# 



**- Project 3 Documentation –**

***Realtors Portal***

|  |  |
| --- | --- |
| **Authors** | **C2209L - Group 3** |
| 1. **Nguyen Cong Minh Duc** 2. **Nguyen Huy Hoang Duong** 3. **Nguyen Hoang Minh** 4. **Vo Bao Thang** 5. **Tran Hoang Dung** |
| **Date** | **29/3/2024** |
| **Mentor** | **Teacher Nguyen Manh Hung** |

- **Ha Noi, 2024 -**

**TABLE OF CONTENT**

***TABLE OF CONTENT***

***LIST OF TABLES***

***LIST OF FIGURES***

***OVERVIEW***

**CHAPTER 1: INTRODUCTION & ABSTRACTION**

***1.1. Introduction 6***

*1.1.1. Purpose*

*1.1.2. Intended Audience and Intended Use*

*1.1.3. Product Scope*

***1.2. Detail project specific requirements* 7**

*1.2.1. Functional Requirements*

*1.2.2. Non – Functional Requirements*

**CHAPTER 2: SYSTEM ARCHITECTURE & BUSINESS ANALYSIS 11**

***2.1. Overview Business System 11***

***2.2. Business Analysis 12***

*2.2.1. Main modules in System*

*2.2.2. Modules process*

**CHAPTER 3: SOFTWARE DESIGN 19**

*3.1. Modular Diagram* **19**

*3.2. Environment Setup and Tools selection for developing* **19**

*3.3. Prototype Model* **21**

*3.4. Class Model* **24**

*3.4.1. ERD*

*3.4.2. FDD*

*3.5. Design Pattern* **34**

*3.5.1. CQRS Pattern*

*3.5.2. Repository Pattern*

*3.5.3. Unit of Work Pattern*

*3.6. UI Design* **37**

**CHAPTER 4: TESTING AND PERFORMANCE APPRAISAL 45**

*4.1. Software Implementation* **48**

*4.2. Deployment - Testing and Performance Appraisal* **49**

*4.2.1. Planning for testing*

*4.2.2. Test implementing*

**LIST OF TABLE**

|  |  |
| --- | --- |
| **Abbreviation** | **Full Information** |
| **ORM** | **Object – Relational Mapping** |
| **ERD** | **Entity Relationship Diagram** |
| **UI** | **User - interface** |
| **CQRS** | **Command and Query Responsibility Segregation** |
| **SQL** | **Structured Query Language** |
| **CPU** | **Central Processing Unit** |
| **UML** | **Uniformed Model Language** |

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Figures** | **Description** | **Page Number** |
| **1** | **Figure 1** | **System Activity Diagram** | **14** |
| **2** | **Figure 2** | **Login Activity Diagram** | **16** |
| **3** | **Figure 3** | **Offering, Searching, View details Diagram** | **17** |
| **4** | **Figure 4** | **Modular Diagram** | **19** |
| **5** | **Figure 5** | **Prototype Model Diagram** | **21** |
| **6** | **Figure 6** | **Entity Relationship Diagram** | **24** |
| **7** | **Figure 7** | **Functional Decomposition Diagram** | **29** |
| **8** | **Figure 8** | **Use – case Diagram** | **33** |
| **9** | **Figure 9** | **Property Listing** | **37** |
| **10** | **Figure 10** | **Search and Filter** | **38** |
| **11** | **Figure 11** | **User Profile** | **39** |
| **12** | **Figure 12** | **Sign Up Page** | **40** |
| **13** | **Figure 13** | **Login Page** | **40** |
| **14** | **Figure 14** | **Post Management** | **41** |
| **15** | **Figure 15** | **Property Detail** | **42** |
| **16** | **Figure 16** | **Testing Schedule** | **48** |

**OVERVIEW**

The Realtors Portal project holds a pivotal role in the learning journey at Aptech, serving as a cornerstone for students to enhance their skills and knowledge in Information Technology (IT) within the context of real estate management. This project not only provides a platform for the application of acquired knowledge but also fosters opportunities for students to showcase their creativity, problem-solving abilities, and understanding of real-world challenges in the realm of real estate.

Microsoft technology plays a vital role in shaping the Realtors Portal project and its impact on the IT landscape, both locally in Vietnam and globally. With Microsoft's suite of tools, frameworks, and cloud services, developers can create innovative features and seamless user experiences within the Realtors Portal, enhancing its functionality and usability.

In chapter 1, an software requirements specification gives all member a complete picture of the entire project. It provides a single source of truth that everyteam

involved in development will follow.

For chapter 2, we provide a general landscape about building first foundation for our project. All modules and architectures are described in this chapter.

In chapter 3: Software Design, we provide content about the project’s system architecture design, development prototype design, database design, software interface design. This document is used to provide the software development team.

Chapter 4: Deployment-testing and Performance evaluation: The content of chapter 3 presents the steps to deploy the system on the server platform as well as the

construction of test procedures (test Process), test cases and assessment of the

system capacity.

In conclusion, final chapter: Conclusion: The content of this section summarizes

and evaluates the Project, giving some development directions for the product.

**CHAPTER 1: INTRODUCTION & ABSTRACTION**

* 1. ***Introduction***

***1.1.1 Purpose***

Our website’s purpose is to provide a platform for users to search, buy, sell, or rent properties. It typically offers features like property listings, search filters, virtual tours, neighborhood information, mortgage calculators, and contact options to connect with real estate agents or property owners. Overall, it serves as a central hub for all things related to real estate transactions and information.

***1.1.2 Intended Audience and Intended Use:***

The intended audience for a real estate website includes prospective buyers, sellers, renters, real estate agents, and investors. The intended use is to facilitate property transactions, provide information about available properties, assist in property searches, offer resources for buyers and sellers, and connect users with real estate professionals. Essentially, it serves as a comprehensive online platform for all parties involved in the real estate market to interact and conduct business.

* + 1. ***Product Scope:***

Our scope encompasses a range of features and functionalities aimed at facilitating property transactions and providing information to users. This includes property listings, search filters, neighborhood information, mortgage calculators, contact options for real estate agents or property owners, market trends, investment opportunities, and possibly additional services such as property management tools or legal resources. The goal is to create a comprehensive platform that meets the diverse needs of individuals involved in real estate transactions.

* 1. ***Detail project specific requirements***
     1. ***Functional Requirements:***
* Property Listings: Ability to create, edit, and display detailed listings for properties including descriptions, photos, videos, and specifications.
* Search Functionality: Advanced search filters allowing users to narrow down property listings based on criteria such as location, price range, property type, number of bedrooms/bathrooms, etc.
* User Registration and Profiles: Capability for users to create accounts, save favorite listings, track search history, and manage communication preferences.
* Contact Options: Easy ways for users to contact real estate agents or property owners for inquiries, scheduling viewings, or making offers.
* Neighborhood Information: Integration of maps and information about local amenities, schools, transportation, crime rates, and other neighborhood details.
* Mortgage Calculator: Tool for users to estimate mortgage payments based on loan amount, interest rate, and term.
* Responsive Design: Ensuring the website is optimized for viewing on various devices including desktops, tablets, and smartphones.
* Secure Payment Processing: If applicable, integration of secure payment processing for services such as listing fees or featured property promotions.
* Analytics and Reporting: Integration of tools to track website traffic, user behavior, and performance metrics to inform decision-making and optimization efforts.
  + 1. ***Non-functional Requirements***

1. *Performance:*

* Fast loading times for property listings and search results.
* Scalability to handle increasing numbers of users and property listings without significant performance degradation.
* Efficient database management to handle large amounts of data.

*Usability:*

* Intuitive user interface design with easy navigation and clear labeling.
* Compatibility with different web browsers and devices to ensure a consistent user experience.
* Accessibility features to accommodate users with disabilities, such as screen readers and keyboard navigation.

1. *Reliability:*

* High availability to ensure the website is accessible to users at all times.
* Regular backups of data to prevent loss of information in case of system failure or data corruption.
* Monitoring and alerting systems to detect and respond to system errors or performance issues promptly.

1. *Scalability:*

* Architecture designed to accommodate growth in user traffic and property listings over time.
* Use of cloud-based infrastructure or scalable hosting solutions to support increased demand.

1. *Security:*

* Secure user authentication and authorization mechanisms to protect user accounts and personal information.
* Encryption of sensitive data such as payment information and user credentials.
* Protection against common web vulnerabilities such as cross-site scripting (XSS) and SQL injection.

**This page intentionally left blank**

**CHAPTER 2: SYSTEM ARCHITECTURE & BUSINESS ANALYSIS**

**2.1: Overview Business System**

A realtor portal is a comprehensive online platform designed specifically for real estate professionals to manage various aspects of their business efficiently. Here's an overview of the key components and functionalities typically found in a realtor portal:

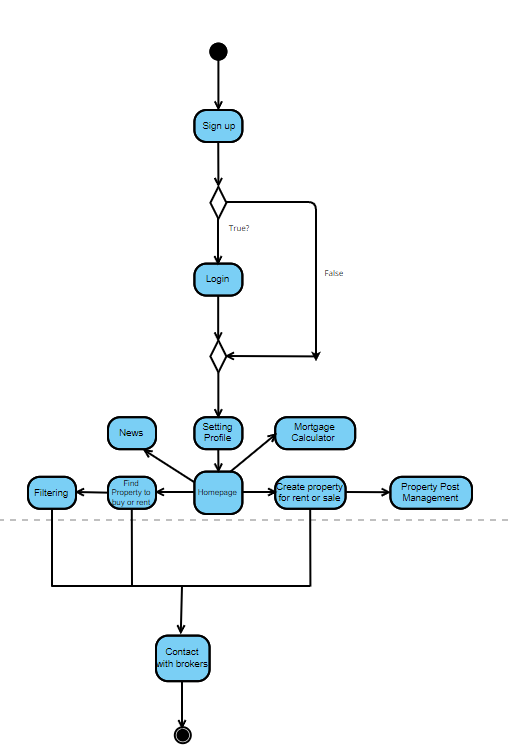
1. Property Listings Management: The portal allows realtors to create, edit, and manage listings for properties they represent. This includes uploading photos, videos, property details, and descriptions.
2. Search and Filtering: Users can search for properties based on specific criteria such as location, price range, number of bedrooms/bathrooms, amenities, etc. Advanced filtering options help narrow down search results.
3. Client Management: Realtors can organize and manage their client contacts, including buyers, sellers, and leads. They can track client preferences, communication history, appointments, and property preferences.
4. Transaction Management: This feature helps realtors manage the entire transaction process, including offers, contracts, negotiations, inspections, and closings. It may include tools for document management, e-signatures, and compliance.

**2.2. Business Analysis**

Our system can be described very clearly at this diagram below.

Generally, here are steps for user to access to Realtors Portal.

1. **Sign-Up/Registration**:
   * Visit the Realtor Portal website.
   * Click on the "Sign-Up" or "Register" button.
   * Fill out the registration form, providing personal information such as name, email address, phone number, and potentially brokerage details.
   * Agree to the terms and conditions and submit the registration form.
2. **Account Verification**:
   * Upon submission, the user may receive an email confirmation to verify their email address.
   * Click on the verification link provided in the email to confirm the account.
3. **Profile Setup**:
   * Once the account is verified, the user may be prompted to complete their profile.
   * This could include adding additional details such as bio, profile picture, brokerage information, and areas of expertise.
4. **Exploring Features**:
   * After profile setup, the user gains access to the various features of the Realtor Portal.
   * They can explore property listings, search for properties based on criteria, and view detailed property information.
5. **Client Management**:
   * Users can start managing their clients by adding contacts, organizing them into categories such as buyers, sellers, or leads.
   * They can track client preferences, communication history, appointments, and property interests.
6. **Property Listings Management**:
   * Realtors can create and manage property listings by adding details, descriptions, photos, and videos.
   * They can edit existing listings and mark properties as sold or pending.
7. **Engaging with Clients**:
   * Users can communicate with clients through the portal via messaging systems or email integration.
   * They can schedule property viewings, meetings, and other appointments directly through the portal.
8. **Transaction Management**:
   * When a property deal progresses, users can manage the transaction process through the portal.
   * This includes handling offers, contracts, negotiations, inspections, and closings.
   * Documents may be uploaded, reviewed, and signed electronically within the portal.



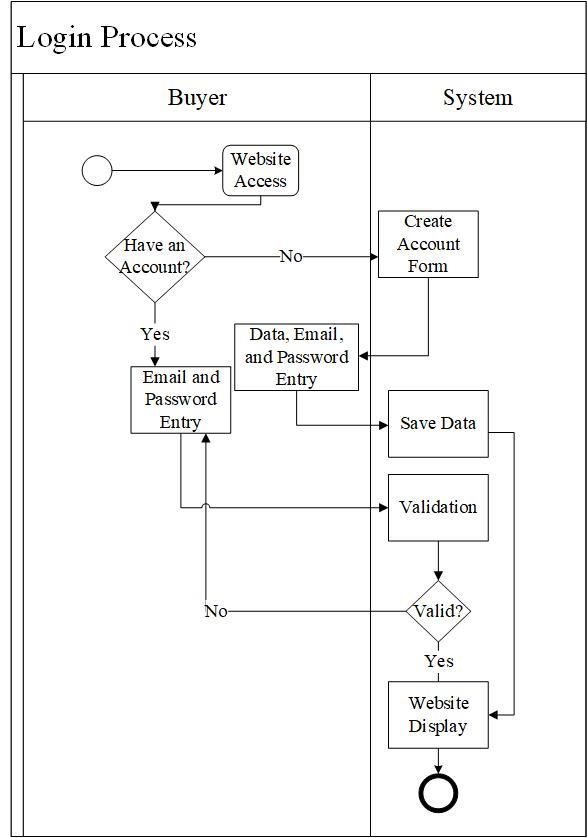
**Figure 1. System Activity Diagram**

**2.2.1. Main modules in System**

|  |  |
| --- | --- |
| **No** | **Description** |
| **Module: User Management** | |
| **1.** | **User profile setting process** |
| **2.** | **User Sign Up process** |
| **3.** | **User Login process** |
| **Module: Real Estate Agent Management** | |
| **1.** | **Add Property for sale** |
| **2.** | **Update Property for sale** |
| **3.** | **Delete Property for sale** |
| **4.** | **Add contact detail** |
| **5.** | **Add Property for rent** |
| **6.** | **Update Property for rent** |
| **7.** | **Delete Property for rent** |
| **Module: Mortgage Calculator** | |
| **1.** | **Calculate the mortgage** |

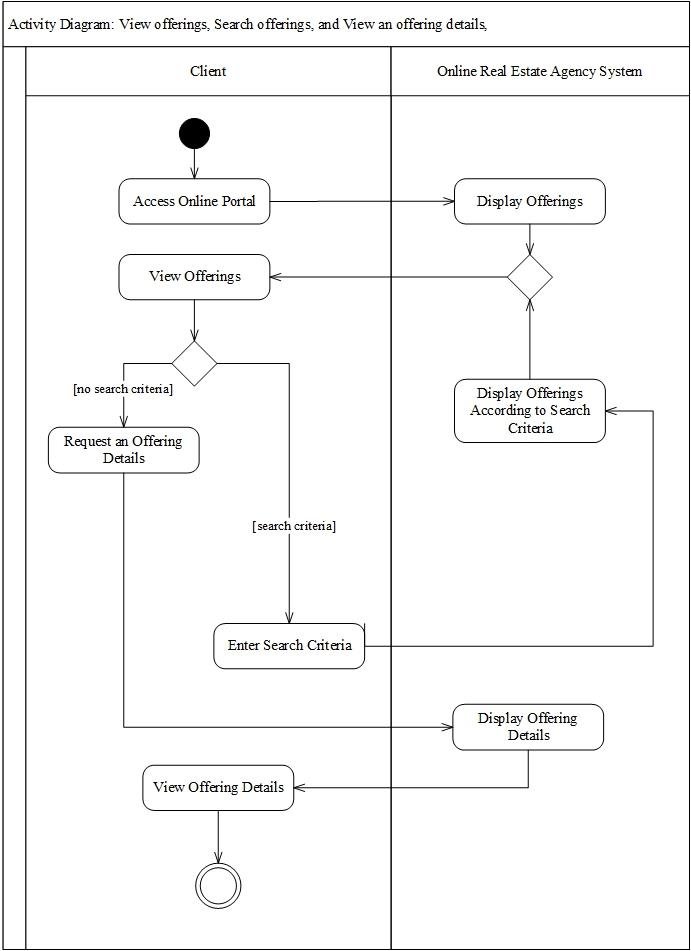
**Module: User Management**

**Login/Sign up process**



**Figure 2. Login Activity Diagram**

**Module: Real Estate Property Finding Process**

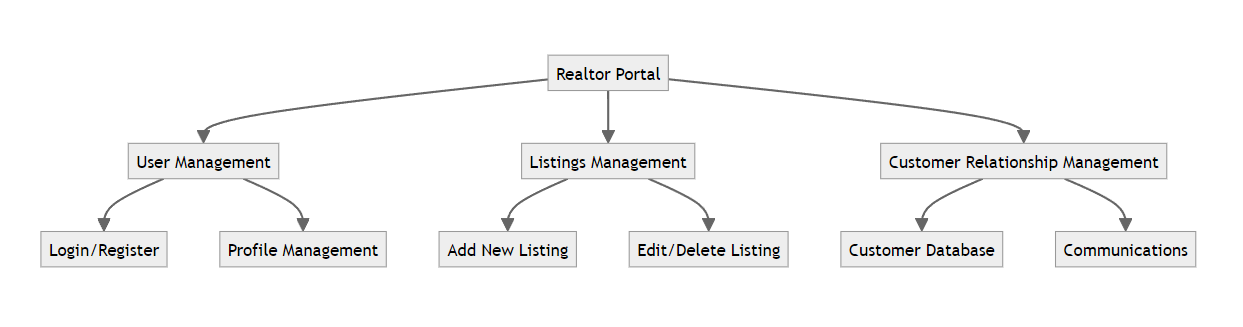


**Figure 3: Offering, Searching, View details diagram**

**This page intentionally left blank**

**CHAPTER 3: SOFTWARE DESIGN**

**3.1. Modular Diagram**

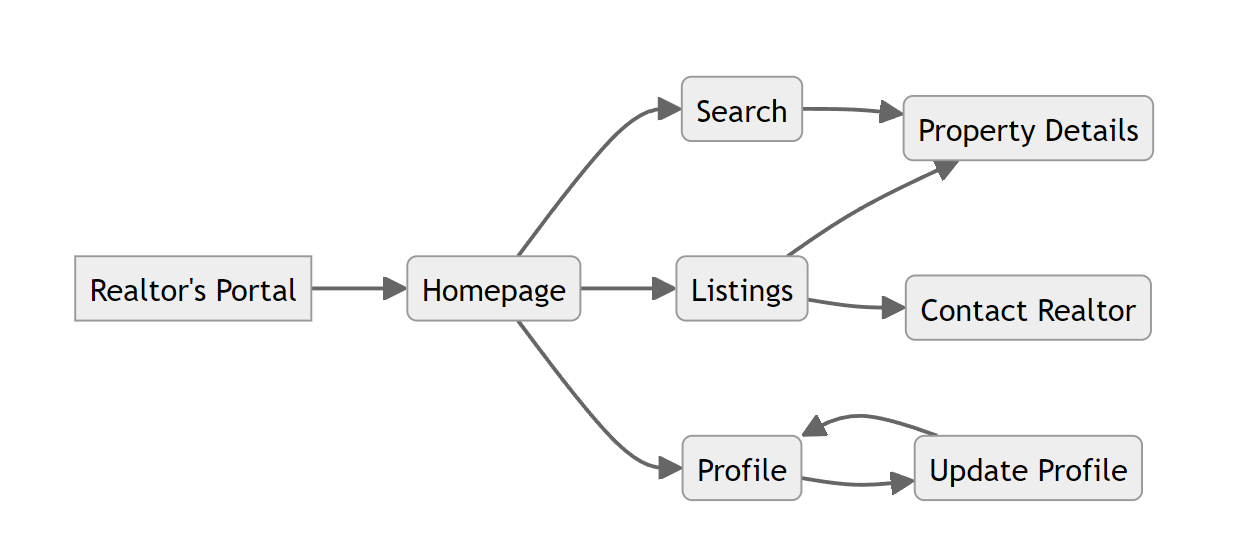
****

**Figure 4: Modular Diagram**

**3.2.***.* **Environment setup and tools selection for developing**

* The theoretical basis of information system analysis and design involves the systematic approach to understanding, modeling, and implementing efficient and effective information systems. It encompasses various methodologies and techniques aimed at developing high-quality systems that meet the needs of users and organizations.
* When it comes to database management systems (DBMS), they play a crucial role in information system analysis and design. DBMS is software that facilitates the creation, organization, and manipulation of data in a structured manner. It provides mechanisms for storing, retrieving, updating, and managing data, ensuring data integrity and security.
* In the context of programming, C# (C Sharp) is a powerful object-oriented programming language developed by Microsoft. It is widely used for building various applications, including information systems. C# offers a rich set of features, such as strong typing, garbage collection, and extensive libraries, making it suitable for developing complex software solutions.
* In this project, we use SQL Server. SQL Server is a popular and widely used relational database management system (RDBMS) developed by Microsoft. It provides a robust and scalable platform for storing, managing, and retrieving data efficiently. SQL Server offers a wide range of features and capabilities that make it suitable for various applications, from small-scale projects to enterprise-level systems.

**3.3. Prototype Model**

****

**Figure 5: Prototype Model Diagram**

**Prototype Model for Realtors' Portal:**

Homepage:

* Display a welcome message and an overview of the portal.
* Include a search bar for property searches.
* Provide quick links to popular property listings or featured properties.
* Show notifications or updates relevant to the realtors.

Search:

* Allow users to search for properties based on various criteria such as location, price range, property type, etc.
* Provide filters to refine search results.
* Display search results in a list or grid format, showing essential information for each property (e.g., price, location, brief description, and an image).
* Include pagination or infinite scrolling for browsing through multiple search results.

Listings:

* Show a detailed list of properties available for sale or rent.
* Include sorting options to arrange listings by price, location, or other relevant factors.
* Each listing should display essential details such as property type, price, location, description, and multiple images.
* Allow users to save or favorite listings for future reference.
* Provide a "Contact Realtor" button for users interested in a particular property.

Profile:

* Allow realtors to create and manage their profiles.
* Include fields for personal information, contact details, professional experience, and a profile picture.
* Enable realtors to update their profile information, including contact details, bio, and areas of expertise.

Property Detail:

* Display comprehensive information about a specific property.
* Include details such as property type, price, location, description, amenities, and multiple high-quality images.
* Show contact information for the realtor associated with the property.
* Provide a form or button for users to request more information or schedule a viewing.

Contact Realtor:

* Include a contact form for users to send messages or inquiries to the realtors.
* Collect user information such as name, email, and message details.
* Send the user's inquiry to the appropriate realtor's email address for follow-up.

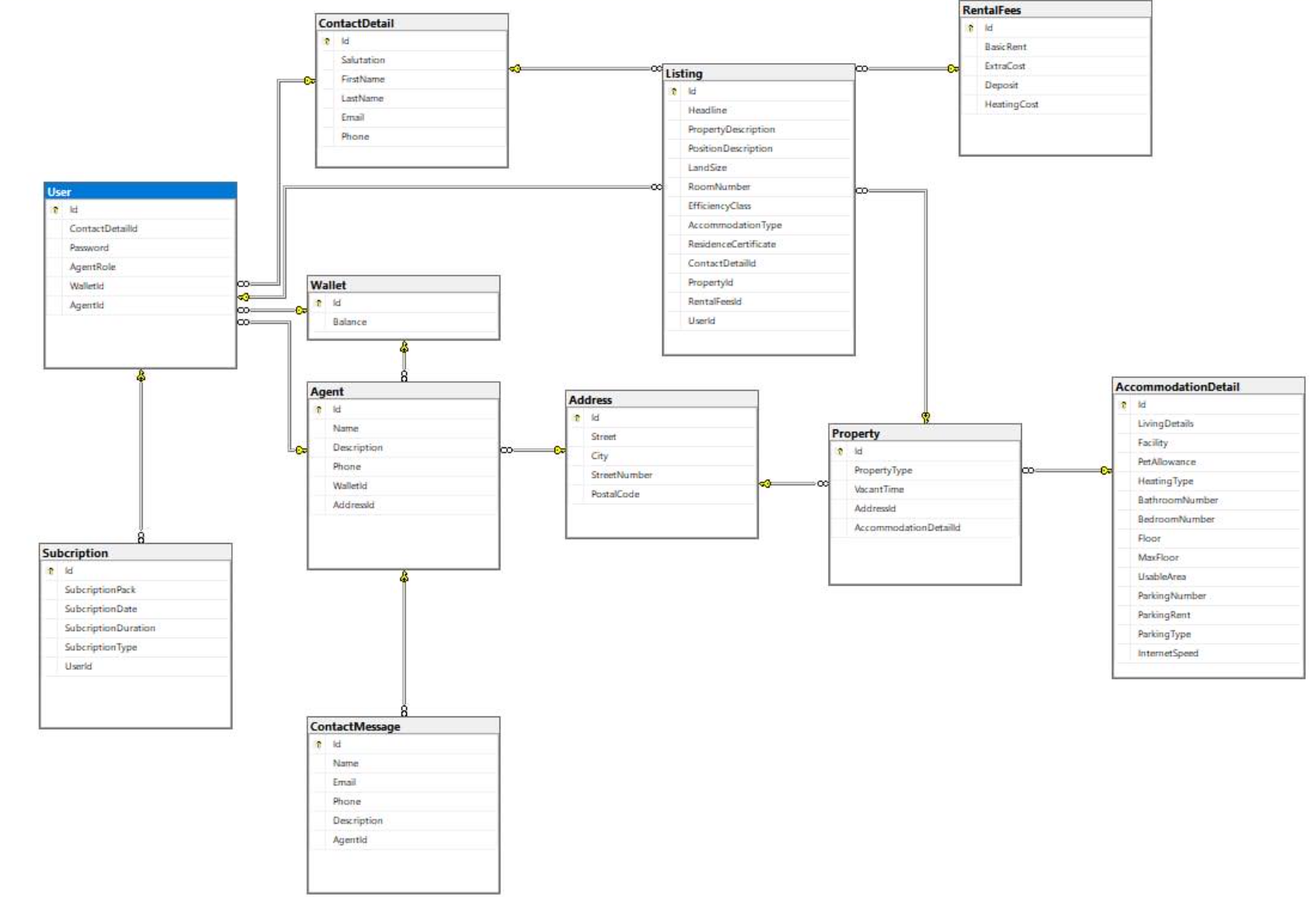
Update Profile:

* Allow realtors to edit and update their profile information.
* Include fields for personal details, contact information, professional experience, and a profile picture.
* Provide options to add or update listings associated with the realtor's profile.

**3.4. Class Model**

*The class model for a realtors' portal represents the structure and relationships between various classes that make up the system. It provides an overview of the entities involved and their attributes and behaviors.*

**3.4.1. Entity Relationship Diagram**

****

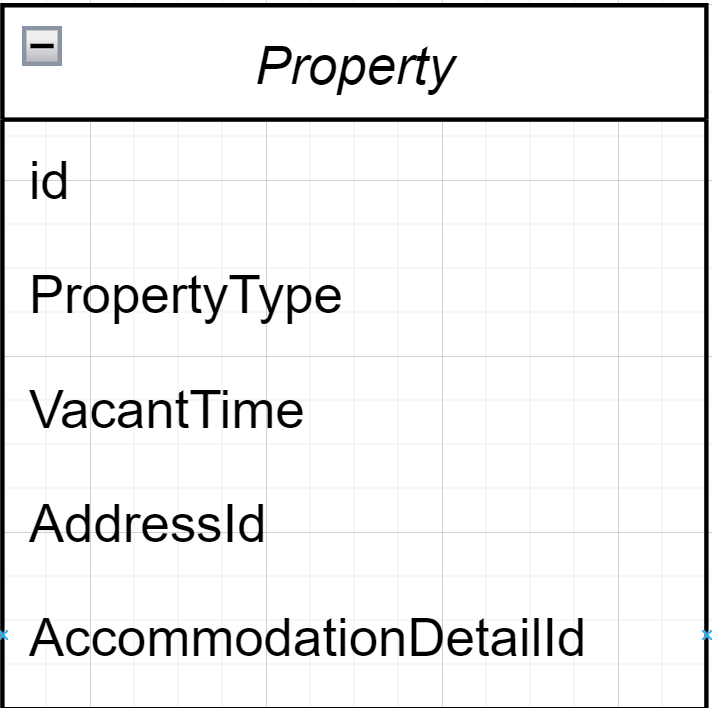
**Figure 6: Entity Relationship Diagram**

**Table details:**

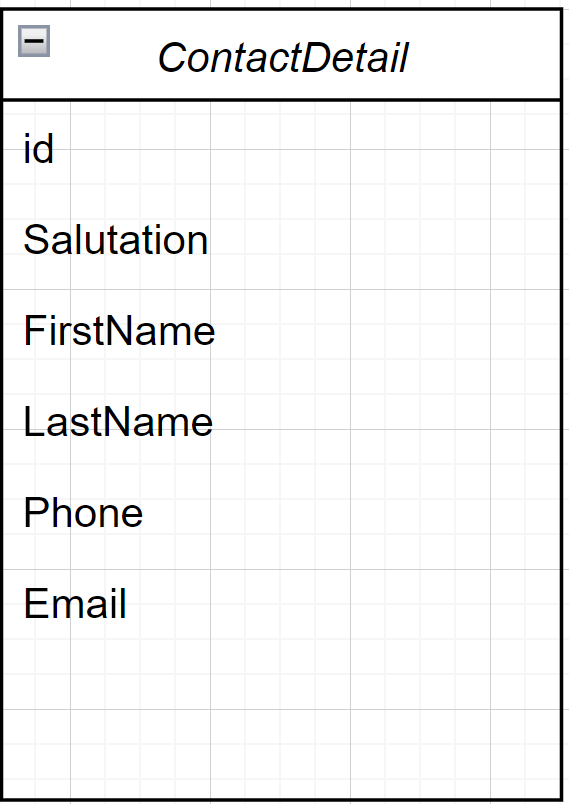
**User table:**

****

**Listing table: Property table:**

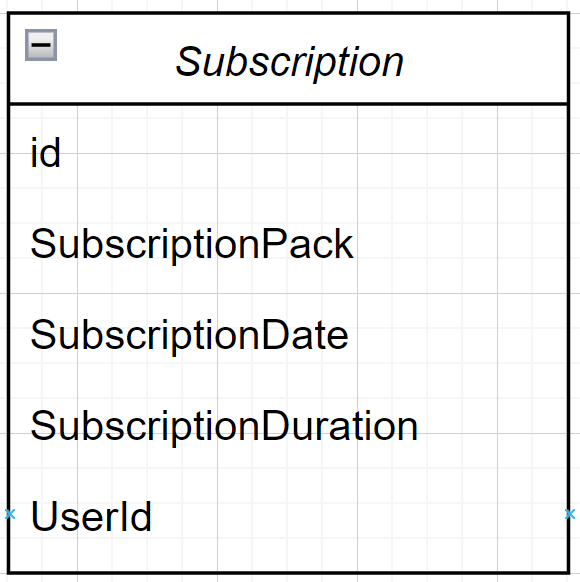
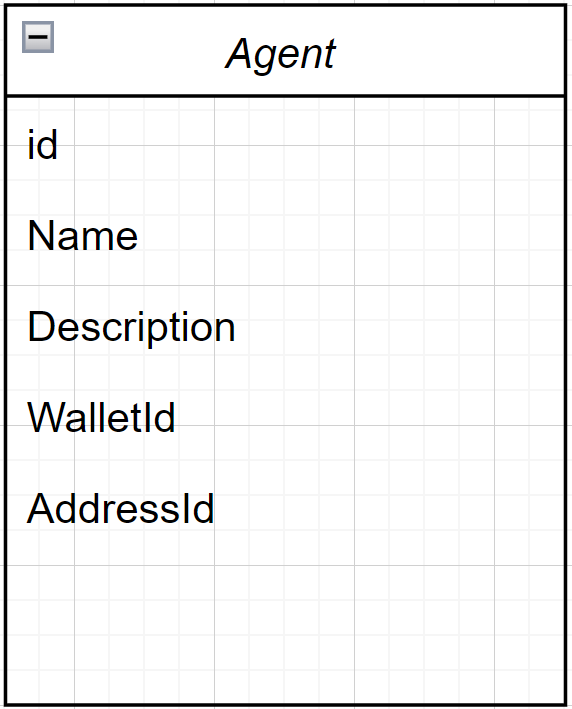
** **

**Contact Detail table: Address table:**

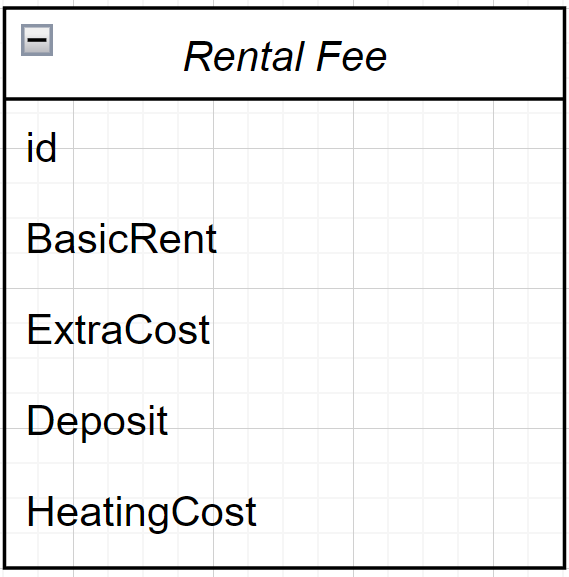
**A screen shot of a computer

Description automatically generated**

**Agent table: Subscription table:**

****

**RentalFee table:**

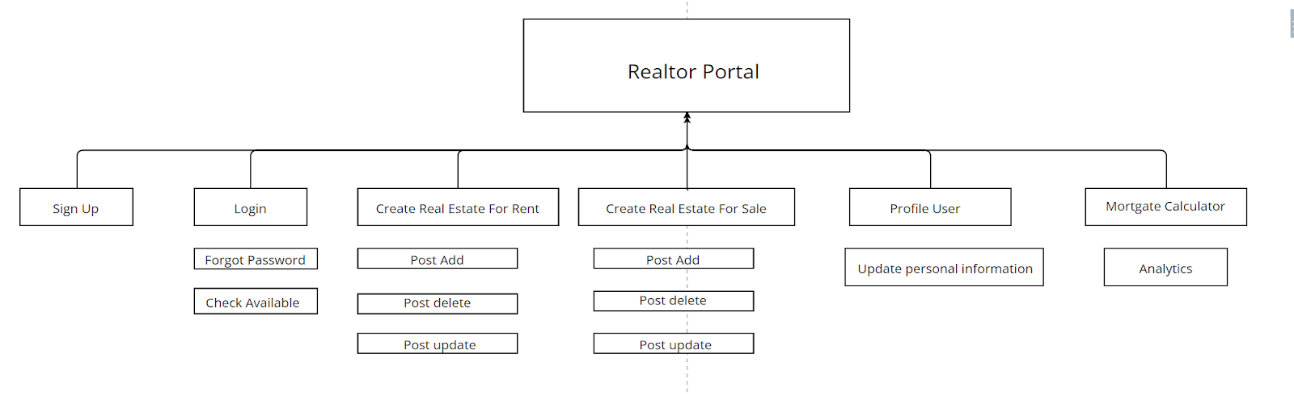
****

**AccomodationDetail table:**

**A screenshot of a computer screen

Description automatically generated**

**3.4.2 Functional Decompostion Diagram**



**Figure 7: Functional Decomposition Diagram**

**Visitor Section**

·       Browse categories

·       Basic search on homepage

·       Advanced search with additional parameters

·       Featured listings on homepage along with thumbnail images

·       Mortgage calculator

·       Static information pages like news, FAQ, about us etc.

·       Contact form with email notification

·       Signup (paid or free as per rules set by admin) for private sellers

·       and agents

**Private Seller Section**

·       Manage Private Seller profile

·       Private Sellers can post ads (buy/sell/rent) as per the package

·       they are subscribed to

·       Private Sellers can edit or delete their ad listings

·       Upload multiple images for the listings

·       Automatic resizing of photos to reduce dimensions and size

·       Purchase additional ad package on expiry

·       Agent Section

**Manage Agent profile**

·       Agents can post ads (buy/sell/rent) as per the package they are

·       subscribed to

·       Agents can edit or delete their ad listings

·       Agents can also upload multiple images for the listings

·       Automatic resizing of photos to reduce dimensions and size

·       Agents can choose to hide email address by allowing interested

·       parties to contact the poster through an online form

·       Purchase additional ad package on expiry

**Admin Section**

·       Configuration Options

·       Automatic expiration of ads after a configurable duration

·       Set the number of ads/images to be displayed per page

·       Currency symbol to use

·       Enable/disable featured ads and extended ads

**Realtors Portal**

·       Edit admin password

·       Configure Paypal payment settings

·       Categories &amp; Listings Management

·       Add/edit/delete/activate/deactivate unlimited listings

·       Add/edit/delete/activate/deactivate categories

·       Add/edit/delete countries

·       Add/edit/delete regions

·       Add/edit/delete cities

·       Add/edit/delete areas

·       Search for classified ads by keywords, categories, cities, status

·       Admin can view detailed info for all listings

**Members Management**

·       Add/edit/delete/activate/deactivate subscription packages

·       Add/edit/delete/activate/deactivate private sellers

·       Add/edit/delete/activate/deactivate agents

·       Add/edit/delete/activate/deactivate admin users

·       Confirmation email on registration of a member

·       Notification email to the member on expiry of ad package

**Reports**

·       View payment transaction history by date range with payment

·       details

·       View statistics such as

·       Payments received today

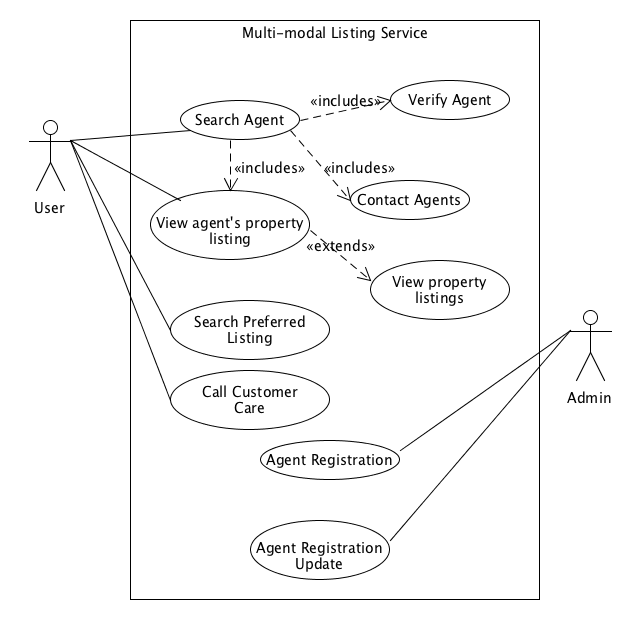
·       Number of unapproved/unverified/approved listings

·       Number of categories

·       Number of private sellers

·       Number of agents

**3.4.3. Use case diagram**

****

**Figure 8: Use – case diagram**

**3.5 Design Pattern**

**3.5.1. CQRS Pattern**

In the context of the realtors' portal project, the CQRS (Command Query Responsibility Segregation) pattern can be utilized to enhance the system's architecture and improve performance and scalability. CQRS is a design pattern that separates the responsibility of handling commands (write operations) from that of handling queries (read operations). It enables the system to optimize and scale each aspect independently, leading to improved performance and flexibility. Here's how CQRS can be applied to the realtors' portal project:

* Command Side: The command side of the CQRS pattern deals with write operations, such as creating, updating, and deleting data. In the realtors' portal, commands can include actions like creating a new property listing, updating a realtor's profile, or deactivating a listing. The command side typically involves the following components:
* Command Handlers: These components receive and process incoming commands. They validate the command, perform necessary business logic, and update the corresponding data.
* Domain Model: The domain model encapsulates the business logic and rules related to the commands. It ensures that the commands are executed correctly and enforce any necessary validations or constraints.
* Command Bus: The command bus is responsible for dispatching commands to the appropriate command handlers. It decouples the sender of the command from the actual processing logic.
* Query Side: The query side of the CQRS pattern is responsible for handling read operations, such as fetching data for displaying property listings, retrieving realtor profiles, or searching for properties. The query side typically involves the following components:
* Query Handlers: These components receive queries and fetch the required data from the appropriate data sources. They transform the data into a format suitable for presentation.
* Read Model: The read model represents a denormalized and optimized view of the data specifically designed for efficient querying. It may involve separate data structures or databases optimized for read operations.
* Query Bus: The query bus handles incoming queries and routes them to the appropriate query handlers. It decouples the sender of the query from the actual data retrieval logic.

**3.5.2 Repository Pattern**

* In the realtors' portal project, the Repository pattern can be employed to provide a structured and consistent way of accessing and managing data from various data sources. The Repository pattern acts as an intermediary between the data source (such as a database) and the application's business logic, encapsulating the data access logic within reusable components.

**3.5.3 Unit of Work**

* In the context of a realtor's portal, the Unit of Work pattern can be applied to manage interactions with the database, ensuring consistency and atomicity in operations related to properties, clients, transactions, and other entities typically found in a real estate application.
* By utilizing the Unit of Work pattern in your realtor's portal MVC API, you can ensure that database transactions are managed effectively, promoting data consistency and simplifying your codebase's structure. This pattern facilitates the organization and maintenance of your application's data access layer.

**3.6 UI Design**

*A realtor portal’s user interface (UI) design plays a crucial role in providing an intuitive, efficient, and visually appealing experience for real estate agents, clients, and other users.*

* Property Listings: The core feature of a realtor portal is to display property listings. The UI should present property details such as images, descriptions, pricing, location maps, and any other relevant information in a clear and organized manner. Users should be able to filter and search for properties based on criteria like location, price range, property type, and amenities.

A close-up of a post

Description automatically generated

**Figure 9: Property Listing**

* Search and Filters: A prominently placed search bar allows users to quickly find properties by entering keywords, addresses, or specific criteria. Advanced filtering options enable users to refine search results based on parameters such as property type, size, number of bedrooms, features, and more.

A website page with a house and palm trees

Description automatically generated

**Figure 10: Search and Filter**

* User Profiles and Dashboards: Realtor portals often include personalized user profiles and dashboards. Agents and clients can log in to access their account information, saved searches, favorite properties, viewing appointments, transaction history, and other relevant data. Dashboards provide a centralized hub for managing property listings, client communications, appointments, and tasks.

A screenshot of a computer

Description automatically generated

**Figure 11: User Profile**

* Responsive Design: With the increasing use of mobile devices, responsive design is essential for ensuring that the realtor portal's UI adapts seamlessly to different screen sizes and resolutions. A mobile-friendly UI enhances accessibility and allows users to access the portal on smartphones and tablets while on the go.
* Virtual Tours and Multimedia: Incorporating virtual tours, 360-degree photos, and videos enhances the visual experience and provides users with a more immersive preview of properties. Users can virtually explore properties from the comfort of their homes, gaining a better understanding of the layout, design, and ambiance.
* User Registration (Sign-Up):Provide a registration form where users can enter their details such as name, email address, password, phone number, and any other required information.

A screenshot of a computer screen

Description automatically generated

**Figure 12: Sign Up Page**

A screenshot of a login form

Description automatically generated

**Figure 13: Login Page**

* Feedback and Reviews: Integrating feedback and review mechanisms allows users to share their experiences, rate properties, and provide feedback on agent services. Positive reviews and testimonials can build trust and credibility, while constructive feedback helps improve the user experience and service quality.
* Post Editing and Management: Enable authorized users to edit and update existing posts, including modifying text, images, and other attributes.

A screenshot of a website

Description automatically generated

**Figure 14: Post Management**

* Accessibility and Compliance: Prioritizing accessibility standards ensures that the realtor portal is usable by individuals with disabilities and complies with relevant accessibility guidelines. Features such as text alternatives for images, keyboard navigation, and clear navigation structures contribute to a more inclusive user experience.

A screenshot of a computer

Description automatically generated

**Figure 15: Property Detail**

Property Detail: Feature which necessary for user to buy or rent or property, they need to know how to contact with the properties’owner.

Contact Information Section:

* Include a dedicated section on the property detail page that displays the contact information of the property owner or listing agent.
* Provide options for users to contact the owner or agent via phone, email, or messaging platforms.
* Display the agent's name, agency (if applicable), phone number, email address, and office address, if available.

**This page intentionally left blank**

**CHAPTER 4. TESTING AND PERFORMANCE APPRAISAL**

**Testing and performance appraisal are crucial aspects of ensuring the reliability, functionality, and efficiency of a realtor portal. Here's how you can approach testing and performance appraisal for a realtor portal:**

* Unit Testing:
  + Write unit tests for individual components, such as controllers, services, repositories, and utility functions.
  + Use mocking frameworks to simulate dependencies and isolate units for testing.
  + Ensure that unit tests cover edge cases, error handling, and boundary conditions.
* Integration Testing:
  + Conduct integration tests to verify interactions between different modules and components within the realtor portal.
  + Test scenarios such as user authentication, post management, property search, and database interactions.
  + Verify the behavior of the portal under various configurations and environments.
* End-to-End Testing:
  + Perform end-to-end tests to validate the functionality and user experience of the realtor portal from start to finish.
  + Use automated testing frameworks to simulate user interactions and test user flows across different pages and features.
  + Test common user scenarios, such as signing up, searching for properties, submitting inquiries, and viewing property details.
* Regression Testing:
  + Implement regression tests to ensure that recent changes or updates to the realtor portal do not introduce new bugs or regressions.
  + Re-run existing tests and compare results against baseline expectations to detect any deviations or unexpected behavior.
* Performance Testing:
  + Conduct performance testing to evaluate the responsiveness, scalability, and reliability of the realtor portal under normal and peak load conditions.
  + Use load testing tools to simulate concurrent user traffic and measure the portal's performance metrics, such as response time, throughput, and resource utilization.
  + Identify performance bottlenecks, scalability limitations, and areas for optimization, such as database queries, network latency, and server-side processing.

Performance Appraisal:

* Response Time:
  + Measure the average response time of critical operations within the realtor portal, such as page loading, search queries, and form submissions.
  + Set performance targets and benchmarks for acceptable response times based on user expectations and industry standards.
* Scalability:
  + Assess the realtor portal's ability to handle increasing user traffic and workload without significant degradation in performance or service disruptions.
  + Conduct stress testing and scalability tests to determine the portal's capacity limits and identify potential scalability bottlenecks.
* Resource Utilization:
  + Monitor and analyze resource utilization metrics, such as CPU usage, memory consumption, disk I/O, and network bandwidth.
  + Optimize resource allocation and configuration to ensure efficient utilization of hardware resources and minimize wastage.
* Reliability and Availability:
  + Evaluate the realtor portal's reliability and availability by measuring uptime, downtime, and error rates.
  + Implement monitoring and alerting mechanisms to detect and respond to performance issues, errors, and failures in real-time.
* User Experience:
  + Solicit feedback from users, stakeholders, and real estate professionals to assess their satisfaction with the realtor portal's performance and usability.
  + Conduct usability testing and user surveys to identify pain points, usability issues, and areas for improvement in the user experience.
* Optimization and Continuous Improvement:
  + Use performance appraisal findings to prioritize optimization efforts and enhancements to the realtor portal.
  + Implement a continuous improvement process to iteratively refine the portal's performance, reliability, and user experience over time.

**4.1 Software Implementation**

* *Configure the server:*
  + Choose a web server to host the website (Recommendation: Apache, Nginx, Express, …)
  + Install and configure the selected web server according to its documentation.
* *Deploy the Application files:*
  + Copy the contents of the `dist` directory to the server.
  + Ensure the server is configured to serve the files.
* *Setup Domains and DNS:*
  + Configure the DNS settings to point to the server’s IP address.
  + Updating necessary domain-related configurations on the server.
* *Test the Deployments:*
  + Accessing the website through the configured domain or server IP address using a web browser.
  + Verify that application loads correctly.
  + Test features and ensure there are no errors or issues.

**4.2 Deployment – Testing and Performance Appraisal**

**4.2.1. Testing Schedule**

**A screenshot of a computer screen

Description automatically generated**

**Figure 16: Testing Schedule**

**4.2.2 Testing Details:**

**Test Case 1: Property listings**

1. **Objective:** Verify that property listing display accurate information.
2. **Steps to procedure:**

* Navigate to the "Properties" section of the website.
* Select a property listing.
* Verify that the property details, including description, photos, videos, and specifications, are accurate and up-to-date.

1. **Expected result:** Property listing displays correct information as described.
2. **Actual result:** All property listings display accurate information. No discrepancies found.
3. **Status:** Pass

**Test Case 2: Search functionality**

1. **Objective:** Test the search functionality to ensure it returns relevant results.
2. **Steps to procedure:**

* Enter search criteria such as location, price range, property type, etc.
* Submit the search query.
* Examine the search results.

1. **Expected result:** Search results should match the entered criteria and display relevant properties.
2. **Actual result:** Search functionality returns accurate results based on the provided criteria. No issues detected.
3. **Status:** Pass

**Test case 3: Contact options**

1. **Objective:** Verify that users can easily contact real estate agents or property owners.
2. **Steps to procedure:**

* Locate the contact information for a property listing.
* Use the provided contact options (e.g., contact form, phone number, email) to reach out to the agent or owner.
* Verify successful communication or prompt response.

1. **Expected result:** Contact options should be functional, and users should be able to contact agents or owners without difficulty.
2. **Actual result:** Contact options work as expected, and communication with agents or owners is successful.
3. **Status:** Pass

**Test case 4: Usability**

1. **Objective:** Evaluate the usability of the website interface.
2. **Steps to procedure:**

* Navigate through various sections of the website.
* Test the website on different devices and screen sizes.
* Attempt common user tasks such as searching for properties, viewing listings, and contacting agents.

1. **Expected result:** The website interface should be intuitive, responsive, and user-friendly.
2. **Actual result:** The website interface is easy to navigate, responsive on different devices, and facilitates user tasks effectively.
3. **Status:** Pass

**Summary:**

All critical functionalities were tested and found to be working correctly without any significant issues. The website interface is intuitive and user-friendly, providing a seamless experience for users.

**Issue identified:** None

**Recommendation:** Continuously monitor website performance and user feedback for ongoing improvement.