Software Requirements Specification

For

House Rental Management System

Prepared by

Abrar Hossain Asif

Rahat Uddin Azad

Al Jaber Nishad

Kamruzzaman Shekh

Institute of Information Technology, NSTU

10.23.2019

Table of Contents

List of Figures

List of tables

1 Introduction

This Software Requirement Specifications (SRS) document provides a full overview of **House Rental Management System** by defining the problem statement, scope of the system and purpose.

1.1 Problem Statement

Over the year's landlords/property managers have had a problem in maintaining and managing their customers and their own records. Management has become difficult because of data growth, lack of computerized system and storing records manually. Also, when a tenant wants to rent house it's been very difficult for him to find within time. This system decreases the problems for both tenants and the landlords.

1.2 Purpose

This House Rental Management System is developed to provide the following services:

- **Online House/Room Booking:** A tools through which customers can book available House online prior to their date of using the house instead of walking around and asking for a vacant house.
- **Customer's registration:** A registration portal to hold customer's details, monitor their transaction and used same to offer better and improve services to them and user account where he/she can view her/his details instead of the poor existing systems where only the administrators control their customer details.
- **Rentals Notice and Blog:** A tool where customers can see and view the details of nearby available House for rent/sale, and also view the current economic design of houses.

1.3 Scope

This project traverses a lot of areas ranging from business concept to computing field and required to perform several researches to be able to achieve the project objectives. The following are the scopes of work:

- Developing a smartphone app as around 97% user uses smartphone now a days.
- Studying the existing systems and learning their weakness hence developing a new system to cater for the challenges the local and world domains faces when dealing with house rental issues.

1.4 Glossary

This subsection contains definitions of all the terms and abbreviations used in the document.

i. MB – Megabytes
ii. UI – User Interface
iii. SRS – Software Requirement Specifications
iv. API – Application Program Interface
v. XML – Extensible Markup Language
vi. RESTful – Representational State Transfer
vii. HTML – Hyper Text Markup Language

1.5 References

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

1.6 Overview

Roam around to rent a house has always been a hassle for people. Especially, on recent times, people have so many priorities based on which they have to rent their house. Some people want their house to be in the commercial space, or some want in a chaos free space. Some people prefer to choose the area of their house relating the religion they belong. Again, there are a lot of people who love pets; therefore, they want a house which has pet allowance. Basically, in this era of modernism people want to rent their house like online shopping. To rent a house in physical world has become less popular now a days. No one wants to roam around here and there to search for a house. People would prefer a virtual system to rent a house. In general youths face much bigger problems while renting a house. They don't give house on rent to bachelors and lot of rules and restrictions by the landlord.

To decode this situation and to represent a hassle-free environment to the people a dynamic system can be implemented. We represent a home rental system which provides every needed facility the primary focus of our work is to implement constraint satisfaction problem in the search option of our home rental system. Our home rental system will have dynamic values for searching and for which constraint satisfaction problem implementation is a better preference. Not only this but also a combination of public transport tracker and a dynamic chat server between admin and agent is also a feature of this rental system. Additionally, there is a dynamic mail alert system in which if any user put any request in the wish list, they will be notified by this system.

2 User Classes and Characteristics

There are 3 types of stakeholders in our House Rental Management System. Such as-

i. **Police Officers:** Police Officers will get a same version of the application but the authorization will be different. They are the higher authority of the system. They can see and collect the tenant information like NID card number, number of people and their name. They can also get landlords information like there house information, how many people lives in a house. This module minimizes the problem of taking tenants information for a house. So, if they need any people's information like where they lived, they can get the information easily by the house rental management system.

ii. **Tenants:** Tenants also use the same application but they can only see the house information and the contact number of the landlords. From all the stakeholders, tenants have a limited access like they can only see house ads and can bookmark them. But this limited access gives them a big facility to look many houses within a short time. They can search houses by their own criteria like flat size, rent per month, specific location. An ad contains the landlords contact information which let them connect to the landlords and get more information about the house. They can also give reviews for the house where they live. Also, they can see review when renting a new house or flat. This gives them little idea about that house environment.

iii. **Landlords:** Landlords are the main stakeholder for the system. Here they can create ads for their house and add the information of a tenant who lives in their house. For giving ads, they have to pay specific amount of money and by removing ads they will get back their money if ads time not expired. They have to add tenant's information whenever some people rent their houses or flats. This will help the police officers to get access the information of tenants.

3 Design and Implementation Constraints

Design and implementation constraints are those that we have used to implement this project make successful. It also describes tool that enables developers and testers to view and interact with the user interface (UI) elements of this application.

3.1 Programming Language

We know that android is the most popular mobile app platform with largest number of devices running on this. It's already a common knowledge that android is based upon Java. Naturally Java being the core of android is likely to be the most preferred language for aspiring android developers. Java is also platform independent; it means that Java code can run on any platform. It doesn't require the source code on that machine at which platform, where it will be executed. Java is an object-oriented language which is another big reason for using this language for application development and it is secure because of using the classes and object.

For all of this major advantage of Java, we will use Java as a programming language.

3.2 XML

XML stands for Extensible Markup Language, which gives us a clue to what it does. A markup language like XML, HTML is slightly different from a programming language. Programming languages create dynamic interactions, here as markup languages generally handle things like static user interfaces. XML uses XML elements or tags to define document structure. Basically, XML is used for our layout designing. All the UI and layout of our app will be designed using xml. Unlike Java (which is Back Bone of your app), xml helps us to design our app, how it will look, how components like buttons, textview etc. will be placed and their styling.

3.3 Database Server

MySQL is a free-to-use, open-source database that facilitates effective management of databases by connecting them to the software. That’s why we will use MySQL database server to store all of the information of this system. It is a stable, reliable and powerful solution with advanced features like the following:

- Data Security
- On-Demand Scalability
- High Performance
- Round-the-clock Uptime
- Comprehensive Transactional Support
- Complete Workflow Control
- The Flexibility of Open Source
- Fault tolerance

3.3.1 Web Server

A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Dedicated computers and appliances may be referred to as Web servers as well. We will use the Apache HTTP server to implement this project. We will use RESTful API to retrieve data from our server to mobile application. RESTful stands for Representational State Transfer. And API stands for Application Programming Interface.

4 Requirement Specification

All the requirements based on elicitation process is described in this section.

4.1 Functional Requirements

Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the system should perform, the processes involved, which data should the system holds and the interfaces with the user.

4.1.1 Search ads by specific criteria

FR-1	Search house ads based on specific criteria
Description	Stakeholders can search ads like search by flat size, rent per month and specific location. Also, they can search by ads name. If there are no ads found based on their criteria, system will show related ads on that criteria.
Stakeholders	Tenants, Landlords
Priority	High

4.1.2 Search ads by location

FR-2	Search house ads based on location
Description	Tenants only needs to click on an option “House nearby” and system at first detect the stakeholder’s position then based on that location it will list all the available houses near that area if any exists.
Stakeholders	Tenants, Landlords

Priority	Medium

4.1.3 Stakeholders can remember ads

FR-3	Stakeholders will bookmark ads to remember them
Description	If tenants found any desired ads he wants to see later, he can bookmark that ads and see the ads in favorite section.
Stakeholders	Tenants, Landlords
Priority	Medium

4.1.4 Stakeholders wants to get notified

FR-4	Stakeholders wants notification when a house is available for rent nearby his location
Description	While stakeholders want to notified when there a house nearby him or on a specific area. For getting notified stakeholders need to enable their notification. And whenever there is a house available for rent stakeholders will be notified.
Stakeholders	Tenants, Landlords
Priority	Low

4.1.5 Rent a house from the available ads

FR-5	Stakeholders wants to rent a house
Description	Stakeholders have to be a valid user of the system. Then he can see all the available ads added in the system. From the given ads list he can choose his favorable one and if everything looks good about the house to him, then he can contact with the house landlord.
Stakeholders	Tenants, Landlords
Priority	High

4.1.6 Communicate with landlords for renting a house

FR-6	Stakeholders can communicate with landlords
Description	After observing house ads, if stakeholders found a vacant house for him, he can contact through SMS, E-Mail, and Phone call with landlords. Phone number, E-Mail account will be placed in ads information.
Stakeholders	Tenants
Priority	High

4.1.7 Ranking for houses

FR-7	Tenants wants to rate a house
Description	For giving rating to a house tenant needs to be live at that house before or now. And select the house and give a rating to the house. The rating is between five stars maximum to no star minimum.
Stakeholders	Tenants
Priority	Low

4.1.8 Add new ads to the system

FR-8	Landlords wants to give ads for his house
Description	Landlords will upload his house info like rent per month, gas and water bill, picture of that house, deposit paid, terms and conditions to follow acceptance, how many days he wants to show the ads. And new ads are created based on the given information.
Stakeholders	Landlords
Priority	High

4.1.9 Boost existing ads

FR-9	Landlords wants to boost his ads

Description	Landlords can boost the ads when they are creating ads. For boosting the ads, they have to choose the specific payment credential and need to pay the exact amount of money. They also can boost existing ads doing the above process.
Stakeholders	Landlords
Priority	High

4.1.10 Remove ads from the system

FR-10	Landlords wants to Remove the ads from the system
Description	As landlords house get rented so they don't want to show the ads. For so they have to go to the ads and select it and then remove. When removing if they selected for 30 day to show the ads and removing it at 5th day of showing then they will get money for other 25 day.
Stakeholders	Landlords
Priority	High

4.1.11 Store tenant's information for a house

FR-11	Landlords can add and store tenant's information
Description	Landlords have to add tenant's information like tenant's National ID card information, phone number, permanent address, tenant's picture.
Stakeholders	Landlords
Priority	High

4.1.12 Access tenants and landlord's information

FR-12	Police Officers can access information from system
Description	Police officers has ability to access tenants and landlord's information like National ID card information, contact number.
Stakeholders	Police Officers
Priority	High

4.2 Data Requirements

For our application we have to store many information-like pictures from landlord's house, tenants picture this information needs to be stored in the system. For storing pictures system database needs to be limited.

4.2.1 Storing image data

DR-1	Stakeholders have to provide specific sized picture for the house
Description	Landlords will upload their ads information, house information and tenant's information. Also, images from the houses needs to be within 3 MB.
Stakeholders	Landlords
Priority	High

4.3 Performance Requirements

It is important to maintain performance of the software system. To ensure performance we maintain these steps:

4.3.1 Speed and Latency Requirements

PR-1	Faster searching houses and load ads quickly
Description	Loading the ads information will be faster and can load it within seconds. And any search result will show up within 1 seconds and images of the house will load within 5 seconds if the network speed is good.
Stakeholders	Landlords, Tenants
Priority	Medium

4.3.2 Precision and Accuracy Requirements

There are no precision and accuracy requirements for our project.

4.3.3 Capacity Requirements

This system can load up to thousands of tenant's information and thousands of ads information.

4.4 Dependability Requirements

If House Rental Management system can provide availability, reliability, safety, security then this system will be dependable.

4.4.1 Availability and Reliability Requirements

This system will be available for 24 hours. Stakeholders can use the system anytime they wanted and can see ads or rate a house and upload information which is very reliable for stakeholders.

4.4.2 Safety Requirements

This system will not contain any malware and this will not harm any stakeholder's device.

4.5 Maintainability and Supportability Requirements

It's very important to provide service to the end users.

4.5.1 Maintainability Requirements

MR-1	Update information
Description	It is very important to update tenant information as tenants will live various houses.
Stakeholders	Landlords
Priority	Low

4.5.2 Supportability Requirements

This system meets Testability, Maintainability, Compatibility, Configurability, Serviceability, install ability which are related to supportability requirements.

4.6 Security Requirements

Securing information is much more important for a system to get users dependability. Here is some of them:

4.6.1 Access Requirements

For accessing information, the system will use some authorization techniques to ensure that correct data is used by correct user.

4.6.2 Integrity Requirements

Integrity requirements refers to a security system which ensures an expectation of data quality. It also ensures that all data of the system would never be exposed to the malicious modification or accidental destruction. For preventing anonymous access to user password, the system will use encryption technique called Hash Function for encrypting user password.

4.6.3 Privacy Requirements

Privacy requirements enhances to protect stakeholder's privacy. In this way, all data or a partial part of data are going to be disclosed according to system's privacy policy. To ensure privacy, the central database should be protected by the anonymous. Users are permitted to get access to those data which are being associated by them which can be ensured by the user log in system.

4.7 Usability and Human-Interaction Requirements

This system will provide more user-friendly environment.

4.7.1 Ease of Use Requirements

Our system will be easier to use by any type of people and they don't need any training to use the system.

4.7.2 Personalization and Internationalization Requirements

There are no personalization and internationalization requirements in our system.

4.8 Look and Feel Requirements

Look and feel requirements mainly refers how the system will look like.

4.8.1 Appearance Requirements

AR-1	Text color and font
Description	All texts and description will be at a good font size so that users can understand what is important and mandatory input fields will be kept red colored until user put correct information.
Stakeholders	Tenants, Landlords, Police Officers
Priority	High

4.8.2 Style Requirements

There are no style requirements in our system.

4.9 Operational and Environmental Requirements

Operational and environmental requirement refers to the capabilities, performance measurements, process, measurements of effectiveness, measurements of performance, measures of sustainability, measurements of technical performances etc.

4.9.1 Expected Physical Requirements

There are no expected physical requirements in our system.

4.9.2 Requirements for Interfacing with Adjacent Systems

There are no requirements for interfacing with adjacent system for our project.

4.9.3 Release Requirements

There are no specific release requirements in our system.

4.10 Legal Requirements

Legal requirements normally refer to the terms and conditions or privacy policy of any organizations. The terms and condition of our application is that, no third-party software or person are allowed to engage to use our data for their business purpose.

5 Requirement Engineering Process

Requirements engineering refers to the process of defining, documenting and maintaining requirements in the engineering design process. It is a common role in systems engineering and software engineering.

5.1 Requirement Elicitation Techniques

Requirement Elicitation and Analysis is the process of interacting with customers and end-users to find out about the domain requirements, what services the system should provide, and the other constraints. We mainly use these techniques for gathering Requirement:

- Interview
- Questionnaire
- Existing System

5.1.1 Interview

Interview is a good technique to investigate issues in-depth, to discover how people think and feel about certain topics. We hold interviews that can be performed with a small group of stakeholders. We mainly perform our interview based on some specific criteria.

- Short description about project (Goals and objectives)
- Registration process
- Ways of search a house
- Security
- Availability
- Advertisement of a house
- Boosting

5.1.2 Questionnaire

Questionnaire is a useful technique to investigate trends, shifts in user attitudes and opinion, user satisfaction with priorities and preferences. Two types of questions set created for proper survey. One for tenants, and one for landlords. Both questionnaires are as short as possible so respondent don’t get bore or get frustrate. Have an underlying reason for every question and group topic areas together to keep the respondent focused. The main advantage behind this survey is responses are gathered in a standardized way. Information can be collected in short period of time from a large number of people, often geographically dispersed.

5.1.3 Perform Existing system Analysis

Existing system Analysis can help reveal how systems currently work or what they are supposed to do. Analysis includes any written information about current systems, business processes, requirements specifications, competitor research. Reviewing and analyzing can help identify functionality that needs to remain, functionality that isn't used. Currently the most property managers manage property and tenants' details on papers. After existing system analysis, we found many several problems on existing system.

Problems of existing system:

• With the current system recording the details of various activities of user is completely manual and entails a lot of paper work.

• The existing system only provides text-based interface which is not as user friendly as Graphical user interface

• The transactions are not secure as papers may get lost or damaged.

6 Use Case Diagram

Use case diagram comprises actors and use cases, where actors perform several cases or one. This also shows which actors have access to which use case. Here is the use case diagram for house management system.

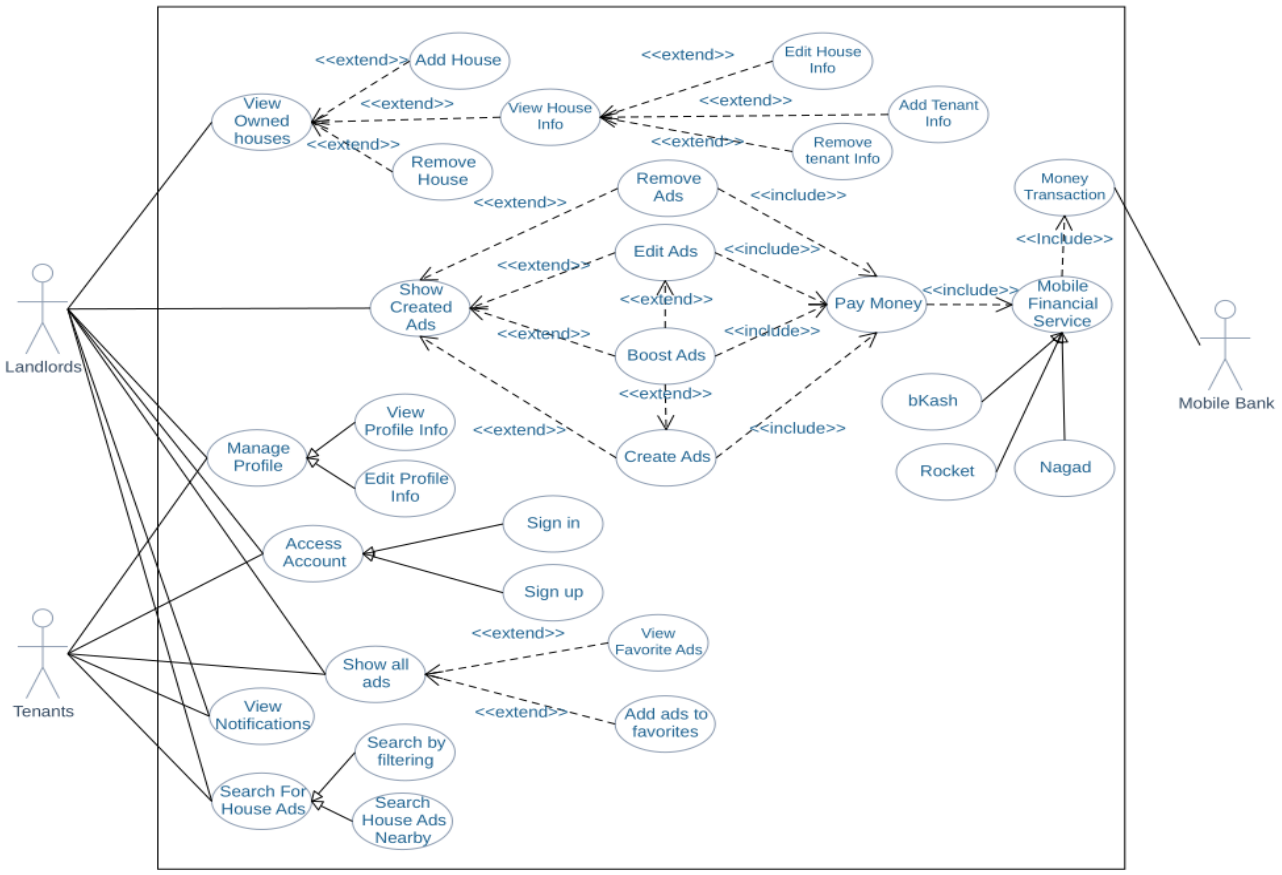

Figure 6.1: Use case diagram

7 Use Case Description

All use cases from use case diagram are explained here.

Table 7.1: Access account

Use Case 1	Access account	
Goal	Stakeholders wants to sign in or sign up to the system.	
Preconditions		
Success End Condition	Stakeholders can access his old account or create new account.	
Failed End Condition	Stakeholders don't have access to account.	
Primary Actors: Secondary Actors:	Landlords, Tenants	
Trigger	Access account request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to access account.
	2	System will show sign in option and sign up option.
Alternative Flows	**Step**	**Branching Action**
		Not applicable.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.2: Sign up

Use Case 2	Sign up	
Goal	Stakeholders will use our system services and be a part to the system.	
Preconditions		
Success End Condition	Stakeholders will be a user of the system. Can use many functionalities based on user access.	
Failed End Condition	Stakeholders won't be a user of the system. Don't have access to the system functionalities.	
Primary Actors: **Secondary Actors:**	Tenant, Owner.	
Trigger	Sign up request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested for sign up
	2	Stakeholders will choose user type form like tenant or landlord.
	3	Stakeholders then need to fill preferred information in the form. For tenants they have to provide their name, mobile number, unique username, password and email. For landlords they have to provide their name, mobile number, unique username, password, house address and email.
	4	After filling all the information correctly system will show the submit button. And stakeholders can click and submit the information to the system.
	5	System will now collect the information of the user.
	6	For confirming the user information, system will send a confirmation code to the user mobile number.
	7	A popup window will appear for entering the confirmation code.
	8	User have to write the code at the popup window and click submit.
	9	System will then check and verify the confirmation code.
	10	System will then show the ads information page as default.
Alternative Flows	**Step**	**Branching Action**
	3a	System will check the username while stakeholders typing on that field and show whether the username is available or not.
	7a	Stakeholders can resend the confirmation code again to their mobile phone.
	10a	If the confirmation code is not valid then system will again ask for the confirmation code.
Quality Requirements	**Step**	**Requirement**
	7	User have to confirm the code within 2 minutes. After this time the confirmation code will be invalid.

Table 7.3: Sign in

Use Case 3	Sign in	
Goal	Stakeholders wants to sign in and use the system functionalities.	
Preconditions	Stakeholders is a user of the system.	
Success End Condition	Showing a page with ads information.	
Failed End Condition	Not a user or the username or password is invalid. Stay on the login page.	
Primary Actors: Secondary Actors:	Tenant, owner	
Trigger	Sign in request	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested for sign in.
	2	Stakeholders enter their username and password.
	3	Stakeholders will click sign in.
	4	System will check the username and password and show the ads information page.
Alternative Flows	**Step**	**Branching Action**
	3a	Stakeholders forget their password.
	3a1	Stakeholders click on forget password.
	3a2	System will prompt for entering stakeholders' mobile number and username.
	3a3	System then check the mobile number whether the current device has the given mobile number. If not, it again prompts the user to enter the valid mobile number.
	3a4	If the mobile number is valid system then sends a confirmation code to that mobile number.
	3a5	System then popup for entering the confirmation code.
	3a6	System will check the confirmation code if it's wrong it again prompt for entering the confirmation code.
	3a7	If the confirmation code is correct system then prompt the user to enter the new password for his username.
	3a8	System now again prompt the user to login with the new username and password.
	4a	If username and password does not match then system will prompt stakeholders to reenter the password.
Quality Requirements	**Step**	**Requirement**
	3a5	User have to confirm the code within 2 minutes. After this time the confirmation code will be invalid.

Table 7.4: View owned houses

Use Case 4	View owned houses	
Goal	Landlords wants to see his houses which is added before in the system.	
Preconditions	Landlords are signed in.	
Success End Condition	The system displays landlords owned houses.	
Failed End Condition	No owned house and house information still added.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	View owned houses request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to see his owned houses list.
	2	System will collect landlords house information by his name and UserID and validate the information.
	3	System will show landlords all added or owned houses as list and each house as house name, house holding number, house address and how many tenants lived in that house.
Alternative Flows	**Step**	**Branching Action**
		Not alternative flow.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.5: Add house

Use Case 5	Add house	
Goal	Landlords wants to add his owned house information in the system.	
Preconditions	Landlords are signed in.	
Success End Condition	The system will add house at owned house list.	
Failed End Condition	House will not be added in owned house list.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Add house request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to add house.
	2	Landlords needs to fill a form by adding house name, house address, number of flats, number of units in each flat, house holding number.
	3	System will then validate the house holding number.
	4	System then add the house as landlords owned house and show a notification message that the house information is added successfully.
	5	The house will then show in the houses list.
Alternative Flows	**Step**	**Branching Action**
	3a	System will prompt that invalid house holding number entered and ask landlords to enter a valid house holding number.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.6: Remove House

Use Case 6	Remove house	
Goal	Landlords wants to remove his old house from owned houses list	
Preconditions	Landlords are signed in. Landlords have owned houses. Landlords select a house.	
Success End Condition	The system will remove the selected house and its information from owned houses list.	
Failed End Condition	No owned house and house information will be deleted.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Delete owned house request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to delete his owned house.
	2	System will prompt for confirm the deletion of the house.
	3	Landlords confirm the deletion.
	4	System will collect landlords house holding number from the selected house.
	5	System then delete the house information for landlords.
Alternative Flows	**Step**	**Branching Action**
	2a	Landlords cancel the deletion.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.7: View House Info

Use Case 7	View House Info	
Goal	Landlords wants to see his house information.	
Preconditions	Landlords are signed in. Landlords have owned houses. Landlords select a house.	
Success End Condition	The system will show the selected house information from owned houses list.	
Failed End Condition	No owned house information will be shown.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	View house information request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to show information of his owned house.
	2	System will collect landlords house holding number from the selected house.
	3	System will show the house information like house name, house address, number of flats, number of units in each flat, house holding number and how many tenants live in the house.
	4	After getting reply from database server, system will show a view with that information collected from database server.
Alternative Flows	**Step**	**Branching Action**
		No alternative flow.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.8: Edit House Information

Use Case 8	Edit house information	
Goal	Landlords wants to edit his house information.	
Preconditions	Landlords are signed in. Landlords have owned houses. Landlords select a house.	
Success End Condition	Landlords will update his house information.	
Failed End Condition	No house information will be updated.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Edit house information request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to edit information of the selected owned house.
	2	System will collect landlords house holding number from the selected house.
	3	System will show selected house information like house holding number, house name, house address, number of flats, number of units in each flat.
	4	Landlords can edit any information shown by the system.
	5	Landlords now can update his edited house information by clicking update.
	6	System will ask landlords to confirm that information will be updated.
	7	System will show notification that house information is updated.
Alternative Flows	**Step**	**Branching Action**
	4a	Landlords choose not to edit any information and cancel to update information.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.9: Add tenant information

Use Case 9	Add tenant information	
Goal	Landlords wants to add tenant information in his house.	
Preconditions	Landlords are signed in. Landlords have owned houses. Landlords select a house. Landlords have tenants in his house.	
Success End Condition	Landlords added tenant information in the selected house.	
Failed End Condition	No tenant information added in the selected house.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Add tenant information request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to add tenant information to his selected owned house.
	2	System will prompt for entering tenants name, tenants' mobile number, tenants NID picture, unit and floor number where he lives.
	3	Landlords have to provide all the information and can click at add tenant.
	4	System will ask to confirm that this information is correct.
	5	After confirmation system will store that information at the selected house as a tenant.
Alternative Flows	**Step**	**Branching Action**
	3a	Click cancel not to provide the information.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.10: Remove tenant information

Use Case 10	Remove tenant information	
Goal	Landlords wants to remove tenant information in his house.	
Preconditions	Landlords are signed in. Landlords have owned houses. Landlords select a house. Landlords have tenants in his house.	
Success End Condition	Landlords removed tenant information in the selected house.	
Failed End Condition	No tenant information removed in the selected house.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Remove tenant information request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to remove tenant information to his selected owned house by.
	2	System will ask to confirm the deletion of tenant information.
	3	Landlords can confirm the deletion.
	4	System then delete the tenant information as house tenant.
	5	System will update the tenant information list.
Alternative Flows	**Step**	**Branching Action**
	2a	System will cancel to delete tenant information.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.11: Show Created Ads

Use Case 11	Show Created Ads	
Goal	Landlords can manage his created ads	
Preconditions	Landlords is signed in.	
Success End Condition	The system displays landlords created ads.	
Failed End Condition	No Ads created.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Show created ads request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to see his created ads list.
	2	System will collect landlords all created ads and will display as ads information list.
Alternative Flows	**Step**	**Branching Action**
	2a	Landlords can remove selected ads. See use case **Remove ads.**
	2b	Landlords can edit selected ads. See use case **Edit ads**.
	2c	Landlords can boost selected ads. See use case **Boost ads**.
	2d	Landlords can create new ads. See use case **Create ads**.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.12: Create Ads

Use Case 12	Create ads	
Goal	Landlord wants to create his own ads.	
Preconditions	Landlords is signed in.	
Success End Condition	Landlords successfully create his house ads.	
Failed End Condition	No new ads created.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Create ads request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested for creating new ads.
	2	System will display a form where landlords have to fill up by giving number of beds, rent per month, gas and water bill, picture of that house, deposit paid, terms and conditions to follow acceptance.
	3	After filling up all the information in the form landlords can submit ads information.
	4	System will request for providing ads duration (how many days ads will show) with a minimum of 3 days and maximum of 2 months.
	5	System will calculate total cost for that ads by multiply per day ads cost with total ads duration and display it below ads duration.
	6	See use case **Pay money**.
	7	System now store the ads information and show newly created ads in created ads list.
Alternative Flows	**Step**	**Branching Action**
	3a	Landlords choose cancel not to submit the information.
	5a	System will show landlords to boost the ad. See use case **Boost Ads**.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.13: Edit ads

Use Case 13	Edit ads	
Goal	Landlords wants to edit his created ads.	
Preconditions	Landlords is signed in. Landlords has created ads. Landlords selected an ad.	
Success End Condition	Landlords successfully update his created ads.	
Failed End Condition	Landlords failed to edit his created ads.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Edit ads request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to edit selected ad.
	2	System will display all information about that ads.
	3	Now landlords can modify any information about that ad which he wants to.
	4	After filling up all the information in the form landlords can submit ads information.
	5	System will check landlords selected ad remaining time duration then ask landlord whether he wants to extend the time.
	6	Landlords can extend the time duration.
	7	System will calculate total cost for that ads by multiply per day ads cost with total ads duration and display it below ads duration.
	8	See use case **Pay money**.
	9	System store the changed information and update the ads information list.
Alternative Flows	**Step**	**Branching Action**
	4a	Landlords choose cancel to update the information.
	6a	Landlords don’t extend the time duration. System then store the changed information.
	6b	If ad is already boosted system will show the remaining boosted time.
	6c	If ads remaining time is zero days then system will prompt landlords to extend the time for the ads.
	6c1	If landlord won’t extend the time duration then system won’t update the ads information
	7a	System will show landlords to boost the ad if it is not boosted. See use case **Boost Ads**.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.14: Remove ads

Use Case 14	Remove ads	
Goal	Landlords wants to remove his created ads.	
Preconditions	Landlords is signed in. Landlords has created ads. Landlords selected an ad.	
Success End Condition	Landlords successfully remove his created ads.	
Failed End Condition	Landlords failed to remove his created ads.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Remove ads request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to remove his selected ad.
	2	System will check the ads time duration is still remaining or not if it is not then system will delete the ads from the created ads list and also from ads information list.
	3	After removing the ads, system will display notification that the ad removed successfully.
Alternative Flows	**Step**	**Branching Action**
	2a	If the ads time duration is still remaining, then the system will show how much money system will pay back to landlords.
	2a1	Landlords will select next.
	2a2	See use case **Pay money**.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.15: Boost ads

Use Case 15	Boost ads	
Goal	Landlords wants to boost his created ads.	
Preconditions	Landlords is signed in. Landlords has created ads. Landlords selected an ad.	
Success End Condition	Landlords successfully boost his created ads.	
Failed End Condition	Landlords failed to boost his created ads.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Boost ads request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to boost his created ads
	2	System will prompt landlords that how many days he wants to boost his ads.
	3	Landlords will provide boost duration as days.
	4	Then system will calculate total cost for boosting that ads by multiply per day boosting cost with total boosting duration.
	5	See use case **Pay money.**
	6	System will boost the ad and update the ad as boosted ads.
Alternative Flows	**Step**	**Branching Action**
	2a	Landlords selected not to boost the ad and clicked cancel.
Quality Requirements	**Step**	**Requirement**

Table 7.16: Pay money

Use Case 16	Pay money	
Goal	Show money amount.	
Preconditions	Landlords are signed in. Landlords either remove ads or edit ads or boost ads or create ads.	
Success End Condition	Given money amount selected.	
Failed End Condition	No money amount selected.	
Primary Actors: **Secondary Actors:**	Landlords, System	
Trigger	Pay money request.	
Main Success Flows	**Step**	**Action**
	1	System ask the landlords to accept the money amount.
	2	Landlords accepted the money amount.
	3	See use case **Mobile financial services**.
Alternative Flows	**Step**	**Branching Action**
	2	Landlords choose cancel to provide the money.
Quality Requirements	**Step**	**Requirement**

Table 7.17: Mobile financial services

Use Case 17	Mobile financial services.	
Goal	Choose financial service.	
Preconditions	Landlords are signed in. Landlords accepted the cost amount.	
Success End Condition	Financial services selected.	
Failed End Condition	No financial service selected.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	Select mobile financial service request.	
Main Success Flows	**Step**	**Action**
	1	System ask the landlords to select financial service such as bKash, Rocket, Nagad.
	2	Landlords selected their preferred financial service.
Alternative Flows	**Step**	**Branching Action**
	2a	See use case **bKash**.
	2b	See use case **Rocket**.
	2c	See use case **Nagad**.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.18: bKash

Use Case 18	bKash	
Goal	Pay through bKash	
Preconditions	Landlords are signed in. Landlords accepted the cost amount.	
Success End Condition	Landlords selected bKash service for paying or getting money.	
Failed End Condition	bKash service not selected.	
Primary Actors: **Secondary Actors:**	Landlords	
Trigger	bKash service selected request.	
Main Success Flows	**Step**	**Action**
	1	Landlords requested to pay through bKash.
	2	System will prompt user to enter a valid bKash number.
	3	Landlords enter the bKash number.
	4	System will validate the bKash number.
	5	See use case **Transaction money**.
Alternative Flows	**Step**	**Branching Action**
	4a	System will prompt user wrong bKash number.
	4a1	System again ask the user to enter valid bKash number.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.19: Transaction money

Use Case 19	Transaction money	
Goal	Transact money from mobile bank.	
Preconditions	Landlords are signed in. Landlords accepted the cost amount. Landlords selected any financial service.	
Success End Condition	Landlords successfully pay money for his ads.	
Failed End Condition	Insufficient money on the selected financial service and ads not boosted or added or removed or edited.	
Primary Actors: **Secondary Actors:**	System Mobile bank	
Trigger	Money transaction request.	
Main Success Flows	**Step**	**Action**
	1	System will give a request to the specific mobile financial bank selected by the landlords for getting transaction of an amount.
	2	System will provide the amount of money and the mobile number.
	3	Mobile bank will validate whether the given mobile number has sufficient money for transaction.
	4	Mobile bank will transfer that amount of money from mobile bank to system bank.
	5	System then ask the stakeholders to enter the transaction id sent from mobile bank to the mobile number (financial service number).
	6	Getting full amount of money system will show a notification that the amount payment completed. And either add ads or remove ads or boost ads or edit ads.
Alternative Flows	**Step**	**Branching Action**
	3a	System will prompt the stakeholders that there is not sufficient money in financial service account.
	3a1	System then go back to use case **Mobile financial services**.
	5a	System will ask the stakeholders to enter the valid transaction id.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.20: Show all ads

Use Case 20	Show all ads	
Goal	Stakeholders will see all ads in ads information page (dashboard).	
Preconditions	Stakeholder are signed in.	
Success End Condition	Stakeholders will see all the available ads.	
Failed End Condition	No ads are available and can't see anything about ads.	
Primary Actors: **Secondary Actors:**	Landlords, Tenants	
Trigger	Requested for showing all ads.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to show all ads.
	2	System will load all ads from the database server.
	3	System will show boosted ads at first then normal ads randomly listed in ads information page.
	4	All ads will contain ads title, address of the house and thumbnail picture of the house.
Alternative Flows	**Step**	**Branching Action**
		No alternative flow.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.21: View favorite ads

Use Case 21	View favorites ads	
Goal	Stakeholders wants to see his bookmarked ads.	
Preconditions	Stakeholders are signed in.	
Success End Condition	Stakeholders can see his bookmarked ads in favorites list.	
Failed End Condition	Favorites list is empty.	
Primary Actors: **Secondary Actors:**	Landlords, Tenants	
Trigger	Requested for seeing favorites ads.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to see his favorites.
	2	System will show the favorite ads list.
	3	All ads will contain ads title, address of the house, thumbnail picture of the house and a button to remove.
Alternative Flows	**Step**	**Branching Action**
	3a	Stakeholders can remove ads from favorites by clicking on remove.
	3a1	System will remove the ads from favorites.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.22: Add ads to favorite

Use Case 22	Add ads to favorite	
Goal	Save or bookmark ads to see them in future.	
Preconditions	Stakeholders are signed in. Stakeholders at the ad's information page.	
Success End Condition	Ads will be added into favorites list.	
Failed End Condition	Ads won't added in the favorites list.	
Primary Actors: Secondary Actors:	Landlords, Tenants	
Trigger	Request for add ads in favorites list.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders will click add to favorite button beside each ad.
	2	System will add the ads in the favorites list.
	3	System will show a notification that ads is added to favorites list.
Alternative Flows	**Step**	**Branching Action**
	1a	If the ads already added to the system then it will remove the ads from favorites list.
	1a1	It will show notification to the user that ads are removed from favorites list.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.23: Manage profile

Use Case 23	Manage profile	
Goal	Stakeholders wants to see or edit his profile information.	
Preconditions	Stakeholders are signed in.	
Success End Condition	Stakeholders can manage his profile.	
Failed End Condition	Can't do anything to the profile.	
Primary Actors: Secondary Actors:	Landlords, Tenants	
Trigger	Manage profile request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to manage profile.
	2	System will give access to stakeholder's profile for viewing his information or editing.
	3	System will show view option and edit option.
Alternative Flows	**Step**	**Branching Action**
		No alternative flow.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.24: View profile information

Use Case 24	View profile information	
Goal	Stakeholders wants to see his profile.	
Preconditions	Stakeholders are signed in.	
Success End Condition	Stakeholders can see his profile information.	
Failed End Condition	No profile information will be shown.	
Primary Actors: Secondary Actors:	Landlords, Tenants	
Trigger	View profile request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested for seeing profile information.
	2	System will collect stakeholder's information from the database server.
	3	System will show stakeholders information like profile picture, stakeholders name, mobile number, email, username.
Alternative Flows	**Step**	**Branching Action**
		No alternative flow.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.25: Edit profile information

Use Case 25	Edit profile information	
Goal	Stakeholders will update his profile information.	
Preconditions	Stakeholders are signed in.	
Success End Condition	Stakeholders profile will be updated.	
Failed End Condition	No information of profile will change or get updated.	
Primary Actors: Secondary Actors:	Landlords, Tenants	
Trigger	Requested for editing profile.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to edit profile.
	2	System will show a form by filling the existing information of the stakeholders.
	3	Stakeholders now can edit his information like his name, mobile number, email and can upload his profile picture.
	4	After editing stakeholder's information now, he will click on update and all old information will be replaced by new one.
	5	System now will reload all information.
Alternative Flows	**Step**	**Branching Action**
	4a	System will prompt stakeholders whether he wants to update the information or not.
	4a1	Stakeholders choose cancel to update the information then system will prompt again whether he wants to discard all the changes currently he made.
	4a2	System will discard all changes made to the form and reload the old information.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.26: View notifications

Use Case 26	View notification	
Goal	Stakeholders wants see their notifications.	
Preconditions	Stakeholders is signed in.	
Success End Condition	Stakeholder successfully can see his notifications.	
Failed End Condition	There are no notifications.	
Primary Actors: **Secondary Actors:**	Landlord, Tenants	
Trigger	Show notification request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to see his notifications.
	2	System will display all notification information as list and each notification will contain subject and sender mail address.
	3	If stakeholder clicks on any of the notification, then system will display all notification subject and notification description of the selected notification.
Alternative Flows	**Step**	**Branching Action**
	3a	If Stakeholder clicks notification remove button beside each notification.
	3a1	System will remove that notification from the notification list.
	3a2	System will show remaining notifications.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.27: Search for house ads

Use Case 27	Search for house ads	
Goal	Stakeholders wants to search house.	
Preconditions	Stakeholders are signed in.	
Success End Condition	Stakeholders can see house ads based on search criteria.	
Failed End Condition	No house ads found based on that criteria.	
Primary Actors: **Secondary Actors:**	Landlords, Tenants	
Trigger	Search house ads request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to search ads.
	2	Stakeholders will give ads title and click on search option.
	3	System will search from the database server based on ads title given by stakeholders.
	4	System then shows ads which is matched the ads title at first then related ads will show as related ads section at last in the list.
Alternative Flows	**Step**	**Branching Action**
		No alternative flow.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.28: Search house ads nearby

Use Case 28	Search house ads nearby	
Goal	Stakeholders wants to search house in his area.	
Preconditions	Stakeholders are signed in.	
Success End Condition	Stakeholders can see house ads near his area.	
Failed End Condition	No house ads found near stakeholders' area.	
Primary Actors: **Secondary Actors:**	Landlords, Tenants	
Trigger	Search house ads nearby request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to search ads nearby.
	2	System will detect stakeholder's location.
	3	System will search from the database server based on location captured from stakeholder's current location.
	4	System then shows ads which is matched the location at first then related ads will show as related ads section at last in the list.
Alternative Flows	**Step**	**Branching Action**
	2a	If system can't get access to device location, system will prompt the stakeholders to turn on the location service and give system permission to access the location.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

Table 7.29: Search house by filtering

Use Case 29	Search house by filtering	
Goal	Stakeholders wants to search house by specific criteria.	
Preconditions	Stakeholders are signed in.	
Success End Condition	Stakeholders can see house ads based on their given criteria.	
Failed End Condition	No house ads found by the given criteria.	
Primary Actors: **Secondary Actors:**	Landlords, Tenants	
Trigger	Search house by filtering request.	
Main Success Flows	**Step**	**Action**
	1	Stakeholders requested to search ads by giving criteria.
	2	System will show a filter box which contains filter by ads title, rooms count, rent per month, location and flat size.
	3	Stakeholders will give necessary filter information and click on search.
	4	System then shows ads which is matched the given criteria at first then related ads will show as related ads section at last in the list.
Alternative Flows	**Step**	**Branching Action**
	2a	If no filter given system will prompt for filling any criteria.
Quality Requirements	**Step**	**Requirement**
		Not applicable.

8 Activity Diagram

An activity diagram is a graphical representation of an executed set of procedural system activities and considered a state chart diagram variation. Activity diagrams describe parallel and conditional activities, use cases and system functions at a detailed level. Activity diagram for house rental management system are given below.

8.1 Sign up

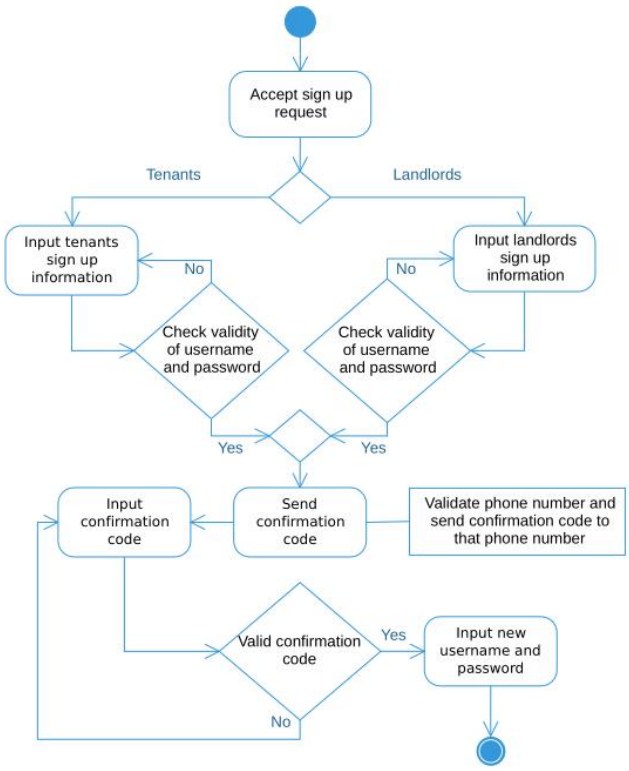

Figure 8.1: Sign up

8.2 Sign in

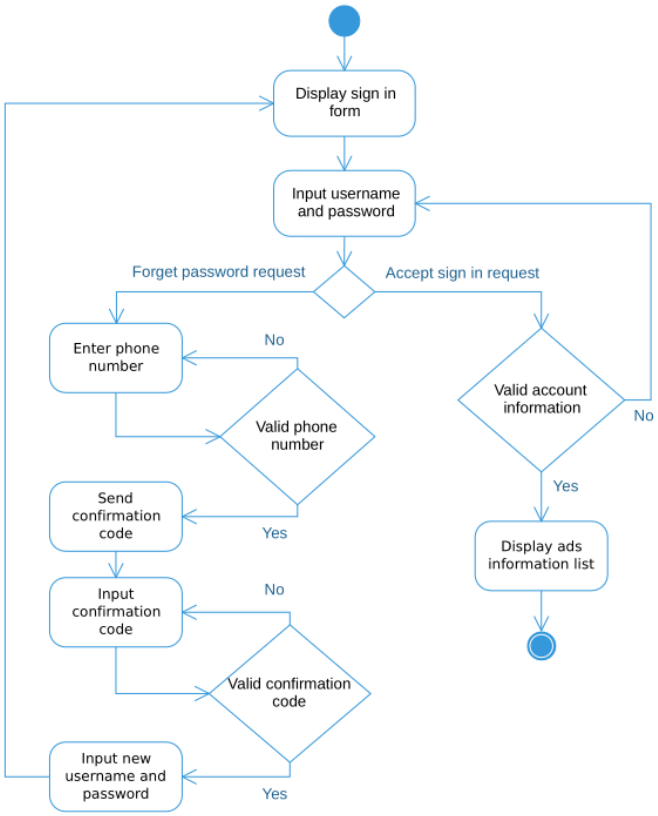

Figure 8.2: Sign in

8.3 View owned house

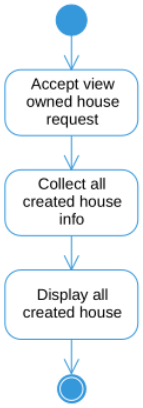

Figure 8.3: View owned house

8.4 Add house

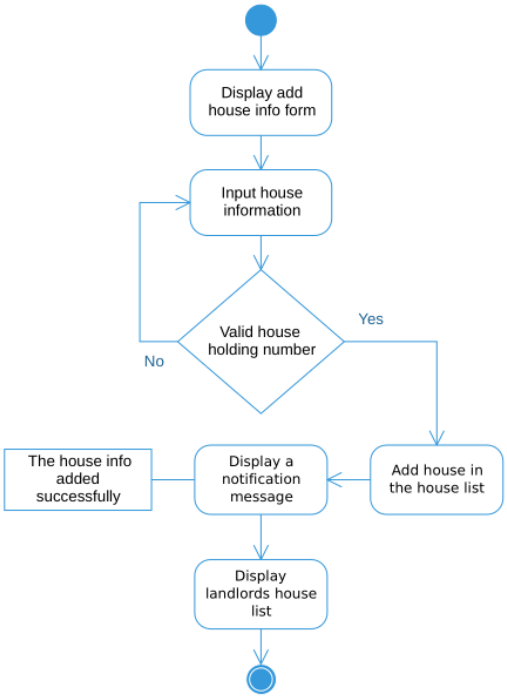

Figure 8.4: Add house

8.5 Remove house

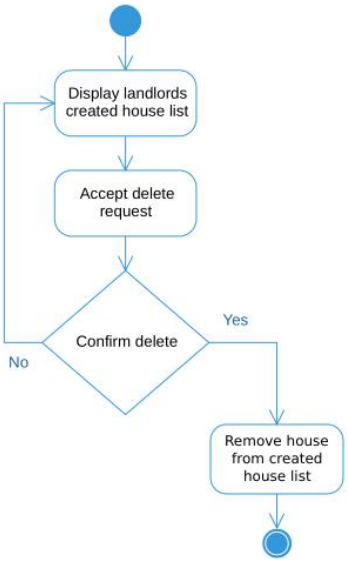

Figure 8.5: Remove house

8.6 View house info

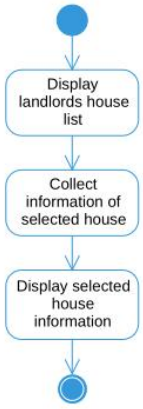

Figure 8.6: View house info

8.7 Edit house info

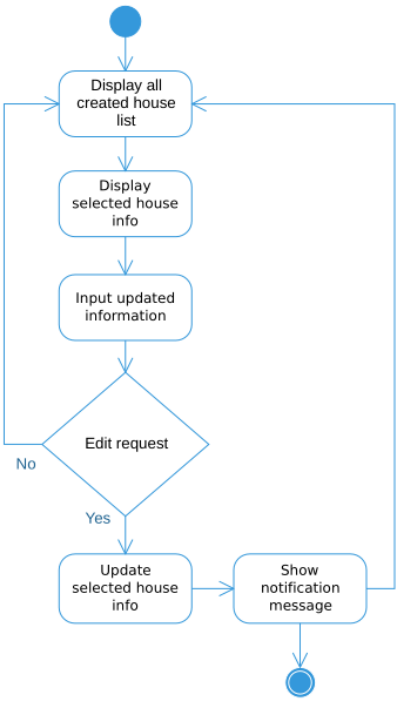

Figure 8.7: Edit house info

8.8 Add tenant info

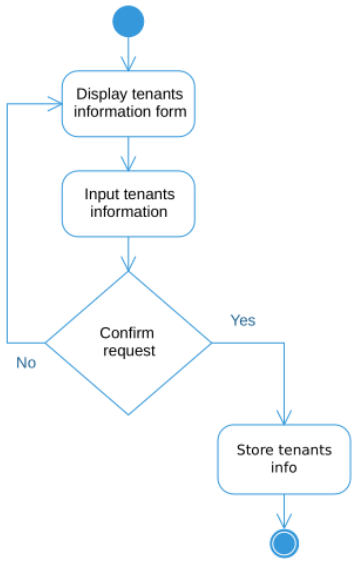

Figure 8.8: Add tenant info

8.9 Remove tenant's information

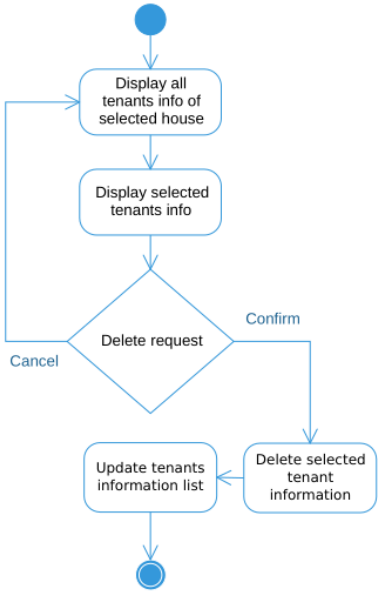

Figure 8.9: Remove tenant's information

8.10 Show created ads

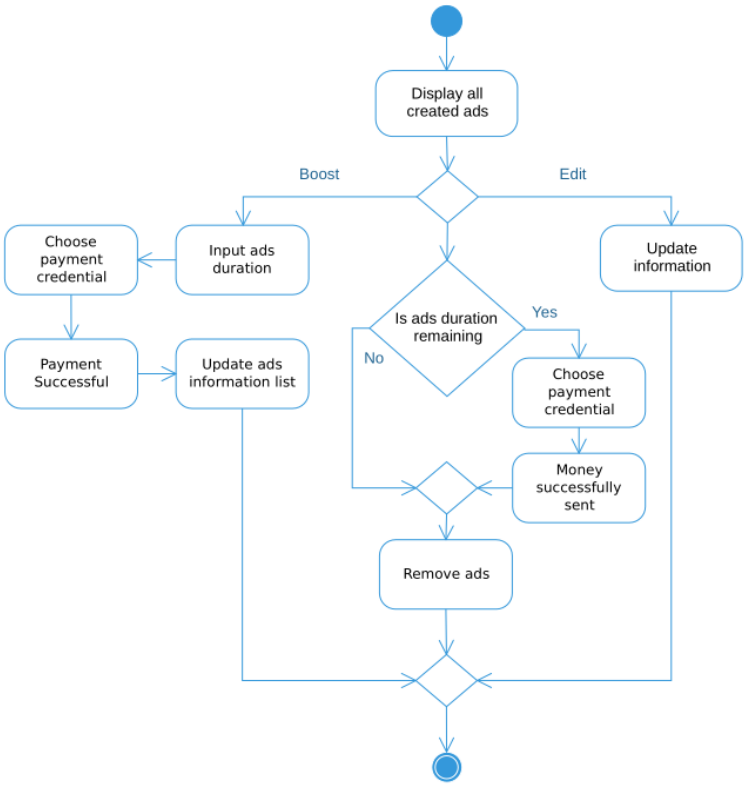

Figure 8.10: Show created ads

8.11 Create ads

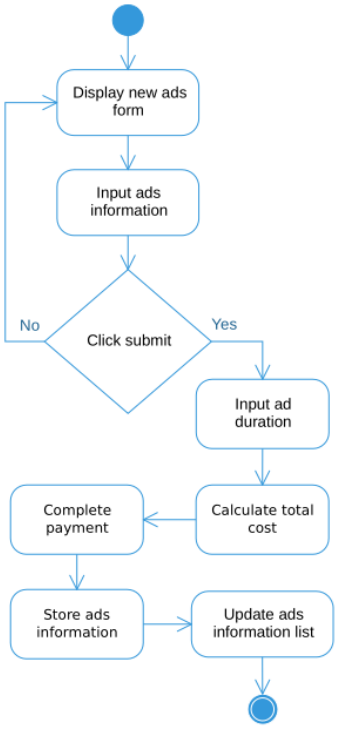

Figure 8.11: Create ads

8.12 Edit ads

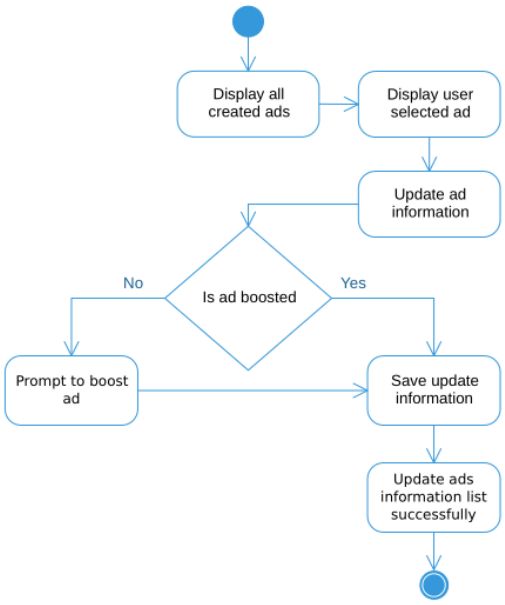

Figure 8.12: Edit ads

8.13 Remove ads

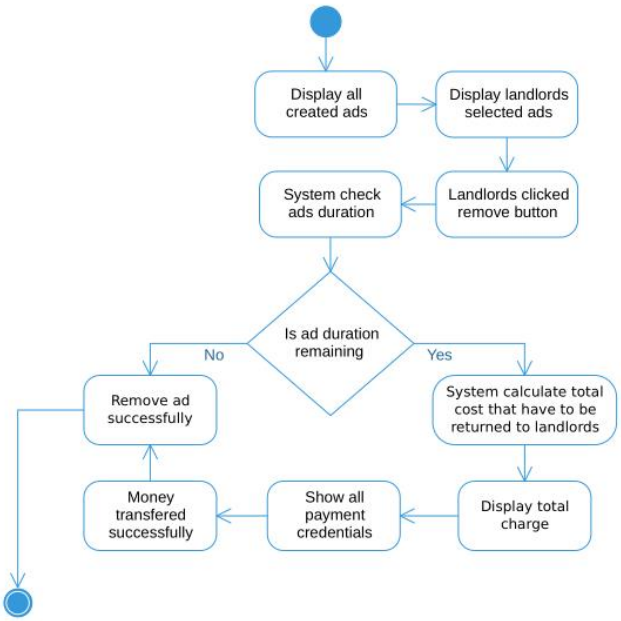

Figure 8.13: Remove ads

8.14 Boost ads

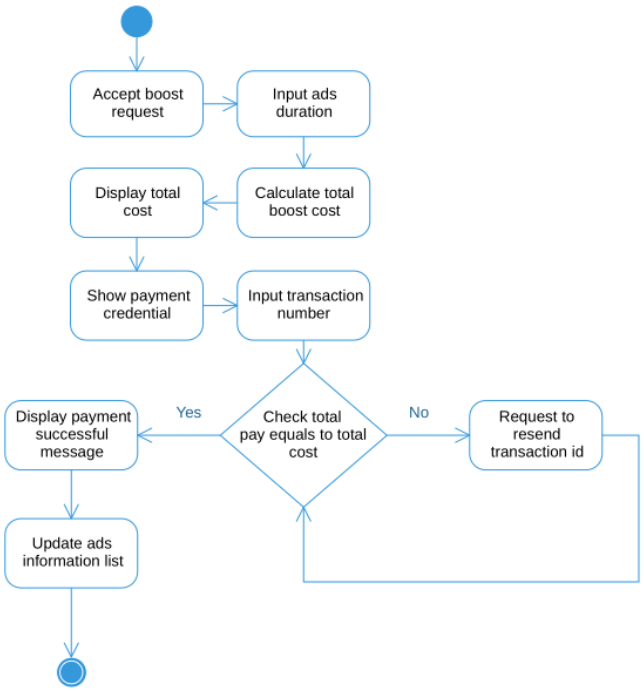

Figure 8.14: Boost ads

8.15 Show all ads

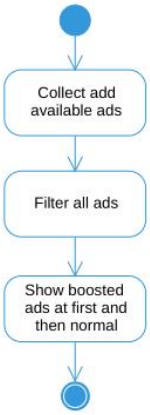

Figure 8.15: Show all ads

8.16 Add ads to favorite

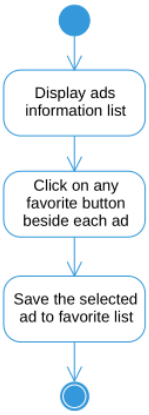

Figure 8.16: Add ads to favorite

8.17 Manage profile

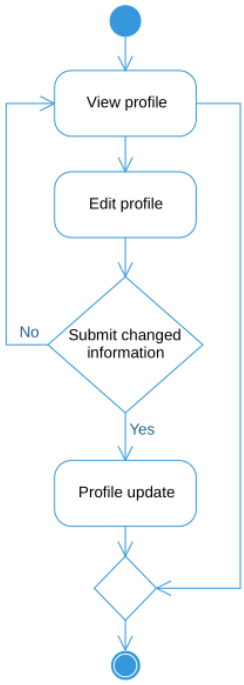

Figure 8.17: Manage profile

8.18 View profile information

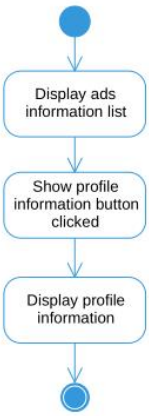

Figure 8.18: View profile information

8.19 View notifications

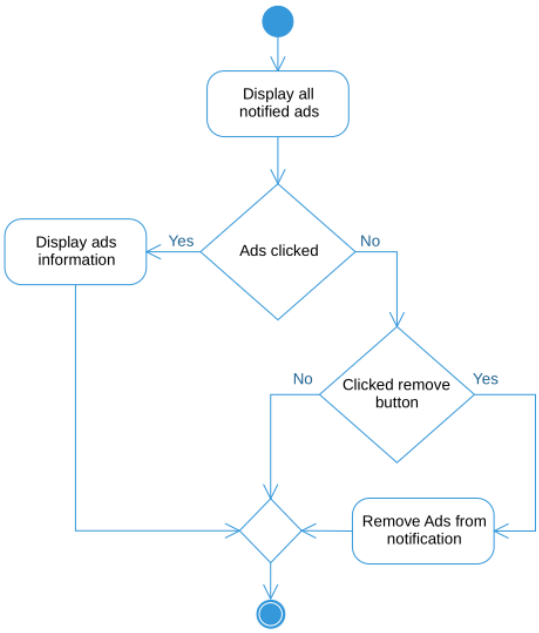

Figure 8.19: View notifications

8.20 Search for house ads

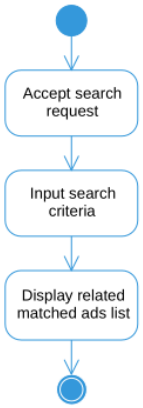

Figure 8.20: Search for house ads

8.21 Search house nearby

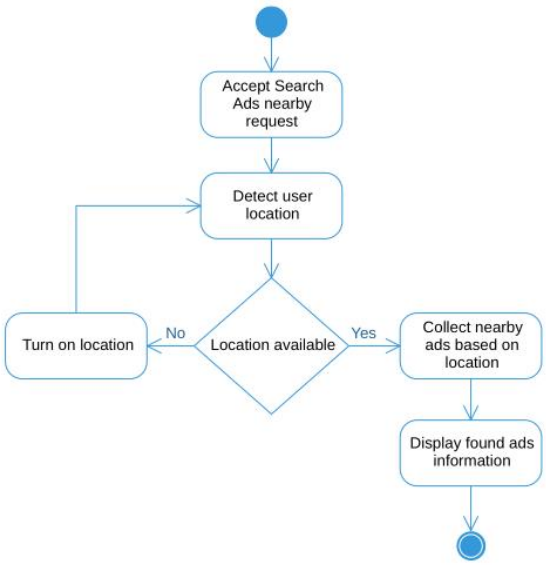

Figure 8.21: Search house nearby

9 Requirements traceability matrix

A traceability matrix is a document, usually in the form of a table, used to assist in determining the completeness of a relationship by correlating any two baselined documents using a many-to-many relationship comparison. It is often used with high-level requirements (these often consist of marketing requirements) and detailed requirements of the product to the matching parts of high-level design, detailed design, test plan, and test cases.

9.1 Business requirements

BR#	Business requirements
BR1	Allow stakeholder to search by location.
BR2	Allow landlords to store and access tenant's information.
BR3	Allow landlords to create remove and edit ads.
BR4	Allow stakeholder to bookmarks vacant ads.
BR5	Allow stakeholder to communicate with landlords for rent.
BR6	Allow landlords to boost his ads.
BR7	Check provided information is correct or not.

9.2 Test cases

Test case No.	Test case
TC1	Chittagong, 2400ft.
TC2	Detect User location.
TC3	AL JABER, NID NO-01234556666, HAKIM, JORINA.
TC4	NUR MONJIL, HN-1234, 4, 6, PICTURE.
TC5	Try to remove ads.
TC6	Try to edit or update existing ads.
TC7	Try to access all ads.
TC8	Try to communicate with landlords.
TC9	Boosting an ad.
TC10	Store tenant's information.

10 Appendix

10.1 Prioritization of requirements

We've prioritized the functional requirements by following **Three-level Scale technique.**

10.1.1 Three-level Scale

When a Business Analyst categorizes the requirements in any of the ordering or ranking scale, it is subject to the analyst's understanding of the business. Many analysts suggest that this method has some drawbacks and advocate methods that have more than one scale.

FR1 – High priority: Its essential requirement for our system. Stakeholder always can't find his vacant house in default dashboard ads. So, it's very important for a stakeholder that he can find a house with some specific criteria.

FR2 – Medium priority: Sometimes stakeholder wants houses nearby where he lives in. So, stakeholder just need to click "House nearby" button, then he can see all those ads which is placed at his location. So, it is very important requirement for our system.

FR3 – Medium priority: It is also urgent requirement for our system. Because sometimes stakeholder wants to bookmark some desire ads so that he can check it later.

FR4 – Low priority: This feature is basically less important for the system as well as for the Stakeholders. Because stakeholders will be rarely notified when there is any available house around his location. So, we implement this requirement later.

FR5 – High priority: It is the main theme of our project. This the most important and urgent requirement in the system. This requirement defines how a stakeholder can rent a house.

FR6 – High priority: This requirement is also highly important because tenants will have to communicate with the house owners

FR7 – Low priority: Rating is not that important for getting a house rented. And searching house ads is not depend on rating of a house.

FR8 – High priority: Adding ads is important. If new ads not added to the system stakeholders won't see ads or get house ads.

FR9 – High priority: This is one of the best requirements in our system. Cause this requirement is basically making the difference between our system and other available systems.

FR10 – High priority: As posting ads on the system is a bit costly so no one would want to continue posting their ads even after getting his house rented. So, the landlord will must remove the ads from the system for saving money.

FR11 – High priority: In our country, normally police collect tenant's information manually. But we want to digitalize this system by storing all the information of the tenants and provide them to the police.

FR12 – High priority: Police officers can access all information that are saved in our system. For security issue police need to access the information about tenants as well as landlords.

DR1 – High priority: Landlords will upload their house's recent pictures. But the pictures must not exceed 3MB limit so that database do not be too much large in size.

PR1 – Medium priority: Searched result will be loaded within a second and pictures will be loaded within 5 seconds. Sometimes it may take some more times, but this is not that much important cause it may take some time that shouldn't matter that much.

MR1 – Medium priority: It's important to update the database of the newly renting peoples because police may need to get any information of the current tenants of any house.

AR1 – High Priority: To attract the online users to use our system more and more, color and font are so much important part. Just to ensure that users do not get distracted for the color combination or font size and font colors.