Software Requirements Specification Document

Property Buy/Sell Website

April 18, 2018

Rishabh Soni, Nihit Neema, Arpit Sharma, Namit Sarode, Siddharth Gupta

Submitted in partial fulfillment of the requirements of
IT 341 Object-Oriented System Development
Spring 2018

Table of Contents

TABLE OF CONTENTS ..... **II**

REVISION HISTORY ..... **IV**

DOCUMENT APPROVAL ..... **IV**

1. INTRODUCTION **1**

1.1 PURPOSE 1
1.2 SCOPE 1
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS 2
1.4 REFERENCES 2
1.5 OVERVIEW 2

2. GENERAL DESCRIPTION **2**

2.1 PRODUCT PERSPECTIVE 2
2.2 USER CHARACTERISTICS 3
2.3 SYSTEM ENVIRONMENT 3
2.4 GENERAL CONSTRAINTS 3
2.5 ASSUMPTIONS AND DEPENDENCIES 3

3. SPECIFIC REQUIREMENTS **3**

3.1 EXTERNAL INTERFACE REQUIREMENTS 4
3.1.1 User Interfaces *4*
3.1.2 Hardware Interfaces *4*
3.1.3 Software Interfaces *4*
3.1.4 Communications Interfaces *4*
3.2 FUNCTIONAL REQUIREMENTS 4
3.3 NON-FUNCTIONAL REQUIREMENTS 12
3.3.1 Performance: *12*
3.3.2 Reliability: *12*
3.3.3 Availability: *12*
3.3.4 Security: *12*
3.3.5 Maintainability:. *12*
3.3.6 Portability: *12*
3.4 DESIGN CONSTRAINTS 12
3.5 OTHER REQUIREMENTS 12

4. DESIGN & DEVELOPMENT **12**

4.1 SOFTWARE PROCESS MODEL 13
4.1.1 Introduction *13*
4.2 OVERALL ARCHITECTURE 14
4.3 SPRINT 1 14
4.3.1 Description *14*
4.3.2 Satisfying Requirements *14*
Unsubscribe from newsletter *14*
4.3.3 Design *15*
4.3.4 Development *15*
4.4 SPRINT 2 16
4.4.1 Description *16*
4.4.2 Satisfying Requirements *16*
4.4.3 Design *17*

4.4.4 Development *18*
4.5 SPRINT 3 21
4.5.1 Description *21*
4.5.2 Satisfying Requirements *22*
4.5.3 Design *22*
4.5.4 Development *22*

Revision History

Date	Description	Author	Comments
02/01	Draft Version	Siddharth	Initial Draft
02/13	Final Version_v1.00	Siddharth	Final Draft for the first deliverable
3/5	Final Version_v2.00	Siddharth	Final Draft for the second deliverable(Sprint-1)
3/28	Final Version_v3.00	Siddharth	Final Draft for the third deliverable(Sprint-2)
4/18	Final Version_v4.00	Siddharth	Final Draft for the third deliverable(Sprint-3)

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	{Your Name}		
	Dr. Rishi Kanth Saripalle		

1. Introduction

The objective of this Software Requirements Specifications (SRS) document is to provide an insight on "Property Buy and Sell Website". This document includes the business preposition of the project, the functional / non-functional requirements and the well-structured architecture diagrams for better understanding of the design. In addition to this, detailed project work schedule is mentioned in the subsequent sections where in project timelines along with the sprints are mentioned.

From the business perspective, the "Property Buy and Sell Website" is web portal for those potential users who are in the search of a scout for any king of property like student apartments, residential or commercial. The website is having dynamic content which has the intended updated information. From the sellers' end, this web portal is a platform to showcase the respective on sale/rent accommodations/land/real estate deals to the customers. This will help in direct contact and negotiations between the buyers and sellers.

1.1 Purpose

The purpose of the SRS document is to provide a deep dive regarding this project, ranging from the intended business goals, requirements, user interfaces and software/hardware constraint and dependencies. From the project team and client/stakeholders' perspective, the SRS details out the SDLC plan and project timelines with the expected deliverables.

1.2 Scope

This "Property Buy and Sell Website" will be of great utility to the real estate line of business where buyers always struggle a lot to get the right property of their choice and need. Also, it should be in their budget. On the other hand, the seller always strives to pitch in the best price for his property and always wants a higher priority in terms of the showcase to the customers. So, this website project will basically designed to create a "Win-win" situation for both seller and the buyer through credible information (deals) and also by making the website user friendly with content that is easy to understand.

However, this website is purely based on fictional and mocked up data/information. The emails and sms notifications have a pre-defined template. It is not a real time/dynamic website. This website is not having any dedicated application for mobile/tablet devices (e.g. Apple Store or Google play). This project targets the audience which is related to or dealing with property matters. The detailed requirements will follow through in the subsequent sections of this SRS.

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition
FAQ	Frequently Asked Questions
Seller	An individual/organization that posts the property details on the website for rent/sale purposes.
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
Stakeholder	Any person with an interest in the project who is not a developer.
User	A person that is looking for a property type on rent/sale.

1.4 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

1.5 Overview

The remainder portion of this document will talk about the general description of the project including project perspective, user characteristics, system environment, general constraints and assumptions/dependencies of this project. The section-3 provides details of specific requirements which include external interface requirements, functional requirements, non-functional requirements, design constraints and other requirements if any. The section-4 provides details about design and development details. This will include which software process model is implemented for this project along with the project timelines and detailed sprint plans.

2. General Description

The “Property Buy and Sell Website” is a web portal for those potential users who are in the search of a scout for any king of property like student apartments, residential or commercial. The website is having dynamic content which has the intended updated information. From the seller’s end, this web portal is a platform to showcase the respective on sale/rent accommodations/land/real estate deals to the customers. This will help in direct contact and negotiations between the buyers and sellers.

2.1 Product Perspective

The following is the high-level specifications (scope) of this project:

1. To allow the user to search for potential properties of any kind for rent/buying purpose.
2. To allow the user to have access to complete details of any property.
3. To allow user to negotiate/contact the potential buyers directly.
4. To allow user for posting queries regarding any property information.

5. To allow buyers to post their property details on the website with high visibility.

2.2 User Characteristics

This project targets the audience which is related to or dealing with property matters. For example, the real estate organizations or the agents who are dealing with land, houses and commercial properties. This website gives a wide platform to advertise, buy and sell, rent a commercial property or a residential property. The basic qualification and pre-requisite of the users should be that they are well versed with operating a computer, internet and website browsing.

The following are the intended users of the website which will affect the specific requirements:

- Seller: A person that posts the property details on the website for rent/sale purposes.
- Customer/ User: A person that is looking for a property type on rent/sale.

2.3 System Environment

This website has the following system environment requirements:
Platform: Windows
Browsers (Java and Java Script compatible browser*)*: Google Chrome, Mozilla Firefox, IE
JAVA IDE: Eclipse
Database: MySQL

2.4 General Constraints

The following are the general constraints for this website:

- Business Constraints: There is no prior business domain knowledge transformation session held, before developing this website. The knowledge pertaining to real estate property management is somewhat limited.
- Technical Constraints: This website will be focused only for desktop users. There is no dedicated application for mobile/tablet users.
- Project Team Constraints: The project team consists of members who have primarily worked as testers and system analyst. There is not much expertise in the development front, in the project team.

2.5 Assumptions and Dependencies

The users are expected to have basic domain knowledge about the real estate line of business. They should be well versed with the operations of a computer and internet.

3. Specific Requirements

The following are the specific requirements:

3.1 External Interface Requirements

3.1.1 User Interfaces

This section is not applicable for this project.

3.1.2 Hardware Interfaces

"The System must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable."[1]

3.1.3 Software Interfaces

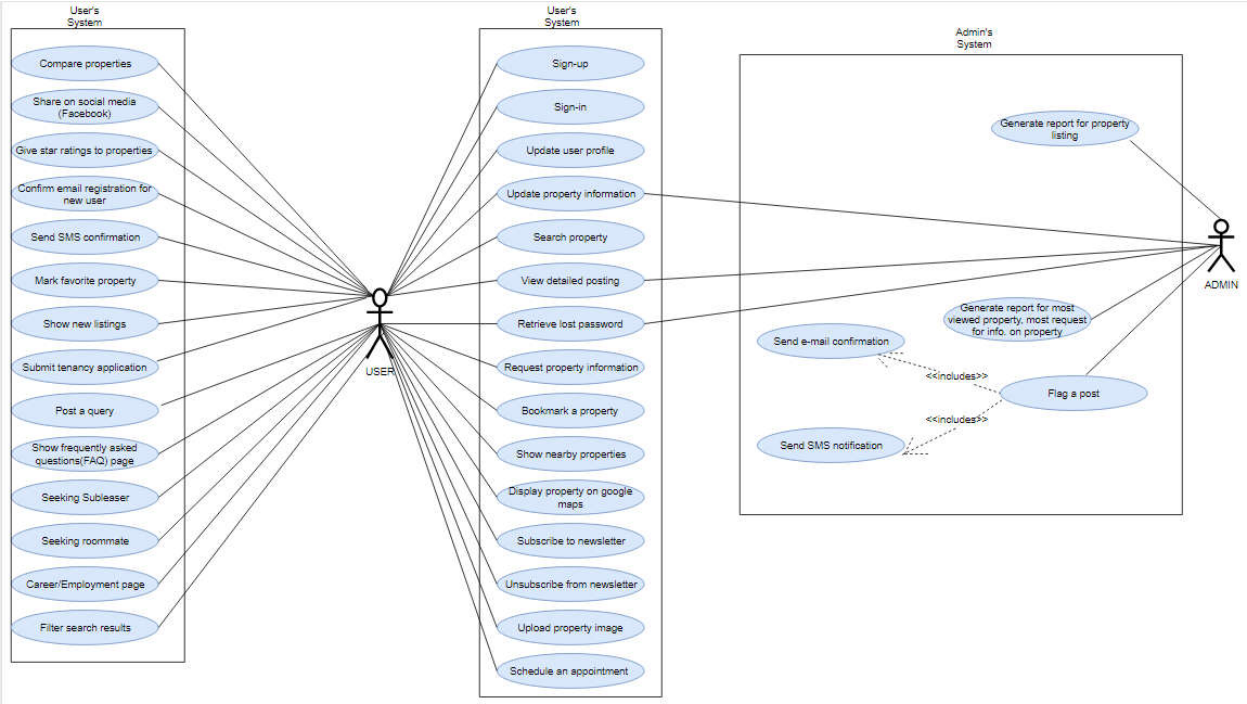
Platform: Windows/Linux/iOS
Browsers: Google Chrome, Mozilla Firefox, IE
Database: MySQL Workbench 6.3

3.1.4 Communications Interfaces

This section is not applicable for this project.

3.2 Functional Requirements

Below is the use case diagram for this website:



3.2.1 Sign-in User

3.2.1.1 Description: User can sign in with a valid username and password for successful login
3.2.1.2 Actor(s): User who has already registered.
3.2.1.3 Conditions
3.2.1.4.1 Pre: User must have a valid user name and password.

3.2.1.4.2 Post: User will log in and navigated to homepage.
3.2.1.4 Related Use Case: N/A

3.2.2 Sign-up/Register User

3.2.2.1 Description: User can register or sign up by creating a new account by filling the details like personal info, occupational info etc.
3.2.2.2 Actor(s): User who has not already registered.
3.2.2.3 Conditions:
3.2.2.4.1 Pre: User should not be registered to this website.
3.2.2.4.2 Post: A successful user profile will be created, shown and stored into the database
3.2.2.4 Related Use Case: N/A

3.2.3 Show new property listing

3.2.3.1 Description: User can view the new property listing.
3.2.3.2 Actor(s): User
3.2.3.3 Conditions
3.2.3.4.1 Pre: User requires to be logged in**.**
3.2.3.4.2 Post: User will see a new list of properties on the webpage page.
3.2.3.3 Related Use Case: Sign-in user

3.2.4 Flagging a post

3.2.4.1 Description: A post can be flagged by the admin for various reasons like credible information, more information required etc.
3.2.4.2 Actor(s): Admin
3.2.4.3 Conditions:
3.2.4.4.1 Pre: Admin must be logged in.
3.2.4.4.2 Post: Both admin and user will see the flagged post after logged in.
3.2.4.4 Related Use Case: Send e-mail notification and send SMS notification

---

3.2.5 Send e-mail notification for flagged post

3.2.5.1 Description: User can get flagged post details of the property through e-mail.
3.2.5.2 Actor(s): Admin
3.2.5.3 Conditions:
3.2.5.4.1 Pre: User must have a valid e-mail id
3.2.5.4.2 Post: User will receive an e-mail notification
3.2.5.4 Related Use Case: Flagging a post by Admin

3.2.6 Send SMS notification for flagged post

3.2.6.1 Description: User can get flagged post details of the property through SMS
3.2.6.2 **Actor(s):** Admin
3.2.6.3 Conditions:
3.2.6.4.1 Pre: User must have a valid phone number.
3.2.6.4.2 Post: User will get an SMS on their phone.
3.2.6.4 **Related Use Case:** Flagging a post by Admin

3.2.7 Show FAQ's

3.2.7.1 Description: User can see the Frequently Asked Question on website.
3.2.7.2 Actor(s): User
3.2.7.3 Conditions:
3.2.7.4.1 Pre: User can navigate to FAQ webpage use this feature
3.2.7.4.2 Post: User can quickly go through the frequently asked questions for easy navigation and whenever there's a problem while using the website.
3.2.7.4 Related Use Case: N/A

3.2.8. Update User Profile

3.2.8.1 Description: This use case gives the capability for the user to edit their user profile. They can add, delete or, modify the information they have put on their profiles.
3.2.8.2 Actor: User
3.2.8.3 Conditions
3.2.8.4.1 Pre: A user should be signed up first and have a registered account before they can update their profile.
3.2.8.4.2 Post: The user profile would be updated successfully and a pop up will be displayed that the profile has been updated.
3.2.8.4 **Related Use Case:** Sign up and Sign in

3.2.9 Search Property through keywords (like zip code, state, property type)

3.2.9.1 **Description:** A user can search a property which they wish to buy/rent according to their needs.
3.2.9.2 **Actor:** User
3.2.9.3 **Conditions**
3.2.9.4.1 Pre: A user must be logged in to search any property.
3.2.9.4.2 Post: A user would see their search results.
3.2.9.4 Related Use Case: None

3.2.10 View Detailed Posting

3.2.10.1 **Description:** This functionality would open the details page of the searched property.

3.2.10.2 **Actor:** User and Admin

3.2.10.3 Conditions

3.2.10.4.1 Pre: A user must be logged in successfully to use this functionality.

3.2.10.4.2 Post: A user would see the details page of their searched property.

3.2.10.3 **Related Use Case:** Search Property

3.2.11 Schedule an Appointment

3.2.11.1 Description: A user can apply for scheduling an appointment with the proper owner/dealer (with email confirmation).

3.2.11.2 Actor: User

3.2.11.3 Conditions

3.2.11.4.1 Pre: A user must be logged in to schedule an appointment. The user must be on the searched property page to schedule an appointment with the dealer/owner.

3.2.11.4.2 Post: An E-mail confirmation will be sent to the user.

3.2.11.4 Related Use Case: None

3.2.12 Request for more property information.

3.2.12.1 Description: A user can request more information on a specific property of their choice from the admin.

3.2.12.2 Actor: User

3.2.12.3 Conditions

3.2.12.4.1 Pre: A user must be on a specific property page to request for more information on that property.

3.2.12.4.2 Post: The user would get the additional information on the property.

3.2.12.4 Related Use Case: None

3.2.13 Recent searched properties.

3.2.13.1 Description: This functionality enables the user to view the most recent searched properties

3.2.13.2: Actor: User

3.2.13.3 Conditions

3.2.13.4.1 Pre: The user must be successfully logged in.

3.2.13.4.2 Post: The most recent searched property will be displayed.

3.2.13.4 Related Use Case: None

3.2.14 Highlight (Special Showcase) property

3.2.14.1 Description: This functionality enables a user to bookmark a property for future reference.

3.2.14.2 Actor: User

3.2.14.3 Conditions:

3.2.14.4.1 Pre: The user must be successfully logged in.

3.2.14.4.2 Post: A user can successfully bookmark a property.

3.2.14.4 Related Use Case: None

3.2.15 Subscribe to newsletter
3.2.15.1 Description- User can subscribe to weekly newsletter through email.
3.2.15.2 Actor(s) - User
3.2.15.3 Conditions
3.2.15.4.1 Pre: User needs to login first.
3.2.15.4.2 Post: The user is successfully subscribed for receiving newsletter.
3.2.15.4 Related Use Case: Unsubscribe to Newsletter.

3.2.16 Unsubscribe from newsletter

3.2.16.1 Description- User can unsubscribe from receiving newsletters through email by just one click.
3.2.16.2 Actor(s) - User
3.2.16.3 Conditions
3.2.16.4.1 Pre: Customer needs to login first.
3.2.16.4.2 Post: The user is successfully unsubscribed from receiving newsletter.
3.2.16.4 Related Use Case: Subscribe to Newsletter.

3.2.17 Update Property Information
3.2.17.1 Description- The admin and user can update his/her property details with "Update Information" tab.
3.2.17.2 Actor(s) - Admin and user
3.2.17.3 Conditions
3.2.17.4.1 Pre: Admin/user needs to login first.
3.2.17.4.2 Post: Admin/user successfully updates the property.
3.2.17.4 Related Use Case: None

3.2.18 Show Surrounding Properties (via Web services (e.g., Zillow)

3.2.18.1 Description- This functionality will allow the user to have a look at the properties surrounding his/her selected property.
3.2.18.2 Actor(s) - User
3.2.18.3 Conditions
3.2.18.4.1 Pre: User needs to select at-least one of the property listed.
3.2.18.4.2 Post: User successfully views all the surrounding properties.
3.2.18.4 Related Use Case: None

3.2.19 Show Property on google maps

3.2.19.1 Description- This functionality will allow the user to have a look at the geographical map of the whole area in which the selected property is located.

3.2.19.2 Actor(s) - User
3.2.19.3 Conditions
3.2.19.4.1 Pre: User needs to select at-least one of the property listed.
3.2.19.4.2 Post: User successfully views the whole locality and nearby places.
3.2.19.4 Related Use Case: None

3.2.20 Regular Property Listing

3.2.20.1 Description- This functionality will enable the admin to download the most recent listing of all the properties in pdf format.
3.2.20.2 Actor(s) - Admin
3.2.20.3 Conditions
3.2.20.4.1 Pre: Admin needs to login first.
3.2.20.4.2 Post: Admin successfully downloads the most recent property listing.
3.2.20.4 Related Use Case: None

3.2.21 Reports generation (e.g., most viewed properties, most request-for-info properties, etc.)

3.2.21.1 Description- This functionality will enable the admin to view an auto generated report. The report will be generated once in every month.
3.2.21.2 Actor(s) - Admin
3.2.21.3 Conditions
3.2.21.4.1 Pre: Admin needs to login first.
3.2.21.4.2 Post: Admin successfully views the generated report.
3.2.21.4 Related Use Case: None

3.2.22 Retrieve lost password

3.2.22.1 Description: This functionality will allow user to retrieve password. The password would be received in the user's registered email.
3.2.22.2 Actor(s): User
3.2.22.3 Conditions
3.2.22.4.1 Pre: User doesn't need to be logged in**.**
3.2.22.4.2 Post: User will receive the password in their email account.
3.2.22.4 Related Use Case: Sign-Up user

3.2.23 Compare Properties

3.2.23.1 Description: If a user wants to compare two properties then he can use this feature.
3.2.23.2 Actor(s): User
3.2.23.3 Conditions:
3.2.23.4.1 Pre: User must be logged in**.**
3.2.23.4.2 Post: User will see two properties side by side.

3.2.23.4 Related Use Case: Search Property

3.2.24 Share on social media(Facebook)

3.2.24.1 Description: Property details can be shared with this use case
3.2.24.2 Actor(s): User
3.2.24.3 Conditions:
3.2.24.4.1 Pre: User must be logged in. A property should be selected to share on social media
3.2.24.4.2 Post: Details of the property will be posted on social media
3.2.24.4 Related Use Case: N/A

3.2.25 Give star rating to property

3.2.25.1 Description: A property can be given feedback by the user in form of star ratings.
3.2.25.2 Actor(s): User
3.2.25.3 Conditions:
3.2.25.4.1 Pre: User must be logged in.
3.2.25.4.2 Post: User would be able to give star ratings and the ratings would be visible.
3.2.25.4 Related Use Case: N/A

3.2.26 Confirm email registration

3.2.26.1 Description: New users can create their account and confirm their email id.
3.2.26.2 Actor(s): User
3.2.26.3 Conditions:
3.2.26.4.1 Pre: User must fill the registration form and give a valid email id.
3.2.26.4.2 Post: User will be redirected to his account
3.2.26.4 Related Use Case: SMS confirmation

3.2.27 Send SMS confirmation

3.2.27.1 Description: New user can get the account confirmation details by SMS
3.2.27.2 Actor(s): User
3.2.27.3 Conditions:
3.2.27.4.1 Pre: User must fill the registration form and give a valid phone number
3.2.27.4.2 Post: User will get SMS on their phone
3.2.27.4 Related Use Case: Email Confirmation

3.2.28 Mark a property as “favorite”

3.2.28.1 Description: A property that interests the user can be made favorite
3.2.28.2 Actor(s): User
3.2.28.3 Conditions:
3.2.28.4.1 Pre: User must be logged in. User must be on the desired property’s page.

3.2.28.4.2 Post: A property would have a favorite mark on it.

3.2.28.4 Related Use Case: N/A

3.2.29 Filter and sort by functionality

3.2.29.1 Description: User can filter and sort the search results based on several criteria/factors like number of baths, bedrooms, price etc**.**

3.2.29.2 Actor(s): User

3.2.29.3 Conditions:

3.2.29.4.1 Pre: User must be logged in. User must select the filter/sort by option for viewing search results.

3.2.29.4.2 Post: The search results should be displayed based on the selected factors.

3.2.29.4 Related Use Case: N/A

3.2.30 Submit an application for tenancy by the user.

3.2.30.1 Description: User can submit an application for tenancy on the website.

3.2.30.2 Actor(s): User

3.2.30.3 Conditions:

3.2.30.4.1 Pre: User must be logged in. User must be on the "Application for tenancy page". All the required details should be filled by the user.

3.2.30.4.2 Post: The application for tenancy will be successfully submitted.

3.2.30.4 Related Use Case: N/A

3.2.31 Post a query on the website regarding a property.

3.2.31.1 Description: User can post a query related to any property on the website.

3.2.31.2 **Actor(s):** User

3.2.31.3 Conditions:

3.2.31.4.1 Pre: User must be logged in. User must post a query with his contact information.

3.2.31.4.2 Post: The query acknowledge message will be displayed on the webpage.

3.2.31.4 Related Use Case: N/A

3.2.32 View "Need Sub leaser" listing on the website.

3.2.32.1 Description: User can view "Need sub leaser" listing on the website.

3.2.32.2 Actor(s): User

3.2.32.3 Conditions:

3.2.32.4.1 Pre: User must be logged in. User must be on the "Need sub leaser" webpage.

3.2.32.4.2 Post: A list of required sub leaser will be displayed.

3.2.32.4 Related Use Case: N/A

3.2.33 View "Need roommate" listing on the website.

3.2.33.1 Description: User can view "Need roommate" listing on the website.

3.2.33.2 **Actor(s):** User

3.2.33.3 Conditions:

3.2.33.4.1 Pre: User must be logged in. User must be on the "Need roommate" webpage.

3.2.33.4.2 Post: A list of required roommates will be displayed.

3.2.33.4 Related Use Case: N/A

3.2.34 Apply for career/employment options on the website.

3.2.34.1 Description: User can view the career/employment options on the website. They can post their resume for a position.

3.2.34.2 Actor(s): User

3.2.34.3 Conditions:

3.2.34.4.1 Pre: User must be logged in. User must be on the "career" webpage.

3.2.34.4.2 Post: A list of available career options should be displayed. The users can apply for any position online.

3.2.34.4 Related Use Case: N/A

3.3 Non-Functional Requirements

3.3.1 Performance: The product shall be based on web and must run from a web server. The product shall take initial load time depending on internet connection strength which also depends on the media (desktop/tab/mobile) from which the product is run. The performance shall depend upon hardware components of the client/customer.

3.3.2 Reliability: The information on the website will be fully creditable and up to date.

3.3.3 Availability: The website is available for all users for 24x7. There is zero downtime anticipated for the users.

3.3.4 Security: The website is secured with treating all user's data as confidential. We are not providing any additional security for the database. We are using the operating system's firewall for this website.

3.3.5 Maintainability: The website will be updated on regular basis (such as quarterly), depending on the needs and advancement in the real estate line of business.

3.3.6 Portability: The design of the website is highly portable and compatible with other technologies/systems as it is highly standardized website.

3.4 Design Constraints

The website is designed keeping in mind the GRASP principles but there is no dedicated plug-in available for creating design diagrams in Eclipse IDE. All design diagrams have been created using Visio.

3.5 Other Requirements

4. Design & Development

This website is created using AGILE methodology (SCRUM). The reason for choosing agile over waterfall model is that we are going for iterative and incremental approach rather than a big-bang approach. The approach is to build on a requirement and refine it in iterations. We are targeting to have a simultaneous development and testing for a requirement which will save time

and effort to find the defects and correct them in the code. This flexibility is very difficult to achieve in waterfall model; hence we have followed the agile methodology.

4.1 Software Process Model

4.1.1 Introduction

This section will provide an overview of the process model chosen and project timelines with respect to each sprint.

4.1.2.1 Plan Description

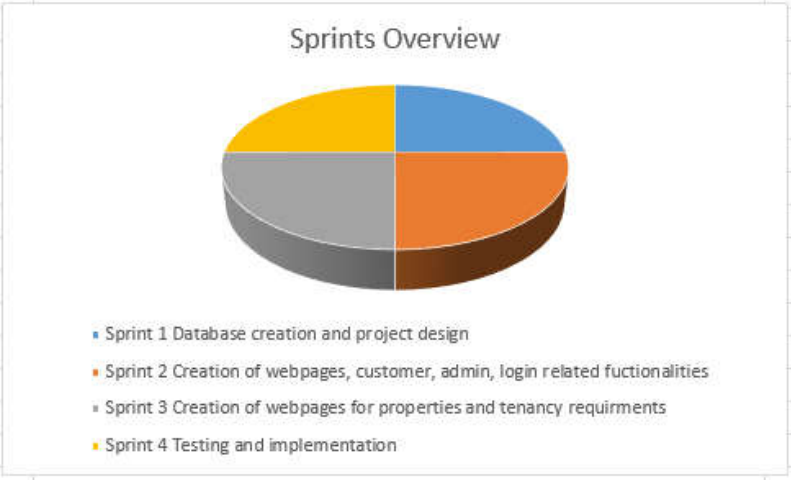
The process model we have chosen for the development of our application is the Iterative &Incremental Model (IIM). This is a model that will grow our application incrementally through iterations (Sprint 1, 2, 3 and 4) using various phases. We thought this would be a perfect fit for our application as we can divide each sprint and follow this cycle of planning, design, implementation, testing, and evaluation beginning of the start of each. Each iteration/sprint will last 2 weeks long.

We plan to meet on weekdays at 12noon for a daily standup meeting to discuss about the progress of work on the project. In addition to this, we are using a Kanban online tool (https://kanbantool.com/online-kanban-board) for keeping track on the user stories and the time elapsed. We will also be meeting on Tuesdays and Thursdays from 6 P.M. to 9 P.M. beginning at the start of the sprint, we will begin by planning and designing the specifications designated on the sprint schedule. Once that is done, we will begin implementing these new features and deploy them. We will then test these features near the end of the sprint and evaluate the system to conclude.

4.1.2.1 Plan Diagram

Below is a high-level project plan diagram:

Sprint Number	Tasks	Start Date	End Date
Sprint 1	Database creation and project design	2/19/2018	3/5/2018
Sprint 2	Creation of webpages, customer, admin, login related fuctionalities	3/5/2018	3/19/2018
Sprint 3	Creation of webpages for properties and tenancy requirments	3/19/2018	4/2/2018
Sprint 4	Testing and implementation	4/2/2018	4/16/2018



4.2 Overall Architecture

4.3 Sprint 1

4.3.1 Description

This sprint is dedicated to finalizing the design diagrams and class structure of the website. In addition to that creating of database is also included in this sprint. From the design perspective, we have worked on how the customer interacts with property class.

4.3.2 Satisfying Requirements

This sprint mainly covers the design for a customer and property class. We have tried to touch various requirement though not completed as yet. The following are the requirement numbers:

Requirement ID	Description
3.2.8	Update User Profile
3.2.9	Search Property through keywords (like zip code, state, property type)
3.2.10	View Detailed Posting
3.2.11	Schedule an Appointment
3.2.15	Subscribe to newsletter
3.2.16	Unsubscribe from newsletter
3.2.17	Update Property Information
3.2.30	Submit an application for tenancy by the user.
3.2.34	Apply for career/employment options on the website.

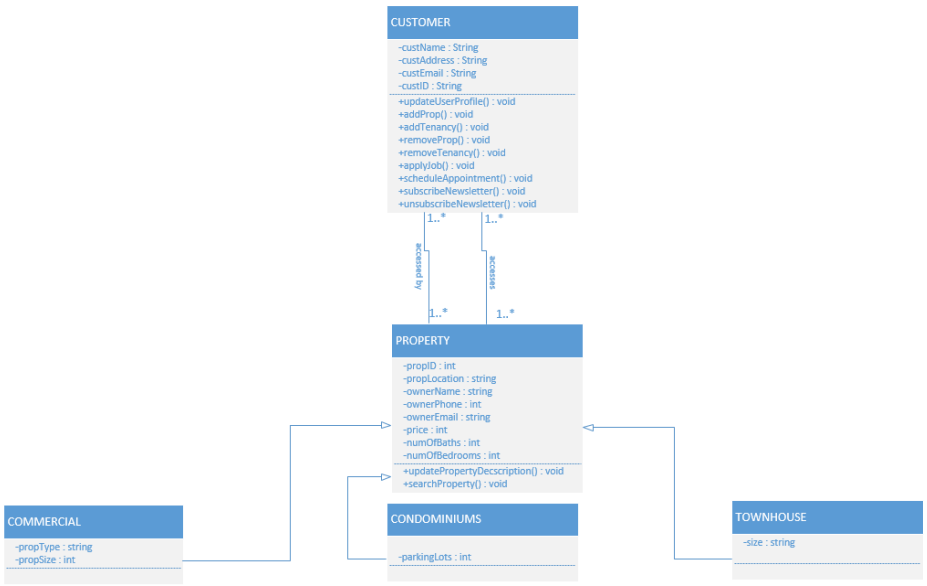
4.3.3 Design

4.3.3.1 Standards, Frameworks and Tools

We used Microsoft Visio Professional 2013 for creating class diagrams.

4.3.3.2 Class Diagram

Below is the class diagram which focuses on the relation between customer and property class:



The customer class can access the property class. One or more customers can access one or more properties. The customer can add a property, remove a property, search a property. Similarly, one or more properties can be accessed by one or more customers. The property class has three child classes namely, “commercial”, “condominiums” and “townhouse”.

4.3.4 Development

4.3.4.1 Description

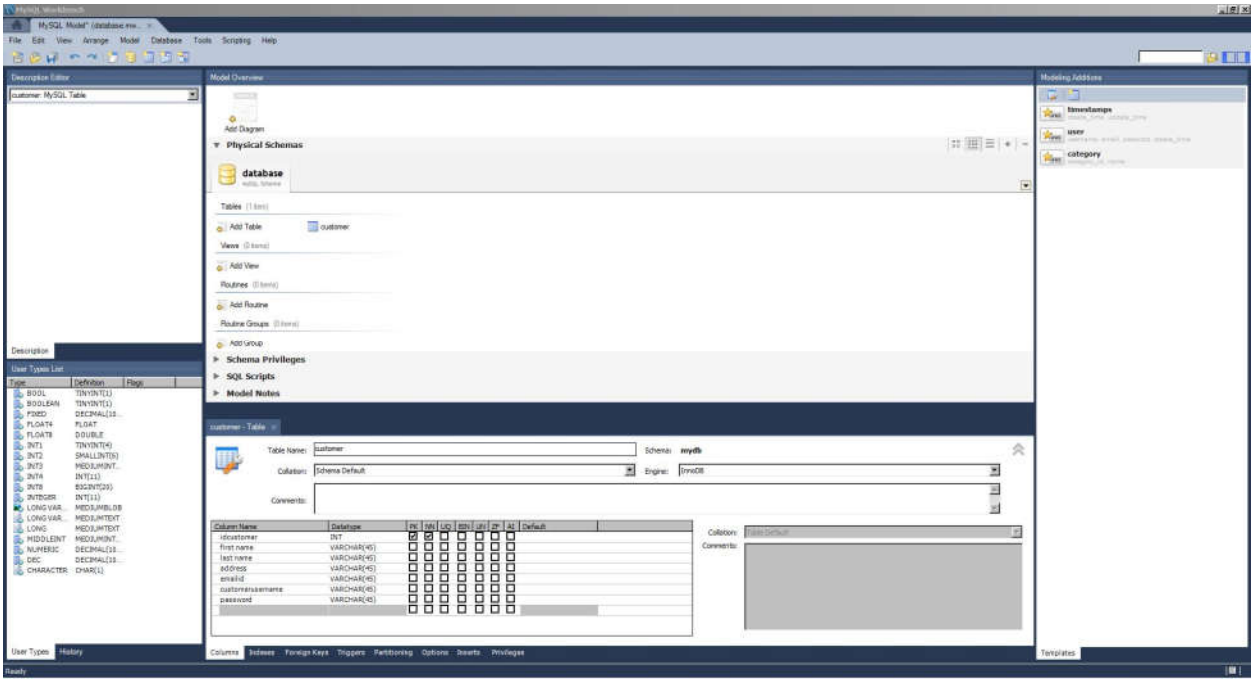
In this sprint, we are targeting to complete the database creation which includes the schema and various tables. We have tried to implement the normalization concepts which building up our database.

4.3.4.2 Standards, Frameworks and Tools

For the creating the database, we used MySQL Workbench 6.3 in order to create the schema and tables we needed.

4.3.4.3 Screenshots

Below is a screenshot from MySQL Workbench where we have created a “customer” table:



4.4 Sprint 2

4.4.1 Description

In this sprint we are targeting to complete the login, register/sign up a new user, verify a user and retrieve lost password requirements. We have tried to implement the login and sign-up forms using Servlets, MySQL and JSP. We have also used HTML and CSS for styling of the pages. So basically, the user would be able to successfully login to the website, can register if he is not having the login credentials and can retrieve the password in case the password is forgotten or lost.

4.4.2 Satisfying Requirements

We have tried to touch various requirement though not completed as yet (the styling and look enhancement of the pages is still in progress). The following are the requirement numbers:

Requirement ID	Description
3.2.1	Sign-in User
3.2.2	Sign up/Register user
3.2.22	Retrieve lost password
3.2.26	Confirm email registration (on sign up)

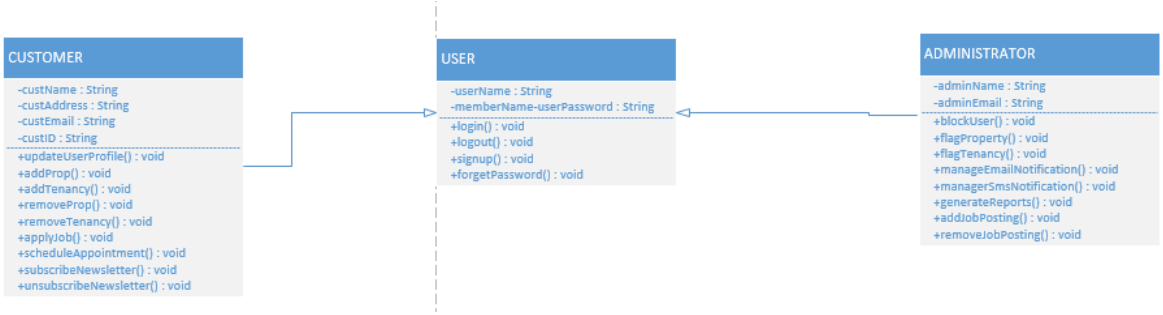
4.4.3 Design

4.4.3.1 Standards, Frameworks and Tools

We have used “lucid chart” tool for creating the activity diagram for this sprint. UML standards are followed in designing the features of this sprint.

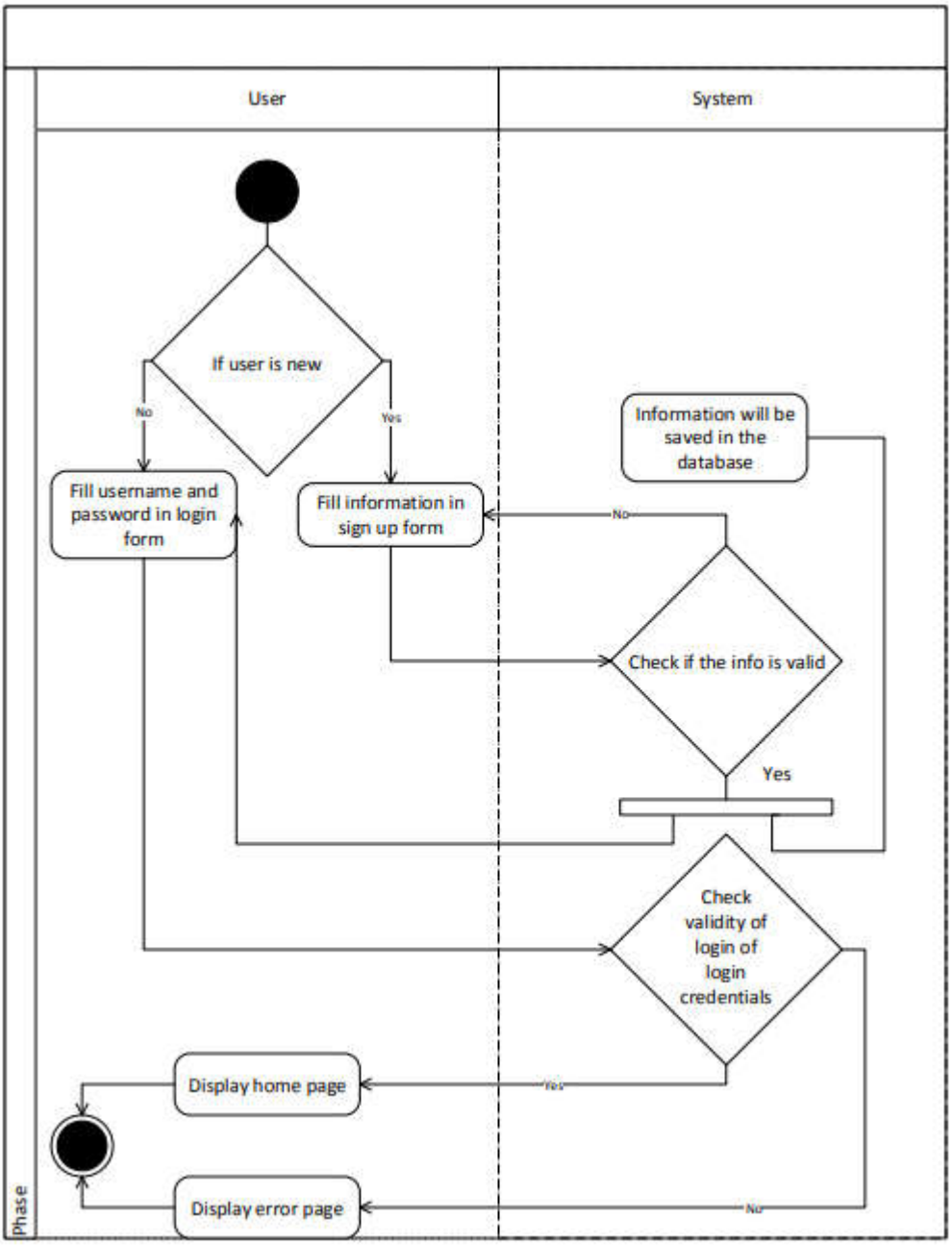
4.4.3.2 Class Diagram

Below if the class diagram which has the sprint-2 features:



We have mainly targetted the USERS class in this sprint which has features like login, logout, signup and forgetpassword.

4.4.3.3 Activity Diagram



4.4.4 Development

4.4.4.1 Description

We have created login and sign-up forms using Servlets, MySQL and JSP.

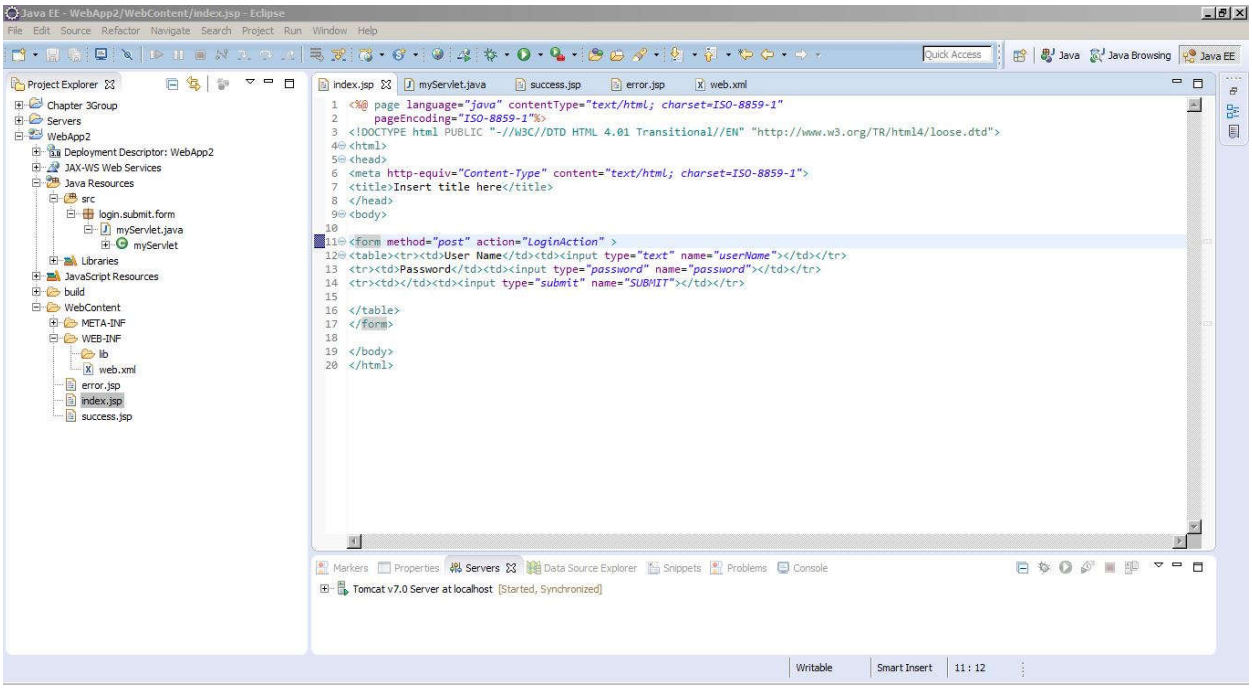
4.4.4.2 Standards, Frameworks and Tools

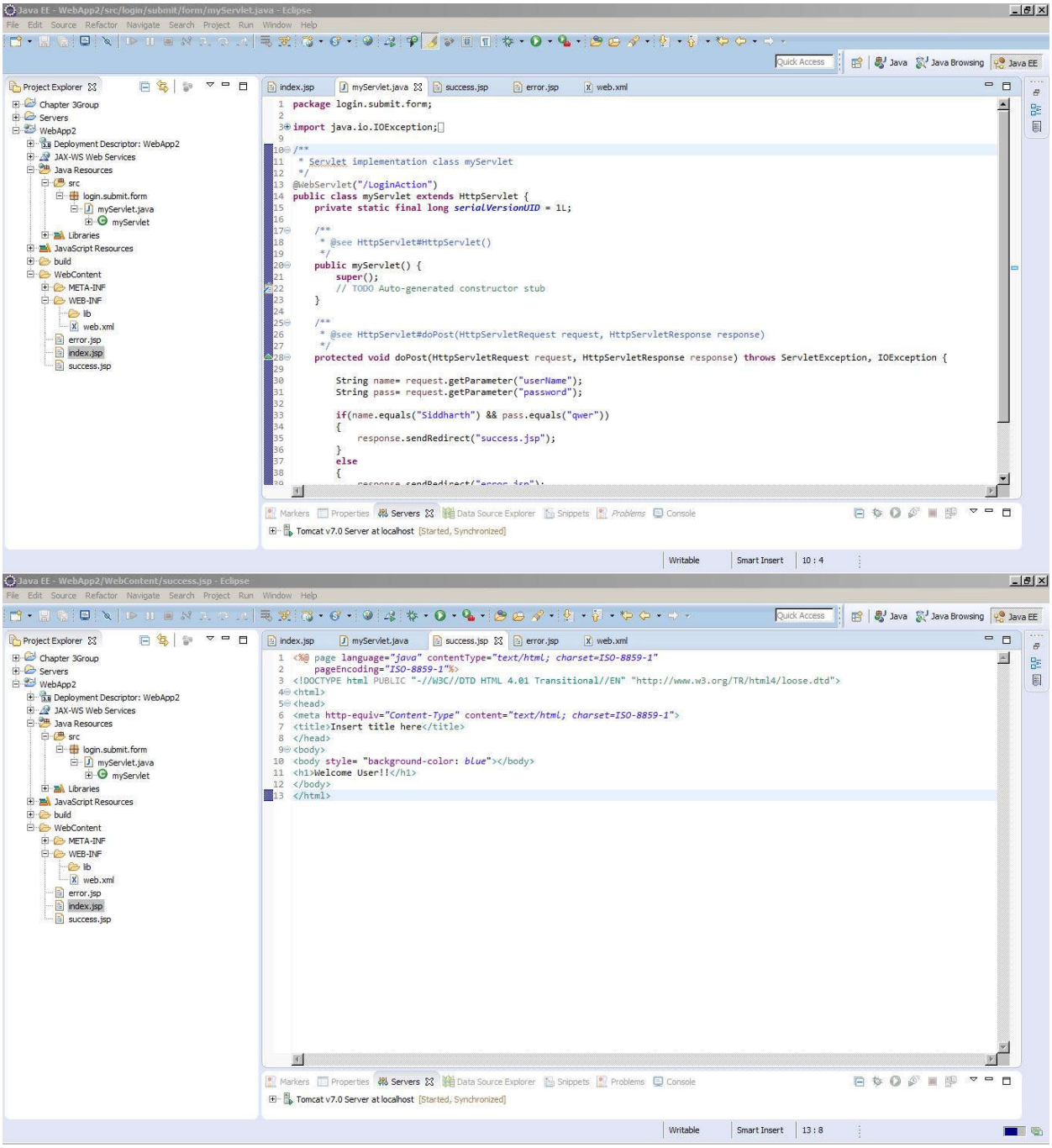
For developing the features, we have used the following:

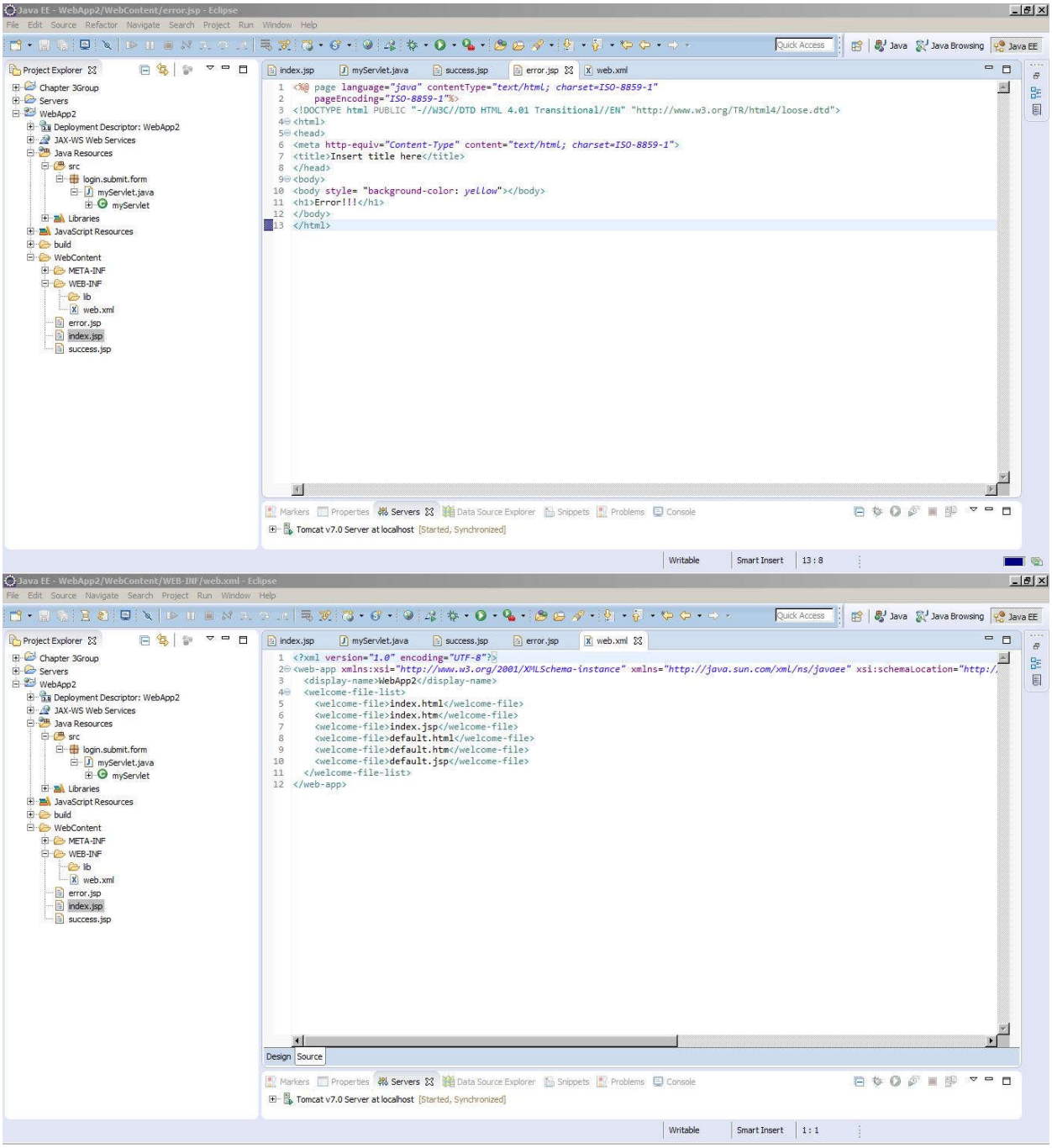
- Eclipse Java EE IDE for Web Developers: for creating java classes, jsp pages, servlets and other front-end files.
- Tomcat v7.0 server: Apache Tomcat is used to deploy your Java Servlets and JSPs
- MySQL Workbench: To create the schema and tables we needed

4.4.4.3 Screenshots

Below are the screenshots to show a "login functionality java code" (mainly JSP and servlet files):




Java EE - WebApp2/src/login/submit/form/myServlet.java - Eclipse
Project Explorer
index.jsp
myServlet.java
success.jsp
error.jsp
web.xml
package login.submit.form;
import java.io.IOException;
@WebServlet("/LoginAction")
public class myServlet extends HttpServlet {
private static final long serialVersionUID = 1L;
public myServlet() {
super();
// TODO Auto-generated constructor stub
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
String name= request.getParameter("userName");
String pass= request.getParameter("password");
if(name.equals("Siddharth") && pass.equals("qwer"))
response.sendRedirect("success.jsp");
else
Tomcat v7.0 Server at localhost [Started, Synchronized]
Writable
Smart Insert
10 : 4
Java EE - WebApp2/WebContent/success.jsp - Eclipse
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<body style= "background-color: blue"></body>
<h1>Welcome User!!</h1>
</body>
</html>
Writable
Smart Insert
13 : 8



4.5 Sprint 3

4.5.1 Description

In this sprint we are targeting to complete filter functionality and to design the User Interface and styling of the webpages.

4.5.2 Satisfying Requirements

We have tried to touch various requirement though not completed as yet (the styling and look enhancement of the pages is still in progress). The following are the requirement numbers:

Requirement ID	Description
3.2.29	Filter and sort by functionality
3.2.9	Search Property through keywords (like zip code, state, property type)

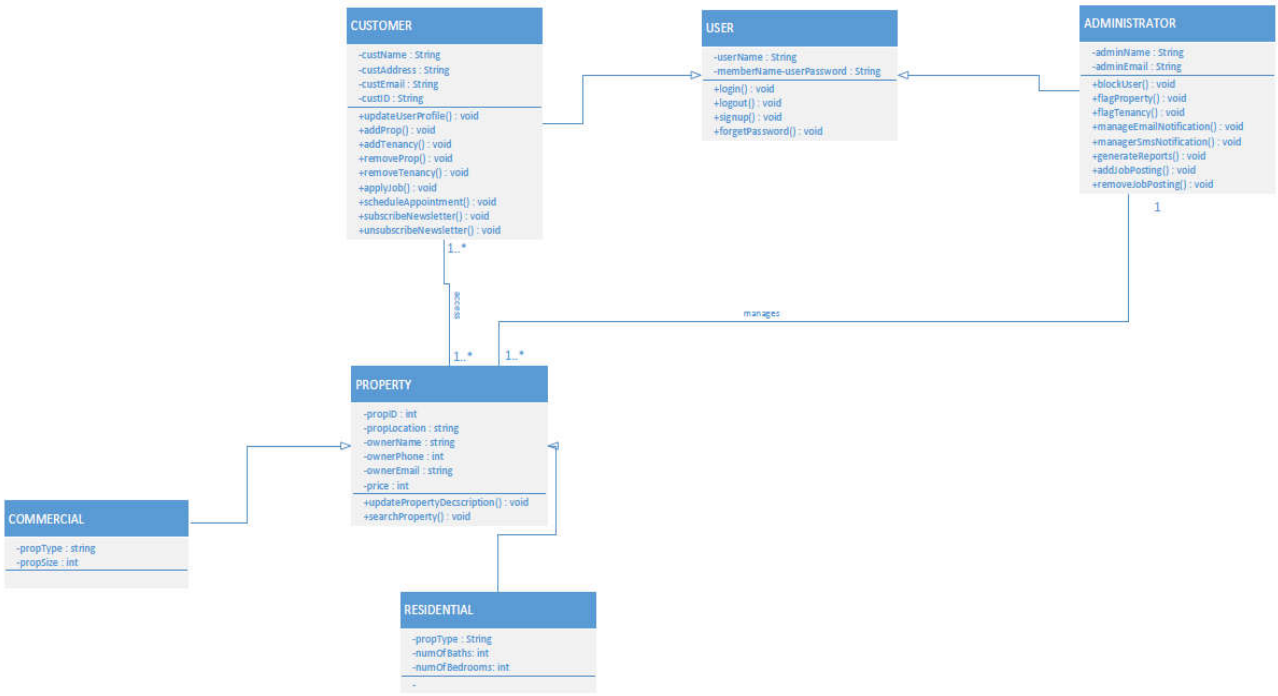
4.5.3 Design

4.5.3.1 Standards, Frameworks and Tools

We have used "lucid chart" tool for creating the activity diagram for this sprint. UML standards are followed in designing the features of this sprint.

4.5.3.2 Class Diagram

Below is the class diagram of the requirements that we have covered in this sprint:



We have targetted the search functionality here. (searchProperty(): void)

4.5.4 Development

4.3.4.1 Description

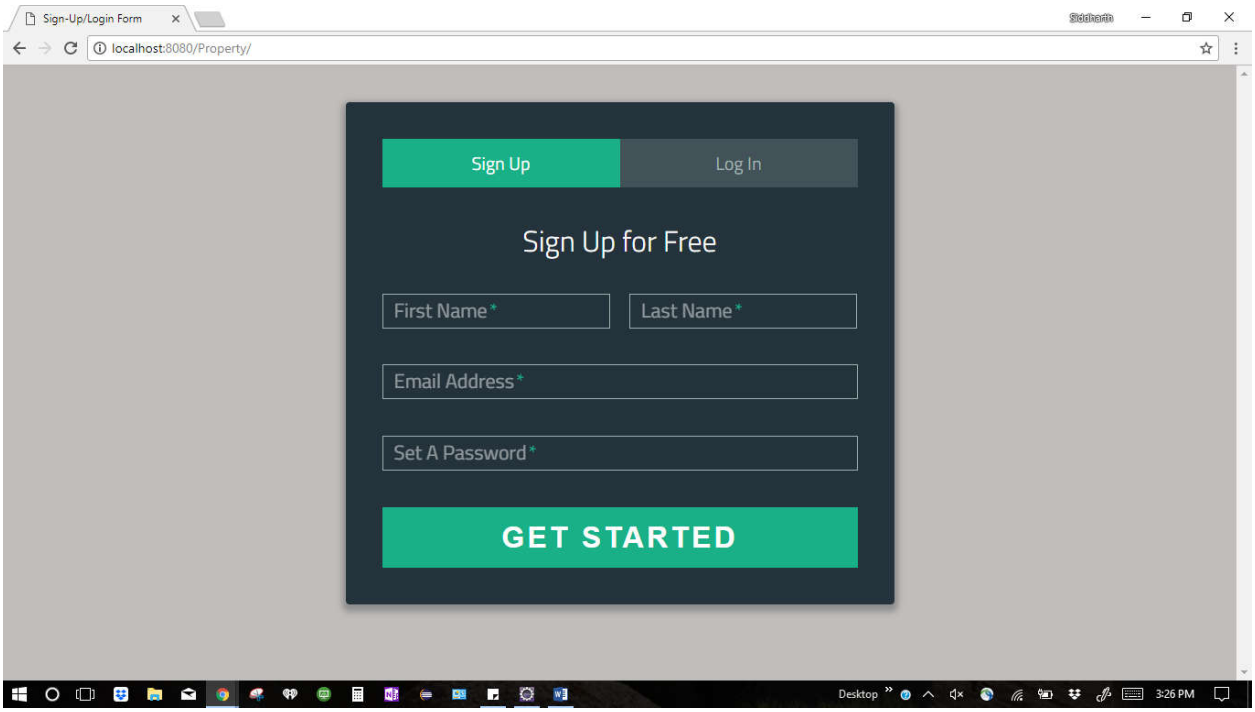
We have created login, sign-up forms and homepage using bootstrap (HTML and CSS styling), Servlets, MySQL and JSP.

4.3.4.2 Standards, Frameworks and Tools

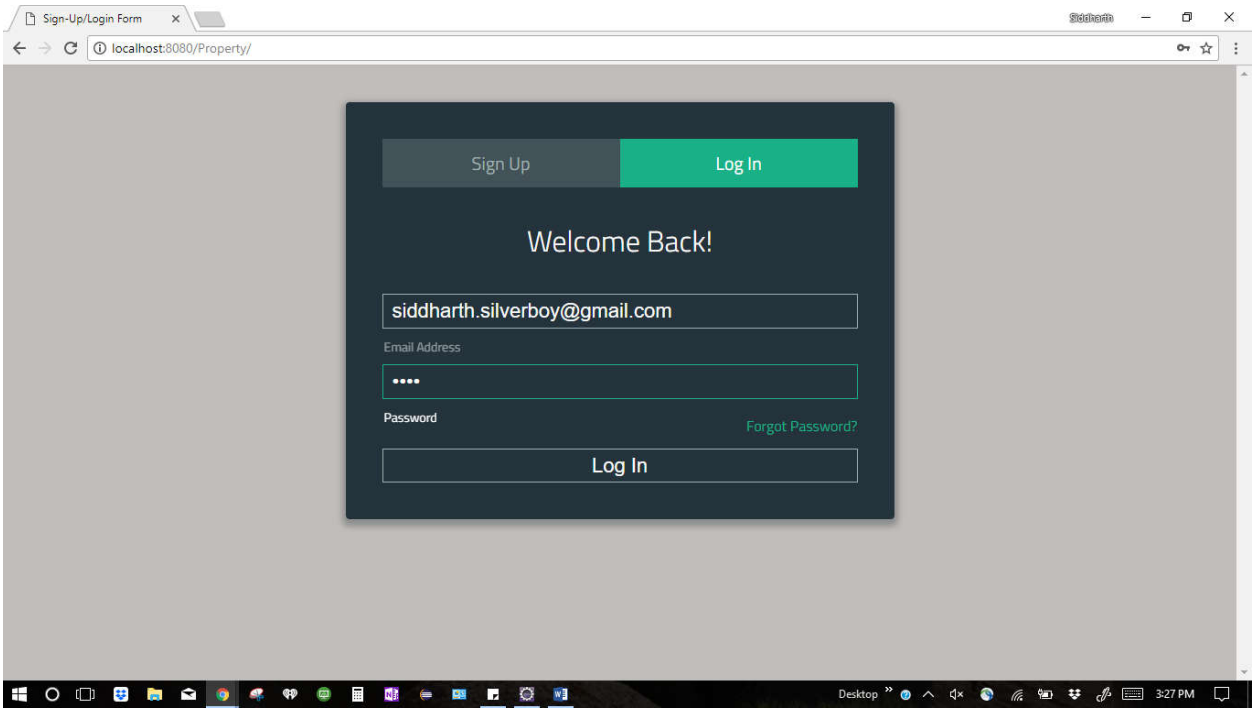
For developing the features, we have used the following:
o Eclipse Java EE IDE for Web Developers: for creating java classes, jsp pages, servlets and other front-end files.
o Tomcat v8.5 server: Apache Tomcat is used to deploy your Java Servlets and JSPs
o MySQL Workbench: To create the schema and tables we needed

4.3.4.3 Screenshots

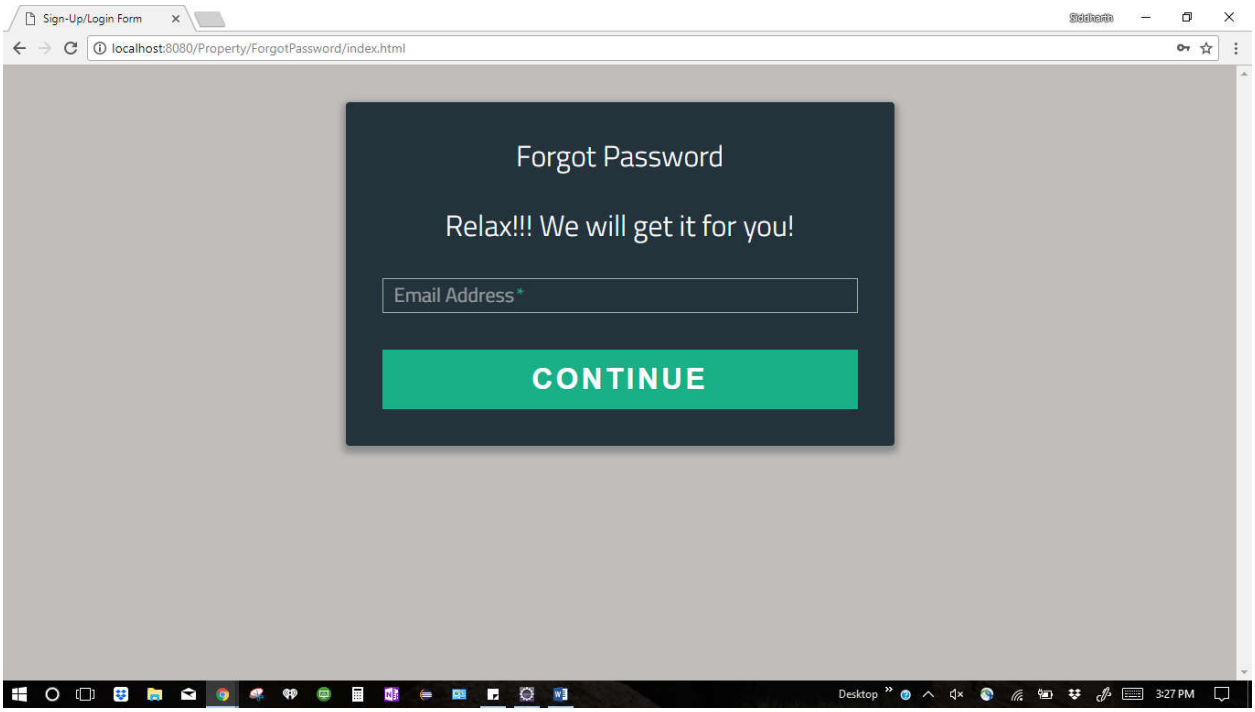
Sign-up Page:



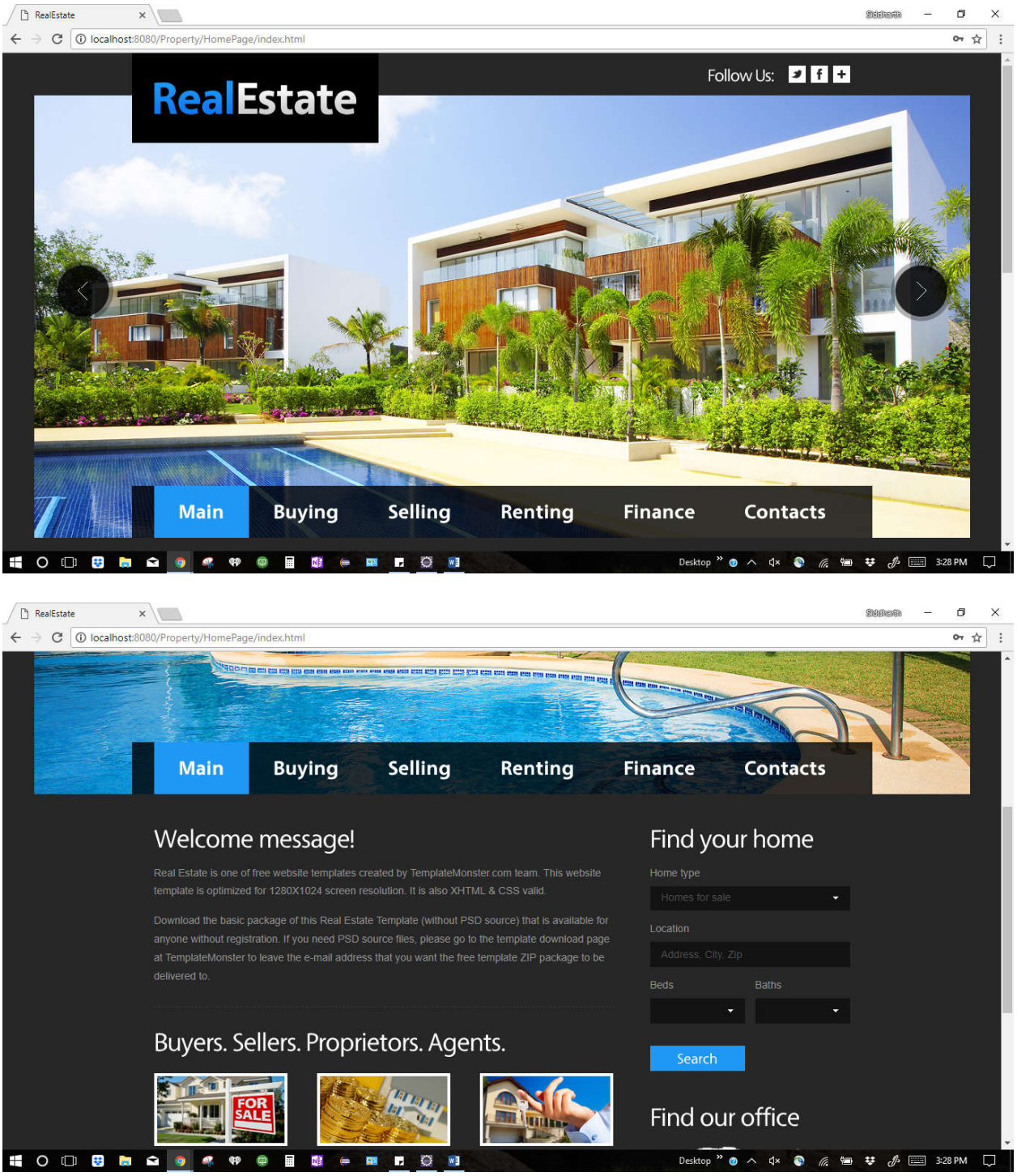
Login Page:



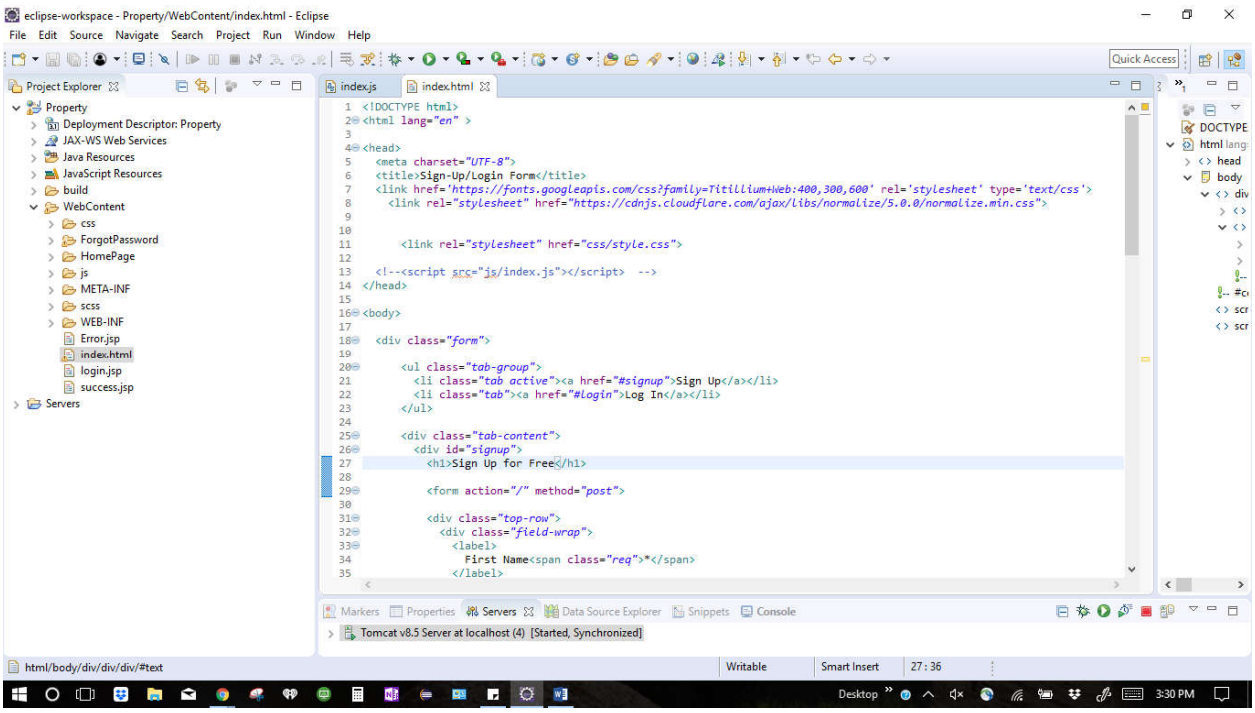
Forget Password:



Home Page:


RealEstate
localhost:8080/Property/HomePage/index.html
RealEstate
Follow Us:
Main
Buying
Selling
Renting
Finance
Contacts
Welcome message!
Real Estate is one of free website templates created by TemplateMonster.com team. This website template is optimized for 1280X1024 screen resolution. It is also XHTML & CSS valid.
Download the basic package of this Real Estate Template (without PSD source) that is available for anyone without registration. If you need PSD source files, please go to the template download page at TemplateMonster to leave the e-mail address that you want the free template ZIP package to be delivered to.
Buyers. Sellers. Proprietors. Agents.
FOR SALE
Find your home
Home type
Homes for sale
Location
Address, City, Zip
Beds
Baths
Search
Find our office

Index.html



4.6 Sprint 4

4.6.1 Description

This subsection of the SRS document provides brief description of this sprint, its activities and achievements.

4.6.2 Satisfying Requirements

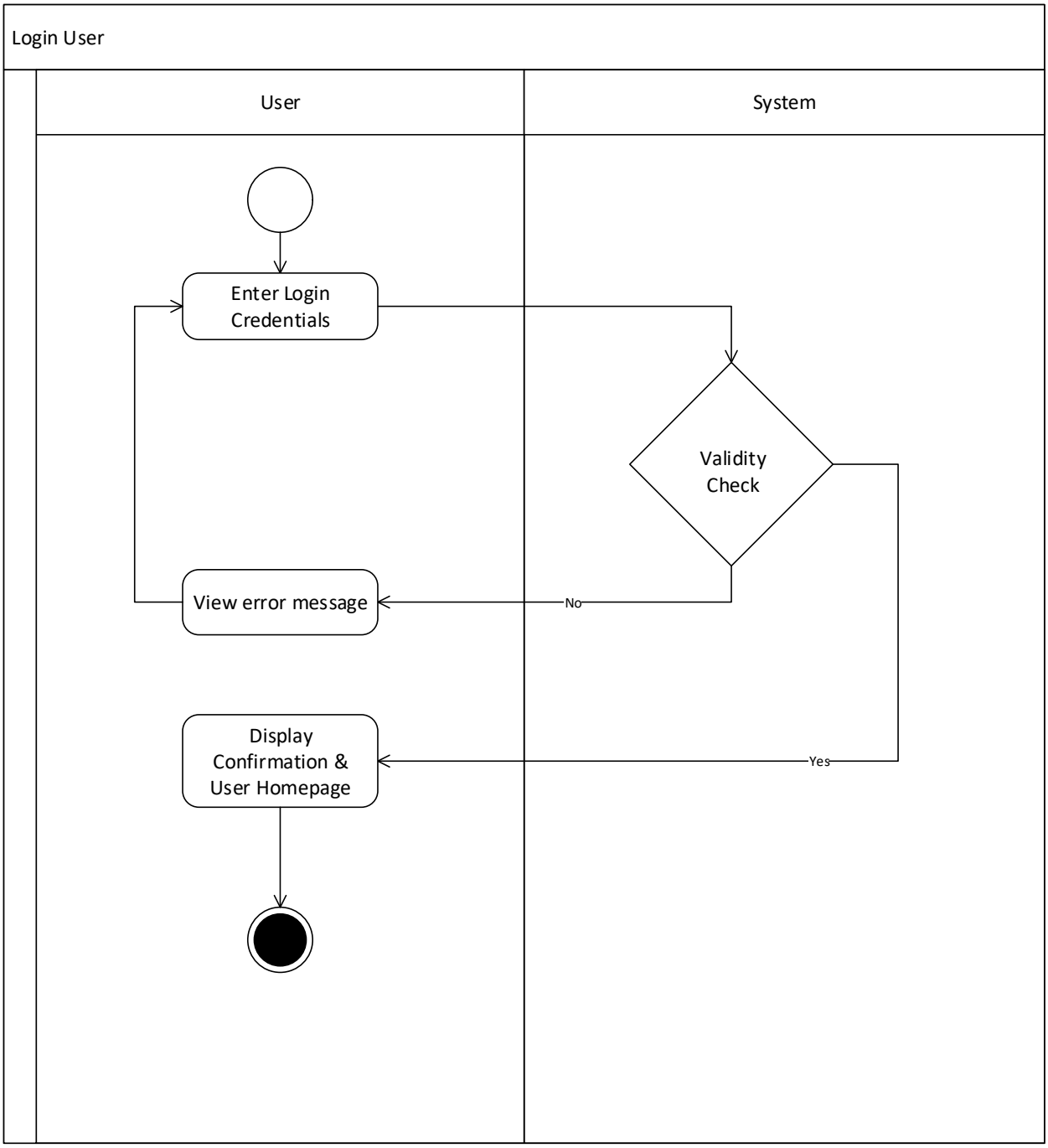
Specify the requirements, (i.e. the requirement number) that are satisfied in this sprint. The requirement number must match the one in Section 3.2.

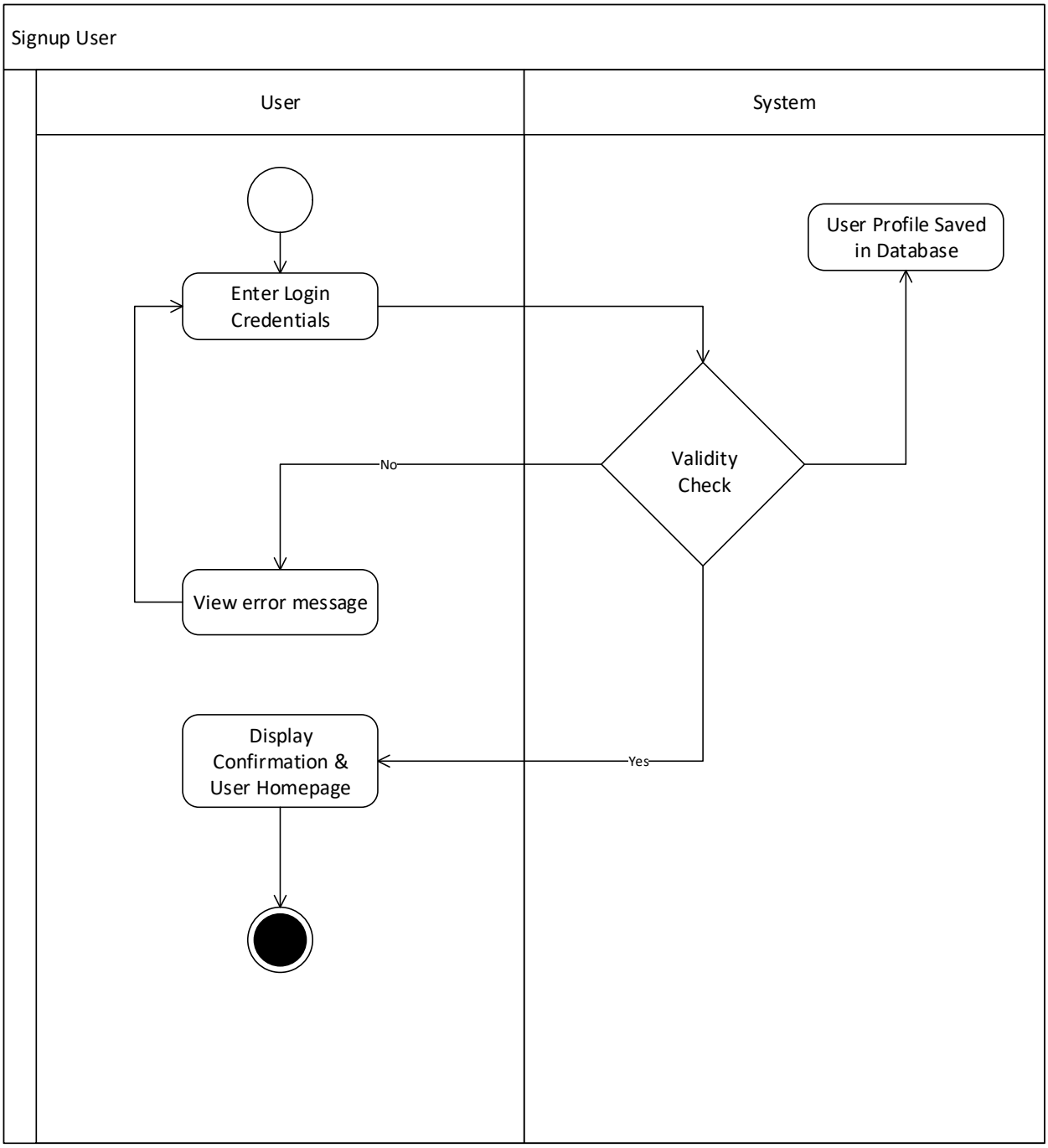
4.6.3 Design

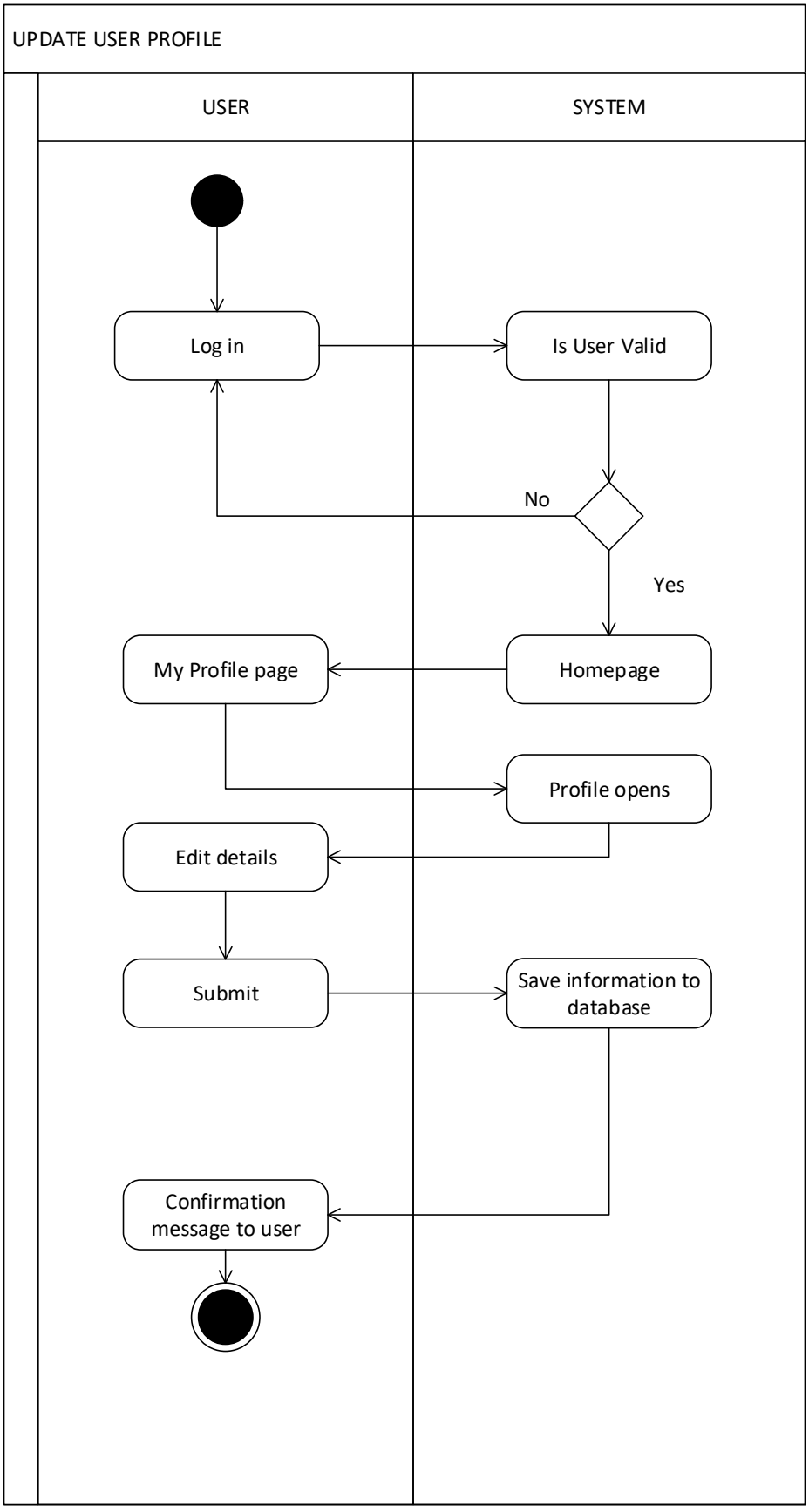
4.6.3.1 Standards, Frameworks and Tools

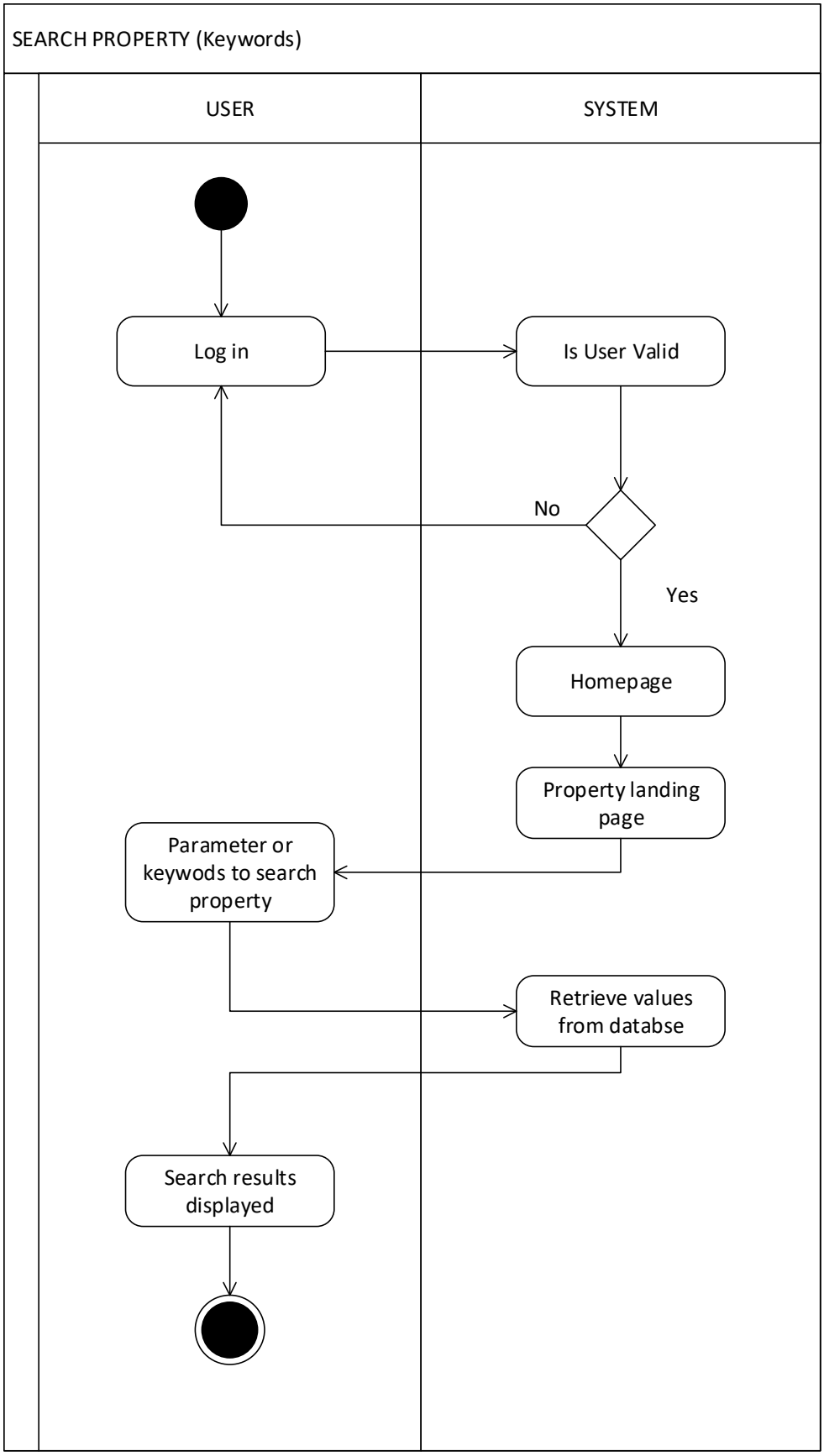
4.6.3.2 Class Diagram

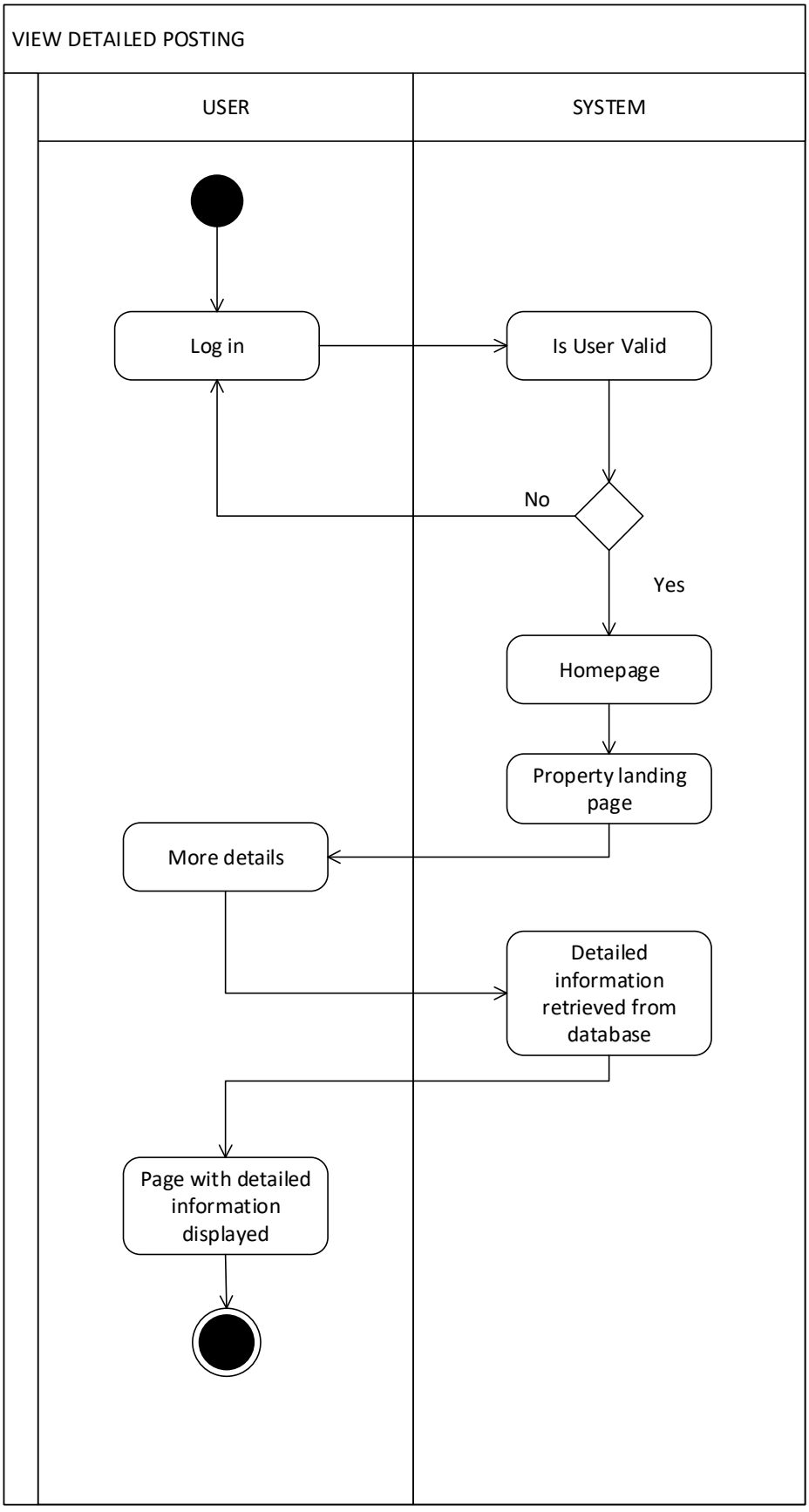
4.6.3.3 Activity Diagram

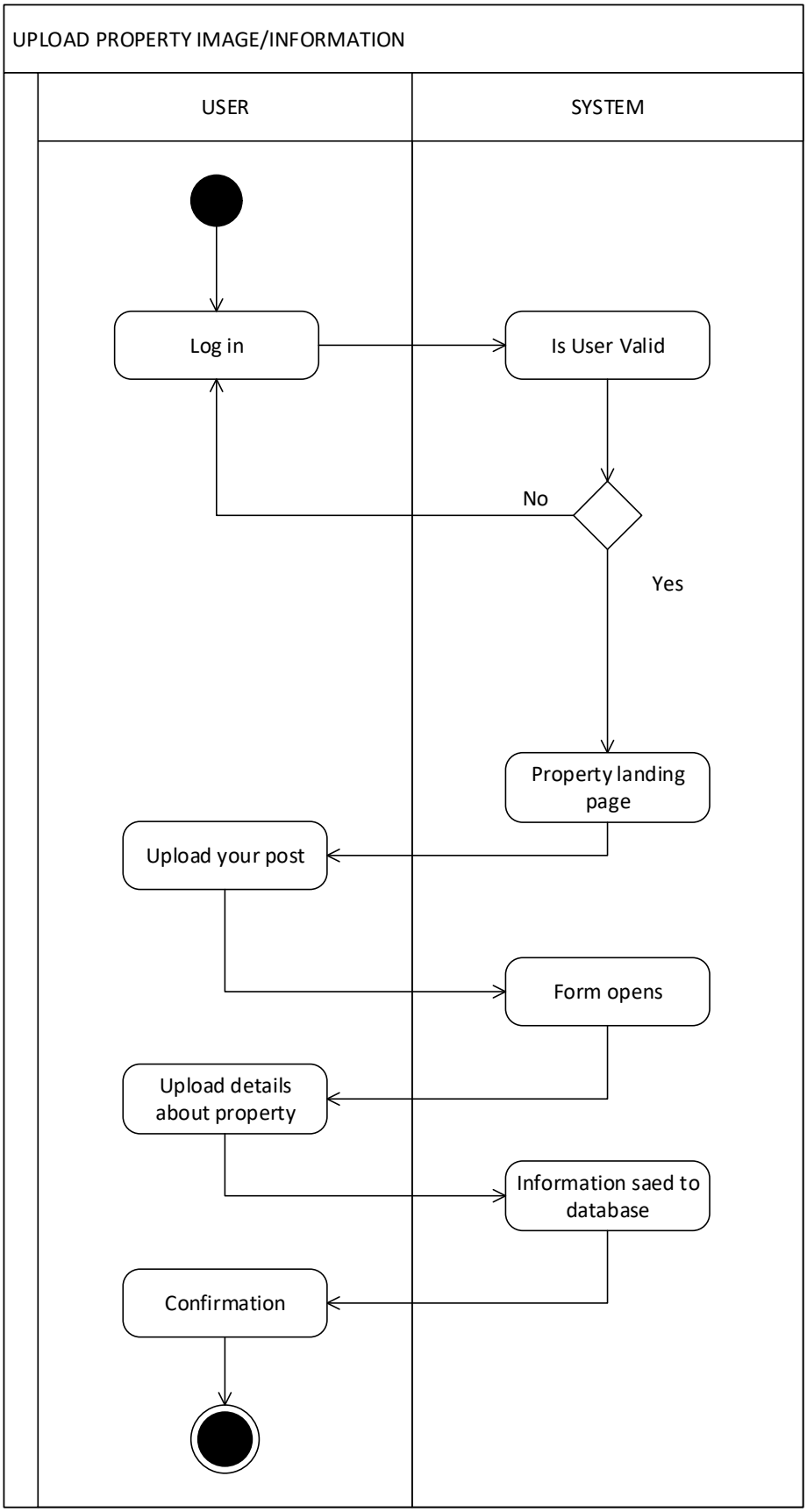

Login User
User
System
Enter Login Credentials
Validity Check
View error message
No
Display Confirmation & User Homepage
Yes

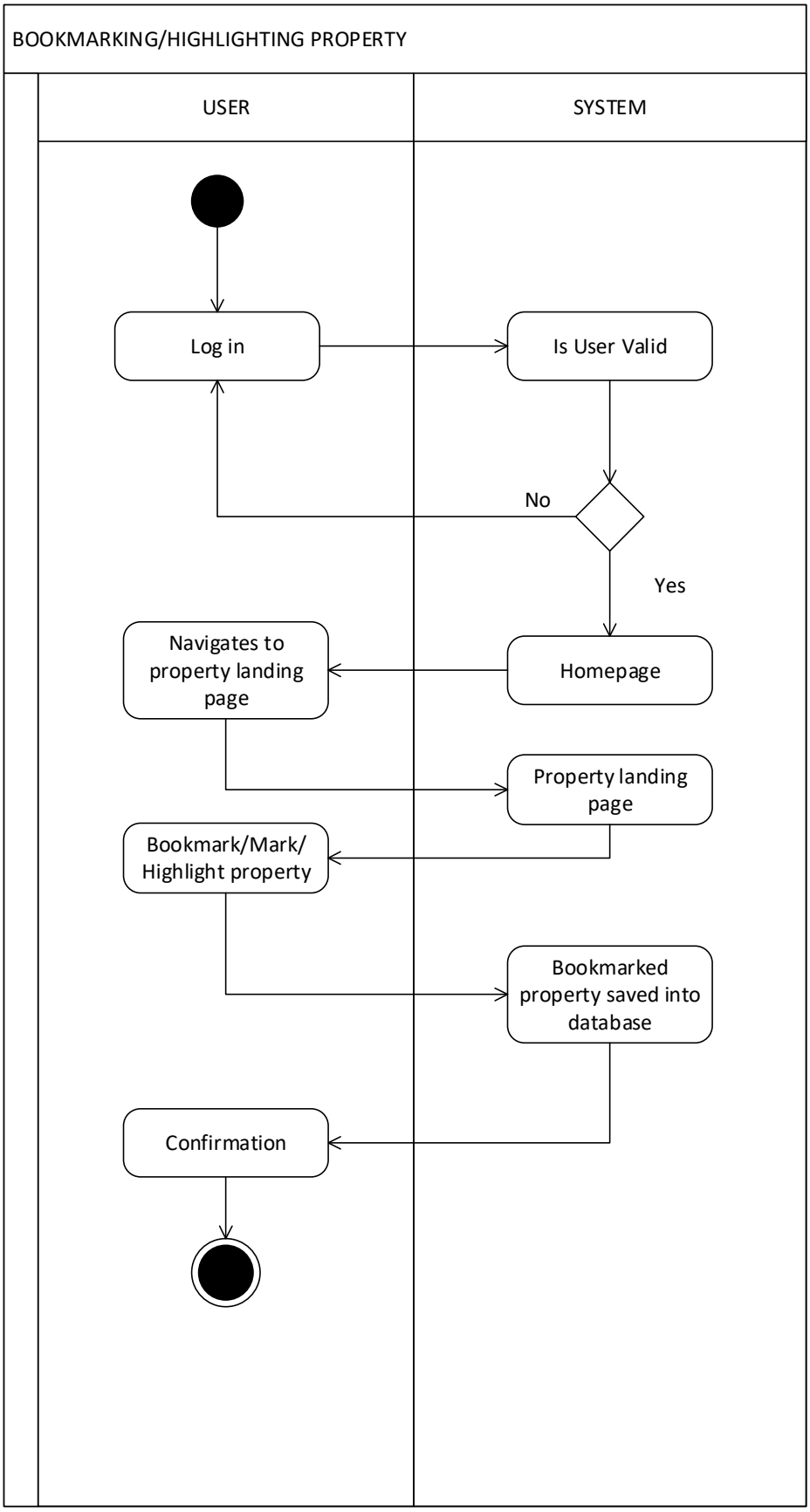

Signup User
User
System
Enter Login Credentials
User Profile Saved in Database
Validity Check
No
View error message
Display Confirmation & User Homepage
Yes

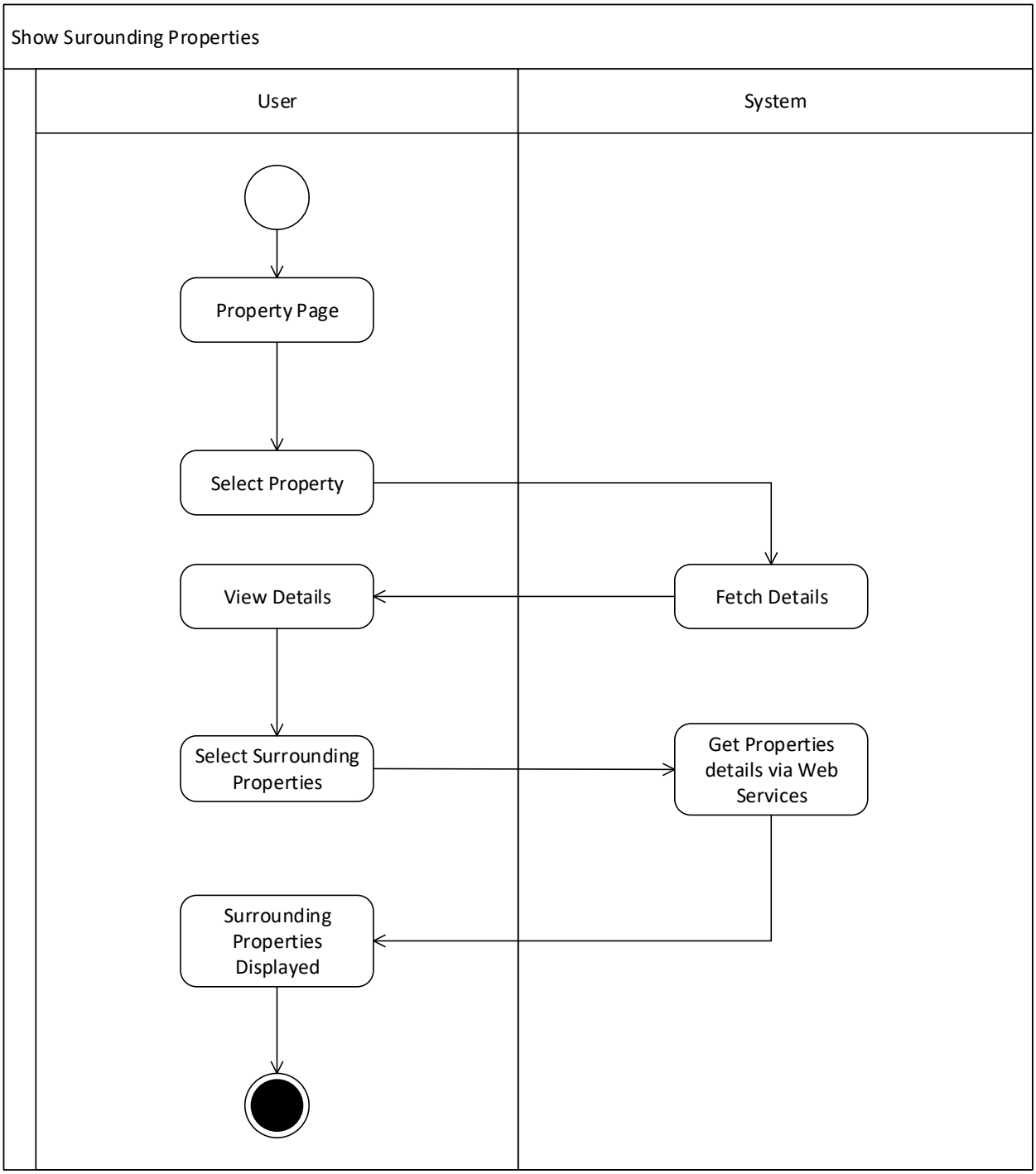

UPDATE USER PROFILE
USER
SYSTEM
Log in
Is User Valid
No
Yes
My Profile page
Homepage
Profile opens
Edit details
Submit
Save information to database
Confirmation message to user

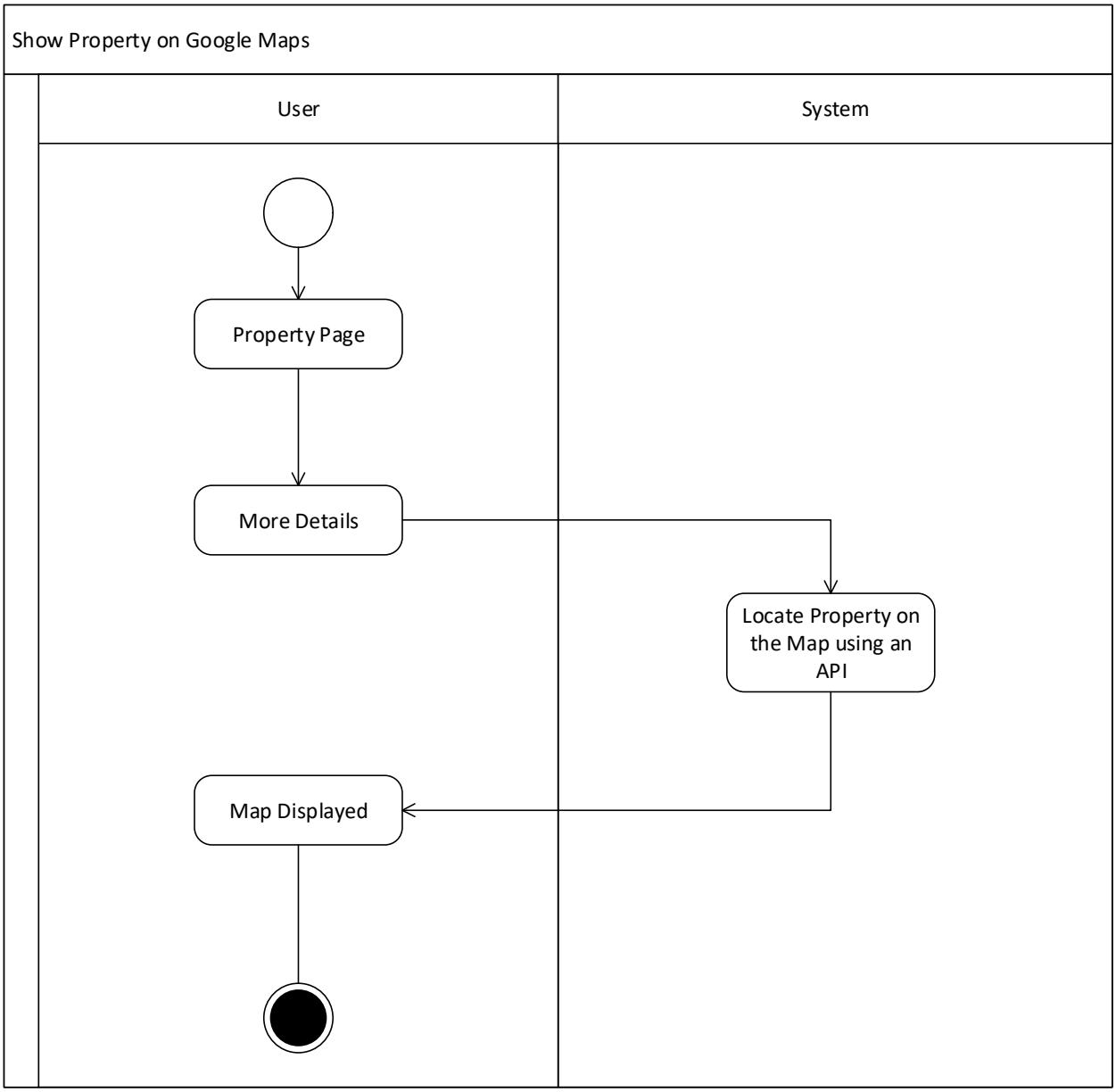

SEARCH PROPERTY (Keywords)
USER
SYSTEM
Log in
Is User Valid
No
Yes
Homepage
Property landing page
Parameter or keywods to search property
Retrieve values from databse
Search results displayed

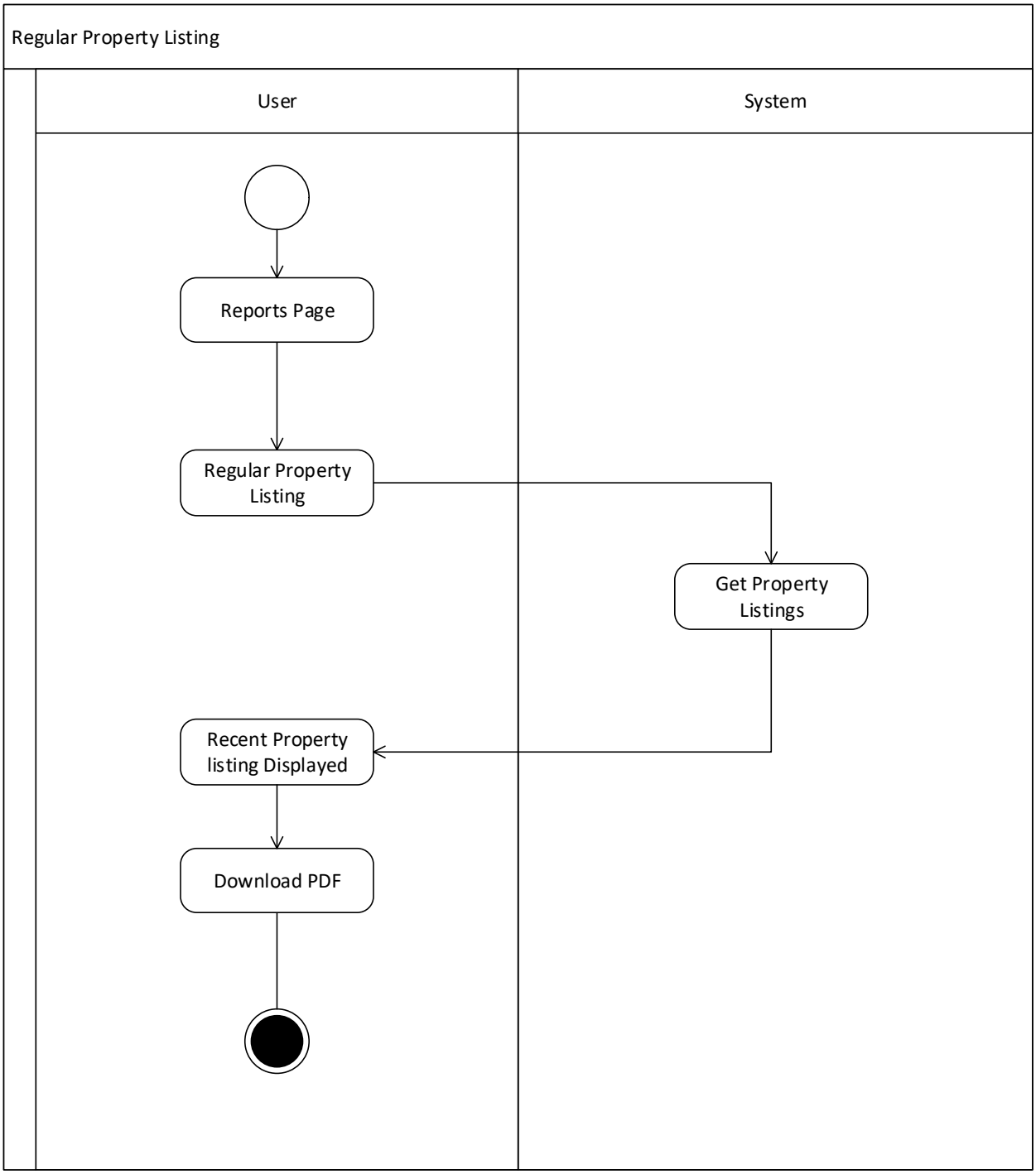

VIEW DETAILED POSTING
USER
SYSTEM
Log in
Is User Valid
No
Yes
Homepage
Property landing page
More details
Detailed information retrieved from database
Page with detailed information displayed

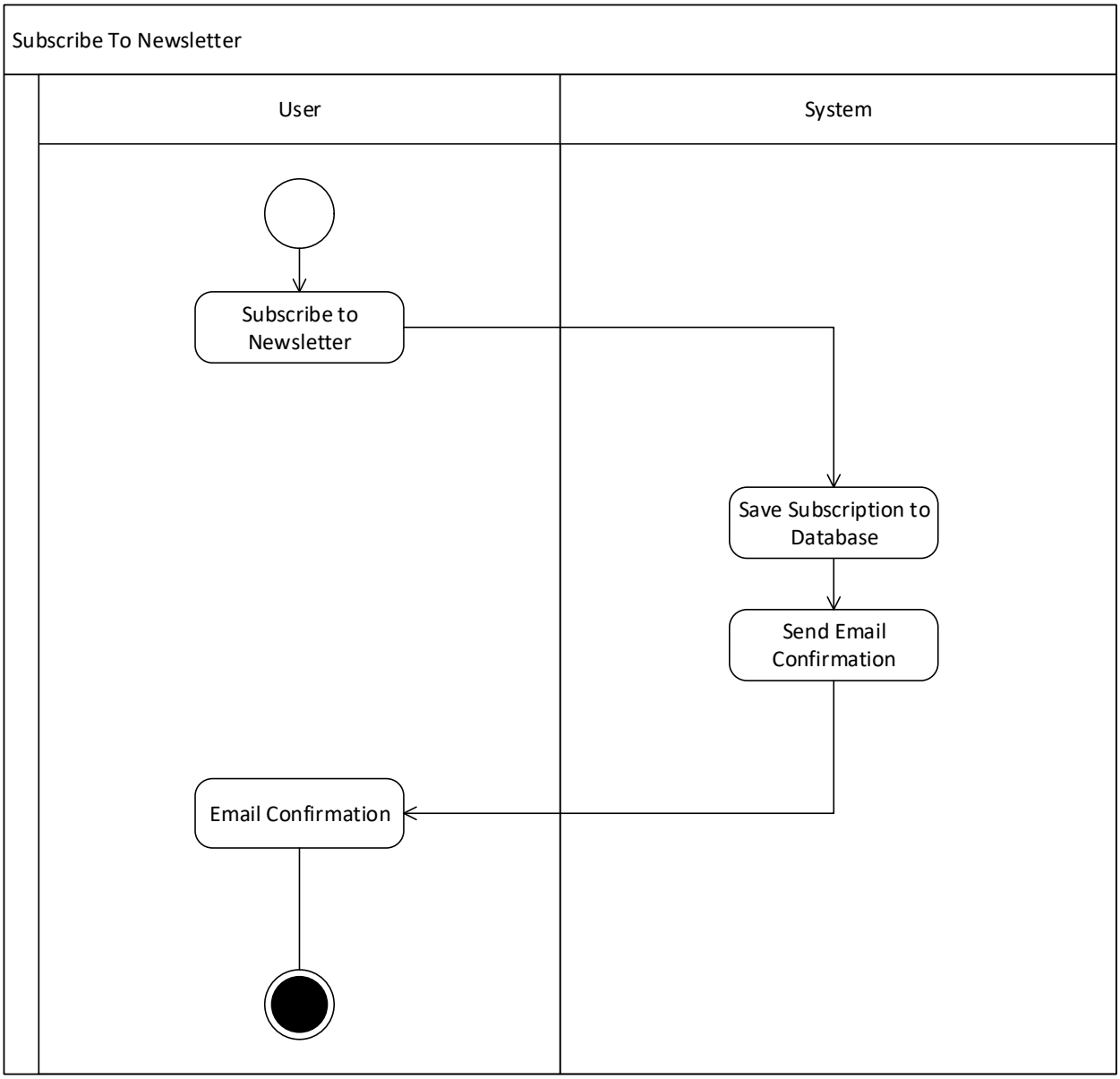

UPLOAD PROPERTY IMAGE/INFORMATION
USER
SYSTEM
Log in
Is User Valid
No
Yes
Property landing page
Upload your post
Form opens
Upload details about property
Information saed to database
Confirmation

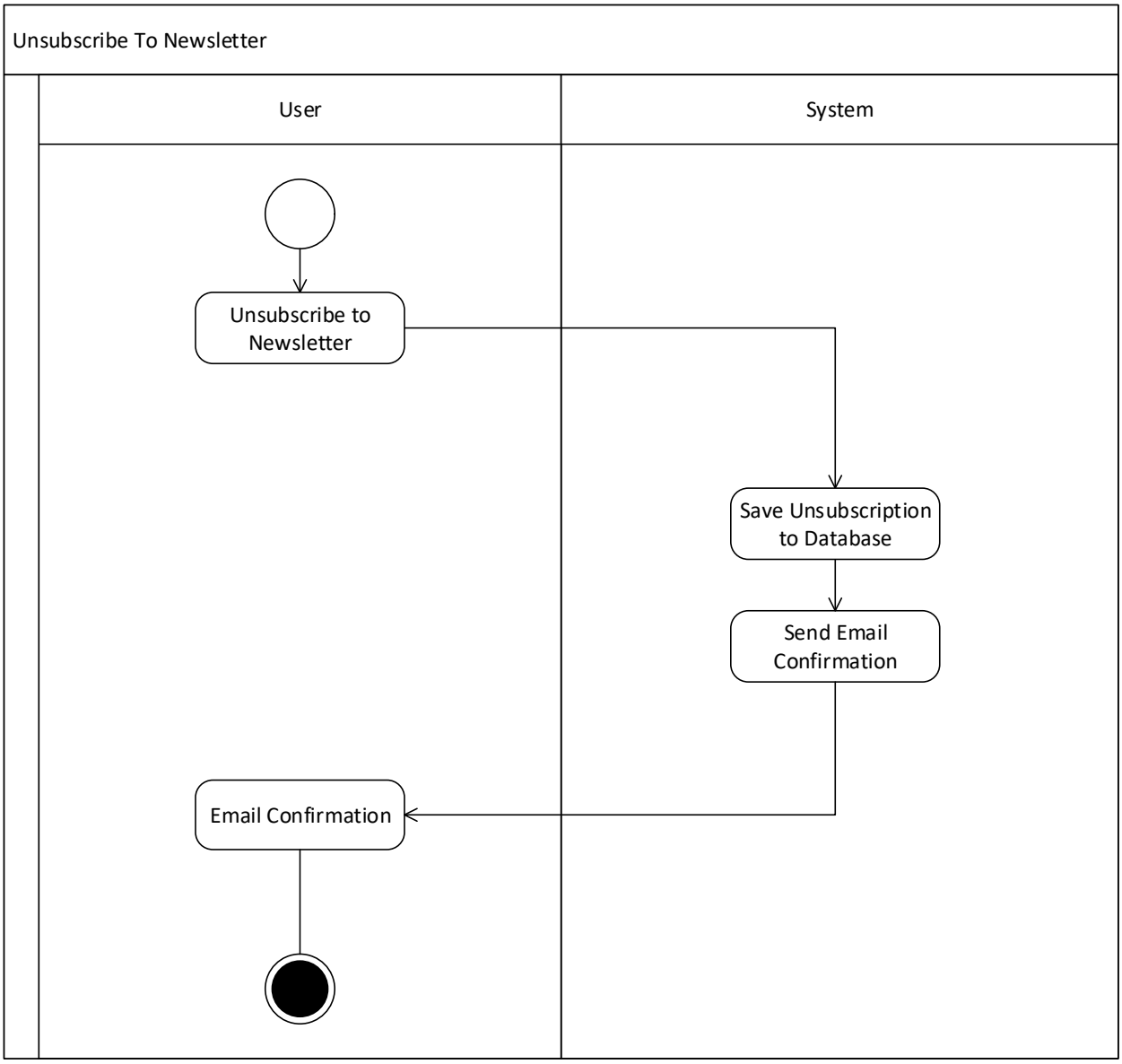

BOOKMARKING/HIGHLIGHTING PROPERTY
USER
SYSTEM
Log in
Is User Valid
No
Yes
Navigates to property landing page
Homepage
Property landing page
Bookmark/Mark/ Highlight property
Bookmarked property saved into database
Confirmation

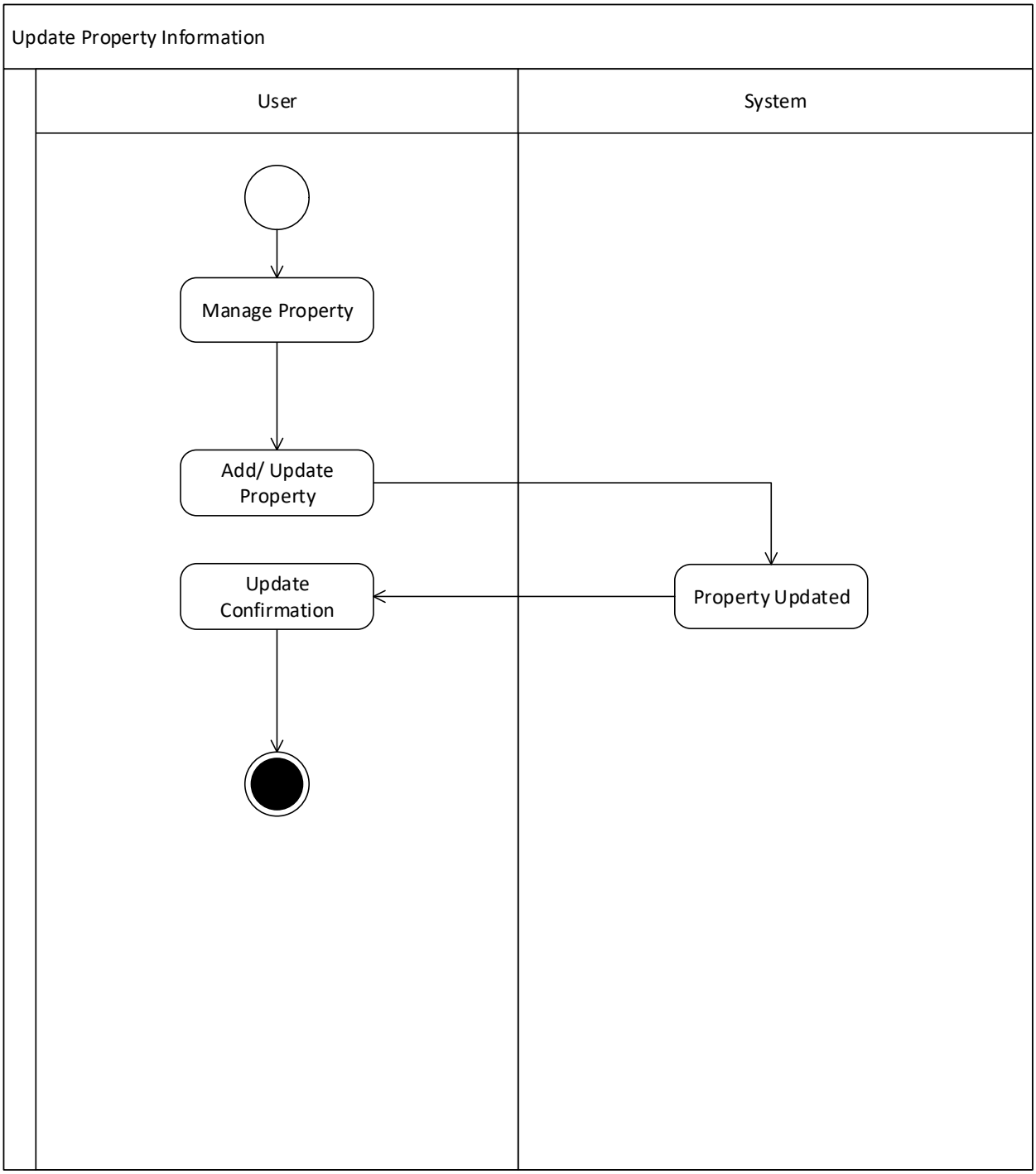

Show Surounding Properties
User
System
Property Page
Select Property
View Details
Fetch Details
Select Surrounding Properties
Get Properties details via Web Services
Surrounding Properties Displayed

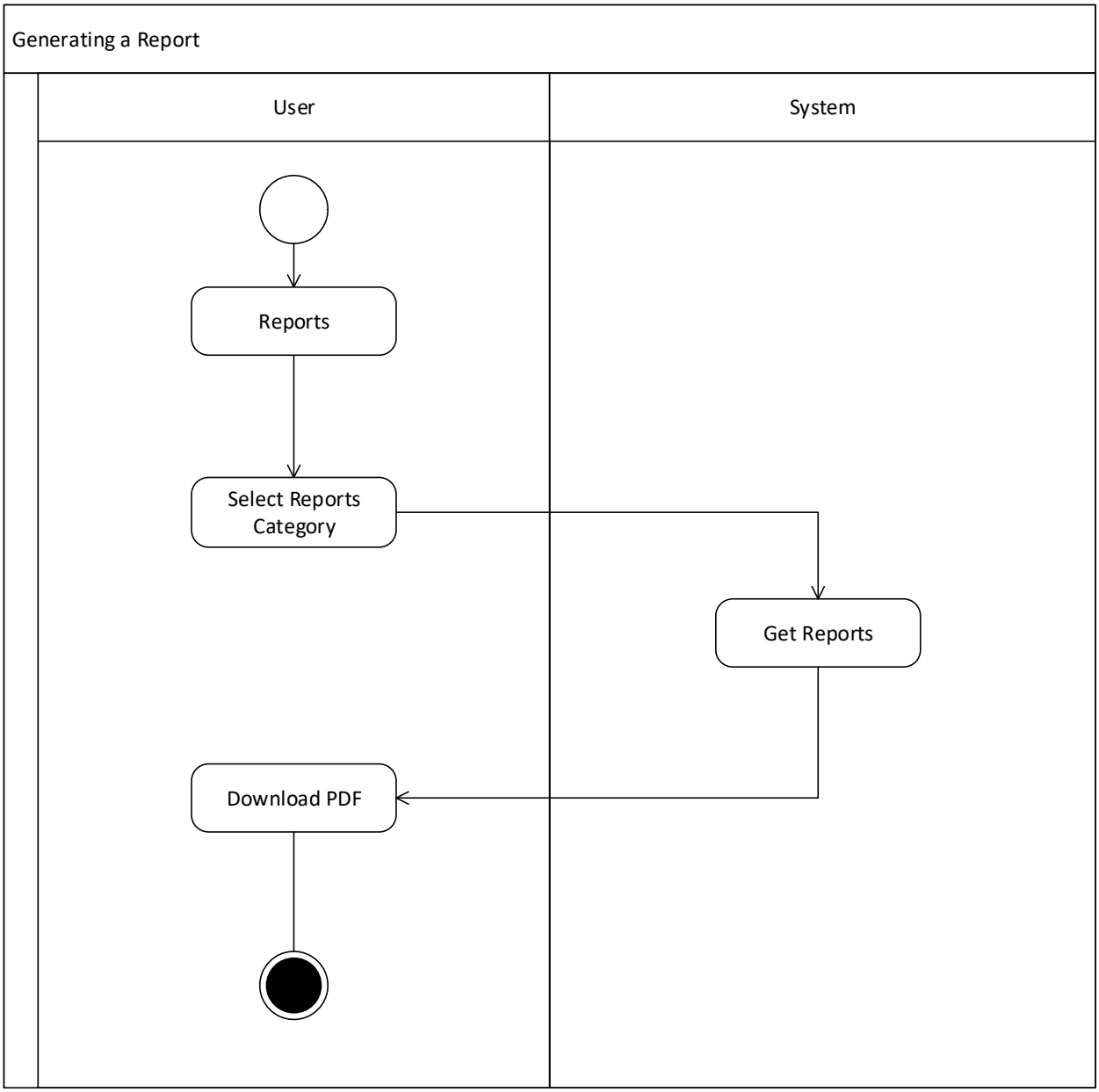

Show Property on Google Maps
User
System
Property Page
More Details
Locate Property on the Map using an API
Map Displayed

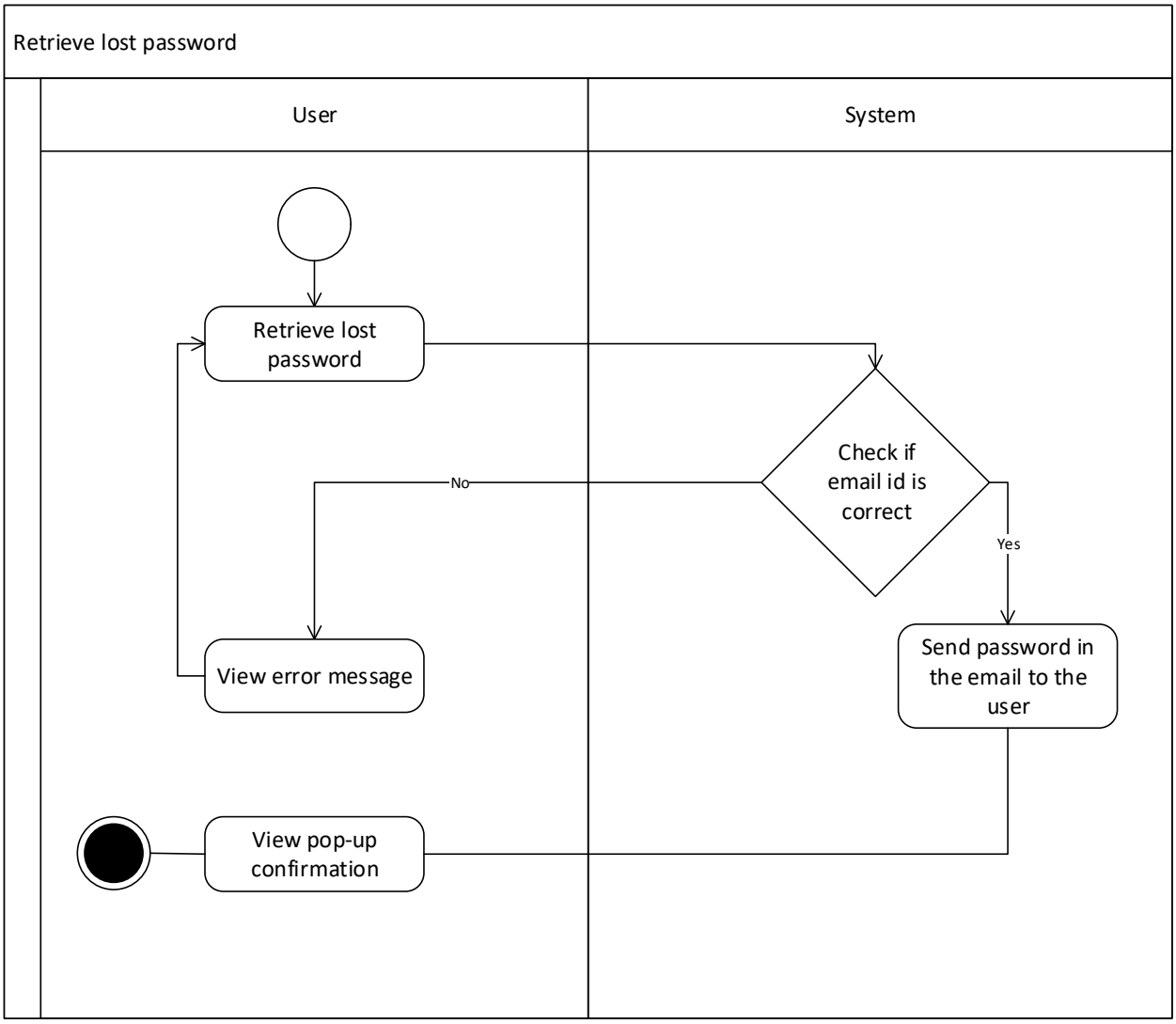

Regular Property Listing
User
System
Reports Page
Regular Property Listing
Get Property Listings
Recent Property listing Displayed
Download PDF

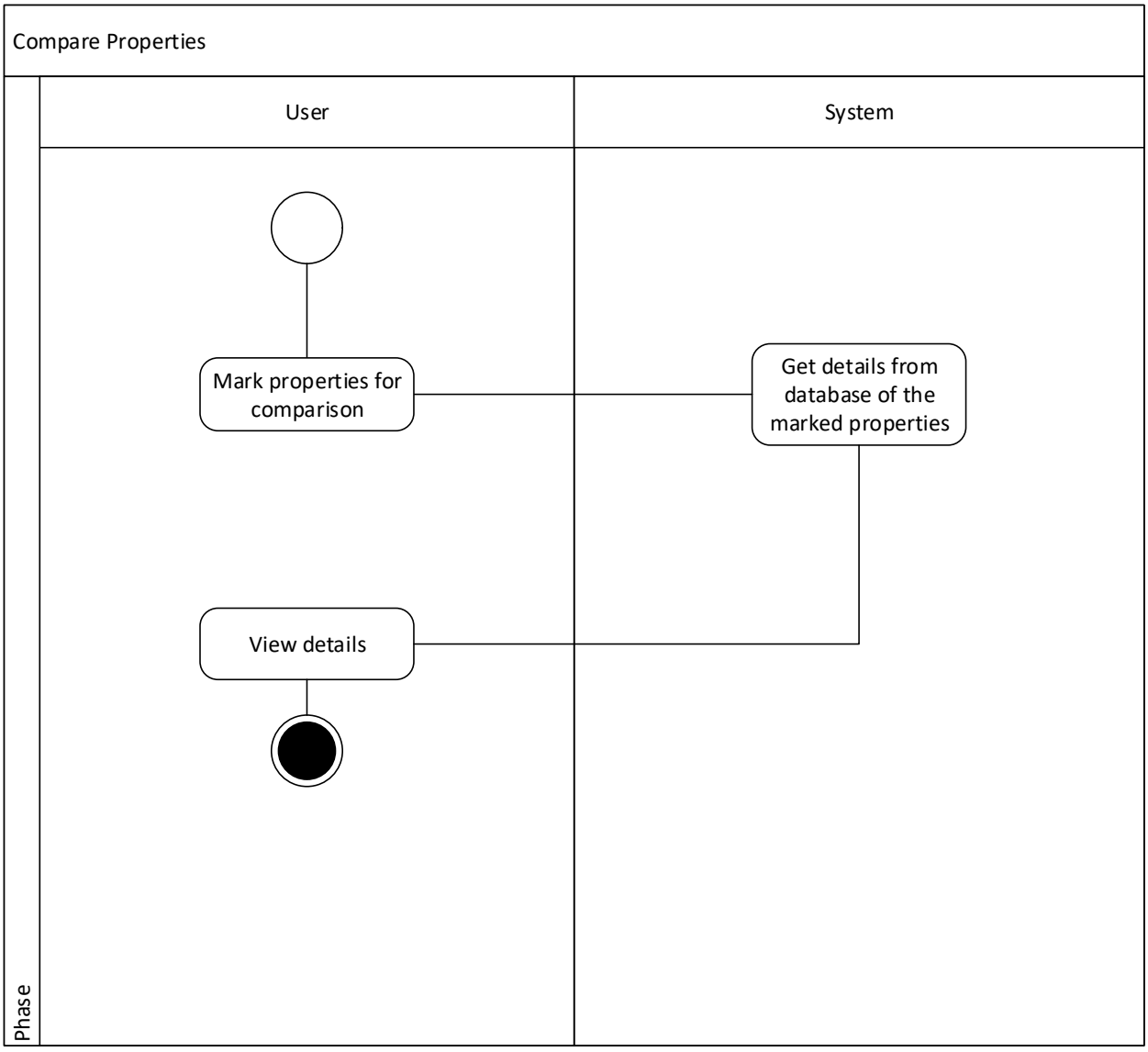

Subscribe To Newsletter
User
System
Subscribe to Newsletter
Save Subscription to Database
Send Email Confirmation
Email Confirmation

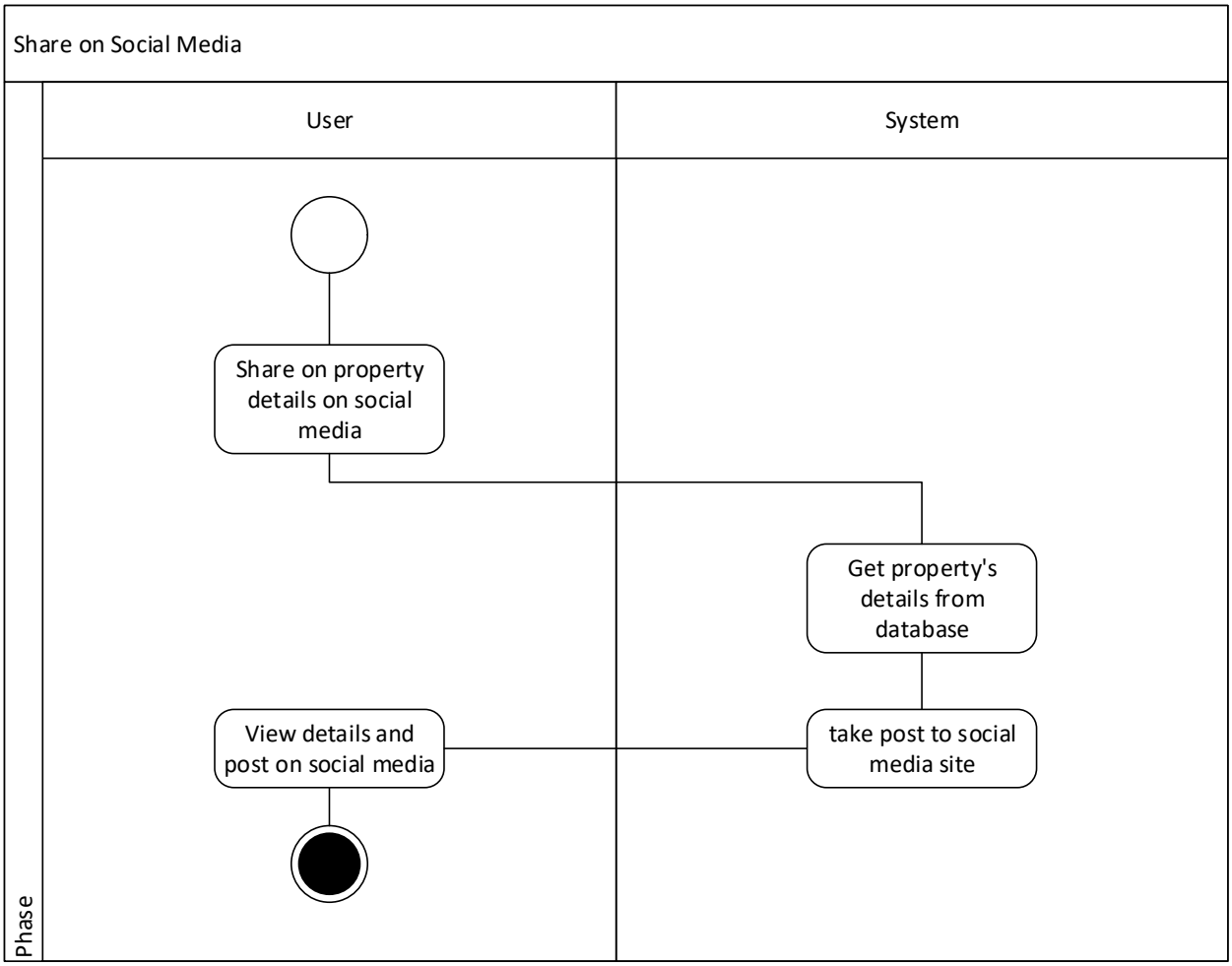

Unsubscribe To Newsletter
User
System
Unsubscribe to Newsletter
Save Unsubscription to Database
Send Email Confirmation
Email Confirmation

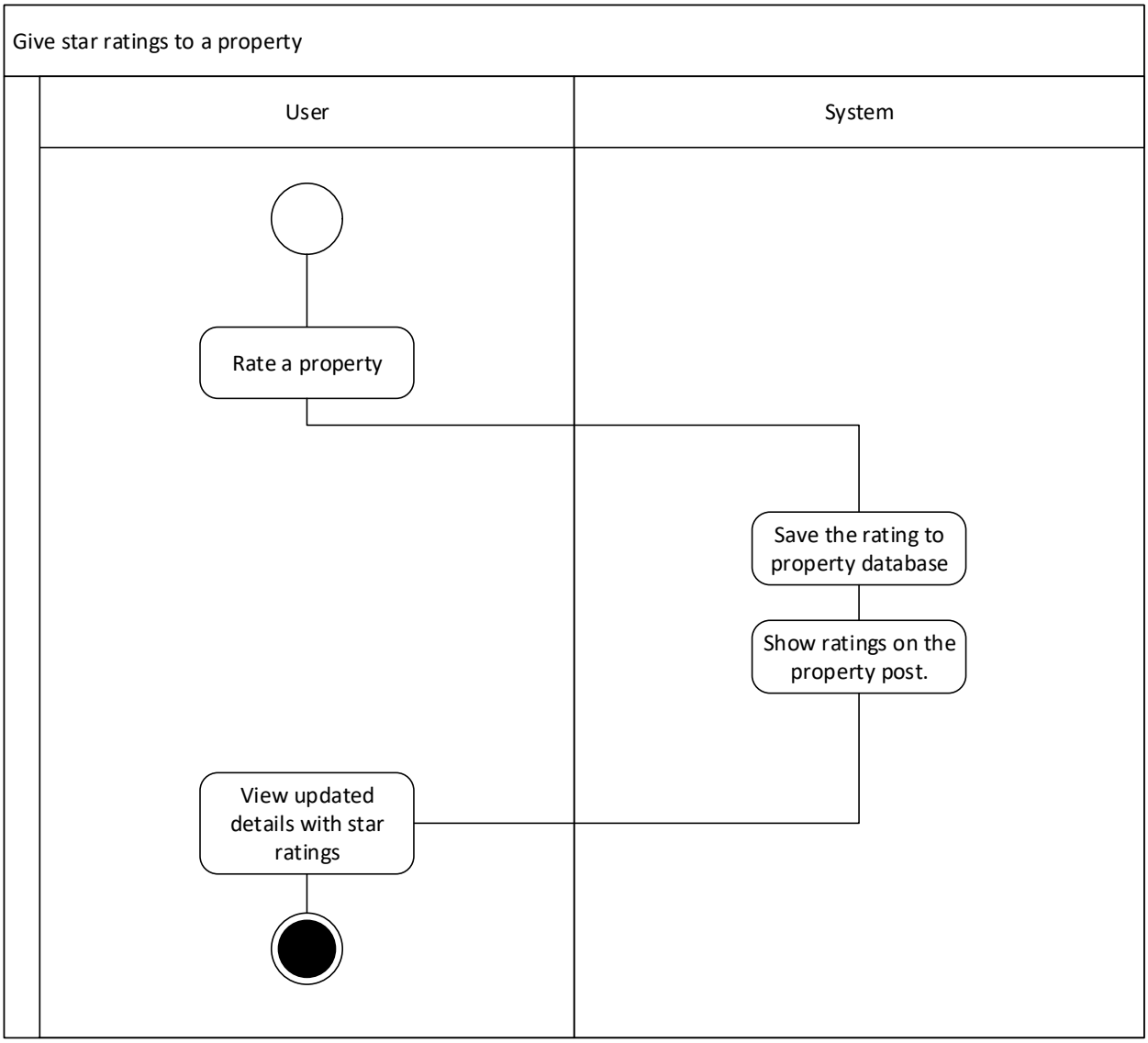

Update Property Information
User
System
Manage Property
Add/ Update Property
Update Confirmation
Property Updated

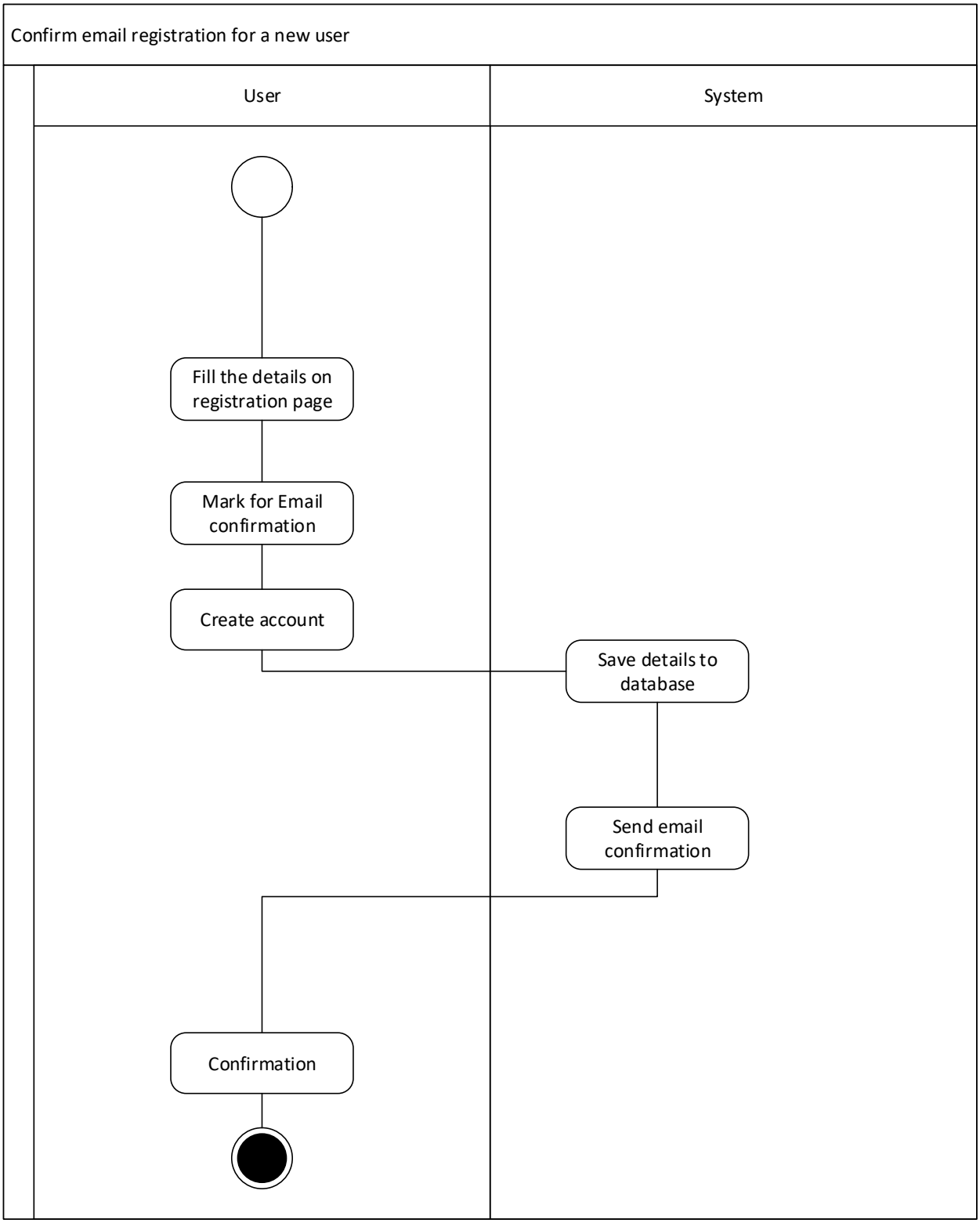

Generating a Report
User
System
Reports
Select Reports Category
Get Reports
Download PDF

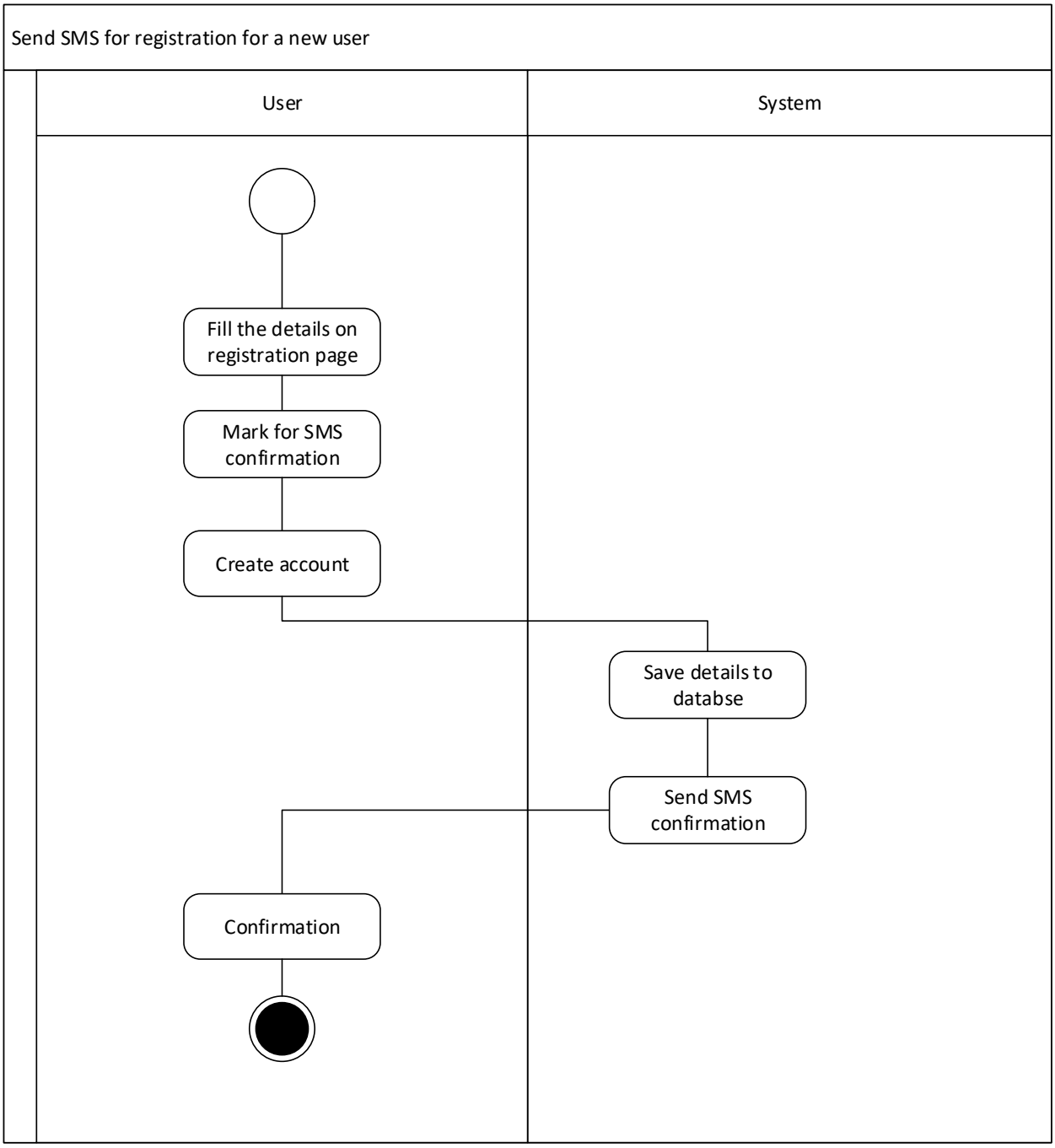

Retrieve lost password
User
System
Retrieve lost password
Check if email id is correct
No
Yes
View error message
Send password in the email to the user
View pop-up confirmation

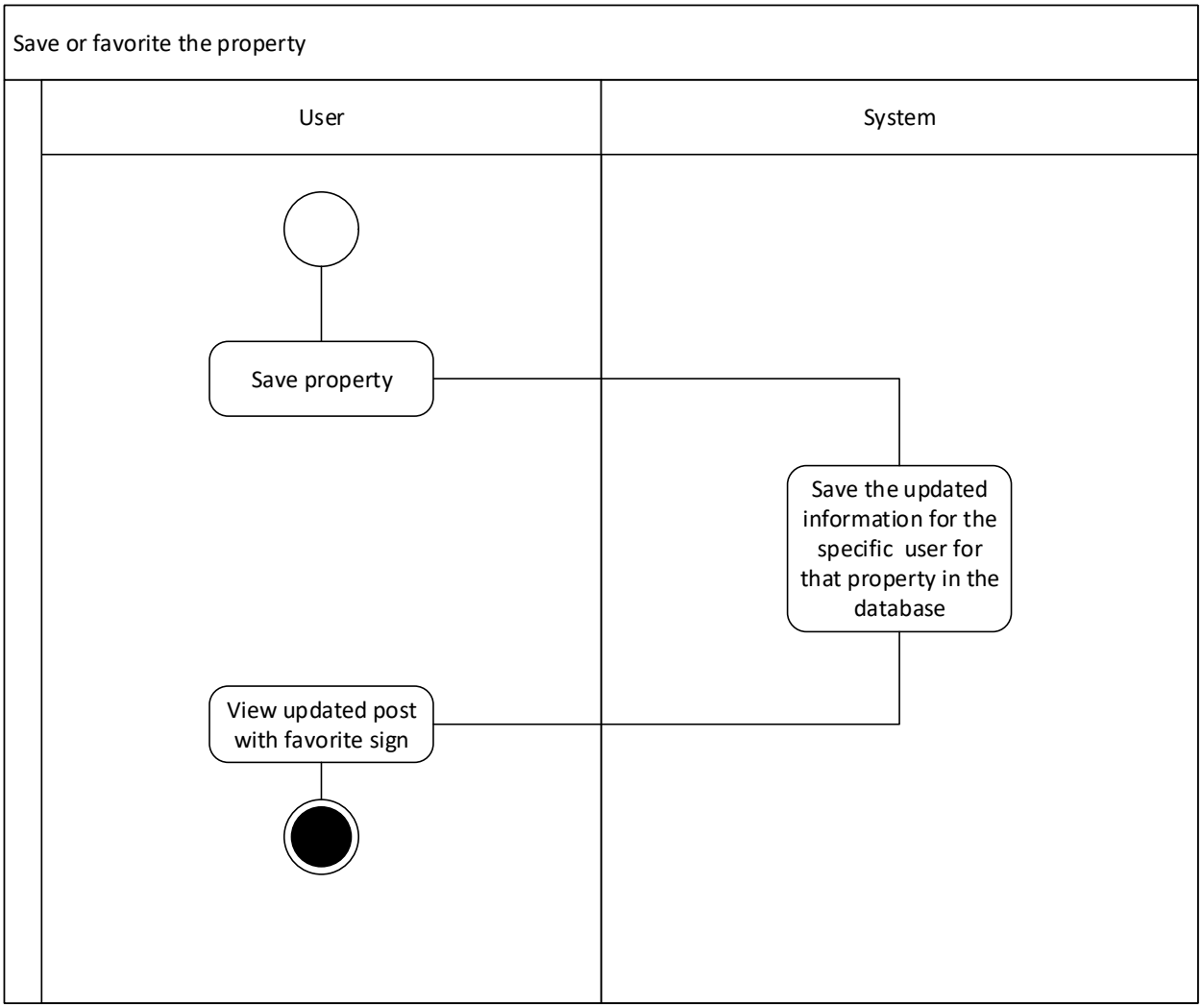

Compare Properties
User
System
Mark properties for comparison
Get details from database of the marked properties
View details
Phase

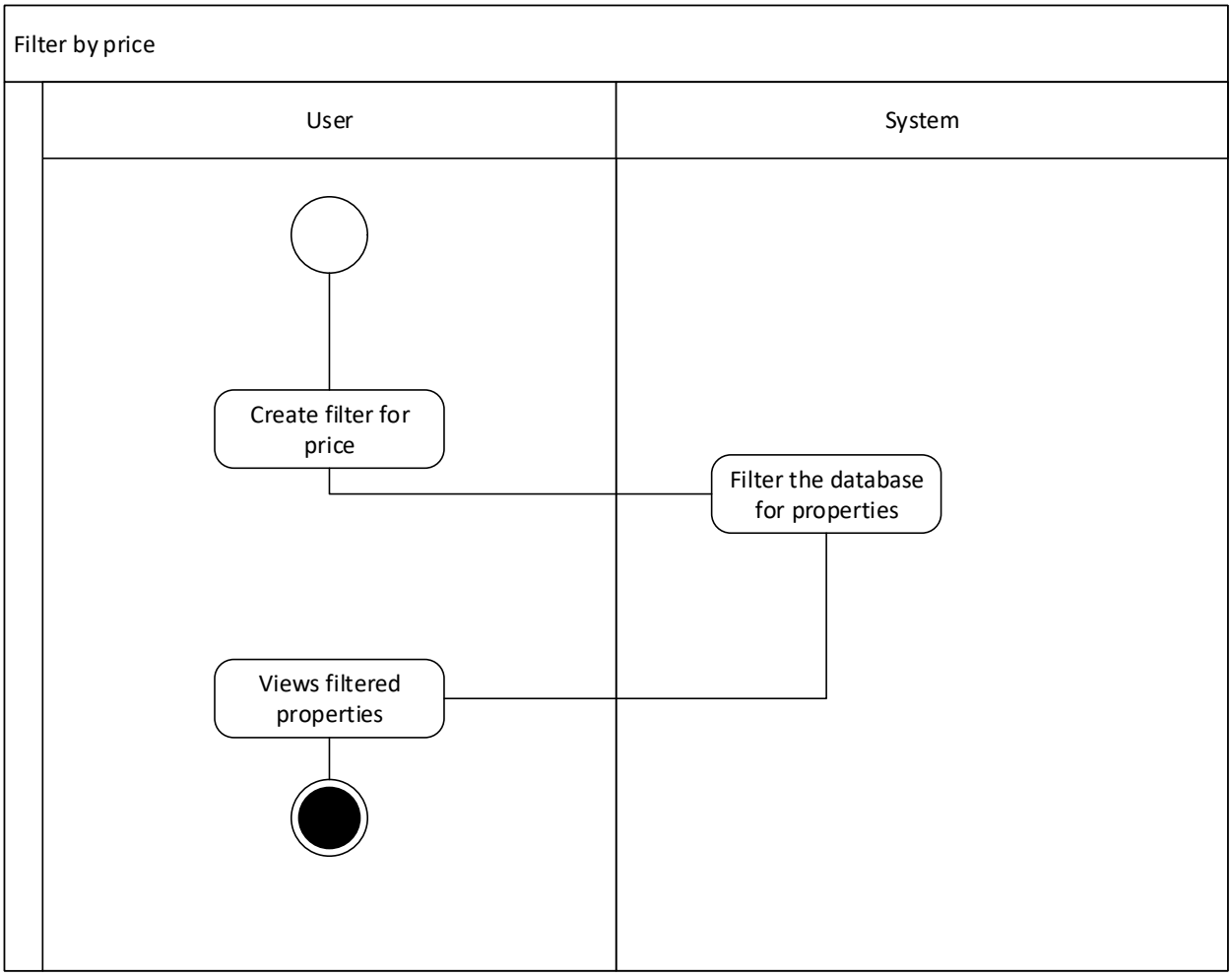

Share on Social Media
User
System
Share on property details on social media
Get property's details from database
View details and post on social media
take post to social media site
Phase

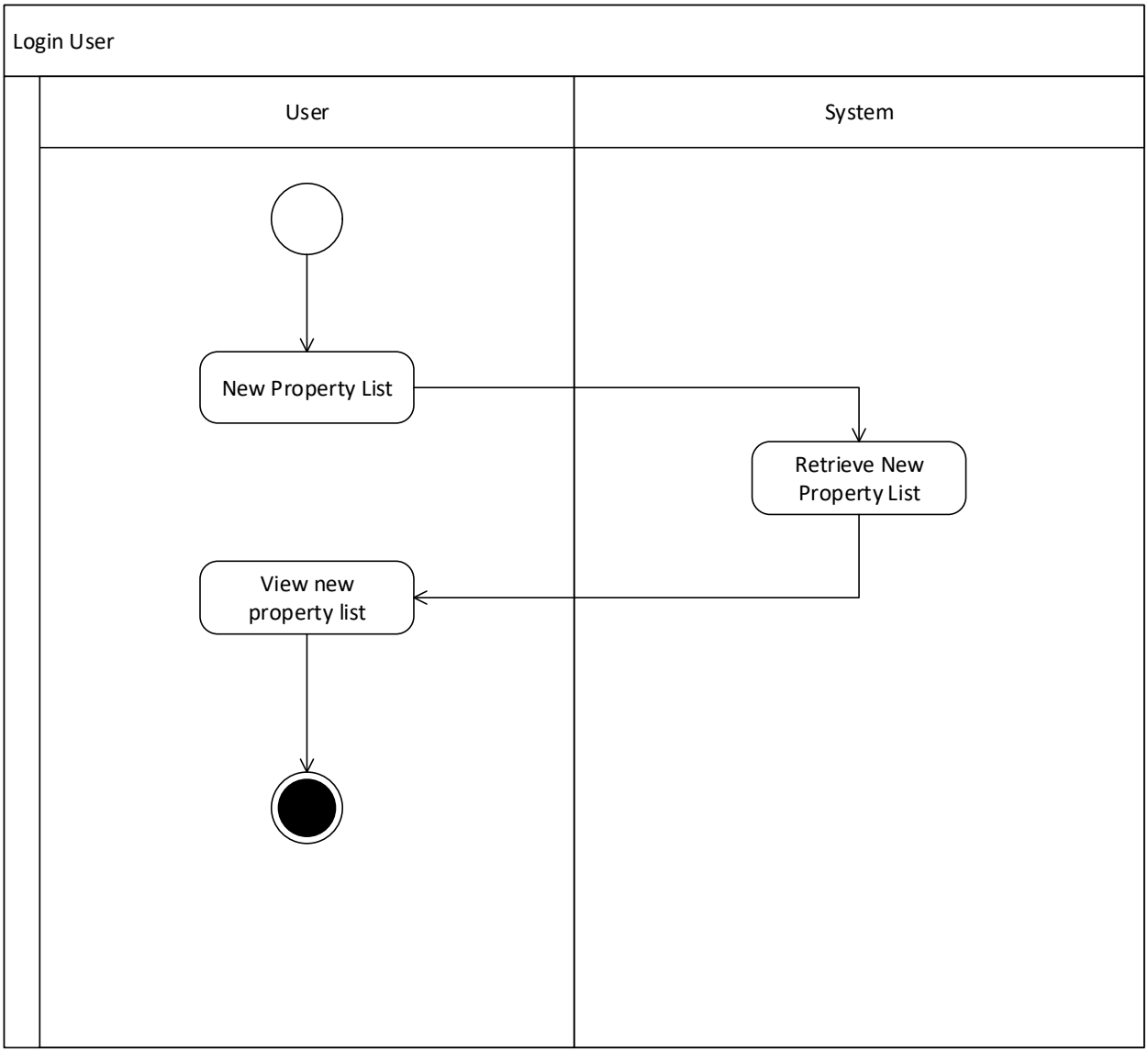

Give star ratings to a property
User
System
Rate a property
Save the rating to property database
Show ratings on the property post.
View updated details with star ratings

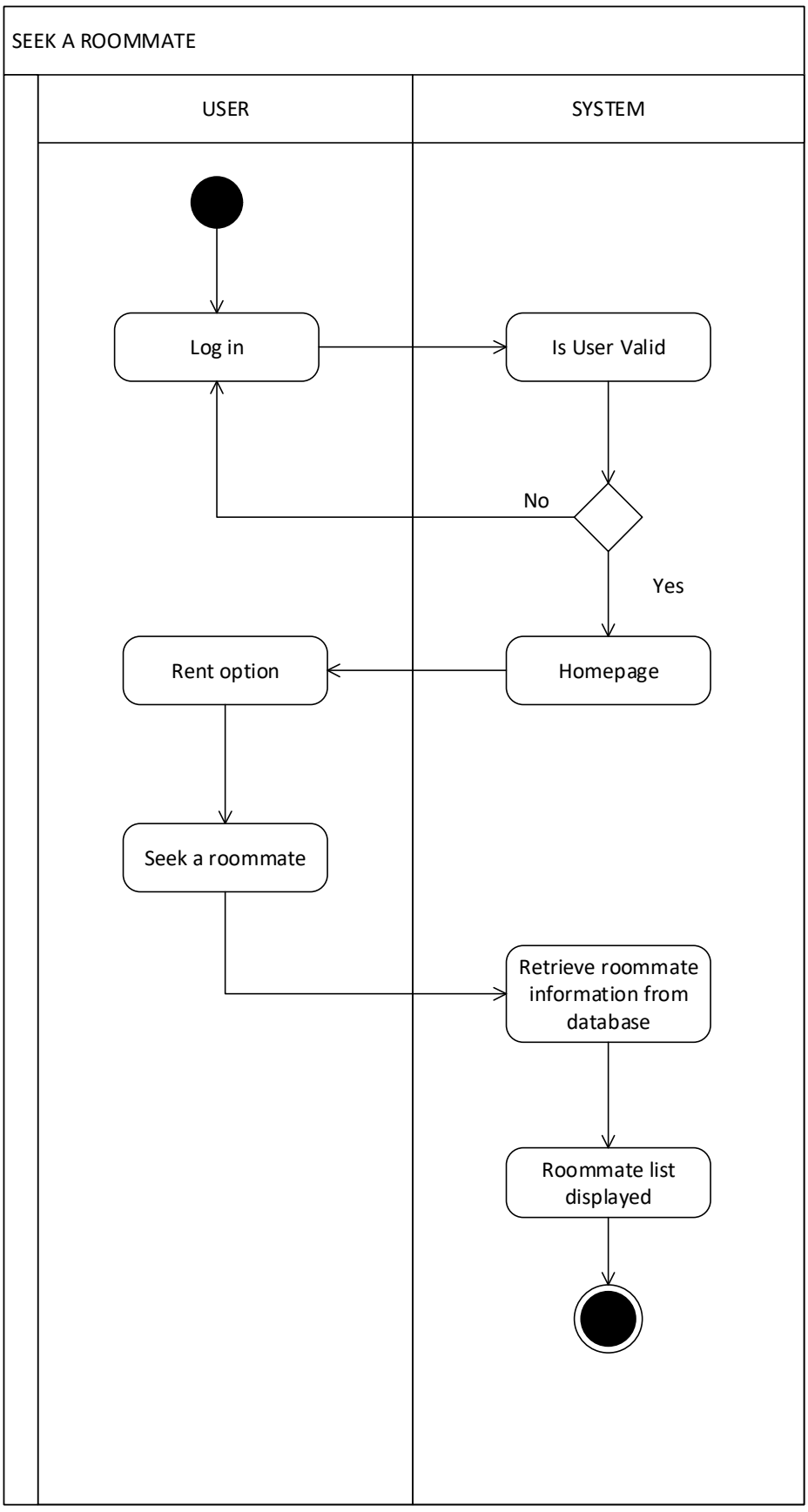

Confirm email registration for a new user
User
System
Fill the details on registration page
Mark for Email confirmation
Create account
Save details to database
Send email confirmation
Confirmation

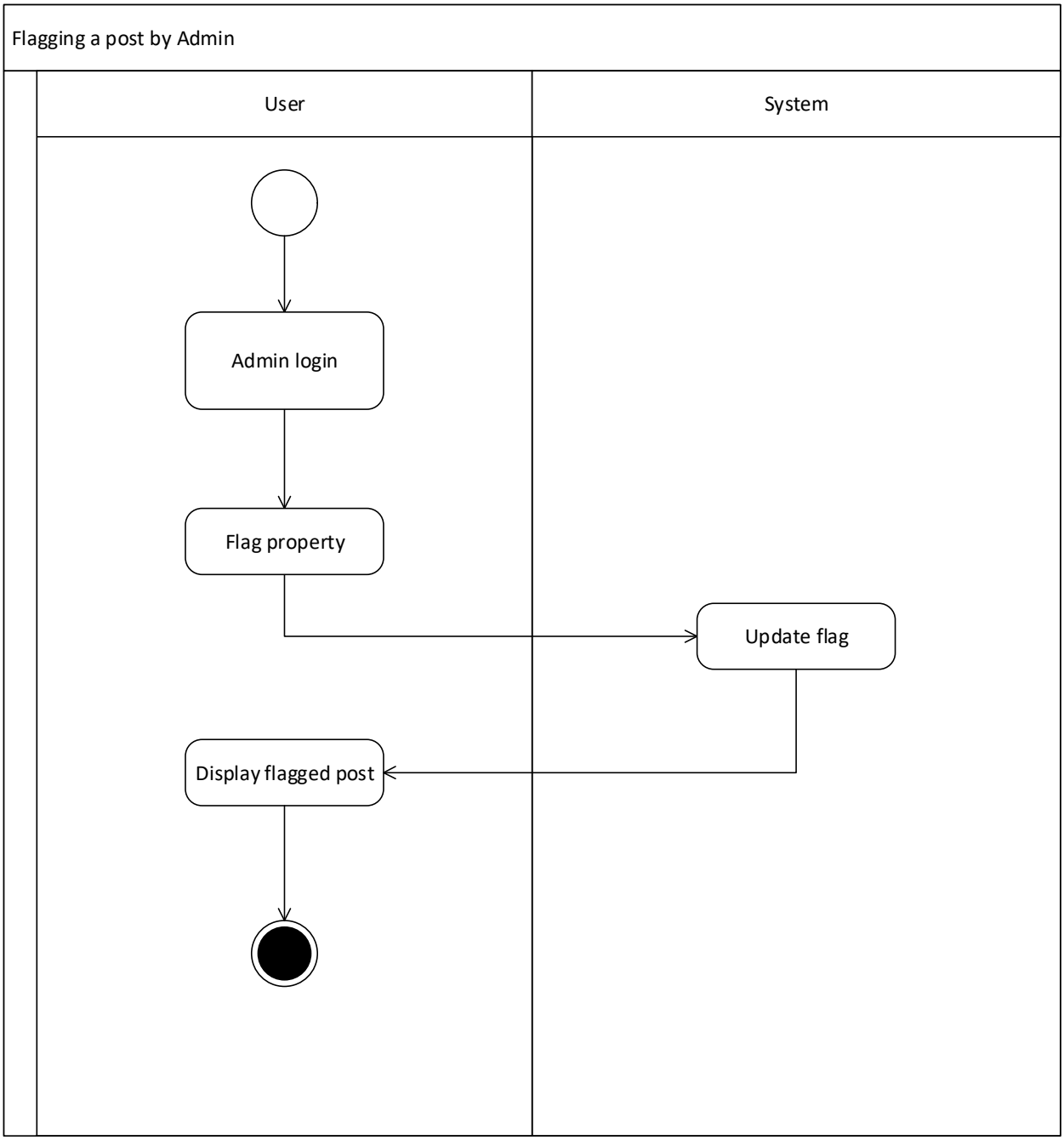

Send SMS for registration for a new user
User
System
Fill the details on registration page
Mark for SMS confirmation
Create account
Save details to databse
Send SMS confirmation
Confirmation

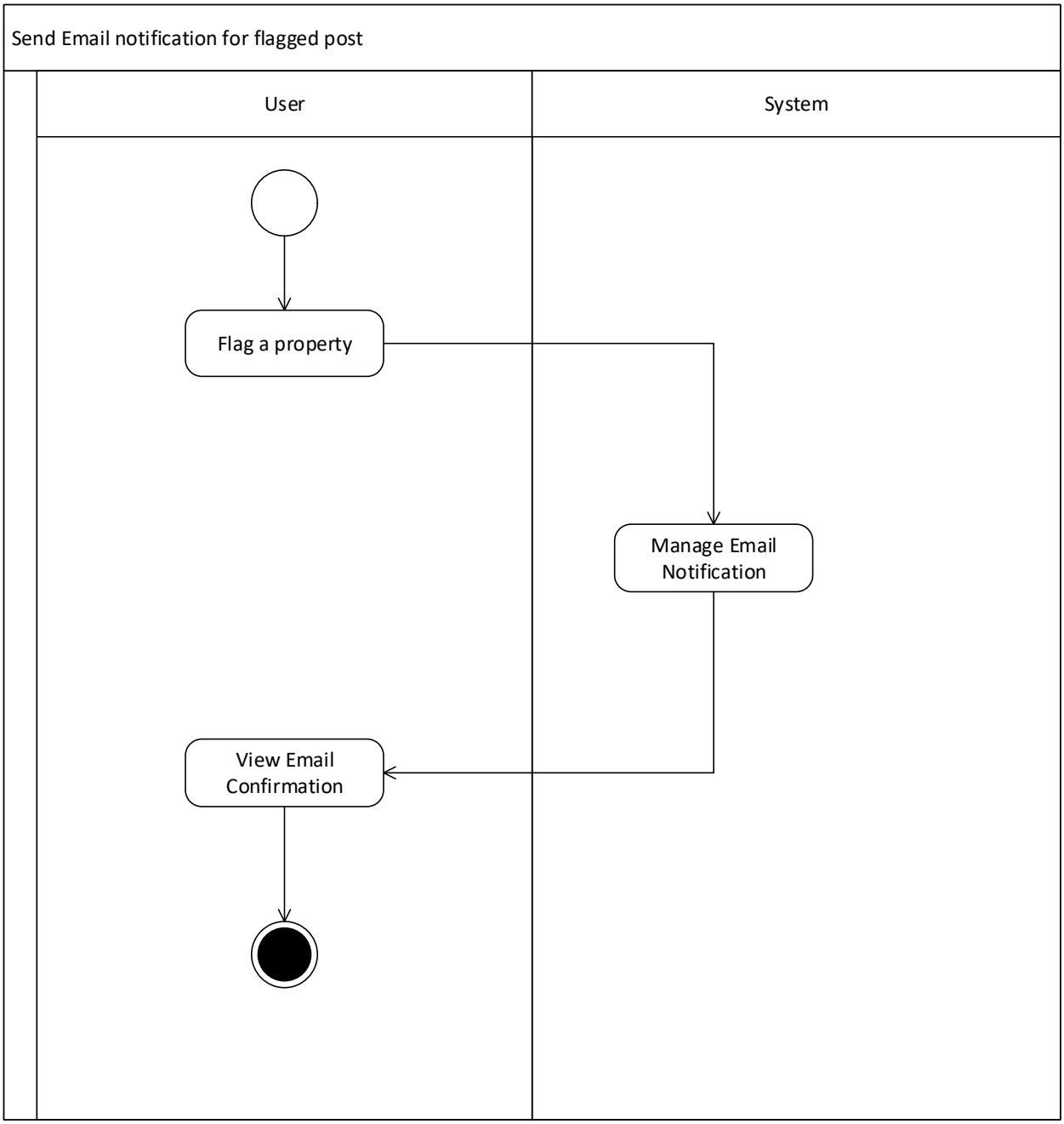

Save or favorite the property
User
System
Save property
Save the updated information for the specific user for that property in the database
View updated post with favorite sign

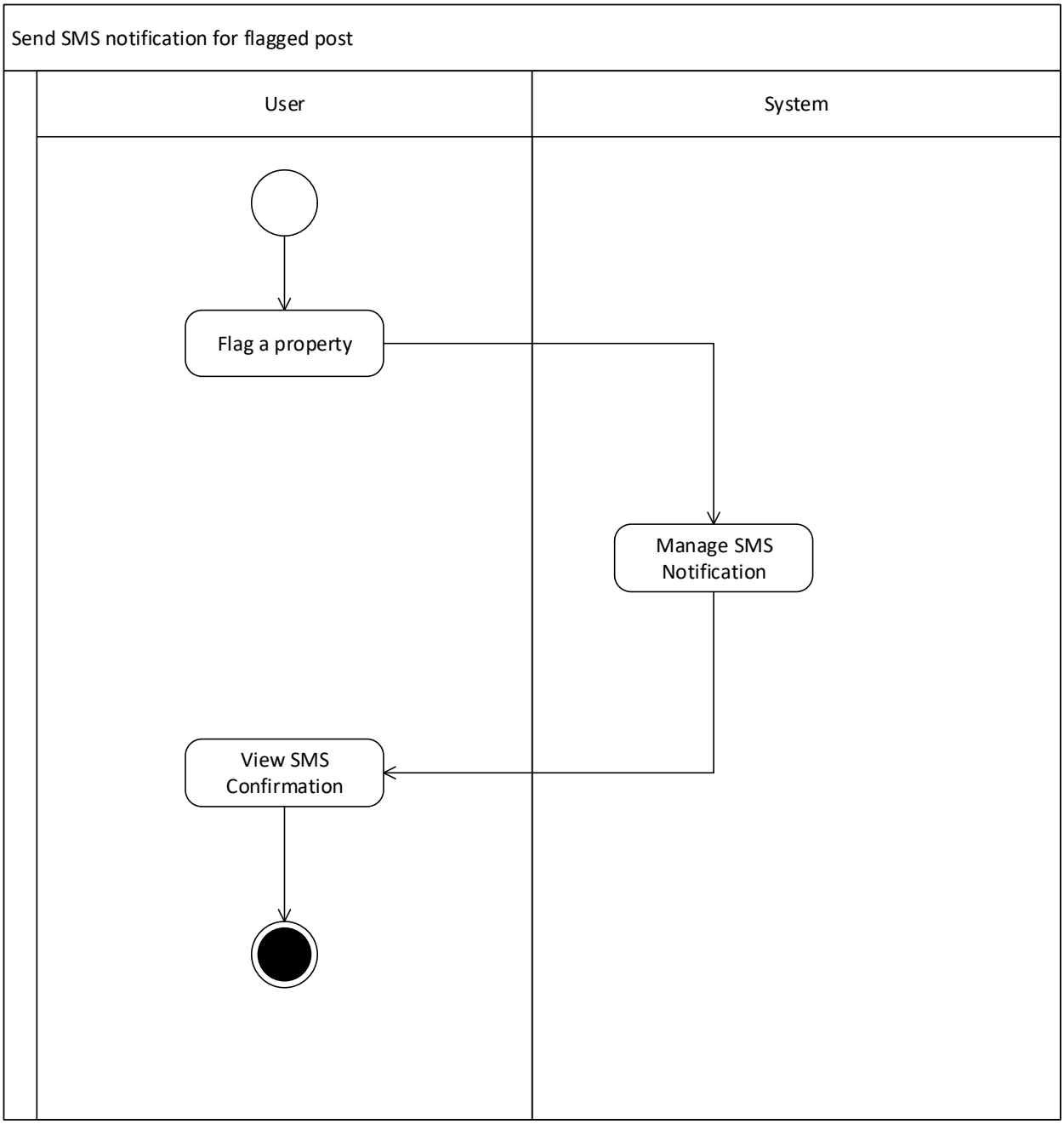

Filter by price
User
System
Create filter for price
Filter the database for properties
Views filtered properties

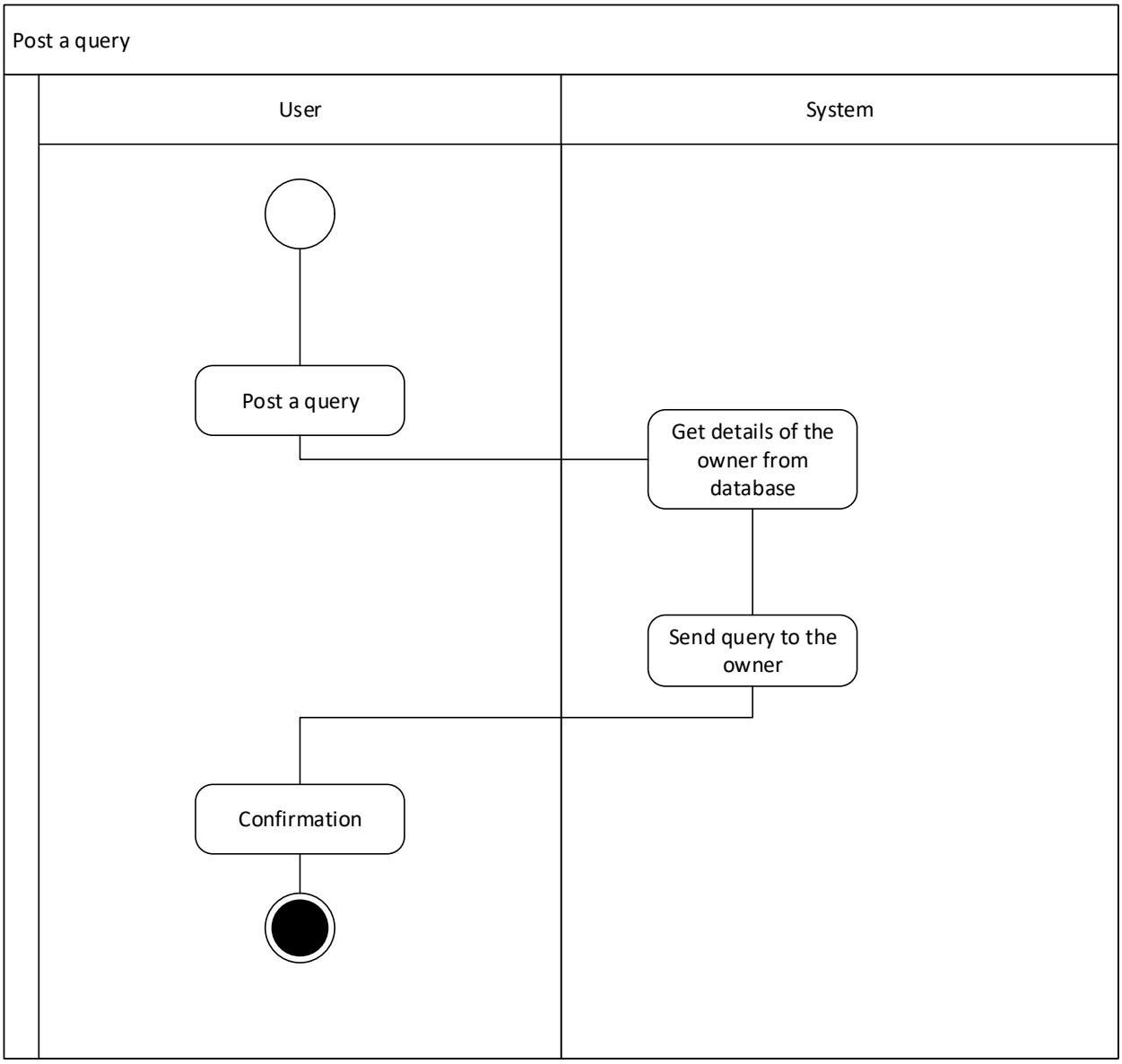

Login User
User
System
New Property List
Retrieve New Property List
View new property list

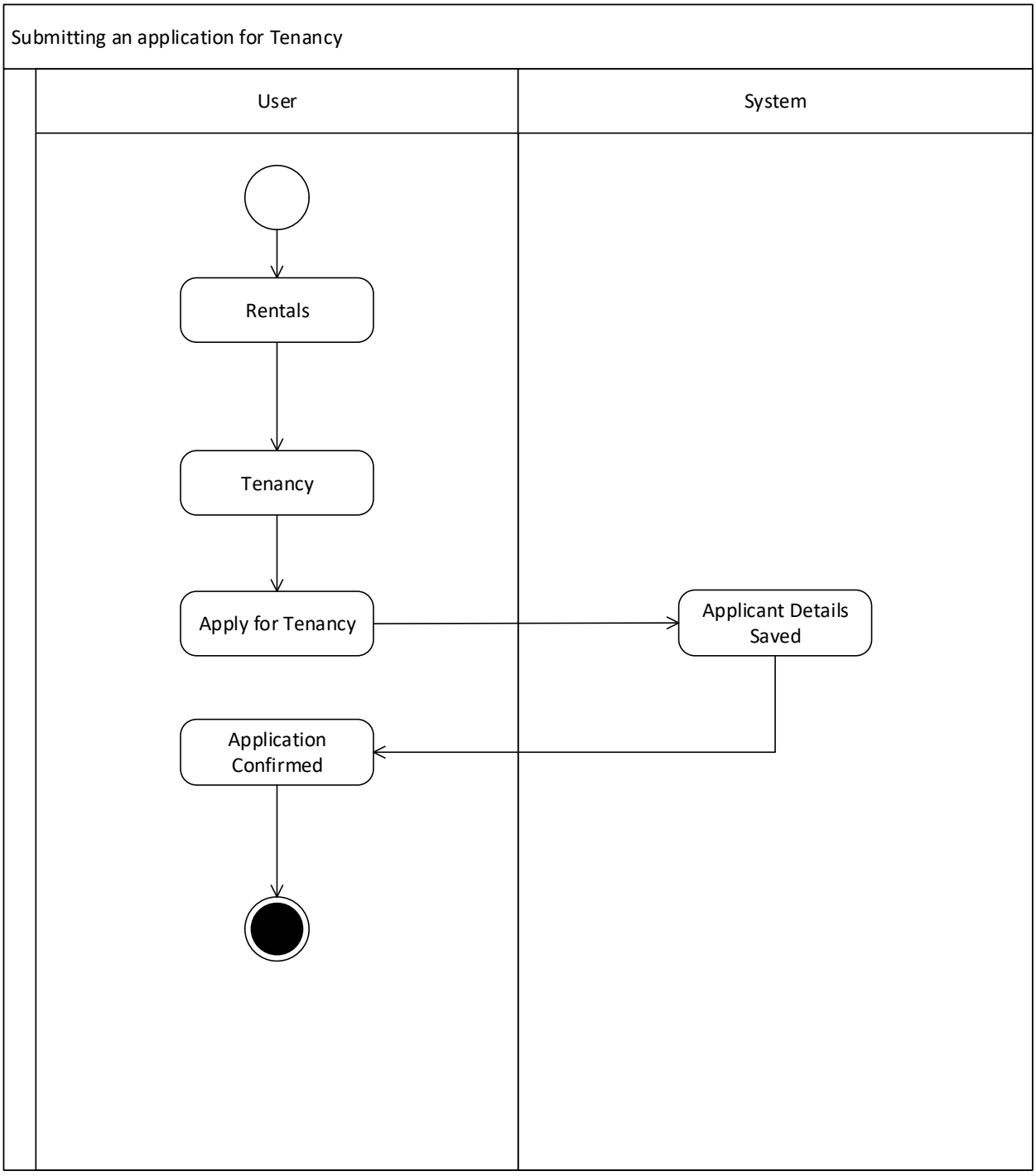

SEEK A ROOMMATE
USER
SYSTEM
Log in
Is User Valid
No
Yes
Rent option
Homepage
Seek a roommate
Retrieve roommate information from database
Roommate list displayed

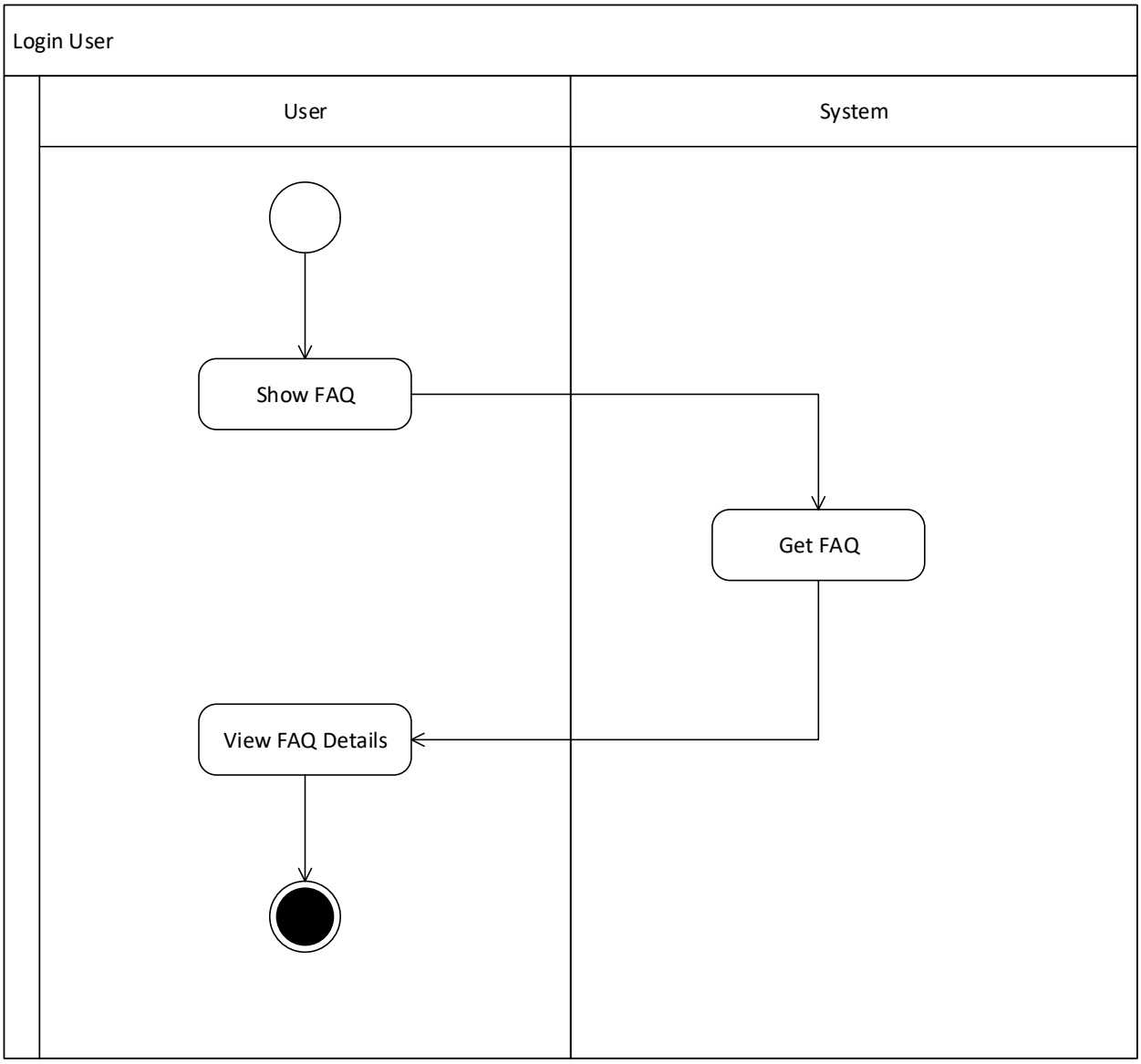

Flagging a post by Admin
User
System
Admin login
Flag property
Update flag
Display flagged post

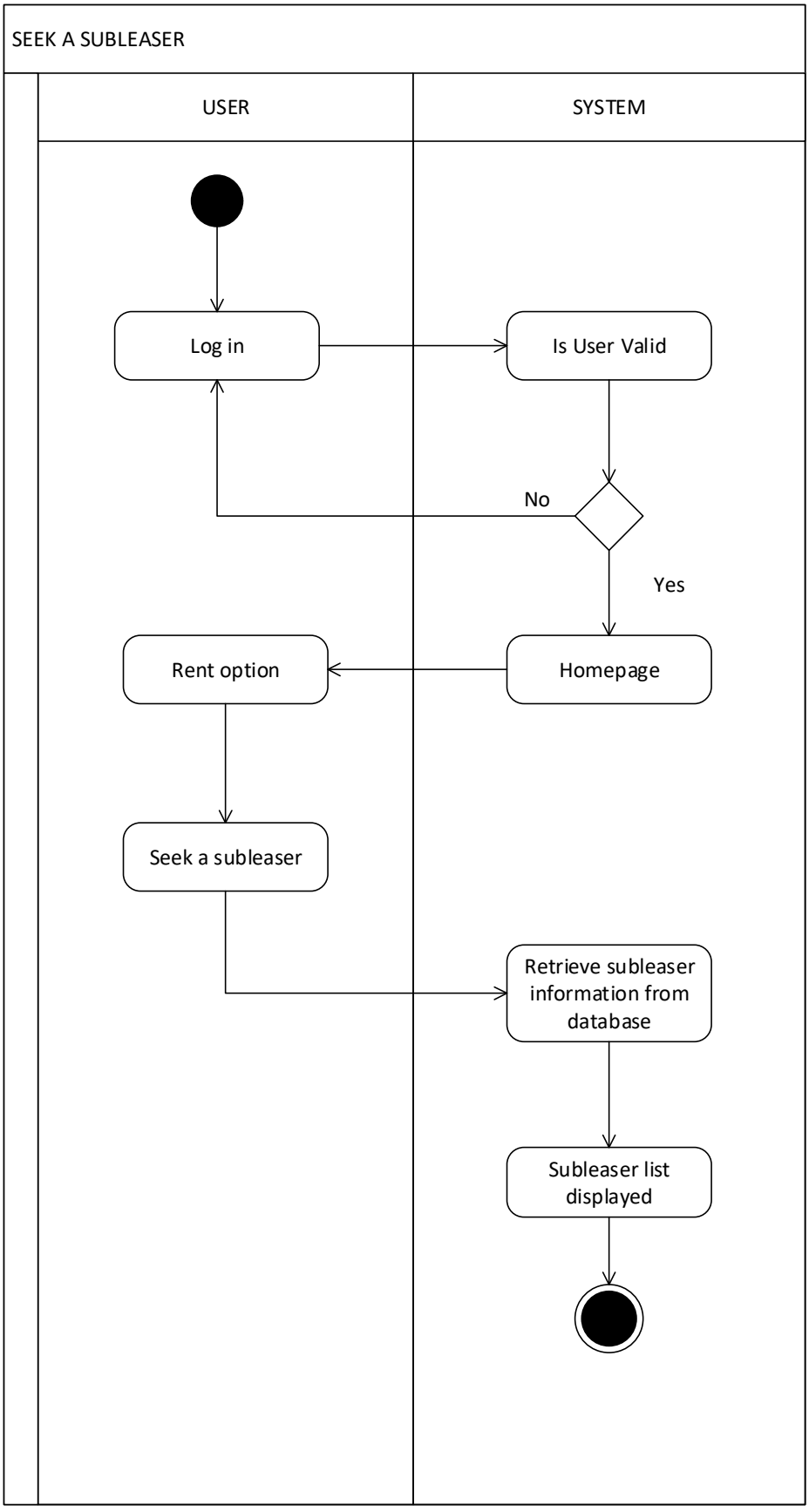

Send Email notification for flagged post
User
System
Flag a property
Manage Email Notification
View Email Confirmation

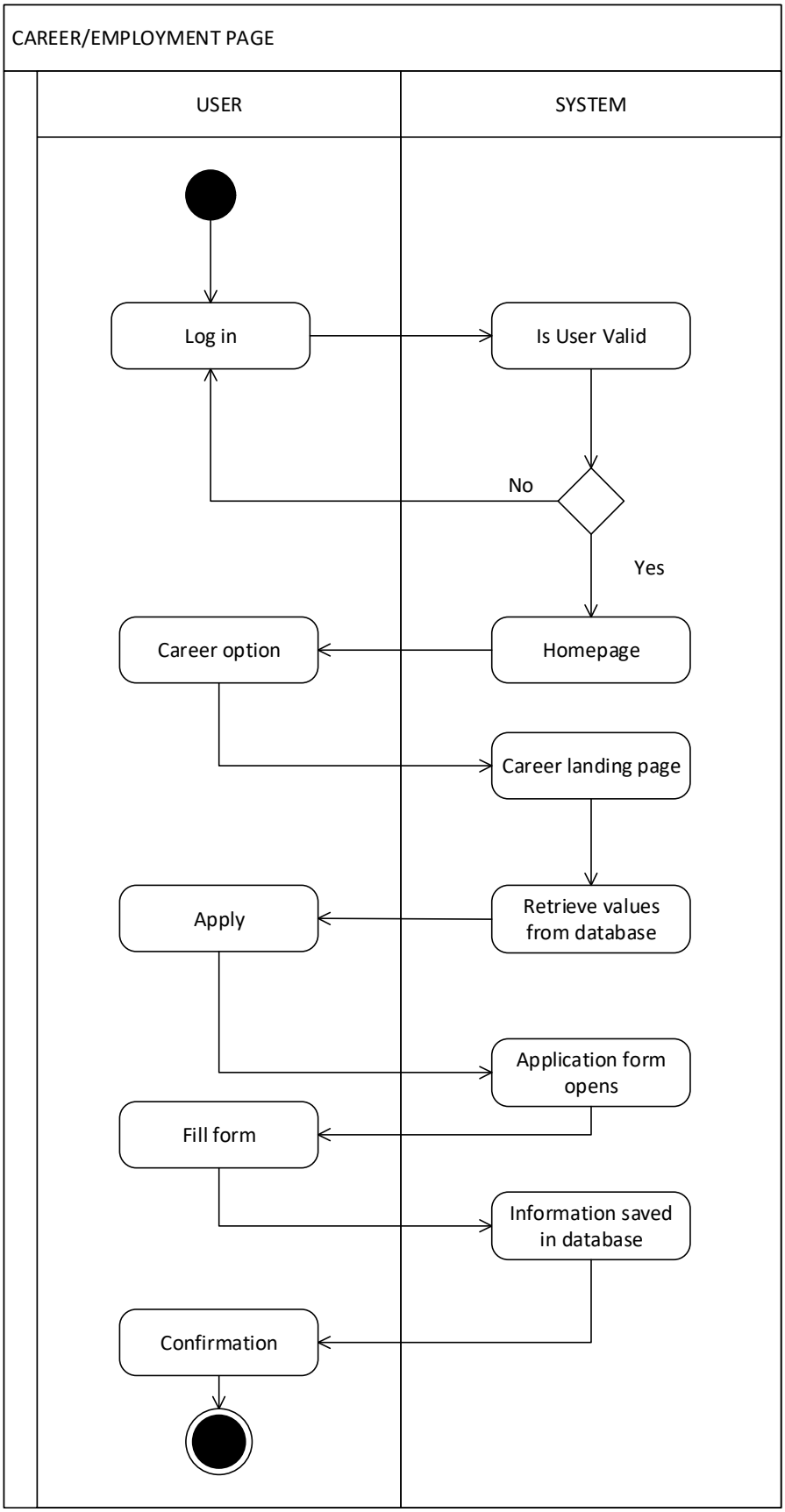

Send SMS notification for flagged post
User
System
Flag a property
Manage SMS Notification
View SMS Confirmation


Post a query
User
System
Post a query
Get details of the owner from database
Send query to the owner
Confirmation


Submitting an application for Tenancy
User
System
Rentals
Tenancy
Apply for Tenancy
Applicant Details Saved
Application Confirmed

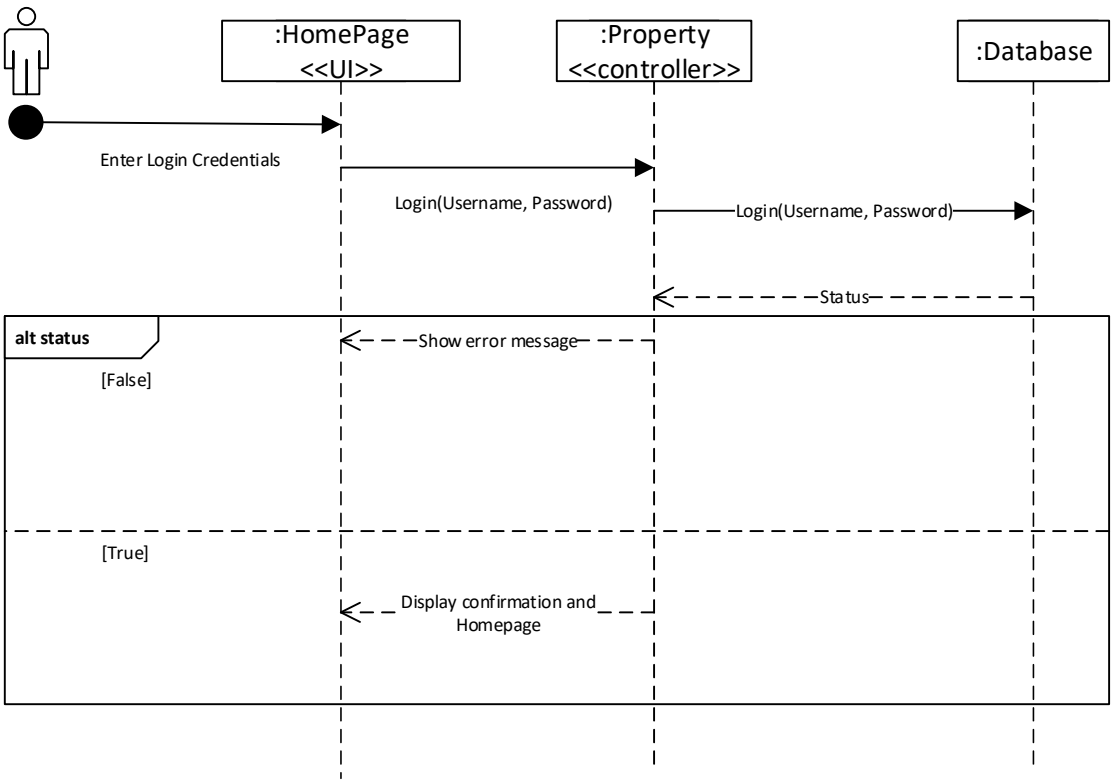

Login User
User
System
Show FAQ
Get FAQ
View FAQ Details


SEEK A SUBLEASER
USER
SYSTEM
Log in
Is User Valid
No
Yes
Rent option
Homepage
Seek a subleaser
Retrieve subleaser information from database
Subleaser list displayed

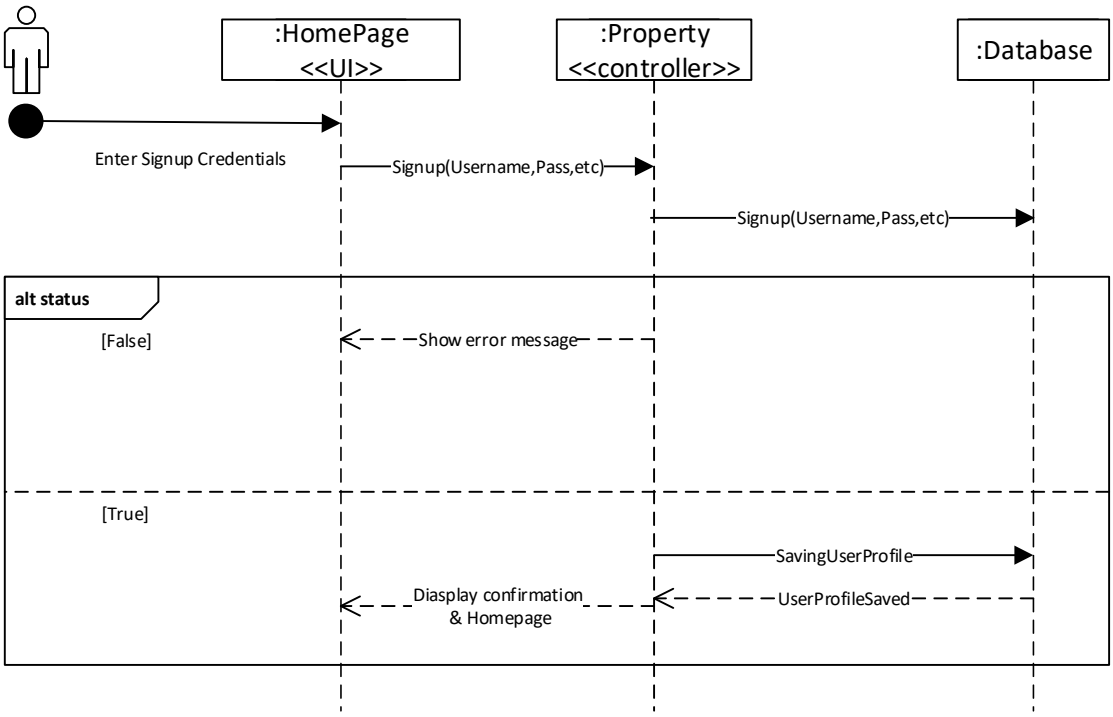

CAREER/EMPLOYMENT PAGE
USER
SYSTEM
Log in
Is User Valid
No
Yes
Career option
Homepage
Career landing page
Apply
Retrieve values from database
Application form opens
Fill form
Information saved in database
Confirmation

4.6.3.4 Sequence Diagram

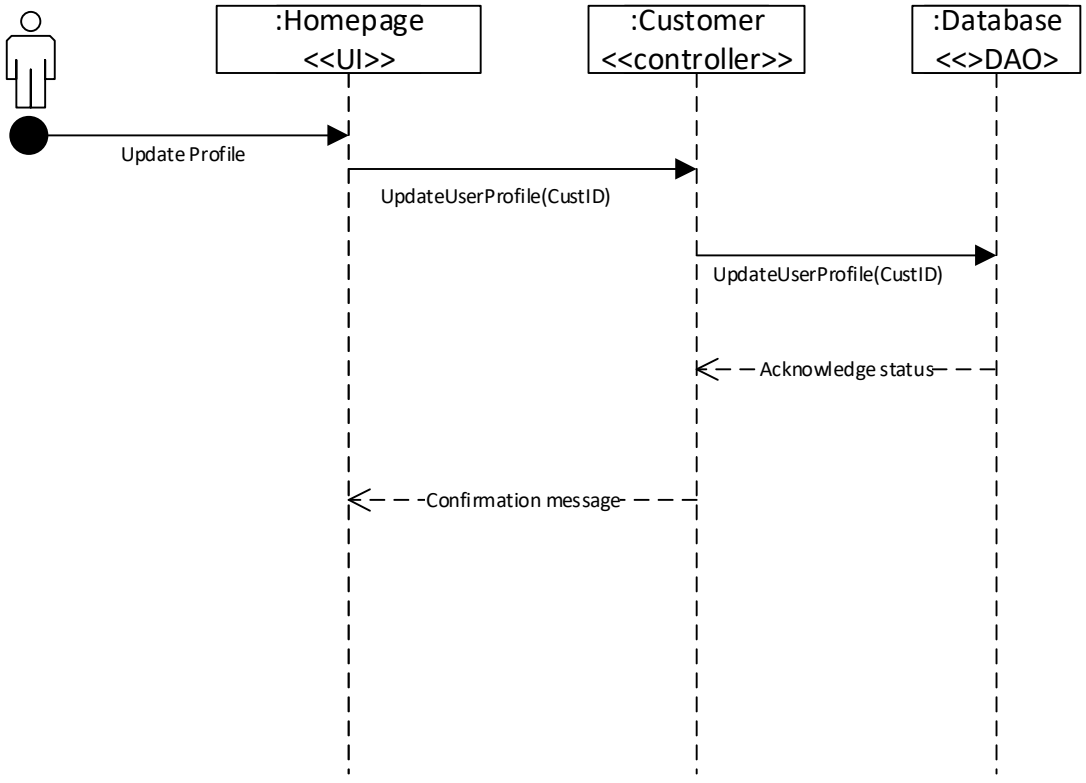
1. Sign in



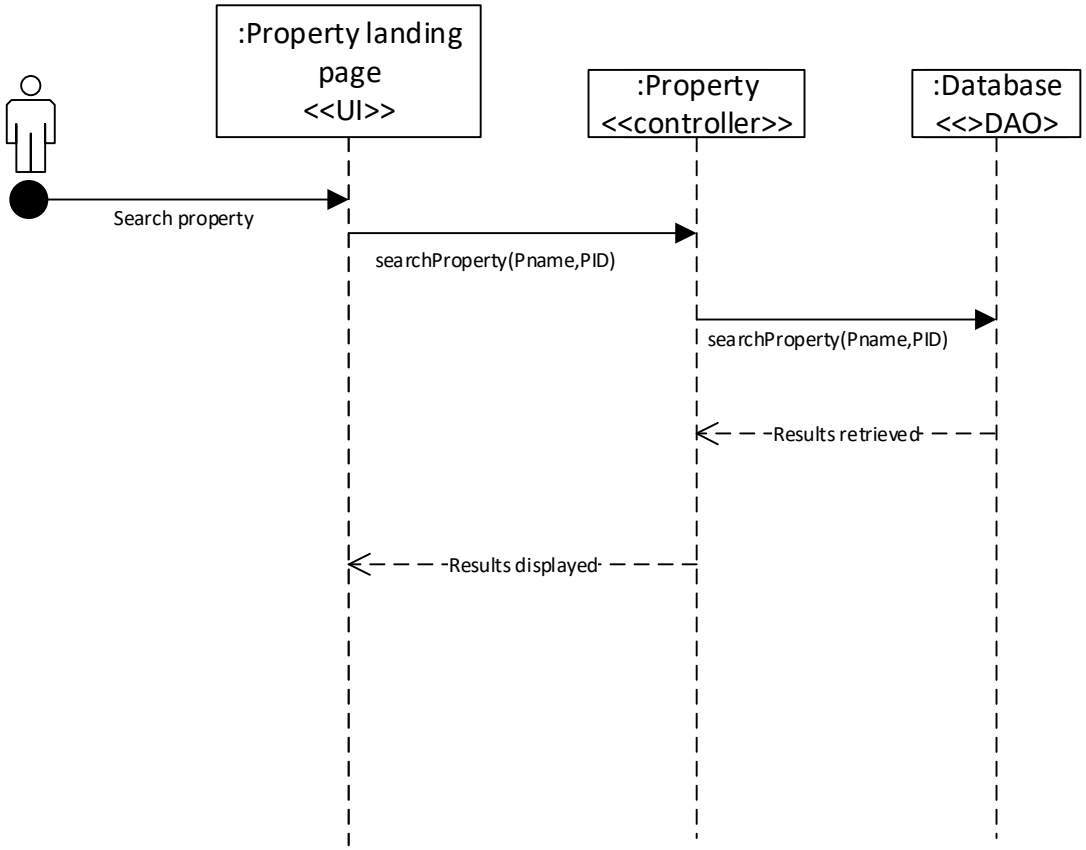
2. Sign up



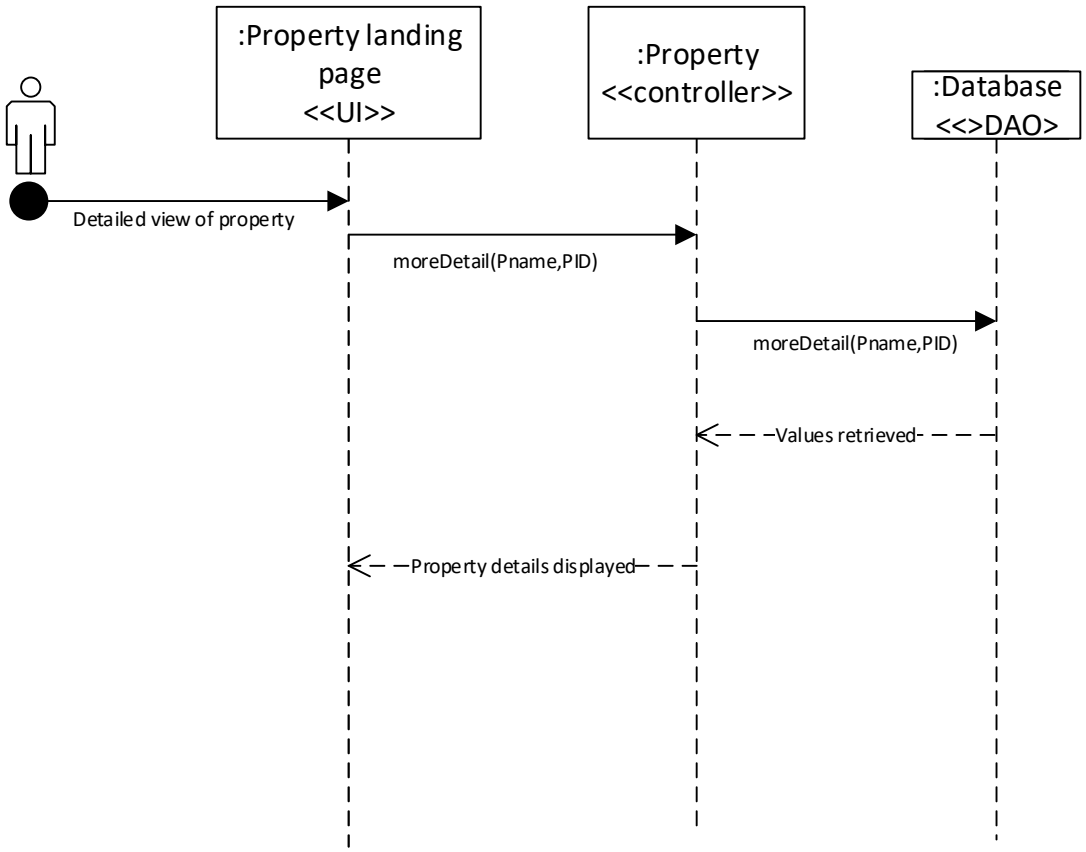
3. Update User profile



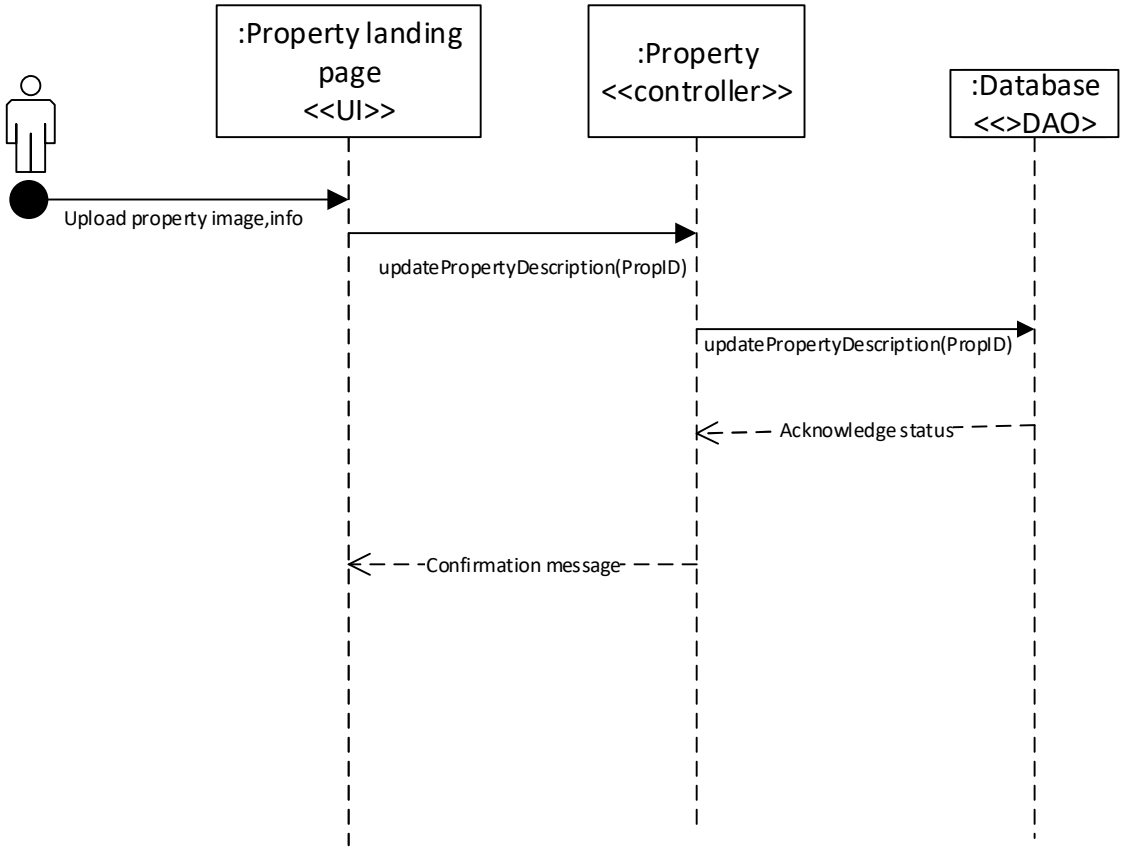
4. Search property



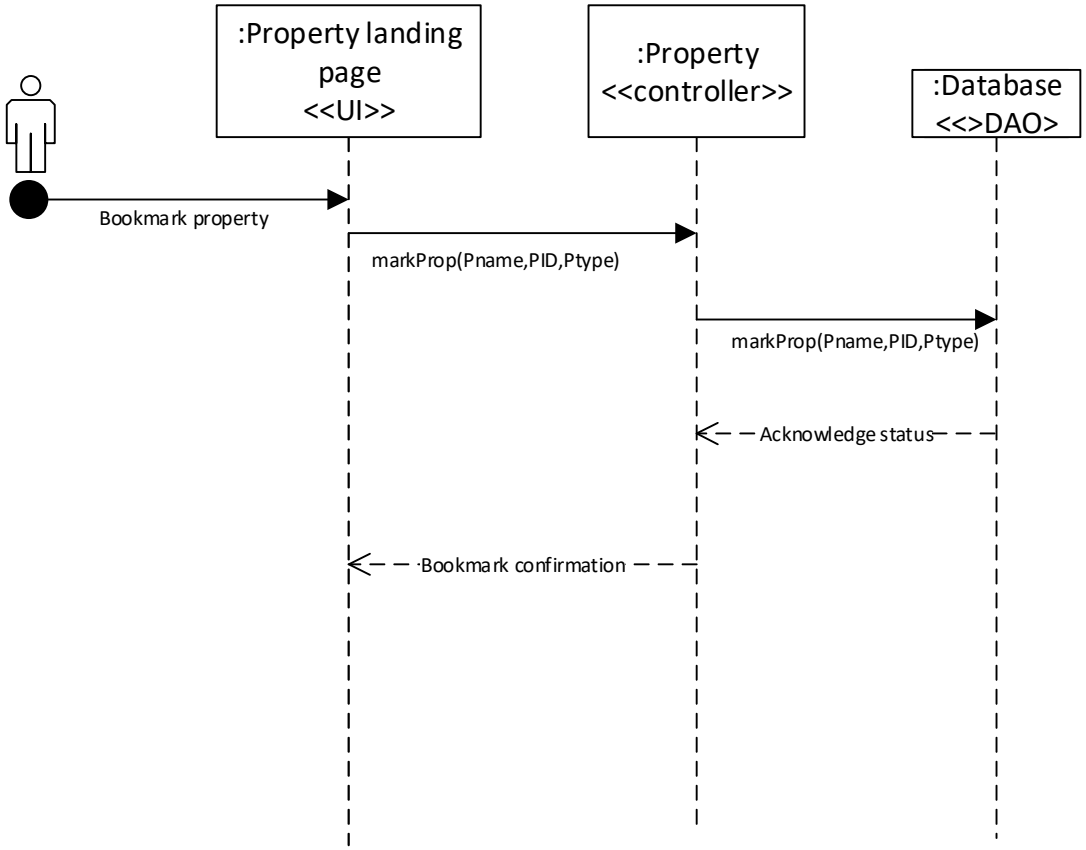
5. View detailed posting


:Property landing page <<UI>>
:Property <<controller>>
:Database <<>DAO>
Detailed view of property
moreDetail(Pname,PID)
moreDetail(Pname,PID)
Values retrieved
Property details displayed

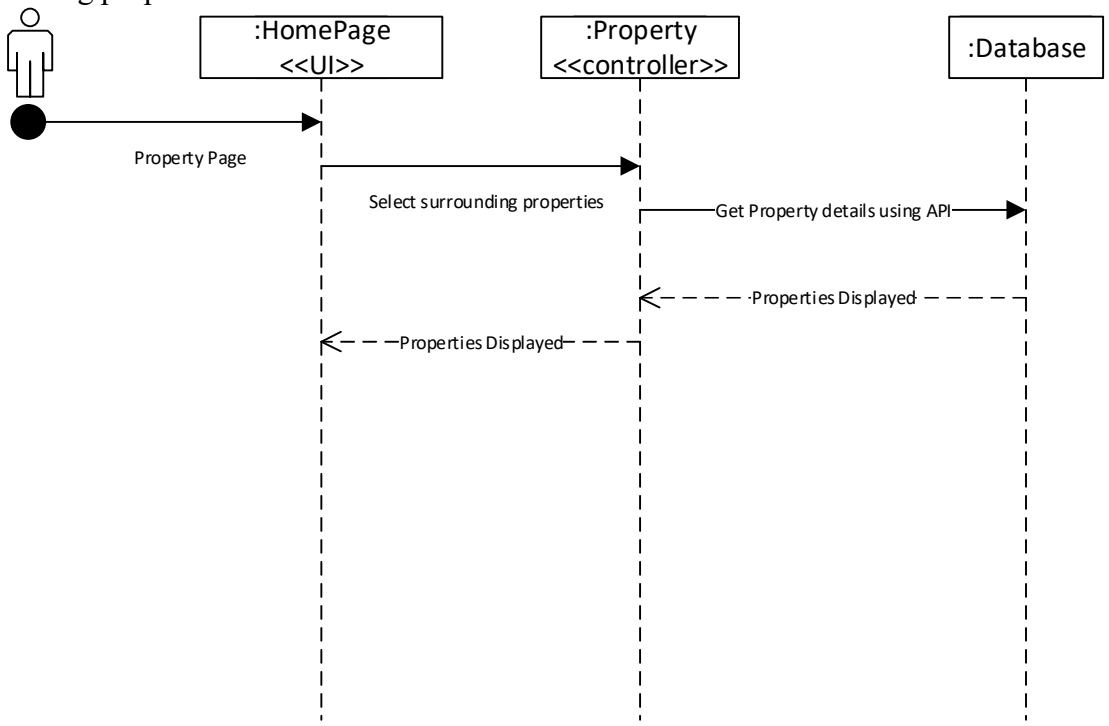
Upload Property



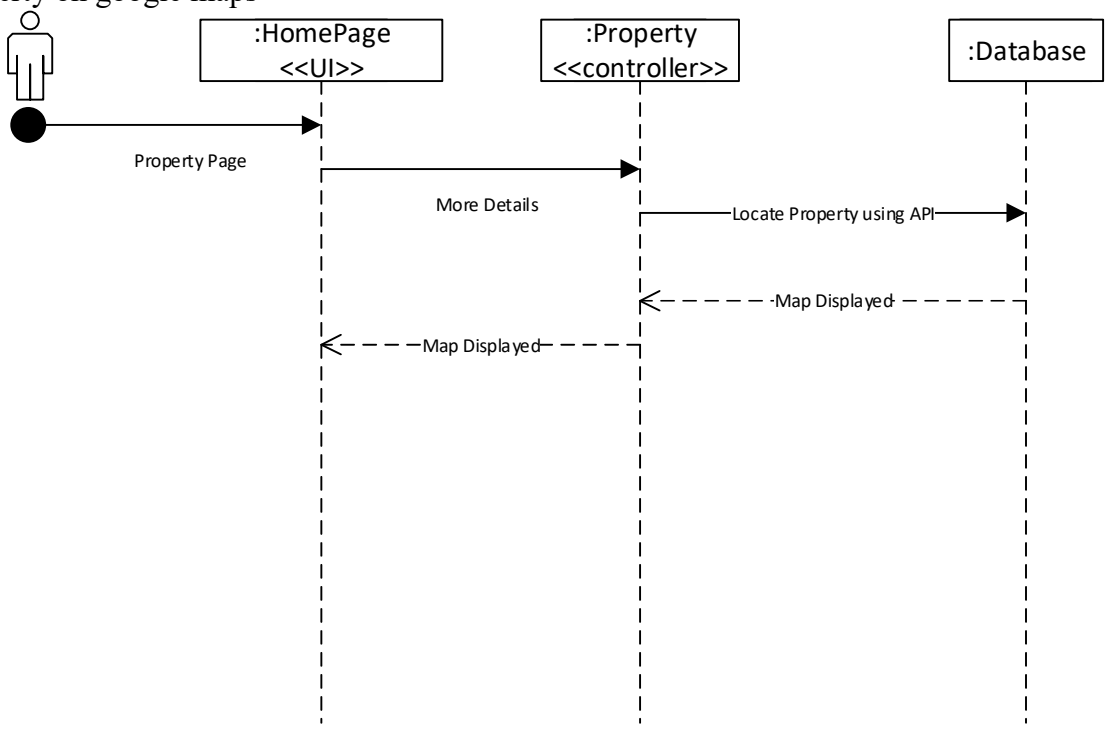
Bookmark property



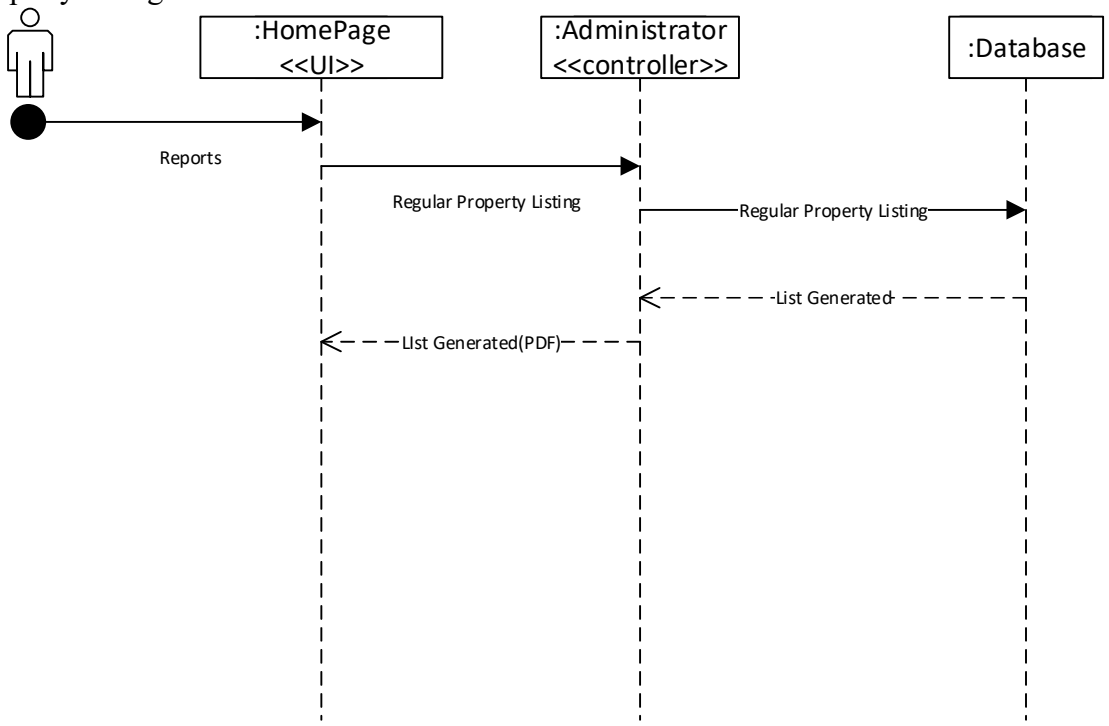
Show surrounding properties



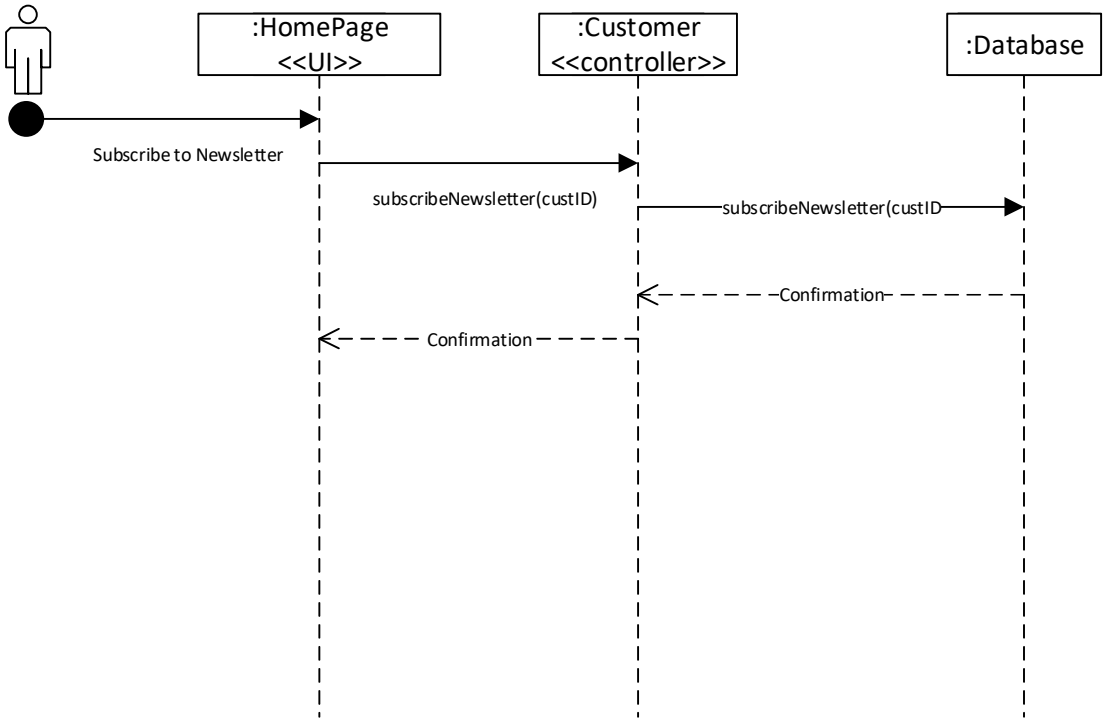
Show property on google maps



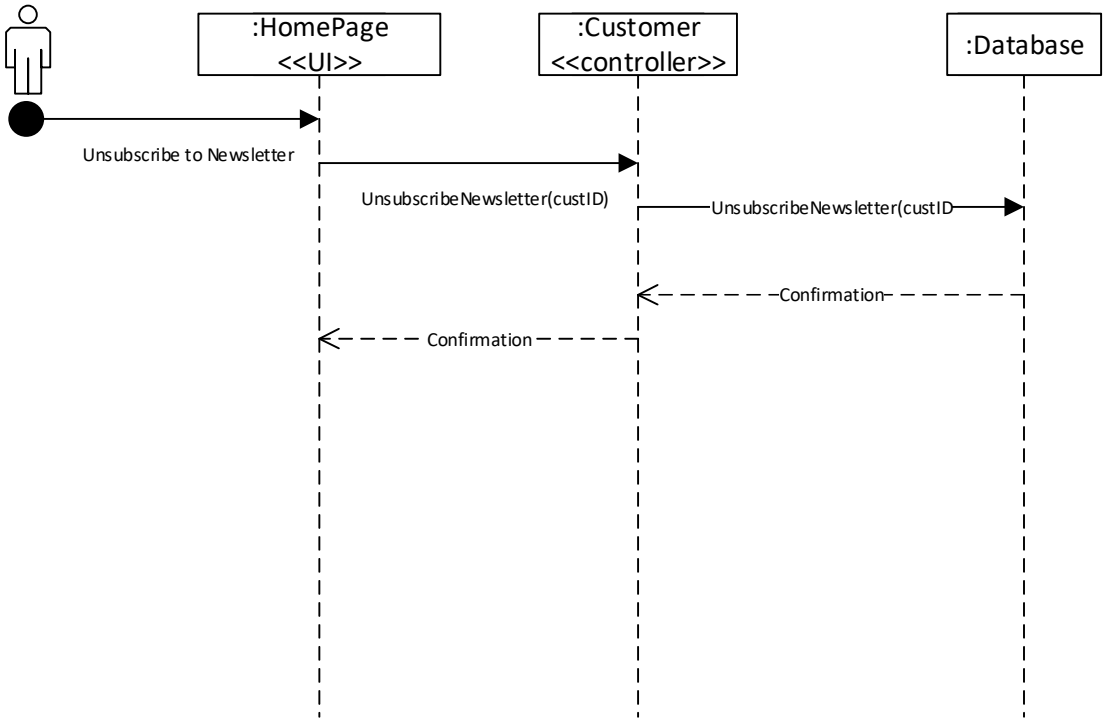
Regular property listing



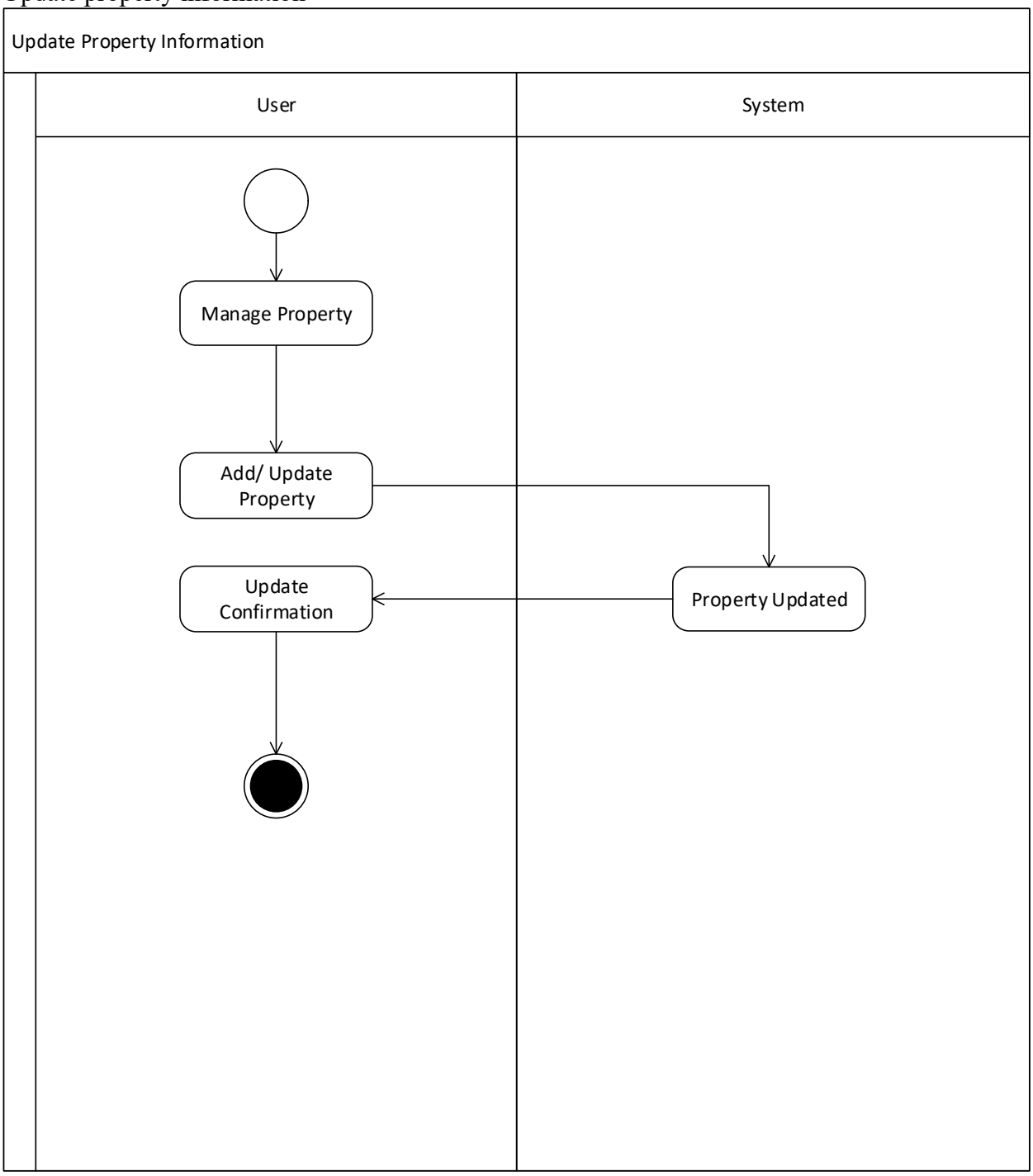
Subscribe to newsletter



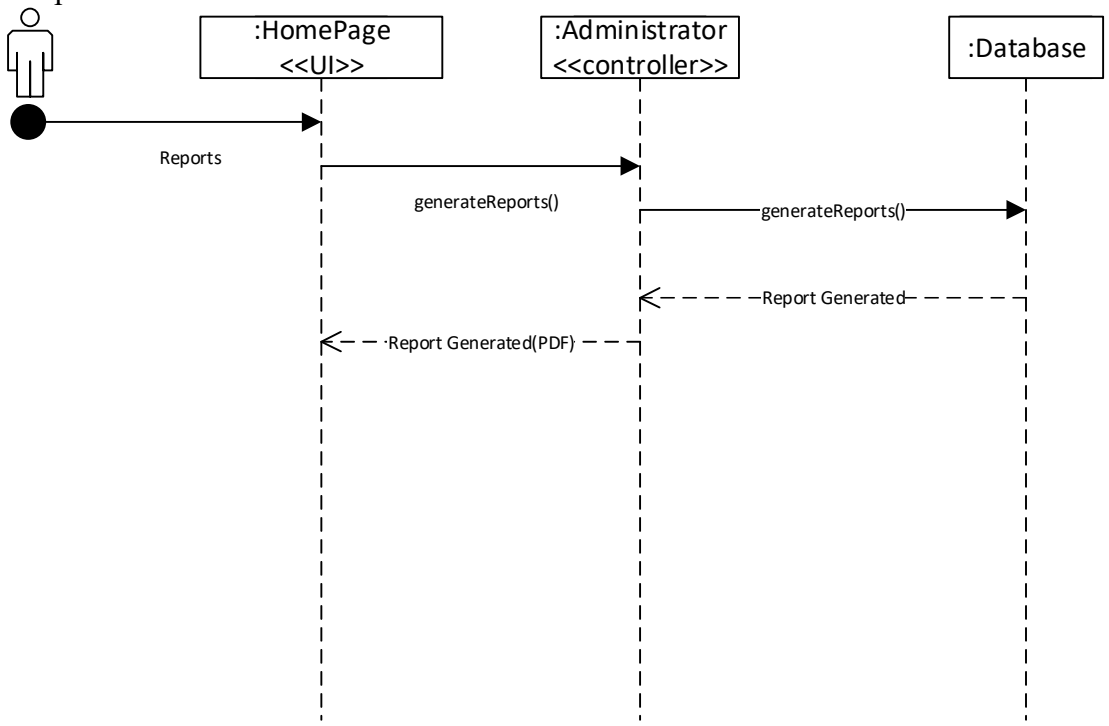
Unsubscribe to newsletter



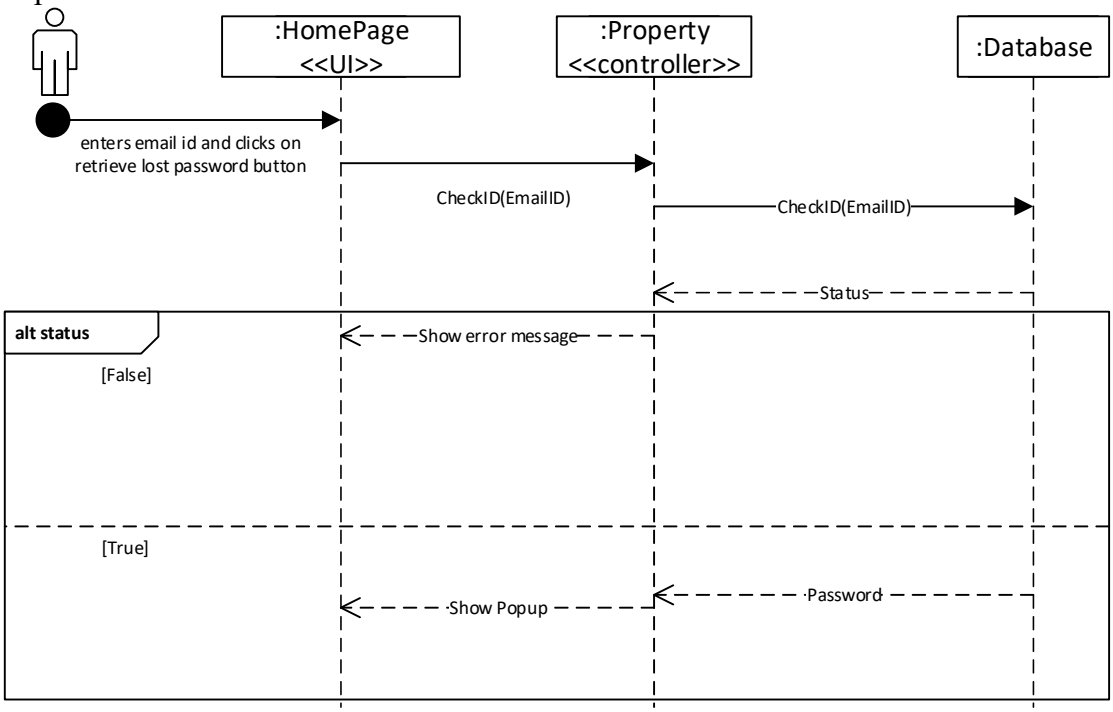
Update property information



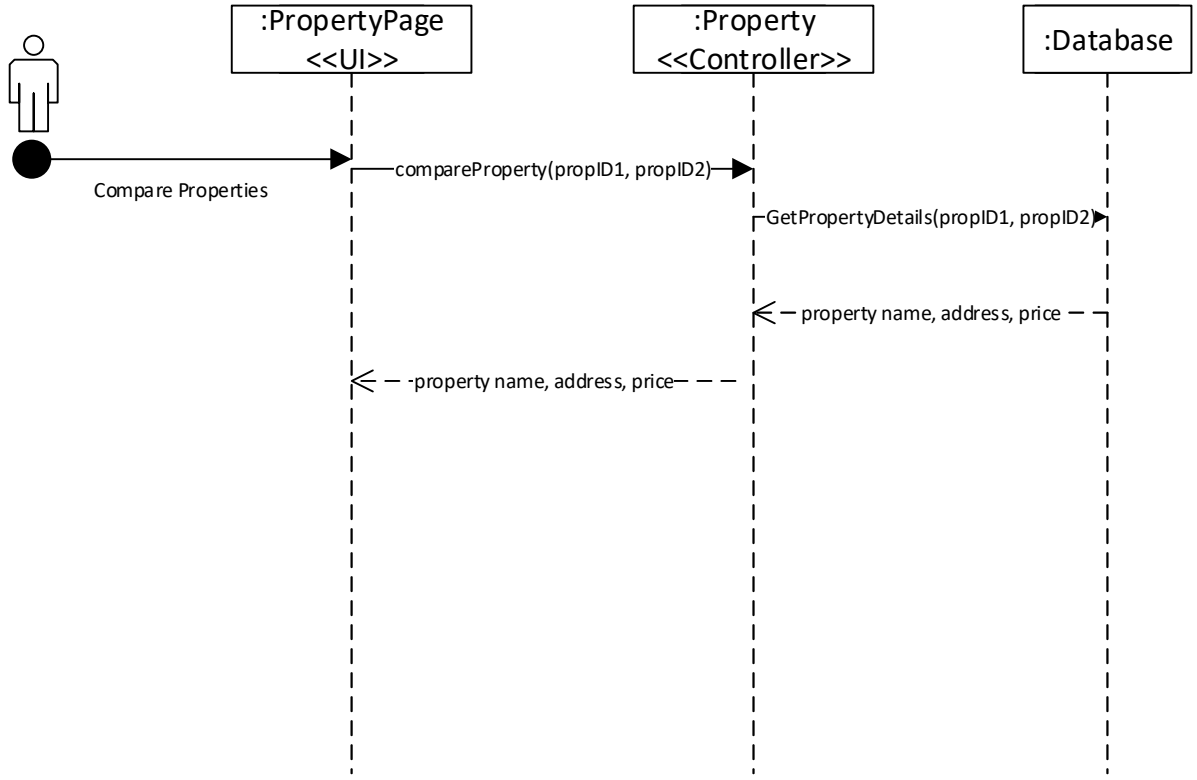
Generating a report



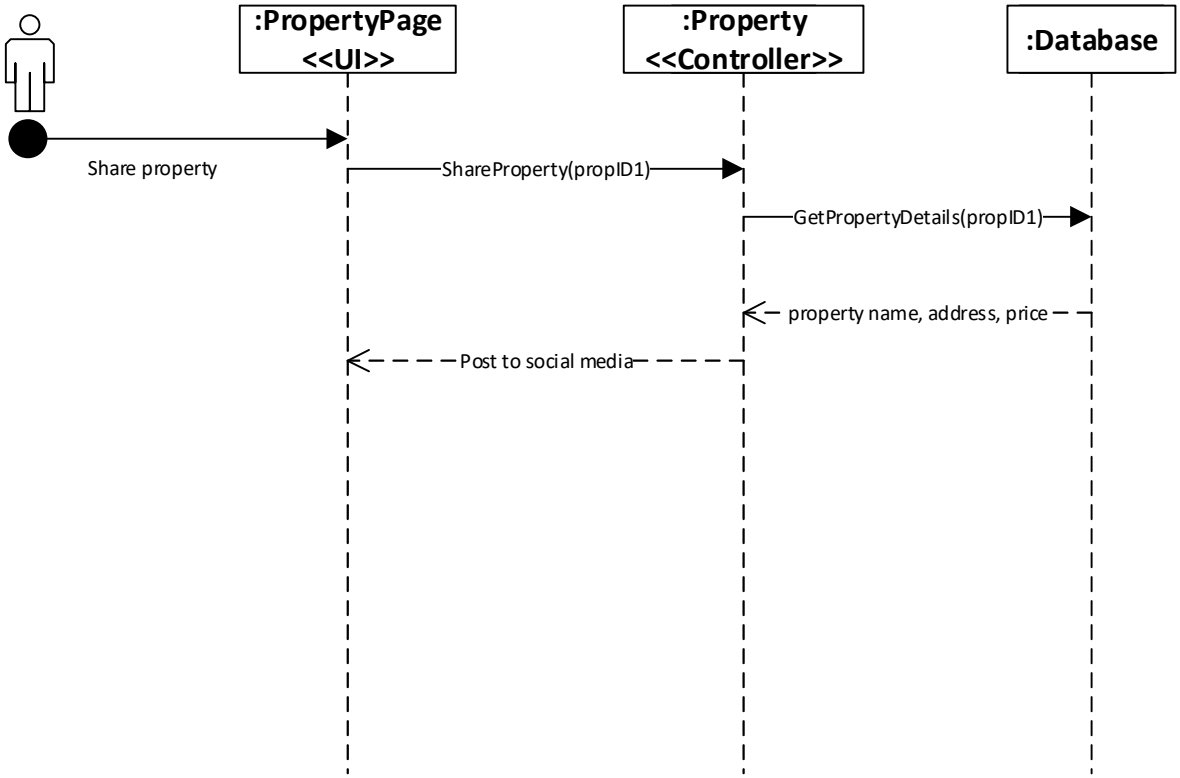
Retrieve lost password



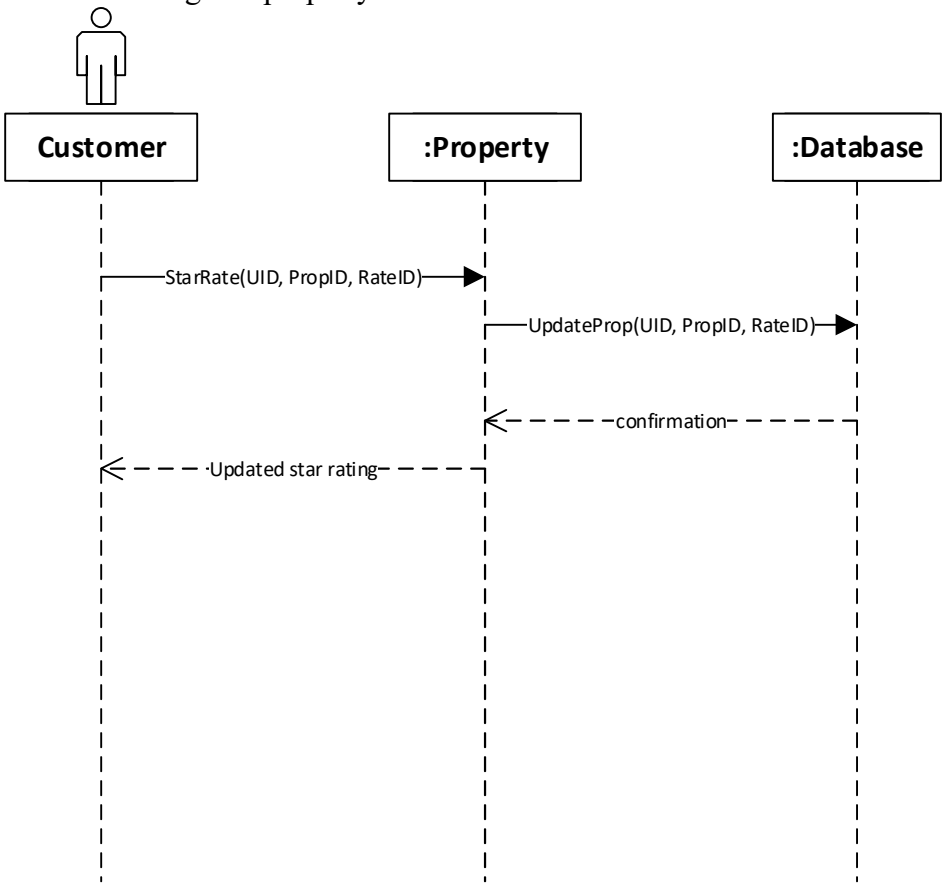
Compare properties



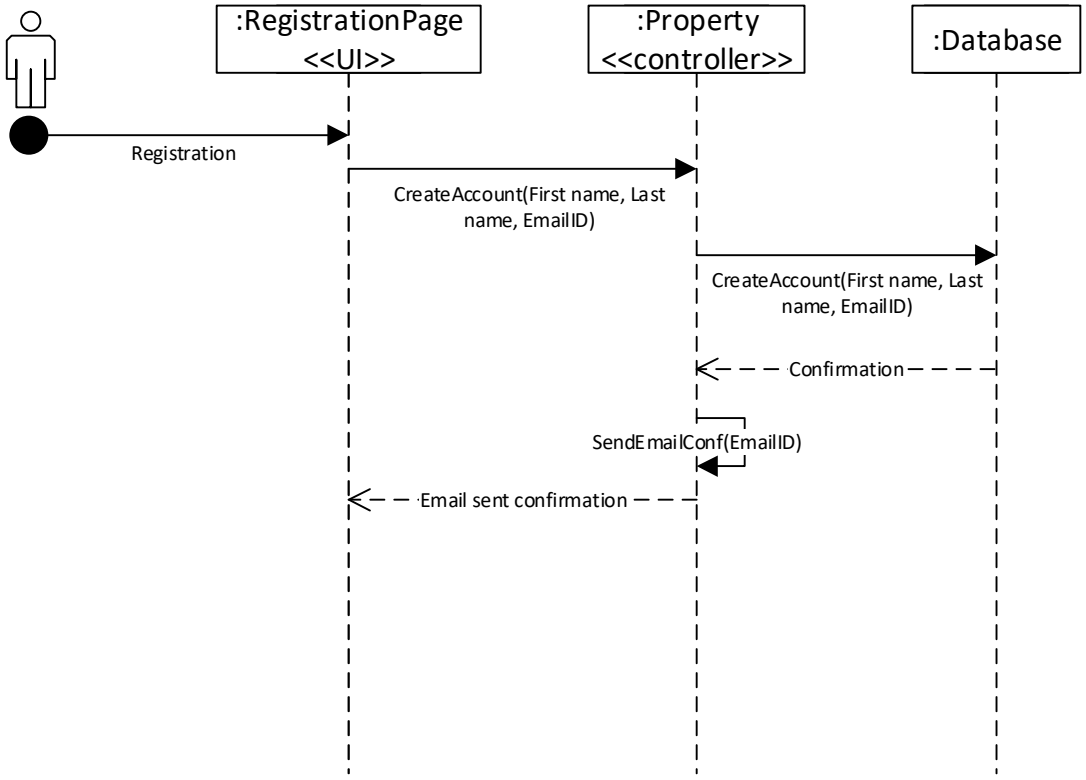
Share on social media



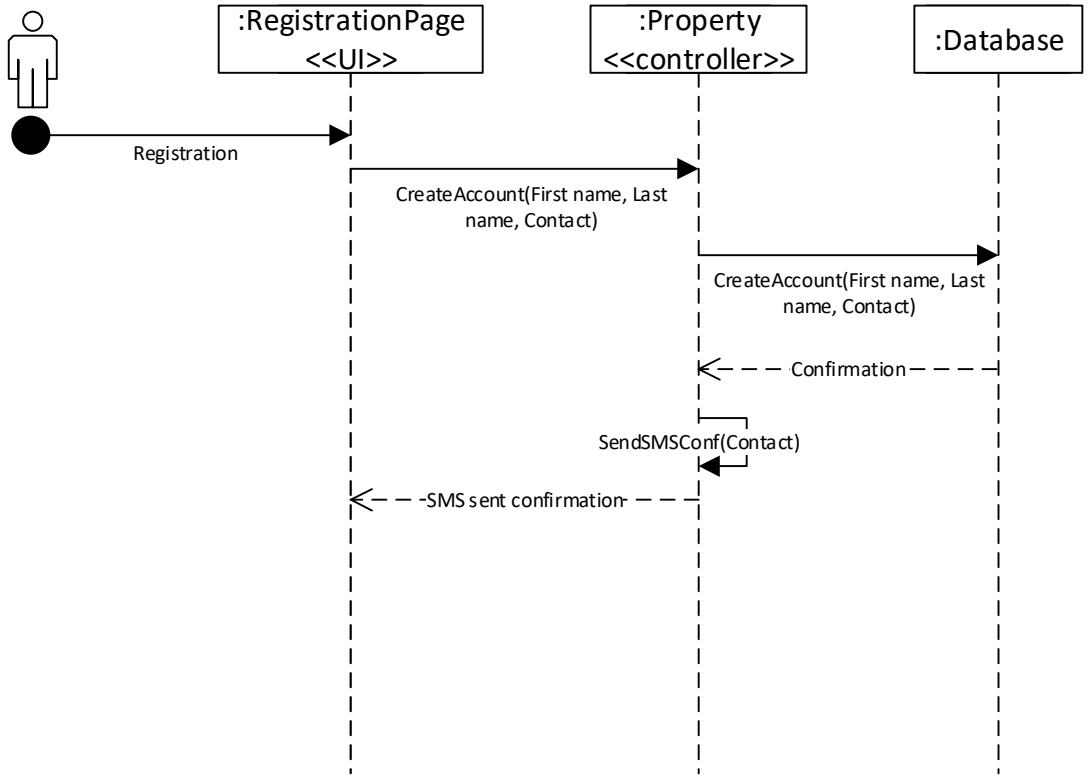
Give star rating to a property



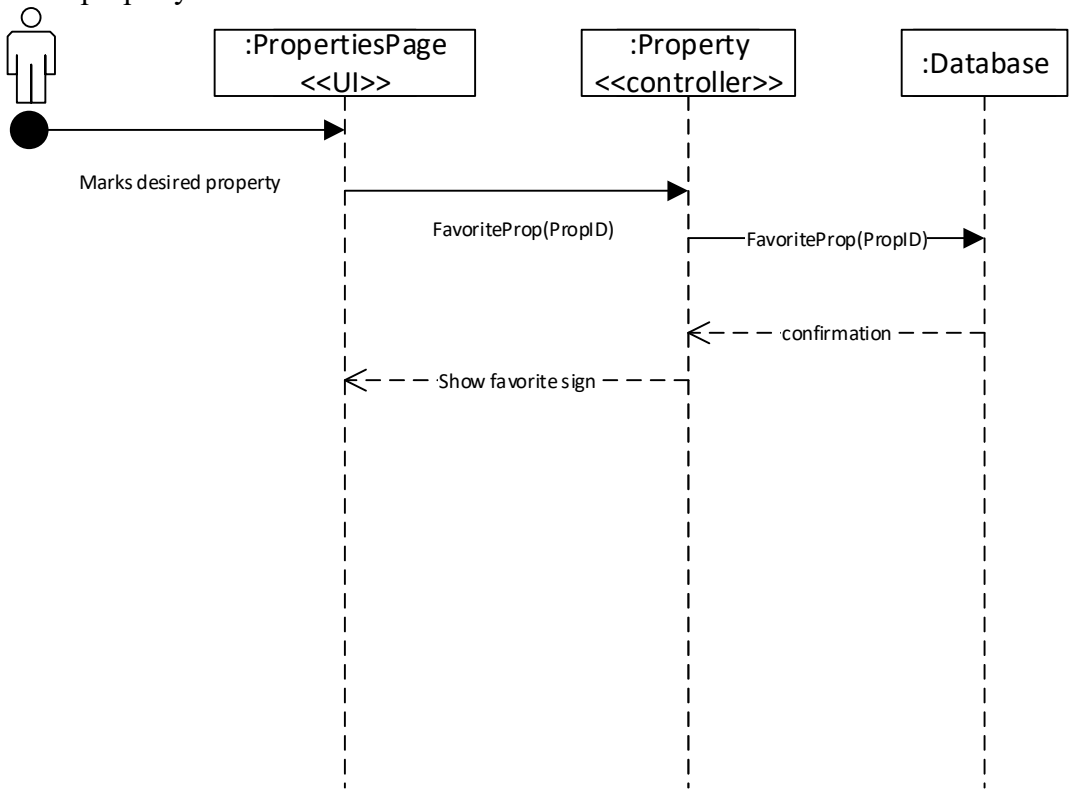
Confirm email registration



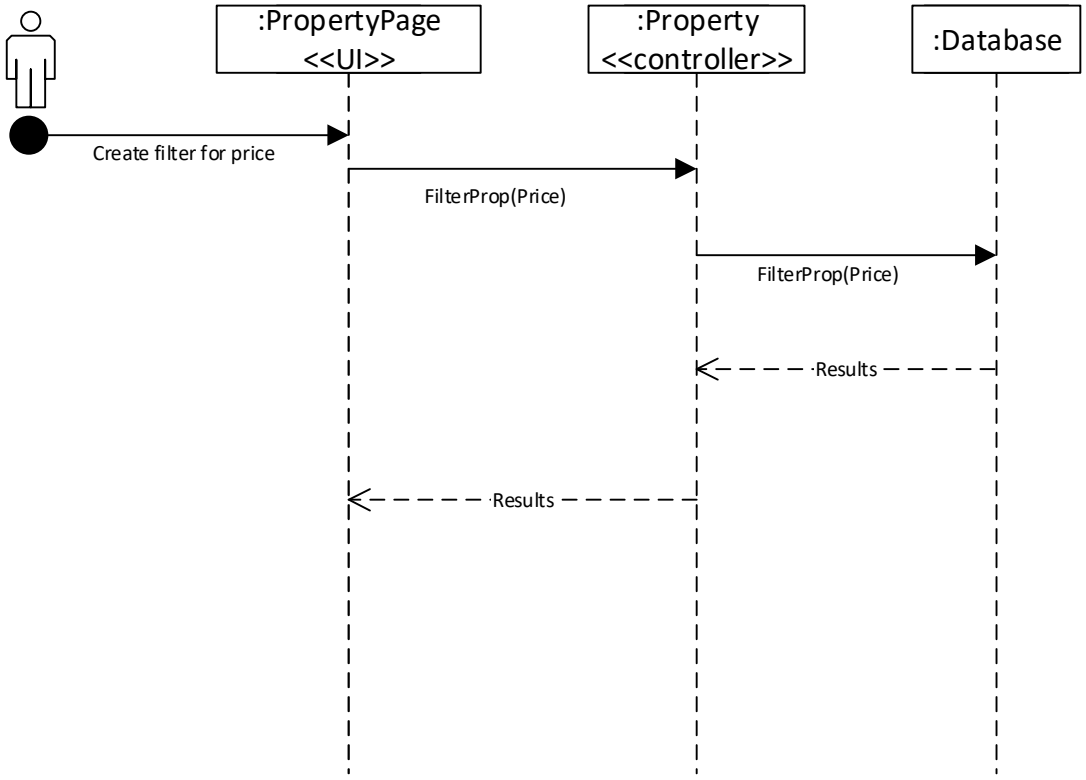
Send SMS


:RegistrationPage
<<UI>>
:Property
<<controller>>
:Database
Registration
CreateAccount(First name, Last name, Contact)
CreateAccount(First name, Last name, Contact)
Confirmation
SendSMSConf(Contact)
SMS sent confirmation

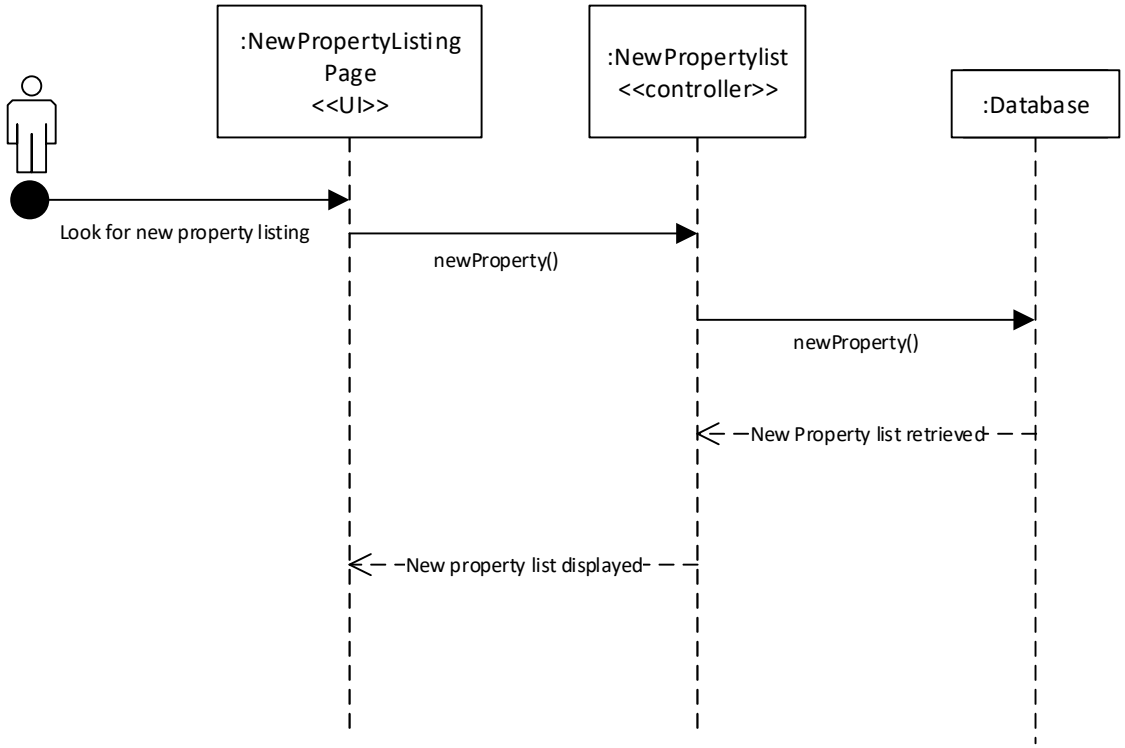
Save or favorite the property



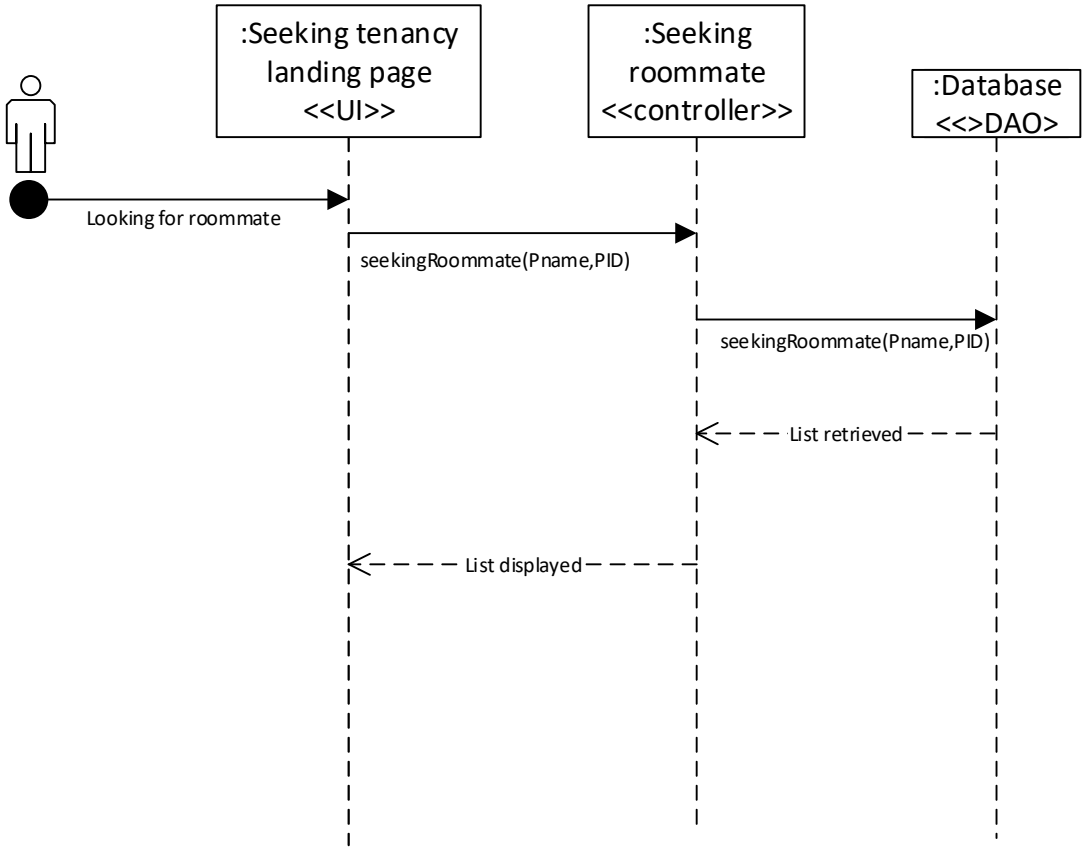
Filter by price



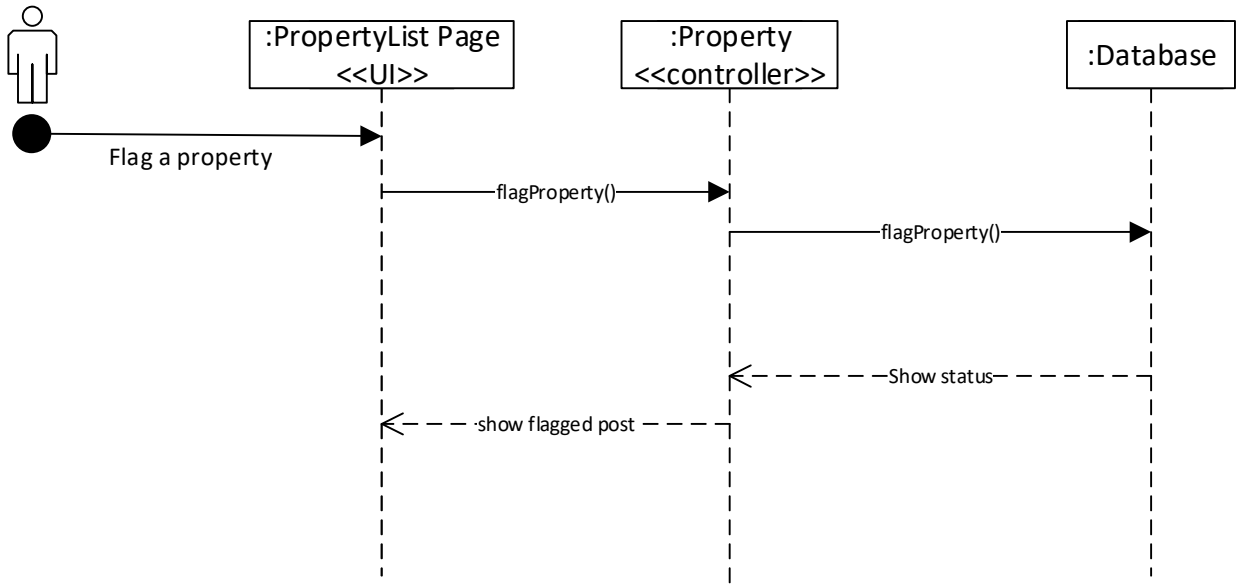
New property list



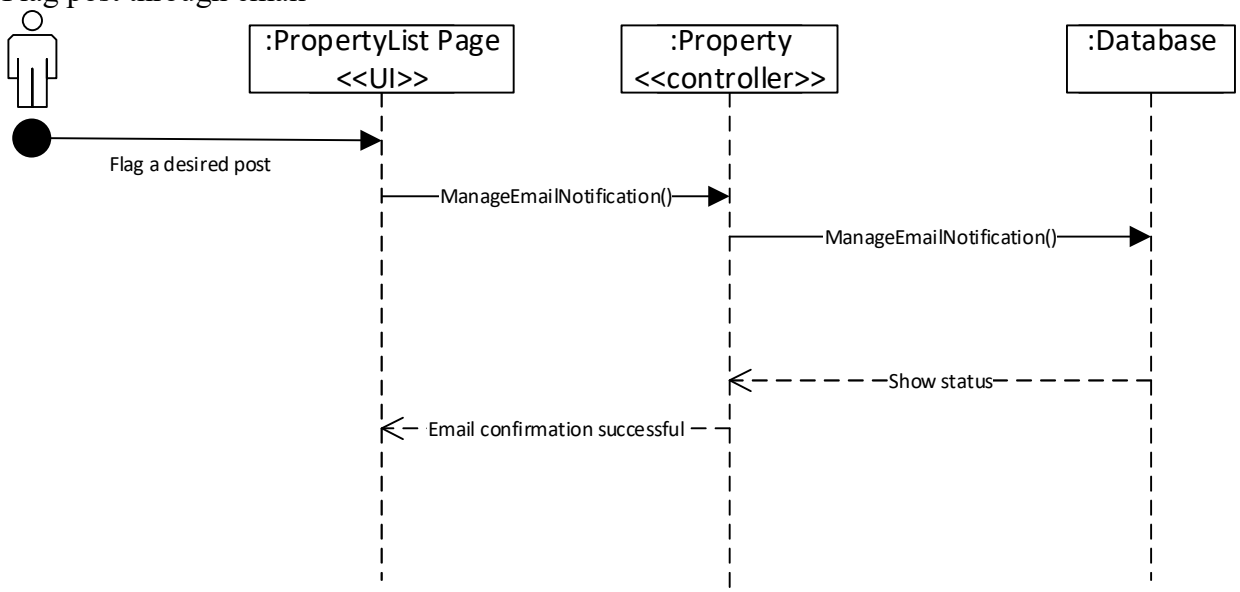
Seek a roommate



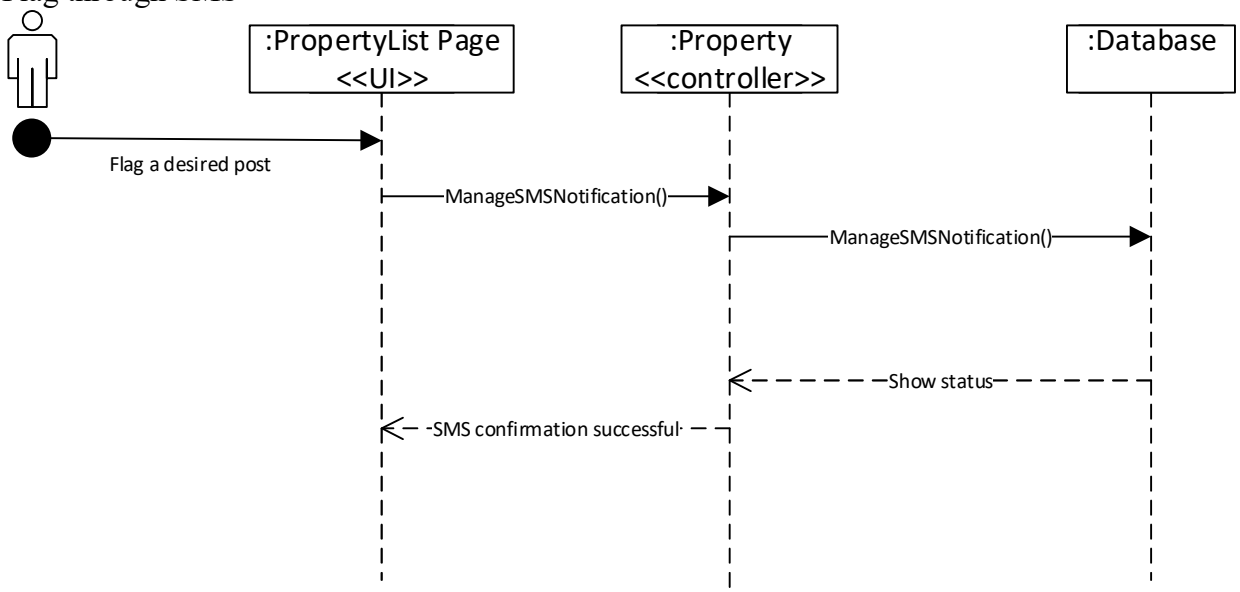
Flag a post by admin



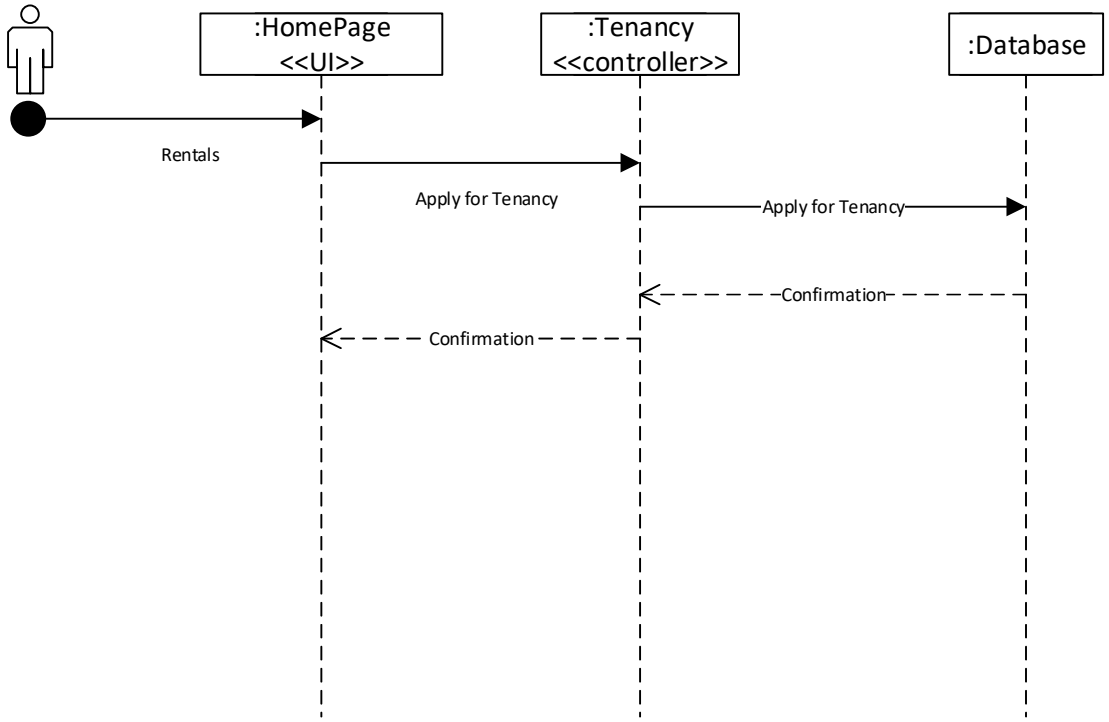
Flag post through email



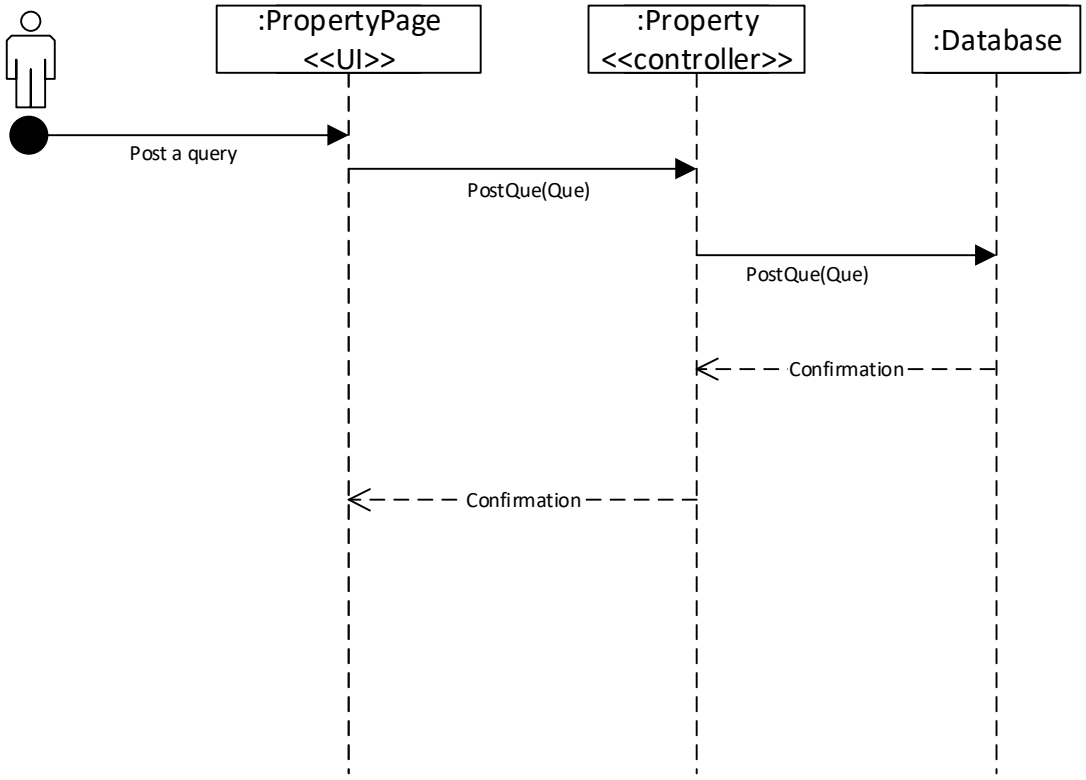
Flag through SMS



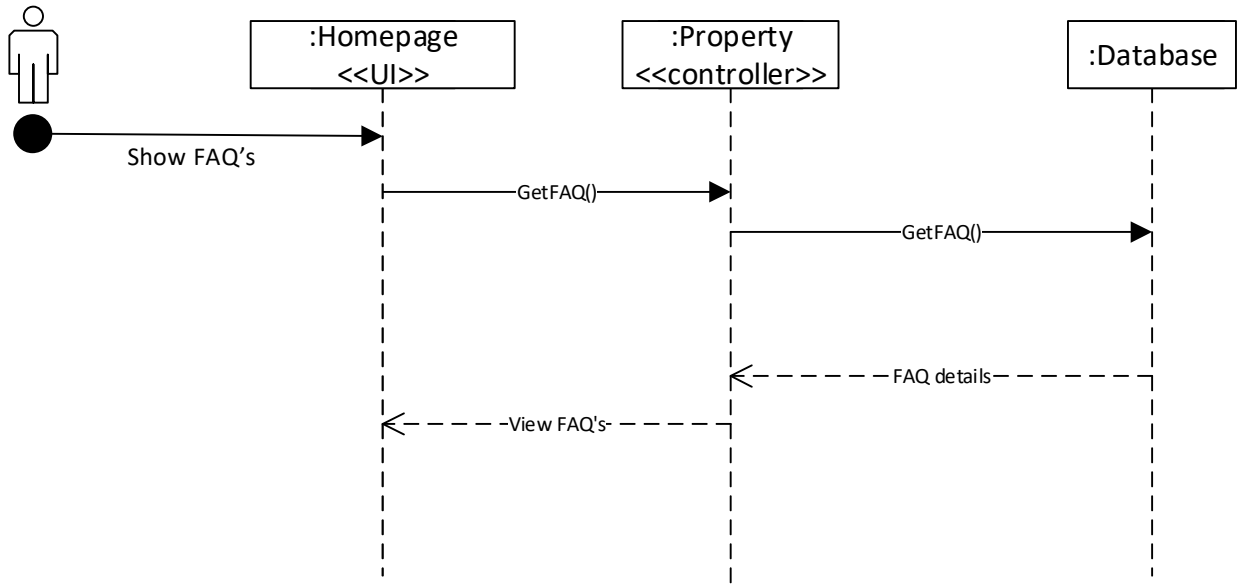
Apply for tenacy



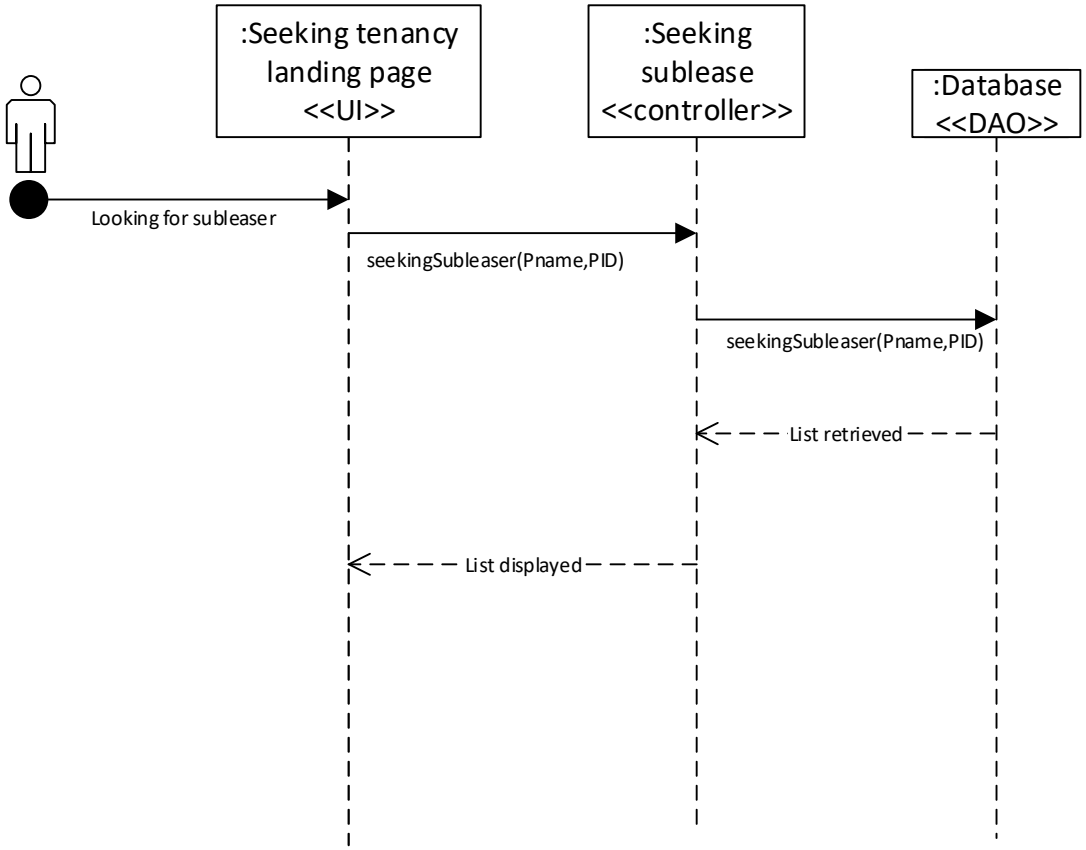
Post a query


:PropertyPage
<<UI>>
:Property
<<controller>>
:Database
Post a query
PostQue(Que)
PostQue(Que)
Confirmation
Confirmation

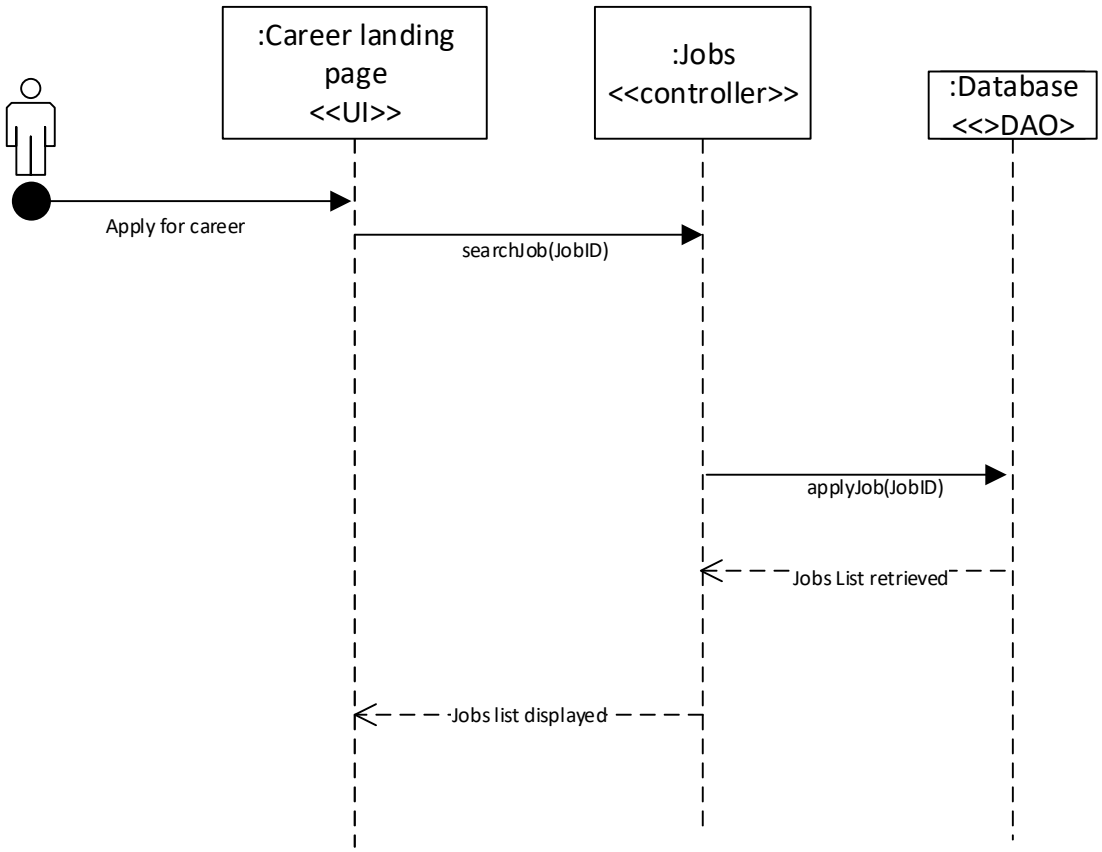
FAQ


:Homepage
<<UI>>
:Property
<<controller>>
:Database
Show FAQ's
GetFAQ()
GetFAQ()
FAQ details
View FAQ's

Seek a sublease



Career employment



4.6.3.5 Other Diagrams

4.6.4 Development

4.6.4.1 Description

4.6.4.2 Standards, Frameworks and Tools

4.6.4.3 Screenshots

References:

[1] https://srs-onlineshopping.blogspot.com/