SOFTWARE REQUIREMENTS SPECIFICATION

for

HOSTEL MANAGEMENT SYSTEM

Prepared by :
1. Nilay Ganvit (200001053)
2. Rishi Parasi (200001068)
3. Sanskar Verma (200001069)
4. Saurabh Kumar Singh (200001071)
5. Vipul Mahajan (200001080)

Submitted to :Dr. Puneet Gupta Sir
Coordinator
Lokendra Birla Sir, Rupesh Kumar Sir
Teaching Assistant

February 16, 2022

Contents

1 Introduction

1.1 Purpose

Manually checking room availability, allocating rooms to incoming students, maintaining hostel fees, visitor records, mess, and other hostel management operations require time, effort, and resources. Furthermore, hostel management necessitates ensuring the safety of students, which authorities can only accomplish by closely monitoring everyday student activities.

The hostel management system is programmed to manage all hostel activities such as admissions, fees, room assignments, mess allotment, hostel stores, and meaningful information for easy payments. It is used to keep account of monthly mess bills, hostel staff salary, and student certificates, among other things.

Hostel Management Software is a one-stop solution for higher education institutions, designed and developed with all of the hostel management issues of employees in mind. It streamlines and automates all day-to-day hostel activities, from student registration to allocating rooms and mess, monitoring student count records and ins-and-outs of students.

1.2 Intended Audience and Reading Suggestions

This SRS is for developers, project managers, users and testers. Further the discussion will provide all the internal, external, functional and also non-functional information about ”Hostel Management System”.

Project Manager	To manage all the processes in the project
Software Designer	To design the models and diagrams that helps the programmer in implementation phase
Software Tester	To test data by applying dummy data
Database Administrator	To perform database operations
Software Analyst	To analyze the requirements of hostel Management System

1.3 Project Scope

The software product ”Hostel Management System” will be an application that will be used for maintaining the records in an organized manner and to replace old paper working on the system. This projects aims at automating the hostel management for smooth working of the hostel by automating almost all the calculations and accounting work would be accurate. It is designed for hostel like (Schools and Universities). It checks the attested application form of students obtained from the internet verify it with student database. If the students are found eligible then they allotted to the hostel room.

1.3.1 Objectives of HMS

The basic HMS objectives as follows:
1. Room Allocation
2. Bill Generation
3. Maintaining Student's Records
4. Provide to student's Complaints.
5. Maintaining Employee Records.

1.3.2 Goals of HMS

The goals of the system are to accept these problems in an effective and optimal manner by:
1. Centralizing the database and thus providing consistent data to all the employees in the Hostel.
2. Make the system more user friendly by providing an intensive user interface.
3. Easy access through reports.
4. Restricted data access to employees thus providing additional security to data.

2 Overall Description

2.1 Product Perspective

"Hostel Management System" is the replacement of the manual hard copy result process. The data have been stored in the hard file or papers, this website will store all of those in the website. Main goal of this project is to minimize the work and maximize the result of this result processing system.

2.2 User Classes and Characteristics

"Hostel Management System" has basically 3 types of users.

- Helper Staff
- Students
- Official Staff

2.3 Product Functions

1. It should allow admins to add hostel students by creating their accounts by adding information like name, hostel, course, email, branch, mobile no., room no. and managing their fees.
2. It should allow admin to add hostel staff by making their accounts, editing their information and managing their pay.

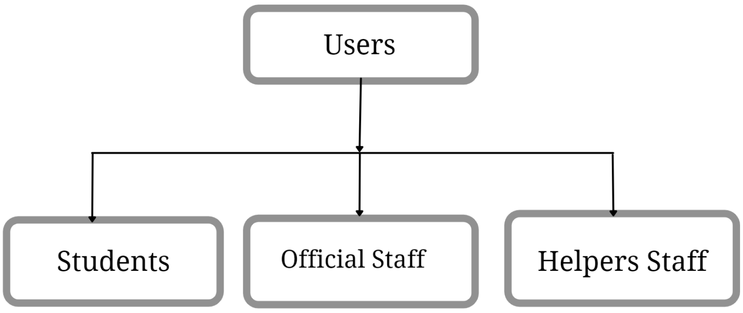

Figure 2.1: type of users

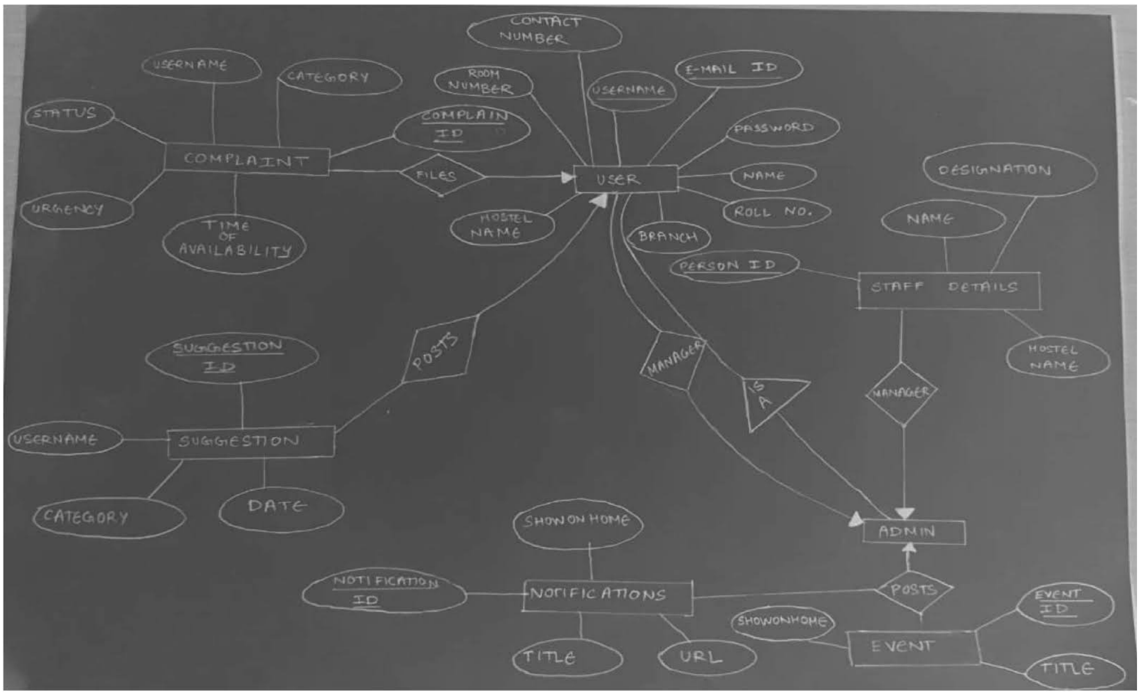

Figure 2.2: Data Flow Diagram

3. It should allow the admins to access profiles and view information about the staff and the students.
4. It should allow admins to add and remove other admins and manage them.
5. It should allow students, staff and admins to view and edit their information and profile.
6. It should allow students to file complaints while having the option to state their urgency.
7. It should allow the students to give suggestions to the hostels and management while also categorising them.
8. It should allow the students and the admin to check the due states of the hostel rent and any other corresponding charges.
9. It should allow the admin to renew the students' registration every year quickly.
10. It should allow push notifications onto the site and students whenever their due date is close or has passed by.
11. It should have the functionality to verify users based on their email and by sending an OTP.
12. It should represent all the hostels and necessary links to other sites.

2.4 Operating Environment

The website will be operate in any Operating Environment local or hosted with a browser Mac, Windows, Linux etc.

2.5 Design

Student activities have 3 steps -

- File complaint
- Give suggestion
- Student Profile
- Make, verify, manage account

Allow selected Student first fill his/her form. After verification, student verifies their account using email and OTP. Then he can enter his profile.
Every student profile contains his/her personal information, hostel,room no, charges and notice.
Notice will contain all the news of Hostel Management System.

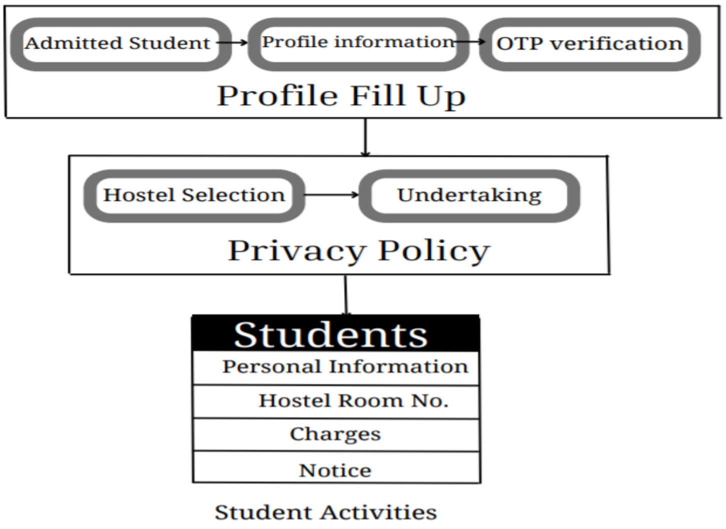

Figure 2.3: Student Activities

Official Staff activities have 2 steps -

- add/manage students
- add/manage helper staff

Official Staff includes hostel warden and other authorities who are in charge of whole hostel and it's functioning.
Official Staff can create students and create notice.

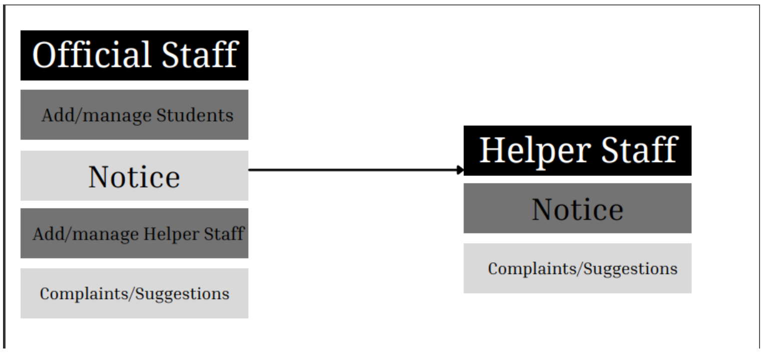

Figure 2.4: Staff Activities

3 System Features

”Hostel Management System” is a result processing web software. So the main art of this product is to mangae students living in hostels.

3.1 Description and Priority

”Hostel Management System” has features that are main and also some are sub. But all the feature is necessary for this software.
The features with priority up to down -

1. Managing Students : This is the goal feature of this software. It is been operated by Official Staff. They manage the students, their hostels and any related stuff.
2. Complaints : Students can file complaints and helper staff and Official staff can see and manage them.
3. Managing Admins : Admins can add/remove other admins and change their profiles
4. Managing Helper staff: Admins can manage helper staff and helper staff can change their own profiles.
5. Student : Students can change their own profiles and view info and status of suggestions and complaints.
6. Notice : notice can be posted by Official and helper staff and all can see them.

3.2 Functional Requirements

The ”Hostel Management System” website is being build on Node.js, express, EJS , MySQL, MongoDB, HTML, CSS, Bootstrap.
Back-End - Express, EJS, MongoDB, Mongoose, MySQL.
Font-End - HTML, CSS, Bootstrap.
Database - MongoDB, MySQL.

4 Non-Functional Requirements

- **Scalability:** The system should be usable for many students and multiple hostels.
- **Reusability:** The system should be easily be replicated and reused for other campuses or Colleges.
- **Security:** The system should be secure and prevent tampering or messing with data without permission.
- **Portability:** The system should work with different devices, operating systems, versions of libraries and software.
- **Maintainability:** The system should be easily maintainable with the proper backing of database management software and should maintain integrity with changes in users and other software dependencies.
- **Efficiency:** The system should work on decent hardware, use fewer resources, be snappy, and adhere to time constraints.
- **Readable and Comprehensible:** The code should be neat, clean and made easy to read and edit by other developers through comments.
- **Easy-to-use:** The software should be reasonably easy to use with interactive UI and rigid structures.
- **Flexibility:** provisions to use different datasets and database management systems if needed.
- **Testability:** The software can be tested with data before being deployed to check if the verification and security features are working as intended.

5 Table

5.1 Use Cases and Scenarios of Administrator

Name	**Add Name**
Description	Scenario Describes the process to add the student.
Actor	Administrator/Student/Database Administrator
Precondition	To add the student in Database record of hostel.
Procedure	1. The administrator login with database. 2. If the Login successful then he checks the availability of room in hostel. 3. The administrator add the student in database. 4. The student will receive the room number.
Outcome	The student has added in database of hostel.

Name	**Login with database**
Description	The scenario describes the process of login with database.
Actor	Administrator/Student/Database Administrator/system
Precondition	To login with system or database of hostel.
Procedure	1. The user boot the system. 2. The user enter password. 3. If the password is valid the user login with database. 4. The user perform his/her tasks.
Outcome	The user has logged with database or system.

Name	**Create Account number**
Description	The scenario describes the process of create account.
Actor	Administrator/Student
Precondition	To create a student account number.
Procedure	1.The administrator login with database. 2.If login successful then create account number. 3.The account number is allocated to student as id card.
Outcome	The account has created and allocated to the student.

Name	**Room Allocation**
Description	The scenario describes the process of room allocation.
Actor	Administrator
Precondition	To allocate the room to students.
Procedure	1.The administrator login with database. 2.1f login successful then check the availability of room. 3.1f the room is available then allocate the room and room number to the student. 4.Successfully room has allocated to the student.
Outcome	The room has allocated to the student.

Name	**Generate student profile**
Description	The scenario describes the process of generation of student profile.
Actor	Administrator/Database administrator
Precondition	To generate the student profile.
Procedure	1.The administrator login with database. 2.1f login successful then check the availability of room. 3.1f the room is available then allocate the room and room number to the student. 4.Successfully room has allocated to the student.
Outcome	The student profile has successfully created and password is allocated to the student.

Name	**Delete Student**
Description	The scenario describes the process of delete the student from database.
Actor	Administrator/Database administrator
Precondition	To delete the student from database or hostel
Procedure	1.The administrator login with database. 2.1f login successful then database administrator searches the student. 3.1f the record of student is appears then checks his/her dues. 4. The database administrator delete the student from system or database.
Outcome	The student has deleted from hostel.

Name	**Change the hostel dues of student.**
Description	The scenario describes the changing of hostel dues of student
Actor	Administrator/Database administrator
Precondition	To Change the hostel dues of students if the student pay his/her payment.
Procedure	1.The administrator login with database. 2. Checks the profile and hostel dues report and dues current status. 3.If the student has payed the bill then the dues will updated.
Outcome	The dues of student has changed or updated by administrator.

5.2 Use Cases and Scenarios of Student

Name	**Request For Room**
Description	This scenario describes the process for requesting the room.
Actor	Student/Administrator
Precondition	To request for room allocation.
Procedure	1.The student request for room to the administrator. 2.The administrator open the room detail. 3.If the room is available then the request is full filled.
Outcome	The request has been submitted.

Name	**Pay Bill**
Description	This scenario describes the process of bill payment of hostel room.
Actor	Student/Database Administrator
Precondition	The student wants to pay the bill of hostel room.
Procedure	1.The student login for bill payment. 2.The database administrator search the student and open the student profile. 3.The database administrator enters the fee bill payment detail. 4.the student pays the bill.
Outcome	The user has logged with database or system.

Name	**Check profile**
Description	This scenario describes the checking of profile of student.
Actor	Student
Precondition	The student wants to check the profile.
Procedure	1.The student will enter his/her password for login. 2.The student login with profile. The profile page will open. 3.The student checks his /her profile. And also log out it.
Outcome	The student has checked his/her profile.

5.3 Use Cases and Scenarios of Database Administrator

Name	**Search Student In Database**
Description	The scenario describes that the DB administrator wants to search the student.
Actor	Data Administrator
Precondition	The administrator wants to search the student.
Procedure	1.The Administrator login with database. 2.the administrator open the profile page. 3.The administrator search the student in database. 4.1f the student found in the database then he performed his required task.
Outcome	The student has been searched successfully.

Name	**State Attendance**
Description	The scenario describes that the DB administrator wants to check the current state of attendance of the students.
Actor	Database Administrator
Precondition	The database administrator wants to state the attendance of the students.
Procedure	1.The Database administrator login with student profile. 2.He opens the attendance sheet and checks the attendance. 3.After the completion of his task he will long out the database.
Outcome	The DB administrator has been checked the state of the student attendance successfully.

Name	**Check the availability of room**
Description	The scenario describes the process of checking the availability of room.
Actor	Database Administrator
Precondition	The database administrator wants to checks the availability of room
Procedure	1.The database administrator log in with database. 2.Open the room detail 3.Checks the availability of room. 4.The database administrator long out the database.
Outcome	The database administrator has been check the availability of room successfully.