

HostelSphere+ - Hostel Management System

INTRODUCTION

- HostelSphere+ is a Full Stack Web Application developed to digitize and streamline hostel management activities within an educational institution.
- Traditional hostel operations such as attendance tracking, complaint handling, room management, and meal scheduling are often handled manually, leading to inefficiencies, data inconsistency, and lack of transparency.
- This project aims to **centralize hostel operations** into a single web-based system where administrators can manage hostel-related activities efficiently and students can access essential services through a secure login.
- HostelSphere+ improves accountability, reduces paperwork, and ensures smooth communication between hostel administration and students.

ABSTRACT

- HostelSphere+ – Hostel Management System is designed to automate core hostel operations using modern web technologies. The system provides **role-based access** where administrators manage attendance, complaints, meals, and rooms, while students can view attendance records, raise complaints, and access meal schedules.
- The application is developed using **Spring Boot** for backend services, **Spring Security** for authentication and authorization, **MySQL** for persistent data storage, and **Thymeleaf with Bootstrap** for frontend rendering. RESTful APIs are tested using **Postman**, and secure password handling is implemented using **BCrypt encryption**.
- The system ensures data integrity, secure access, and an organized workflow, making hostel administration more efficient and reliable.

CLIENT REQUIREMENTS

The client requires a **secure and scalable Hostel Management System** with the following features:

- Admin should be able to manage students, rooms, meals, attendance, and complaints.
- Students should be able to view their own attendance and meal schedules.
- Students should be able to raise complaints and track their status.
- Attendance must be **date-wise** and **role-restricted**.
- Complaints should follow a **status-based workflow**.
- Login and registration must be secure with role-based dashboards.

SYSTEM FEATURES

Admin Capabilities

- Create, update, view, and delete student records.
- Mark attendance for students' date-wise.
- View and resolve student complaints.
- Manage meal schedules.
- Maintain room data.
- Access Admin Dashboard.

SYSTEM FEATURES

Student Capabilities

- Login securely.
- View personal attendance records.
- Raise complaints and track complaint status.
- View meal schedules.
- Access Student Dashboard.

TECHNICAL REQUIREMENTS

- Secure Login & Registration
- Role-based Authentication (ADMIN / STUDENT)
- RESTful APIs
- Proper Exception Handling
- Input Validation
- Password Encryption using BCrypt
- Modular Architecture
- Backend Deployment Ready

TECHNOLOGIES & TOOLS USED

1. Backend

- Java
- Spring Boot
- Spring MVC
- Spring Data JPA (Hibernate)
- Spring Security

2. Frontend

- Thymeleaf
- HTML
- CSS
- Bootstrap 5

TECHNOLOGIES & TOOLS USED

3. Database

- MySQL

4. Tools

- Spring Tool Suite (STS)
- MySQL Workbench
- Postman
- Git & GitHub

SYSTEM REQUIREMENTS

Software Requirements

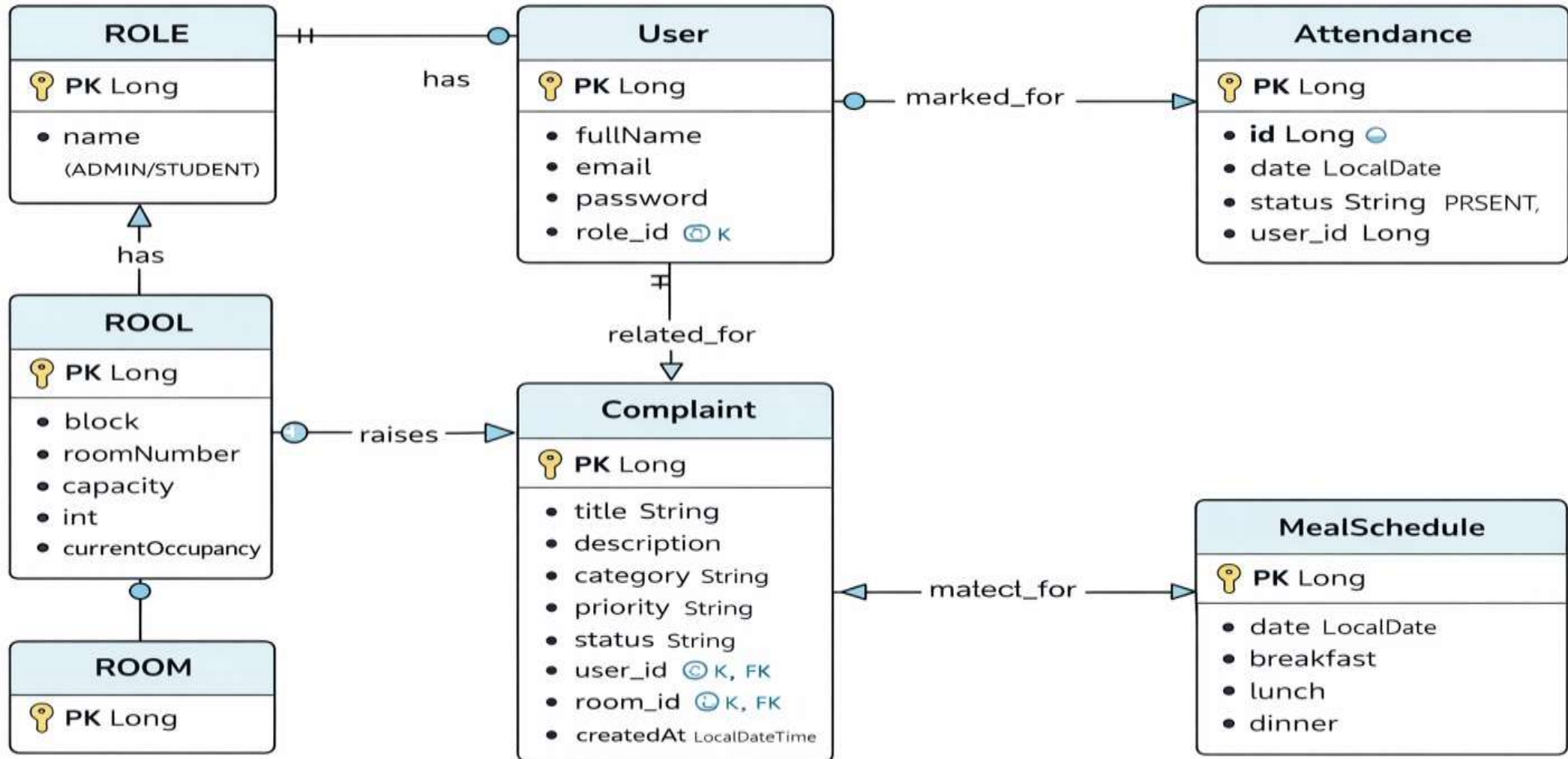
- Operating System: Windows / Linux
- Backend: Spring Boot
- Database: MySQL Server
- Web Server: Apache Tomcat (Embedded)
- IDE: Spring Tool Suite (STS)
- Browser: Google Chrome
- Testing Tool: Postman

PROJECT MODULES

- Admin Module
- Student Module
- Attendance Module
- Complaint Module
- Meal Management Module
- Room Management Module
- Authentication & Authorization Module

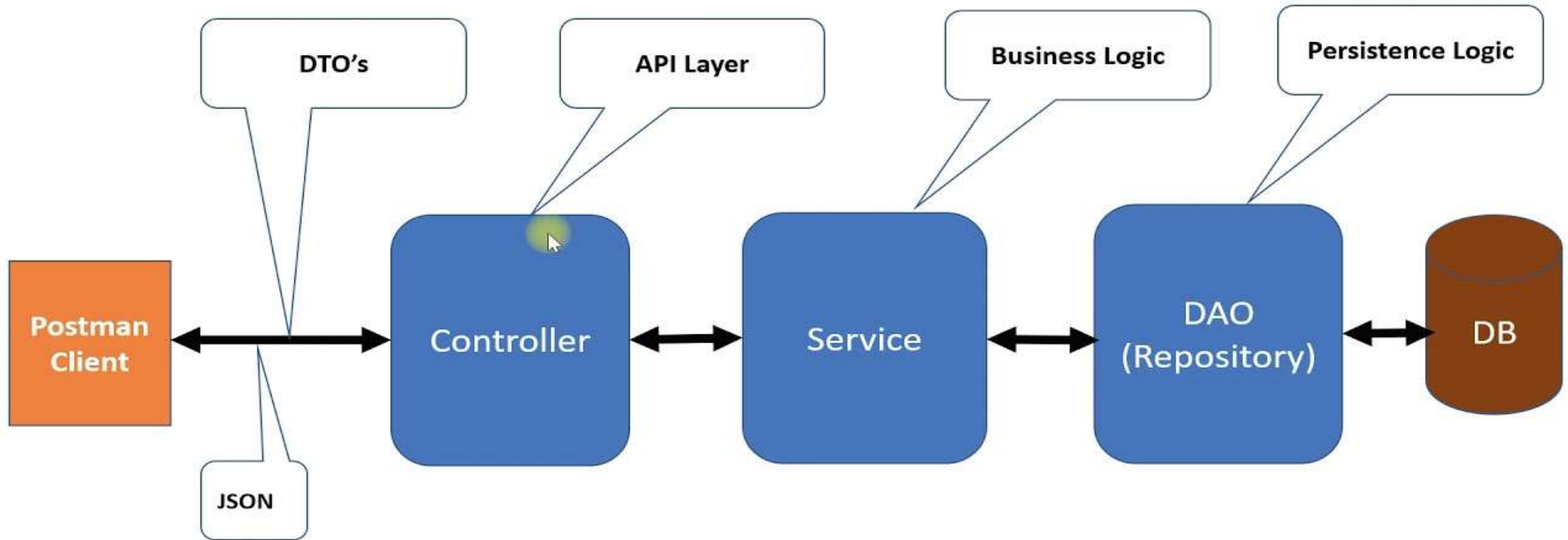
ER DIAGRAM

HostelSphere+ – ER Diagram



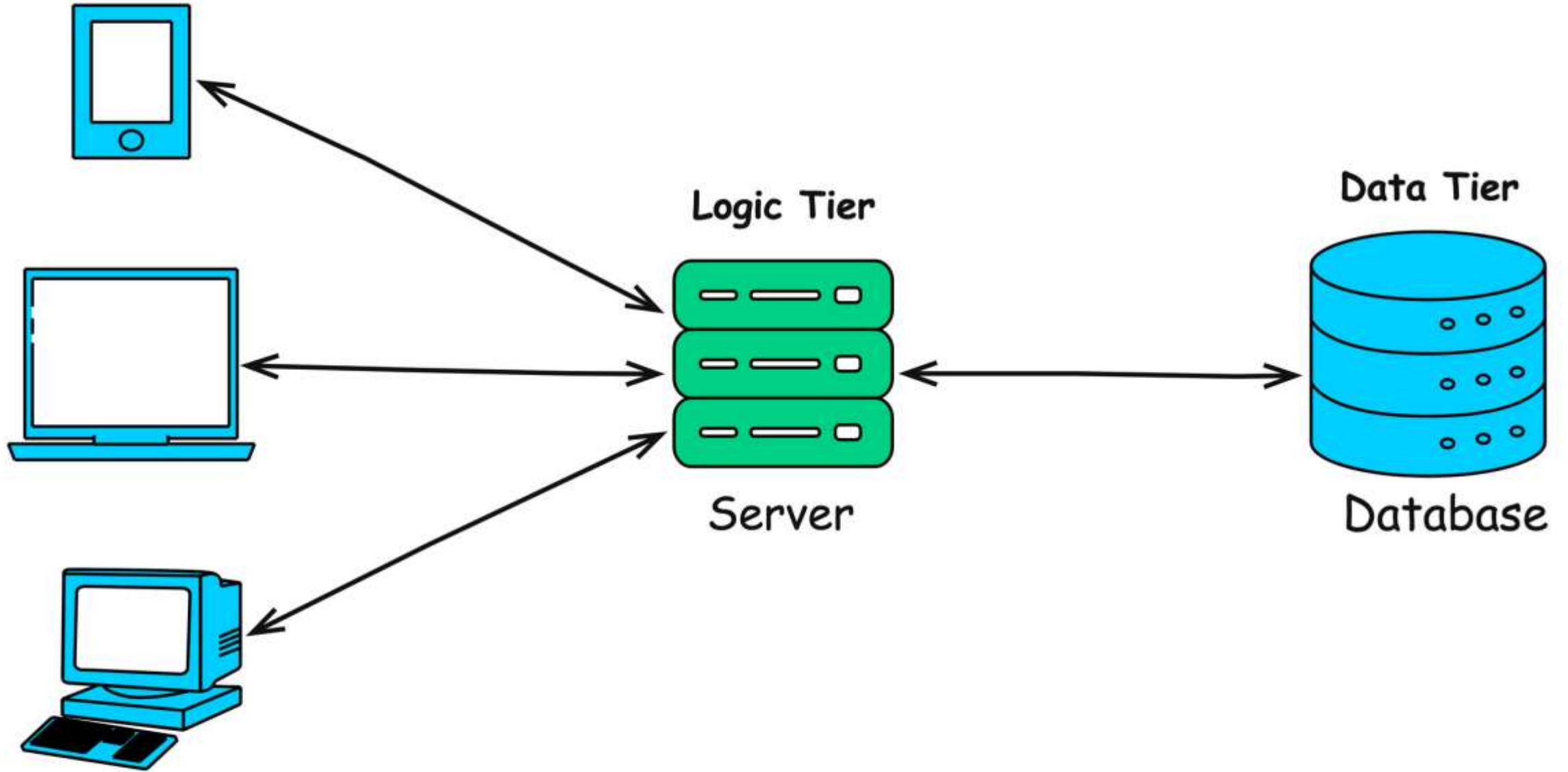
CLIENT-SERVER ARCHITECTURE

Spring Boot Application Architecture



Presentation Tier

Three-Tier Architecture



Clients

ADMIN MODULE

- Admin logs in securely using credentials.
- Admin performs CRUD operations on student data.
- Admin marks attendance date-wise.
- Admin manages meal schedules.
- Admin views and updates complaint statuses.
- Admin dashboard provides centralized control.

STUDENT MODULE

- Student logs in using registered credentials.
- Student views personal attendance records.
- Student raises complaints.
- Student checks complaint status.
- Student views meal schedule.
- Student dashboard shows personalized data.

ATTENDANCE MODULE

- Attendance is marked **only by Admin**.
- Attendance is recorded date-wise.
- Students can **only view their own attendance**.
- Role-based access ensures data privacy.

COMPLAINT MANAGEMENT MODULE

- Students raise complaints from dashboard.
- Complaints have statuses:
 - Pending
 - In Progress
 - Resolved
- Admin updates complaint status.
- Complete complaint lifecycle tracking.

MEAL MANAGEMENT MODULE

- Admin creates and updates meal schedules.
- Students can view daily meal timetable.
- Ensures transparency in hostel food planning.

ROOM MANAGEMENT MODULE

- Admin manages room details in backend.
- Room data is associated with students.
- Supports structured hostel allocation.

DATABASE DESIGN

The database consists of structured tables for:

- Admin
- Student
- Attendance
- Complaint
- Meals
- Rooms

Relationships are maintained using foreign keys to ensure data integrity.

SECURITY IMPLEMENTATION

- Spring Security for authentication
- Role-based authorization
- BCrypt password encryption
- Unauthorized access restricted

TESTING

- APIs tested using Postman
- Validated CRUD operations
- Authentication & authorization tested
- Role restrictions verified

ADVANTAGES OF THE SYSTEM

- Centralized hostel management
- Secure access
- Reduced paperwork
- Improved transparency
- Time-efficient administration
- Scalable architecture

FUTURE ENHANCEMENTS

- Notification system
- Mobile application support
- Online fee payment
- Report generation
- Parent access module