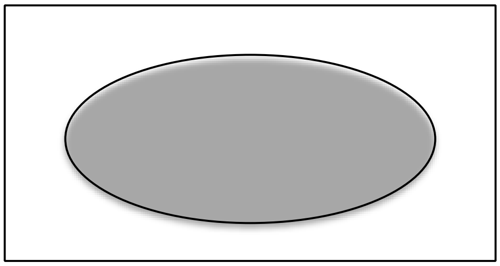


Figure : Terminal

Input/Output: A parallelogram denotes any function of input/output type. Program instructions that take input from input devices and display output on output devices are indicated with parallelogram in a flowchart.

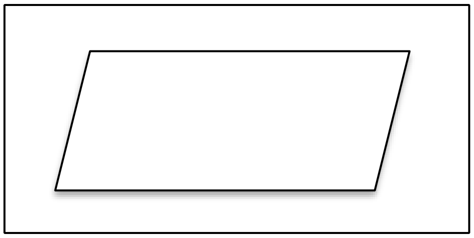


Figure : Input/output

Processing: A box represents arithmetic instructions. All arithmetic processes such as adding, subtracting, multiplication and division are indicated by action or process symbol.

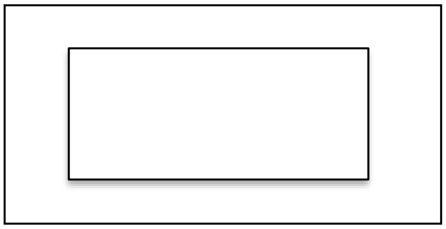


Figure ; Processing

Decision Diamond symbol represents a decision point. Decision based operations such as yes/no question or true/false are indicated by diamond in flowchart.

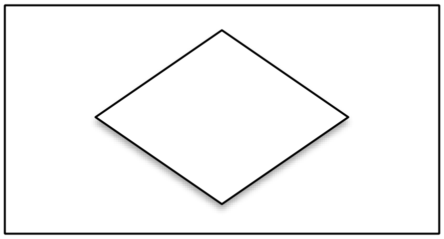


Figure ; Decision

Connectors: Whenever flowchart becomes complex or it spreads over more than one page, it is useful to use connectors to avoid any confusions. It is represented by a circle.

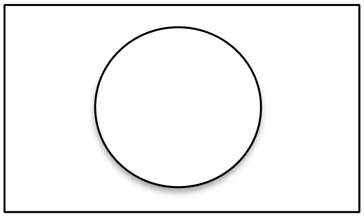


Figure : Connectors

Flow lines: Flow lines indicate the exact sequence in which instructions are executed. Arrows represent the direction of flow of control and relationship among different symbols of flowchart.

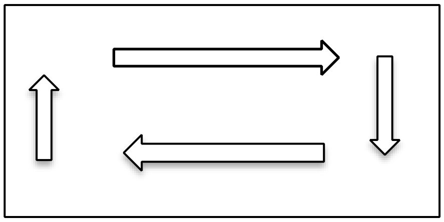


Figure : Flow lines

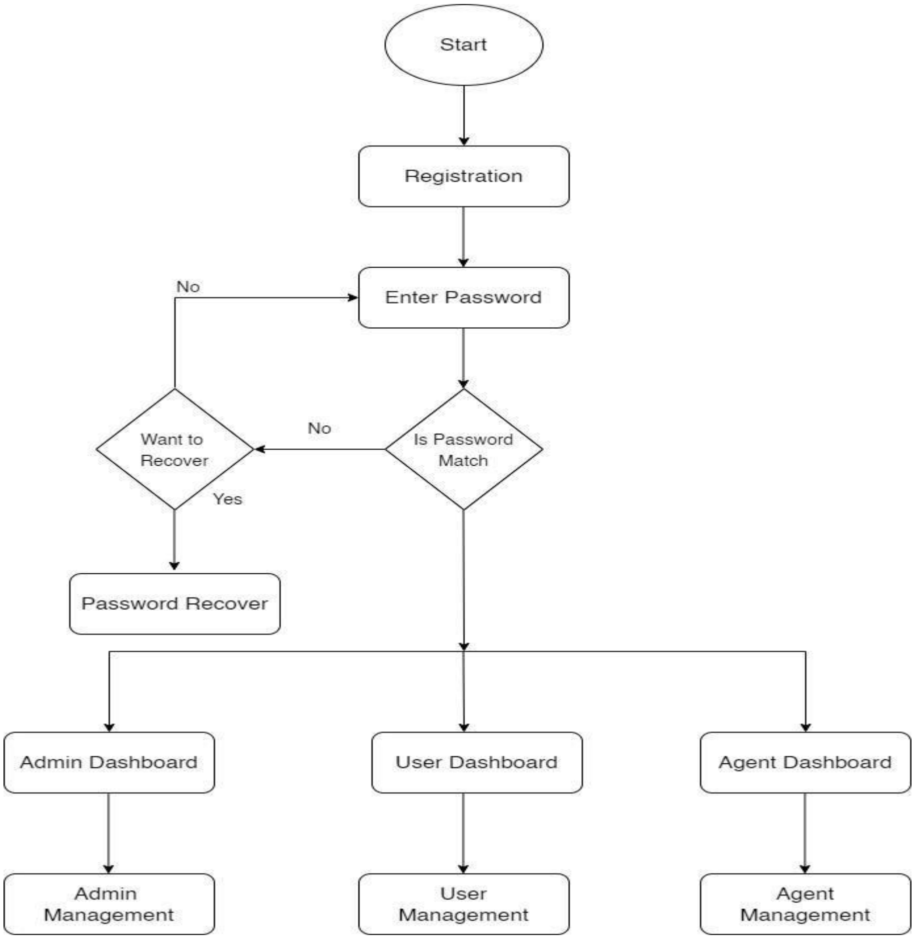


Figure 5.2: Flow Chart

5.3 Data Flow Diagram (DFD)

A data flow diagram (DFD) is a diagram that describes the flow of data and the process that changes data throughout a system. It's a structured analysis and design tool that can be used for flow charting in place of or in association with information.

The DFD reviews the current system, prepares input and output specification, specifies the implementation plan etc. Using any convention's DFD rules or guidelines, the symbols depict the four components of data flow diagrams.

External Entity: An outside system that sends or receives data, communicating with the system being diagrammed. They are the sources and destinations of information entering or leaving the system. They might be an outside organization or person, a computer system or a business system. They are also known as terminators, sources and sinks or actors. They are typically drawn on the edges of the diagram.

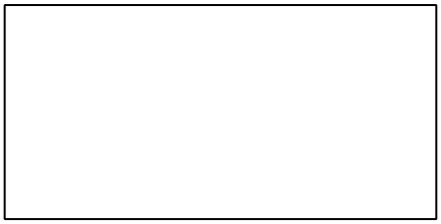


Figure : External Entity

Process: Any process that changes the data, producing an output. It might perform computations, or sort data based on logic, or direct the data flow based on business rules. A short label is used to describe the process, such as "Submit payment."

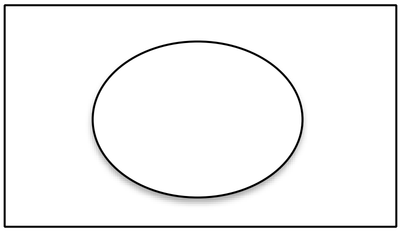


Figure : Process

Data Store: Files or repositories that hold information for later use, such as a database table or a membership form. Each data store receives a simple label, such as “Orders”.

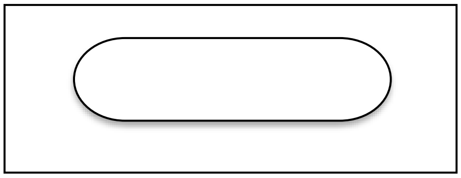


Figure : Data Store

Data Flow: The route that data takes between the external entities, processes and data stores. It portrays the interface between the other components and is shown with arrows, typically labeled with a short data name, like “Billing details."

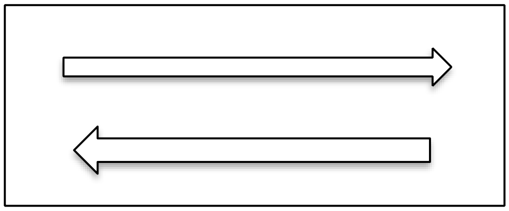


Figure : Data flow

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction. Typically, an event needs to be achieved by some operations, particularly where the operation is intended to achieve a number of different things that require coordination, or how the events in a single use case relate to one another, in particular, use cases where activities may overlap and require coordination. It is also suitable for modeling how a collection of use cases coordinate to represent business workflows.

5.3.1 DFD Level 0 Diagram

A Level 0 Data Flow Diagram (DFD) is a high-level representation of a system or process that shows the main external entities involved and the major data flows between them. It provides an overview of the entire system, showing how different entities interact with each other in terms of data exchange. In a Level 0 DFD, the system's processes are represented as bubbles, and the external entities and data flows are connected to these processes.

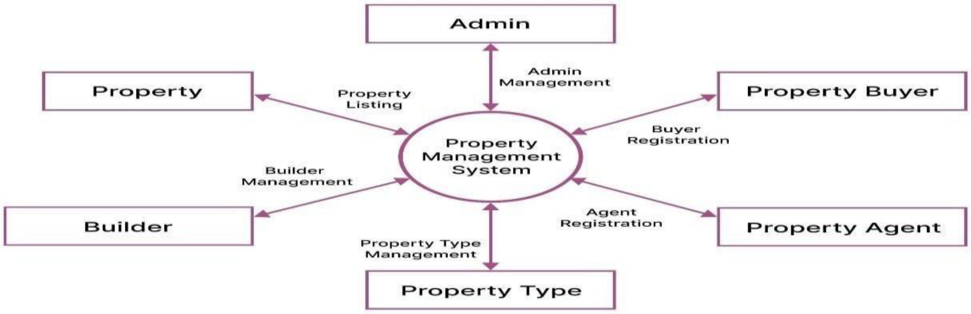


Figure 5.3.1: DFD Level 0 Diagram

5.3.2 DFD Level 1 Diagram

A Level 1 Data Flow Diagram (DFD) provides a more detailed view of the processes and data flows within a system that was initially depicted in the Level 0 DFD. In a Level 1 DFD, each process from the Level 0 DFD is decomposed into sub-processes, and additional data flows and data stores are shown to represent more specific interactions and data handling.

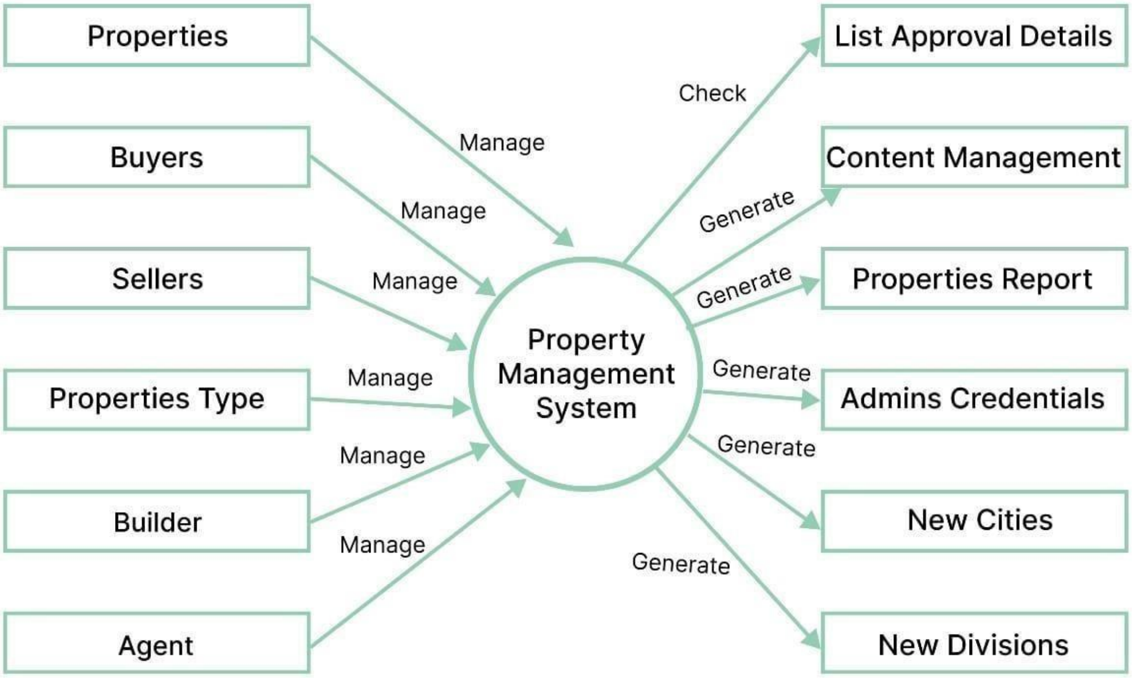


Figure 5.3.2: DFD Level 1 Diagram

5.3.3 DFD level 2 Diagram

Level 2 Data Flow Diagram (DFD) involves further decomposing the processes from the Level 1 DFD into more detailed sub-processes and data flows.

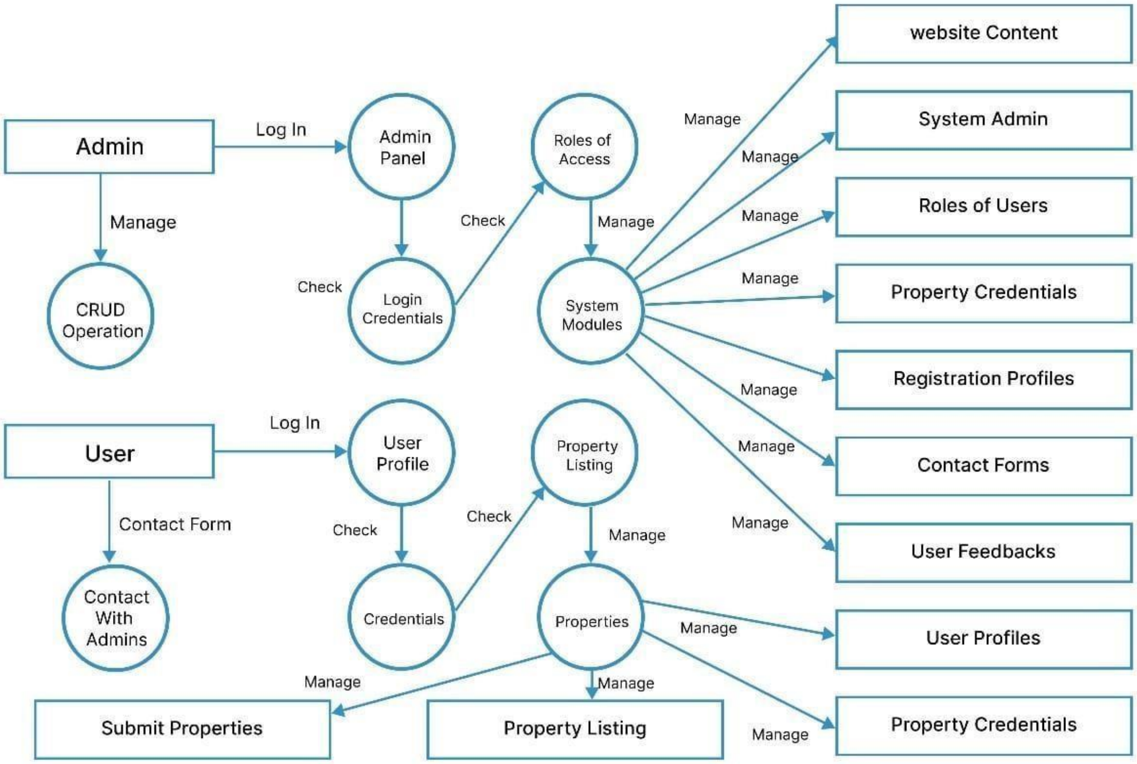


Figure 5.3.3: DFD Level 2 Diagram

5.4 ER Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relations

An ER diagram is a means of visualizing how the information a system produces is related. There are five main components of an ERD:

Entities, which are represented by rectangles. An entity is an object or concept about which you want to store information.

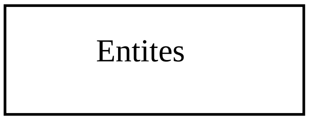


Figure : Entity

A weak entity is an entity that must be defined by a foreign key relationship with another entity as it cannot be uniquely identified by its own attributes alone.

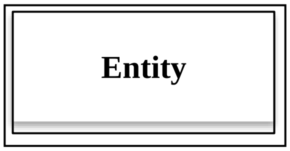


Figure : Entity

Actions, which are represented by diamond shapes, show how two entities share information in the database. In some cases, entities can be self-linked. For example, employees can supervise other employees.

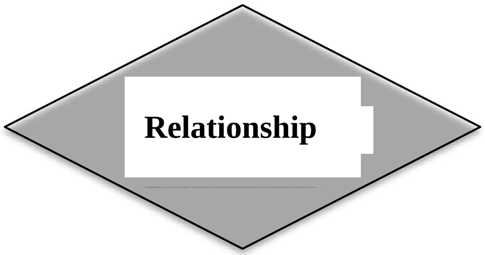


Figure : Relationship

Attributes, which are represented by ovals. A key attribute is the unique, distinguishing characteristic of the entity. For example, an employee's social security number might be the employee's key attribute.

Attributes

Figure: Attributes

A multivalued attribute can have more than one value. For example, an employee entity can have multiple skill values.

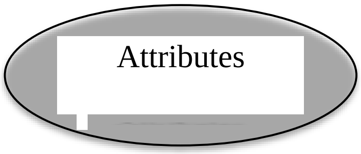


Figure : Attributes

A derived attribute is based on another attribute. For example, an employee's monthly salary is based on the employee's annual salary

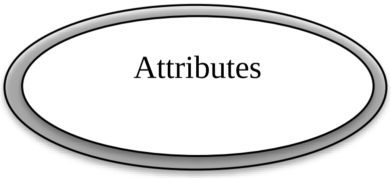


Figure : Attributes

An entity is an object or concept about which you want to store information. Here is our ER diagram is given bellow,

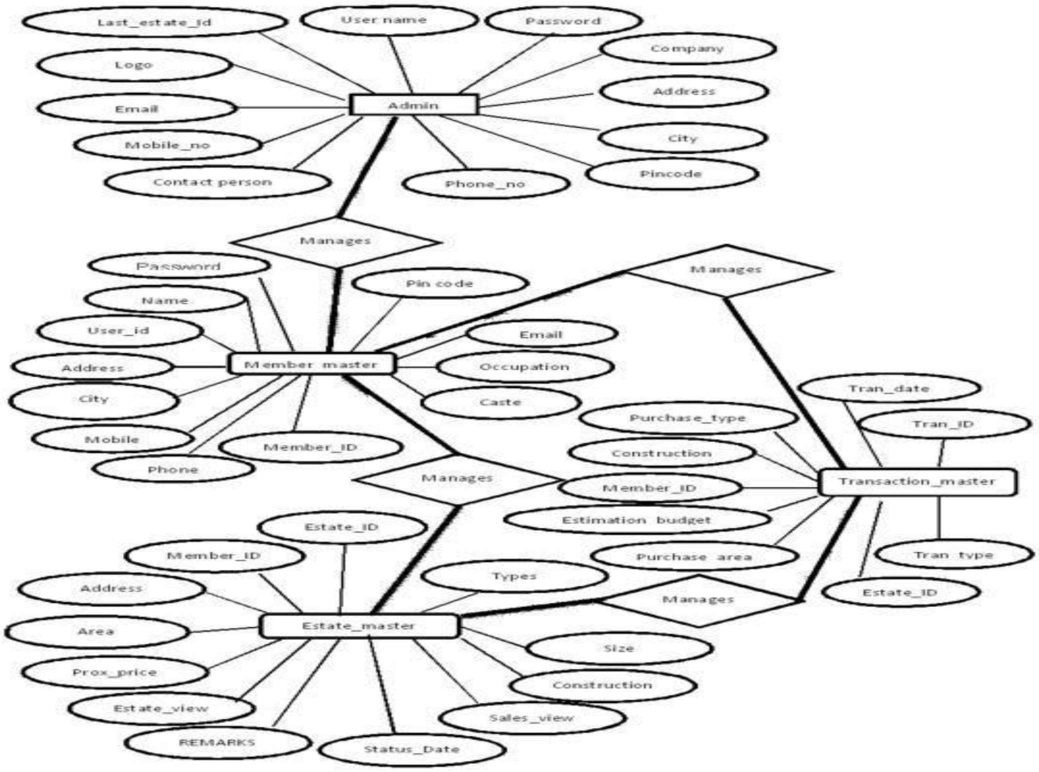


Figure 5.4: ER Diagram

Chapter: 06
IMPLEMENTATION

6.1 Introduction

The implementation phase is a crucial step in turning the conceptual vision into a tangible reality. Building upon the planning and design stages, this phase focuses on translating the outlined features into a functional web-based platform. The chapter details the strategies, methodologies, and steps used to develop, integrate, and test system components. By adhering to best practices and utilizing chosen technologies, the goal is to create a robust and user-friendly real estate and housing accommodation platform. This chapter highlights the technical intricacies, development approaches, and quality assurance efforts that contribute to achieving a fully functional system.

6.2 Home Page

The home page of this system is the starting point for users, designed to be intuitive and engaging. It features an advanced search bar for quick property searches and highlights featured listings. Users can explore property categories, virtual tours, and reviews. The personalized dashboard offers easy access to saved searches and recently viewed properties. Overall, the home page offers a dynamic entry into the system's functionalities and a seamless exploration of real estate options.

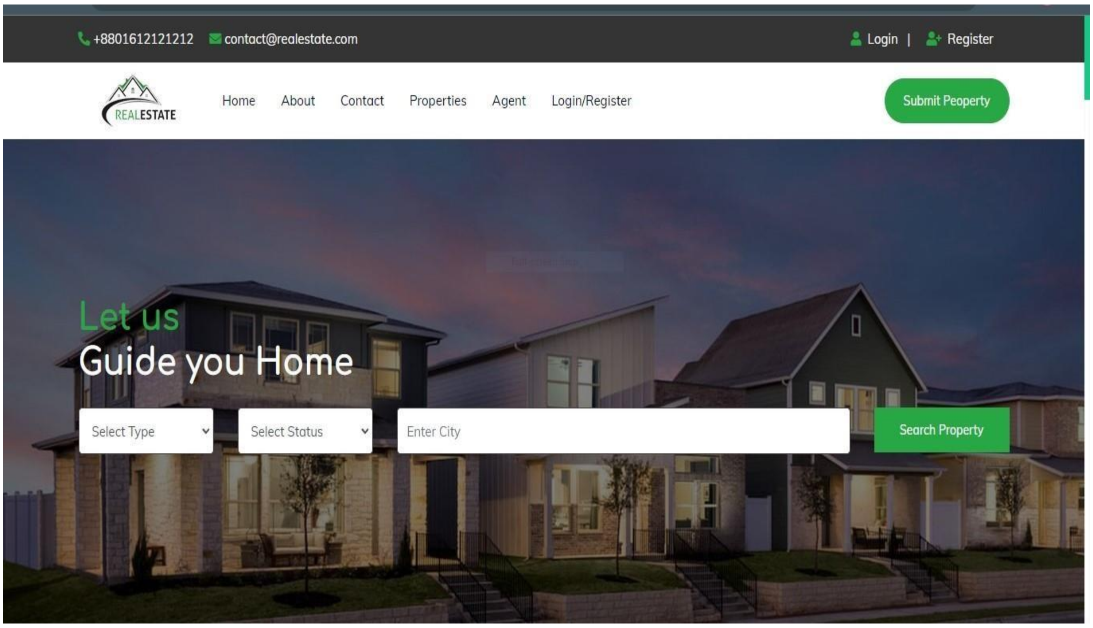
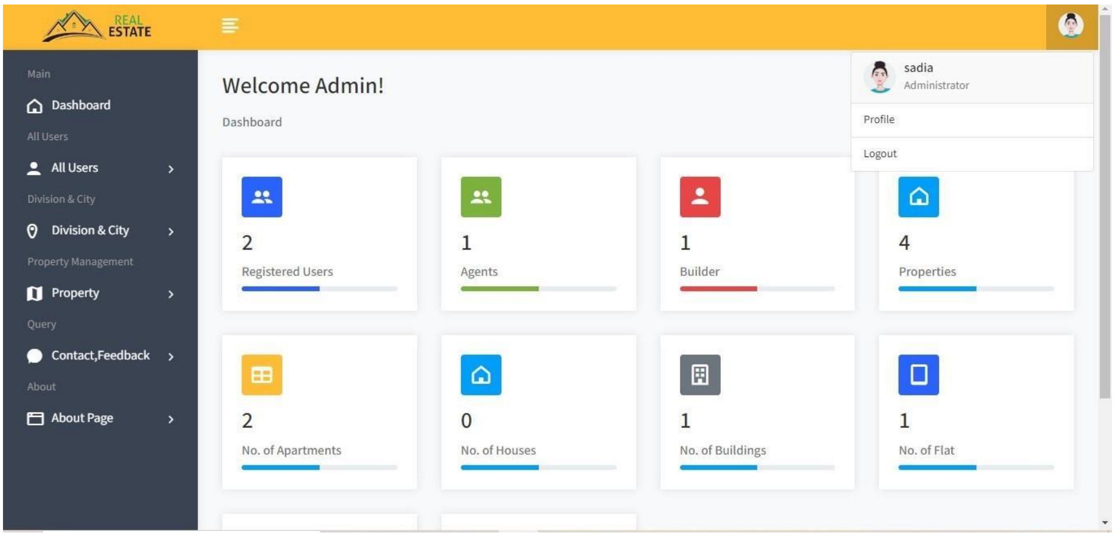


Figure 6.2: Home page

6.3 Dashboard

The dashboard is the user's command center, offering a concise overview of their platform activities. It facilitates property management, displays personalized recommendations, and enables direct communication. Integrated analytics provide insights, enhancing the overall real estate experience.

6.3.1 Dashboard (Admin)



Welcome Admin!

Dashboard

2 Registered Users

1 Agents

1 Builder

4 Properties

2 No. of Apartments

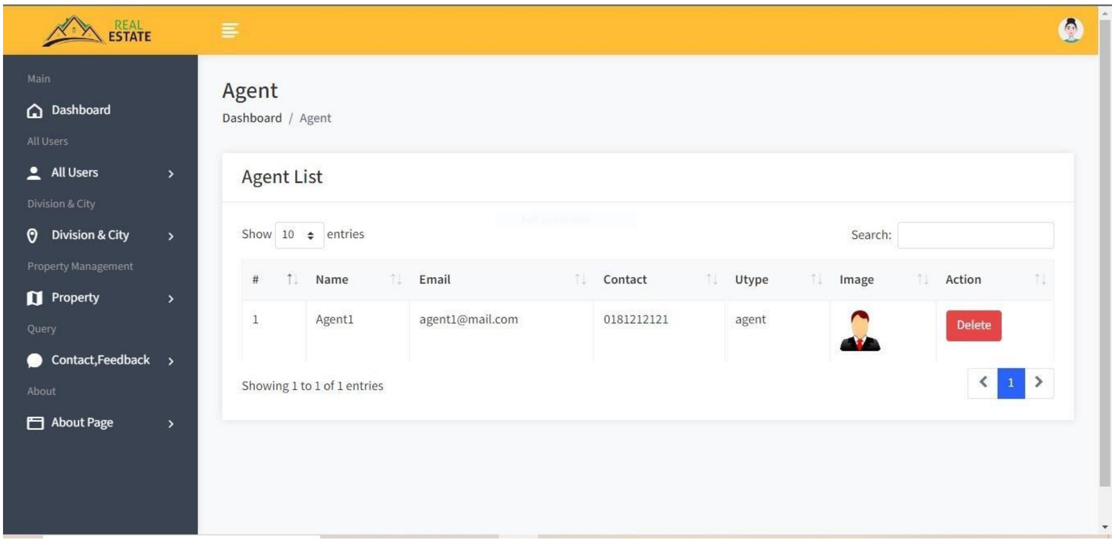
0 No. of Houses

1 No. of Buildings

1 No. of Flat

Figure 6.3.1: Admin dashboard

6.3.2 Dashboard (Agent)



Agent

Dashboard / Agent

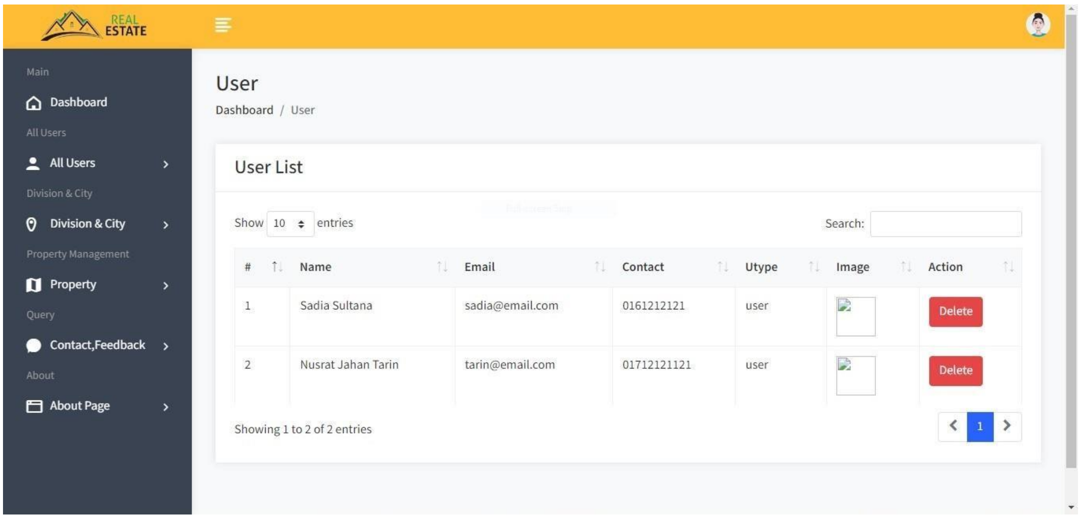
Agent List

#	Name	Email	Contact	Utype	Image	Action
1	Agent1	agent1@mail.com	0181212121	agent		Delete

Showing 1 to 1 of 1 entries

Figure 6.3.2: Dashboard (Agent)

6.3.3 Dashboard (User)



User

Dashboard / User

User List

Show 10 entries

Search:

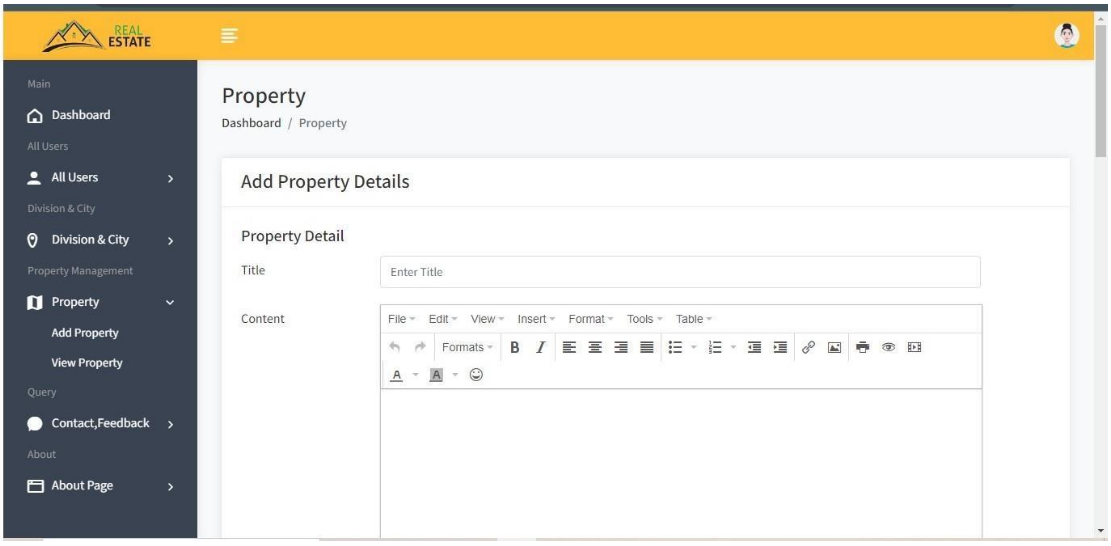
#	Name	Email	Contact	Utype	Image	Action
1	Sadia Sultana	sadia@email.com	0161212121	user		Delete
2	Nusrat Jahan Tarin	tarin@email.com	01712121121	user		Delete

Showing 1 to 2 of 2 entries

Figure 6.3.3: Dashboard (User)

6.4 Property Add Details

The "Add Property" feature simplifies property listing. Users input details, upload images, and set pricing. Virtual tours can also be added. The secure interface guides users through the process, resulting in an appealing and informative listing for potential buyers or tenants.



Property

Dashboard / Property

Add Property Details

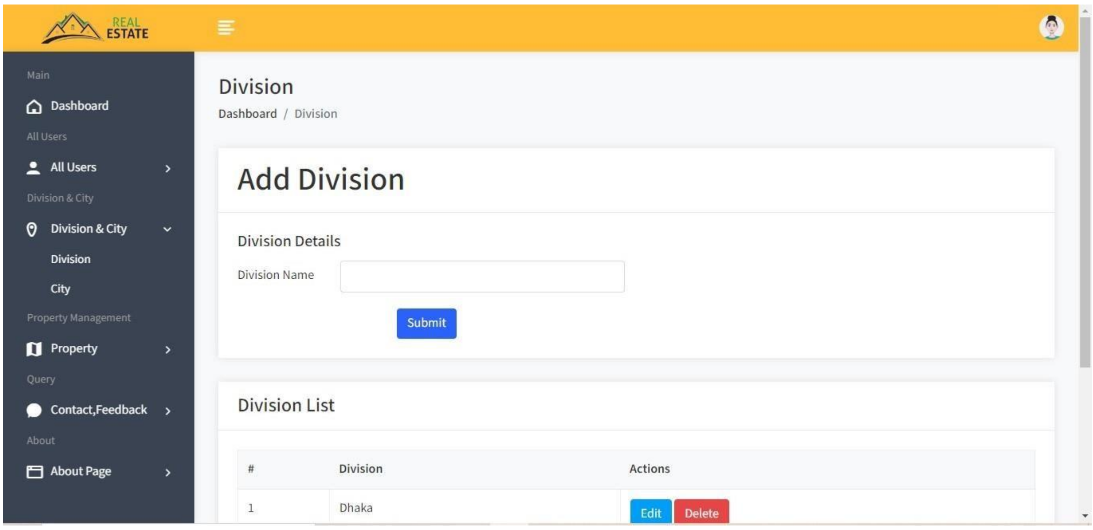
Property Detail

Title: Enter Title

Content

Figure 6.4: Add property details

6.5 Division Add



Division

Dashboard / Division

Add Division

Division Details

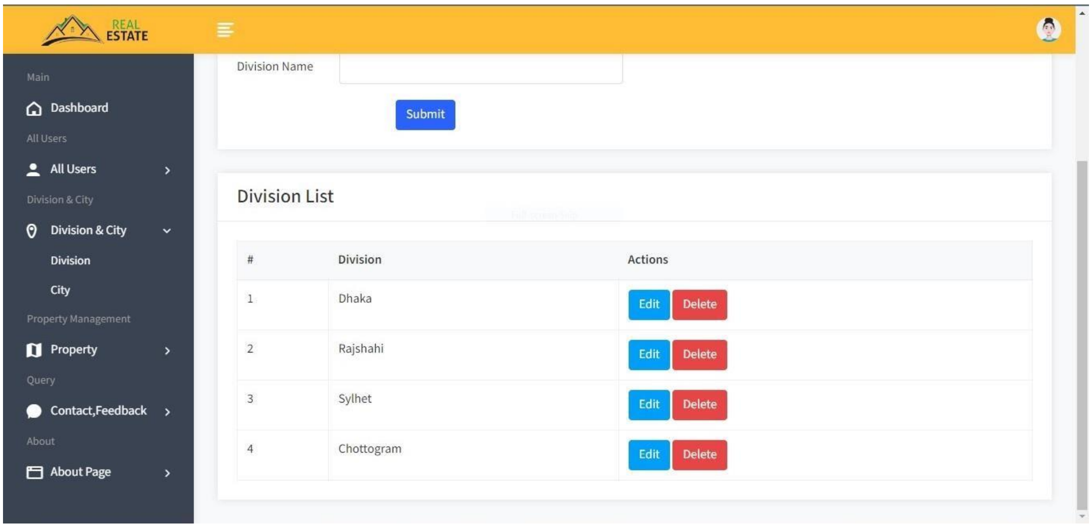
Division Name

Division List

#	Division	Actions
1	Dhaka	Edit Delete

Figure 6.5: Add Division

6.6 Division List



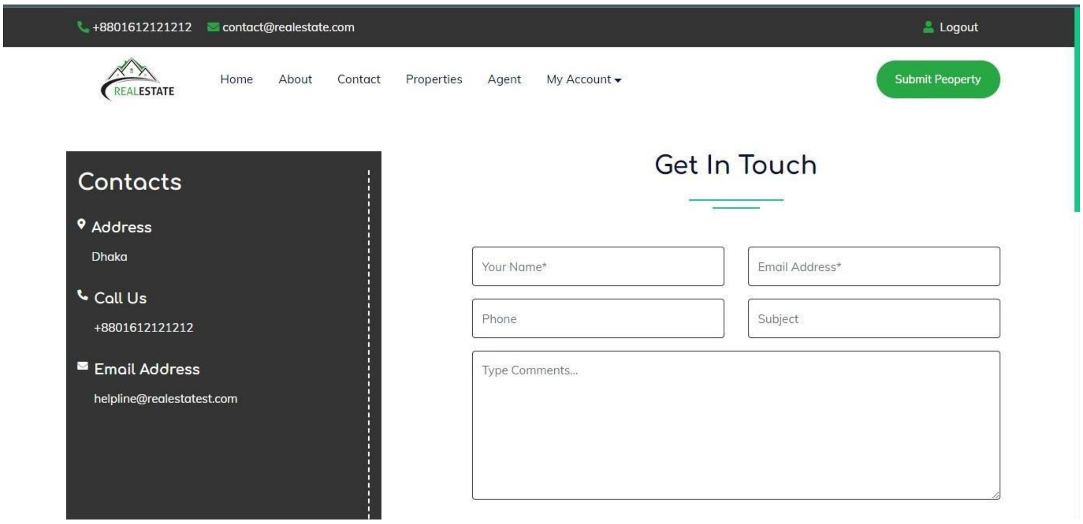
Division Name

Division List

#	Division	Actions
1	Dhaka	Edit Delete
2	Rajshahi	Edit Delete
3	Sylhet	Edit Delete
4	Chottogram	Edit Delete

Figure 6.6: Division List

6.7 Contact page



☎ +8801612121212 ✉ contact@realestate.com

Logout

REALESTATE

Home About Contact Properties Agent My Account ▾

Submit Peoperty

Contacts

Address

Dhaka

Call Us

+8801612121212

Email Address

helpline@realestatest.com

Get In Touch

Your Name*

Email Address*

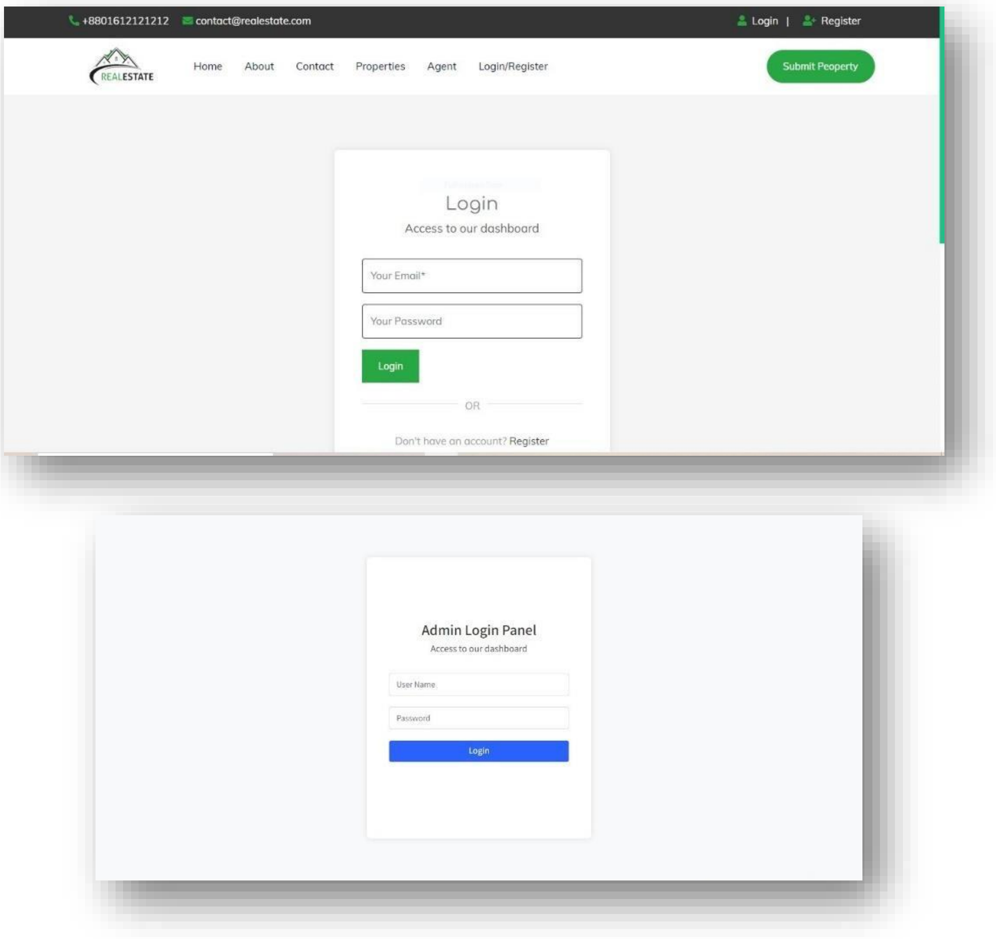
Phone

Subject

Type Comments...

Figure 6.7: Contact Page

6.8 Login page



☎ +8801612121212 ✉ contact@realestate.com

Login | Register

REALESTATE

Home About Contact Properties Agent Login/Register

Submit Peoperty

Login

Access to our dashboard

Your Email*

Your Password

Login

OR

Don't have an account? Register

Admin Login Panel

Access to our dashboard

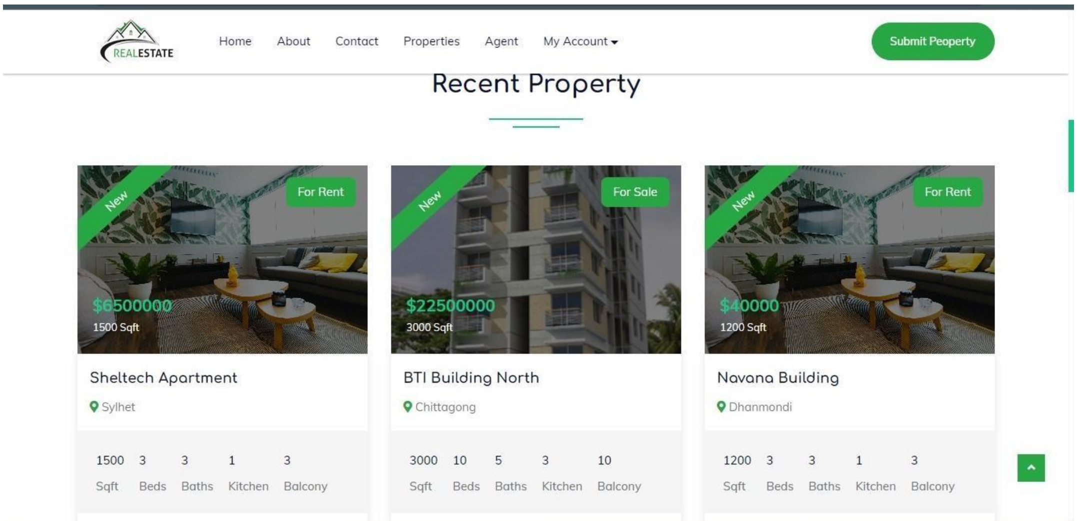
User Name

Password

Login

Figure 6.8: Login page(upper page for user and lower page for admin)

6.9 Property view



Recent Property

$6500000
1500 Sqft
Sheltech Apartment
Sylhet

1500	3	3	1	3
Sqft	Beds	Baths	Kitchen	Balcony

$22500000
3000 Sqft
BTI Building North
Chittagong

3000	10	5	3	10
Sqft	Beds	Baths	Kitchen	Balcony

$40000
1200 Sqft
Navana Building
Dhanmondi

1200	3	3	1	3
Sqft	Beds	Baths	Kitchen	Balcony

Figure 6.9: Property view

Chapter: 07
CONCLUSION

7.1 Conclusion

This project has been a rewarding experience in more than one way. The entire project work has enlightened us in the following areas. We have gained an insight into the working of the Real Estate Business. This represents a typical real-world situation. Our understanding of database design has been strengthened because in order to generate the final reports of database designing has to be properly followed. Scheduling a project and adhering to that schedule creates a strong sense of time management. Sense of teamwork has developed and confidence in handling real life projects has increased to a great extent.

Initially, there were problems with the validation but with discussions, we were to implement validations.

7.2 Limitations of the system

The current version of the system does not support online payment. This means that users cannot pay for goods or services using a credit card or other online payment method. The data delete and edit system is not available for all sections of the system. This means that users cannot delete or edit data in certain areas of the system. User accounts that are not verified by mobile SMS are not available in this system. This means that users must verify their account using a mobile phone number in order to use the system. Data loss due to mismanagement is a risk associated with using this system. This means that there is a risk that data could be lost or corrupted due to human error or technical problems.

Here are some additional details about each of these limitations:

- Online payment is not available because the system does not have the necessary security features to protect user financial information.
- The data delete and edit system is not available for all sections of the system because some data is essential for the system to function properly.
- User accounts that are not verified by mobile SMS are not available in this system because this is a security measure to prevent unauthorized access to the system.

- Data loss due to mismanagement is a risk associated with using any system, but it is especially important to be aware of this risk when using a system that contains sensitive data.

7.3 Future Enhancement

The proposed system is the Real Estate Management System. We can enhance this system by including more facilities like adding an auction system for the custom of online auctioning. Providing such features enable the users to include more comments into the system.

Reference:

[1] Bangladesh Real Estate - *https://en.wikipedia.org/wiki/Real_estate_in_Bangladesh*

[2] Agile software development - *https://en.wikipedia.org/wiki/Agile_software_development*

[3] System requirements - *https://en.wikipedia.org/wiki/System_requirements*

[4] Predicting performance of software systems during feasibility study of software project management - *https://ieeexplore.ieee.org/document/4449845*

[5] System requirements - *https://en.wikipedia.org/wiki/System_requirements*

[6] Agile software development - *https://en.wikipedia.org/wiki/Agile_software_development*

[7] Bangladesh Real Estate - *https://en.wikipedia.org/wiki/Real_estate_in_Bangladesh*

Appendix

Source Code:

```
<?php
ini_set('session.cache_limiter','public');
session_cache_limiter(false);
session_start();
include("config.php");

<link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="css/bootstrap-slider.css">
<link rel="stylesheet" type="text/css" href="css/jquery-ui.css">
<link rel="stylesheet" type="text/css" href="css/layerslider.css">
<link rel="stylesheet" type="text/css" href="css/color.css" id="color-change">
<link rel="stylesheet" type="text/css" href="css/owl.carousel.min.css">
<link rel="stylesheet" type="text/css" href="css/font-awesome.min.css">
<link rel="stylesheet" type="text/css" href="fonts/flaticon/flaticon.css">
<link rel="stylesheet" type="text/css" href="css/style.css">
        <div class="overlay-black w-100 slider-banner1 position-relative"
style="background-image: url('images/banner/rshmpg.jpg'); background-size: cover;
background-position: center center; background-repeat: no-repeat;">
            <div class="container h-100">
                <div class="row h-100 align-items-center">
                    <div class="col-lg-12">
                        <div class="text-white">
                            <h1 class="mb-4"><span class="text-success">Let us</span><br>
                            Guide you Home</h1>

                            <form method="post" action="propertygrid.php">
                                <div class="row">
                                    <div class="col-md-6 col-lg-2">
                                        <div class="form-group">
                                            <select class="form-control" name="type">
                                                <option value="">Select Type</option>

            <option value="apartment">Apartment</option>

            <option value="flat">Flat</option>

            <option value="building">Building</option>

            <!-- <option value="house">House</option>

            <option value="villa">Villa</option>
```

```
                      </select>
                    </div>
                  </div>
                  <div class="col-md-8 col-lg-6">
                    <div class="form-group">
                      <input type="text" class="form-control" name="city"
placeholder="Enter City" required>
            <div class="col-lg-3 col-md-6">
              <div class="p-4 text-center hover-bg-white hover-shadow rounded
mb-4 transation-3s">
                                                <i class="flaticon-for-rent
text-success flat-medium" aria-hidden="true"></i>
-white hover-shadow rounded mb-4 transation-3s">
                                                <i class="flaticon-
diagram text-success flat-medium" aria-hidden="true"></i>
                <h5 class="text-secondary hover-text-success py-3 m-0"><a
href="#">Legal Investment</a></h5>
                <p>This is a dummy text for filling out spaces. Just some random

            <!--     How it work -->
    <div class="full-row">
      <div class="container">
        <div class="row">
          <div class="col-lg-12">
            <h2 class="text-secondary double-down-line text-center mb-5">How
It Work</h2>
            </div>
        </div>
        <div class="row">
          <div class="col-md-4">
            <div class="icon-thumb-one text-center mb-5">
              <div class="bg-success text-white rounded-circle position-absolute
z-index-9">1</div>
              <div class="left-arrow"><i class="flaticon-investor flat-medium
icon-success" aria-hidden="true"></i></div>
              <h5 class="text-secondary mt-5 mb-4">Discussion</h5>
              <p>Nascetur cubilia sociosqu aliquet ut elit nascetur nullam duis
tincidunt nisl non quisque vestibulum platea ornare ridiculus.</p>
            </div>
          </div>
          <div class="col-md-4">
            <div class="icon-thumb-one text-center mb-5">
              <div class="bg-success text-white rounded-circle position-absolute
z-index-9">2</div>
              <div class="left-arrow"><i class="flaticon-search flat-medium icon-
success" aria-hidden="true"></i></div>
```

```
                    <h5 class="text-secondary mt-5 mb-4">Files Review</h5>
                    <p>Nascetur cubilia sociosqu aliquet ut elit nascetur nullam duis
tincidunt nisl non quisque vestibulum platea ornare ridiculus.</p>
                </div>
            </div>
            <div class="col-md-4">
                <div class="icon-thumb-one text-center mb-5">
                    <div class="bg-success text-white rounded-circle position-absolute
z-index-9">3</div>
                    <div><i class="flaticon-handshake flat-medium icon-success" aria-
hidden="true"></i></div>
                    <h5 class="text-secondary mt-5 mb-4">Acquire</h5>
                    <p>Nascetur cubilia sociosqu aliquet ut elit nascetur nullam duis
tincidunt nisl non quisque vestibulum platea ornare ridiculus.</p>
                </div>
            </div>
; background-position: center center; background-repeat: no-repeat;">
        <div class="container">
            <div class="fact-counter">
                <div class="row">
                    <div class="col-md-3">
                        <div class="count wow text-center  mb-sm-50" data-wow-
duration="300ms"> <i class="flaticon-house flat-large text-white" aria-
hidden="true"></i>
                                                    <?php

        $query=mysqli_query($con,"SELECT count(pid) FROM property");

        while($row=mysqli_fetch_array($query))

        {
                                                                        ?>
                        <div class="count-num text-success my-4" data-speed="3000"
data-stop="<?php

        $total = $row[0];

        echo $total;?>">0</div>
                                                    <?php } ?>
                        <div class="text-white h5">Sale Property Available</div>
                    </div>
                </div>
                                            <div class="col-md-3">
                    <div class="count wow text-center  mb-sm-50" data-wow-
duration="300ms"> <i class="flaticon-house flat-large text-white" aria-
hidden="true"></i>
```

```
                                                                                    <?php

        $query=mysqli_query($con,"SELECT count(pid) FROM property where
stype='rent'");

        while($row=mysqli_fetch_array($query))

        {
                                                                                                                    ?>
                    <div class="count-num text-success my-4" data-speed="3000"
data-stop="<?php

        $total = $row[0];

        echo $total;?>">0</div>
                                                                                    <?php } ?>
                    <div class="text-white h5">Rent Property Available</div>
                  </div>
                </div>
                <div class="col-md-3">
                  <div class="count wow text-center  mb-sm-50" data-wow-
duration="300ms"> <i class="flaticon-man flat-large text-white" aria-
hidden="true"></i>
                    <?php

        $query=mysqli_query($con,"SELECT count(uid) FROM user");

        while($row=mysqli_fetch_array($query))

        {
                                                                                                                    ?>
                    <div class="count-num text-success my-4" data-speed="3000"
data-stop="<?php

        $total = $row[0];

        echo $total;?>">0</div>
                                                                                    <?php } ?>
                    <div class="text-white h5">Registered Users</div>
                  </div>
                </div>

              </div>
            </div>
          </div>
        </div>
        <!--        Popular Place -->
        <!-- <div class="full-row bg-gray">
            <div class="container">
                <div class="row">
                    <div class="col-lg-12">
                        <h2 class="text-secondary double-down-line text-center mb-
5">Popular Places</h2></div>
                </div>
                <div class="col-lg-12">
                    <div class="row">
                        <div class="col-md-6 col-lg-3 pb-1">
                            <div class="overflow-hidden position-relative overlay-secondary
hover-zoomer mx-n13 z-index-9"> <img src="images/thumbnail4/1.jpg" alt="">
                                <div class="text-white xy-center z-index-9 position-absolute text-
center w-100">
                                    <?php
                                                                                            //
$query=mysqli_query($con,"SELECT * FROM property where city='dhaka'");

                                        $query = mysqli_query($con,"select * from property where
division='Dhaka'");
                                        $total = mysqli_num_rows($query);

            while($row=mysqli_fetch_array($query))

            {
                                                                                            ?>
                                    <h4 class="hover-text-success text-capitalize"><a
href="divisionproperty.php?id=<?php echo $row['pid']?>"><?php echo
$row['division'];?></a></h4>

            <?= $total;?> Properties Listed</span> </div>
                                                                                <?php  } ?>
                            </div>
                        </div>
                        <div class="col-md-6 col-lg-3 pb-1">
                            <div class="overflow-hidden position-relative overlay-secondary
hover-zoomer mx-n13 z-index-9"> <img src="images/thumbnail4/2.jpg" alt="">
                                <div class="text-white xy-center z-index-9 position-absolute text-
center w-100">
                                                                                <?php
                                                                                            //
$query=mysqli_query($con,"SELECT * FROM property where city='Rajshahi'");
```

```
                              $query = mysqli_query($con,"select * from property where
division='Sylhet'");
                              $total = mysqli_num_rows($query);

         while($row=mysqli_fetch_array($query))

         {
                                                                                ?>
                           <h4 class="hover-text-success text-capitalize"><a
href="divisionproperty.php?id=<?php echo $row['pid']?>"><?php echo
$row['division'];?></a></h4>

         <?= $total;?> Properties Listed</span> </div>
                                                                         <?php  } ?>
                     </div>
                  </div>
                  <div class="col-md-6 col-lg-3 pb-1">
                     <div class="overflow-hidden position-relative overlay-secondary
hover-zoomer mx-n13 z-index-9"> <img src="images/thumbnail4/3.jpg" alt="">
                        <div class="text-white xy-center z-index-9 position-absolute text-
center w-100">
                           <?php
                                                                                //
$query=mysqli_query($con,"SELECT * FROM property where city='Rajshahi'");

                              $query = mysqli_query($con,"select * from property where
division='Chottogram'");
                              $total = mysqli_num_rows($query);

         while($row=mysqli_fetch_array($query))

         {
                                                                                ?>
                           <h4 class="hover-text-success text-capitalize"><a
href="divisionproperty.php?id=<?php echo $row['pid']?>"><?php echo
$row['division'];?></a></h4>

         <?= $total;?> Properties Listed</span> </div>
                                                                         <?php  } ?>
                     </div>
                  </div>
                  <div class="col-md-6 col-lg-3 pb-1"

         <?= $total;?> Properties Listed</span> </div>
                                                                         <?php  } ?>
                     </div>
```

```
        <!--          Popular Places -->

<script src="js/jquery.min.js"></script>
<!--jQuery Layer Slider -->
<script src="js/greensock.js"></script>
<script src="js/layerslider.transitions.js"></script>
<script src="js/layerslider.kreaturamedia.jquery.js"></script>
<!--jQuery Layer Slider -->
<script src="js/popper.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/owl.carousel.min.js"></script>
<script src="js/tmpl.js"></script>
<script src="js/jquery.dependClass-0.1.js"></script>
<script src="js/draggable-0.1.js"></script>
<script src="js/jquery.slider.js"></script>
<script src="js/wow.js"></script>
<script src="js/YouTubePopUp.jquery.js"></script>
<script src="js/validate.js"></script>
<script src="js/jquery.cookie.js"></script>
<script src="js/custom.js"></script>
```