Software Requirements Specification (SRS)

For Rent iT A House Renting Solution.

Version 1.0

Prepared by Team 16

University of Liberal Arts Bangladesh

Date: 25.10.2023

Contents

[1. Introduction 3](#_Toc149320901)

[1.1 Purpose 3](#_Toc149320902)

[1.2 Scope 3](#_Toc149320903)

[1.3 Definitions, Acronyms, and Abbreviations 3](#_Toc149320904)

[1.4 Overview 3](#_Toc149320905)

[2. Overall Description 4](#_Toc149320906)

[2.1 Product Perspective 4](#_Toc149320907)

[2.2 Product Functions 4](#_Toc149320908)

[2.3 User Characteristics 4](#_Toc149320909)

[2.4 Constraints 4](#_Toc149320910)

[2.5 Assumptions and Dependencies 4](#_Toc149320911)

[3. System Features and Requirements 4](#_Toc149320912)

[3.1 Functional Requirements 4](#_Toc149320913)

[3.2 Non-Functional Requirements 6](#_Toc149320914)

[3.3 System Interfaces 7](#_Toc149320915)

[3.4 User Interfaces 7](#_Toc149320916)

[3.5 Hardware Interfaces 7](#_Toc149320917)

[3.6 Software Interfaces 8](#_Toc149320918)

[3.7 Communications Interfaces 8](#_Toc149320919)

[3.8 Operations 8](#_Toc149320920)

[4. Diagrams 9](#_Toc149320921)

[4.1 Use case diagram 9](#_Toc149320922)

[4.2 Class Diagram 10](#_Toc149320923)

[4.3 Data Flow Diagram 10](#_Toc149320924)

[5. Other Requirements 10](#_Toc149320925)

[5.1 Legal and Regulatory Compliance: 11](#_Toc149320926)

[5.2 Security Requirements: 11](#_Toc149320927)

[5.3 Backup and Disaster Recovery: 11](#_Toc149320928)

[5.4 Performance and Scalability: 11](#_Toc149320929)

[5.5 Ethical and Social Considerations: 11](#_Toc149320930)

[5.6 Language and Localization: 11](#_Toc149320931)

[6. Supporting Information 12](#_Toc149320932)

[6.1 Glossary and Terminology: 12](#_Toc149320933)

[6.2 Assumptions and Dependencies: 12](#_Toc149320934)

[6.3 Data Dictionary: 12](#_Toc149320935)

[6.4 User Personas and Scenarios: 12](#_Toc149320936)

[6.5 Technical Constraints and Limitations: 12](#_Toc149320937)

[6.6 Standards and Guidelines: 12](#_Toc149320938)

[6.7 Security Protocols: 13](#_Toc149320939)

[6.8 Mockups and Prototypes: 13](#_Toc149320940)

[6.9 Change Management and Version Control: 13](#_Toc149320941)

[6.10 Risk Management: 13](#_Toc149320942)

[6.11 Quality Assurance and Testing: 13](#_Toc149320943)

[- An outline of the quality assurance practices and testing strategies that will be employed 13](#_Toc149320944)

[6.12 Training and User Support: 13](#_Toc149320945)

[6.13 Legal and Regulatory Documentation: 13](#_Toc149320946)

[6.14 Contact Information: 13](#_Toc149320947)

# 

# 1. Introduction

## 1.1 Purpose

Many students, employees and other professionals find difficulties in finding house on rent. Also there are property owners who are searching for a platform to post their property. This is the best platform for of them. Also it is such system which excludes the work of middlemen, i.e. the broker and thus, brings transparency in the process.

## 1.2 Scope

The Things it can do

* The system provides login/signup facilities to the users including OTP verification.
* The system provides the members with the option of forgot password and also delete user profile.
* Owners can add the images of their property along with other information.
* The user can search for house deals using sorting and filters and also add it to favorites.
* The application provides list of the house deals.
* The system provides options to delete/edit/view house deals and to contact owner.

## 1.3 Definitions, Acronyms, and Abbreviations

SRS : Software Requirements Specification

UI : User Interface

UX : User Experience

## 1.4 Overview

This document includes a detailed description of Rent iT software requirements, providing a comprehensive outline for stakeholders, developers, and designers involved in the project.

# 2. Overall Description

## 2.1 Product Perspective

Rent iT is an independent house renting system, an Android Mobile Application which will help students, employees and other professionals who are living as PGs or on rent and also for the property owners who are searching for a platform to post their property. It also brings complete transparency between the customer and the owner removing the middlemen.

## 2.2 Product Functions

* The system provides login/signup facilities to the users including OTP verification.
* The system provides the members with the option of forgot password and also delete user profile.
* Owners can add the images of their property along with other information.
* The user can search for house deals using sorting and filters and also add it to favorites.
* The application provides list of the house deals.
* The system provides options to delete/edit/view house deals and to contact owner.

## 2.3 User Characteristics

The primary users of Rent iT will be individuals who looking for house in rent and another user is house owner who want to give his house in rent.

## 2.4 Constraints

- Internet dependency for real-time data fetching and booking

- Dependency on external APIs for house and other(address, house size, house rent) information etc.

## 2.5 Assumptions and Dependencies

* Assuming that user do have the basic knowledge of operating the internet and access it.
* The administrator is expected to be familiar with the interface of the tech support system.
* The software is highly dependent on the type of the operating system i.e. it is an android application so it will not work with phones having different operating systems.

# 3. System Features and Requirements

## 3.1 Functional Requirements

**F1:Authentication and Authorization**

**F1.1** User login/signup along with the OTP verification.

* Input: Username and Password for login or Username/E-mail, Password, Mobile Number for signup
* Output: Valid user or not valid user for login or new user created for signup along with OTP verification.

**F1.2** Forgot Password and Reset Password in case the user forgets the password.

* Input: Selects Forgot Password and Reset Password option and inputs new password.
* Output: New password is set.

**F2: Edit or Delete User Profile:**

**F2.1** Edit User Profile

* Input: User makes changes in his/her profile data.
* Output: Profile is edited.

**F2.2** Delete User Profile Input: Select delete account.

* Output: User Profile is deleted.

**F3: Provide comprehensive details:**

**F3.1** System lists out all the details of house.

* Output: List of houses is displayed.

**F3.2** Add houses to favorites

* Input: User login and select add to favorites
* Output: House added to favorites

**F3.3** Search houses using sort(basis on rent – low to high/high to low, post time – newest first, recently updated) and filter (city, apartment type, BHK type, property size, no. of floors)

* Input: Select any sort or filters.
* Output: List of houses appears according to the sort or filters applied.

**F3.4** Owner can edit/delete the house deal  Input: Edit or delete the property.

* Output: Does not display that property. Property is removed from the database.

**F4: Add property:**

**F4.1** User can add the property along with its images.

* Input: Add the property and its images.
* Output: Property is added and its images are displayed.

**F4.2** Show details of the property  Input: Select any property.

* Output: All types of information related to the location, facilities, amenities, cost, space of property is provided by the owner. The owner can also add the location using the Google Maps.

**F5: Contact between the owner and tenant:**

* The owner and the tenant can have direct contact between each other.
* There is no middleman required in the process.

## 3.2 Non-Functional Requirements

**Performance Requirements**

The system must be efficient to handle traffic. For now, in our system 10 user can simultaneously use the application without any error.

**Safety Requirements**

There are two major types of software f failure which may cause possible loss or damage to system: Software logic errors and Software support errors.

Software logic errors are often a result of the programmer making errors in the coding, whether it is simply a mistake on their part, or an incorrect set of requirements they are following. In addition, it may have been a mistake made in the design phase which follows through into the implementation of the system.

Software support errors are linked to the software being used to create the program. Perhaps these errors are from the compiler, an external library being used, or even the programming language.

We have made some safety requirements trying to solve the problems above. They are:

1) Analyze before coding

1. Be careful when coding
2. Do large amount of tests after coding

**Security Requirements**

At the time of registration, as they will register a one-time password will be send to the registered number and the user will be allowed to proceed further only after he/she verifies that code. The password generated by the user will be encrypted. There is full assurance of no data loss.

**Software Quality Attributes:**

**Usability:**

Usability defines how difficult it will be for a user to learn and operate the system.

The system should satisfy a maximum number of customer needs.

No predefined skills are required to use this system. User can easily interact with the system.

Most of the tasks a user can complete without any help. It does not have complex design so any user can easily interact.

**Flexibility:**

Flutter is an open-source mobile application development framework. It is cross-platform so can be used for both Android as well as iOS.

**Availability:**

There are no restrictions for the use of the system. The system is open for all interested seekers. The user must have access to any hardware device with Android support and internet connection.

**Reliability:**

Reliability defines how likely it is for the software to work without failure for a given period of time. The database update process must rollback all related updates when any update fails.

**Security:**

The system shall use secure sockets in all interactions that include any confidential user information. The system shall log out all the users after a period of inactivity.

**Performance:**

Performance is a quality attribute that describes the responsiveness if the system to various user interactions with it. The front-page load time must be no more than 5 seconds.

These requirements are essential for ensuring that the Rent iT platform operates effectively, securely, and to the satisfaction of its users. By addressing both the functional and non-functional aspects, the platform aims to provide a comprehensive and user-friendly house renting experience.

## 3.3 System Interfaces

- Database for storing user profiles, house information, and bookings

- External APIs for fetching house and other information

## 3.4 User Interfaces

- Responsive web interface

- Mobile application (Android)

## 3.5 Hardware Interfaces

- Web servers

- Database servers

## 3.6 Software Interfaces

- Mobile operating systems (Android)

## 3.7 Communications Interfaces

- HTTPS for secure communication

- RESTful APIs for data exchange

## 3.8 Operations

- 24/7 availability

- Regular updates and maintenance

# 4. Diagrams

## 4.1 Use case diagram

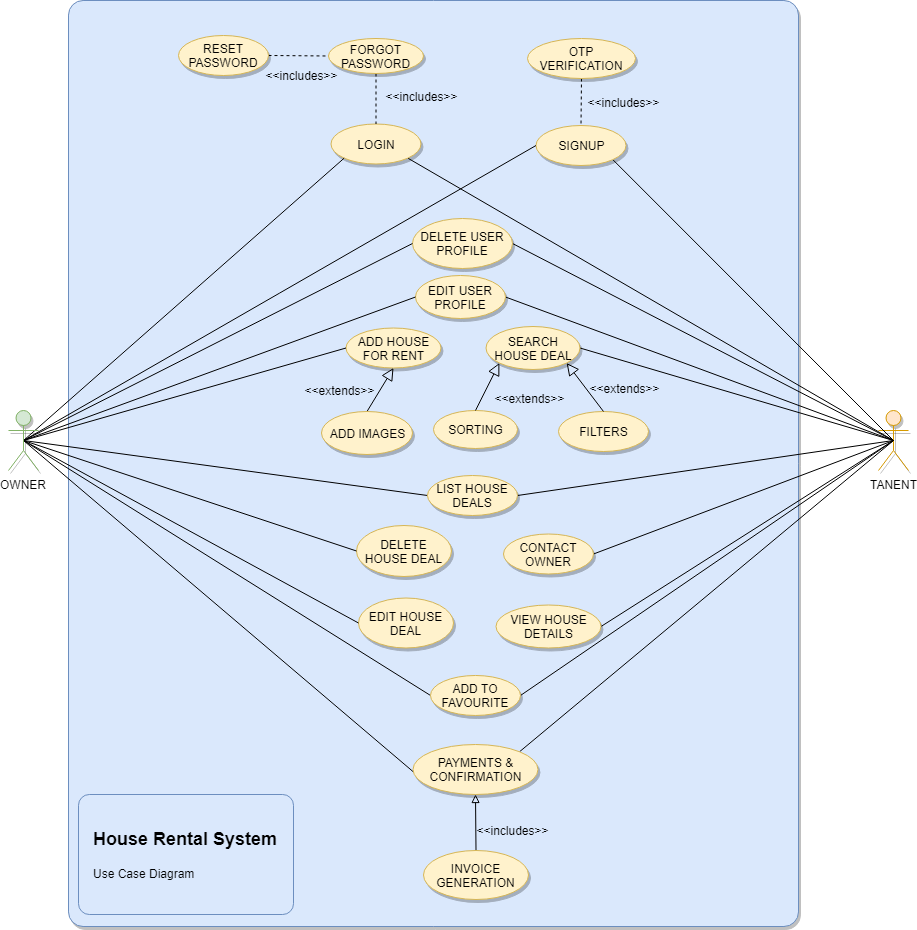


Fig. 4.1 Use Case Diagram

## 4.2 Class Diagram

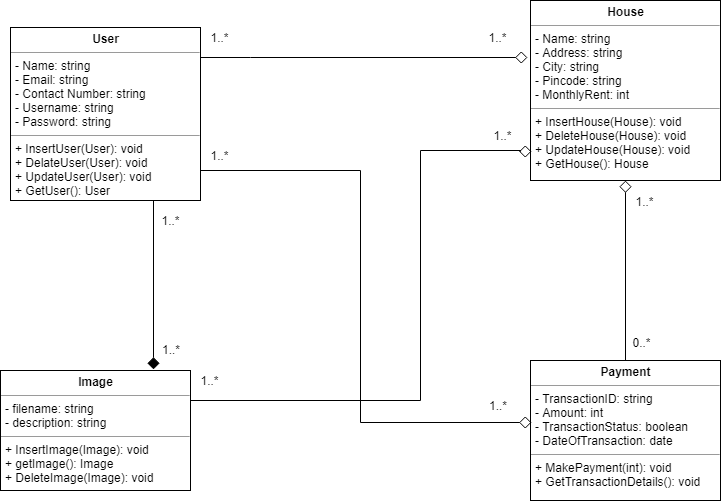


Fig. 4.2 Class Diagram

## 4.3 Data Flow Diagram

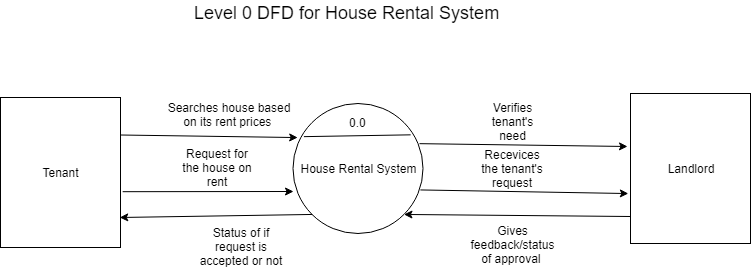


Fig. 4.3 Data Flow Diagram (level 0 DFD)

# 5. Other Requirements

[Include any additional requirements such as legal, regulatory, or compliance-related requirements]

In developing Rent iT exclusively for Bangladesh, it is crucial to adhere to the country-specific legal, regulatory, and compliance-related standards to ensure the platform operates within the legal framework, provides secure services, and maintains the trust of its users. Below are the additional requirements tailored for the Bangladeshi context:

## 5.1 Legal and Regulatory Compliance:

- Data Protection and Privacy: Comply with the Bangladesh Data Protection Act, ensuring that user data is handled securely, and privacy is maintained. Users should have the ability to access, update, and request deletion of their personal data.

- Intellectual Property Rights: Ensure all content, including images, text, and other media, comply with Bangladeshi copyright laws, and any third-party content is properly licensed.

- Consumer Rights Protection: Abide by the Consumer Rights Protection Act of Bangladesh, providing clear and truthful information about renting packages, accommodations and maintaining transparency in pricing and terms of service.

- Accessibility: Ensure the platform is accessible to individuals with disabilities, adhering to internationally recognized accessibility standards.

## 5.2 Security Requirements:

- Data Encryption: Implement strong encryption protocols for user data and transaction information to prevent unauthorized access.

- Secure Payment Processing: Ensure secure payment processing, particularly focusing on popular payment methods in Bangladesh, and comply with relevant financial regulations.

- Regular Security Audits: Conduct security audits regularly to identify and mitigate potential risks, ensuring the platform’s integrity.

## 5.3 Backup and Disaster Recovery:

- Data Backup: Establish a robust data backup system, ensuring all critical data is backed up regularly to prevent loss due to unforeseen incidents.

- Disaster Recovery Plan: Implement a comprehensive disaster recovery plan to ensure quick service restoration in case of system failures or other emergencies.

## 5.4 Performance and Scalability:

- Local Server Hosting: Consider hosting servers locally in Bangladesh to ensure optimal performance and reduced latency for Bangladeshi users.

- Scalability: Ensure the platform is scalable to accommodate growth in user base and data volume, ensuring consistent performance.

## 5.5 Ethical and Social Considerations:

- Cultural Sensitivity: Ensure that content and services are culturally appropriate and respectful to the diverse population of Bangladesh.

- Environmental Responsibility: Promote eco-friendly house options and contribute to sustainable renting practices in Bangladesh.

## 5.6 Language and Localization:

- Bengali Language Support: Provide full support for Bengali, the official language of Bangladesh, ensuring that the platform is accessible and user-friendly for local users.

- Local Currency and Payment Methods: Support Bangladeshi Taka (BDT) and popular local payment methods to facilitate seamless transactions.

By addressing these requirements, Rent iT aims to create a trustworthy, secure, and user-friendly platform that is fully tailored to the needs and legal standards of Bangladesh, ensuring a seamless and enjoyable house renting experience for Bangladeshi users.

# 6. Supporting Information

[Additional information to support the requirements described in this document]

To supplement the requirements outlined in this document and provide further clarity on the Rent iT platform’s functionality and design, the following supporting information is provided:

## 6.1 Glossary and Terminology:

- A comprehensive glossary of terms and abbreviations used throughout this document and the platform, ensuring all stakeholders have a clear understanding of the language and concepts used.

## 6.2 Assumptions and Dependencies:

- A list of assumptions made during the requirements gathering process and dependencies that may affect the implementation and operation of the platform.

## 6.3 Data Dictionary:

- Detailed descriptions of the data elements used in the system, including format, type, and any constraints, to ensure consistency across the platform.

## 6.4 User Personas and Scenarios:

- Development of user personas representing the target audience of Rent iT, accompanied by scenarios illustrating typical user interactions with the platform, helping to contextualize the requirements.

## 6.5 Technical Constraints and Limitations:

- A description of any technical constraints or limitations that may impact the design or functionality of the Rent iT platform.

## 6.6 Standards and Guidelines:

- A list of industry standards, guidelines, and best practices that the platform will adhere to, ensuring quality, performance, and security.

## 6.7 Security Protocols:

- Detailed description of the security measures and protocols in place to protect user data and ensure the integrity of the platform.

## 6.8 Mockups and Prototypes:

- Provision of visual mockups or prototypes of the user interface, helping stakeholders visualize the final product and ensure alignment with the requirements.

## 6.9 Change Management and Version Control:

- A description of the procedures in place for managing changes to the requirements document, ensuring all stakeholders are aware of updates and modifications.

## 6.10 Risk Management:

- Identification of potential risks associated with the platform’s development and operation, along with strategies for mitigation.

## 6.11 Quality Assurance and Testing:

- An outline of the quality assurance practices and testing strategies that will be employed to ensure the Rent iT platform meets all requirements and operates reliably.

## 6.12 Training and User Support:

- Details on any training programs or user support services that will be provided to help users navigate and utilize the Rent iT platform effectively.

## 6.13 Legal and Regulatory Documentation:

- Copies or references to any legal or regulatory documents that have been cited in the requirements, ensuring full transparency and access to source material.

## 6.14 Contact Information:

- A list of contact information for all stakeholders and team members involved in the Rent iT project, facilitating easy communication and collaboration.

By providing this supporting information, stakeholders and team members involved in the Rent iT project are equipped with a comprehensive understanding of the platform, its requirements, and the context in which it operates, contributing to a smooth development process and successful implementation.

This Software Requirements Specification provides a thorough description of all the functions and specifications of the Rent iT platform. It serves as a guide for the development team and a point of reference for all stakeholders involved in the project.