

CONTENTS	
1	INTRODUCTION
1.1	SYSTEM INTRODUCTION..... 1
1.2	BACKGROUND OF THE SYSTEM..... 1
1.3	OBJECTIVES OF THE SYSTEM..... 2
1.4	SIGNIFICANCE OF THE SYSTEM..... 2
2	OVERALL DESCRIPTION
2.1	PRODUCT PERSPECTIVE
2.2	PRODUCT SCOPE 3
2.3	PRODUCT FUNCTIONALITY
2.4	USERS AND CHARACTERISTICS 5
2.5	OPERATING ENVIRONMENT..... 8
3	SPECIFIC REQUIREMENTS
3.1	FUNCTIONAL REQUIREMENTS 10
3.2	BEHAVIOUR REQUIREMENTS..... 14
3.3	EXTERNAL INTERFACE REQUIREMENTS 15
4	OTHER NON-FUNCTIONAL REQUIREMENTS.....
4.1	PERFORMANCE REQUIREMENTS..... 22
4.2	SAFETY AND SECURITY REQUIREMENTS
4.3	SOFTWARE QUALITY ATTRIBUTES 24
5	DESIGN DESCRIPTION
5.1	COMPOSITE VIEWPOINT 26
5.2	LOGICAL VIEWPOINT 28
5.3	INFORMATION VIEWPOINT 30
5.4	INTERACTION VIEWPOINT 40
5.5	STATE DYNAMICS VIEWPOINT 41
5.6	ALGORITHM VIEWPOINT Error! Bookmark not defined. 43

1 Introduction

1.1 System Introduction

Pak Properties is a dynamic web-based platform designed to transform the way individuals engage in property transactions, including buying, selling, and renting. The platform caters to the needs of buyers, sellers, and renters by providing a centralized system that simplifies property management and search processes. One of its standout features is the integration of a robust video management system, allowing sellers to upload detailed video tours of their properties. This functionality empowers buyers and renters to explore properties virtually, reducing the need for physical visits and enabling informed decision-making. By offering rich property descriptions and visual media, the platform ensures a transparent and interactive experience that bridges the gap between property owners and seekers.

In addition to its video management capabilities, Pak Properties incorporates essential features such as property categorization, advanced search filters, and real-time updates to enhance usability. The categorization system organizes listings into well-defined categories such as homes, offices, apartments, and plots, making it easier for users to find suitable options. The advanced search filters allow users to refine their searches by location, price range, property type, and more. Real-time updates ensure that users receive the latest information on property availability and changes. By integrating these features with a responsive design, Pak Properties delivers an intuitive and accessible platform tailored for a diverse audience, including first-time buyers, experienced investors, and property managers.

1.2 Background of the System

The real estate market has experienced a dramatic shift toward digital platforms as property seekers increasingly rely on online tools to streamline their buying, selling, and renting processes. Existing platforms like Property Hub (on RMS), Zameen.com, and local property portals have addressed some of these needs by providing basic listings and property search functionalities. However, these solutions often fall short in offering immersive experiences or tools to simplify property rentals. Most lack comprehensive video management systems that allow users to explore properties in detail from the comfort of their homes, and their rental management features are often limited, leaving a gap in the market for a more holistic solution. This creates opportunities for platforms like Pak Properties to redefine the real estate experience by addressing these shortcomings with user-centric innovations.

Pak Properties distinguishes itself by incorporating a robust video management system and an efficient rental management module, features that go beyond the traditional property listing websites. The inclusion of video tours allows property seekers to virtually explore listings, offering

greater transparency and reducing the reliance on in-person visits. Additionally, the system's advanced property sorting options enable users to search and filter listings based on their specific needs, such as location, price range, and property type, ensuring a more tailored experience.

1.3 Objectives of the System

Here's the updated list of objectives for **Pak Properties**, excluding blog functionality and map integration:

The primary objectives of Pak Properties are:

- To create a unified platform for buying, selling, and renting properties.
- To integrate a video management system for property tours.
- To ensure secure and role-based access for different user types (Admin, Users).
- To implement advanced search filters to help users find properties based on criteria like location, price, and property type.
- To allow users to add, edit, and manage property listings easily.
- To enable users to save their favorite properties for future reference.
- To deliver an intuitive dashboard for both users and admins for efficient management of listings and accounts.
- To support property categorization for better organization (e.g., homes, apartments, offices, plots).
- To maintain a responsive and mobile-friendly interface for accessibility across devices.
- To ensure a user-friendly design for seamless navigation and interaction.
- To provide a feedback mechanism for users to report issues or share suggestions.
- To offer property sorting options, such as by price, location, or newest listings.

1.4 Significance of the System

Pak Properties holds significant importance in transforming the real estate market by addressing the limitations of current digital platforms. Traditional property portals often lack interactive

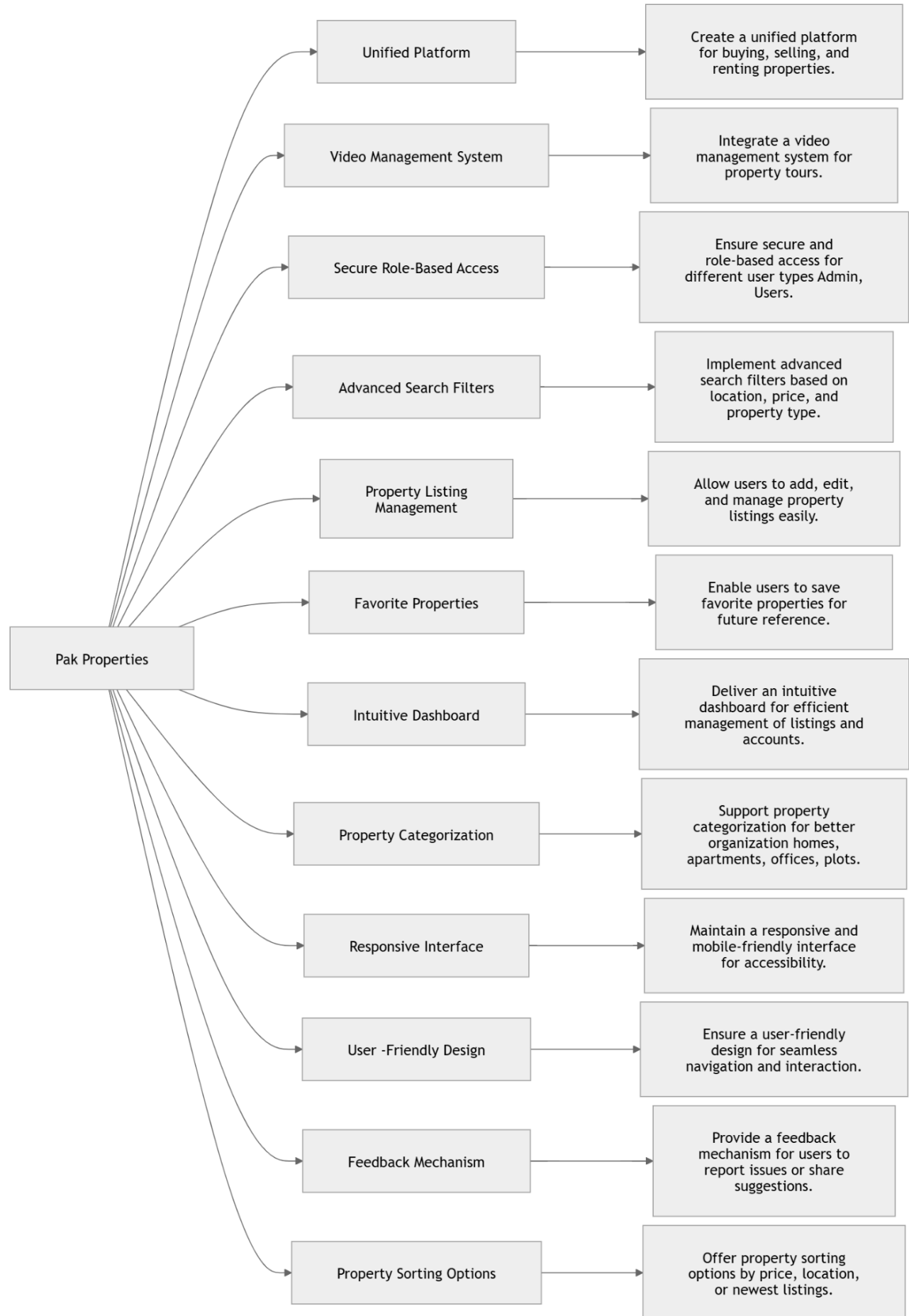
features and streamlined tools for rental and property management. Pak Properties bridges this gap by offering an innovative platform that integrates video management for property tours, enabling users to virtually explore properties in a convenient and engaging way. This reduces the reliance on physical visits and empowers buyers and renters to make informed decisions efficiently. The platform's advanced search filters and categorization tools further simplify the property search process, allowing users to tailor their searches to specific needs such as price range, property type, or location.

Additionally, the rental management tools provided by Pak Properties streamline the process for landlords and tenants, offering an organized approach to managing listings and agreements. By ensuring role-based access control, the system guarantees secure and personalized experiences for different user types, such as admins and regular users. The platform's user-friendly interface and responsive design ensure accessibility across all devices, making it a valuable tool for a wide range of demographics, including property owners, renters, and investors.

2 Overall Description

2.1 Product Perspective

Pak Properties is a new, self-contained platform developed to address the evolving needs of the real estate market. Unlike existing property platforms such as Zameen.com or Property Hub, which primarily focus on listing properties, Pak Properties provides a more comprehensive solution by integrating a video management system, advanced search filters, and rental management tools. It is designed to serve as an all-in-one platform for property buying, selling, and renting, improving transparency and efficiency throughout the process. The system will allow users to interact seamlessly with property listings, watch video tours, ensuring an enhanced experience compared to traditional methods. This product is self-contained, focusing on property listings and rental management, without the need for external integrations.



2.2 Product Scope

The scope of Pak Properties includes the creation of a platform that facilitates property transactions (buying, selling, and renting) through online listings. The platform allows users to upload, view, and manage property listings, complete with video tours for enhanced visualization. The system includes advanced search and filtering capabilities to help users find properties based on various criteria. It also supports rental management. However, the scope does not include payment processing systems, which sets boundaries for the platform's functionality. The platform will not support advanced features like virtual 3D property tours or complex financial transaction features such as payments, mortgages, or lending.

2.3 Product Functionality

The primary functionalities of Pak Properties include:

Property Listings Management: Users can create, edit, and remove property listings.

Video Management: Sellers can upload and display video tours of properties for potential buyers and renters.

Advanced Search and Filters: Users can search properties based on price, location, type, and other criteria.

Favorite Properties: Users can save their preferred properties for future reference.

User Role Management: Different users (admins, buyers, sellers) have role-based access to system features.

Responsive Design: The platform is fully optimized for both desktop and mobile devices.

User Dashboard: Provides a personalized view of listings, favorites, and account settings for both buyers and sellers.

Admin Dashboard: Admins can manage user roles, view property statistics, and monitor platform activity.

2.4 Users and Characteristics

1. Admin

Characteristics:

- **Role-Based Access:** Admins have the highest level of control over the platform. They are

responsible for managing users, approving property listings, ensuring system functionality, and monitoring overall performance.

- **Frequency of Use:** Admins will use the platform regularly to perform tasks such as user management, property approval, and content moderation. They have access to the backend system where they can make changes and track user activities.
- **Technical Expertise:** Admins are expected to have a high level of technical expertise, particularly in understanding system administration, user roles, and platform functionality. They should be comfortable navigating the admin interface and managing multiple user accounts.
- **Responsibilities:**
 - Approve or reject property listings submitted by sellers.
 - Manage user roles (assign admin, seller, or buyer roles).
 - Address any system issues or user concerns.
- **Privileged Access:** Admins can access all areas of the platform, including the backend, user data, system settings, and property statistics.

Importance: The admin is the most important user of the system as they are responsible for maintaining order, ensuring the platform runs smoothly, and managing all other users. Their decisions influence the content displayed on the platform and overall user experience.

2. Seller

Characteristics:

- **Role-Based Access:** Sellers are property owners, real estate agents, or agencies who want to list properties for sale or rent. They have access to a seller dashboard where they can manage their property listings.
- **Frequency of Use:** Sellers will log in periodically to add or update property listings, upload video tours, and manage the visibility of their properties.
- **Technical Expertise:** Sellers need basic technical skills, such as familiarity with uploading property details (texts, images, and videos) and understanding the platform's search functionality. The system is designed to be user-friendly, so advanced technical expertise is not required.
- **Responsibilities:**
 - Create, edit, and remove property listings.
 - Upload property images and video tours.
 - Respond to inquiries from potential buyers or renters.
 - Monitor the performance of their listings and update details as necessary.
- **Privileged Access:** Sellers can only manage their own listings, view interaction details, and access basic platform functionalities that pertain to their properties. They do not have access to admin features or other user accounts.

Importance: Sellers are essential to the platform's success, as they provide the content (property listings) that drives user engagement. Without active sellers, the platform would lack the property data needed to attract buyers and renters.

3. Buyer/Renter

Characteristics:

- **Role-Based Access:** Buyers and renters are the primary consumers of the platform who are looking for properties to purchase or rent. They can search and browse through available property listings, view details, and watch video tours.
- **Frequency of Use:** Buyers and renters will use the platform frequently, especially when they are in the market for a property. Their interactions are often centered around property searches, saving favorites, and contacting sellers.
- **Technical Expertise:** Buyers and renters do not need advanced technical skills. The platform is designed to be intuitive and accessible to users with basic computer literacy. A simple understanding of navigation, search filters, and viewing property details is sufficient.
- **Responsibilities:**
 - Search for properties based on filters such as location, price, and property type.
 - View property listings and associated media (images, video tours).
 - Save favorite properties for future reference.
 - Contact sellers for more details or to arrange property viewings.
- **Privileged Access:** Buyers and renters have access to property listings, search functionality, and video tours. They cannot add or modify listings or interact with administrative functions.

Importance: Buyers and renters are crucial users because they are the end consumers who drive demand for properties. Their experience on the platform needs to be seamless and user-friendly to encourage active use and return visits. Ensuring that this user group has an intuitive, informative, and responsive platform is key to the platform's success.

4. Guest User

Characteristics:

- **Role-Based Access:** Guest users are visitors who have not registered or logged into the system. They can view property listings but have limited interaction capabilities.
- **Frequency of Use:** Guest users typically use the platform sporadically, either out of curiosity or during their initial search phase.
- **Technical Expertise:** No technical expertise is required. Guest users are simply browsing through listings.
- **Responsibilities:**

- Browse property listings without the ability to save favorites or make direct inquiries to sellers.
 - View public property details and video tours.
- **Privileged Access:** Guests have minimal access to the platform's features and cannot interact with sellers or save properties.

Importance: Guest users are less critical in terms of platform management, but they serve as potential leads who may convert into registered buyers or sellers. Ensuring that the platform attracts and retains guest users is important for growing the user base.

User Importance and Prioritization

The **Admin** and **Seller** roles are the most important to satisfy because they directly influence the platform's content, operations, and overall functionality. Without active and engaged sellers, the platform would have no property listings, and without admins, the platform would lack control, security, and effective management. The **Buyer/Renter** role is also highly important, as it represents the core audience that uses the property listings and drives engagement. Ensuring that their experience is easy, informative, and seamless is critical for platform retention.

The **Guest User** has the least impact on the platform's overall functionality, but they serve as potential users who may later sign up, so providing an easy and welcoming entry point for guests is also a priority.

2.5 Operating Environment

Hardware Platform

Pak Properties will be hosted on cloud-based infrastructure to ensure scalability, availability, and performance across various geographical locations. The application itself is designed to be responsive, meaning it will be optimized for a wide variety of devices, including:

- **Desktop Computers:** Users will access the platform using web browsers on desktop computers, which will need a stable internet connection to interact with the platform.
- **Mobile Devices:** The platform will be fully responsive and optimized for use on smartphones and tablets. This will ensure accessibility and a user-friendly experience on devices with varying screen sizes.
- **Server Requirements:** The back-end of the application will be hosted on a cloud platform (e.g., Amazon Web Services (AWS), Microsoft Azure, or Google Cloud) with scalable server resources

Operating System and Versions

Pak Properties is designed to be platform-independent, meaning it will run on major operating systems without compatibility issues. The platform will support:

- **Server-Side:**
 - **Operating Systems:** The server-side components of the platform will run on Linux-based servers, such as Ubuntu, CentOS, or similar distributions, which are popular for web hosting due to their stability and security.
 - **Web Server:** Apache or Nginx will be used as the web server to manage HTTP requests and serve the application to users.
 - **Backend Framework:** The server-side application will be built using ASP.NET Core, which supports cross-platform deployment, allowing it to run on both Windows and Linux environments.
- **Client-Side:**
 - **Operating Systems:** Users will access the platform through their web browsers on a variety of operating systems, including:
 - **Windows**
 - **macOS**
 - **Linux**
 - **Mobile Platforms:** The platform will also be accessible on Android and iOS devices through web browsers like Google Chrome, Safari, or Firefox.

Software Components

Pak Properties will depend on a variety of software components and technologies for its operation. These components work together to ensure a smooth and efficient user experience. The key software components include:

- **Backend Framework:** The platform will be developed using **ASP.NET Core 6** for the back-end. This framework provides a robust, high-performance environment for building scalable web applications.
 - **.NET Core SDK:** Version 6 or later
 - **C#:** The primary programming language for backend development
- **Frontend Technologies:** For the front-end, **HTML**, **CSS**, and **JavaScript** will be used, along with libraries such as:
 - **Bootstrap** or **Tailwind CSS** for responsive design and styling.
 - **jQuery** (optional) for dynamic content manipulation.
 - **Video.js** or similar JavaScript libraries for handling video tour playback.
 - **Chart.js** for displaying charts and property statistics.
- **Database:** The system will use **SQL-Server** as its relational database management system (RDBMS) to store property listings, user data, and related information.
- **Version Control:** Git will be used for source code management, with **GitHub** or **GitLab** as the repository hosting service.

Minimum Platform Requirements

To ensure optimal operation, users must meet the following minimum system requirements:

- **Web Browser:** Latest versions of Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge. The platform will be optimized for these browsers to ensure a consistent and responsive user experience.
- **Internet Connection:** A stable internet connection with a minimum speed of 1 Mbps to ensure fast loading times and smooth video streaming. Users with slower connections may experience delays in loading images or videos.
- **Mobile Devices:** The platform will be fully optimized for mobile browsers, supporting Android and iOS devices. Users should have smartphones with at least **2 GB of RAM** and a screen resolution of **720p or higher** for the best experience.

3 Specific Requirements

3.1 Functional Requirements

3.1.1 User Management

This area outlines the functionality related to user registration, authentication, and profile management. The platform supports role-based access, with different privileges assigned to users depending on their role (Admin, Seller, Buyer/Renter).

- **User Registration:**

The system must allow new users to create an account by providing necessary details (e.g., name, email, password). A confirmation email will be sent to the user to verify the email address.

 - Fields required: Name, email, password, user type (buyer/seller).
 - Email verification to confirm the account before activation.
- **Login/Logout:**

Registered users must be able to log into the platform using their email and password. After successful login, users will be redirected to their respective dashboards.

 - Login via email and password.
 - Option to reset password if the user forgets it.
- **Profile Management:**

Users can view and edit their profile, including personal information, contact details, and profile picture. Sellers can also manage business-related information in their profiles.

 - Buyers/renters can edit their personal contact information.
 - Sellers can add business details (real estate agency name, contact information).

- **Role-Based Access Control:**

There should be different user roles with varied privileges:

- **Admin:** Full access to all features, including user management, property management, reports, and system settings.
- **Seller:** Access to property listing, video uploads, editing and removing properties, and viewing their listings.
- **Buyer/Renter:** Ability to browse properties, save favorites, and contact sellers.

3.1.2 Property Management

This area focuses on how the system handles property listings, including adding, editing, deleting, and searching for properties. It also includes functionalities for managing property media like images and videos.

- **Add New Property Listing:**

Sellers must be able to add new property listings by entering property details like title, description, price, property type (apartment, house, office, etc.), and location. They can also upload images and video tours of the property.

- Fields required: Title, description, price, address, property type, number of rooms, area (sq. ft.), images, and video.

- **Edit Property Listing:**

Sellers should be able to edit their listings after submission. They can update property details such as price, description, images, and video.

- **Remove Property Listing:**

Sellers should have the ability to delete their property listings if they no longer wish to offer them for sale or rent.

- **Property Search and Filters:**

Buyers and renters must be able to search for properties based on various criteria, including location, price, property type, number of rooms, and more. Filters should allow the user to refine the search results and view properties that match their needs.

- Search criteria: Location (city, neighborhood), price range, property type, number of rooms, etc.
- Ability to save searches and receive notifications for new listings that meet the search criteria.

- **Property Details and Media:**

Property pages should display detailed information, including the property's description, features, and high-quality images. Video tours should also be available for each property listing.

- Display property images in a gallery.
- Embed video tours or provide links to view them.

- **Property Sorting:**

The platform should allow users to sort property listings based on different parameters such as price (low to high, high to low), newest listings, and location proximity.

- **Favorite Properties:**

Buyers and renters can save their favorite properties for future reference. The system should allow them to view a list of saved properties in their dashboard.

3.1.3 Video Management

This functional area deals with the uploading, viewing, and management of video tours for each property listing.

- **Video Uploading:**

Sellers must be able to upload video tours of their properties. The system should accept common video formats such as MP4, AVI, and MOV. The platform should handle video compression and ensure that video files do not exceed a specified size.

- **Video Viewing:**

Buyers and renters should be able to view video tours associated with each property. The videos should be embedded directly on the property listing page.

- **Video Management:**

Sellers must be able to manage their video uploads, including updating or removing existing videos.

3.1.4 Search and Filter System

The search and filter functionality is crucial for improving user experience by enabling quick and accurate property searches.

- **Basic Search:**

Users must be able to search properties by keywords (e.g., location, price range, property type) and view results that match their search criteria.

- **Advanced Filters:**

The system should offer advanced filters for users to narrow down search results based on more specific criteria, such as:

- Property type (house, apartment, office)
- Price range
- Number of bedrooms/bathrooms

- Area size (in sq. ft.)
 - Availability status (available, sold, rented)
- **Search Results Page:**
Search results should be displayed in a grid or list view, showing key information like property name, price, and thumbnail image. Users should be able to sort these results by various parameters.

3.1.5 Rental Management

This feature supports landlords and tenants by simplifying the process of managing rental contracts and agreements.

- **Rental Listing Management:**
Sellers who are listing rental properties must be able to create and manage rental agreements. This includes specifying the rental price, contract duration, and other terms (e.g., deposit required, maintenance responsibilities).
- **Contract Creation:**
Landlords and tenants should be able to digitally create and agree upon rental contracts through the platform. The system should support generating contract templates with customizable fields for both parties to fill in.
- **Contract Renewal/Termination:**
Landlords and tenants can manage contract renewal or termination through the platform. Automatic notifications can be sent to both parties to remind them when a contract is nearing its end.

3.1.6 Reporting and Analytics

The system must generate reports and provide analytics for both admins and sellers.

- **Admin Reports:**
Admins should have access to a dashboard with key metrics, such as:
 - Total number of listings.
 - Active user count (buyers, sellers).
 - Property views and engagement metrics.
- **Seller Reports:**
Sellers should have access to performance reports for their listings, such as:
 - Number of views per listing.
 - Number of inquiries or messages received for their properties.

3.1.7 Security and Privacy

This area outlines the security measures that must be implemented to protect user data and ensure privacy.

- **User Authentication and Authorization:**

The platform must have secure login mechanisms, including encrypted passwords and two-factor authentication (optional for enhanced security).

- Role-based access control ensures that only authorized users can perform actions specific to their roles.

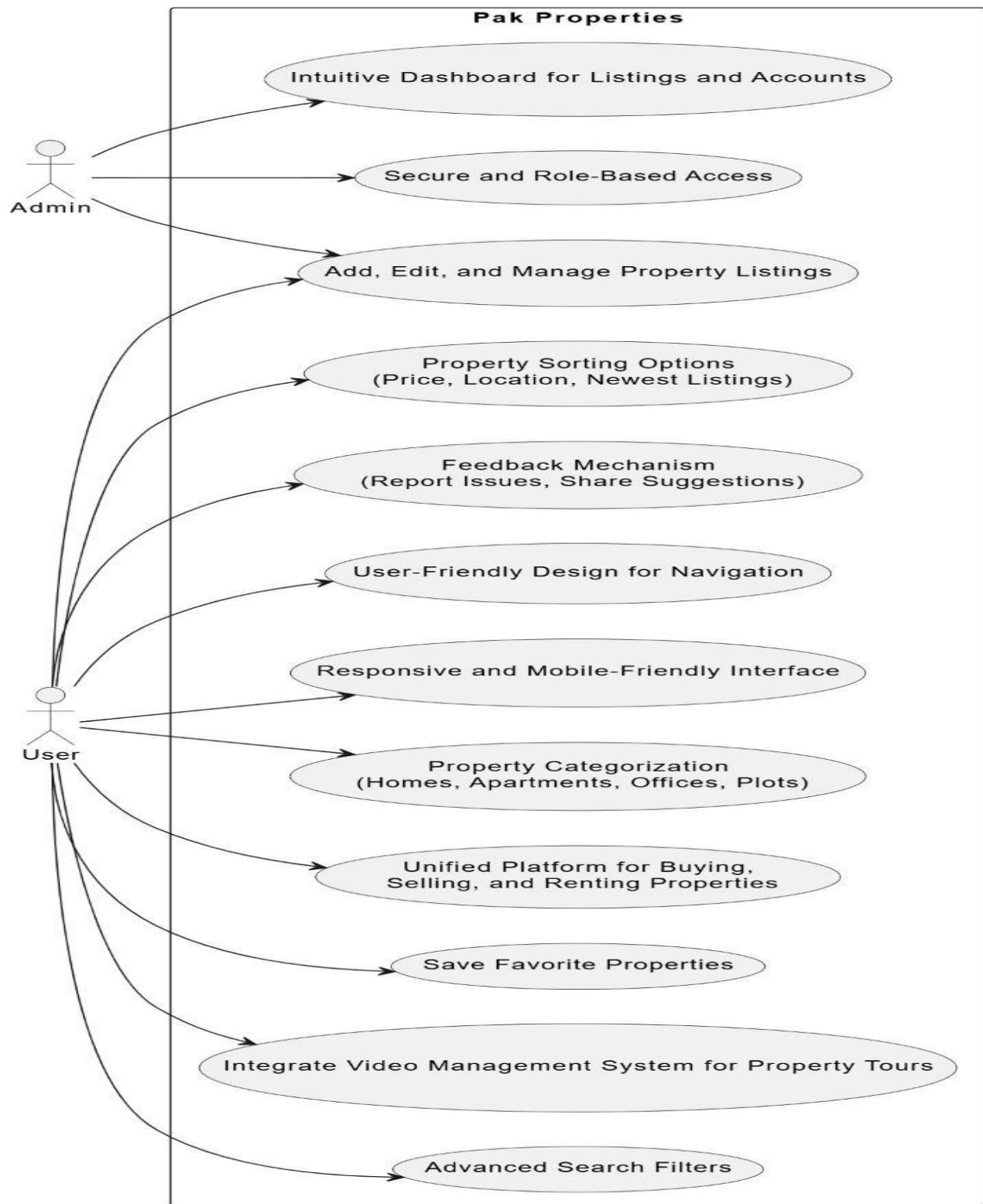
- **Data Protection:**

User data, including personal details and property information, must be stored securely, and sensitive information must be encrypted both at rest and in transit.

- **Secure File Uploads:**

The platform must have security measures in place to ensure that uploaded files (images, videos) are scanned for malware before being accepted.

3.2 Behaviour Requirements



3.3 External Interface Requirements

3.3.1 User Interfaces

1. Common User Interface Elements

Regardless of the user type (Admin, Seller, or Buyer/Renter), the following interface components will be common across the platform:

- **Navigation Bar:**

The main navigation bar at the top of each screen will provide access to the core features of the platform, including:

- **Home:** Link to the homepage.
- **Search Bar:** Allows users to search for properties.
- **Profile:** Access to the user's account information and settings.
- **Logout:** Button to log out of the system.

- **Footer:**

The footer will contain links to additional resources and pages, including:

- **About Us**
- **Contact Information**
- **Privacy Policy**
- **Terms of Service**

- **Buttons:**

Standard buttons will be used throughout the platform to trigger common actions. Examples include:

- **Save:** Used to save form data (e.g., saving a new property listing, saving profile updates).
- **Cancel:** Used to cancel an ongoing action and return to the previous screen or dashboard.
- **Submit:** Used to submit forms, such as listing a property or submitting feedback.
- **Delete:** Used to delete a property listing or user account (with confirmation prompt).

- **Error Message Display:**

Error messages will be displayed clearly and in a consistent manner across the platform. These messages will:

- Be shown near the element causing the issue (e.g., incorrect form input).
- Include a brief description of the error (e.g., "Please enter a valid email address").
- Use a red or alert color to draw attention.

- Provide clear instructions on how to resolve the error.

2. User Interface for Admins

The Admin user interface will be designed to give administrators control over the system, allowing them to manage users, properties, and site performance. Key screens for admins include:

- **Admin Dashboard:**

The Admin dashboard will display key metrics, such as:

- Total number of properties listed.
- Number of active users (buyers, sellers).
- System notifications and alerts (e.g., pending user verifications, new property listings).
- Quick links to manage properties, users, and reports.

- **User Management:**

The Admin will be able to access a user management screen to:

- View and search for registered users (buyers, sellers).
- Assign or change user roles (e.g., from buyer to seller).
- Activate/deactivate user accounts.
- View user activity logs (e.g., recent logins, property interactions).

- **Property Management:**

Admins will have a screen for managing property listings, where they can:

- View all listed properties, including details like price, location, and seller information.
- Approve or reject property listings submitted by sellers.
- Edit or delete properties that violate platform rules.

- **Reports & Analytics:**

Admins will have access to detailed reports, including:

- Property statistics (e.g., most viewed properties, top-performing listings).
- User activity and engagement metrics.
- Site performance and traffic data.

- **Settings:**

Admins can configure site-wide settings, such as:

- Email notifications (when new listings are added, or account activities occur).
- Privacy and security settings.

3. User Interface for Sellers

Sellers (property owners, real estate agents) will have a user interface designed to help them manage their property listings and interactions with buyers/renters. Key screens for sellers include:

- **Seller Dashboard:**

The seller dashboard will provide an overview of their activity on the platform, including:

- Number of active listings.
- View count for each property.
- Recent inquiries or messages from potential buyers/renters.
- Quick links to manage listings and settings.

- **Property Listing Management:**

The seller will have a dedicated screen to manage their property listings, including:

- **Add New Property:** A form where sellers can input property details (price, description, type, location, etc.) and upload images/videos.
- **Edit Property:** A form to update existing listings (change price, description, or images).
- **Delete Property:** Option to remove a listing, with a confirmation prompt.
- **Upload Video Tour:** A video upload form to allow sellers to add property video tours to their listings.

- **Messages and Inquiries:**

A screen where sellers can manage and respond to messages or inquiries from buyers or renters. Features include:

- **Inbox:** A list of unread and read messages.
- **Message Detail:** A detailed view of each message, with options to reply or mark as read.

- **Seller Profile:**

Sellers can manage their account information, including:

- **Business Name** (for real estate agents).
- **Contact Information** (phone number, email).
- **Profile Picture:** A photo representing the seller or agency.

4. User Interface for Buyers/Renters

Buyers and renters will have an interface designed to search for and view properties, save favorites, and interact with sellers. Key screens for buyers/renters include:

- **Home Page:**
The home page will display featured listings and a search bar for quick property search. It may also show promotional content or highlighted properties.
- **Search and Filter:**
Buyers and renters can search for properties using various filters (location, price, property type, etc.). The screen will display a list or grid of search results, with the following features:
 - **Sort Options:** Allows users to sort results by price, newest listings, etc.
 - **Filters:** Options for narrowing search results by more specific criteria.
 - **Property Thumbnails:** Small images or videos of each property.
- **Property Details Page:**
When a buyer or renter clicks on a property listing, they will be taken to a detailed property page, which includes:
 - **Property Description:** Detailed text about the property's features and benefits.
 - **Image Gallery:** A collection of property images.
 - **Video Tour:** If available, an embedded video tour of the property.
 - **Contact Seller:** A button to send an inquiry or message to the seller.
 - **Save to Favorites:** An option to add the property to the user's favorites list for later viewing.
- **Favorites Page:**
Buyers and renters can access their list of saved properties, with quick links to view details or remove properties from the list.
- **User Profile:**
Buyers and renters can manage their personal account settings, including contact details and preferences for notifications.

5. Standard Error Messages and Alerts

In case of errors or issues, the following types of messages will be displayed across the platform:

- **Invalid Input:**
When a user submits an incorrect or incomplete form (e.g., missing required fields), an error message will appear near the affected field.
 - Example: "Please enter a valid email address."
- **Success Confirmation:**
After successfully completing an action (e.g., submitting a property listing), a confirmation message will be displayed to inform the user that the action was successful.
 - Example: "Your property listing has been submitted for approval."
- **Access Denied:**
If a user tries to access a restricted area without proper authorization, an access denied message will appear.
 - Example: "You do not have permission to access this page."

3.3.2 Hardware Interfaces

3.3.3 Software Interfaces

1. Operating System Interface

The **Pak Properties** platform will be deployed on cloud infrastructure, typically running on a Linux-based operating system for backend servers, while users will interact with the platform through web browsers on various operating systems. The operating systems involved are:

- **Client-Side Operating Systems:**
 - **Windows, macOS, and Linux** for desktop users.
 - **Android and iOS** for mobile users, where the platform will be accessible through web browsers (e.g., Chrome, Safari, Firefox).
 - The client-side browsers must support modern web standards (HTML5, CSS3, JavaScript) for the platform to render correctly.
- **Server-Side Operating System:**
 - **Linux** (e.g., Ubuntu, CentOS, or similar distributions) will be used to host the

backend server. This operating system is chosen for its stability, scalability, and security features.

- The server will also support **Apache** or **Nginx** as the web server to serve HTTP requests and manage traffic.

2. Database Interface

The **Pak Properties** platform will store data in a **relational database**, and the system will interface with the database for operations like reading and writing property listings, user data, and more. The database interface is critical for managing the data flow within the system.

- **Database Management System (DBMS):**

- **SQLServer** will be used for data storage. Both are widely used open-source relational databases that provide reliability, scalability, and efficient querying capabilities.
- The database will store:
 - **User Information:** User profiles, roles, preferences, and login credentials.
 - **Property Listings:** Details about properties, including description, price, images, videos, and status (active, sold, rented).
 - **Messages/Inquiries:** Data related to communications between sellers and buyers.
 - **Rental Contracts:** Information related to rental agreements.

- **Data Access Layer:**

- The application will use **Entity Framework Core** or **Dapper** (for .NET) to interact with the database. These tools allow the platform to map data between the application and the database, providing efficient query capabilities.

3. External APIs and Services

The platform may need to integrate with external APIs and services for functionality such as sending emails, file storage, or processing video uploads. These APIs will provide additional capabilities and improve the overall performance of the system.

- **Email Service API:**

- The system will use third-party email services like **SendGrid**, **Mailgun**, or **Amazon SES** to handle email notifications and confirmations.

- **Purpose:** Sending registration confirmation emails, password reset links, property inquiry notifications, and alerts for new listings matching saved search criteria.
- **Data Flow:** The system will send a JSON payload to the email service containing the email recipient, subject, body text, and any relevant dynamic content (e.g., user's name, property details).

3.3.4 Communications Interfaces

Web

Browser

Communication:

The platform will primarily rely on web-based communication between the client (user's browser) and the server (backend). Communication between the user and the system will follow the **HTTP** (Hypertext Transfer Protocol) standard. To ensure secure data transmission, **HTTPS** (HTTP Secure) will be used to encrypt all communication between the client and server, protecting data such as user credentials, property details, and video uploads from unauthorized access during transmission.

Email

Communication:

The system will use email services (such as **SendGrid**, **Mailgun**, or **Amazon SES**) to handle user notifications, including registration confirmations, password resets, and property inquiry notifications. Emails will be sent using **SMTP** (Simple Mail Transfer Protocol), and all messages will be sent securely over **TLS** (Transport Layer Security) to encrypt email data in transit, ensuring that sensitive information remains protected.

4 Other Non-functional Requirements

4.1 Performance Requirements

Response Time:

The platform must provide fast response times to maintain a smooth user experience. For instance:

- **Search Performance:** Property searches should return results within **3 seconds** for a list of properties. This is essential to keep users engaged, especially when dealing with large datasets (e.g., hundreds of property listings).
- **Page Load Time:** Each page (e.g., property details, user profile) should load within **5 seconds** under normal network conditions.

Transaction Processing:

- **Property Listing Submission:** When a seller submits a property listing, the system should process and store the data within **10 seconds** to ensure that the listing appears promptly for

potential buyers/renters.

- **Video Uploads:** Video uploads by sellers should be handled efficiently. The time for processing and storing a video should not exceed **30 seconds** for average video sizes (up to 100 MB).

Simultaneous Users:

The system must be able to handle a significant number of simultaneous users without performance degradation:

- The platform should support **1000 concurrent users** without significant degradation in response time.
- **Database Queries:** Queries, such as fetching property listings, should be optimized for performance and should complete in **less than 2 seconds** under heavy load.

Scalability:

The system should be designed to handle an increasing number of users and listings without major changes to the architecture. As the platform grows, it should be capable of scaling horizontally by adding more server resources as needed.

4.2 Safety and Security Requirements

Data Encryption:

- All sensitive data, including **user passwords, personal details, and property information**, must be encrypted both during transit (using **SSL/TLS**) and at rest (using **AES-256** encryption for database storage).

User Authentication and Authorization:

- **User Login:** The platform will support **two-factor authentication (2FA)** for both admins and users to add an extra layer of security.
- **Password Storage:** Passwords must be stored securely using a strong hashing algorithm like **bcrypt** or **PBKDF2** to ensure that passwords are not compromised even if the database is breached.

Role-based Access Control (RBAC):

- The platform must restrict user access based on their roles (Admin, Seller, Buyer/Renter). Admins should have access to all system features, while sellers and buyers/renters should only have access to their respective features (e.g., sellers can edit their property listings, but cannot manage user accounts).

Data Privacy and Compliance:

- The platform must comply with global privacy regulations, such as the **General Data Protection Regulation (GDPR)** and **California Consumer Privacy Act (CCPA)**, ensuring that personal data is handled, stored, and processed with consent, and users can request data deletion if desired.

Session Management:

- Sessions should time out after **15 minutes of inactivity** to prevent unauthorized access.
- Secure session cookies should be used, and session hijacking must be prevented by implementing **secure**, **HttpOnly**, and **SameSite** cookie flags.

4.3 Software Quality Attributes

4.3.1 Reliability

- **System Uptime:**
The platform must ensure **99.9% uptime**, meaning that the system is operational and available for use almost all the time. This ensures a reliable experience for users, particularly in high-traffic situations.
- **Error Handling:**
The system should have robust error handling mechanisms. Any system errors should be logged and reported in real-time to administrators to enable fast resolution.

4.3.2 Usability

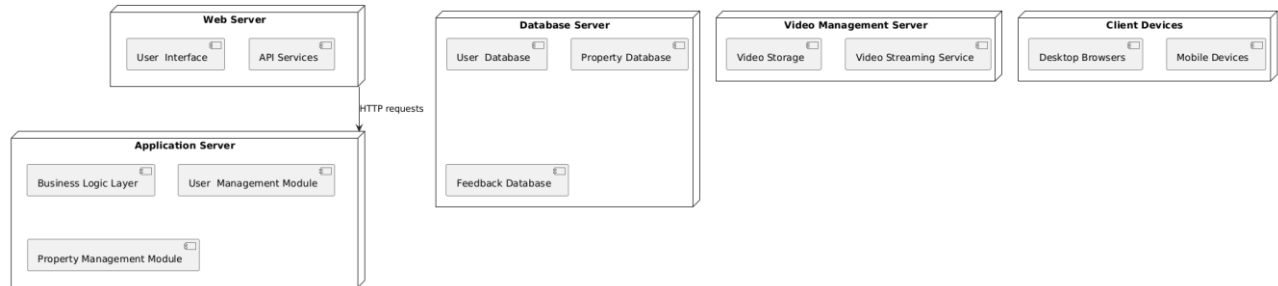
- **User Interface Design:**
The platform must have an intuitive and easy-to-navigate user interface that allows users to quickly learn how to use it. The design should prioritize ease of use over complexity, with clearly labeled buttons and consistent design across screens.
- **Mobile Usability:**
The platform must be fully responsive, providing a seamless user experience on both desktop and mobile devices, ensuring users can perform tasks easily on any device.

4.3.3 Maintainability

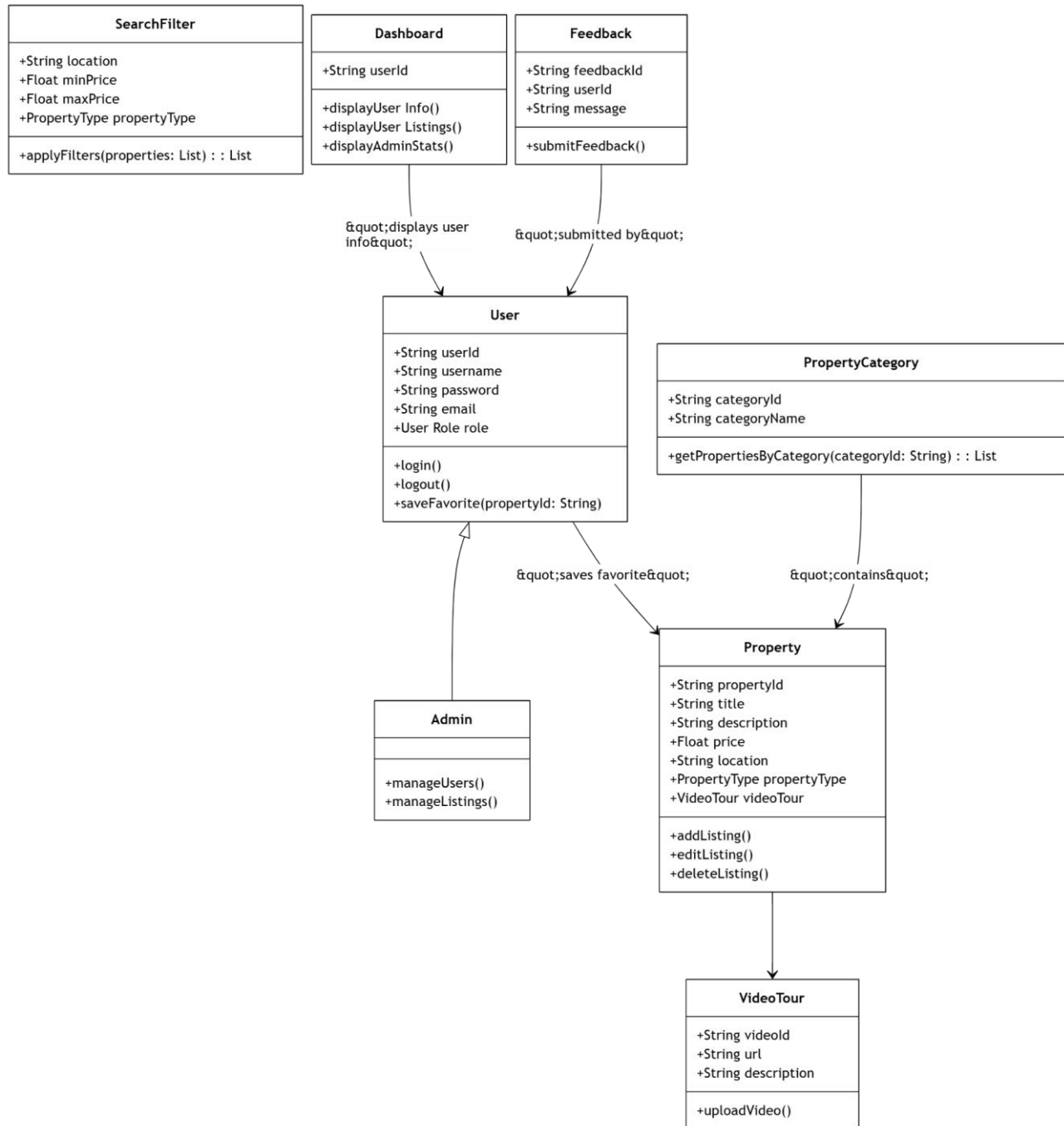
- **Modular Codebase:**
The system must be built with a modular architecture, enabling easy updates and the addition of new features without disrupting the overall platform. This includes clear separation of concerns between the frontend, backend, and database layers.
- **Code Documentation:**
Proper documentation of the codebase must be maintained to ensure that developers can easily understand the logic and make modifications or enhancements when needed.

5 Design Description

5.1 Composite Viewpoint

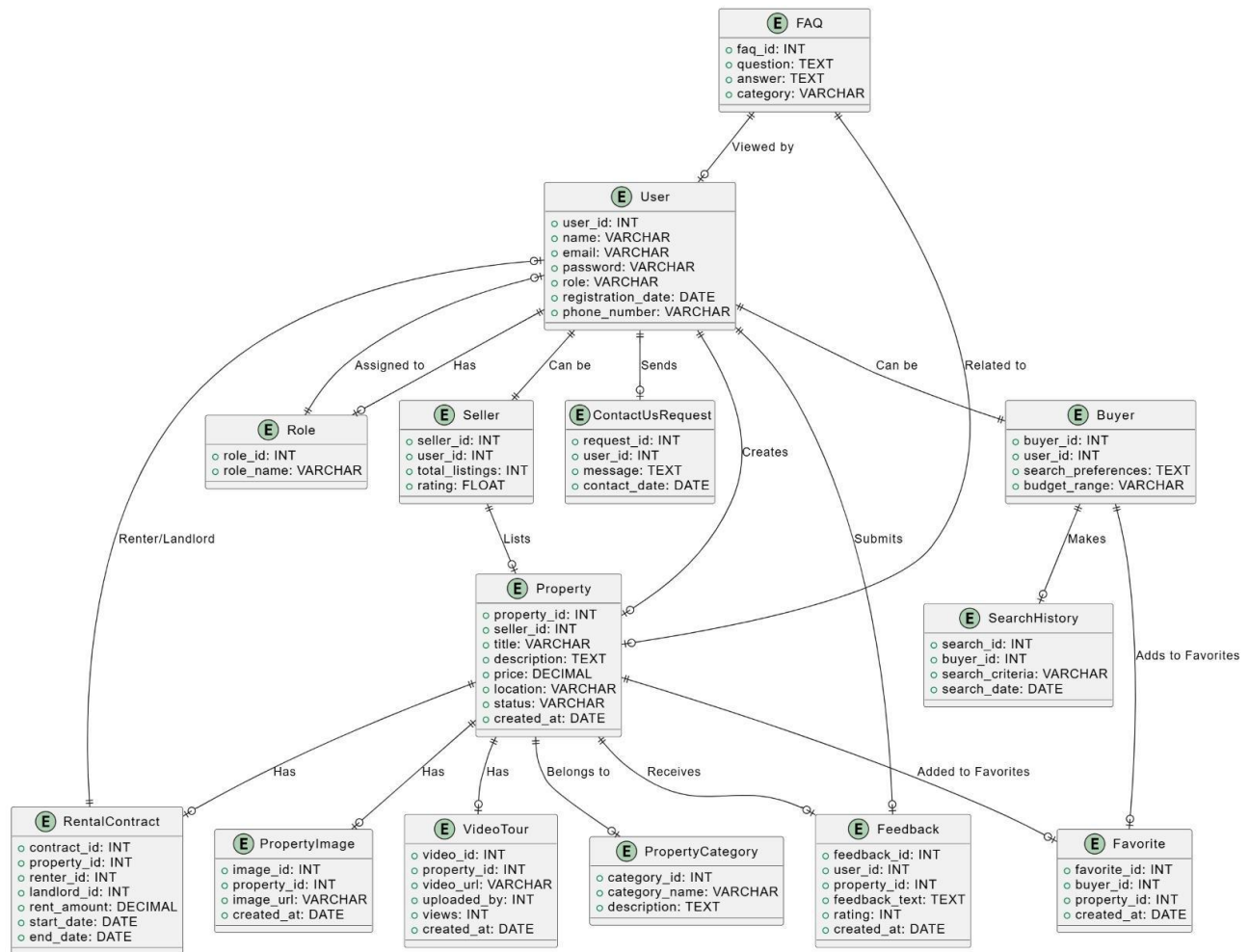


5.2 Logical Viewpoint



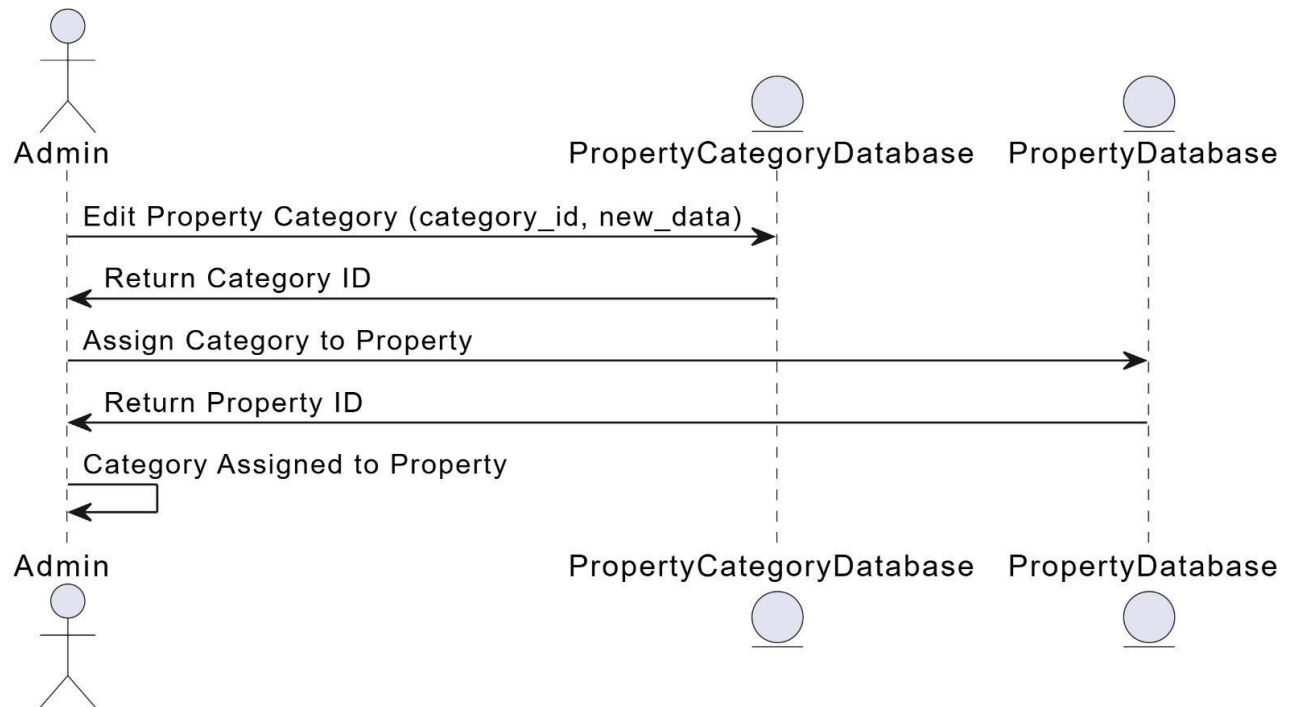
5.3 Information Viewpoint

Pak Property - Complete Entity Relationship Diagram

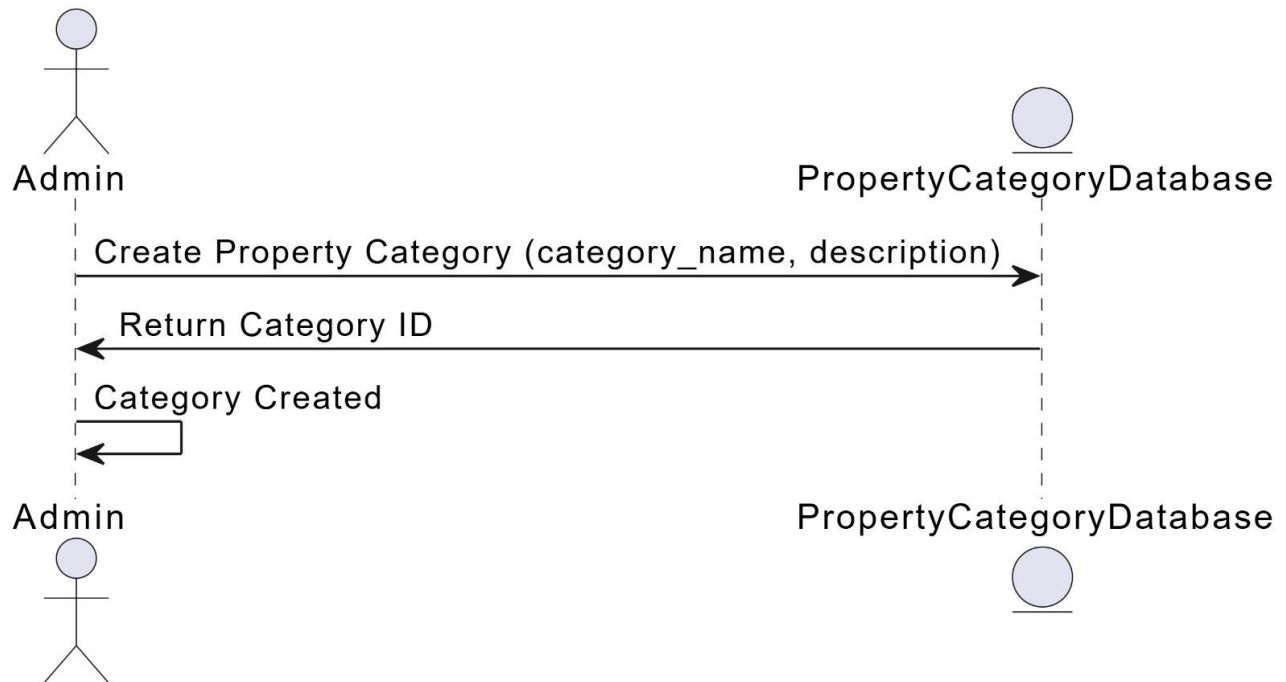


5.4 Interaction Viewpoint

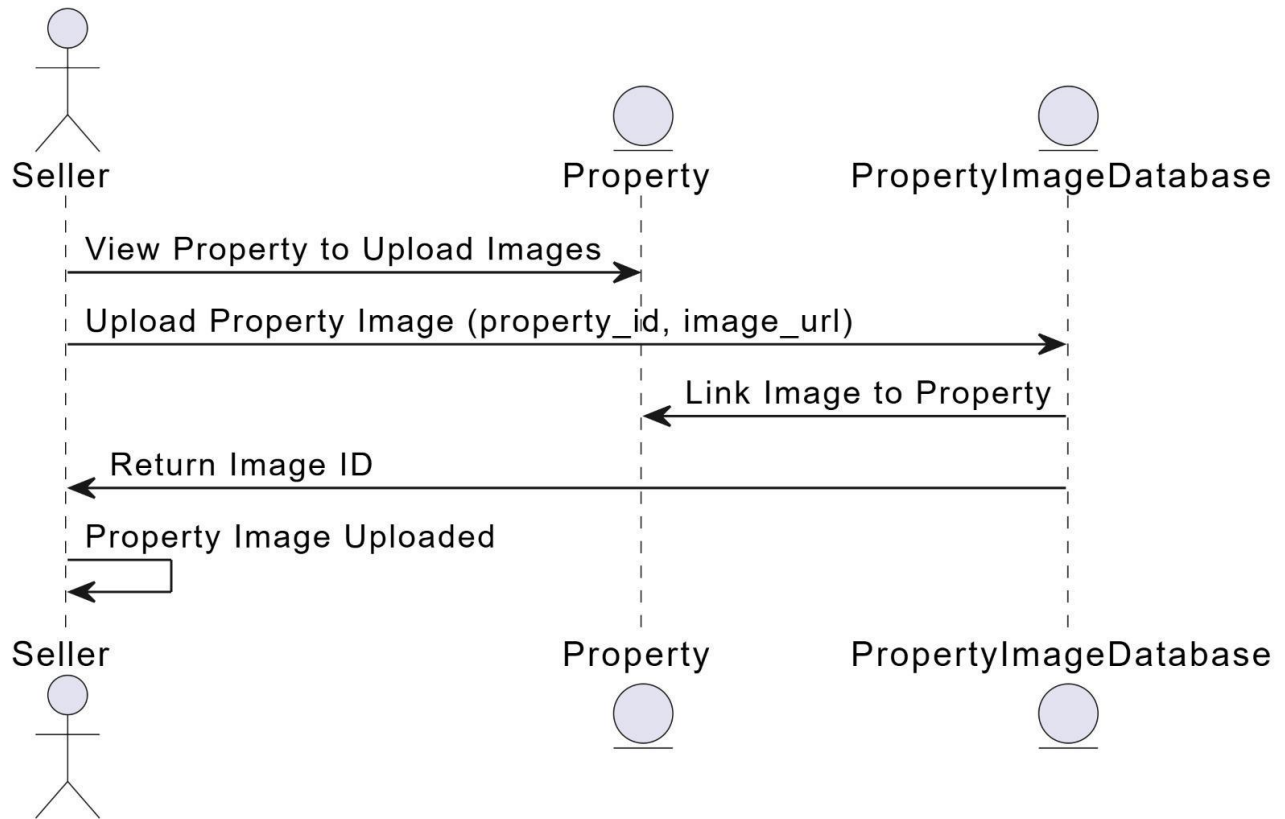
Property Category Management



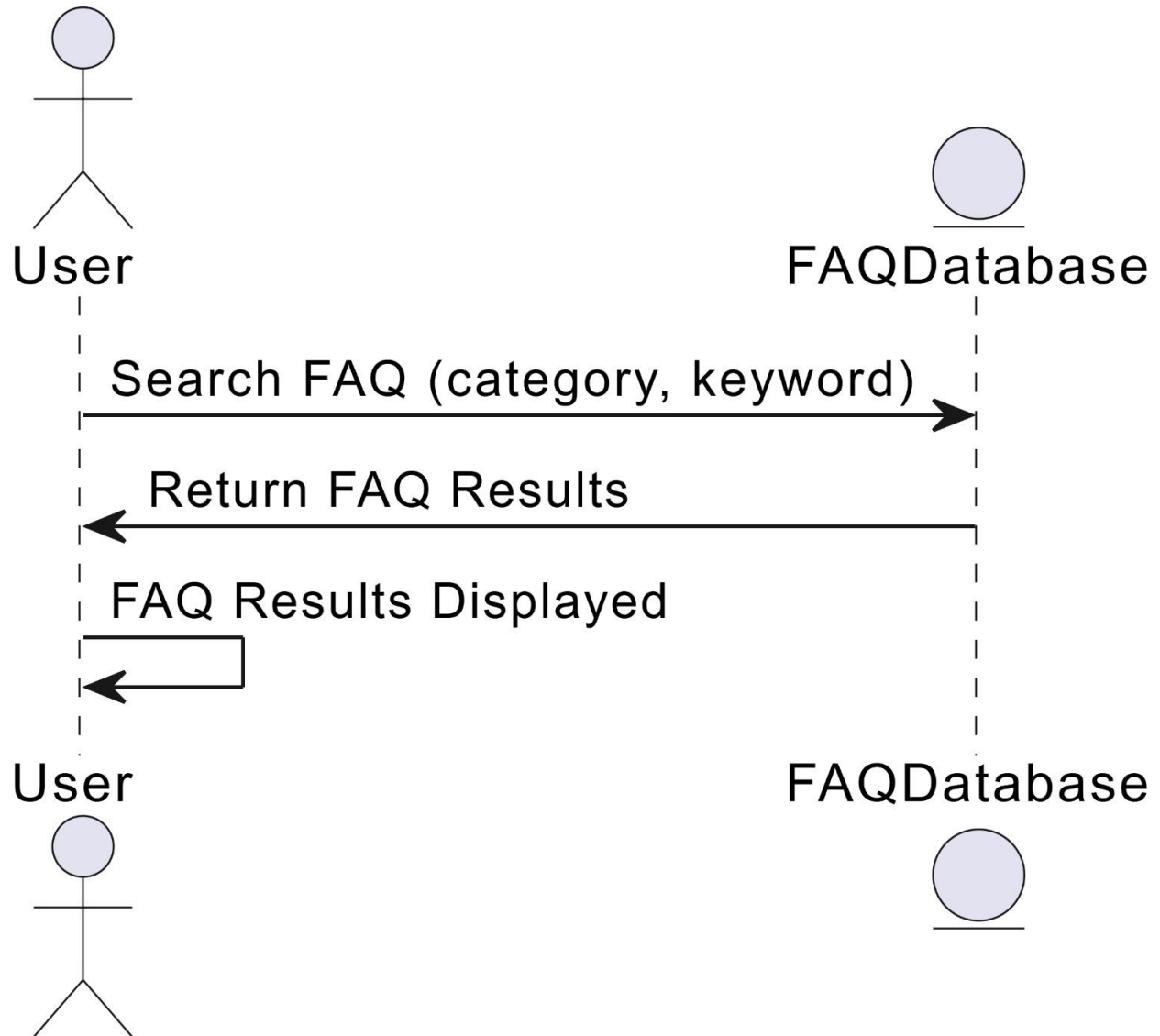
Creating Property Category

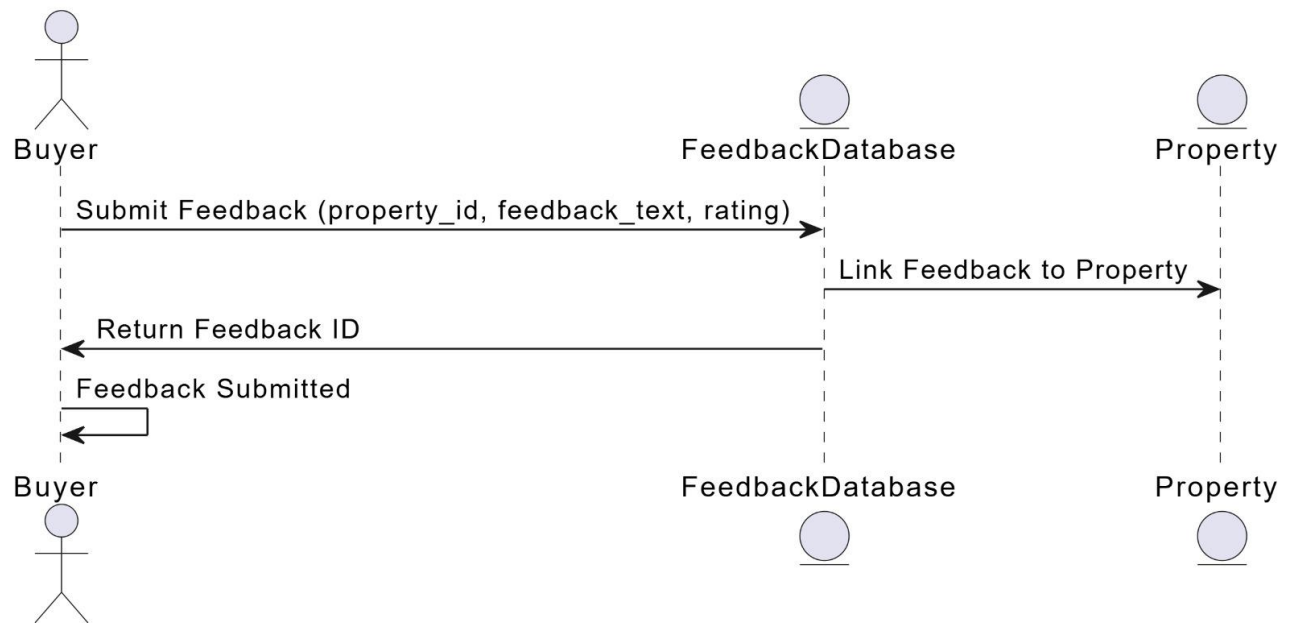


Managing Property Images

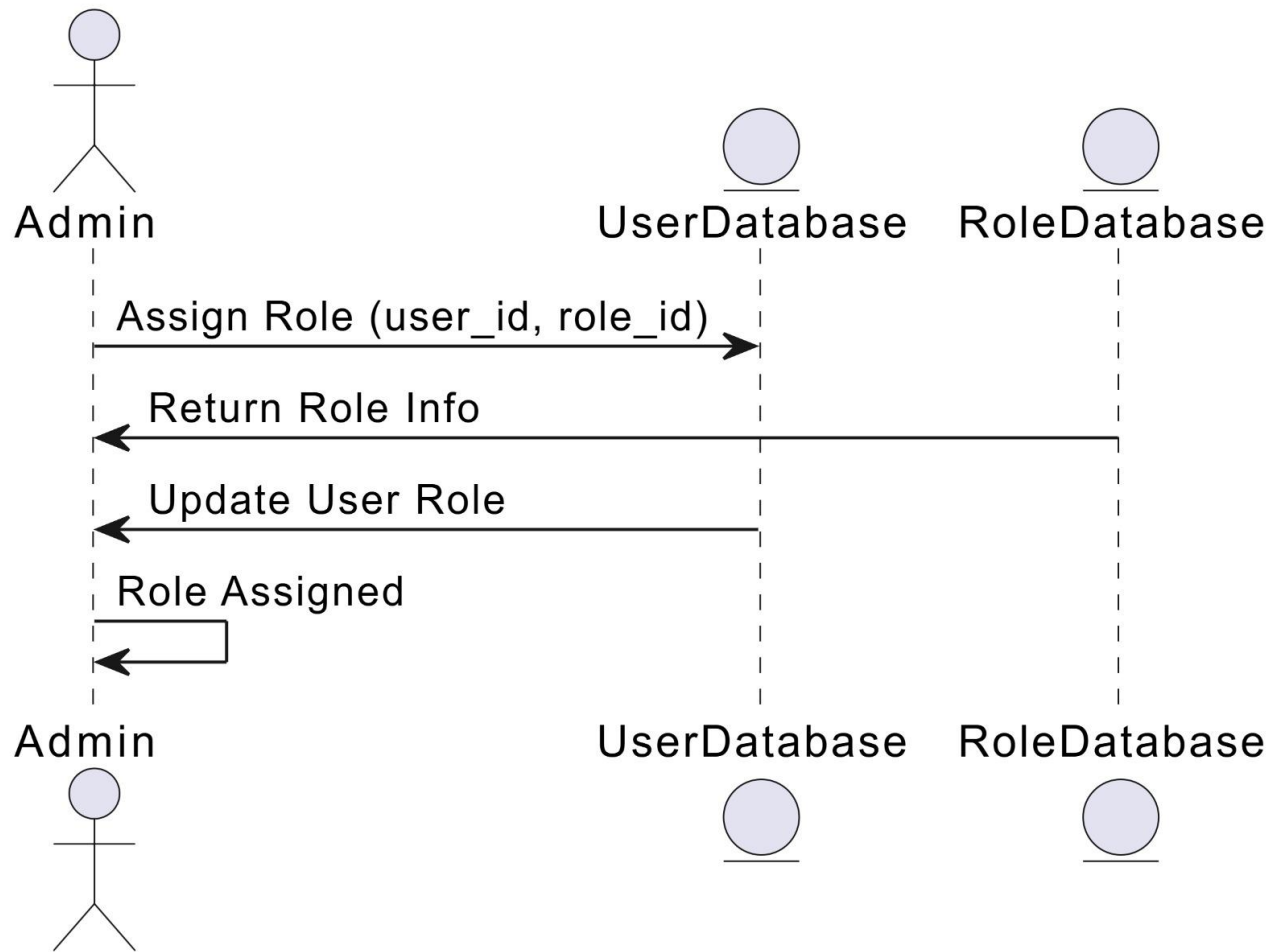


FAQ Search

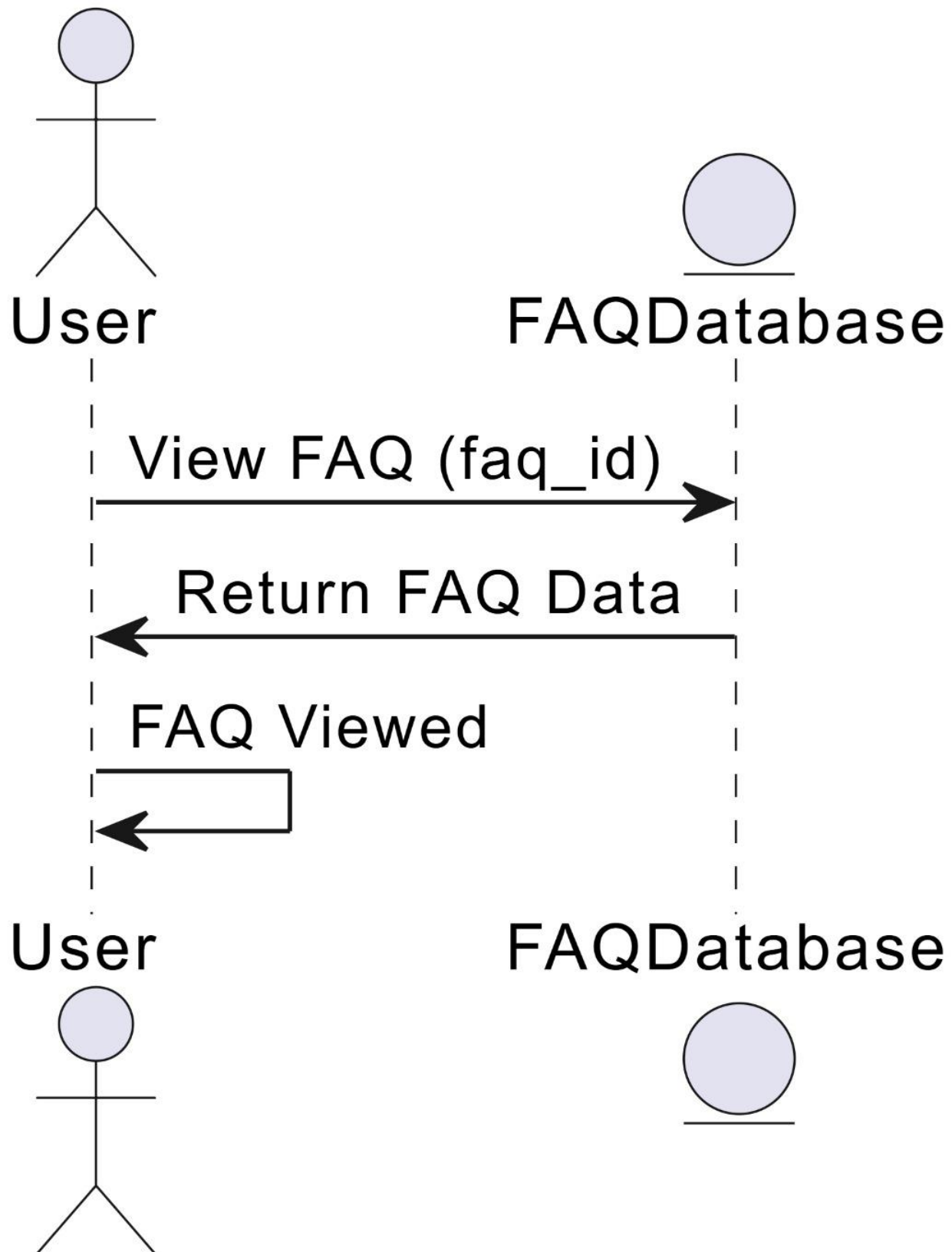


Property Feedback

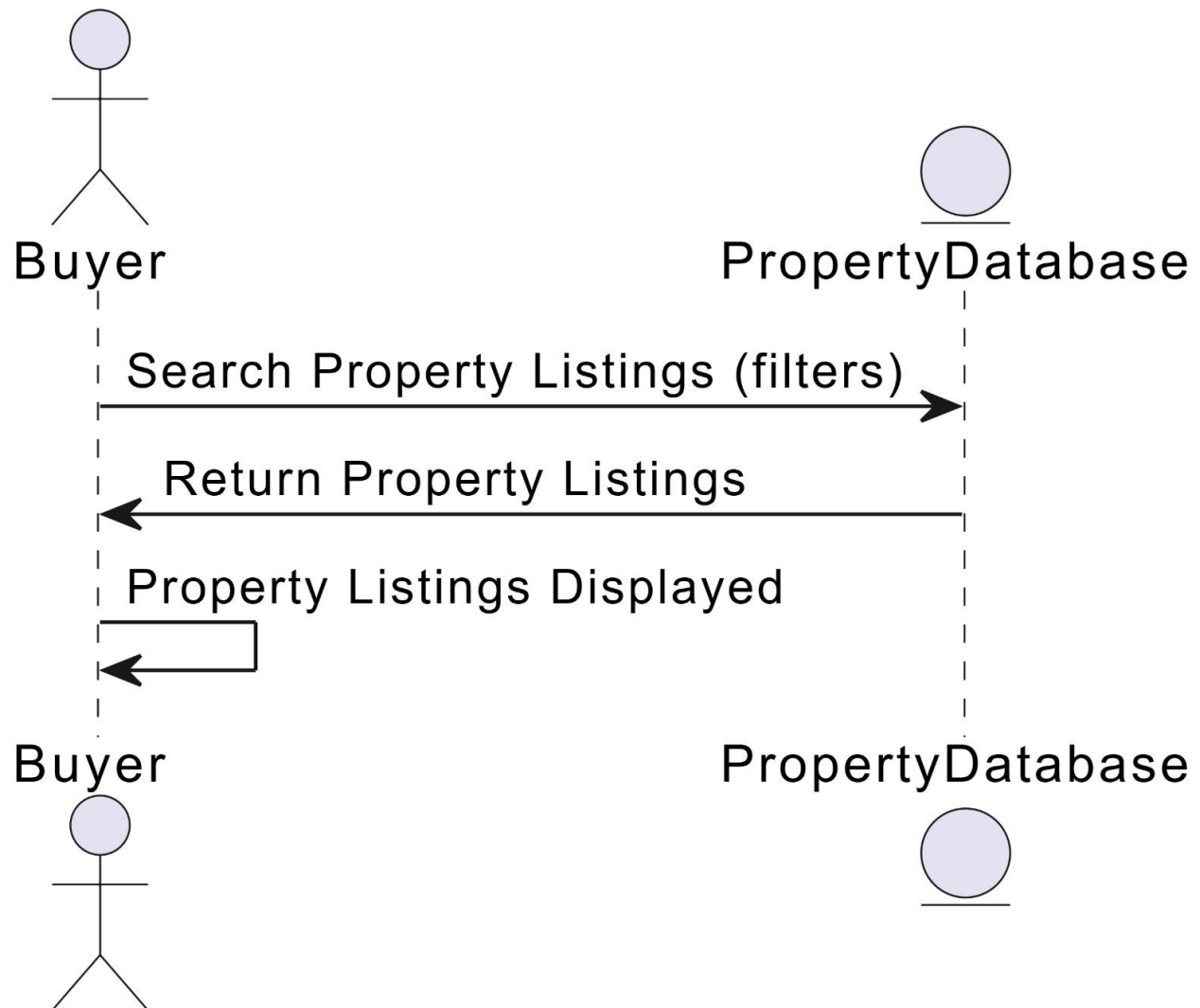
Role Assignment



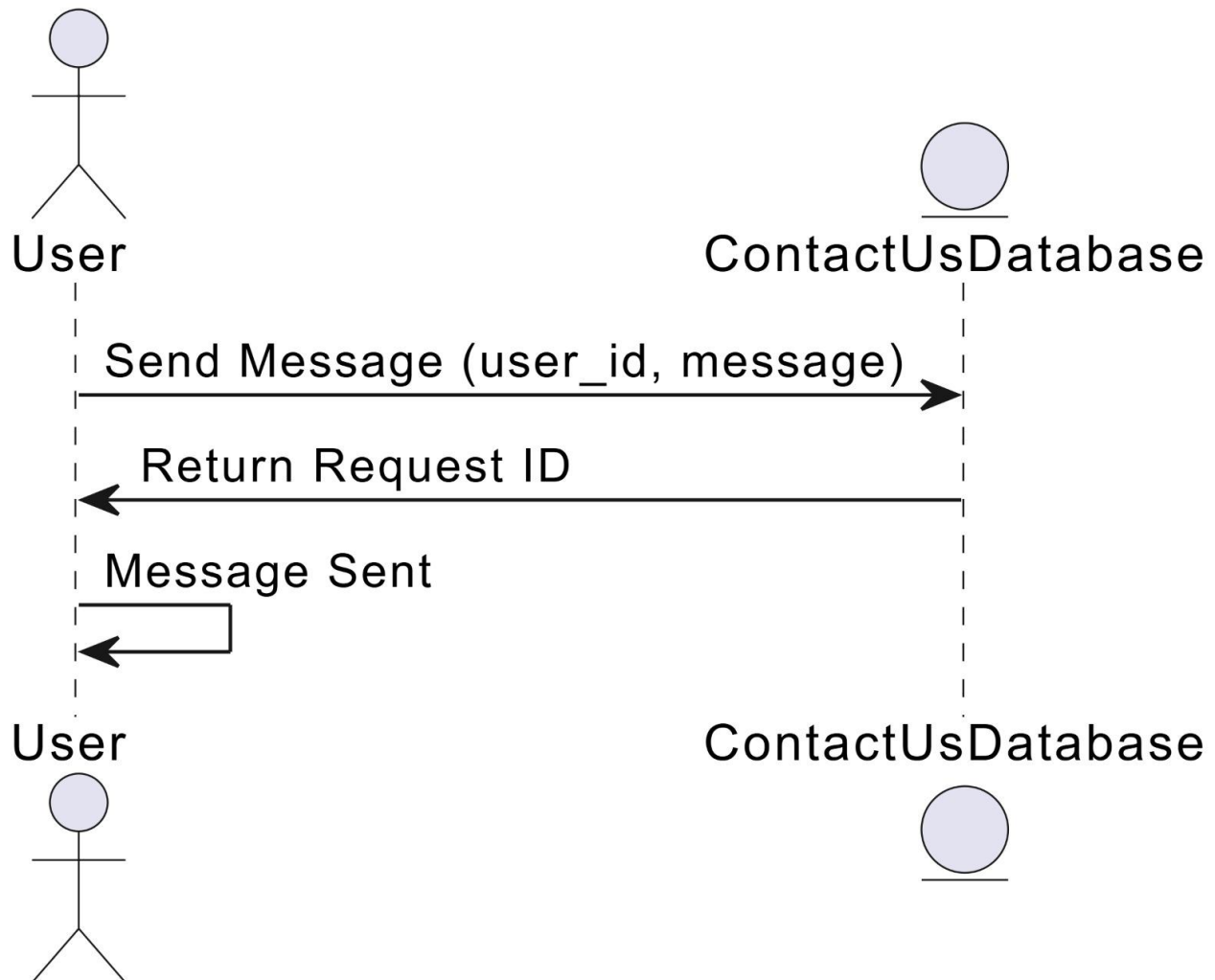
Viewing FAQs



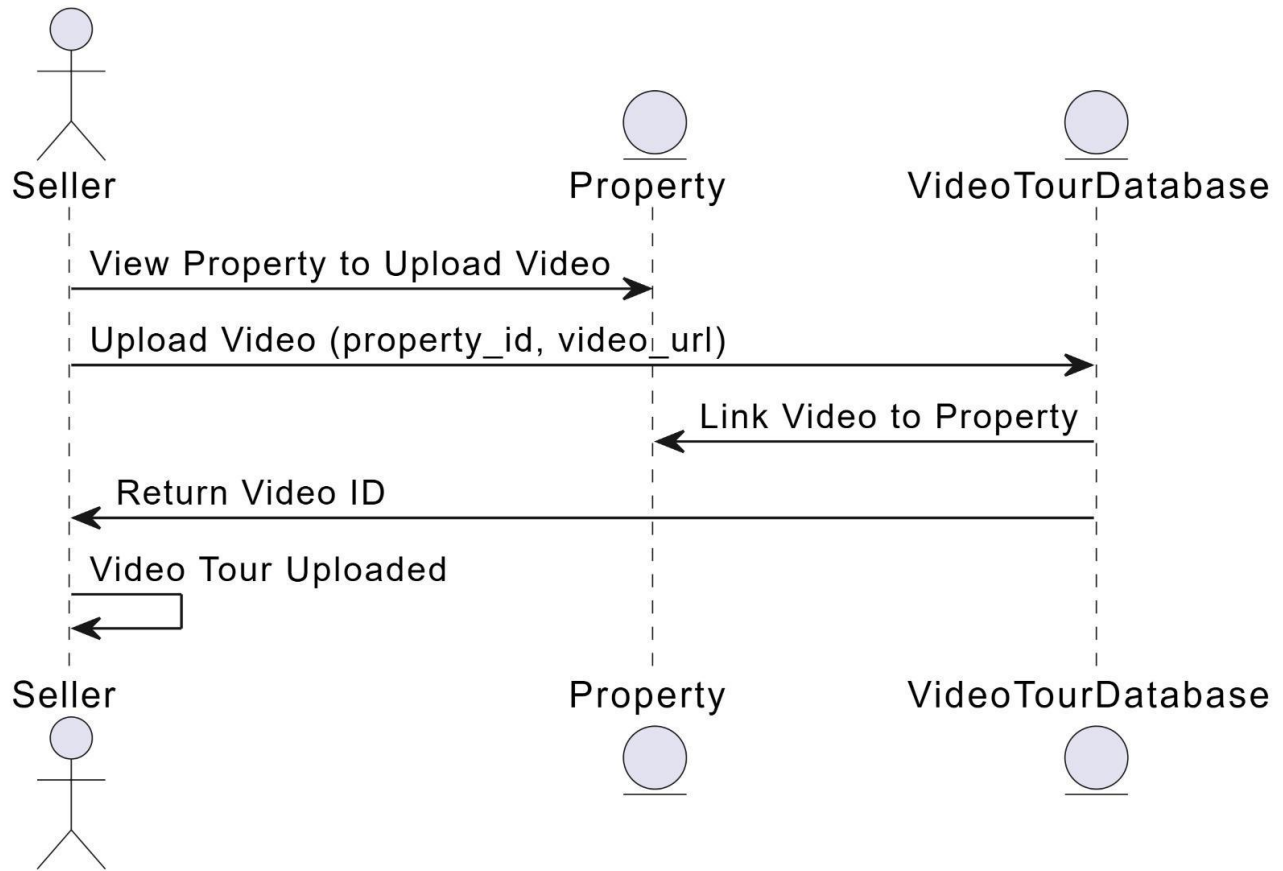
Viewing Property Listings



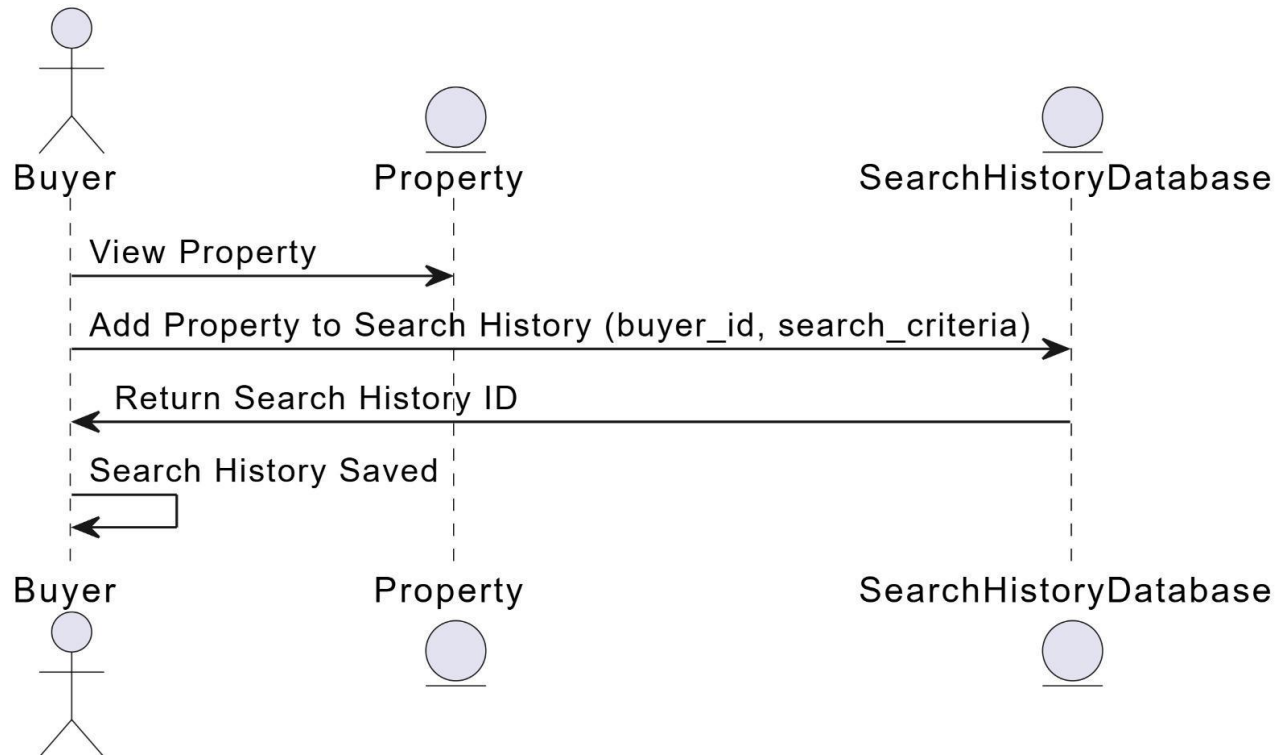
Sending Contact Us Request



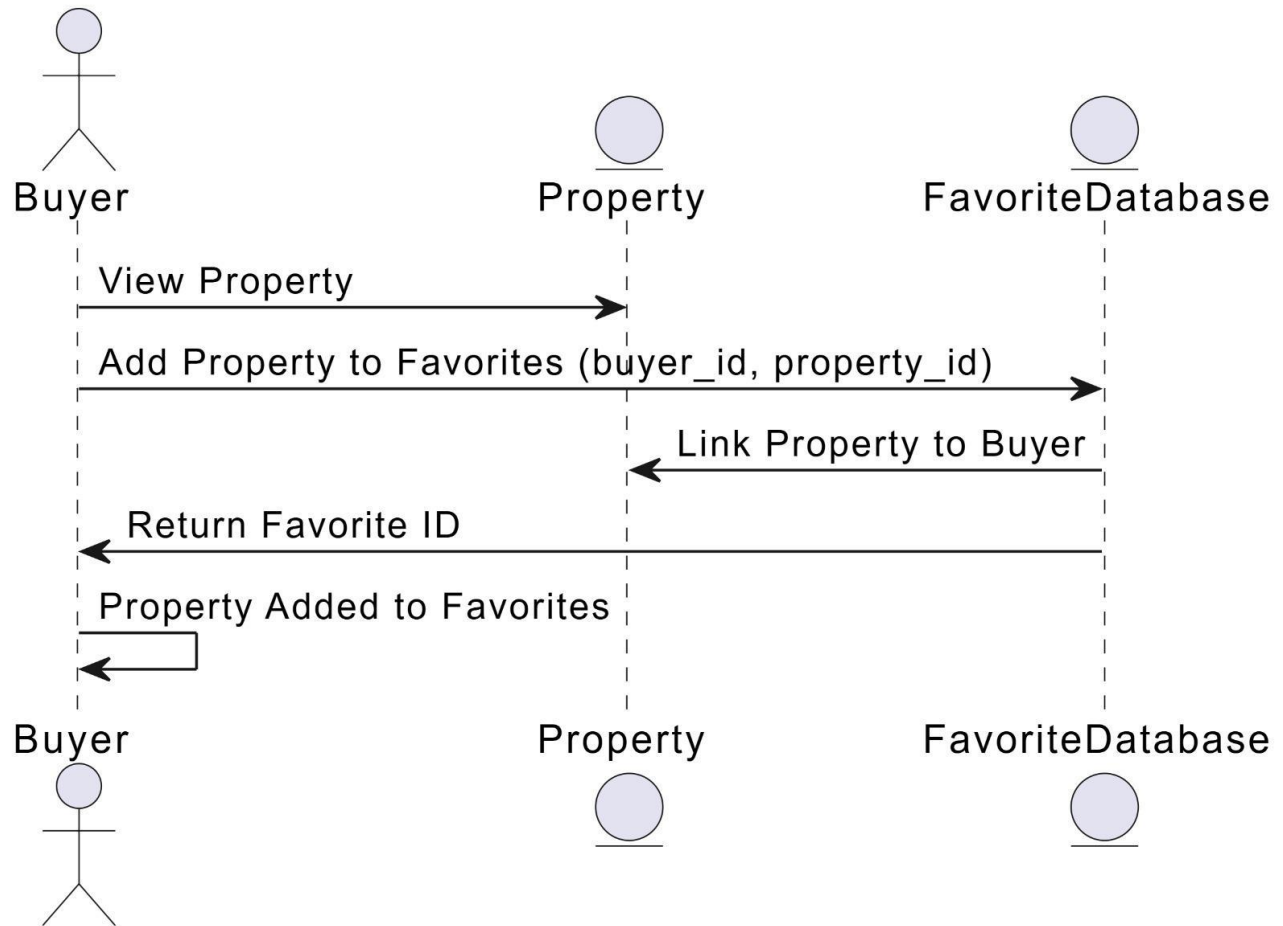
Property Video Tour Upload



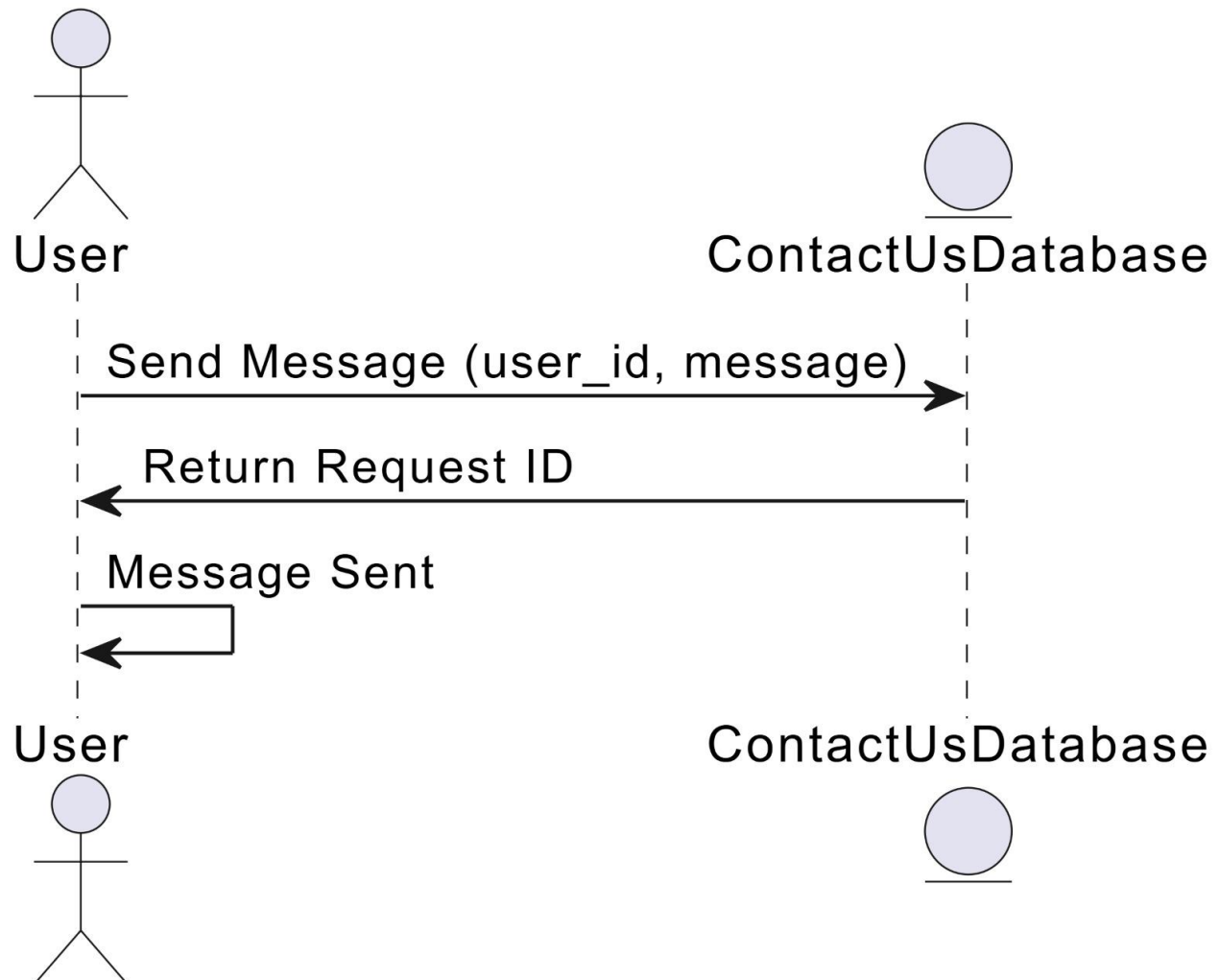
Adding Property to Search History



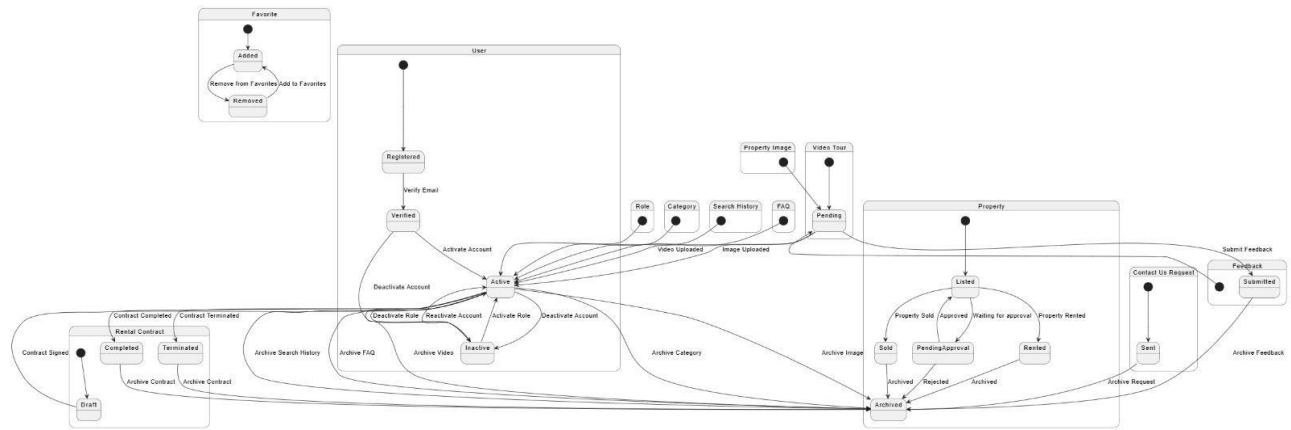
Adding Property to Favorites



Sending Contact Us Request



5.5 State Dynamics Viewpoint



Entity	Current State	Event	Next State
User	Registered	Verify Email	Verified
User	Verified	Activate Account	Active
User	Verified	Deactivate Account	Inactive
User	Inactive	Reactivate Account	Active
User	Active	Deactivate Account	Inactive
Property	Listed	Property Sold	Sold
Property	Listed	Property Rented	Rented
Property	Listed	Pending Approval	PendingApproval
Property	PendingApproval	Property Approved	Listed
Property	PendingApproval	Property Rejected	Archived
Property	Sold	Archive Property	Archived
Property	Rented	Archive Property	Archived
Rental Contract	Draft	Contract Signed	Active
Rental Contract	Active	Contract Completed	Completed
Rental Contract	Active	Contract Terminated	Terminated
Rental Contract	Completed	Archive Contract	Archived
Rental Contract	Terminated	Archive Contract	Archived
Video Tour	Pending	Video Uploaded	Active
Video Tour	Active	Archive Video	Archived
Feedback	Pending	Submit Feedback	Submitted
Feedback	Submitted	Archive Feedback	Archived
Favorite	Added	Remove from Favorites	Removed
Favorite	Removed	Add to Favorites	Added
Property Image	Pending	Image Uploaded	Active
Property Image	Active	Archive Image	Archived
Role	Active	Deactivate Role	Inactive
Role	Inactive	Activate Role	Active
Category	Active	Archive Category	Archived
Search History	Active	Archive Search History	Archived
Contact Us Request	Sent	Archive Request	Archived
FAQ	Active	Archive FAQ	Archived

5.6 Algorithm Viewpoint

BEGIN PAK_Properties

// GLOBAL VARIABLES

DEFINE users, properties, favorites, videos, filters, roles

// AUTHENTICATION MODULE

FUNCTION Login(username, password):

 IF credentials_valid(username, password):

 RETURN UserSession(user_role)

 ELSE:

 DISPLAY "Invalid credentials"

// ROLE MANAGEMENT MODULE

FUNCTION AssignRole(user, role):

 IF current_user.role == 'Admin':

 user.role = role

 RETURN "Role assigned successfully"

 ELSE:

 RETURN "Unauthorized access"

// PROPERTY LISTINGS MANAGEMENT MODULE

FUNCTION AddProperty(user, property_details):

 IF user.role == 'Seller' OR user.role == 'Admin':

 ADD property_details TO properties

 RETURN "Property added successfully"

ELSE:

 RETURN "Unauthorized access"

FUNCTION EditProperty(user, property_id, new_details):

 IF user.role == 'Seller' AND property_id.owner == user OR user.role == 'Admin':

 UPDATE properties[property_id] WITH new_details

 RETURN "Property updated successfully"

 ELSE:

 RETURN "Unauthorized access"

FUNCTION DeleteProperty(user, property_id):

 IF user.role == 'Seller' AND property_id.owner == user OR user.role == 'Admin':

 REMOVE property_id FROM properties

 RETURN "Property deleted successfully"

 ELSE:

 RETURN "Unauthorized access"

// VIDEO MANAGEMENT MODULE

FUNCTION UploadVideo(user, property_id, video):

 IF user.role == 'Seller' AND property_id.owner == user:

 ADD video TO property_id.media

 RETURN "Video uploaded successfully"

 ELSE:

 RETURN "Unauthorized access"

// SEARCH AND FILTER MODULE

FUNCTION SearchProperties(filters):

 results = FILTER properties BASED ON filters

 RETURN results

// FAVORITES MANAGEMENT MODULE

FUNCTION AddToFavorites(user, property_id):

 IF user.role == 'Buyer':

 ADD property_id TO favorites[user]

 RETURN "Property added to favorites"

 ELSE:

 RETURN "Unauthorized access"

FUNCTION ViewFavorites(user):

 RETURN favorites[user]

// RESPONSIVE DESIGN HANDLING

FUNCTION RenderResponsiveView(device_type):

 IF device_type == 'mobile':

 RENDER MobileLayout()

 ELSE:

 RENDER DesktopLayout()

// DASHBOARD MODULES

FUNCTION LoadUserDashboard(user):

 IF user.role == 'Seller':

 DISPLAY "Seller Dashboard"

```
        DISPLAY user.properties, user.account_settings

    ELSE IF user.role == 'Buyer':

        DISPLAY "Buyer Dashboard"

        DISPLAY user.favorites, search_options

    ELSE:

        DISPLAY "Unauthorized access"


FUNCTION LoadAdminDashboard(admin):

    IF admin.role == 'Admin':

        DISPLAY "Admin Dashboard"

        DISPLAY statistics, user_management, property_overview

    ELSE:

        DISPLAY "Unauthorized access"


// MAIN PROGRAM

WHILE platform_running:

    DISPLAY LoginScreen()

    user_session = Login(user_input.username, user_input.password)

    IF user_session:

        SWITCH user_session.role:

            CASE 'Admin':

                LoadAdminDashboard(user_session.user)

            CASE 'Seller':

                LoadUserDashboard(user_session.user)

            CASE 'Buyer':
```

LoadUserDashboard(user_session.user)

DEFAULT:

DISPLAY "Invalid role"

ELSE:

DISPLAY "Login failed"

END PAK_Properties