Kabul University

Computer Since Faculty

Software Engineering Department

SRS

<Real Estate >

Version 1.0 approved

Prepared by <Maiwand Shirzad>

Assistant Teacher: Salim Hamdard

2025/3/26

0. Introduction

0.1 Purpose

The purpose of this document is to outline the requirements for a real estate website, which allow users to search, list and inquiry for sale, buy, and rent items. The system will provide an intuitive user experience for buyers, sellers, and agents.

0.2 Project Scope

This real estate website will include:

- Items Listing with images, prices, and details.

- User authentication (buyers, sellers, agents)

- Contact form for inquiries.

- A search and filter system for items.

- Admin dashboard for manage listings.

- A blog for real estate insights

0.3 Definition and abbreviations:

- Agent: A person who lists and sell properties.

- Buyer: A user interested in purchasing or renting a property.

- Admin: The owner who approve and manage listings.

1. Functional requirements

1.1 User Roles & Features

- Visitors

  o View Items listings

  o Search and filter Items.

  o Read blogs

  o Contact agents via forms.

- Registered Users (Buyers & Sellers)

- o   Sign up/ Login

- o   Save favorite properties

- o   Post Item listings (for Sellers)

- o   Manage their (edit, delete, up\

- o   date)

- Admin

  - o   Approve/ Reject Item listings.

  - o   Manage users and their roles.

  - o   View inquiries and messages
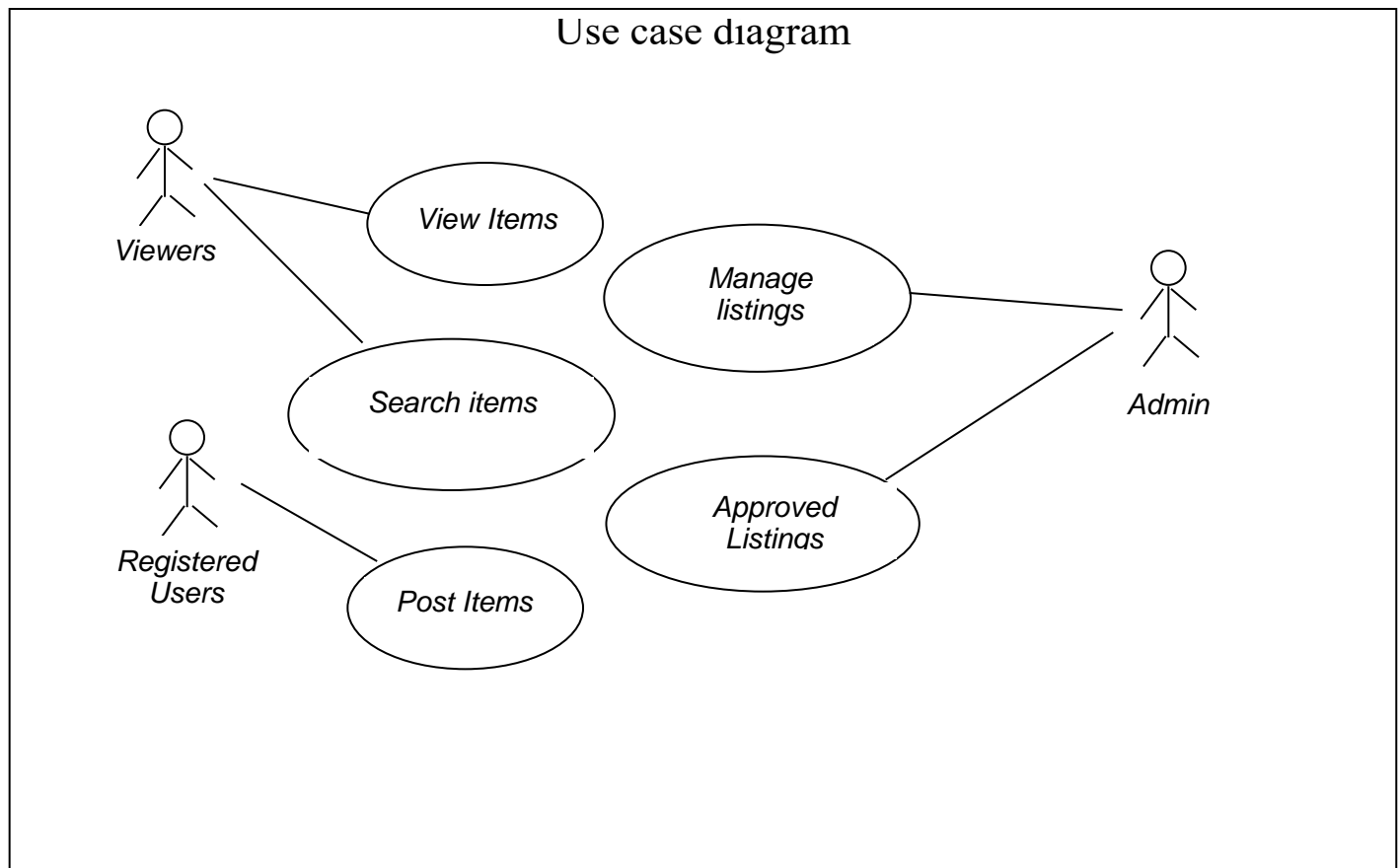
2. System Features

- **User Authentication:** Login/Signup system.

- **Items listings:** Add, Update, delete, and view items details.

- **Search & Filter:** Based on location, price, range, type, etc.

- **Contact System:** Form for user to contact items owner.

- **Responsiveness:** Mobile-friendly layout.

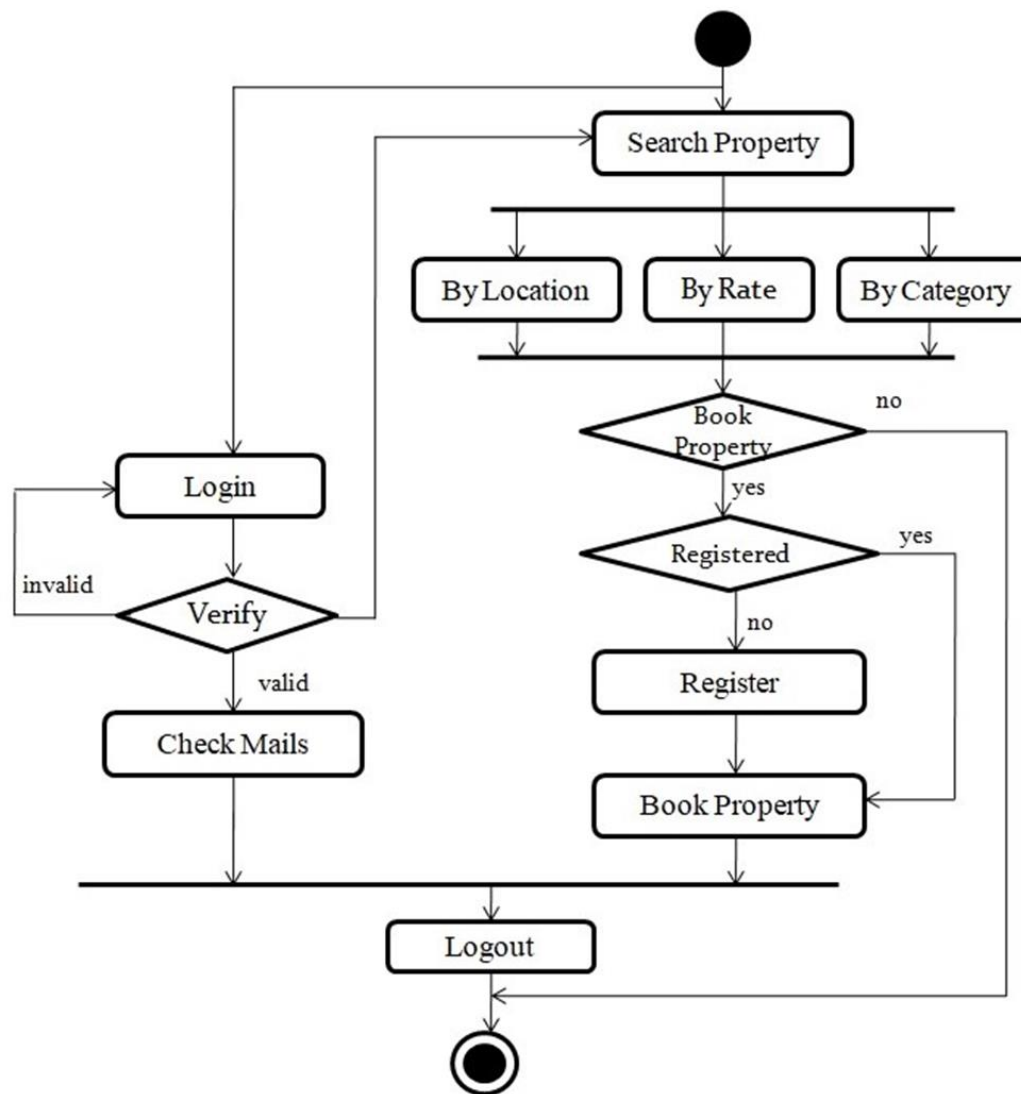- **Admin dashboard:** Manage listings and users.

3. Non-Functional Requirements

- Security: User authentication with encryption.

- **Performance:** Website should load within 2-3 seconds.

- **Compatibility:** works on Desktop, tablet, and mobile**.**

- **Scalability:** Should support at least 1000 item listings at lunch.

4. Tech Stack

- **Frontend:** HTML, CSS, JavaScript (React.js or Vanilla JS)

- **Backend:** Node.js with Express, or Python with Django.

- **Database:** MySQL or MongoDB.

- **Hoisting:** Firebase, Vercel, or AWS.

Use case diagram

Activity Diagram for a Real Estate system

Class Diagram For a Real Estate Project

User

-User Id (int)
-Name (string)
-Email (string)
-Phone (int)

+Sign UP ()
+Login ()
+Browse Listings ()
+Upload Documents ()
+Images ()
+Status ()

Account Details

-ID: int
-Customer ID: int

Items

-Item ID int
-Item Address: char
-Item Price: int

Seller

+seller Id (int)
+Name (string)
+Email (string)
-Phone (int)
-Address: char

+Sign UP ()
+Login ()
+Post Listings ()
+Manage Applications ()

*   *
1
*
1 *   1
*   *