

# Hostel Management System

## Software Project Documentation

# Table of Contents

1. [Project Overview](#)
2. [System Features](#)
3. [User Roles & Permissions](#)
4. [System Architecture](#)
5. [Technology Stack](#)
6. [Database Design Overview](#)
7. [Deployment Guide](#)

## 1. Project Overview

**Project Name:** Hostel Complaint Management System

**Client:** Educational Institutions (Hostels)

**Developed By:** PYDAH SOFT Team

**Version:** 1.0

### Purpose

The Hostel Complaint Management System is a full-stack web application designed to digitize and streamline hostel operations. It addresses key challenges such as complaint tracking, student leave management, room allocation, attendance monitoring, and administrative oversight. The system is built to support multiple user roles and provide real-time updates, analytics, and notifications.

### Goals

- Improve hostel operational efficiency
- Enable real-time complaint tracking and resolution
- Provide analytics for decision-making
- Enhance communication between students, wardens, and administrators

### Scope

The system covers all major hostel operations including student registration, room management, complaint handling, leave requests, attendance tracking, announcements, and polls. It is designed for scalability and can be deployed across multiple hostels or institutions.

## 2. System Features

---

### Authentication & Role Management

- JWT-based Authentication: Secure login using JSON Web Tokens with expiration and refresh mechanisms.
  - Role Hierarchy: Supports six distinct roles:
    - Super Admin: Full system control, user creation, permission assignment
    - Sub Admin: Module-specific access (e.g., student management, billing)
    - Warden: Student oversight, attendance, leave request review
    - Principal: Approvals for stay requests, attendance monitoring
    - Student: Raise complaints, apply for leave, view personal data
    - Security: Verify gate passes and student check-in/out
  - Session Management: Token expiration, logout, and secure session handling
  - Protected Routes: Frontend routes are guarded based on user roles
  - Permission Control: Admins can assign or revoke access to specific modules or actions
- 

### Student Management

- Bulk Registration: Upload student data via Excel with validation and preview
- Self-Service Portal: Students can update profiles, reset passwords, upload photos
- Batch Management: Group students by academic year, course, and branch

- Photo Management: Upload and store student and guardian photos in AWS S3
- 

## Room Management

- Room Categories: A+, A, B+, B, C based on amenities and pricing
  - Gender-based Allocation: Separate room pools for male and female students
  - Bed Configuration: Define number of beds per room
  - Electricity Billing:
    - Monthly bill generation
    - Preview before final submission
    - Bill history with edit and delete options
  - Student-Room Mapping: Assign and track which students are in which room
  - Room Occupancy Reports: View vacant and occupied rooms
- 

## Complaint Management

- Complaint Categories:
  - Canteen
  - Internet
  - Maintenance (Housekeeping, Plumbing, Electricity)
  - Others
- Status Workflow:
  - Received → Pending → In Progress → Resolved → Closed
- Staff Assignment: Auto-assign complaints based on category to available staff

- Student Feedback: Students can rate and comment on resolved complaints
  - Reopen Complaints: Students can reopen unresolved issues
  - Analytics Dashboard:
    - Complaint trends by category
    - Pending vs resolved ratio
    - Member performance heatmaps
- 

## **Member Management**

- Category-wise Assignment: Assign staff to handle specific complaint types
  - Performance Metrics:
    - Number of complaints handled
    - Average resolution time
    - Student feedback score
  - Minimum Member Rule: System prevents deletion if fewer than 2 members are assigned to a category
  - Workload Distribution: Visual indicators of member workload for rebalancing
- 

## **Announcements & Polls**

- Announcement Module:
  - Create, edit, and delete announcements
  - Schedule announcements for future dates
  - Target by role (e.g., only students, only wardens)
- Polling System:

- Create polls with multiple options
    - Real-time voting and result visualization
    - Poll expiry and result locking
  - Engagement Tracking: View how many students viewed or voted
- 

## Leave Management

- Leave Types:
    - Leave: Long-term absence (e.g., vacation)
    - Permission: Short-term outing (e.g., evening out)
    - Stay in Hostel: Request to remain during holidays
  - Approval Workflow:
    - Leave → Warden → Principal
    - Permission → Warden only
    - Stay in Hostel → Warden recommends → Principal approves
  - QR Code Generation: For secure gate verification
  - OTP Verification: For permission requests
  - Gate Pass Integration: Security staff can scan and validate
- 

## Warden Management

- Bulk Outing Management:
  - Select multiple students and create a single outing request
  - Track outing status and history
- Stay in Hostel Requests:

- Review and recommend student requests
    - Add remarks and supporting documents
  - Attendance Management:
    - Take morning and evening attendance
    - View attendance history by date or student
  - Student Oversight:
    - Filter students by course, branch, gender, room, category
    - View student status (Active/Inactive)
  - Notifications: Receive alerts for complaints, leave requests, and announcements
- 

## **Principal Management**

- Stay in Hostel Approval:
    - View recommended requests from wardens
    - Approve or reject with comments
  - Attendance Monitoring:
    - View attendance reports
    - Filter by date, session, or batch
  - Student Oversight:
    - Access to all student profiles and activity logs
  - Dashboard Access:
    - Summary of pending approvals
    - Notifications for pending actions
-



## Attendance Management

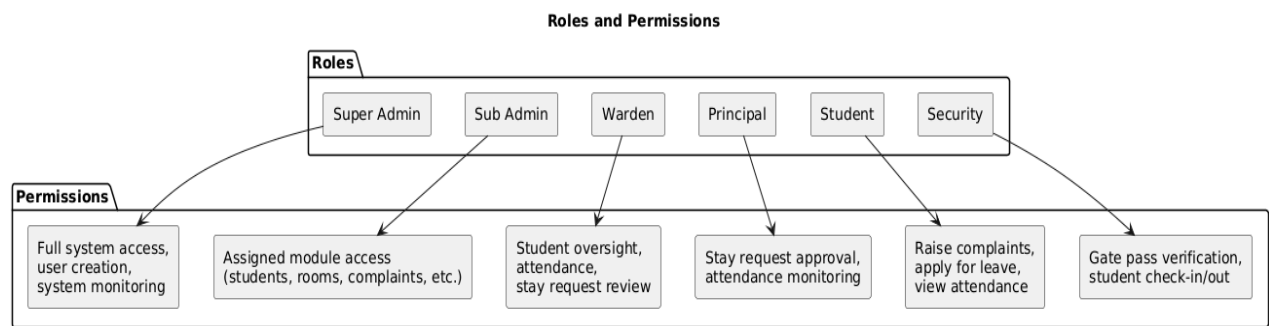
- Role-based Access:
    - Wardens, Principals, and Admins can take attendance
    - Students can only view their own records
  - Dual Sessions:
    - Morning and evening attendance
  - Attendance Analytics:
    - Daily, weekly, and monthly reports
    - Absentee trends and alerts
  - Real-time Sync:
    - Updates reflect instantly across all dashboards
- 

## Notification System

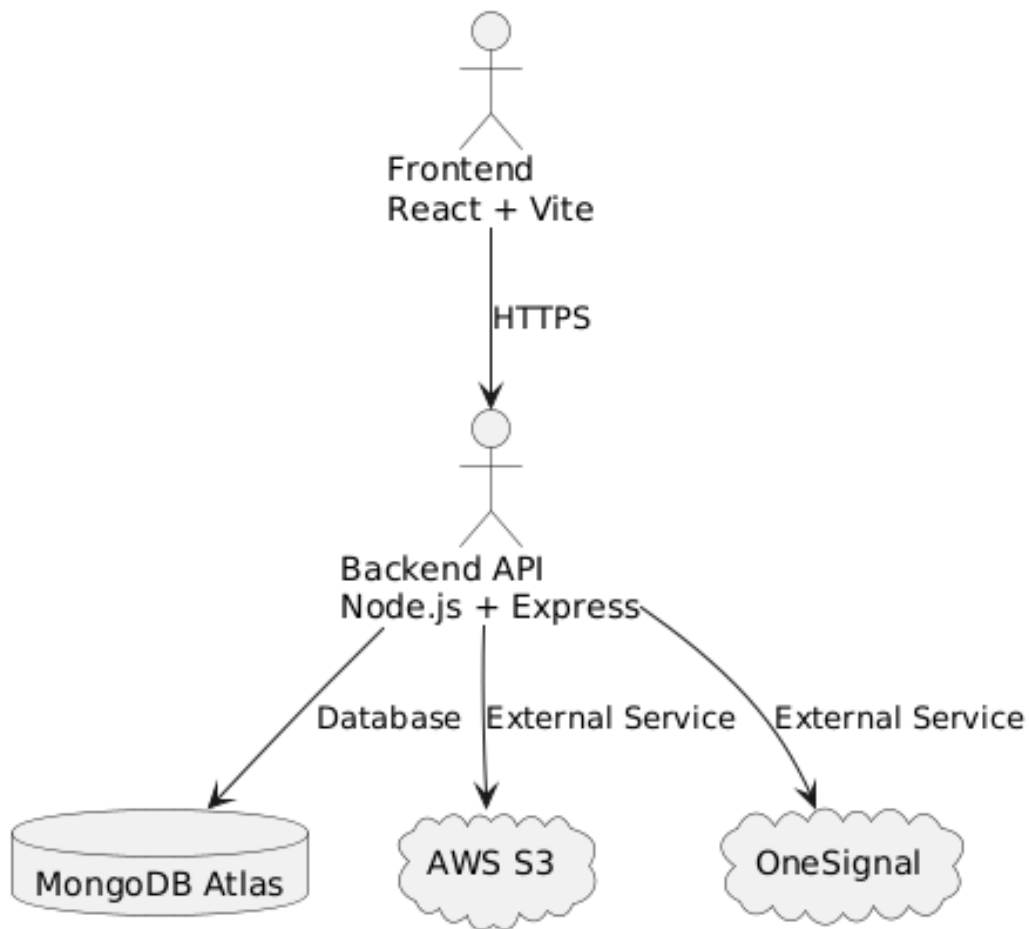
- OneSignal Integration:
  - Cross-platform push notifications
  - Targeted notifications by role or user
- Socket.IO:
  - Real-time in-app notifications
- Multi-channel Support:
  - Push, in-app, and email (future)
- Notification History:
  - Log of all notifications with timestamps
  - Mark as read/unread

### 3. User Roles & Permissions

Role	Permissions
Super Admin	Full system access, user creation, system monitoring
Sub Admin	Assigned module access (students, rooms, complaints, etc.)
Warden	Student oversight, attendance, stay request review
Principal	Stay request approval, attendance monitoring
Student	Raise complaints, apply for leave, view attendance
Security	Gate pass verification, student check-in/out



## 4. System Architecture



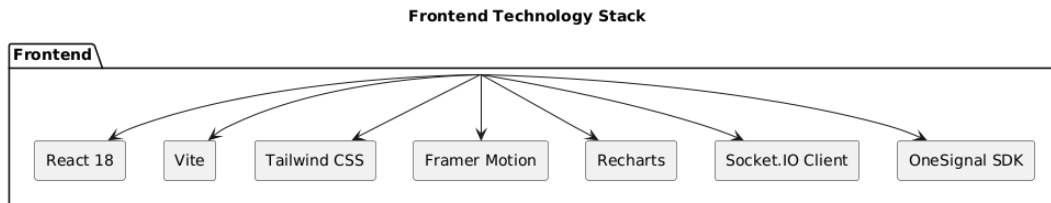
### Key Components

- Frontend: React with Tailwind CSS, Framer Motion, Recharts
- Backend: RESTful API with Express, JWT auth, Socket.IO
- Database: MongoDB with Mongoose schemas
- File Storage: AWS S3 for photos and documents
- Notifications: OneSignal for push notifications

## 5. Technology Stack

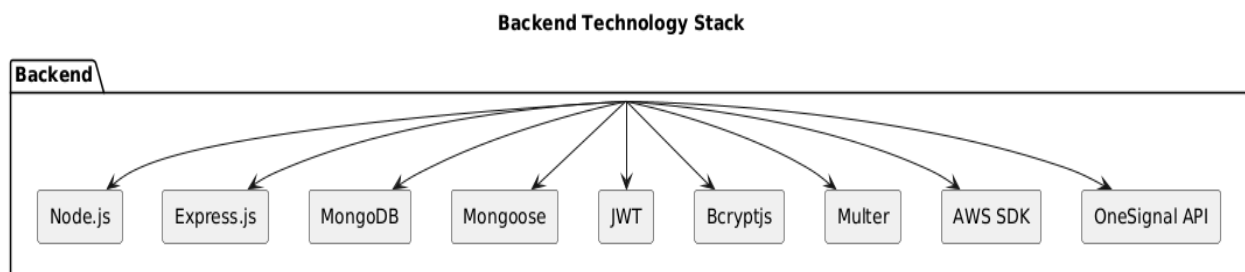
### Frontend

- React 18, Vite, Tailwind CSS, Framer Motion, Recharts, Socket.IO Client, OneSignal SDK



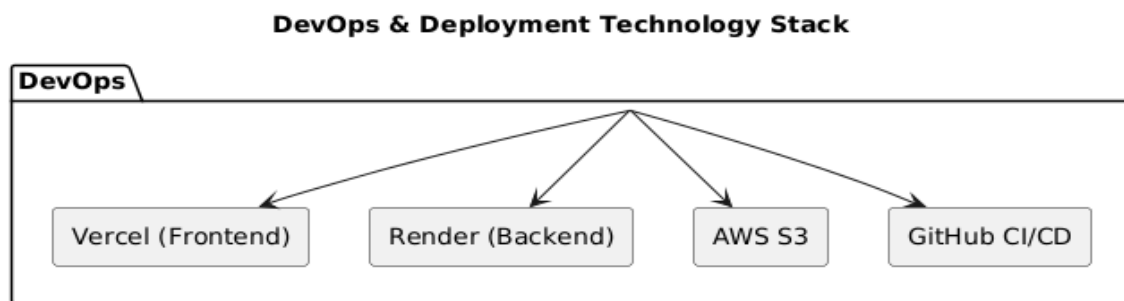
### Backend

- Node.js, Express.js, MongoDB, Mongoose, JWT, Bcryptjs, Multer, AWS SDK, OneSignal API



### DevOps

- Vercel (Frontend), Render (Backend), AWS S3, GitHub CI/CD



## 6. Database Design Overview

### ✓ Key MongoDB Collections

Collection Name	Description
users	Stores all user profiles (students, wardens, principals, admins, security)
complaints	Stores complaint details, status, category, and assigned staff
rooms	Room details, bed capacity, category, and assigned students
electricitybills	Room-wise electricity bill records with billing history
leaves	Leave, permission, and stay-in-hostel requests with approval workflow
attendance	Daily attendance records (morning/evening) for students
announcements	Admin announcements with scheduling and status
polls	Polls with options, votes, and results
notifications	Push/in-app notification logs with read/unread status
members	Staff members assigned to complaint categories
bulkoutings	Bulk outing requests created by wardens
menus	Hostel meal menu records

## 7. Deployment Guide

### Prerequisites

- Node.js 18+
- MongoDB URI
- AWS S3 credentials
- OneSignal App ID & API Key

### Steps

1. Clone repo
2. Install dependencies (`npm install`)
3. Configure `.env` files (client & server)
4. Run `npm run dev` for local testing
5. Build and deploy frontend (Vercel)
6. Deploy backend (e.g., Railway, Render, Heroku)