Table of Contents

1. INTRODUCTION

1.1 Introduction

The *Hostel Management and Food Surplus Distribution Platform* is a comprehensive web-based system designed to modernize and streamline hostel administration processes in educational institutions. The platform digitizes essential hostel operations such as room allotment, student management, complaint handling, staff coordination, and parent monitoring through a structured, role-based access system.

In addition to hostel management, the system integrates a *Food Surplus Distribution module* aimed at promoting sustainability and social responsibility. This module enables mess managers to record surplus food availability and facilitates its redistribution to registered NGOs, thereby reducing food wastage and supporting community welfare initiatives.

By leveraging modern web technologies and a role-driven architecture, the platform bridges communication gaps between hostel authorities, students, parents, staff, and NGOs. It enhances transparency, improves operational efficiency, and ensures controlled access to information based on user roles. Overall, the system serves as a unified digital solution that supports efficient hostel administration while contributing positively to environmental sustainability and social impact.

1.2 Problem Statement

Traditional hostel management systems in many educational institutions rely heavily on manual processes and fragmented digital tools, leading to inefficiencies, lack of transparency, and poor coordination among stakeholders. Room allotment is often error-prone, complaint handling is slow due to unclear responsibility tracking, and communication between students, staff, wardens, and parents is limited. Additionally, parents usually lack real-time visibility into their ward’s hostel status, which affects trust and accountability.

Another significant challenge is the absence of a structured mechanism to manage surplus food generated in hostel messes. Large quantities of edible food are often wasted due to the lack of coordination between mess management and social welfare organizations. Existing systems do not provide real-time information about surplus availability, pickup scheduling, or distribution tracking.

The lack of an integrated, role-based digital platform results in operational delays, poor data management, and missed opportunities for sustainability. Therefore, there is a strong need for a centralized, technology-driven solution that can efficiently manage hostel operations while enabling responsible food surplus redistribution. The proposed system aims to overcome these challenges by providing a transparent, secure, and scalable platform tailored to institutional hostel environments.

1.3 Objectives

The primary objectives of the *Hostel Management and Food Surplus Distribution Platform* are to:

- **Digitize hostel operations** by providing a centralized, role-based platform for managing room allotment, complaints, staff coordination, and student records efficiently.
- **Improve transparency and communication** among hostel stakeholders, including students, wardens, chief wardens, staff, and parents, through controlled and role-specific access to information.
- **Streamline complaint management** by enabling students to raise complaints digitally, allowing staff to update resolution status, and facilitating monitoring and escalation by wardens and the chief warden.
- **Enable effective food surplus management** by allowing mess managers to record surplus food availability and facilitating timely redistribution to registered NGOs.
- **Promote sustainability and social responsibility** by reducing food wastage and supporting community welfare through structured surplus food distribution.
- **Ensure system scalability and responsiveness** to accommodate multiple hostels, increasing numbers of users, and future feature enhancements without compromising performance.

1.4 Methodology

The development process for the *Hostel Management and Food Surplus Distribution Platform* involves the following steps:

- **Requirement Gathering:**
  Identify functional and non-functional requirements by analyzing hostel operations and stakeholder needs.

- **Technology Stack:**
  Use React for frontend development with Tailwind CSS to create a responsive and role-based user interface.

- **Development and Review:**
  Implement core modules such as room management, complaints, and food surplus distribution, followed by testing and iterative improvements.

1.5 Limitations

- Internet connectivity is required for accessing the system and real-time updates.

- The system depends on accurate data entry by hostel authorities and users for effective operation.

- The platform does not include automated decision-making or advanced analytics in its current version.

- User adoption may be slow in institutions with limited digital infrastructure or technical familiarity.

2 LITERATURE SURVEY

2.1 Need for Research

The increasing complexity of hostel administration in educational institutions highlights the need for efficient and technology-driven management systems. Traditional hostel operations often suffer from manual processes, delayed complaint resolution, limited transparency, and poor coordination among stakeholders. Additionally, the lack of structured mechanisms to manage surplus food in hostel messes leads to significant food wastage and missed opportunities for social contribution.

This research aims to address these challenges by proposing a comprehensive digital platform that integrates hostel management with food surplus distribution. By leveraging modern web technologies and role-based access control, the system seeks to improve operational efficiency, enhance transparency, and promote sustainability in institutional hostel environments.

2.2 Existing System

The existing hostel management practices in many institutions are largely manual and poorly integrated. Some of the major limitations include:

- **Manual Processes:**

  Hostel records such as room allotment, complaints, and staff details are often maintained manually, leading to errors and delays.

- **Lack of Real-Time Monitoring:**

  There is no real-time visibility into room occupancy, complaint status, or student information for wardens and parents.

- **Limited Role-Based Access:**

  Existing systems do not clearly differentiate access levels for students, staff, wardens, and parents, resulting in security and transparency issues.

- **Inefficient Complaint Handling:**

  Complaints are usually communicated verbally or through registers, making tracking and accountability difficult.

- **No Structured Food Waste Management:**

  Surplus food generated in hostel messes is often discarded due to the absence of a systematic redistribution mechanism.

2.3 Proposed System

The proposed *Hostel Management and Food Surplus Distribution Platform* overcomes the limitations of the existing system by providing a unified, role-based, and scalable digital solution. The key features of the proposed system include:

- **Role-Based Access Control:**

  A structured login system ensures secure and controlled access for chief wardens, wardens, staff, students, parents, mess managers, and NGOs.

- **Digital Room Management:**

  A fixed 6×6 room grid with color-coded occupancy provides real-time visibility of room allocation and availability.

- **Efficient Complaint Management:**

  Students can raise complaints digitally, staff can update resolution status, and wardens can monitor and escalate issues effectively.

- **Food Surplus Distribution Module:**

  Mess managers can record surplus food availability, and registered NGOs can request and track food pickup, reducing food wastage.

- **Scalability and Transparency:**

  The system is designed to support multiple hostels, increasing users, and future enhancements while maintaining transparency and operational efficiency.

Table 2.1 Existing Systems Features and Limitations

System	Key Features	Limitations	How Proposed System Overcomes
Manual Hostel Registers	Paper-based room allocation, complaint registers, staff logs	Error-prone, time-consuming, no real-time updates	Digital room grid, automated records, real-time status tracking
Basic Hostel Portals	Student information storage, ` limited online access	No role-based control, poor UI, limited modules	Role-driven dashboards, modern UI, integrated modules
Complaint Register System	Written or verbal complaint reporting	No tracking, delayed resolution, lack of accountability	Digital complaint system with status updates and monitoring
Mess Management (Manual)	Daily food preparation planning	No tracking of surplus food, high food wastage	Dedicated food surplus module with NGO integration
Parent Communication (Offline)	Phone calls, notices	No real-time visibility, poor transparency	Read-only parent dashboard with student and room details
Standalone Staff Systems	Separate staff coordination tools	Lack of integration, inefficient workflows	Unified staff management under warden control
No Surplus Redistribution System	Food disposal without planning	Social and environmental loss	Structured surplus food redistribution to NGOs

Table 2.2 API’s used in our Application

Category	Purpose	API / Service	Method / Details
General APIs (All Roles)	Login	/api/auth/login	POST
	Get own profile	/api/auth/me	GET
	View notices	/api/notices	GET
	Forgot password	/api/auth/password/send-code	POST
	Verify & reset password	/api/auth/password/verify-change	POST
Chief Warden APIs	Create warden	/api/auth/chief/warden	POST
	Create student	/api/auth/chief/student	POST
	Create mess manager	/api/auth/chief/mess-manager	POST
	View complaints (by hostel)	/api/complaints/chief?hostelId=	GET
	Complaint reports	/api/reports/complaints/summary	GET
	Chief dashboard	/api/dashboard/chief	GET
	Reset academic data	/api/admin/reset-academic	DELETE
Warden APIs	Create staff	/api/auth/warden/staff	POST
	View hostel rooms	/api/rooms/hostel/:hostelId	GET
	View unresolved complaints	/api/complaints/warden	GET
	Close complaint	/api/complaints/warden/close/:complaintId	PUT

Student APIs	Student dashboard	/api/dashboard/student	GET
	View room details	/api/dashboard/student/room	GET
	Create complaint	/api/complaints/create	POST
Staff APIs	View assigned complaints	/api/complaints/staff?hostelId=	GET
	Update complaint status	/api/complaints/update/:complaintId	PUT
Mess Manager APIs	Post surplus food	/api/mess/surplus/create	POST
	View surplus history	/api/mess/surplus/all	GET
Parent APIs	Parent dashboard	/api/dashboard/parent	GET
External Services	Email (OTP, reset)	Brevo	Transactional emails
	Image storage	Cloudinary	Complaint & notice images

3 TECHNOLOGIES USED

The Smart Hostel Management System leverages modern, scalable, and cloud-integrated technologies to deliver a secure, efficient, and user-friendly platform for managing hostel operations. For frontend development, React is used to build the web application, while React Native is used to develop a cross-platform mobile application that runs seamlessly on both Android and iOS devices. React provides a responsive and interactive web dashboard for roles such as Chief Warden, Warden, Staff, Mess Manager, and NGO, whereas React Native ensures a consistent and native-like experience for mobile users including Students and Parents. Tailwind CSS is utilized for UI styling, enabling rapid development of clean, responsive, and visually consistent interfaces through utility-based styling. The component-based architecture of React and React Native improves code reusability, simplifies UI maintenance, and ensures smooth navigation and consistent performance across all user roles.

The backend of the system is developed using Node.js with the Express.js framework, which enables the creation of a robust RESTful API architecture. Express.js efficiently handles routing, middleware, authentication, and business logic, ensuring smooth communication between the frontend and backend services. This backend structure supports role-based access control (RBAC), secure complaint management, room allocation, surplus food handling, and report generation.

For database management, MongoDB is utilized as a NoSQL database, allowing flexible and scalable storage of structured and semi-structured data. MongoDB efficiently manages complex relationships between entities such as hostels, rooms, users, complaints, notices, and surplus food records. Its document-oriented model is well-suited for dynamic hostel data and supports high performance even as the system scales across multiple hostels.

User authentication and authorization are implemented using JSON Web Tokens (JWT), ensuring secure session management and protected API access. Role-based authentication ensures that each user can only access features relevant to their role, thereby maintaining system integrity and data privacy. Passwords are securely stored using hashing mechanisms, and sensitive operations are safeguarded through middleware validation.

To support email-based communication, the system integrates Brevo (formerly Sendinblue) as an external email service provider. Brevo is used for sending password reset OTPs, verification codes, and system notifications, ensuring reliable and secure email delivery. This integration enhances user account recovery and improves overall system security.

For image handling and cloud storage, Cloudinary is employed to manage multimedia content such as complaint images, complaint resolution images, and notice images. Cloudinary provides secure cloud storage, automatic image optimization, and fast content delivery via CDN, eliminating the need for local file storage and improving application performance.

The system also incorporates scheduled background tasks using cron jobs, particularly for automatically handling surplus food expiry based on deadlines. This ensures that expired food cannot be claimed by NGOs, maintaining operational accuracy without manual intervention.

Development and testing are facilitated using tools such as Visual Studio Code for coding and debugging, Postman for API testing and validation, and Git/GitHub for version control and collaboration. These tools ensure reliable development workflows, efficient debugging, and seamless integration between frontend and backend components.

Overall, the chosen technology stack provides significant advantages. React enables the development of a responsive and interactive web application, while React Native supports cross-platform mobile development for Android and iOS. Node.js and Express.js offer a scalable and efficient backend architecture, MongoDB ensures flexible and schema-less data management, and JWT provides secure authentication and role-based access control. Brevo is used for reliable email communication and notifications, and Cloudinary manages cloud-based image storage and media handling. Together, these technologies form a powerful, scalable, and maintainable Smart Hostel Management System capable of effectively meeting real-world hostel administration requirements.

4 SOFTWARE REQUIREMENT SPECIFICATION

4.1 The Overall Description

The *Hostel Management and Food Surplus Distribution Platform* is a web-based, role-driven system designed to streamline hostel administration and promote sustainable food management within educational institutions. The system enables efficient room allocation, complaint handling, staff coordination, parent monitoring, and food surplus redistribution through a centralized digital platform.

By integrating modern frontend technologies and role-based access control, the platform ensures transparency, operational efficiency, and secure access for different stakeholders. The system is designed to be scalable, reliable, and user-friendly, making it suitable for institutional deployment and future backend integration.

4.2 Product Perspective

The *Hostel Management and Food Surplus Distribution Platform* is a standalone, web-based application developed using React and Tailwind CSS for a modern and responsive user interface. The system is designed as an integrated platform that manages hostel operations such as room allocation, complaint handling, staff coordination, and parent monitoring.

In addition, the platform incorporates a food surplus distribution module that connects mess managers with NGOs to reduce food wastage. Unlike traditional manual or partially digital systems, the proposed system offers a centralized, role-based solution with improved transparency, scalability, and accessibility.

4.3 Constraints

- The system requires a stable internet connection for accessing features and real-time updates.
- System efficiency depends on accurate data entry by authorized users.
- The platform is limited to frontend implementation in the current phase and requires backend integration.
- Advanced automation and analytics are outside the scope of the current system version.

4.4 Assumptions and Dependencies

- Users have access to internet-enabled devices such as computers or smartphones.
- Authorized users enter accurate and valid information into the system.
- Hostel authorities and staff actively use the platform for daily operations.
- Required frontend frameworks and development tools remain available and supported.

4.5 Functions

- **Room Management:**

  Enables digital room allocation and real-time occupancy display using a fixed 6×6 grid structure.

- **Complaint Management:**

  Allows students to raise complaints and enables staff and wardens to track and update complaint status.

- **User and Staff Management:**

  Supports creation and management of users and staff based on role-specific permissions.

- **Food Surplus Distribution:**

  Allows mess managers to record surplus food and enables NGOs to request and track food pickup.

- **Parent Monitoring:**

  Provides parents with read-only access to student details, room information, and complaints.

4.6 Performance Requirements

- The system should respond to user interactions within a few seconds to ensure smooth usability.

- Room occupancy and complaint status updates should reflect promptly across dashboards.
- The platform should support multiple concurrent users without noticeable performance issues.
- Key actions such as complaint updates and food pickup requests should be processed without delay.

4.7 Standard Compliance

- The system follows standard data protection and privacy practices to ensure secure handling of user information.
- The user interface is designed to be accessible and usable across different devices and screen sizes.

4.8 Software System Attributes

- **Scalability:**
  The system supports increasing users and multiple hostels without performance degradation.
- **Reliability:**
  The platform is designed to provide consistent and stable operation during regular use.
- **Usability:**
  A clean and intuitive interface ensures ease of use for all user roles.
- **Security:**
  Role-based access control ensures secure authentication and data protection.

4.9 Software and Hardware Requirements

Software Requirements:

- Operating System: Windows, macOS, or Linux with a modern web browser
- Framework: React
- Styling: Tailwind CSS
- Development Tools: Visual Studio Code

Hardware Requirements:

- Client Devices: Desktop, laptop, tablet, or smartphone with internet access
- Server: Backend server or cloud infrastructure (for future integration)

5 DESIGN PHASE

5.1 ER-Diagram:

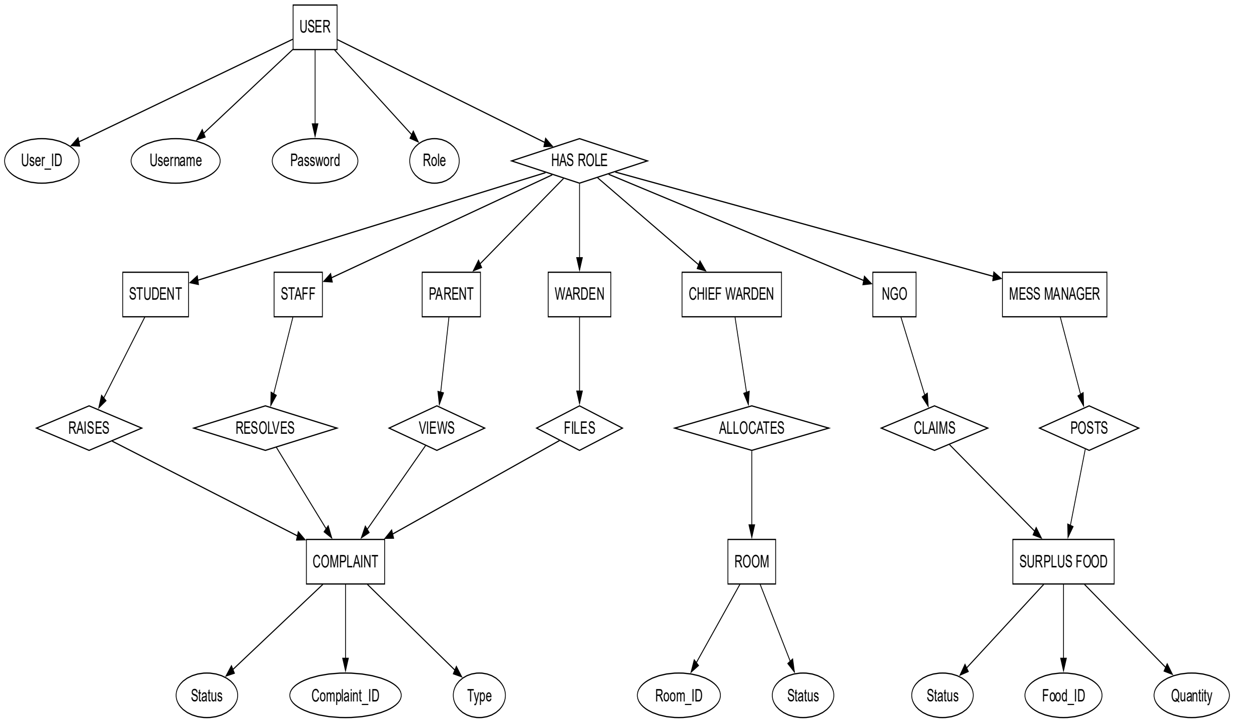


Fig 5.1 ER - Design

ER Diagram Description

The ER diagram represents the data structure and relationships within the *Hostel Management and Food Surplus Distribution Platform*.

The **Login/User** entity serves as the central authentication entity and stores common login credentials such as username, password, and role. All system users including Chief Warden, Warden, Staff, Student, Parent, Mess Manager, and NGO are authenticated through this entity.

The **Chief Warden** entity manages overall hostel administration and oversees wardens and complaints. The **Warden** entity is responsible for maintaining rooms and monitoring students and complaints within the assigned hostel. Each warden maintains multiple **Room** entities, where each room follows a fixed 3-sharing rule and is allocated to students.

The **Student** entity represents hostel residents. Each student is allocated to a room and can raise complaints related to hostel facilities. Students are linked to the **Parent** entity, allowing parents to log in using the student's credentials and monitor room details and complaints in a read-only manner.

The **HostelComplaint** and **StudentComplaint** entities manage complaints raised by students. These complaints can be viewed and resolved by **Staff** members and monitored by wardens and the chief warden. Staff members are responsible for resolving assigned complaints and updating their status.

The **Mess Manager** entity manages food-related operations and posts details of surplus food through the **SurplusFood** entity. Registered **NGO** entities can view available surplus food and request pickup, enabling structured food redistribution and reducing wastage.

5.2 Architectural Design:

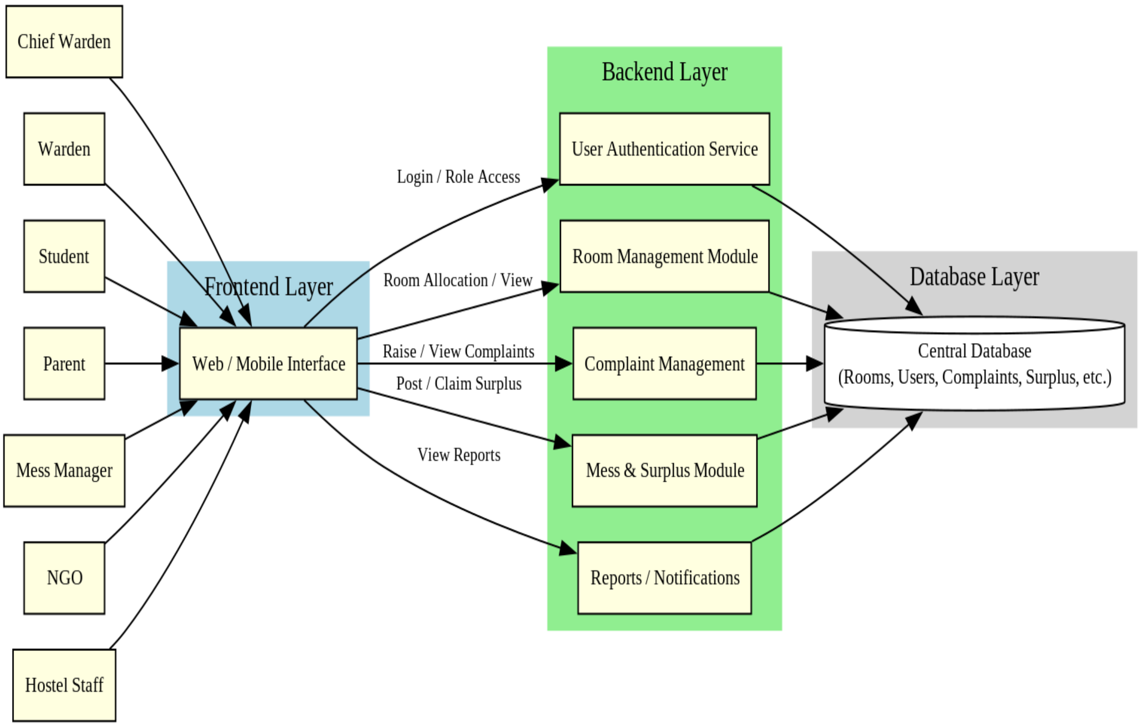


Fig 5.2 Architectural Design

System Architecture Description

The architecture of the *Hostel Management and Food Surplus Distribution Platform* follows a layered approach to ensure modularity, security, and scalability.

- **Presentation Layer (User Interface)**

This layer provides the web/mobile interface through which users such as Chief Warden, Warden, Student, Parent, Staff, Mess Manager, and NGO interact with the system. It handles user input, displays dashboards, room grids, complaints, and food surplus information.

- **Application Layer**

This layer contains the core business logic of the system. It processes user requests such as room allocation, complaint handling, staff management, and surplus food requests. It ensures that actions are executed based on role-based access rules.

- **Data Management Layer**

This layer manages all data storage and retrieval operations. It maintains centralized records related to users, rooms, complaints, students, staff, and surplus food while ensuring data consistency and integrity.

- **Security Layer**

The security layer enforces authentication and authorization mechanisms. It ensures that users can access only the features permitted for their role and protects sensitive data from unauthorized access.

- **Integration and External Services Layer**

This layer supports integration with external services if required, such as notification services or future third-party system integrations.

- **Update Management Layer**

This layer handles system updates and enhancements. It ensures smooth deployment of bug fixes, feature updates, and system improvements without affecting existing functionality.

5.3 Data flow Diagrams :

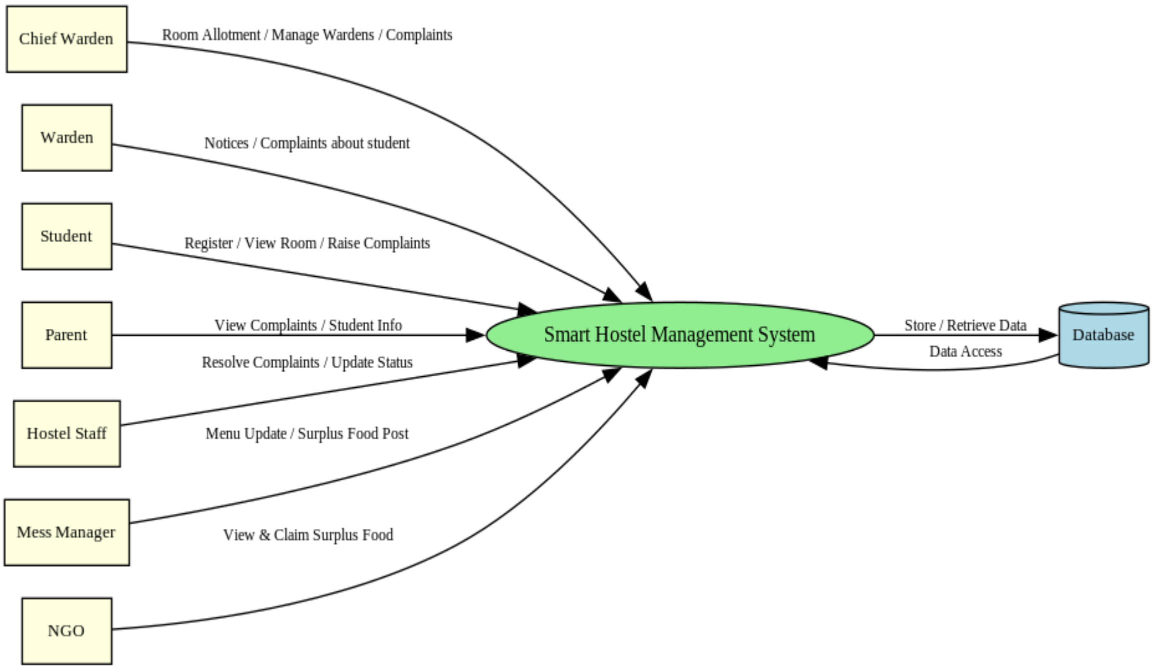

Chief Warden
Room Allotment / Manage Wardens / Complaints
Warden
Notices / Complaints about student
Student
Register / View Room / Raise Complaints
Parent
View Complaints / Student Info
Resolve Complaints / Update Status
Hostel Staff
Menu Update / Surplus Food Post
Mess Manager
View & Claim Surplus Food
NGO
Smart Hostel Management System
Store / Retrieve Data
Data Access
Database

Fig 5.3.1 Level 0 DFD

Data Flow Diagram (DFD) – Level 0 Description

The DFD Level-0 represents the high-level data flow of the *Hostel Management and Food Surplus Distribution Platform*. It shows how external entities interact with the central system and how data is stored and retrieved from the database.

External Entities

- **Chief Warden:**

  Manages room allotment, wardens, and monitors complaints.

- **Warden:**

  Views notices, monitors student complaints, and oversees hostel operations.

- **Student:**

  Views room details and raises complaints related to hostel facilities.

- **Parent:**

  Views student information and complaint status in a read-only manner.

- **Hostel Staff:**

Resolves complaints and updates their status in the system.

- **Mess Manager:**

  Posts surplus food details generated from the hostel mess.

- **NGO:**
  Views available surplus food and requests food pickup.

Process

- **Smart Hostel Management System:**

  This is the core process that handles hostel operations such as room management, complaint handling, staff coordination, and food surplus distribution based on role-based access.

Data Flows

• Room allotment, complaint details, surplus food information flow from users to the system
• Complaint status, room information, and surplus availability flow from the system to users
• Data is stored in and retrieved from the **Database** to maintain records related to users, rooms, complaints, and food surplus

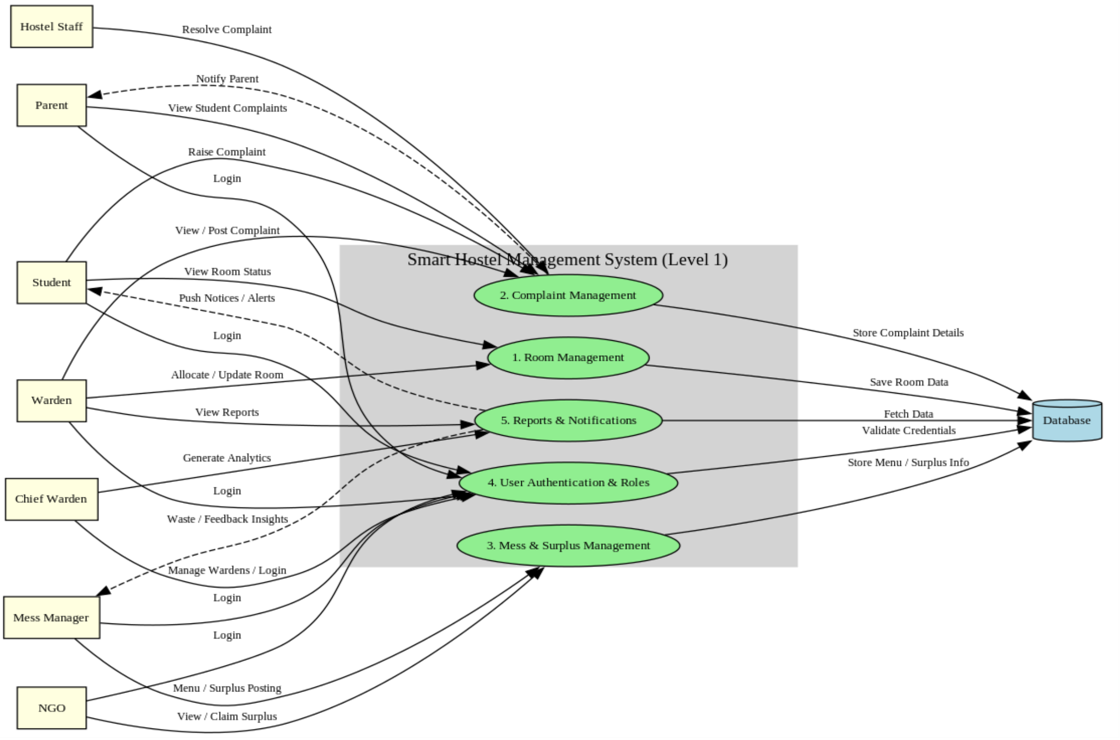


Fig 5.3.2 Level 1 DFD

Data Flow Diagram (DFD) – Level 1 Description

The DFD Level-1 provides a detailed view of the internal processes of the *Hostel Management and Food Surplus Distribution Platform*. It expands the main system into functional modules and shows how different users interact with each process.

External Entities

- **Chief Warden:**

  Logs in, manages wardens, views reports, and monitors overall hostel operations.

- **Warden:**

  Allocates and updates rooms, views complaints, and monitors student activities.

- **Student:**

  Logs in, views room status, raises complaints, and receives notices or alerts.

- **Parent:**

  Views student information and complaint status in read-only mode.

- **Hostel Staff:**

  Views assigned complaints and updates their resolution status.

- **Mess Manager:**

  Logs in and posts surplus food details generated from the mess.

- **NGO:**

  Views available surplus food and requests or claims food pickup.

Processes

- **User Authentication and Roles:**

  Handles login and role-based access for all users.

- **Room Management:**

  Manages room allocation, updates room status, and retrieves room details.

- **Complaint Management:**

  Processes complaints raised by students and allows staff and wardens to update and monitor complaint status.

- **Mess & Surplus Management:**

  Handles surplus food posting by mess managers and food requests by NGOs.

- **Reports & Notifications:**

  Generates reports, alerts, and notifications for authorized users.

Data Stores

- **Database:**
  Stores user details, room data, complaints, staff records, and surplus food information. All processes retrieve and store data through this centralized database.

Data Flows

- Login credentials flow from users to the authentication process
- Room, complaint, and surplus data flow between users and respective modules
- Processed data is stored in and retrieved from the database for consistency and tracking

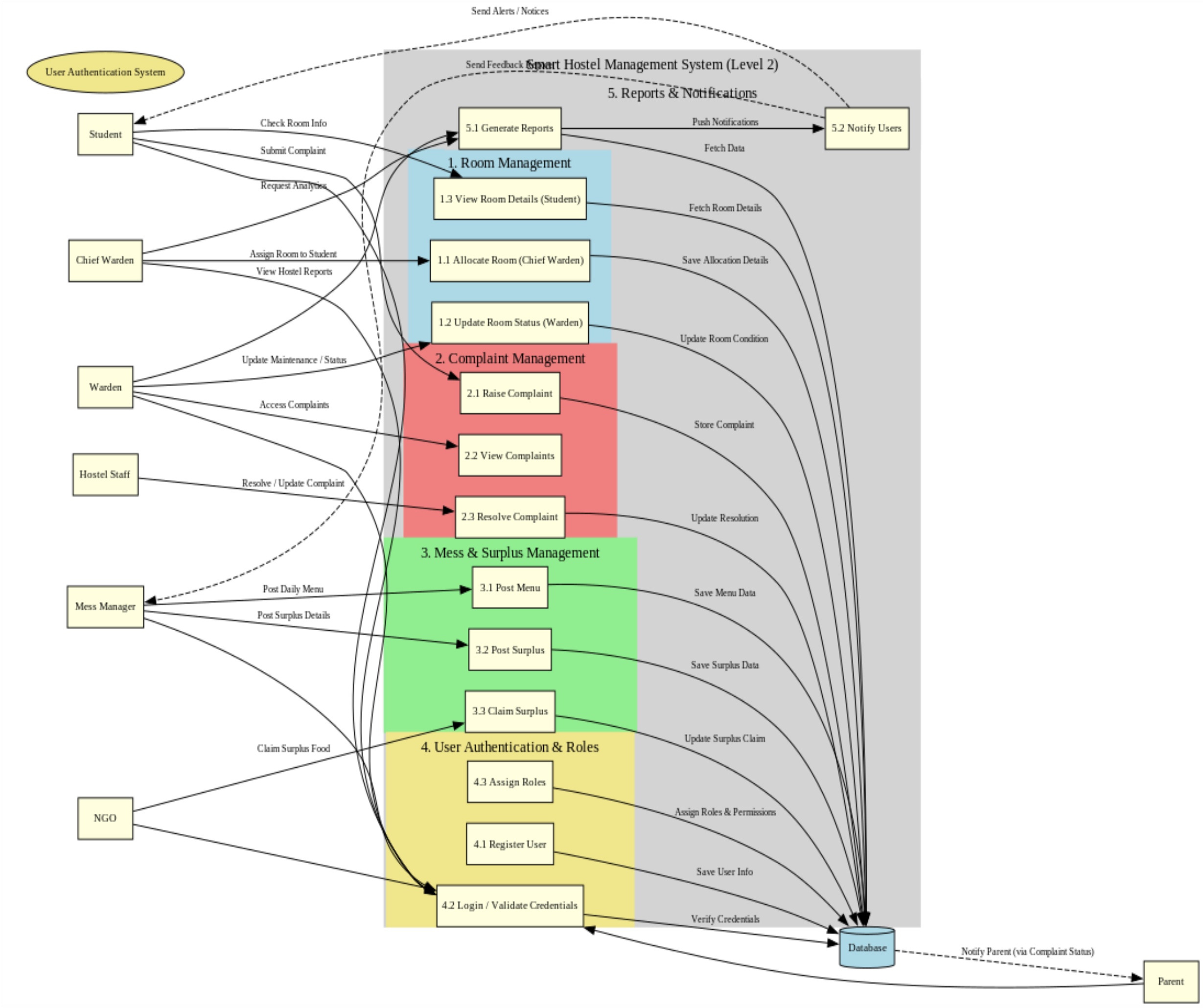


Fig 5.3.3 Level 2 DFD

Data Flow Diagram (DFD) – Level 2 Description

The DFD Level-2 provides a detailed breakdown of the internal processes of the *Hostel Management and Food Surplus Distribution Platform*. It further decomposes major system modules to show how specific operations are performed and how data flows between users, processes, and the database.

External Entities

- **Chief Warden:**

  Allocates rooms, manages wardens, and views reports.

- **Warden:**

Updates room status, accesses complaints, and monitors maintenance activities.

- **Student:**

  Views room details, raises complaints, and receives notifications.

- **Hostel Staff:**

  Resolves complaints and updates resolution status.

- **Mess Manager:**

  Posts daily menu and surplus food details.

- **NGO:**

  Views and claims available surplus food.

- **Parent:**

  Receives notifications and views complaint status in read-only mode.

Processes

1. **Room Management**

- Allocate Room (Chief Warden)
- Update Room Status (Warden)
- View Room Details (Student)

2. **Complaint Management**

- Raise Complaint (Student)
- View Complaints (Warden/Staff)
- Resolve Complaint (Staff)

3. **Mess & Surplus Management**

- Post Menu (Mess Manager)
- Post Surplus Food (Mess Manager)
- Claim Surplus Food (NGO)

4. **User Authentication & Roles**

- Register User
- Login and Validate Credentials
- Assign Roles and Permissions

5. **Reports & Notifications**

- Generate Reports
- Notify Users and Parents

Data Store

- **Central Database:**

Stores user information, room details, complaint records, menu data, surplus food details, and notification logs.

Data Flows

- User actions are sent to respective processes
- Processed data is stored in the database
- Required information is retrieved from the database for display and notifications

5.4 Class Diagram:

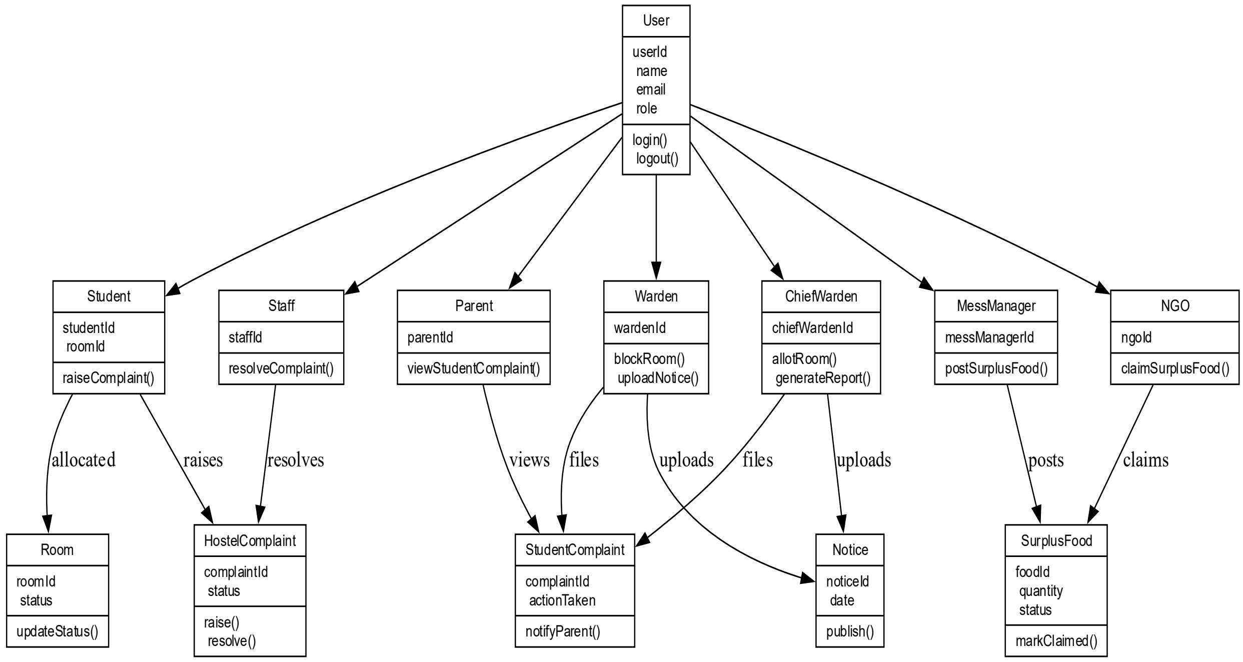


Fig 5.4 Class Diagram

Class Diagram Description

The class diagram represents the static structure of the *Hostel Management and Food Surplus Distribution Platform* by defining the system classes, their attributes, methods, and relationships.

The **User** class is the base class that contains common attributes such as userId, name, email, phone, role, and password. All system roles inherit from this class.

The **ChiefWarden** class extends User and has responsibilities such as creating wardens, students, and mess managers, managing hostels, viewing reports, and resetting academic data.

The **Warden** class manages hostel-specific operations including staff management, room monitoring, and complaint supervision.

The **Student** class stores student-specific details such as USN, roomId, and hostelId. Students can raise complaints and view room and notice information. The **Parent** class is logically linked to the Student class and provides read-only access to student details and complaints.

The **Staff** class handles maintenance-related tasks and updates complaint status after resolution.

The **Room** class represents hostel rooms with attributes like roomNumber, capacity, and occupancy. Each room is associated with multiple students (3-sharing).

The **Complaint** class manages student complaints with attributes such as complaintId, description, status, and resolutionImage. Complaints are raised by students and handled by staff and wardens.

The **MessManager** class manages food-related operations and is associated with the **SurplusFood** class, which stores details such as foodName, quantity, pickupTime, and status.

The **NGO** class interacts with surplus food data by viewing and claiming available food.

Overall, the class diagram clearly shows inheritance, associations, and responsibilities of each class, ensuring modular design, role separation, and maintainability of the system.

5.5 Use case Diagram:

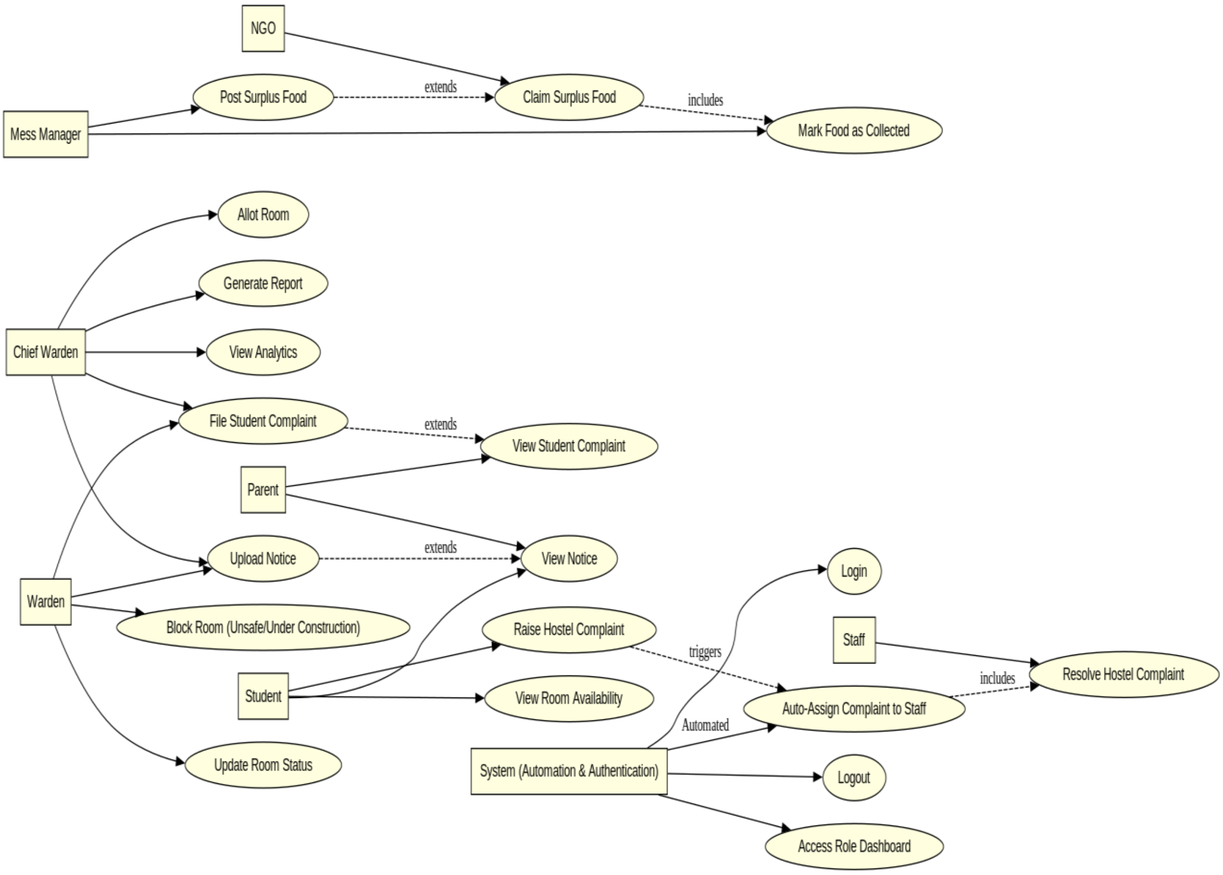


Fig 5.5 Use case Diagram

Use Case Diagram Description

The use case diagram represents the interactions between different users and the *Hostel Management and Food Surplus Distribution Platform*. It highlights the major functionalities performed by each actor based on their roles.

Actors

- **Chief Warden:**

  Manages overall hostel operations such as room allotment, viewing analytics, generating reports, uploading notices, and monitoring student complaints.

- **Warden:**

  Updates room status, blocks unsafe or under-construction rooms, uploads notices, and monitors hostel-related complaints.

- **Student:**

  Views room availability, raises hostel complaints, and accesses notices and room-related information.

- **Parent:**

  Views student complaints and notices in a read-only manner.

- **Hostel Staff:**

  Logs into the system and resolves hostel complaints assigned to them.

- **Mess Manager:**

  Posts surplus food details generated from the hostel mess.

- **NGO:**

  Views available surplus food, claims surplus food, and marks food as collected.

- **System (Automation & Authentication):**

  Handles user authentication, role-based dashboard access, auto-assignment of complaints to staff, and system logout.

Major Use Cases

- Room allotment and room status management
- Raising, viewing, and resolving hostel complaints
- Automatic assignment of complaints to staff
- Posting, claiming, and collecting surplus food
- Uploading and viewing notices
- User login, logout, and role-based dashboard access

5.6 State Diagram:

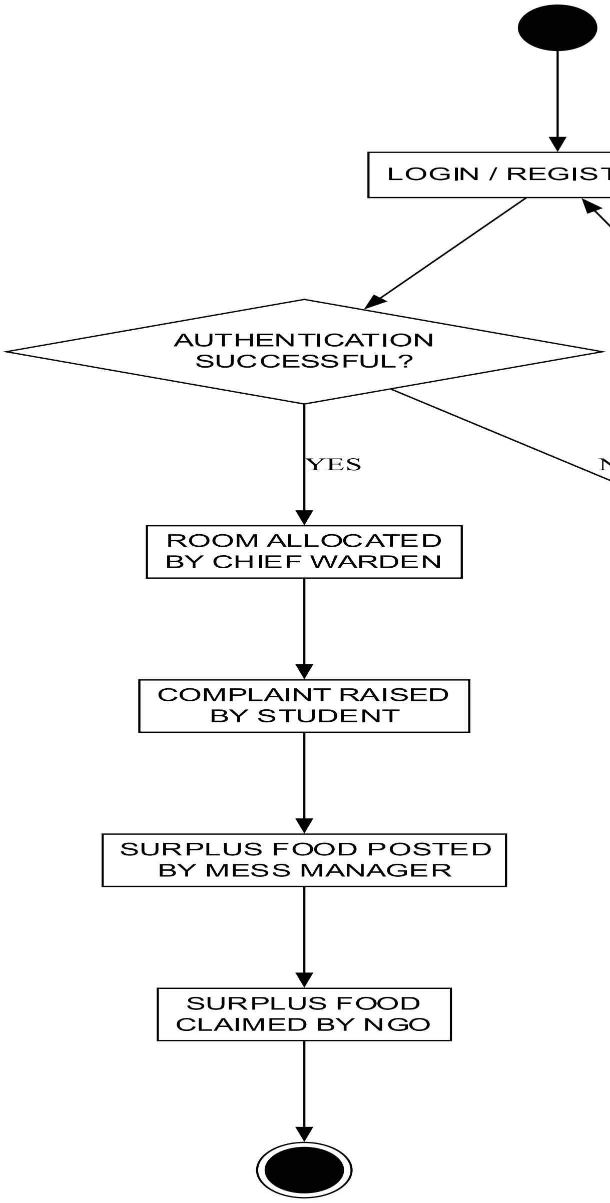

LOGIN / REGISTRATION
AUTHENTICATION SUCCESSFUL?
YES
NO
ROOM ALLOCATED BY CHIEF WARDEN
UNSUCCESSFUL
COMPLAINT RAISED BY STUDENT
SURPLUS FOOD POSTED BY MESS MANAGER
SURPLUS FOOD CLAIMED BY NGO

Fig 5.6 State Diagram

State Diagram Description

The state diagram represents the flow of states involved in the *Hostel Management and Food Surplus Distribution Platform* from user interaction to task completion.

- **Start State:**

  The process begins when a user accesses the system.

- **Login / Authentication:**

  The user logs in using role-based credentials. If authentication fails, the system returns to the login state.

- **Dashboard Access:**

  Upon successful login, the user is redirected to the role-specific dashboard.

- **Action Initiation:**

  The user performs actions such as room allocation, raising a complaint, posting surplus food, or viewing details based on their role.

- **Processing State:**

  The system processes the request by validating data, assigning tasks (such as auto-assigning complaints), or updating records.

- **Completion State:**

  Once the action is successfully processed, the system updates the status and displays confirmation to the user.

- **End State:**

  The process ends when the user logs out or completes the operation.

If any step fails or requires correction, the system transitions back to the previous appropriate state to ensure continuity and correctness.

5.7 Sequence Diagram

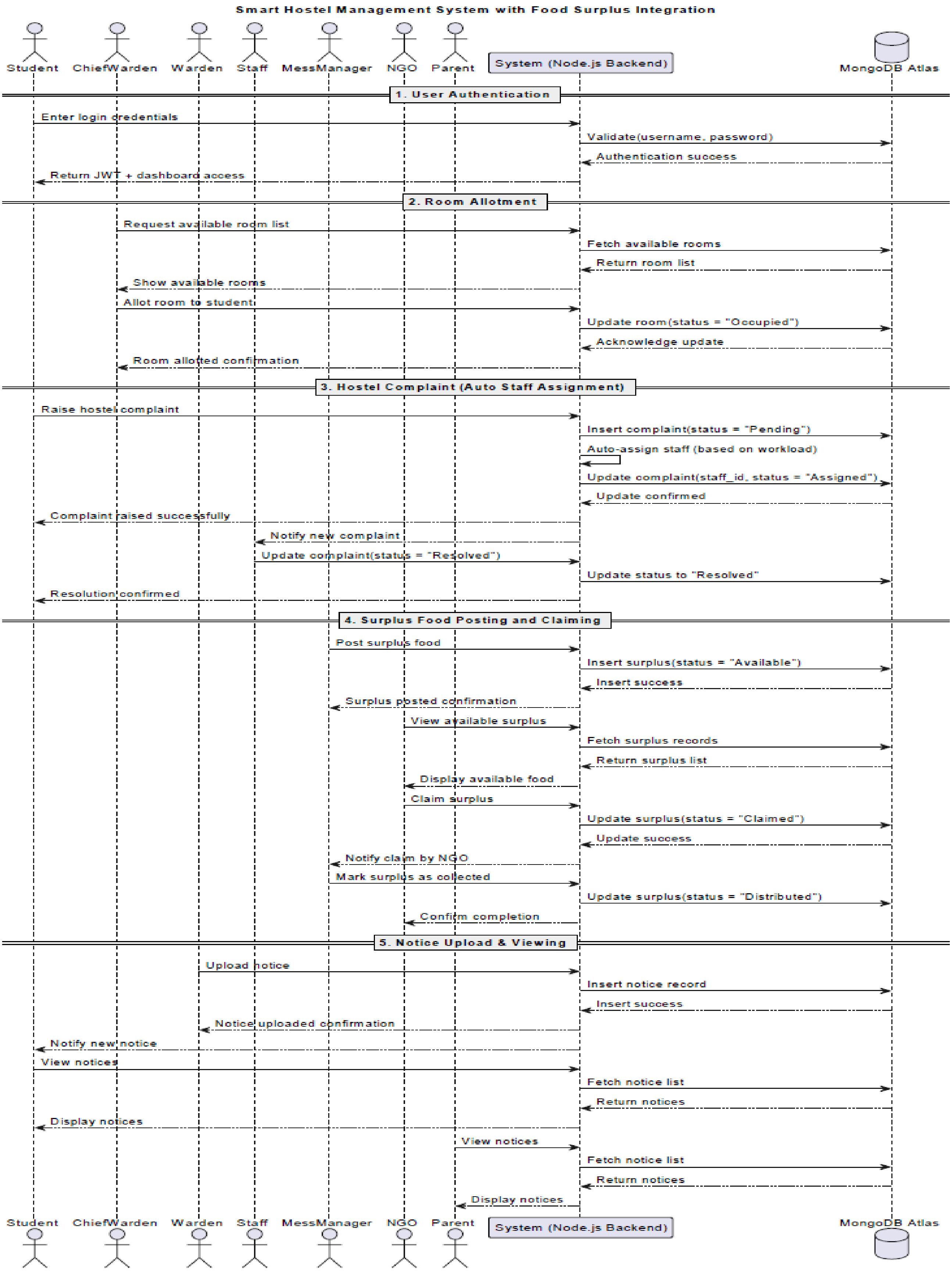


Fig 5.7 Sequence Diagram

Sequence Diagram Description

The sequence diagram illustrates the interaction flow between users, the system, and the database in the *Hostel Management and Food Surplus Distribution Platform*.

- **User Login:**

  The user (Student, Chief Warden, Warden, Staff, Mess Manager, NGO, or Parent) logs into the system. The system validates credentials and grants role-based access.

- **Room Allotment Process:**

  The Chief Warden requests available rooms. The system fetches room details from the database and allows room allocation to students. The updated room status is stored in the database.

- **Complaint Management:**

  A student raises a hostel complaint. The system records the complaint, auto-assigns it to staff, and updates the complaint status. Staff resolve the complaint, and the system updates the resolution status.

- **Food Surplus Posting and Claiming:**

  The Mess Manager posts surplus food details. The system stores the information and displays it to NGOs. NGOs claim surplus food, and the system updates the status to collected or distributed.

- **Notice Upload and Viewing:**

  Wardens or the Chief Warden upload notices. The system stores notices and allows students and parents to view them.

- **Notifications and Updates:**

  The system sends confirmations and notifications for actions such as room allotment, complaint resolution, surplus food collection, and notice updates.

- **Logout:**

  After completing actions, the user logs out, ending the interaction.

6 IMPLEMENTATION PHASE

6.1 Authentication & User Management

The Authentication and User Management module handles secure access control and role-based login for all users in the system.

Functionalities:

- Multi-Role Authentication
  - ➢ Supports login for Student, Parent, Warden, Chief Warden, Staff, NGO, and Mess Manager.
  - ➢ Uses JWT (JSON Web Tokens) for session management.
- User Registration (Admin Controlled)
  - ➢ Chief Warden registers Students, Wardens, and Mess Managers.
  - ➢ System auto-generates secure passwords and assigns roles.
- Parent Login Handling
  - ➢ Parents log in using their child's USN and password.
  - ➢ Role-based access ensures parents can only view student-related data.
- Authorization Middleware
  - ➢ Protected routes ensure users can only access dashboards permitted by their role.
  - ➢ Unauthorized access redirects to the login page.

Process Flow:

- User submits credentials.
- Backend validates credentials using encrypted passwords.
- JWT token is generated and stored in the client.
- Role-based redirection to respective dashboards.

6.2 Student & Room Allocation Management

This module manages hostel rooms, occupancy, and student allocations.

Functionalities:

- Room Creation & Status Management
    - Rooms are categorized as available, full, or maintenance.
    - Capacity and occupancy are tracked dynamically.
- Student Allocation
    - Students are assigned to rooms during registration.
    - Room occupancy updates automatically.
- Room Grid Visualization
    - Chief Warden views a grid layout of rooms with color-coded status.
    - Clicking a room displays student details (name & USN).
- Maintenance Handling
    - Empty rooms can be marked as maintenance.
    - Maintenance rooms cannot accept new students.

Process Flow:

- Rooms fetched via backend API.
- Occupancy calculated based on assigned students.
- UI updates room status in real time.
- Maintenance status updated through secured API calls.

6.3 Complaint Management System

This module enables structured complaint handling across different roles.

Functionalities:

- Student Complaints
    - Students raise infrastructure-related complaints with optional images.
    - Categories include electrician, plumber, carpenter, etc.

- Staff Handling
  - ➢ Staff update complaint status (in-progress, resolved, not-resolvable).
  - ➢ Resolution images can be uploaded if required.
- Warden & Chief Warden Oversight
  - ➢ Wardens can view closed complaints.
  - ➢ Chief Warden monitors all complaints hostel-wise.
- Complaint Lifecycle Tracking
  - ➢ Status transitions tracked with timestamps.
  - ➢ History preserved for auditing.

Process Flow:

- Complaint submitted by student.
- Assigned staff updates status.
- Resolution stored in database.
- Complaint removed from active list when closed.

6.4 Student Behavioral Complaint Module

This module handles disciplinary and behavioral complaints raised by wardens or chief wardens.

Functionalities:

- Behavioral Complaint Creation
  - ➢ Wardens raise complaints related to student behavior or violations.
  - ➢ Separate from infrastructure complaints.
- Parent Visibility
  - ➢ Parents can view only behavioral complaints related to their child.
  - ➢ Ensures transparency without exposing internal maintenance issues.
- Role-Based Access Control
  - ➢ Students cannot view behavioral complaints.
  - ➢ Only parents and wardens can access this module.

Process Flow:

- Warden submits complaint using student USN.
- Complaint stored separately in student-complaints collection.
- Parents fetch data using secured parent dashboard API.

6.5 Notice & Announcement Management

This module manages hostel-wide announcements.

Functionalities:

- Notice Creation
    - Wardens and Chief Warden can upload notices.
    - Supports optional image attachments.
- Notice Distribution
    - Notices visible to Students, Parents, and Wardens.
    - Sorted by most recent announcements.
- Role Tracking
    - Each notice records the role of the creator (warden/chief warden).

Process Flow:

- Authorized user uploads notice.
- Image stored in Cloudinary.
- Notice metadata saved in database.
- Notices fetched on dashboards.

6.6 Mess Surplus & NGO Coordination

This module manages food surplus sharing with NGOs.

Functionalities:

- Surplus Posting

  - Mess Manager posts surplus food with quantity and deadline.
  - Images supported.
- NGO Claiming
  - NGOs view available surplus.
  - Can claim surplus before expiry.
- Auto Expiry Handling
  - Surplus automatically marked expired after deadline.
- History Tracking
  - Claimed surplus visible to Mess Manager and NGOs.

Process Flow:

- Mess Manager posts surplus.
- NGOs fetch available surplus.
- NGO claims surplus.
- Status updated and removed from availability list.

6.7 Security & Data Integrity

This module ensures the system is secure and reliable.

Functionalities:

- JWT-Based Security
  - Tokens validated on every protected request.
- Role Authorization
  - Middleware ensures correct role access.
- Input Validation
  - Backend validates all critical inputs.
- Cloud Storage Security
  - Images stored securely using Cloudinary.

Process Flow:

- Token verified by middleware.
- Role checked against allowed roles.
- Request processed or rejected accordingly.

6.8 Deployment & Hosting

This module covers live deployment of the system.

Functionalities:

- Backend Hosting
  - ➢ Node.js backend deployed on Render.
  - ➢ MongoDB Atlas used as database.
- Frontend Hosting
  - ➢ Frontend deployed on Vercel.
  - ➢ Environment variables configured for production API.
- CI/CD Integration
  - ➢ Automatic redeployment on GitHub commits.

7 TESTING PHASE

7.1 System Setup

Hardware Setup

- **Devices:** Desktop, laptop, tablet, and mobile devices with different screen sizes for responsive testing.
- **Environment:** Physical devices and browser-based testing environments for validation.

Software Requirements

- **Operating System:** Windows, macOS, or Linux.
- **Frontend:** React with Tailwind CSS.
- **Database:** Backend or cloud database (for testing and future integration).
- **Testing Tools:**
    - Browser developer tools
    - React testing utilities
    - Manual testing tools
- **Network Conditions:**
    - Test under normal and low internet connectivity.
- **Version Control:**
    - Git for tracking changes and managing versions.
- **Integration Setup:**
    - Frontend module integration
    - Authentication and role-based access verification

7.2 Types of Tests Carried Out

- **Unit Testing**
    - **Purpose:** Test individual components and functions independently.

- ➢ **Examples:**
  - Testing login form validation
  - Testing room grid rendering and complaint submission logic

- **Integration Testing**
  - ➢ **Purpose:** Verify interaction between different modules.
  - ➢ **Examples:**
    - Login flow connected to dashboard
    - Complaint submission linked with staff resolution

- **Functional Testing**
  - ➢ **Purpose:** Validate complete system functionality.
  - ➢ **Examples:**
    - Room allocation process
    - Complaint lifecycle from student to staff resolution
    - Food surplus posting and claiming

- **UI Testing**
  - ➢ **Purpose:** Ensure proper appearance and responsiveness of the UI.
  - ➢ **Examples:**
    - Dashboard layout across devices
    - Role-based navigation and grid display

6 Performance Testing

- ➢ **Purpose:** Check system responsiveness and stability.
- ➢ **Examples:**
  - Page load time
  - Handling multiple users simultaneously

7 Security Testing

- ➢ **Purpose:** Ensure secure access and data protection.

- **Examples:**
    - Role-based access control verification
    - Authentication and authorization checks

7.3 Test Cases

1. Login Functionality

- **Test Case:** Validate user login with correct role-based credentials.
- **Input:** Valid identifier (USN / Email / Phone) and password.
- **Expected Output:** Successful login and redirection to role-specific dashboard.
- **Steps:**
    - Select user role.
    - Enter valid credentials.
    - Click **Login**.
    - Verify dashboard access.
- **Success Criteria:** User is logged in without errors and correct dashboard is displayed.

2. Room Allocation

- **Test Case:** Validate room allocation by Chief Warden.
- **Input:** Valid student details and room selection.
- **Expected Output:** Student successfully allocated to selected room.
- **Steps:**
    - Login as Chief Warden.
    - Select room from 6×6 grid.
    - Enter student details.
    - Save allocation.
- **Success Criteria:** Room occupancy updates correctly.

3. Complaint Submission

- **Test Case:** Validate student complaint submission.
- **Input:** Complaint description.
- **Expected Output:** Complaint recorded and visible to staff and warden.
- **Steps:**
  - ➢ Login as Student.
  - ➢ Navigate to Complaints section.
  - ➢ Submit complaint.

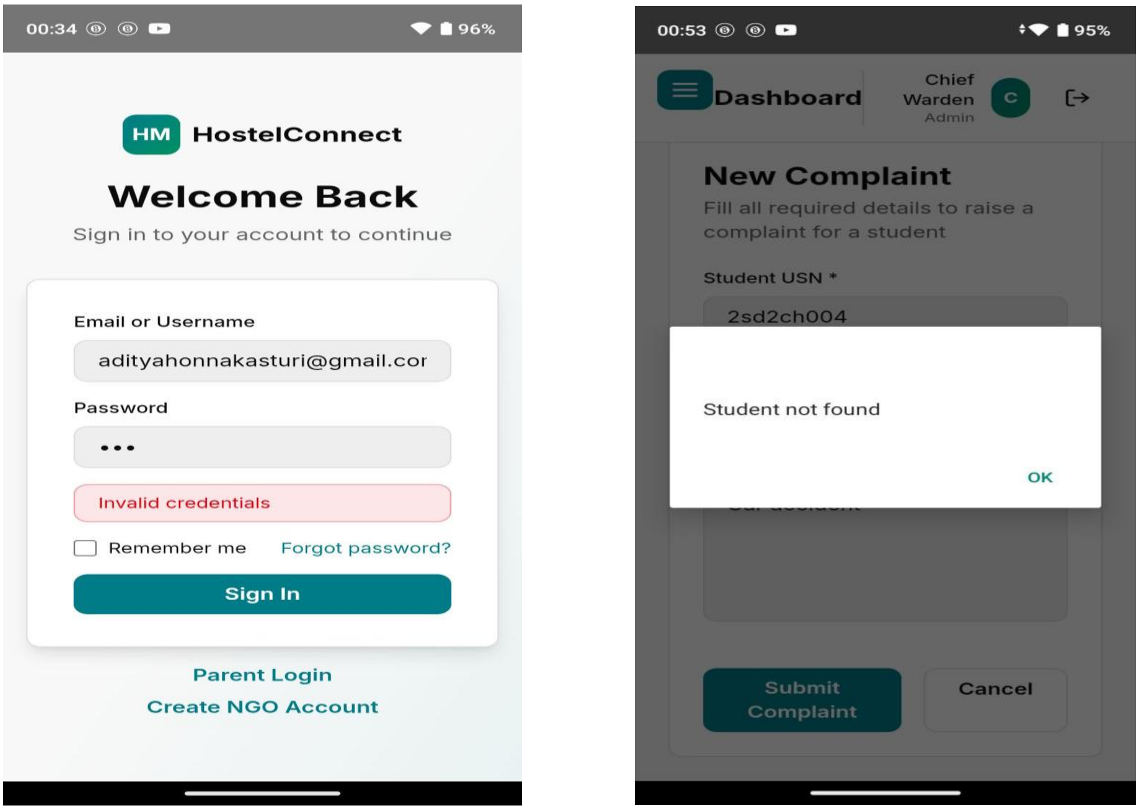
4. Complaint Resolution

- **Test Case:** Validate complaint resolution by staff.
- **Input:** Complaint status update and resolution details.
- **Expected Output:** Complaint marked as resolved.
- **Steps:**
  - ➢ Login as Staff.
  - ➢ View assigned complaint.
  - ➢ Update status.
- **Success Criteria:** Complaint status updated successfully.

5. Food Surplus Posting

- **Test Case:** Validate surplus food posting by Mess Manager.
- **Input:** Food name, quantity, pickup time.
- **Expected Output:** Surplus food visible to NGOs.
- **Steps:**
  - ➢ Login as Mess Manager.
  - ➢ Add surplus food details.
  - ➢ Submit form.
- **Success Criteria:** Food entry saved and displayed correctly.

Negative Test Cases :



00:34 96%

HM HostelConnect

Welcome Back

Sign in to your account to continue

Email or Username

adityahonnakasturi@gmail.cor

Password

•••

Invalid credentials

□ Remember me Forgot password?

Sign In

Parent Login

Create NGO Account

00:53 95%

Dashboard Chief Warden Admin C

New Complaint

Fill all required details to raise a complaint for a student

Student USN *

2sd2ch004

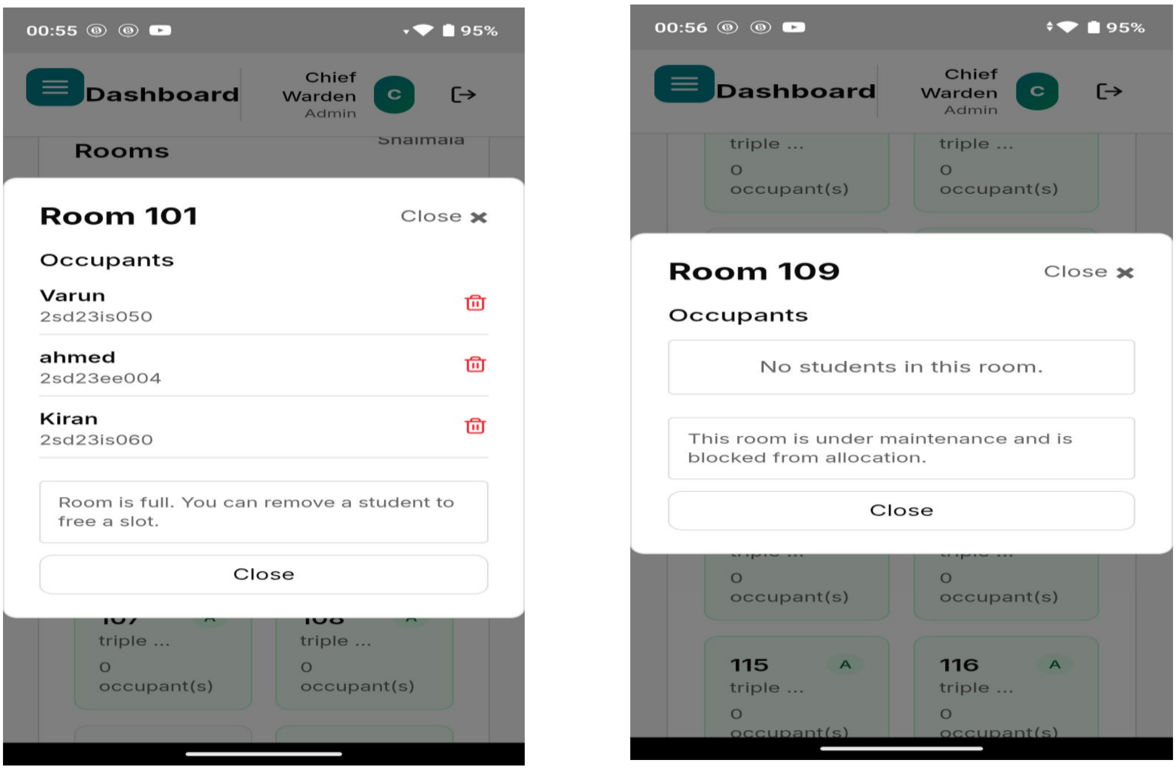
Student not found

OK

Submit Complaint

Cancel

Fig 7.1 Authentication Testing



00:55 95%

Dashboard Chief Warden Admin C

Rooms

Room 101 Close ✖

Occupants

Varun
2sd23is050

ahmed
2sd23ee004

Kiran
2sd23is060

Room is full. You can remove a student to free a slot.

Close

00:56 95%

Dashboard Chief Warden Admin C

Room 109 Close ✖

Occupants

No students in this room.

This room is under maintenance and is blocked from allocation.

Close

115 A triple ... 0 occupant(s)

116 A triple ... 0 occupant(s)

Fig 7.2 Functionality Testing

8 RESULT AND DISCUSSION

Results

- **Enhanced User Accessibility:**

The role-based authentication system ensures secure and controlled access for all users such as chief warden, warden, students, staff, parents, mess managers, and NGOs. This improves usability and prevents unauthorized access.

- **Efficient Hostel Operations:**

Digital room allocation using a fixed 6×6 grid and real-time occupancy updates has streamlined hostel management and reduced manual errors.

- **Improved Complaint Management:**

The online complaint system enables faster reporting, tracking, and resolution of hostel issues, improving accountability and response time.

- **Effective Food Surplus Distribution:**

The surplus food module allows mess managers to post excess food and NGOs to claim it efficiently, reducing food wastage and supporting social responsibility.

- **Transparency for Parents:**

The parent dashboard provides read-only access to student room details and complaints, enhancing transparency and trust.

- **User-Friendly Interface:**

The modern, responsive UI ensures smooth navigation and consistent experience across devices.

Discussion

- **Impact on Institutional Management:**

The platform bridges gaps in traditional hostel administration by digitizing core processes such as room management, complaints, and staff coordination, leading to improved operational efficiency.

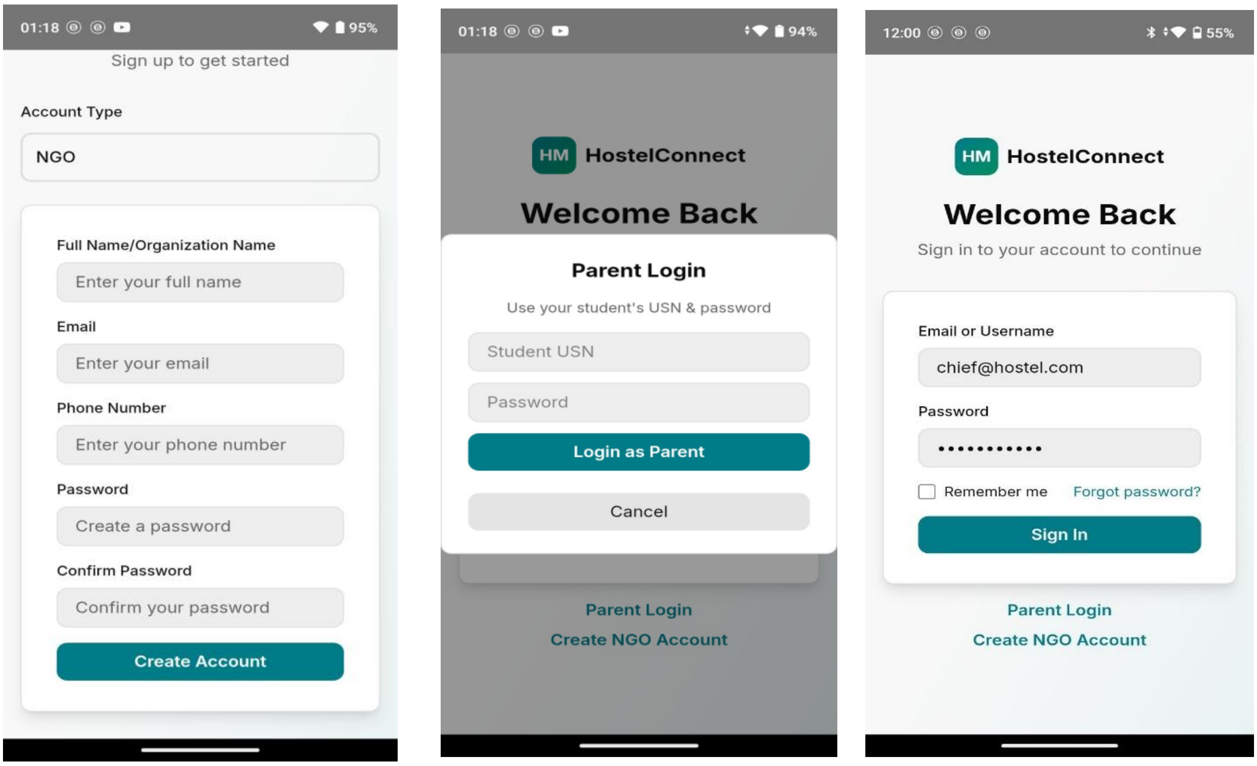
- **User Experience:**

The intuitive role-based dashboards and clean UI design make the system easy to use for both technical and non-technical users.

- **Challenges Addressed:**
  - **Transparency:** Real-time updates improve monitoring and accountability.
  - **Coordination:** Centralized data reduces communication delays.
  - **Food Wastage:** Structured surplus redistribution minimizes waste.
- **Opportunities for Improvement:**
  - Multilingual support for wider adoption
  - Advanced analytics and reporting
  - Mobile app extension for enhanced accessibility
- **Future Scope:**

The system can be expanded with backend integration, automated notifications, advanced reporting, and collaboration with more NGOs, making it scalable for multiple institutions.

Results :



01:18 95%

Sign up to get started

Account Type

NGO

Full Name/Organization Name

Enter your full name

Email

Enter your email

Phone Number

Enter your phone number

Password

Create a password

Confirm Password

Confirm your password

Create Account

01:18 94%

HM HostelConnect

Welcome Back

Parent Login

Use your student's USN & password

Student USN

Password

Login as Parent

Cancel

Parent Login

Create NGO Account

12:00 55%

HM HostelConnect

Welcome Back

Sign in to your account to continue

Email or Username

chief@hostel.com

Password

•••••••••••

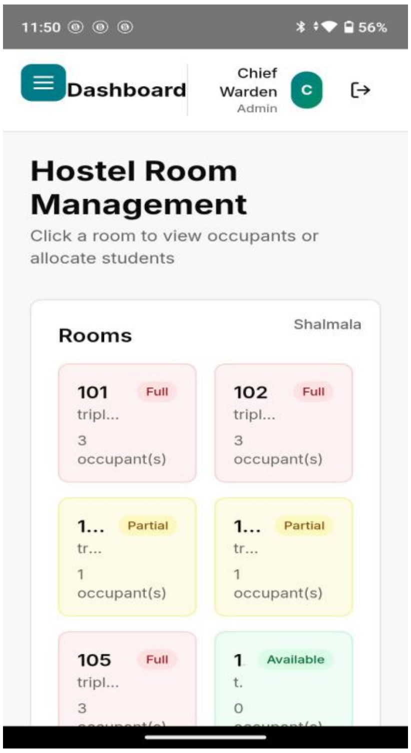
□ Remember me Forgot password?

Sign In

Parent Login

Create NGO Account

Fig 8.1 Splash Screen And Login Screens



11:50 56%

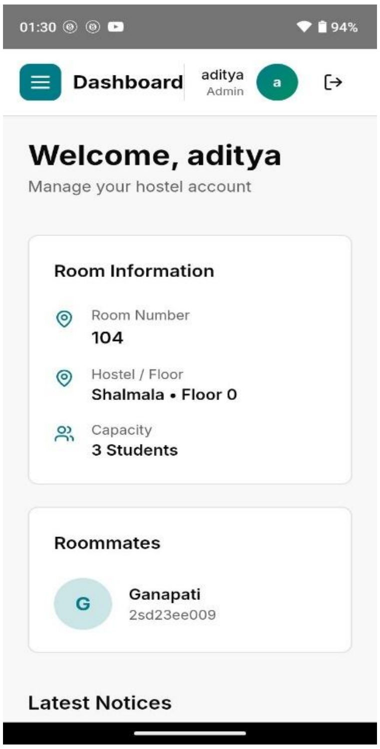
Dashboard Chief Warden Admin C

Hostel Room Management

Click a room to view occupants or allocate students

Rooms Shalmala

101 Full tripl... 3 occupant(s)	102 Full tripl... 3 occupant(s)
1... Partial tr... 1 occupant(s)	1... Partial tr... 1 occupant(s)
105 Full tripl... 3	1 Available t. 0



01:30 94%

Dashboard aditya Admin a

Welcome, aditya

Manage your hostel account

Room Information

Room Number
104

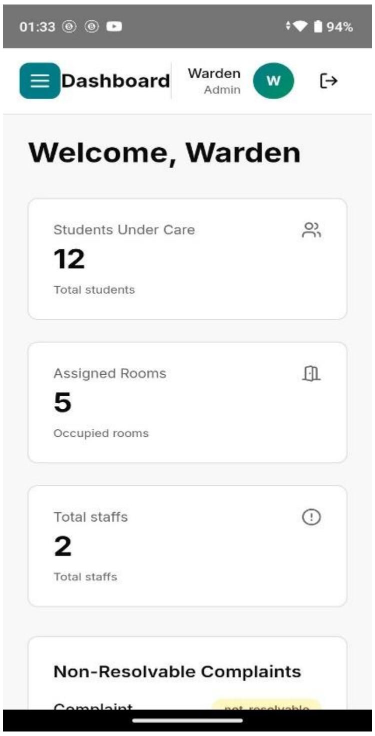
Hostel / Floor
Shalmala • Floor 0

Capacity
3 Students

Roommates

G Ganapati
2sd23ee009

Latest Notices



01:33 94%

Dashboard Warden Admin W

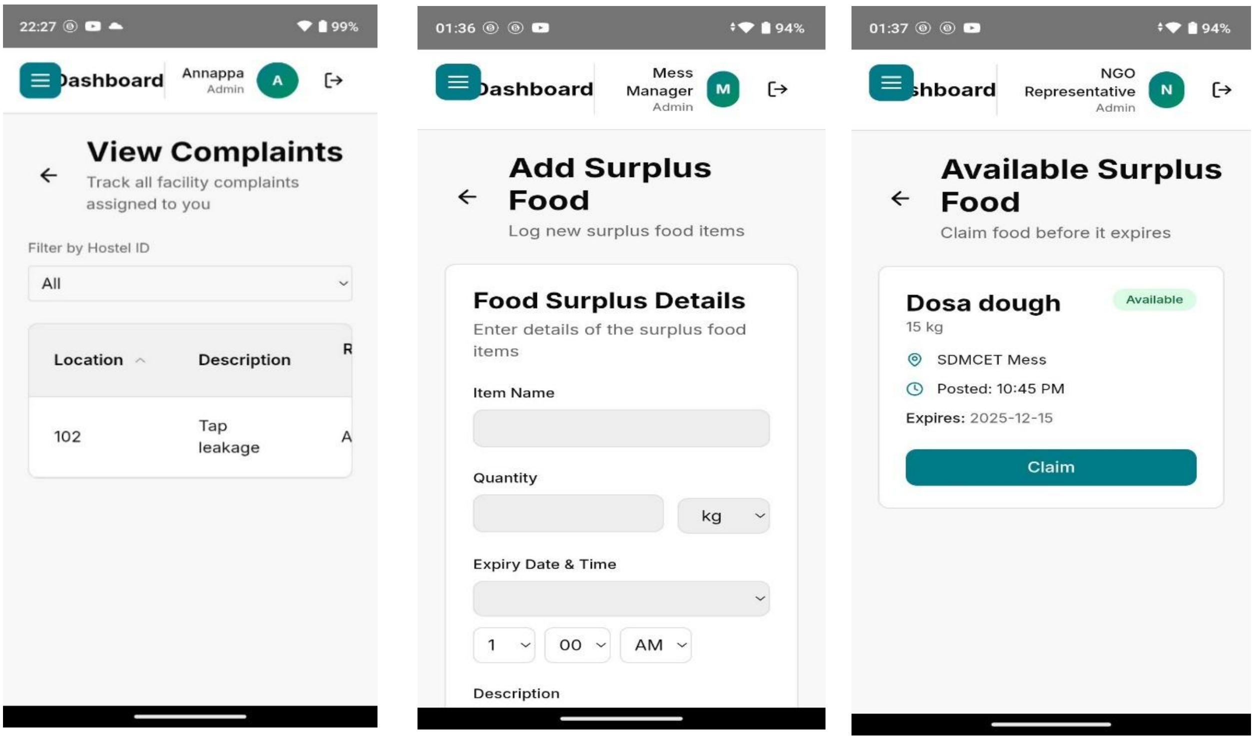
Welcome, Warden

Students Under Care
12
Total students

Assigned Rooms
5
Occupied rooms

Total staffs
2
Total staffs

Non-Resolvable Complaints



22:27 99%

Dashboard Annappa Admin A

View Complaints

Track all facility complaints assigned to you

Filter by Hostel ID

All

Location	Description	R
102	Tap leakage	A

01:36 94%

Dashboard Mess Manager Admin M

Add Surplus Food

Log new surplus food items

Food Surplus Details

Enter details of the surplus food items

Item Name

Quantity

kg

Expiry Date & Time

1 00 AM

Description

01:37 94%

shboard NGO Representative Admin N

Available Surplus Food

Claim food before it expires

Dosa dough Available

15 kg

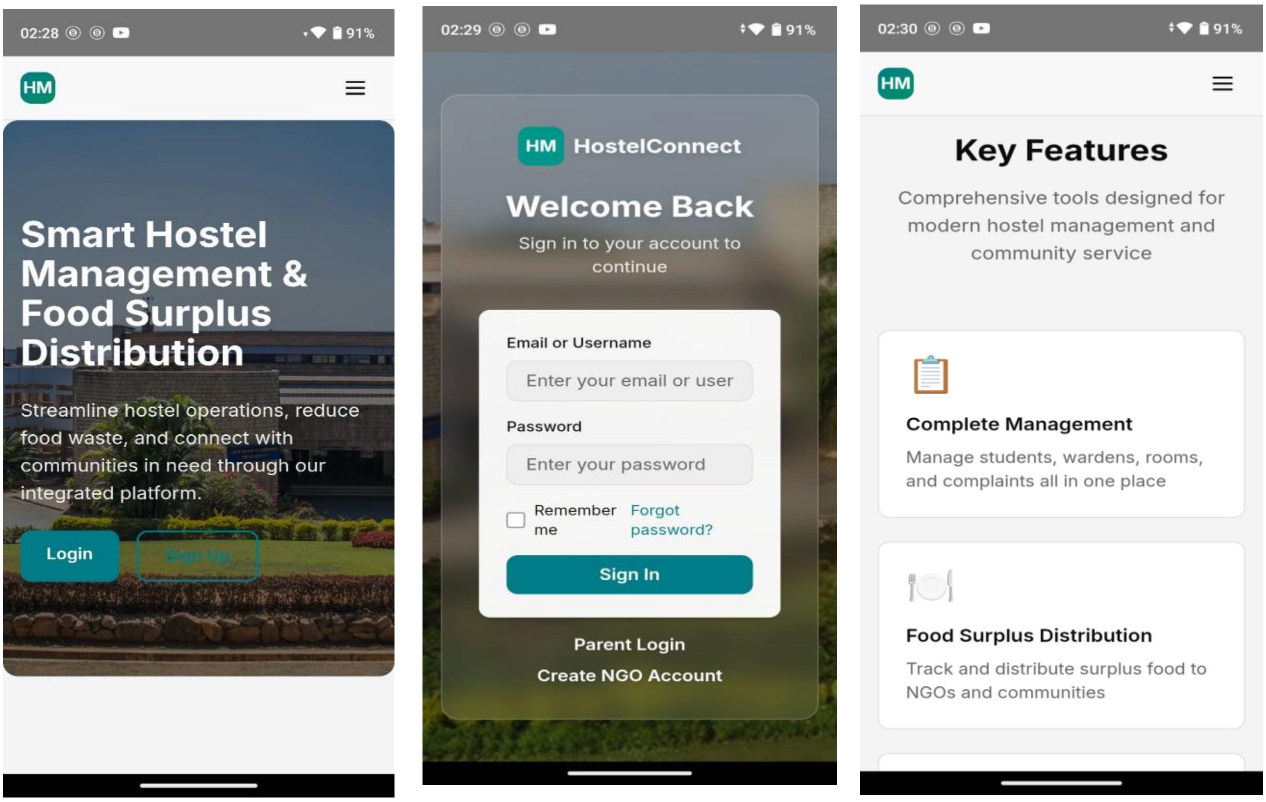
SDMCET Mess

Posted: 10:45 PM

Expires: 2025-12-15

Claim

Fig 8.2 Various Dashboard



02:28 91%

HM

Smart Hostel Management & Food Surplus Distribution

Streamline hostel operations, reduce food waste, and connect with communities in need through our integrated platform.

Login Sign Up

02:29 91%

HM HostelConnect

Welcome Back

Sign in to your account to continue

Email or Username

Enter your email or user

Password

Enter your password

□ Remember me Forgot password?

Sign In

Parent Login

Create NGO Account

02:30 91%

HM

Key Features

Comprehensive tools designed for modern hostel management and community service

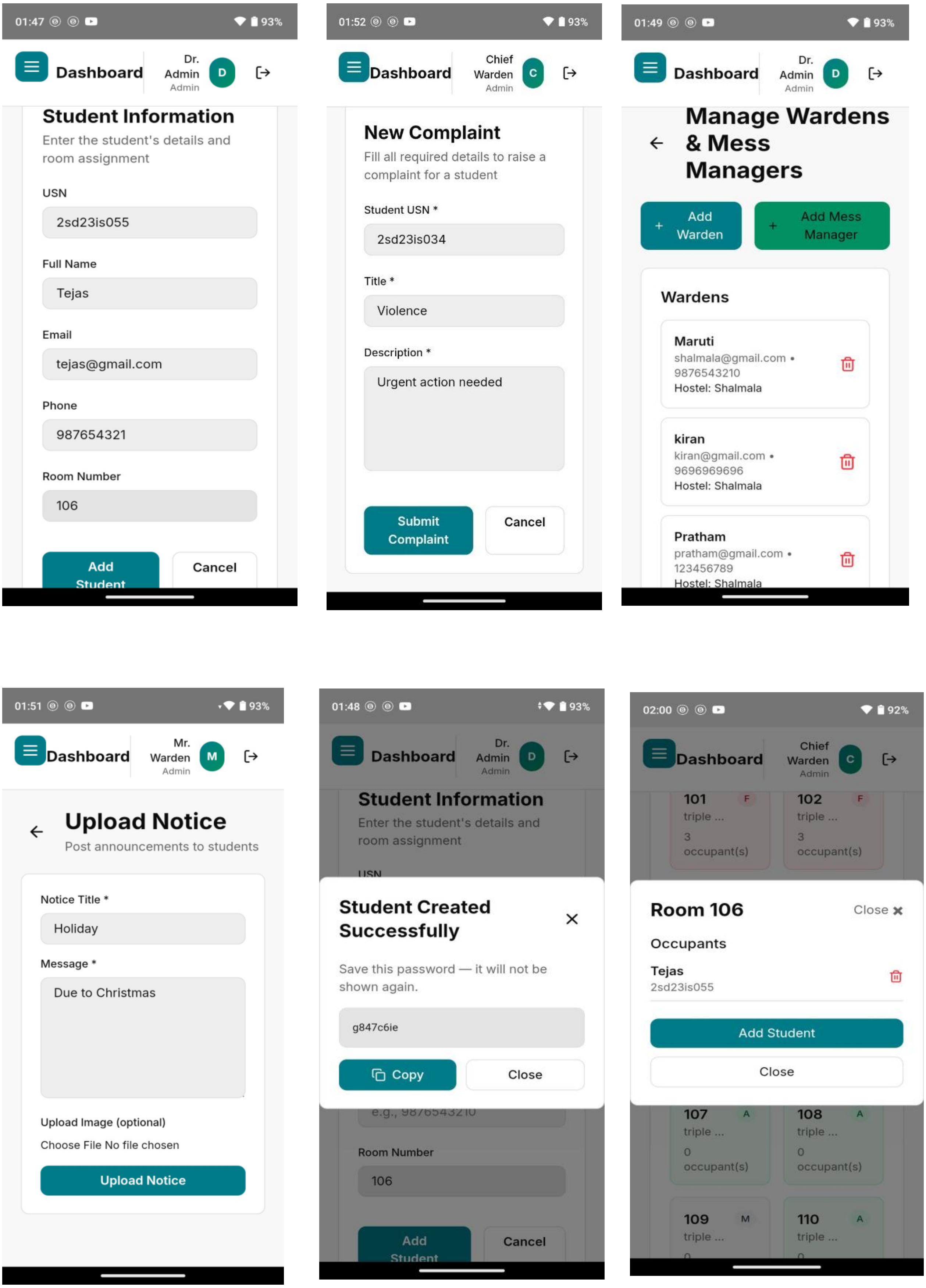
Complete Management

Manage students, wardens, rooms, and complaints all in one place

Food Surplus Distribution

Track and distribute surplus food to NGOs and communities

Fig 8.3 Home Screens



Student Information

Enter the student's details and room assignment

USN: 2sd23is055

Full Name: Tejas

Email: tejas@gmail.com

Phone: 987654321

Room Number: 106

New Complaint

Fill all required details to raise a complaint for a student

Student USN *: 2sd23is034

Title *: Violence

Description *: Urgent action needed

Manage Wardens & Mess Managers

Wardens

Maruti
shalmala@gmail.com • 9876543210
Hostel: Shalmala

kiran
kiran@gmail.com • 9696969696
Hostel: Shalmala

Pratham
pratham@gmail.com • 123456789
Hostel: Shalmala

Upload Notice

Post announcements to students

Notice Title *: Holiday

Message *: Due to Christmas

Upload Image (optional)

Choose File No file chosen

Student Information

Enter the student's details and room assignment

USN

Student Created Successfully

Save this password — it will not be shown again.

g847c6ie

e.g., 9876543210

Room Number: 106

101 F triple ... 3 occupant(s)	102 F triple ... 3 occupant(s)

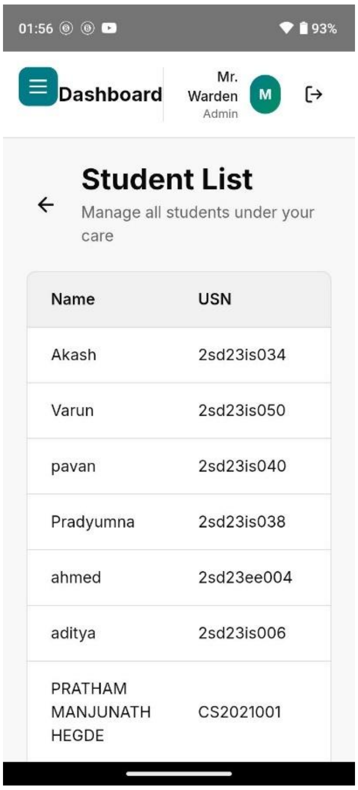
Room 106

Occupants

Tejas
2sd23is055

107 A triple ... 0 occupant(s)	108 A triple ... 0 occupant(s)
109 M triple ... 0	110 A triple ... 0

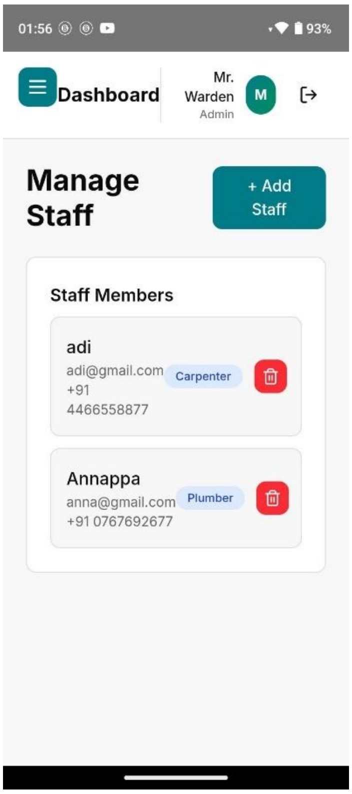
Fig 8.4 Chief Warden Screen



Student List

Manage all students under your care

Name	USN
Akash	2sd23is034
Varun	2sd23is050
pavan	2sd23is040
Pradyumna	2sd23is038
ahmed	2sd23ee004
aditya	2sd23is006
PRATHAM MANJUNATH HEGDE	CS2021001

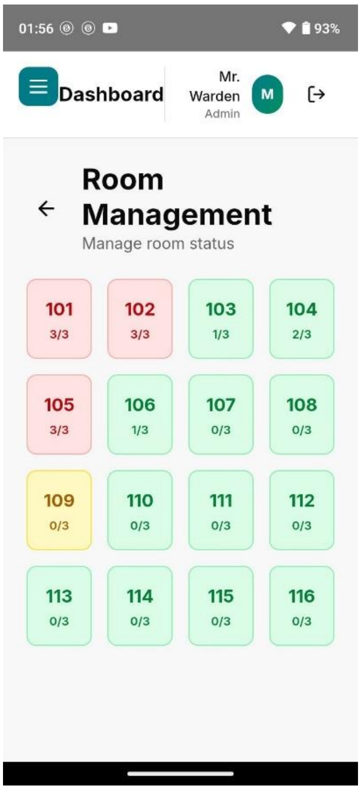


Manage Staff

Staff Members

adi
adi@gmail.com
+91 4466558877
Carpenter

Annappa
anna@gmail.com
+91 0767692677
Plumber

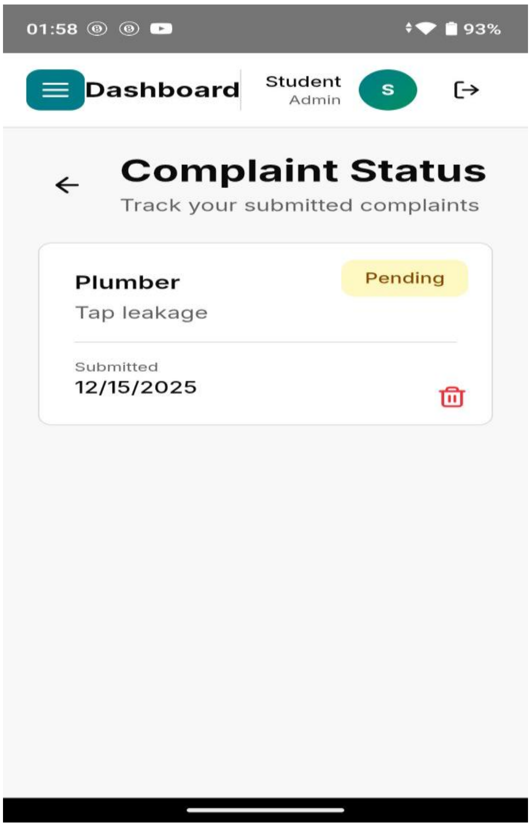


Room Management

Manage room status

101 3/3	102 3/3	103 1/3	104 2/3
105 3/3	106 1/3	107 0/3	108 0/3
109 0/3	110 0/3	111 0/3	112 0/3
113 0/3	114 0/3	115 0/3	116 0/3

Fig 8.5 Warden Screen

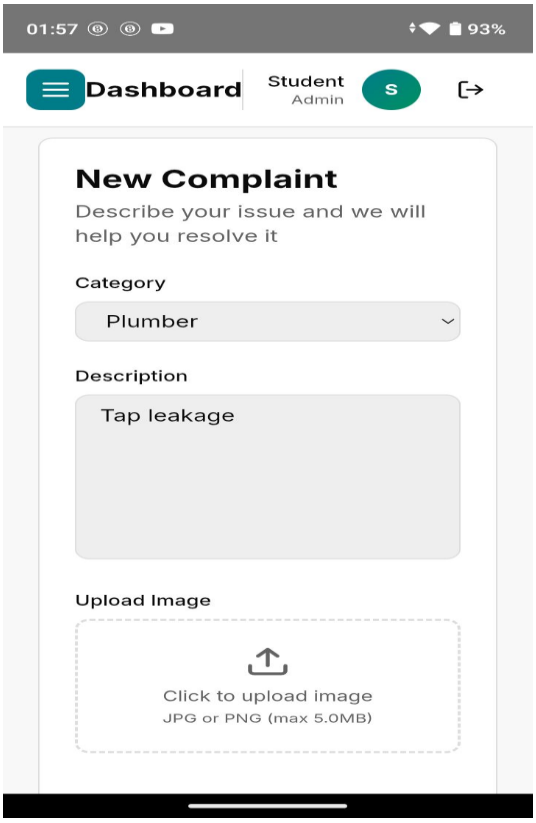


Complaint Status

Track your submitted complaints

Plumber
Pending
Tap leakage

Submitted
12/15/2025



New Complaint

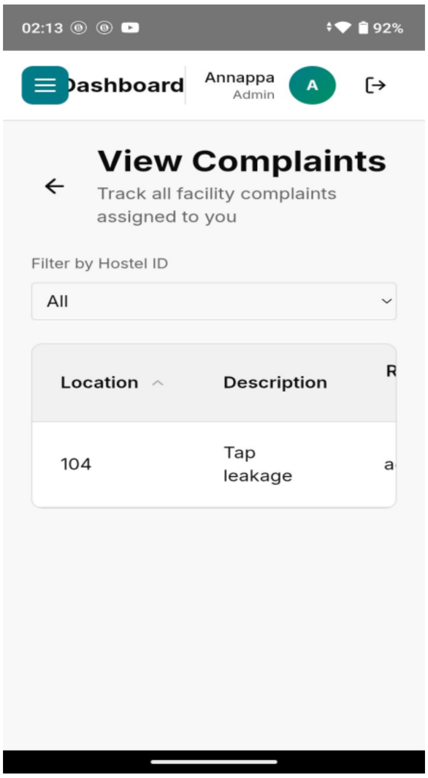
Describe your issue and we will help you resolve it

Category
Plumber

Description
Tap leakage

Upload Image
Click to upload image
JPG or PNG (max 5.0MB)

Fig 8.5 Student Screen



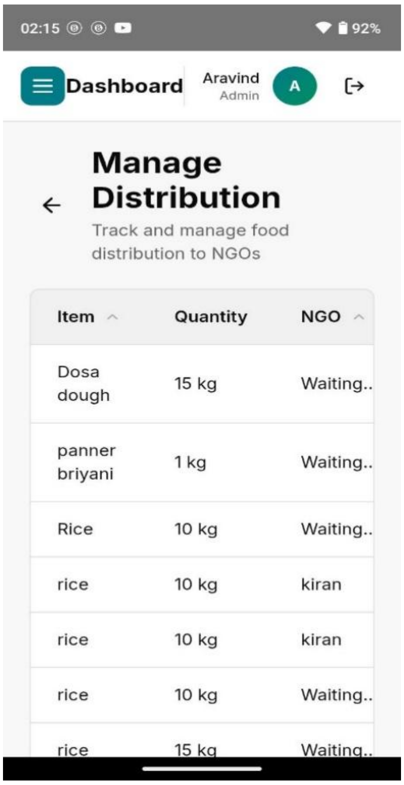
View Complaints

Track all facility complaints assigned to you

Filter by Hostel ID

All

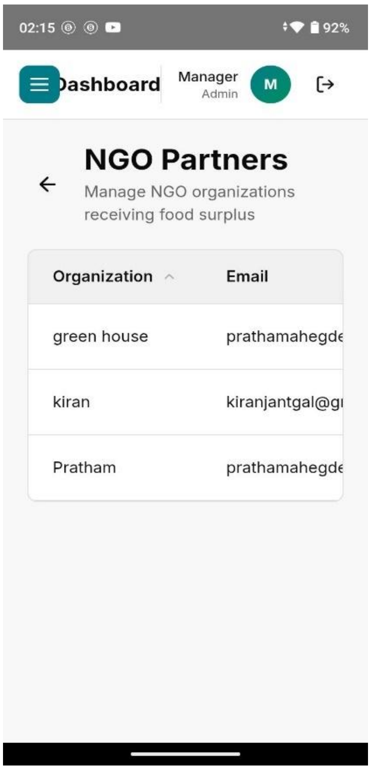
Location	Description	R
104	Tap leakage	a



Manage Distribution

Track and manage food distribution to NGOs

Item	Quantity	NGO
Dosa dough	15 kg	Waiting..
panner briyani	1 kg	Waiting..
Rice	10 kg	Waiting..
rice	10 kg	kiran
rice	10 kg	kiran
rice	10 kg	Waiting..
rice	15 kg	Waiting..

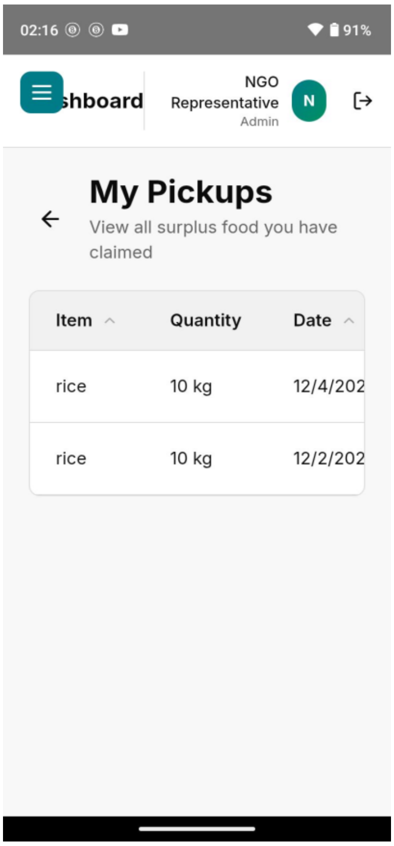


NGO Partners

Manage NGO organizations receiving food surplus

Organization	Email
green house	prathamahegde
kiran	kiranjantgal@gı
Pratham	prathamahegde

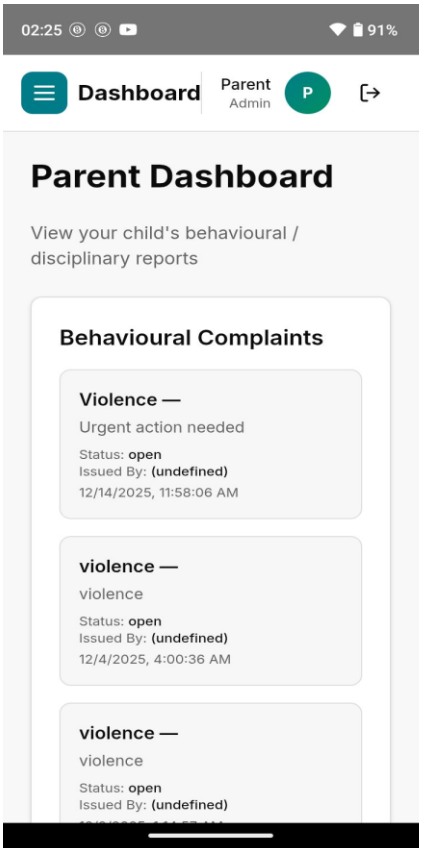
Fig 8.6 Staff and Mess Screen



My Pickups

View all surplus food you have claimed

Item	Quantity	Date
rice	10 kg	12/4/202
rice	10 kg	12/2/202



Parent Dashboard

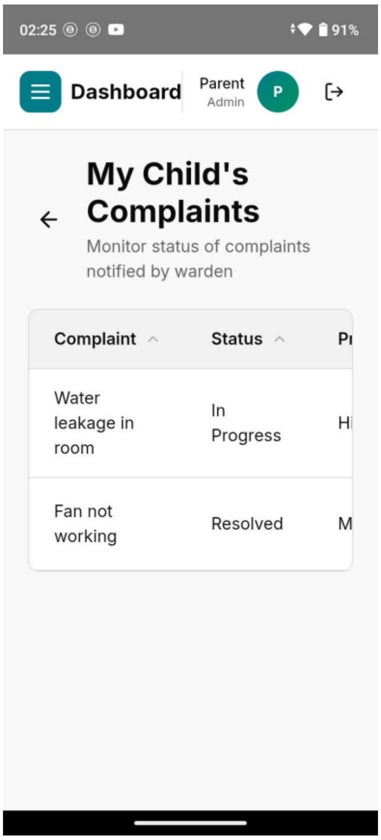
View your child's behavioural / disciplinary reports

Behavioural Complaints

Violence —
Urgent action needed
Status: **open**
Issued By: **(undefined)**
12/14/2025, 11:58:06 AM

violence —
violence
Status: **open**
Issued By: **(undefined)**
12/4/2025, 4:00:36 AM

violence —
violence
Status: **open**
Issued By: **(undefined)**



My Child's Complaints

Monitor status of complaints notified by warden

Complaint	Status	Pı
Water leakage in room	In Progress	H
Fan not working	Resolved	M

Fig 8.7 Ngo and Parents Screen

9 APPLICATION

- **Efficient Hostel Room Management:**

Enables digital room allocation and real-time occupancy tracking, helping hostel authorities manage rooms effectively.

- **Online Complaint Handling System:**

Students can raise hostel-related complaints, and staff can resolve them efficiently, improving maintenance and service quality.

- **Parent Monitoring and Transparency:**

Parents can view student details, room information, and complaint status in a read-only mode, ensuring transparency and trust.

- **Food Surplus Distribution:**

Mess managers can post surplus food details, and NGOs can claim and collect food, reducing wastage and supporting social responsibility.

- **Staff and Warden Coordination:**

Wardens and staff can coordinate hostel operations digitally, ensuring quicker response to student needs.

- **Institutional Sustainability Support:**

Promotes eco-friendly practices by minimizing food waste and encouraging responsible resource management.

- **Multi-Hostel Management:**

The platform can be extended to manage multiple hostels under a single institution with centralized control.

- **Educational Institution Deployment:**

Suitable for colleges, universities, and residential institutions seeking a modern and efficient hostel management solution.

10 SUSTAINABLE DEVELOPMENT GOALS (SDGs) ALIGNMENT

- **Introduction / Problem Statement**

The *Hostel Management and Food Surplus Distribution Platform* addresses multiple Sustainable Development Goals (SDGs) by improving institutional efficiency, transparency, and sustainability. Traditional hostel management systems suffer from poor coordination, manual processes, and food wastage in hostel messes. These challenges directly relate to responsible resource utilization, student well-being, and sustainable community practices.

This project primarily contributes to SDG 12 (Responsible Consumption and Production) and SDG 4 (Quality Education), while also supporting SDG 11 (Sustainable Cities and Communities) and SDG 17 (Partnerships for the Goals).

- **Objectives / Aims (SDG Linked)**

➢ To digitize hostel operations and reduce paper-based processes, supporting SDG 12: Responsible Consumption and Production.

➢ To provide a structured and transparent hostel environment that supports student well-being and learning, contributing to SDG 4: Quality Education.

➢ To reduce food wastage by redistributing surplus food to NGOs, aligning with SDG 2: Zero Hunger and SDG 12.

➢ To enable collaboration between institutions and NGOs, supporting SDG 17: Partnerships for the Goals.

- **Methodology / Approach**

The project follows a role-based digital approach using modern web technologies to align with SDG principles. Hostel operations such as room allocation, complaints, and food management are digitized to reduce resource wastage and improve efficiency. The food surplus module directly supports sustainable consumption by ensuring excess food is redistributed instead of discarded, aligning with global sustainability practices.

- **Impact / Expected Outcomes**

➢ Reduced food wastage through structured surplus redistribution (SDG 12, SDG 2).

- Improved hostel transparency and student support systems (SDG 4).
- Better institutional resource management and sustainable campus practices (SDG 11).
- Strengthened collaboration between educational institutions and NGOs (SDG 17).

- **Conclusion / Future Work**

The project demonstrates how digital solutions can effectively contribute to sustainable development goals within educational institutions. Future enhancements such as analytics for food waste reduction, multi-institution deployment, and NGO expansion can further strengthen its contribution to sustainability, education quality, and community development.

Table 10.1 SDG Mapping Table

SDG No. & Goal	Project Feature / Objective	Impact / Contribution
SDG 2 – Zero Hunger	Food surplus distribution to NGOs	Reduces food wastage and supports hunger relief
SDG 4 – Quality Education	Efficient hostel management & student support	Improves living conditions that support learning
SDG 9 – Industry, Innovation and Infrastructure	Web-based, role-driven digital platform	Encourages innovation using scalable technology
SDG 11 – Sustainable Cities and Communities	Smart hostel operations and transparency	Supports sustainable and well-managed communities
SDG 12 – Responsible Consumption and Production	Paperless operations & surplus food management	Promotes sustainable resource utilization
SDG 17 – Partnerships for the Goals	Collaboration with NGOs for food distribution	Encourages institutional and social partnerships

11 CONCLUSION

The *Hostel Management and Food Surplus Distribution Platform* is a comprehensive web-based system developed to modernize and streamline hostel operations in educational institutions. By leveraging modern frontend technologies such as React and Tailwind CSS, the platform provides an efficient, user-friendly, and role-based solution for managing rooms, complaints, staff coordination, and parent monitoring.

The integration of a food surplus distribution module further enhances the system by addressing food wastage and promoting social responsibility. Through structured coordination between mess managers and NGOs, the platform supports sustainable resource utilization while contributing to community welfare. Role-based authentication and controlled access ensure system security, transparency, and accountability across all stakeholders.

Beyond its core functionalities, the platform improves operational efficiency, enhances communication, and builds trust between hostel authorities, students, and parents. With its scalable design and clean user interface, the system is well-suited for institutional deployment and future backend integration.

In the future, the platform can be extended with advanced analytics, automated notifications, mobile application support, and integration with institutional systems, further improving usability and impact. Overall, the project demonstrates an effective application of technology in improving hostel management while fostering sustainability and social responsibility.

12 FUTURE SCOPE

- **AI-Based Room and Complaint Analytics:**

  Artificial intelligence can be integrated to analyze room occupancy patterns and complaint trends, helping hostel authorities improve decision-making and resource allocation.

- **Multi-Institution Expansion:**

  The platform can be extended to support multiple institutions and hostels across different regions, with multilingual support for wider adoption.

- **Integration with Institutional Systems:**

  The system can be integrated with college ERP systems for student records, attendance, and fee management to provide a unified campus solution.

- **Automated Notifications and Alerts:**

  Real-time notifications for complaints, room changes, food pickup schedules, and announcements can improve responsiveness and communication.

- **Advanced Reporting and Analytics:**

  Dashboards with detailed reports on complaints, room utilization, and food surplus distribution can help administrators track performance and sustainability impact.

- **Mobile Application Support:**

  A dedicated mobile application can be developed to enhance accessibility for students, staff, and parents.

- **Smart Sustainability Tracking:**

  The platform can include analytics to measure food wastage reduction and social impact, supporting institutional sustainability goals.

REFERENCES

[1]. IEEE – Smart Hostel Management System Using Web Technologies

This paper discusses the design and implementation of digital hostel management systems to improve efficiency and transparency.

https://ieeexplore.ieee.org/search/searchresult.jsp?queryText=smart%20hostel%20management%20system

[2]. IEEE – Role-Based Access Control in Web Applications

Authored by Sandhu et al., this study explains secure role-based authentication mechanisms applicable to multi-user systems.

https://ieeexplore.ieee.org/document/613372

[3]. IEEE – Complaint Management Systems for Educational Institutions

This work highlights digital complaint handling frameworks to enhance accountability and response time.

https://ieeexplore.ieee.org/search/searchresult.jsp?queryText=complaint%20management%20system%20educational%20institutions

[4]. IEEE – Web-Based Campus Management Systems

Presented by Kumar et al., the paper focuses on centralized platforms for managing campus facilities and student services.

https://ieeexplore.ieee.org/search/searchresult.jsp?queryText=web%20based%20campus%20management%20system

[5]. IEEE – Technology-Driven Sustainability Solutions in Institutions

This study explores the use of digital platforms to reduce resource wastage and promote sustainability.

https://ieeexplore.ieee.org/search/searchresult.jsp?queryText=sustainability%20digital%20solutions%20institutions

[6]. React Official Documentation

https://react.dev
Reference for component-based frontend development and state management.

[7]. Tailwind CSS Documentation

https://tailwindcss.com
Used for responsive and modern UI design principles.

[8]. Smart Campus and Hostel Portals (Case Studies)

Various institutional case studies demonstrating digital hostel and campus management solutions.

[9]. Food Waste Management in Educational Institutions

Research studies and reports emphasizing structured food surplus redistribution models.