HOSTEL MANAGEMENT.

DONE BY

OMEGA S
RITHANYAA G V
YAZEEN RIZWAN A

COIMBATORE INSTITUTE OF TECHNOLOGY

(A GOVERNMENT AIDED AUTONOMOUS INSTITUTION AFFILIATED TO ANNA UNIVERSITY)

COIMBATORE – 641 014

TAMILNADU, INDIA

ABSTRACT :

The **Hostel Management System** is a web-based application built using the MEAN stack (MongoDB, Express.js, Angular, and Node.js) to make managing hostel tasks easier. Traditionally, hostels rely on manual methods like paper records or spreadsheets to handle things like room assignments, rent tracking, and keeping track of allotees. These methods can be slow, lead to mistakes, and cause confusion. This system helps solve those problems by automating key tasks, making the entire process more efficient and reliable for hostel administrators.

The **goal** of this project is to develop a platform that can automate important tasks such as managing rooms, tracking allotees, processing rent payments, and creating reports. The system will also have role-based access, meaning administrators will have full access to all features, while staff will have limited access depending on their role. The expected outcome is to improve how hostels are managed by reducing errors, saving time, and making it easier to get accurate information whenever needed.

In the **current system**, many hostels still use manual methods or basic software like spreadsheets, which makes managing rooms, rent, and allotee details difficult and slow. It can also lead to mistakes and is not suitable for large hostels with many allotees. Additionally, generating reports or looking up past data can be a hassle, making it harder to manage daily operations efficiently.

The **new system** will be built using the **MEAN stack**, which includes **Angular** for the front-end, **Node.js** and **Express.js** for the back-end, and **MongoDB** for the database. This combination of technologies provides a flexible and scalable solution for managing hostels. Key features include user management, room and allotee management, rent tracking, and reporting. The system will use **Netlify** to host the front-end, making the deployment process smooth and ensuring quick updates when needed. The back-end and database will be hosted separately, but the integration with Netlify will make the system run seamlessly for users.

PROBLEM STATEMENT :

Managing hostels can be challenging because it involves many manual and repetitive tasks like keeping track of room availability, handling rent payments, and maintaining hostel facilities. Traditional methods, such as using paper records or simple spreadsheets, can lead to problems like errors, inefficiencies, and inconsistent data. Without a central system, administrators often struggle to access important information quickly and manage day-to-day operations effectively. This can result in delays and mismanagement, especially in larger hostels. To solve these issues, the Hostel Management System is developed as a web application that automates these tasks, making hostel management more accurate, efficient, and well-organized. This system aims to simplify the process, reduce manual effort, and improve the overall administration of hostels.

OBJECTIVE & SCOPE :

The objective of the Hostel Management System is to simplify and automate the various tasks involved in managing a hostel, reducing the manual workload and improving overall efficiency. The system aims to streamline operations such as room and bed allocation, tracking allotee details, handling rent payments, and maintaining accurate financial records. By incorporating role-based access control, the system ensures that administrators have full access to all functionalities while user can perform their specific tasks.

Scope of the Project:

- Develop a user-friendly web application for hostel management.
- Implement user authentication and role-based access control (admin and staff).
- Manage rooms, beds, and hostel facilities efficiently.
- Track room assignments and personal details.
- Provide a centralized system for easy access to information and decision-making.

USERS OF THE SYSTEM :

- ***Administrator:*** Has full access to all functionalities, Manage user accounts (add, edit, remove users), Oversee room management (add, edit, remove room details), Respond to user queries and provide support.
- ***User:*** Limited access, View available room packages and details, Check room availability for booking, Update personal profile information, Submit queries related to room bookings or account issues.

FUNCTIONAL REQUIREMENTS :

Administrator Functional Requirements:

1. **Login**: The administrator logs into the system.
2. **Dashboard**: The main interface where the admin can choose functionalities.
3. **User Management**:
    - **Add Student**: Admin can add new students.
    - **View Users**: Admin can view existing users.

- **Edit Student**: Admin can edit details of existing students.
- **Remove Student**: Admin can remove students from the system.

4. **Room Management**:
    - **Update Room Price**: Admin can update pricing details for rooms.
    - **View Room Details**: Admin can see details about rooms and their availability.
5. **Reply to Queries**: Admin can respond to user queries, which may be linked to both user and room management functions.
6. **Update Room Availability**: Admin can manage room availability based on student assignments or other factors.

User Functional Requirements:

1. **Login**: The user logs into the system.
2. **Dashboard**: The main interface where the user can choose functionalities.
3. **Profile Management**:
    - **Update Profile**: Users can update their personal information.
4. **Room Management**:
    - **View Room Packages**: Users can browse available room packages.
    - **View Room Availability**: Users can check which rooms are available for booking.
5. **Ask Query**: Users can submit queries related to their profile or room management.

MODULES FORMATION:

1. User Module: Handles user authentication, role management, and user profile management.

2. Room Management Module: Manages details about rooms, bed assignments, and availability.

3. Allotee Management Module: Keeps track of allotee details, their room assignments, and stay duration.

4. Reporting Module: Generates reports on occupancy, financial status, and other key metrics.

HARDWARE & SOFTWARE REQUIREMENTS :

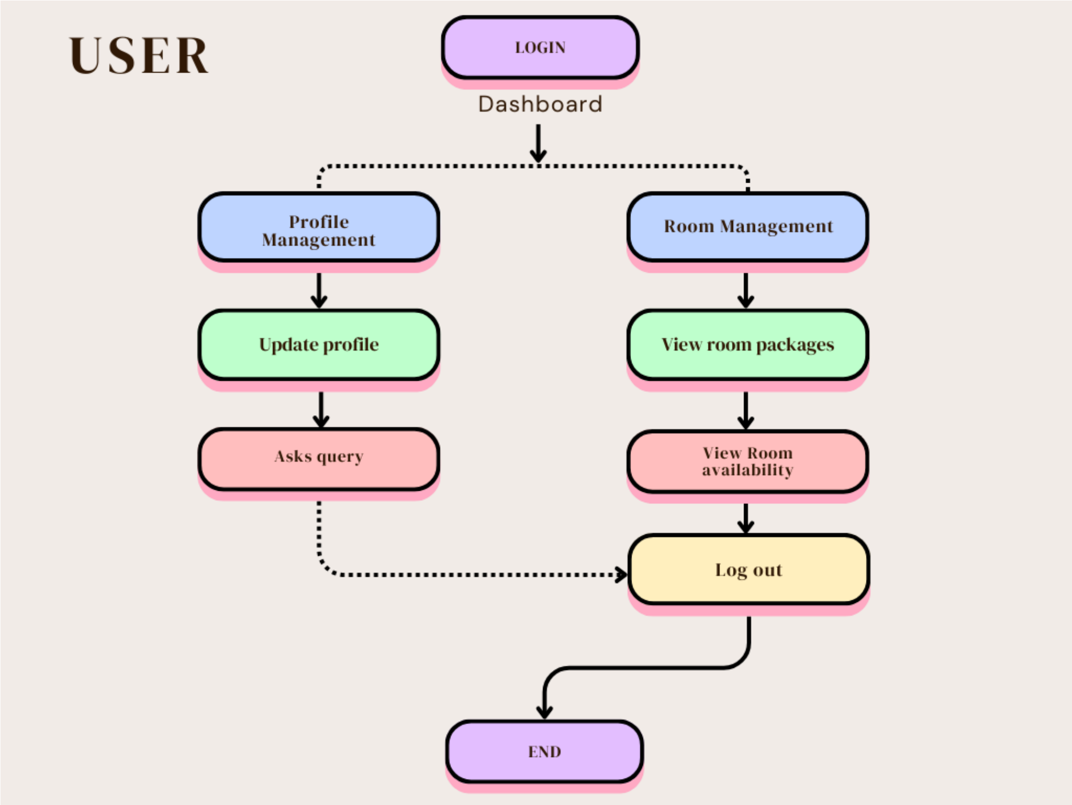
Hardware Requirements:

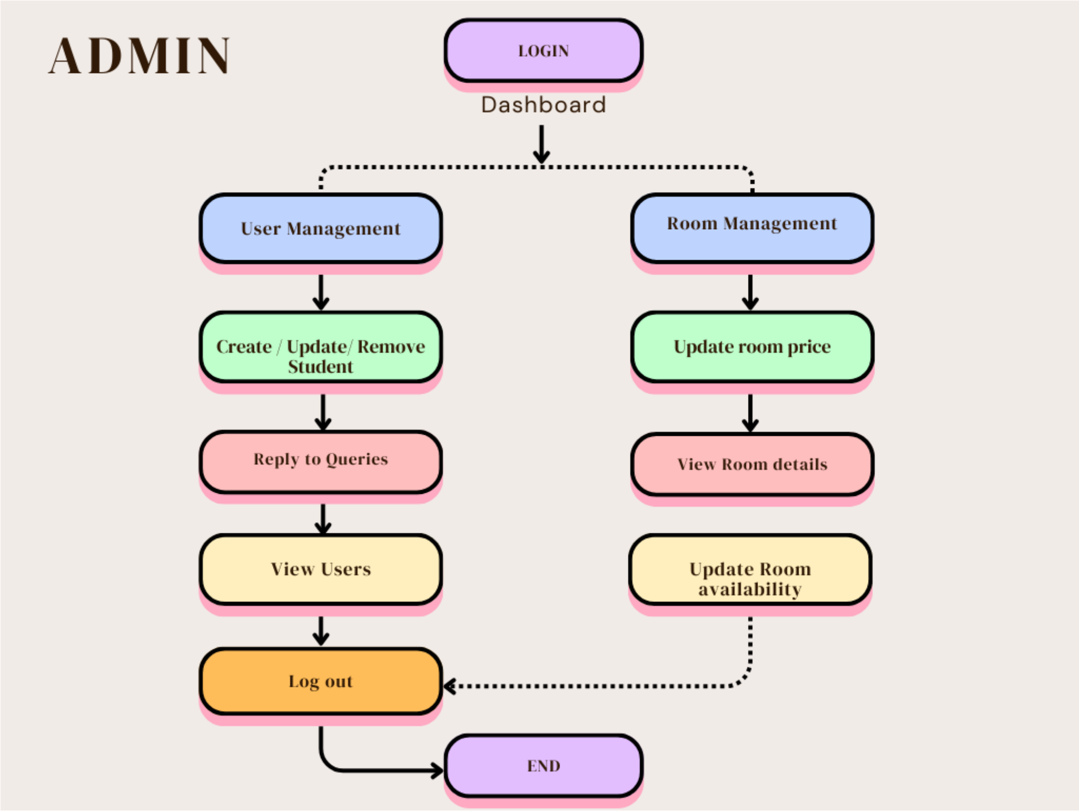
- A computer or server with the following specifications:
  - Processor: Minimum Dual-core CPU (Intel i3 or equivalent)
  - RAM: At least 8 GB
  - Storage: 100 GB of available hard drive space
  - Network: Stable internet connection for cloud access and deployment

Software Requirements:

- Operating System: Windows 10 or later, macOS, or Linux
- Web Browser: Google Chrome, Mozilla Firefox, or any modern web browser
- Development Environment:
  - Node.js (version 16)
  - Express.js (latest stable version)
  - Angular (latest stable version)
- Database: MongoDB for storing application data
- Additional Tools:
  - Code editor (e.g., Visual Studio Code)

USER WISE NAVIGATION DIAGRAM :


USER
LOGIN
Dashboard
Profile Management
Room Management
Update profile
View room packages
Asks query
View Room availability
Log out
END


ADMIN
LOGIN
Dashboard
User Management
Room Management
Create / Update/ Remove Student
Update room price
Reply to Queries
View Room details
View Users
Update Room availability
Log out
END