UE22CS352B - Object Oriented Analysis & Design

Mini Project Report
Real estate management system

Submitted by:

PES1UG22CS581 Shreyas S Magadi

PES1UG22CS591 Siddhant P Gopi

PES1UG22CS579 Shreyas Gowda

6th semester J section

Facultly Name

Dr. Bhargavi Mokashi

January - May 2025

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100ft Ring Road, Bengaluru – 560 085, Karnataka, India

Problem Statement:
Real estate management system

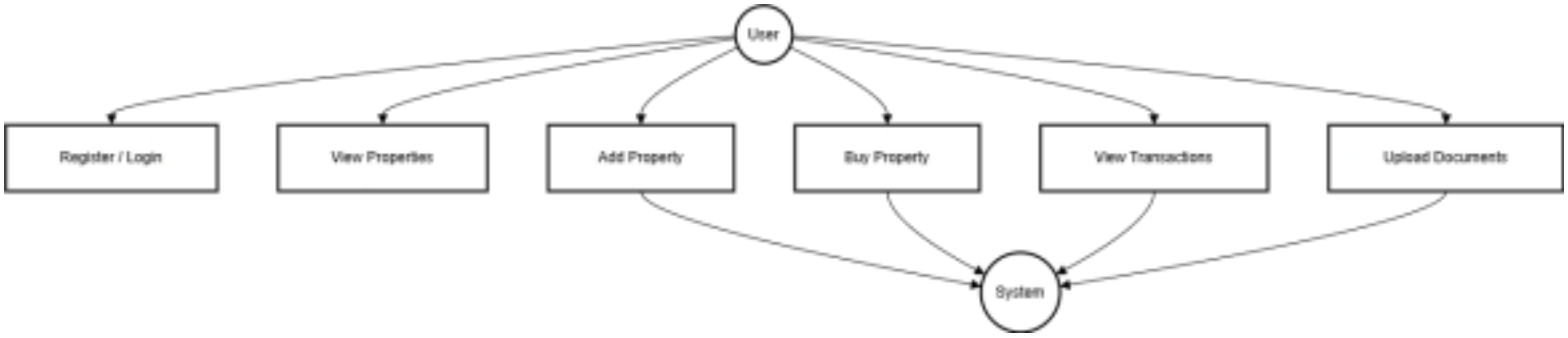
The objective of our project is to design and implement a Real Estate Management System that enables users to perform end-to-end property management using modern object-oriented design practices. Users should be able to register and authenticate securely, add and view properties with detailed metadata, perform transactions such as buying and selling, and track their financial performance. The solution must adhere to the principles of modularity, reusability, and maintainability using UML design models, an MVC-based architectural pattern, and appropriate design principles and patterns. The backend must be developed using Java with Spring Boot and persist data using a relational database. Only web and desktop applications are permitted as per project constraints.

Key Features:

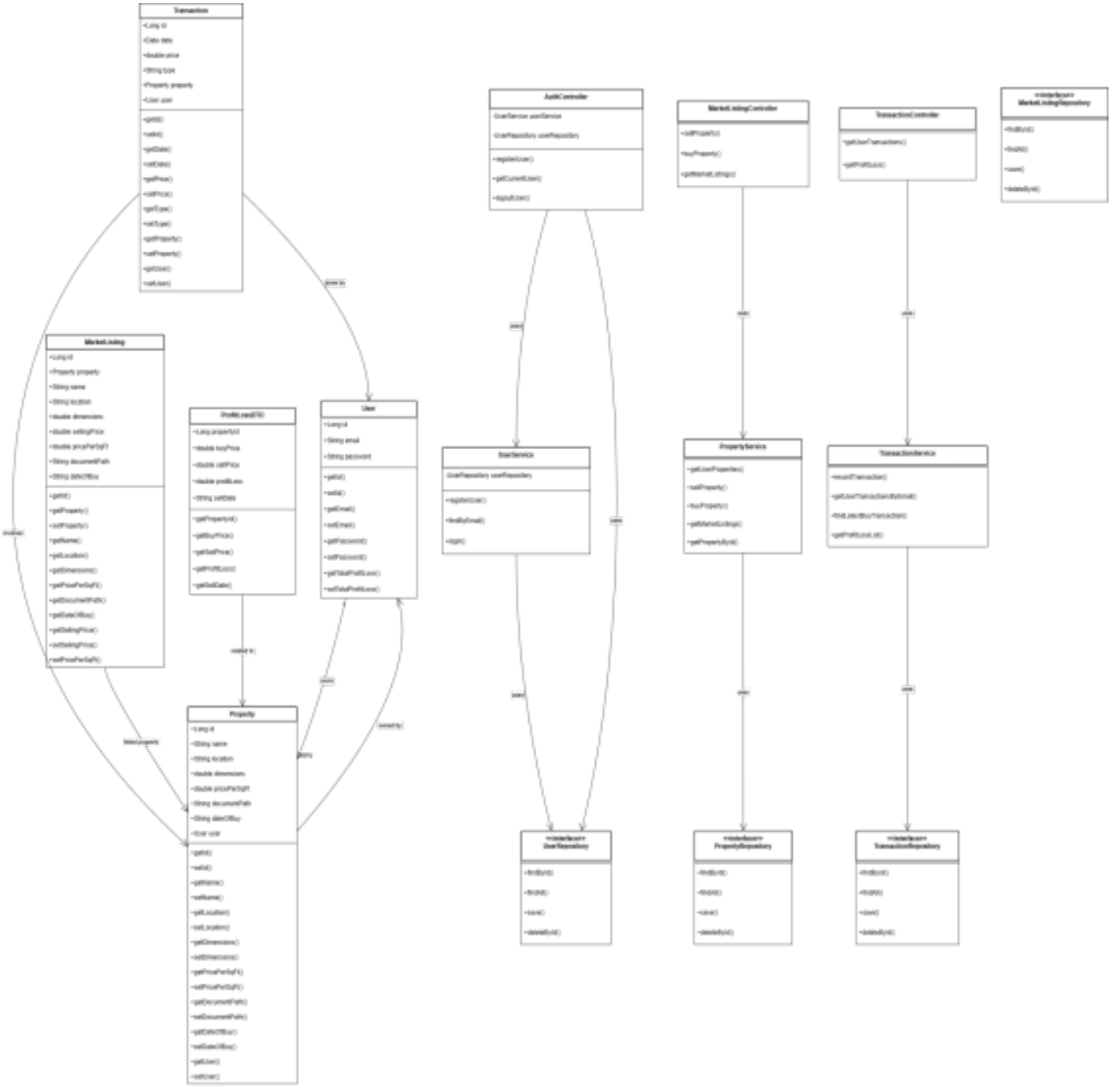
- User Registration and Authentication: A secure login system uses encrypted passwords.
- Add/View Properties: Users can add new properties with relevant details, including location, price, and associated PDF documents (e.g., sale deed).
- Property Listings: Properties listed for sale are shown on a dashboard for potential buyers.
- Buy/Sell Transactions: Users can buy or sell properties with appropriate balance checks and transaction records.
- Profit & Loss Tracking: Real-time calculation of profit/loss from each property transaction using DTOs.
- Spring MVC Architecture: Clearly separates concerns across controllers, services, models, and repositories.
- MySQL Integration: Database is used to persist user, property, and transaction data.
- PDF Upload Support: Relevant documents like sale deeds can be uploaded and linked to properties.

Models:

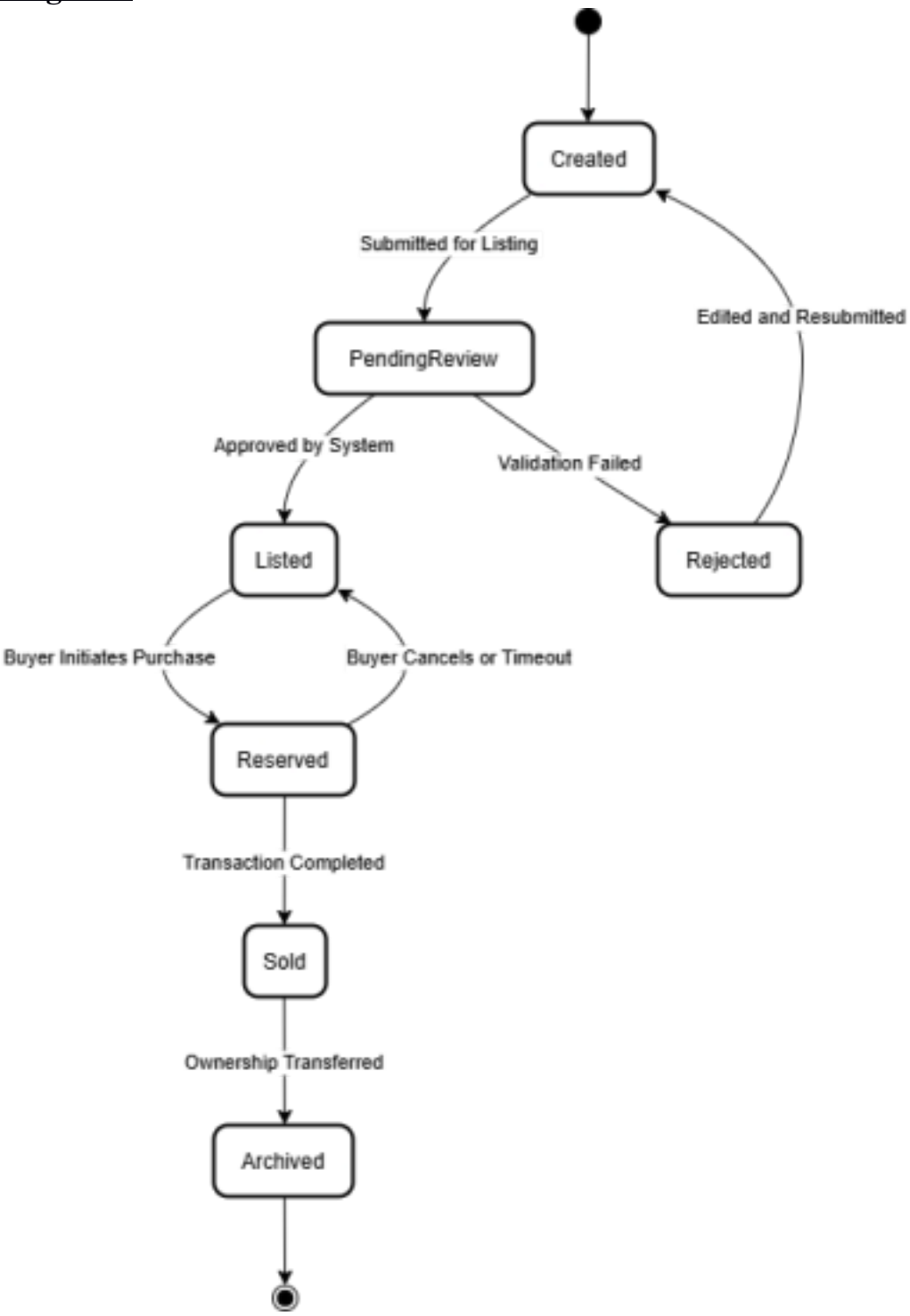
Use Case Diagram:



Class Diagram:

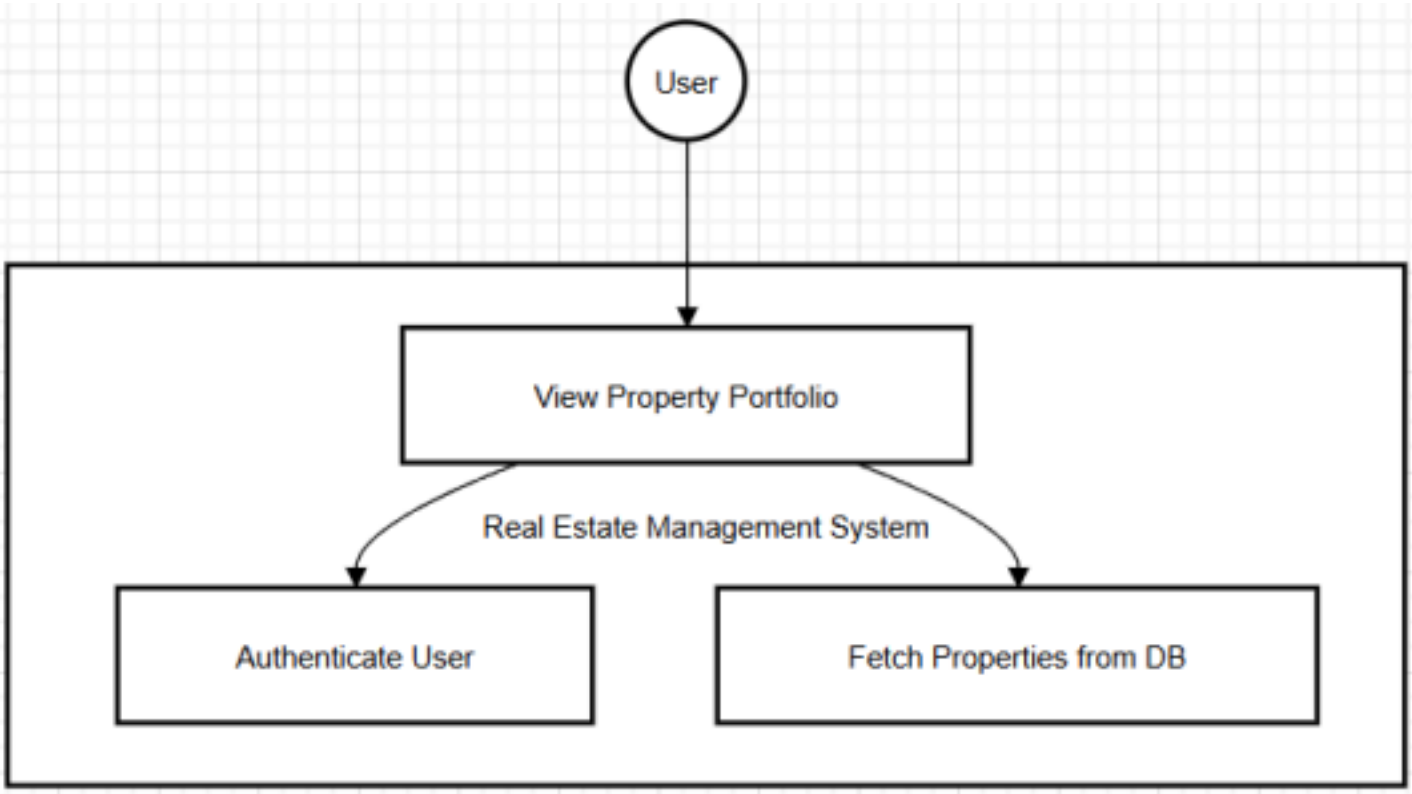


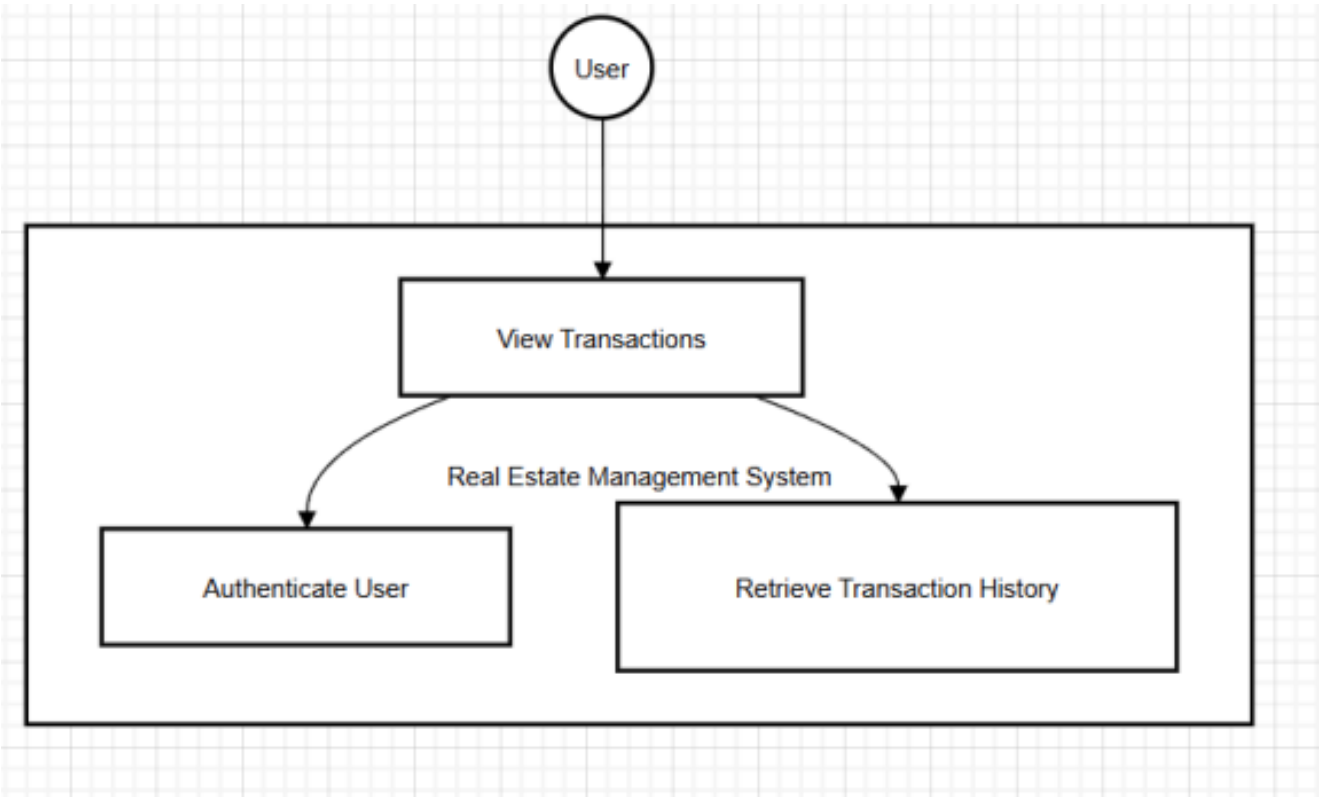
State Diagram:

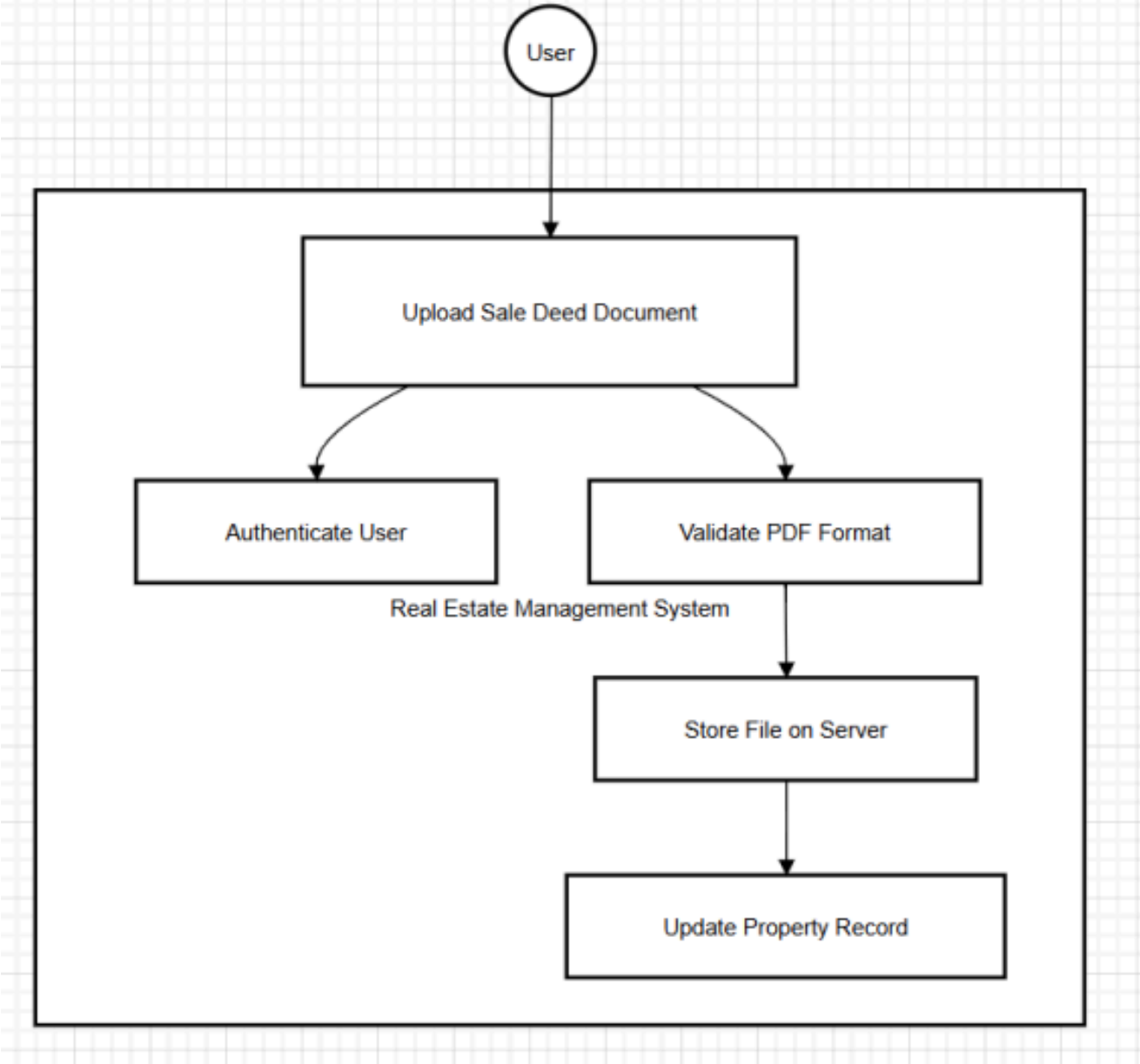


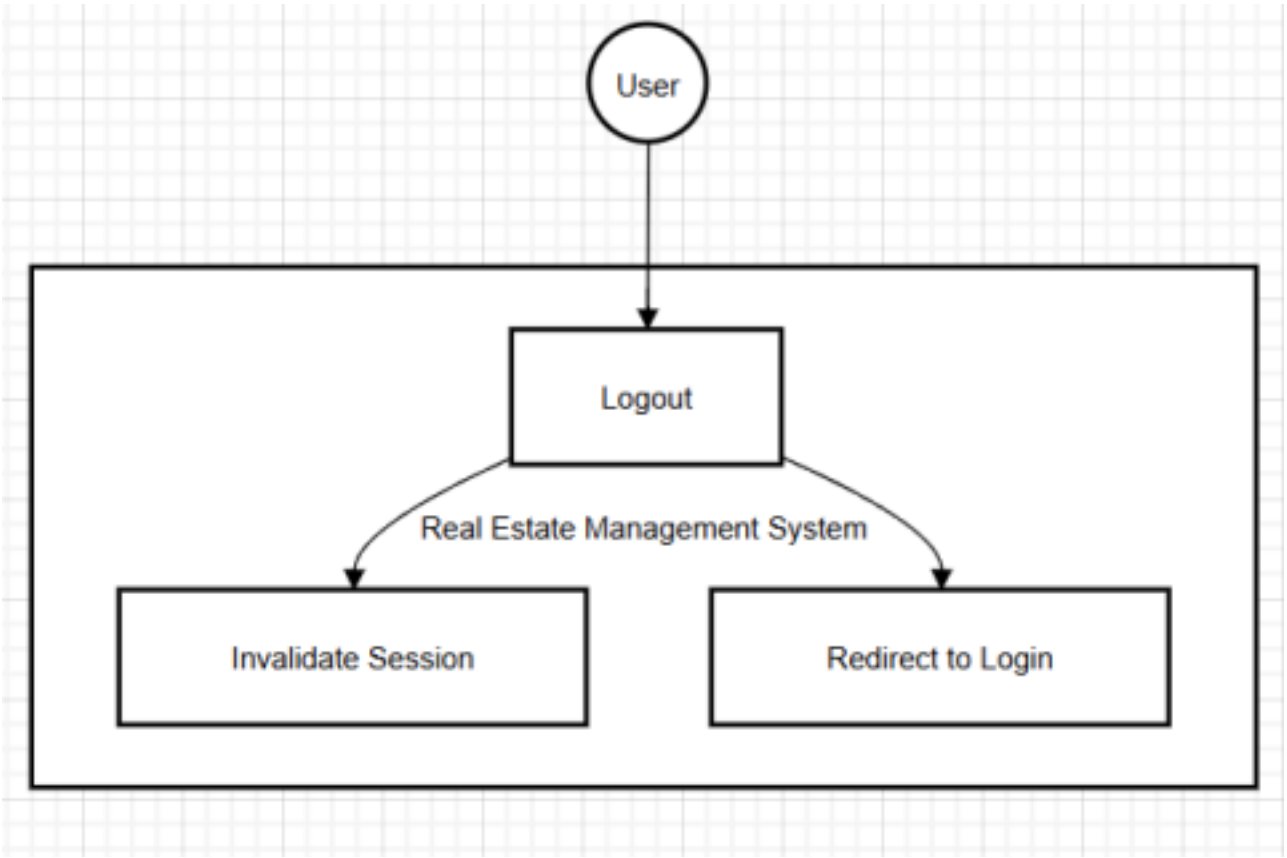
Activity Diagrams:

1. Major Usecase

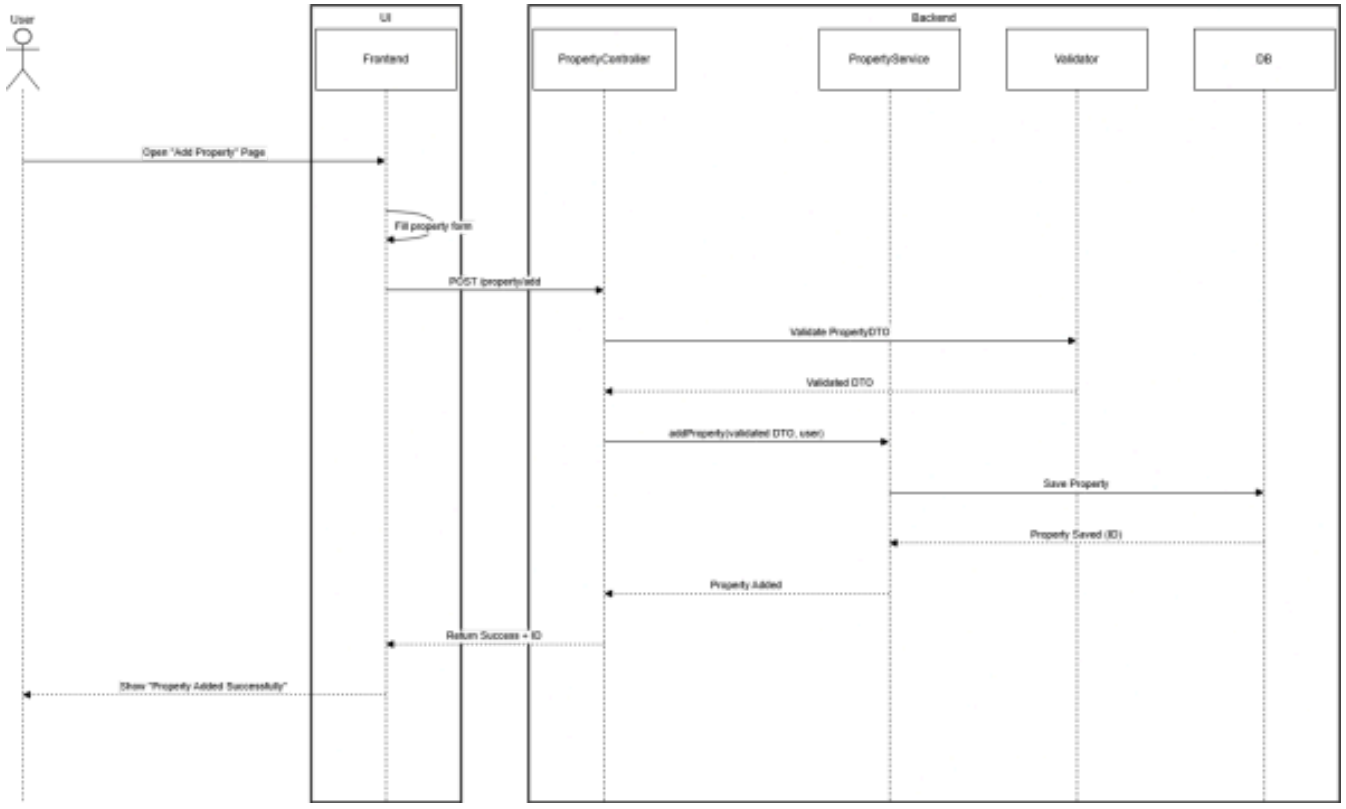

User
View Property Portfolio
Real Estate Management System
Authenticate User
Fetch Properties from DB

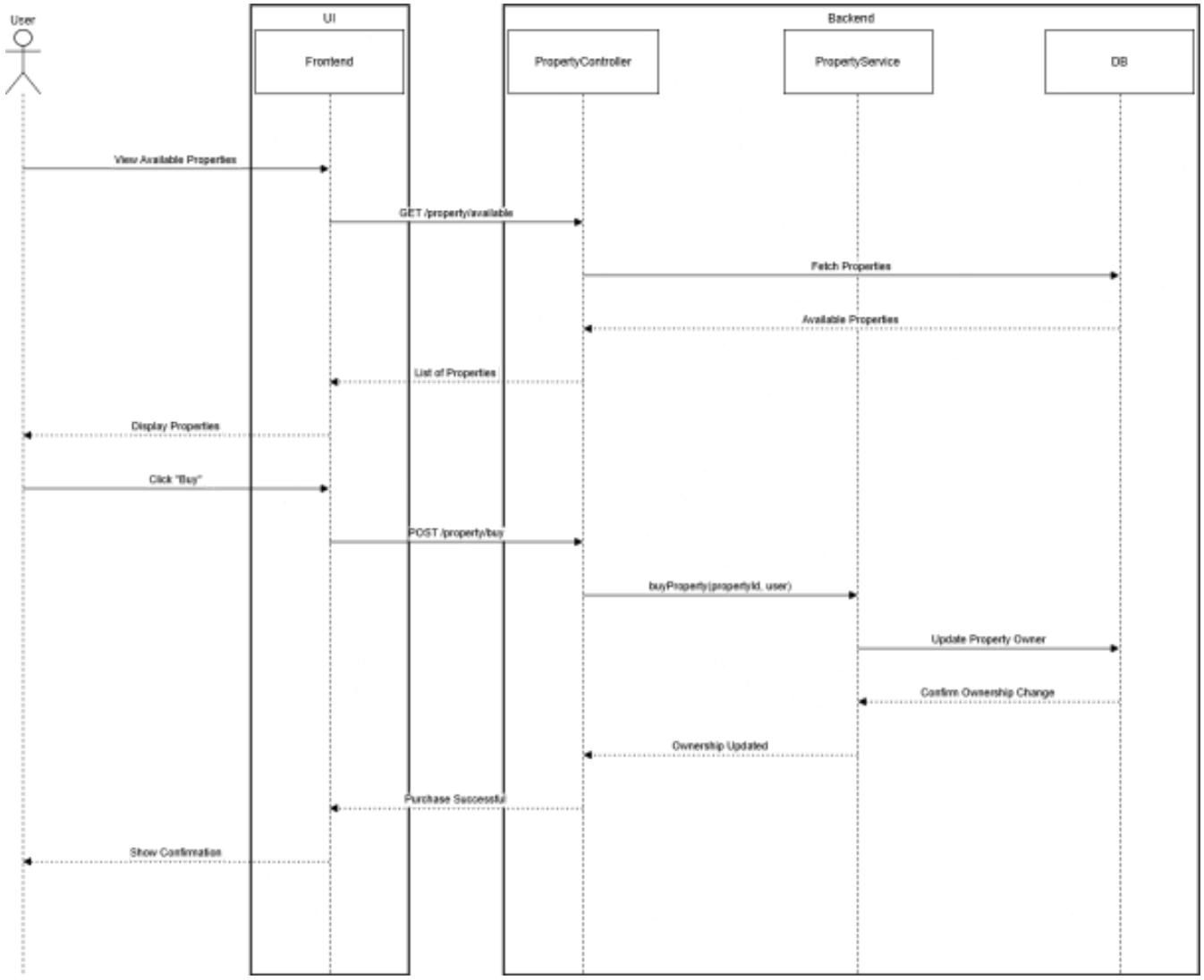

User
View Transactions
Real Estate Management System
Authenticate User
Retrieve Transaction History

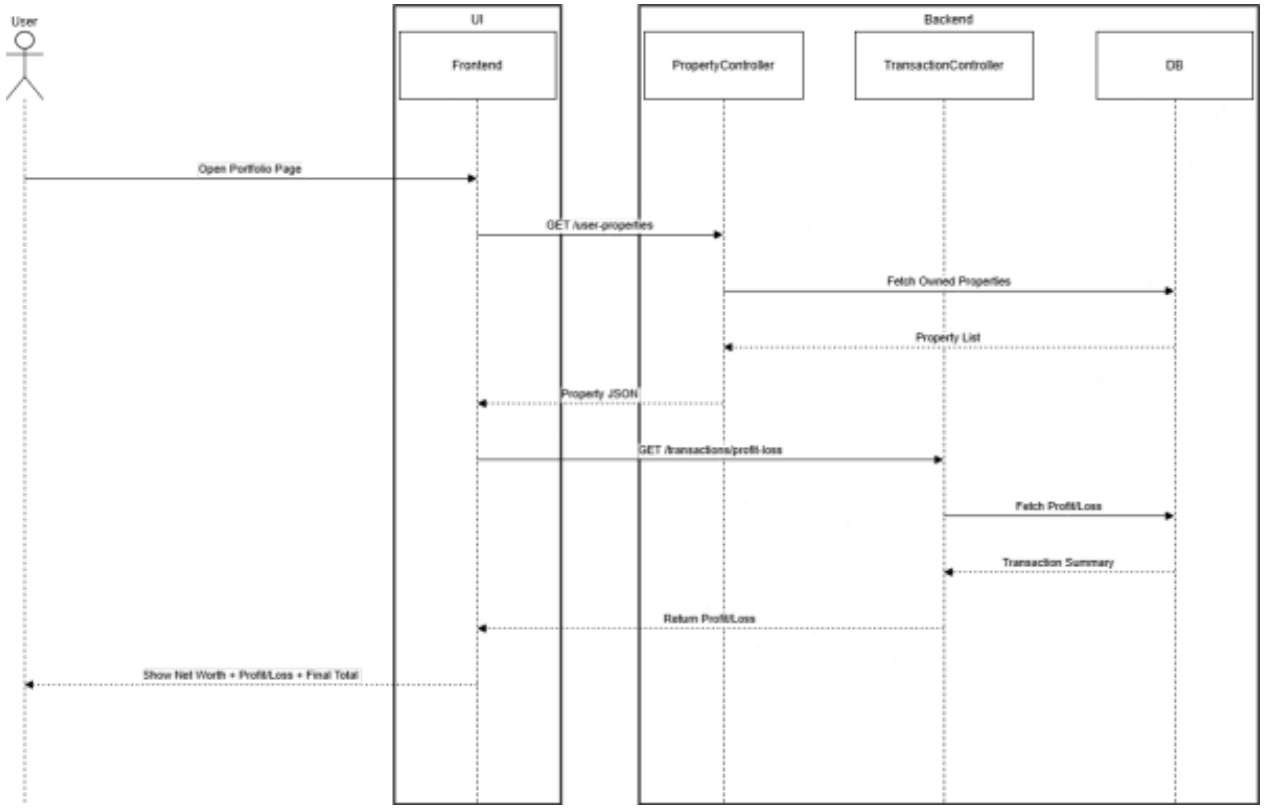

User
Upload Sale Deed Document
Authenticate User
Validate PDF Format
Real Estate Management System
Store File on Server
Update Property Record

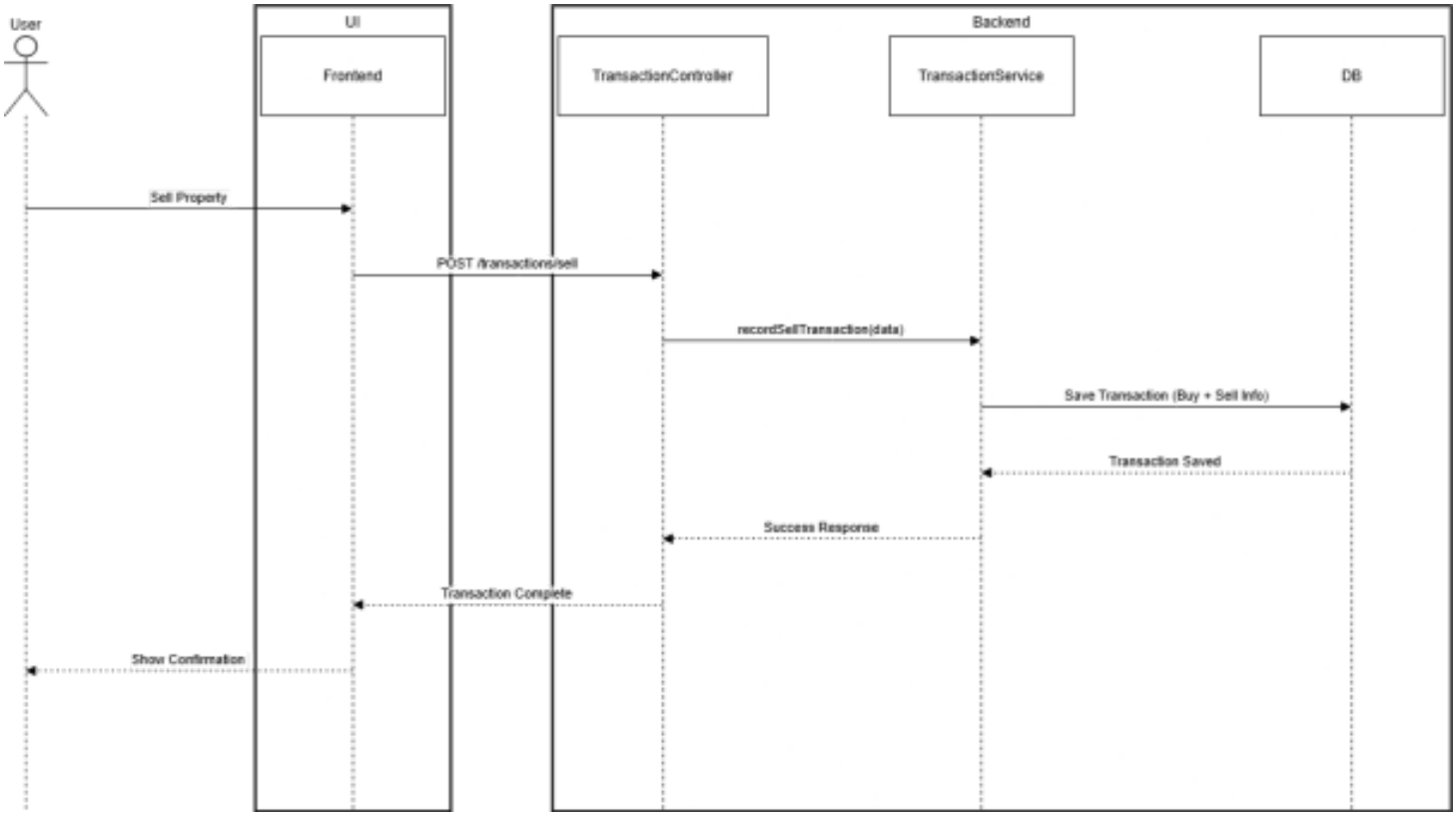


2. Major Use case




User
UI
Frontend
Backend
PropertyController
PropertyService
DB
View Available Properties
GET /property/available
Fetch Properties
Available Properties
List of Properties
Display Properties
Click "Buy"
POST /property/buy
buyProperty(propertyId, user)
Update Property Owner
Confirm Ownership Change
Ownership Updated
Purchase Successful
Show Confirmation





Architecture Patterns, Design Principles, and Design Patterns:

Architecture Patterns

1. Model–View–Controller (MVC) Pattern

The MVC pattern is the core architectural style used in the project. It separates the application into three interconnected components:

- Model: Contains the business logic and data. In this project, classes like `User`, `Property`, and `Transaction` represent the model.
- View: Represents the UI or frontend. Here, HTML pages (like `dashboard.html`, `buy.html`) serve as views and are rendered to the user.
- Controller: Handles incoming HTTP requests and routes them appropriately. Classes like `AuthController` and `PropertyController` act as controllers in the system.

2. Layered (N-Tier) Architecture

The system follows a layered architecture, which divides the application into distinct logical layers:

- **Presentation Layer**: Includes controllers and static HTML files.
- **Service/Business Logic Layer**: Includes services like `UserService` and `TransactionService`, which implement the business logic.
- **Data Access Layer**: Includes repositories such as `UserRepository`, `PropertyRepository`, etc., which interact with the database.

3. Repository Pattern

This pattern is used to abstract away the persistence logic. By extending `JpaRepository`, each repository interface provides built-in methods for database operations such as saving, finding, and deleting entities.

Design Patterns

1. Singleton Pattern

Description:
Ensures that only one instance of a class is created and used throughout the application. In Spring Boot, most

service and repository beans (`@Service`, `@Repository`) are automatically singletons.

2. DTO (Data Transfer Object)

Description:
Used to transfer specific data across application layers without exposing full entity objects. In our project, `ProfitLossDTO` encapsulates only the data needed for the frontend to display profits/losses.

3. Controller Pattern

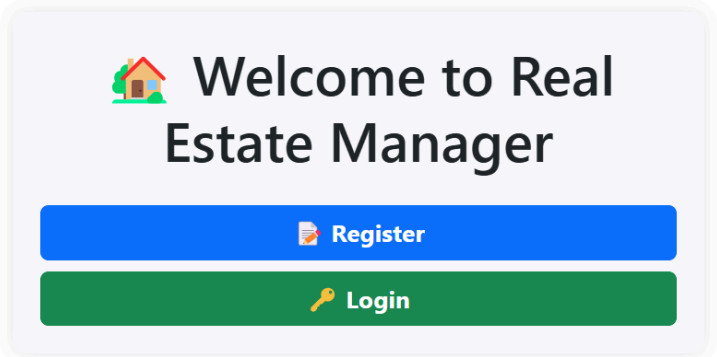
Description:
Acts as the entry point to handle requests, delegate logic, and return responses. Spring Boot's `@RestController` classes like `AuthController`, `PropertyController` implement this pattern.

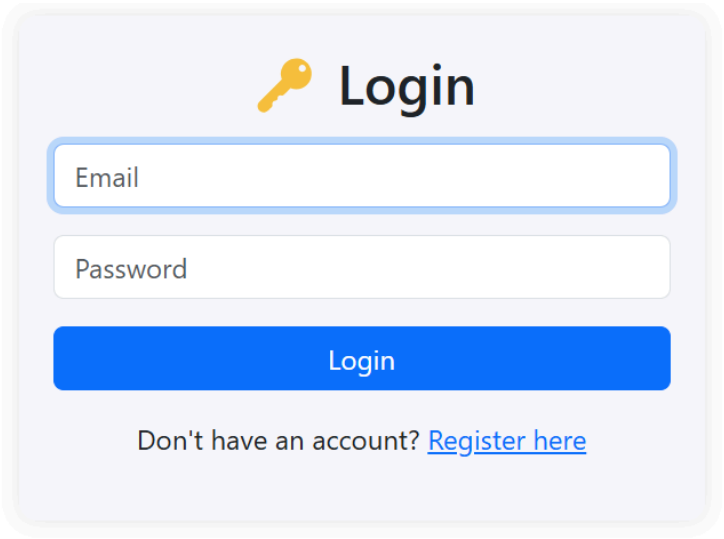
Github link to the Codebase:

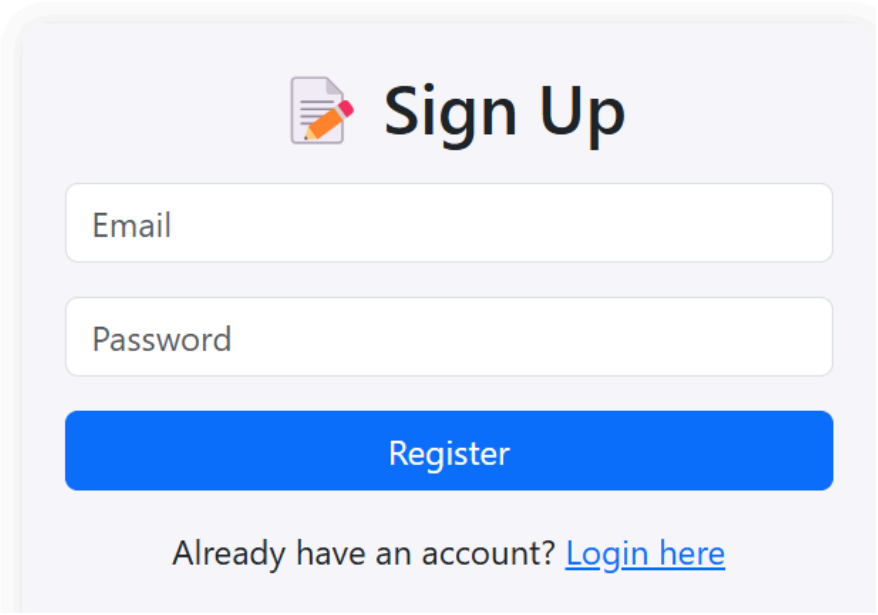
https://github.com/siddhantgopi/Java-Spring-Boot-Project.git

Screenshots

UI:




Login
Email
Password
Login
Don't have an account? Register here


Sign Up
Email
Password
Register
Already have an account? Login here

🎯 Welcome to Your Dashboard

🏠 Add Property

📋 View All Properties

💰 Buy/Sell Property

📊 See Portfolio

💳 Transactions

🚪 Logout

🏠 Add Property

Property Name:

Location:

Dimensions (feet x feet):

Price per Square Feet ($):

Date of Purchase:

mm/dd/yyyy

Sale Deed (PDF):

Choose File | No file chosen

Submit

All Properties

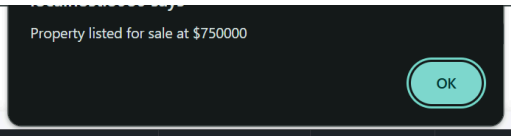
Name	Location	Dimensions (sq ft)	Price per Sq Ft	Total Price	Document
Suburban Home	Austin, TX	2500	$675.25	$1688125.00	View Document
789 Elm Street	Metropolis	2000	$175.00	$350000.00	View Document

Available Properties for Purchase

Name	Location	Dimensions	Price	Action
789, 5th ave, Apt 25B	New York City	1200 sq.ft	$1500000	Buy
Luxury Villa	Los Angeles, CA	4500 sq.ft	$6500000	Buy
Modern Apartment	New York, NY	1200 sq.ft	$1020900	Buy
Beach House	Miami, FL	3000 sq.ft	$3000000	Buy
Penthouse Suite	Chicago, IL	1800 sq.ft	$2100000	Buy
Lakefront Cabin	Denver, CO	2000 sq.ft	$1451600	Buy

Your Properties for Sale

Name	Location	Dimensions	Price	Set Selling Price	Action
Luxury Villa	Los Angeles, CA	4500 sq.ft	$6500000	Enter pri	Sell
Suburban Home	Austin, TX	2500 sq.ft	$1688125	Enter pri	Sell
789 Elm Street	Metropolis	2000 sq.ft	$350000	Enter pri	Sell



Name	Location	Dimensions	Price	Set Selling Price	Action
Luxury Villa	Los Angeles, CA	4500 sq.ft	$6500000	750000	Sell
Suburban Home	Austin, TX	2500 sq.ft	$1688125	Enter pri	Sell
789 Elm Street	Metropolis	2000 sq.ft	$350000	Enter pri	Sell

Portfolio Overview

Net Holdings: $2,038,125
Total Profit/Loss: $-1,612,500
Adjusted Value: $425,625

Property ID	Buy Price	Sell Price	Profit/Loss	Sell Date
7	$300,000	$300,000	$0	2025-03-28 16:03:52.615228
7	$300,000	$300,000	$0	2025-03-28 16:11:18.45348
8	$350,000	$210,000	$-140,000	2025-03-28 16:39:37.513555
8	$350,000	$250,000	$-100,000	2025-03-28 22:26:19.918781
7	$300,000	$260,000	$-40,000	2025-03-28 22:31:21.592557
8	$350,000	$315,000	$-35,000	2025-03-29 21:00:12.382669
3	$3,000,000	$3,000,000	$0	2025-03-29 21:09:18.100482
6	$0	$1,451,600	$1,451,600	2025-03-29 21:11:12.066658
2	$0	$1,020,900	$1,020,900	2025-03-29 21:11:58.961191
3	$3,000,000	$3,000,000	$0	2025-03-29 21:12:07.417936
4	$0	$1,980,000	$1,980,000	2025-03-29 21:13:01.130269
5	$1,688,125	$1,688,125	$0	2025-03-29 21:13:08.161291
1	$6,500,000	$6,500,000	$0	2025-03-30 10:40:14.771242
1	$6,500,000	$750,000	$-5,750,000	2025-04-26 12:05:41.656067

Transaction History

Date	Price	Type	Property ID
3/26/2025, 9:08:27 PM	$250000.00	ADD	7
3/26/2025, 11:25:13 PM	$1500000.00	ADD	9
3/28/2025, 12:10:33 AM	$1500000.00	SELL	9
3/28/2025, 12:10:49 AM	$5402250.00	BUY	1
3/28/2025, 4:03:52 PM	$300000.00	SELL	7
3/28/2025, 4:11:07 PM	$300000.00	BUY	7
3/28/2025, 4:11:18 PM	$300000.00	SELL	7
3/28/2025, 4:13:49 PM	$0.00	BUY	8
3/28/2025, 4:39:37 PM	$210000.00	SELL	8
3/28/2025, 4:39:56 PM	$210000.00	BUY	8
3/28/2025, 10:26:19 PM	$250000.00	SELL	8
3/28/2025, 10:27:34 PM	$250000.00	BUY	8
3/28/2025, 10:31:12 PM	$300000.00	BUY	7
3/28/2025, 10:31:21 PM	$260000.00	SELL	7
3/29/2025, 9:00:12 PM	$315000.00	SELL	8
3/29/2025, 9:06:31 PM	$0.00	BUY	3
3/29/2025, 9:09:18 PM	$3000000.00	SELL	3
3/29/2025, 9:10:42 PM	$3000000.00	BUY	3
3/29/2025, 9:10:56 PM	$0.00	BUY	6

Individual contributions of the team members:

Name	Module worked on
Siddhant P Gopi	Buy and sell, Transactions
Shreyas S Magadi	Add property, Portfolio
Shreyas Gowda	Login, Register