**Software Requirements Specification (SRS)**

**For Hostel Management System**

Prepared by:

Group members :-­­­

Nandan Murari SE22UARI110

Krishna Kalyan Thommandru SE22UARI192

**Table of Contents**

**Introduction**

1.1 Document Purpose

1.2 Product Scope

1.3 Intended Audience and Document Overview

1.4 Definitions, Acronyms, and Abbreviations

1.5 Document Conventions

1.6 References and Acknowledgments

**Overall Description**

2.1 Product Overview

2.2 Product Functionality

2.3 Design and Implementation Constraints

2.4 Assumptions and Dependencies

**Specific Requirements**

3.1 External Interface Requirements

3.2 Functional Requirements

3.3 Use Case Model

**Other Non-functional Requirements**

4.1 Performance Requirements

4.2 Safety and Security Requirements

4.3 Software Quality Attributes

**Other Requirements**

**Appendices**

**1. Introduction**

**1.1 Document Purpose**

This document provides a detailed Software Requirements Specification (SRS) for the Hostel Management System (HMS). The system is designed to manage hostel-related activities such as room allotment, student check-in/check-out, fee management, complaint handling, and staff management efficiently.

The purpose of this document is to define the functional and non-functional requirements of the system in a structured manner. It serves as a guide for developers, testers, stakeholders, and project managers to understand and implement the system correctly.

**1.2 Product Scope**

The Hostel Management System is designed to provide a centralized digital solution for managing hostel operations. The primary goals include:

Efficient room allocation for students based on availability and preferences.

Digital fee payment and tracking system for hostel dues.

Automated complaint management system for handling maintenance requests.

Staff management, including tracking duties of wardens, security personnel, and maintenance staff.

Integration with RFID or biometric-based access control for enhanced security.

This system will reduce paperwork, streamline operations, and improve transparency between students and hostel administration.

**1.3 Intended Audience and Document Overview**

This document is intended for the following audiences:

Developers: To understand the technical requirements.

Project Managers: To track progress and ensure requirements are met.

Testers: To validate the system functionalities.

Stakeholders (University/Hostel Administrators): To review the scope and features.

**1.4 Definitions, Acronyms, and Abbreviations**

HMS: Hostel Management System

RFID: Radio-Frequency Identification

**1.6 References and Acknowledgments**

IEEE SRS Template

University Hostel Management Guidelines

**2. Overall Description**

**2.1 Product Overview**

The Hostel Management System aims to automate and digitalize hostel operations. It will include a web-based application accessible by students, administrators, and staff members.

**2.2 Product Functionality**

The key functionalities of HMS include:

Student Registration & Profile Management

Room Allocation & Availability Tracking

Hostel Fee Management & Online Payments

Complaint Registration & Tracking

Staff Duty Roster Management

Access Control via RFID/Biometrics

**2.3 Design and Implementation Constraints**

The system must be web-based and support multiple devices.

Data security measures must be implemented (e.g., encryption for payment transactions).

The system should be scalable to accommodate multiple hostels.

**2.4 Assumptions and Dependencies**

Users must have a stable internet connection to access the system.

Database hosting will be managed by the university IT department.

**3. Specific Requirements**

**3.1 External Interface Requirements**

**3.1.1 User Interfaces**

The system will have a web-based UI with a dashboard for different user roles.

Students will interact via a mobile-friendly interface to manage their hostel records.

**3.1.2 Hardware Interfaces**

Integration with RFID/Biometric scanners for access control.

Payment gateway for online fee transactions.

**3.1.3 Software Interfaces**

The system will integrate with university student databases.

**3.2 Functional Requirements**

FR-1: Student Registration

Students shall be able to register with their student ID, name, email, and phone number.

FR-2: Room Allocation

Admin shall allocate rooms based on availability and student preferences.

FR-3: Complaint Management

Students shall submit complaints via the portal, and staff shall update resolution status.

FR-4: Fee Payment System

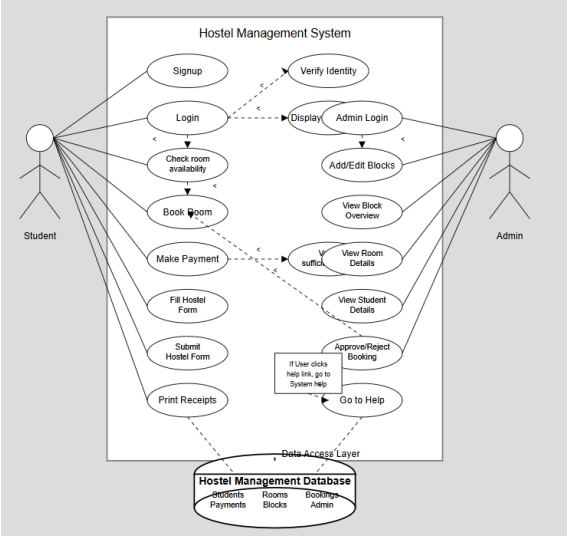
The system shall support UPI, Credit/Debit Card, and Net Banking for payments.

FR-5: Security & Access Control

Only authorized users shall access hostel facilities via RFID/Biometric authentication.

**3.3 Use Case Model**

A use case diagram will be attached illustrating actors and system interactions.



**4. Other Non-functional Requirements**

**4.1 Performance Requirements**

The system shall support simultaneous access by at least 500 users.

The system shall provide a response time of ≤ 3 seconds for page loads.

**4.2 Safety and Security Requirements**

SSL encryption shall be used for secure transactions.

Data backups shall be taken daily to prevent loss.

**4.3 Software Quality Attributes**

Reliability: The system must be 99.5% uptime.

Maintainability: The codebase should follow modular programming principles for easy updates.

**5. Other Requirements**

The system must comply with university policies on student data privacy.

**6. Appendices**

**Appendix A: Data Dictionary**

A table defining database attributes, constraints, and relationships will be included.

**Appendix B: Group Log**

A record of team meetings, discussions, and development progress will be maintained.