Shreedeep Grand Management

Software Requirements Specification

Version 1.0

Prepared by:
22CE024 – Naitik Desai
22CE028 – Roshni Dholariya
22CE029 - Saloni Gadhiya
22CE030 – Rishee Gandhi

Revision History

Date	Version	Description	Author
25/02/2024	1.0	Software Requirement Specification Document Initial Release	Naitik Desai

Table of Contents

Software Requirements Specification

1. Introduction

1.1 User Story

- Mr. Nimesh, the hostel stakeholder, is currently managing hostel operations manually using excel sheets to track all records. With a growing number of students and increasing complexities in managing room allocations, fee payments, and other hostel-related tasks, Mr. Nimesh finds it challenging to ensure smooth hostel operations.

1.2 Purpose

- To solve hostel management system challenges, customize software that automates hostel operations, track student leaves, manage complaint requests and enhances overall efficiency.

1.3 Scope

Hostel Management System provides a software interface with features of Fees Payment, Food Menu and few other features as per given below:

- Login for Admin
- Login for Students
- Complaint section for students
- Leave entries by students
- Reset password of account
- Noticeboard management by Admin
- Food Menu update by Admin

1.4 Overview

The SRS will provide a detailed description of Shreedeep Grand Management, which is a hostel management system. This document will provide the outline of the requirements, and an overview of the characteristics and constraints of thesystem.

- **Section 2:** This section of the SRS will provide the general factors that affect the product and its requirements. It provides the background for those requirements. The items such as product perspective, product function, user characteristics, constraints, assumption and dependencies are described in this section.

- **Section 3:** This section of SRS contains all the software requirements mentioned in section 2 in detail, sufficient enough to enable the designers to design the system to satisfy the requirements and testers to test if the system satisfies those requirements.

2. Overall Description

2.1 Product Perspective

The Shreedeep Grand Management is software to be used by our client Mr. Nimesh to digitalize the hostel management which he has been maintaining manually. The development of Shreedeep Grand Management will be beneficial for our client as well as the students. The Shreedeep Grand Management project aims to develop a smooth and accurate management platform to assist hostel owner to efficiently manage daily hostel task like, updating food menu, fees data of students, manage noticeboard, resolve student's complaint.

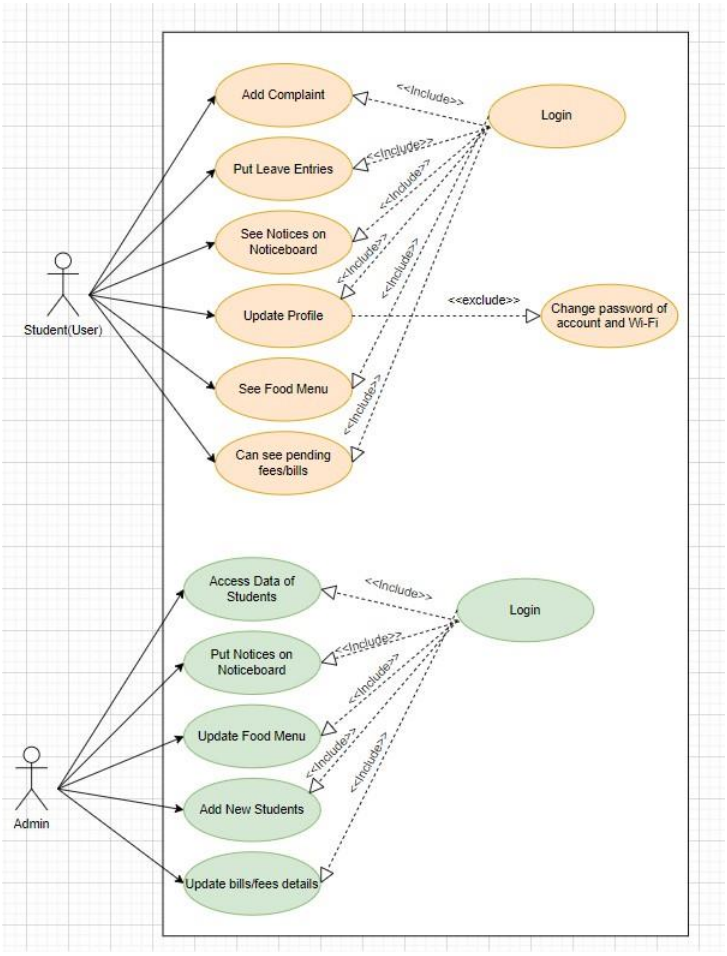

Figure 1: Use Case Diagram

2.2 Product Functions

The function of the system includes:

1. Login for Admin
2. Login for Students
3. Complaint Registration by students
4. Leave Entries by students.
5. Students can see food menu, Notices, bill/fee details.
6. Students can update their profile and passwords of their account.
7. Admin will have all the data of the students.
8. Admin will manage notices.
9. Admin can update food menu.
10. Admin can update students' data and add new student data.
11. Admin can manage requests for bill/fees payment.

2.3 User Characteristics

The users of the system are students. The administrator of the system is assumed to have basic knowledge of computers. The appropriate user interface, user's manual, online help and the guide to install and maintain the system must be efficient to educate the admin on how to use the system without any problem. The users can access the system easily through its user-friendly interface.

2.4 Constraints

1. The Information regarding all of the students should be stored in the database that is accessible by Shreedeep Grand Management.
2. Shreedeep Grand Management should be installed on the Administrator's computers.
3. The users can access Shreedeep Grand Management from any device that has an Internet connection.
4. The users must have their correct username and passwords to gain access to their accounts on Shreedeep Grand Management.

2.5 Assumptions and Dependencies

1. The administrator and students should have a basic knowledge of computers.
2. Admin's computer should have an Internet Connection.
3. Users accessing the website should have an Internet Connection.
4. Student is assured about his username & password by admin.

3. Specific Requirements

- This Section describes all the Functional as well as the Non-functional requirements. It contains a description of functions and capabilities that the product must provide.

3.1 Functional Requirement

- Functional requirements describe how a product must behave, what are its features and functionalities.

F1: Login for Admin

The admin can log in to the system using their username and password.

Input: Admin Username and Password.

Output: Admin Dashboard with access to administrative features.

Processing: The system verifies the admin's credentials against the database, and if valid, grants access to the Admin Dashboard.

F2: Admin can access student details.

The admin adds and updates student information.

Input: Student full name, College program information, parents/guardian information and other student relevant data. Update current student information.

Output: Added/updated data reflects to database and displayed to Admin.

F3: Admin can put notice in noticeboard section and food menu.

The admin can manage notices and update the food menu.

Input: Updates food menu and notice.

Output: Students can see new notices in the noticeboard section, updated foodmenu.

F4: Admin puts bills/fees.

Input: Admin will upload electricity bill and student fees details.

Output: Students can see pending bills and fees.

F5: Login for students

- Login: The user can login to the system with his/her username and password.

  Input: Username and Password.

  Output: Student Dashboard

  Processing: Username and password are verified from the database and if a student exists in the database then the user interface will be displayed with his/her information.

- Logout: The student can logout from the system.

  Input: 'Logout' option is selected.

  Output: Student login Screen will be displayed.

- Password reset: Students can reset their password.

  Input: New Password

  Output: New Password will reflect to database.

F6: Students can register a complaint.

Students can complain related to assets of room, about cleaning, about misbehaviour of any student.

Input: Description of complaint by student.

Output: Complaint goes to Admin Dashboard in Complaints section.

F7: Students can update their profile.

Users can update information like contact details, address, parent's details and other things.

Output: Reflects to database.

F8: Students can see bill/fees details, also notices in noticeboard section, check food menu.

Output: Students can check notices and pending fee details, check electricity bills, see food menu.

F9: Student puts leave requests.

The student put details like where he is going, at what time to what time are needed.

Input: Adds place(home/nadiad/anand/other), time of leave and coming back.

Output: Admin can see and accept the request.

3.2 Non-Functional Requirement

3.2.1 Usability

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

Efficiency of use:

- Users can easily interact with the system. Most of the tasks a user can complete withoutany help. It does not have a complex design and as a result, any user can easily interactwith the system.

3.2.2 Reliability

- Reliability defines how likely it is for the software to work without any failure for a given period of time. Reliability decreases because of bugs present in the code, hardware failures, or problems with other system components.

- The Database update process must roll back all related updates when any update fails.

3.2.3 Performance

- Performance is a quality attribute that describes the responsiveness of the system to various user interactions with it.
- The front-page load time must not be more than 5 seconds.

3.2.4 Security

- Security requirements ensure that the software is protected from unauthorized access to the system and its stored data. For instance, data privacy is a security characteristic that describes who can create, see, copy, change, or delete information.
- Unauthorized users cannot login to the system.

3.3 Design Constraints

3.3.1 No cost for cloud:

- Our client has refused to pay monthly for cloud, so we have to design accordingly.

3.3.2 Software must not have any dependency and must be lightweight:

- Software must not have any dependency so can be used in any system.
- It must be light weight so students and admin don’t have to wait much or face any difficulty in their work.