BANNARI AMMAN INSTITUTE OF TECHNOLOGY
An Autonomous Institution Affiliated to Anna University - Chennai, Accredited by NAAC with A+ Grade
Sathyamangalam - 638401 Erode District, Tamil Nadu, India

Student Name: RITTHYK M

Seat No: 256

Project ID: 16

Project title: Hostel maintenance

Components - Tech Stack Used:

Components	Tech Stack
Backend	Html, CSS, JavaScript
Frontend	Python, Django
Database	MySQL

Implementation Timeline:

Project Phases	Deadline	Status
Planning and Requirement Gathering	02/05/2024	Under review
Design and Prototyping		Not started
DB Designing and implementation		Not started
Backend Development		In progress
Integration and Testing		Not started
Deployment		Not started

Project Overview Document

Stage 1: Planning and Requirement Gathering

1. **Gather project requirements from stakeholders**: The project team collected requirements from various stakeholders, including students, faculty, administrative staff, and IT department.
2. **Identified challenges in hostel maintenance:** Challenges such as managing hostel occupancy, updating hostel capacities, handling room type changes, staff attendance tracking, and handling leave requests were identified.
3. **Defined purpose, scope, and business context:** The purpose of the project is to develop a comprehensive hostel maintenance application to efficiently manage hostel-related tasks and improve administrative processes. The scope includes various functionalities such as updating hostel occupancy, managing staff and student capacities, handling room type changes, tracking staff attendance, managing leave requests, handling complaints, and sending hostel-wise announcements. The business context focuses on enhancing operational efficiency and communication within the hostel administration.
4. **Established user personas and user stories:** User personas, including hostel administrators, staff, and students, were defined. User stories were created to capture their requirements effectively.

Stage 2: Design and Prototyping

1. **Designed architecture:** The architecture of the hostel maintenance application was designed to ensure scalability, reliability, and efficiency.
2. **Created wireframes and prototypes:** Wireframes and prototypes were developed for the user interface to visualize the system's design and functionality.
3. **Reviewed and finalized design decisions:** Design decisions were reviewed and finalized based on feedback from stakeholders to ensure alignment with project goals.

Stage 3: DB Designing

1. **Designed database schema:** The database schema was designed to store hostel occupancy data, staff and student capacities, attendance records, leave requests, complaints, and announcements efficiently.
2. **Ensured scalability and efficiency:** Measures were taken to ensure that the database design is scalable and efficient to handle large volumes of data.
3. **Established relationships between entities:** Relationships between different database entities were established to maintain data integrity and consistency.

Stage 4: Backend Implementation

1. Implemented user authentication: Google OAuth was implemented for secure user authentication.

2. Developed functionality for hostel-wise strength update: Forms or interfaces were developed for administrators to update the number of occupants in each hostel.
3. Implemented functionality for total hostel capacity and students' capacity: Functionality was implemented to calculate and update the total capacity of each hostel and the capacity for students based on occupancy data.
4. Implemented functionality for changing from day scholar to hostellers and vice versa: Forms were developed for students to request status changes, and administrators can approve or decline them.
5. Implemented functionality for changing hostel rooms: Functionality was implemented to handle room type change requests from students and approve them based on availability.
6. Implemented functionality for staff attendance: Functionality was implemented to record staff attendance using a biometric system or an attendance tracking application and store attendance data in the database.
7. Developed functionality for vacancy details update: Functionality was implemented to update vacancy details for staff and students in each hostel based on changes in occupancy.
8. Implemented functionality for hostel attendance: Functionality was implemented to record hostel attendance using biometric thumb registration and track students' attendance status.
9. Developed functionality for students' leave portal: An online portal was developed for students to apply for leave, and administrators can approve or decline leave requests. The system automatically cancels leave if the student does not register their biometric thumb.
10. Implemented functionality for students' complaints through online: An online portal was developed for students to submit complaints or maintenance requests, which can be tracked and resolved by the hostel administration.
11. Developed functionality for mailing students hostel-wise: Functionality was implemented to send emails to students hostel-wise for important announcements or updates.

Stage 5: Testing & Implementation

1. Conducted tests: Unit tests, integration tests, and system tests were conducted to ensure the functionality and reliability of the hostel maintenance application.
2. Fixed bugs and addressed issues: Bugs and issues identified during testing were fixed to improve system performance and user experience.
3. Deployed system: The hostel maintenance application was deployed to production for use by hostel administrators, staff, and students.
4. Provided training and support: Training and support were provided to users to help them effectively use the hostel maintenance application.
5. Monitored system performance: The system's performance was monitored, and any operational issues were addressed promptly to ensure smooth operation.

Problem Statement

The decentralized nature of hostel maintenance tasks within educational institutions leads to several challenges, including:

- Inefficient management of hostel occupancy
- Difficulty in updating hostel capacities and handling room type changes
- Challenges in tracking staff attendance and managing leave requests
- Fragmented communication and difficulty in handling complaints and announcements

Project Flow

1. **Purpose**

   To develop a comprehensive hostel maintenance application to efficiently manage hostel-related tasks and improve administrative processes.

2. **Scope**

   The system includes various functionalities such as updating hostel occupancy, managing staff and student capacities, handling room type changes, tracking staff attendance, managing leave requests, handling complaints, and sending hostel-wise announcements.

3. **Business Context**

   The hostel maintenance application aims to enhance operational efficiency and communication within the hostel administration, thus improving the overall hostel management process. Primary stakeholders include hostel administrators, staff, and students.

4. **Consideration**
   - All users possess active Google accounts for authentication.
   - Users have regular access to internet-enabled devices.
   - Dependencies
   - Integration with Google OAuth for user authentication.
   - Consistent performance and availability of the existing email server.

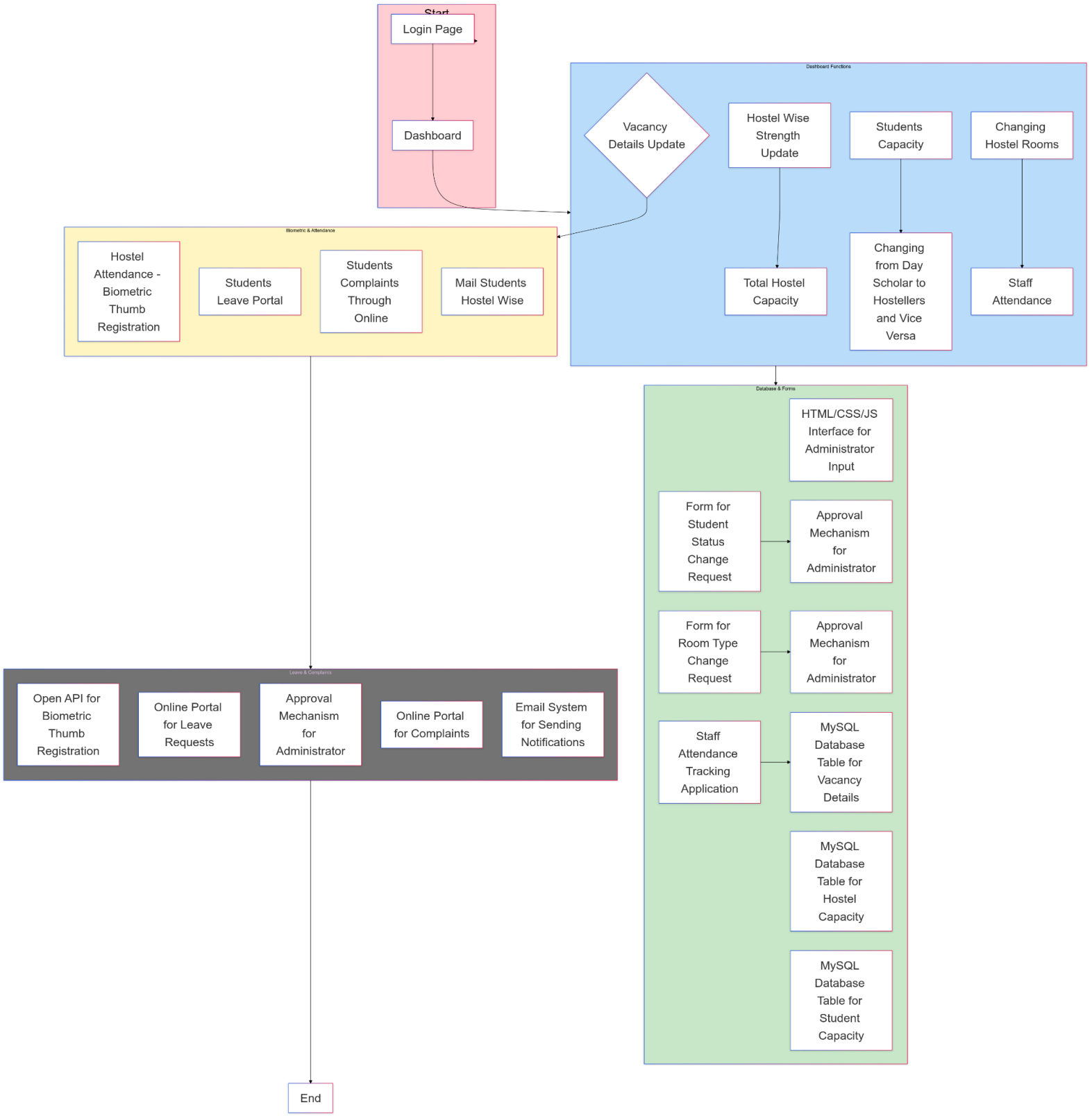
5. **User Personas**
   - Hostel Administrator: Responsible for managing hostel-related tasks, including occupancy updates, capacity management, attendance tracking, leave approval, and complaint resolution.
   - Staff: Responsible for supporting hostel administration tasks and handling various

hostel-related activities.

- Students: Users of the hostel maintenance application who interact with the system to request room changes, apply for leave, submit complaints, and receive announcements.

6. **User Stories**
    - As a hostel administrator, I want to efficiently manage hostel occupancy, update hostel capacities, track staff attendance, and handle leave requests and complaints to ensure smooth hostel operations.
    - As a staff member, I want to support hostel administration tasks effectively and handle various hostel-related activities efficiently.
    - As a student, I want to easily request room changes, apply for leave, submit complaints, and receive hostel-wise announcements to stay informed and engaged in hostel activities.
7. **Functional Requirements**
    - User Authentication: Secure login using Google OAuth.
    - Hostel Wise Strength Update: Functionality for updating the number of occupants in each hostel.
    - Total Hostel Capacity: Functionality for calculating and updating the total capacity of each hostel based on occupancy data.
    - Students Capacity: Functionality for calculating and updating the capacity for students in each hostel based on occupancy data.
    - Changing from Day Scholar to Hostellers and Vice Versa: Functionality for updating the status of a student from day scholar to hosteller or vice versa.
    - Changing Hostel Rooms: Functionality for handling room type change requests from students and approving them based on availability.
    - Staff Attendance: Functionality for recording staff attendance using a biometric system or an attendance tracking application.
    - Vacancy Details Update: Functionality for updating vacancy details for staff and students in each hostel based on changes in occupancy.
    - Hostel Attendance - Biometric Thumb Registration: Functionality for recording hostel attendance using biometric thumb registration and tracking students' attendance status.
    - Students Leave Portal: Online portal for students to apply for leave, and administrators can approve or decline leave requests. The system automatically cancels leave if the student does not register their biometric thumb.
    - Students Complaints Through Online: Online portal for students to submit complaints or maintenance requests, which can be tracked and resolved by the hostel administration.
    - Mail Students Hostel Wise: Functionality for sending emails to students hostel-wise for important announcements or updates.

FLOW CHART:


Start
Login Page
Dashboard
Dashboard Functions
Vacancy Details Update
Hostel Wise Strength Update
Students Capacity
Changing Hostel Rooms
Total Hostel Capacity
Changing from Day Scholar to Hostellers and Vice Versa
Staff Attendance
Biometric & Attendance
Hostel Attendance - Biometric Thumb Registration
Students Leave Portal
Students Complaints Through Online
Mail Students Hostel Wise
Database & Forms
HTML/CSS/JS Interface for Administrator Input
Form for Student Status Change Request
Approval Mechanism for Administrator
Form for Room Type Change Request
Approval Mechanism for Administrator
Staff Attendance Tracking Application
MySQL Database Table for Vacancy Details
MySQL Database Table for Hostel Capacity
MySQL Database Table for Student Capacity
Leave & Complaints
Open API for Biometric Thumb Registration
Online Portal for Leave Requests
Approval Mechanism for Administrator
Online Portal for Complaints
Email System for Sending Notifications
End